Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXXVII, págs. 2045-2054 (artículo completo)
A. Cardona, L. Garelli, J.M. Gimenez, P.A. Kler, S. Márquez Damián, M.A. Storti (Eds.)
Santa Fe, 5-7 Noviembre 2019

EVALUACIÓN EMPÍRICA DE LA ROBUSTEZ DE DIFERENTES REDES NEURONALES USADAS PARA LA DETECCIÓN DE OBJETOS

EMPIRICAL EVALUATION OF THE ROBUSTNESS OF DIFFERENT NEURAL NETWORKS USED FOR OBJECT DETECTION

Matías Olmedo[a], Javier A. Redolfi[a,b], Diego González Dondo[a] y R. Gastón Araguás[a]

[a]*Centro de Investigación en Informática para la Ingeniería (CIII), Facultad Regional Córdoba, Universidad Tecnológica Nacional, Córdoba, Argentina*

[b]*Grupo de Investigación Sobre Aplicaciones Inteligentes (GISAI), Facultad Regional San Francisco, Universidad Tecnológica Nacional, San Francisco, Argentina*

Palabras clave: redes neuronales convolucionales, detección de objetos.

Resumen. Existen muchos algoritmos para la detección de objetos en imágenes pero dependiendo de las necesidades computacionales, velocidad de respuesta y condiciones de trabajo, resulta difícil seleccionar el que se ajuste a los requerimientos particulares. En este trabajo se presenta la evaluación de diferentes redes neuronales convolucionales aplicadas a la detección de objetos. Se exploran sus comportamientos bajo diferentes condiciones: cambios en los tamaños de objetos a detectar, en la iluminación y en los ambientes. Se evalúan sus tiempos de cómputo y la posibilidad de su uso en tiempo real. Los resultados demuestran la factibilidad del uso de estas redes para detección de objetos en ambientes industriales pero de los experimentos surgen una serie de conclusiones sobre las condiciones de funcionamiento necesarias para lograr resultados óptimos. Estas están relacionadas con la red a usar dependiendo de la velocidad, las condiciones de iluminación, el tamaño de los objetos y el entorno de trabajo. A futuro, se espera que alguno de estos algoritmos sea utilizado como parte de un sistema de seguridad industrial.

Keywords: convolutional neural networks, object detection

Abstract. There are many algorithms for object detection in images but depending on the computational needs, response speed and working conditions, it is difficult to select the one that fits the particular requeriments. In this work is presented the evaluation of different convolutional neural networks applied to object detection. Their behaviors under different conditions are explored: changes in the size of object to be detected, in illuminations and in environments. Its computation times and the posibility of his use in real time are evaluated. The results demostrates the feasibility of using these networks for object detection in industrial environments but from the experiments a series of conclusions about the operating conditions necessary to achieve optimal results arise. These are related to the network to be used depending on the speed, illumination conditions, object sizes and work environments. In the future, some of these algorithms are expected to be used as part of an industrial security system.

1. INTRODUCCIÓN

La visión por computadora tiene cada vez más aplicaciones en la industria, las cuales permiten automatizar tareas repetitivas, tediosas o peligrosas como por ejemplo control de calidad, manejo de inventario, ordenado de piezas, líneas de ensamblaje, seguridad de los operarios, etc. (Dunsmore, 2000; Hirano et al., 2006; Benhimane et al., 2008; Luan et al., 2018). En muchas de estas aplicaciones es necesario contar con un algoritmo que detecte los objetos de interés en las imágenes para su posterior procesamiento. Además, en determinados contextos, la detección de objetos utilizando visión por computadora es la única solución posible. Por ejemplo cuando se deben detectar objetos distantes, o simplemente cuando no se cuenta con otra información que la visual sobre los objetos a procesar.

En la literatura existen muchos algoritmos para la detección de objetos, pero debido a la cantidad de modelos existentes, a las necesidades computacionales de los mismos, a la velocidad de respuesta requerida y a las condiciones del entorno, resulta un tanto engorroso seleccionar el modelo que se ajuste a los requerimientos de trabajo particulares. Por ejemplo condiciones de iluminación (mañana, tarde y noche), gran diferencia de tamaño entre los objetos de interés o cuando se cambia el entorno de funcionamiento de los algoritmos (entrenamiento en un lugar y uso en otro lugar). Por otro lado, este tipo de técnicas se encuentran muy poco aplicadas en entornos industriales, donde los ambientes de trabajo son visualmente complejos.

La motivación de este trabajo es el estudio y evaluación de diferentes algoritmos para la detección de objetos en movimiento y su aplicación en ambientes industriales visualmente complejos usando diferentes redes neuronales convolucionales. Para ello exploraremos el comportamiento de la detección de objetos usando estas redes bajo diferentes condiciones: cambios en los tamaños de objetos a detectar, en la iluminación y en los ambientes (con muchos objetos, ruidosos, complejos). También evaluaremos los tiempos de cómputo de las distintas redes y analizaremos la posibilidad de su uso en condiciones de tiempo real. Además, para los experimentos construiremos un conjunto de datos etiquetados el cual pondremos a disponibilidad pública para una futura evaluación de estos tipos de algoritmos por parte de la comunidad.

2. ESTADO DEL ARTE

En el trabajo de Viola et al. (2001) se presenta uno de los primeros métodos que muestra resultados interesantes, donde la detección se realiza mediante el uso de clasificadores en cascada y descriptores rectangulares. Luego se proponen otros basados en nuevos descriptores como HOG (Dalal y Triggs, 2005), en modelos de partes deformables (Felzenszwalb et al., 2010) y también en modelos basados en bolsa de palabras (Farooq, 2016) y vectores de Fisher (Gokberk Cinbis et al., 2013). En la actualidad el aprendizaje profundo se encuentra en auge, en parte gracias a la disponibilidad de una gran cantidad de imágenes etiquetadas (Deng et al., 2009; Lin et al., 2014) y al poder de cómputo basado en Unidades de Procesamiento Gráfico (GPU, por sus siglas en inglés) o Unidades de Procesamiento Tensorial (TPU, por sus siglas en inglés). Los algoritmos o métodos que muestran mejores resultados son los basados en Redes Neuronales Convolucionales (CNN, por sus siglas en inglés), como por ejemplo R-CNN (Girshick et al., 2014), Faster R-CNN (Ren et al., 2015) y YOLO (Redmon et al., 2016).

Uno de los primeros trabajos basados en CNN es el de Girshick et al. (2014), en el cual se presenta la red R-CNN. Este método se basa primero en generar propuestas de regiones en donde puede haber objetos, aproximadamente unas 2000 propuestas por imagen y luego sobre estas regiones se computa un descriptor usando una CNN. Esta red convolucional fue entrenada en otro conjunto de datos para resolver el problema de clasificación de imágenes y sus primeras

capas convolucionales pueden ser utilizadas como descriptores de imágenes (Sharif Razavian et al., 2014). Luego se calculan los descriptores sobre las regiones y por último se clasifican dichos descriptores usando Máquinas de Soporte Vectorial (SVM, por sus siglas en inglés). Para la propuesta de regiones utilizan un algoritmo llamado Búsqueda Selectiva (SS, por sus siglas en inglés). El método SS (Uijlings et al., 2013) combina técnicas de segmentación y búsqueda exhaustiva para generar propuestas de regiones en donde haya alta probabilidad de encontrar objetos. Esta búsqueda consiste en generar ventanas en cada posición y escala de la imagen para encontrar posibles objetos; dicha técnica conocida como ventana deslizante termina siendo muy lenta debido a la gran cantidad de ventanas a evaluar. Para solucionarlo, primero realizan una segmentación de la imagen y luego una estrategia de combinación de segmentos en varias escalas usando diversas medidas de similitud. Aunque con el método R-CNN (Girshick et al., 2014) se obtienen una excelente exactitud en la detección de objetos, este tiene varias desventajas como por ejemplo que el entrenamiento es un proceso de varias etapas consecutivas, muy costoso en espacio en memoria, tiempo de cómputo y además la detección de objetos en tiempo de evaluación es muy lenta.

En el trabajo de Girshick (2015) se propone otro método conocido como Fast R-CNN, el cual corrige varias de las desventajas de R-CNN mejorando la velocidad y la exactitud. Esta red toma como entrada la imagen y un conjunto de propuesta de regiones con objetos. A diferencia de R-CNN que utilizaba una red entrenada sobre otro conjunto de datos para la extracción de descriptores y luego los clasificaba usando SVM, Fast R-CNN utiliza la misma red para la extracción de descriptores pero luego estos descriptores alimentan una secuencia de capas totalmente conectadas las cuales generan dos salidas: la primera que produce una probabilidad sobre las clases y la segunda salida es un vector de 4 elementos con un refinamiento en la posición original de la región en la imagen (x, y) y su tamaño. Las ventajas principales de esta red, aparte de producir mejor exactitud, son que el entrenamiento se realiza en una sola etapa, optimizando una sola función para todas las capas. Esto permite reducir los tiempos de entrenamiento en aproximadamente 9 veces y el tiempo de evaluación en 150 veces con respecto al método anterior.

El cuello de botella en el cómputo de la red R-CNN es el algoritmo SS usado para la propuesta de regiones. En el paper de Ren et al. (2015) se propone una red para la propuesta de regiones (RPN, por sus siglas en inglés). Esta red comparte los pesos para las propuestas de regiones con los pesos para la generación de descriptores, reduciendo casi a cero el tiempo necesario para la propuesta de regiones porque se aprovechan los cálculos realizados para la detección de objetos. En esta red se le agregan 2 capas a R-CNN, la primera que convierte la última capa convolucional compartida en un descriptor de 256 dimensiones y la segunda que genera una puntuación sobre cuan probable es que haya un objeto. Para el entrenamiento alternan entre ambas redes dejando una de ellas fija y modificando la otra. La segunda capa genera 2 vectores de salida, el primero que contiene 2 componentes para la probabilidad de objeto y no objeto y la segunda con 4 coordenadas que representan la posición y tamaño de la ventana. Además trabaja en 3 escalas y en 3 relaciones de aspecto, por lo tanto por cada punto a evaluar se generan 9 predicciones. Esta red logra una mejora en la exactitud de aproximadamente un 10 % y se reduce el tiempo de evaluación a aproximadamente la mitad en comparación con Fast R-CNN (Girshick, 2015).

Una estrategia diferente a los métodos analizados es la que toman los autores de YOLO (Redmon et al., 2016). Los trabajos anteriores plantean la detección como un problema de clasificación, en cambio YOLO lo plantea como un problema de regresión para predecir las regiones con objetos y las probabilidades para cada clase. Con una sola evaluación de la red se logra predecir

las regiones y las probabilidades y además el entrenamiento se puede optimizar de extremo a extremo sobre la exactitud en la detección. A diferencia de los métodos anteriores que generan regiones con posibles objetos, luego clasifican las mismas y posteriormente refinan estos resultados, YOLO plantea la detección como un simple problema de regresión, directamente desde pixeles a regiones y probabilidades para las diferentes clases. El sistema divide la imagen de entrada en una grilla de S x S, si el centro de un objeto cae dentro de la grilla de una celda, esa celda es la responsable de detectar a dicho objeto. Cada grilla luego genera B regiones y sus correspondientes probabilidades de clases para dichas regiones. La principal ventaja de ese método es que es extremadamente rápido, logrando la evaluación de 45 imágenes por segundo. Además YOLO genera la mitad de falsos positivos con respecto a la red Fast R-CNN (Girshick, 2015) debido a que trabaja con la imagen completa lo cual le permite ver el contexto, a diferencia de las otras redes. En comparación con Faster R-CNN, YOLO tiene una exactitud menor, aproximadamente del 10 % pero la velocidad de procesamiento es 3 veces más rápida.

Por el mismo camino de YOLO, en el trabajo de Liu et al. (2016) los autores proponen un nuevo detector llamado Detector de Disparo Único (SSD, por sus siglas en inglés). Este detector está formado por una red convolucional entrenada para clasificación de imágenes a la cual le eliminan las capas de salida y agregan nuevas capas convolucionales. Estas nuevas capas van disminuyendo en tamaño progresivamente y sobre ellas se aplican filtros que permiten la detección en múltiples escalas. Además de la detección multiescala estos filtros son generados para cuatro relaciones de aspectos. Para cada posición en donde se quiere detectar un objeto se genera un vector de $(c + 4)k$ elementos, en donde c es el número de clases y k el número de relaciones de aspecto; el 4 adicional de la fórmula corresponde a un refinamiento de la posición y tamaño del objeto. Las principales ventajas de este método es que es más rápido porque no realiza predicción de regiones con posibles objetos y que al estar entrenado de extremo a extremo la exactitud es mayor. Según los autores obtienen una exactitud comparable a Faster R-CNN pero con una velocidad 10 veces mayor y una exactitud aproximadamente 10 % mayor que YOLO pero con un tiempo de cómputo levemente menor.

3. COMPARACIÓN PROPUESTA

En este trabajo se plantea la comparación del comportamiento de tres algoritmos de detección de objetos actuales bajo diferentes condiciones de trabajo. Las redes a comparar son SSD (Liu et al., 2016), Faster R-CNN (Ren et al., 2015) y YOLO (Redmon et al., 2016). El problema que deberán resolver dichas redes es la detección de manos y manos con guantes bajo diferentes condiciones de trabajo. Las condiciones de trabajo que se evaluarán son las siguientes: 1) cambios en la iluminación debidos a diferentes horas del día, 2) cambios en el tamaño de los objetos y 3) cambio del entorno durante el entrenamiento y la evaluación.

Con respecto a los cambios en la iluminación debidos a las diferentes horas del día se consideran tres escenarios: mañana (M), tarde (T) y noche (N). Para la comparación del comportamiento ante diferentes condiciones de iluminación se planea realizar el entranamiento de la red sobre dos condiciones de iluminación y evaluar sobre la restante para las tres combinaciones posibles. Luego se entrenará la red en las tres condiciones de iluminación y se evaluará por separado en cada una de las condiciones de iluminación.

En relación con el tamaño de los objetos se consideran dos diferentes, objetos pequeños (P) y objetos grandes (G). Para la comparación se plantea entrenar sobre uno de los tamaños y evaluar sobre ambos tamaños; y por último entrenar sobre ambos tamaños de objetos y evaluar sobre ambos tamaños en forma independiente.

Por último se consideran dos entornos, el primero es un salón de un laboratorio y el se-

gundo es un ambiente industrial con mayor contaminación visual. Para esta comparación, se plantea el entrenamiento sobre el primero de los entornos y la evaluación en forma separada en ambos entornos; luego se agrega al entrenamiento el entorno industrial y se realiza la misma evaluación.

4. EXPERIMENTOS

En la Tabla 1 se muestran los diferentes experimentos realizados para la evaluación ante cambios en las condiciones de iluminación donde: M ∪ T indica la unión de los conjuntos **Manaña-Tarde**, M ∪ N indica la unión de los conjuntos **Manaña-Noche**, T ∪ N indica la unión de los conjuntos **Tarde-Noche** y M ∪ T ∪ N indica la unión de los conjuntos **Mañana-Tarde-Noche**. En la Tabla 2 se muestran los diferentes experimentos planteados para evaluar cambios

Entrenamiento	M ∪ T	M ∪ N	T ∪ N	M ∪ T ∪ N	M ∪ T ∪ N	M ∪ T ∪ N
Evaluación	N	T	M	M	T	N

Tabla 1: Experimentos para evaluar los cambios ante diferentes condiciones de iluminación.

en el tamaño de los objetos en donde P ∪ G indica la unión de los conjuntos con imágenes pequeñas y grandes. En la Tabla 3 se muestran los experimentos para evaluar un cambio en el

Entrenamiento	P	G	P ∪ G	P ∪ G	P ∪ G
Evaluación	P ∪ G	P ∪ G	P	G	P ∪ G

Tabla 2: Experimentos para evaluar los cambios ante diferentes tamaños de objetos.

entorno en donde **EI** indica un entorno industrial, **EI(-)** indica un entorno industrial, pero solo con muestras negativas (esto sería imágenes sin el objeto de interés) y **EI(+-)** indica un entorno industrial con muestras positivas y negativas.

Entrenamiento	M ∪ T ∪ N	M ∪ T ∪ N	M ∪ T ∪ N ∪ EI(-)	M ∪ T ∪ N ∪ EI(-+)
Evaluación	M ∪ T ∪ N	EI	EI	EI

Tabla 3: Experimentos para evaluar un cambio en el entorno.

4.1. Conjunto de Datos

Para la realización de los experimentos fué necesaria la construcción de un conjunto de datos formado por imágenes que recreen las condiciones de cada experimento. Las imágenes fueron capturadas en un laboratorio del Centro de Investigación en Informática para la Ingeniería de la FRC de la UTN. Para los cambios de iluminación se tomaron imágenes a la mañana en donde había iluminación solar en forma directa, a la tarde en donde había iluminación solar indirecta y a la noche con iluminación artificial. En las imágenes aparece una persona realizando actividades manuales sin guantes o utilizando diferentes guantes. Las personas se encuentran principalmente en 2 posiciones, cercanas a la cámara y alejadas de la misma para recrear diferentes tamaños de objetos; los objetos pequeños tienen un tamaño de aproximadamente 40 x 40 pixeles y los grandes 300 x 300 pixeles. En la Tabla 4 se muestra la cantidad de imágenes en total y cuantas corresponden a mañana, tarde y noche y cuantas contienen objetos pequeños y

	M	T	N	Subtotal
G	1504	1359	1340	4203
P	1498	1285	1523	4306
Subtotal	3002	2644	2863	**8509**

Tabla 4: Cantidad de imágenes del conjunto de datos. Discriminación entre M, N, T, G y P.

grandes. Para la captura se ubicó sobre un tripode una cámara Imaging Source modelo DFK 23UP031 con una resolución de 1920 x 1080 pixeles con su óptica correspondiente la cual se conectó a una notebook. En las figuras 1 y 2 se muestran ejemplos de imágenes bajo diferentes condiciones de iluminación y tamaños de objetos.

(a) Mañana (b) Tarde (c) Noche

Figura 1: Ejemplos de imágenes con diferentes condiciones de iluminación.

(a) Grande (b) Pequeño

Figura 2: Ejemplos de imágenes con diferentes tamaños de objetos.

Para la evaluación de un cambio en el entorno se capturaron imágenes de un operario realizando sus tareas habituales en una industria de la ciudad de Córdoba. La captura se realizó con una cámara compacta Sony modelo W830 con una resolución de 1920 x 1080 pixeles. Para estos experimentos se capturaron 2444 imágenes. En la Fig. 3 se muestra un ejemplo del entorno industrial utilizado.

Después de la captura es necesario identificar en cada imagen el objeto de interés. Este proceso se conoce como etiquetado y debe ser realizado en forma manual, dibujando un rectángulo sobre el objeto de interés como se muestra en la Fig. 4 y luego guardando en un archivo separado las coordenadas del rectángulo. En caso de que haya más de un objeto de interés en la imagen, todos son identificados. Para esto se utilizó el software Label Me [1]. En las 10953 imágenes se etiquetaron aproximadamente 20000 manos.

[1] http://www.labelme.org/

Figura 3: Ejemplos de imagen en el entorno industrial.

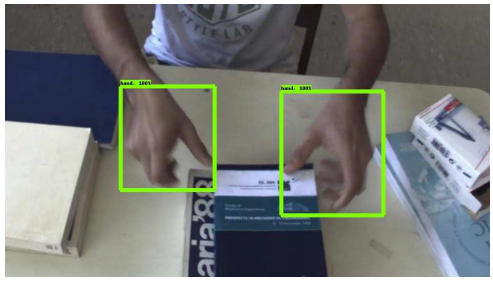

Figura 4: Ejemplo de imagen con sus etiquetas.

Red	Versión
SSD	SSD MobileNet v1
Faster R-CNN	Faster R-CNN Resnet101
YOLO	YOLO v3

Tabla 5: Versiones de modelos preentrenados utilizados en los experimentos.

4.2. Configuraciones

Para los experimentos se utilizaron los modelos oficiales de SSD y Faster R-CNN de la librería TensorFlow v1.12.0 [2] y el modelo oficial de YOLO v3 de la librería Darknet [3]. En la Tabla 5 se muestra la versión específica de cada uno de los modelos preentrenados utilizados. Para dichos modelos se usaron las configuraciones por defecto. El conjunto de datos se dividió en el 70 % de las imágenes para entrenamiento y el 30 % restante para evaluación. Para evaluar la exactitud se utilizó como medida la cantidad de objetos correctamente detectados sobre la cantidad total de objetos considerando una detección como válida cuando la intersección sobre la unión supera al 50 %. Para SSD y Faster R-CNN se realizaron 200k pasos de entrenamiento y para YOLO 25k porque se notó que para ambos después de esa cantidad de pasos la exactitud no mejoraba.

5. ANÁLISIS DE LOS RESULTADOS

En la Tabla 6 se muestran los resultados obtenidos ante cambios en las condiciones de iluminación. Para las 3 redes la evaluación sobre el conjunto **Noche** es la de menor exactitud. Esto se debe principalmente a las condiciones de iluminación, en donde se mezcla la iluminación del laboratorio con la proveniente de pasillos, escaleras y oficinas cercanas y también la proveniente del alumbrado exterior; estas fuentes de iluminación hacen que haya partes de la imagen muy iluminadas y otras no tanto, generando sombras indeseadas. La mayor exactitud se obtiene cuando se evalúa sobre el conjunto **Tarde**. La principal razón de esto son las mejores condiciones de iluminación porque la luz solar llega en forma indirecta y además aún la luz artificial no incide. Sobre el conjunto **Mañana** la exactitud se encuentra en un punto intermedio, esto se debe a que, aunque la luz es más directa que en el conjunto **Mañana**, aún no tenemos problemas con la luz artificial. En todos los casos la red que mejores resultados muestra es Faster R-CNN.

En la Tabla 7 se muestran los resultados obtenidos en la evaluación de diferentes tamaños de objetos. En este experimento se puede notar una gran diferencia entre entrenar con un tamaño

[2] https://www.tensorflow.org/
[3] https://pjreddie.com/darknet/

Entrenamiento	Evaluación	SSD	Faster R-CNN	YOLO
M ∪ T	N	0.147	0.723	0.445
M ∪ N	T	0.683	0.923	0.882
T ∪ N	M	0.577	0.836	0.716
M ∪ T ∪ N	M	0.748	0.956	0.916
M ∪ T ∪ N	T	0.684	0.927	0.845
M ∪ T ∪ N	N	0.643	0.889	0.832

Tabla 6: Resultados de la evaluación ante cambios en la iluminación.

o entrenar con ambos tamaños. Por ejemplo entrenando en P se obtiene una exactitud de 0.684 sobre P ∪ G, pero entrenando en P ∪ G esa exactitud sube a 0.942 usando la red Faster R-CNN. Nuevamente Faster R-CNN es la de mejor exactitud, pero la diferencia con las otras redes es más pequeña que la del experimento anterior. Esto nos da una pauta de que la iluminación es más problemática que el tamaño de los objetos.

Entrenamiento	Evaluación	SSD	Faster R-CNN	YOLO
P	P ∪ G	0.525	0.684	0.517
G	P ∪ G	0.479	0.553	0.513
P ∪ G	P	0.756	0.919	0.842
P ∪ G	G	0.837	0.965	0.964
P ∪ G	P ∪ G	0.800	0.942	0.893

Tabla 7: Resultados de la evaluación ante cambios en el tamaño de los objetos.

En la Tabla 8 se muestran los resultados de la evaluación ante cambios de entorno. Tanto SSD como YOLO tienen una notable disminución en la exactitud cuando la evaluación se realiza en otro entorno haciéndolas casi inusables, en cambio Faster R-CNN aunque tiene una caída de aproximadamente un 20 % su exactitud aún sigue siendo considerable. Esto último nos habla de la capacidad de generalización de Faster R-CNN con respecto a las otras. Cuando agregamos muestras del **EI** al entrenamiento, tanto en SSD como en YOLO sube la exactitud con respecto al experimento sin muestras del **EI** pero la exactitud no se acerca a la obtenida en el experimento en el laboratorio; esto se puede deber a las siguientes razones: 1) las muestras de entrenamiento en el **EI** no son suficientes y 2) las muestras de entrenamiento del laboratorio perjudican al resultado cuando no evaluamos en el laboratorio. Con respecto a la evaluación de Faster R-CNN en otro entorno pero agregando muestras de entrenamiento del otro entorno se logró superar la exactitud obtenida con Faster R-CNN en el ambiente de laboratorio. Esto nos muestra lo importante que es agregar mustras de entrenamiento con el fondo que se va a utilizar en el momento de evaluación.

Entrenamiento	Evaluación	SSD	Faster R-CNN	YOLO
M ∪ T ∪ N	M ∪ T ∪ N	0.727	0.940	0.882
M ∪ T ∪ N	EI	0.008	0.778	0.126
M ∪ T ∪ N ∪ EI(-+)	EI	0.594	0.952	0.785

Tabla 8: Resultados de la evaluación ante un cambio en el entorno de evaluación.

Por último en la Tabla 9 se muestran el tiempo que tarda cada una de las redes en evaluar una

imagen completa de 1280 x 720 pixeles y los cuadros por segundo (fps, por sus siglas en inglés) correspondientes a esos tiempos. Estos experimentos fueron realizados en una PC de escritorio con un procesador Intel core i7-7700 de 3.6GHz con 16GB de RAM y una GPU Geforce GTX 1060. Como se puede apreciar la red YOLO es la más rápida duplicando la velocidad de SSD y siendo unas 15 veces más rápida que Faster R-CNN. Tanto SSD como YOLO pueden ser ejecutadas en tiempo real.

Evaluación	SSD	Faster R-CNN	YOLO
Tiempo (s)	0.0625	0.5000	0.0322
fps	16	2	31

Tabla 9: Tiempo de evaluación por imagen y cuadros por segundo correspondientes.

6. CONCLUSIONES Y TRABAJO A FUTURO

Se propuso el estudio y evaluación de técnicas de detección de objetos en movimiento y su aplicación en ambientes industriales visualmente complejos usando CNN. Para ello se exploró el comportamiento de la detección de objetos usando estas redes bajo cambios en los tamaños de objetos a detectar, en la iluminación y en los ambientes. También se evaluaron los tiempos de cómputo de las distintas redes. En términos generales podemos decir que la red Faster R-CNN es la que mejor exactitud presenta aunque es la más costosa computacionalmente y difícil de aplicar en problemas en tiempo real, en cambio YOLO es la más rápida y puede ser aplicada en tiempo real aunque la exactitud es menor a la de Faster R-CNN. Con respecto a los cambios de iluminación se encontró que es muy importante que los algoritmos sean entrenados con la iluminación en la cual van a ser utilizados y también que el tamaño de los objetos con que entrenamos a las redes influye en los resultados aunque no tanto como los cambios en la iluminación. Para el caso de cambios de entorno se encontró que es muy importante entrenar los algoritmos en los lugares en donde serán utilizados aún cuando se trabaje con los mismos objetos, esto nos habla de lo importante que es el fondo para una correcta detección; aunque la red Faster R-CNN mostró una capacidad de generalización muy superior a las demás.

Una de las cosas pendientes a evaluar es porque YOLO y SSD no logran la misma exactitud en el laboratorio que en el ambiente industrial. Como se planteó esto se puede deber a que no hay suficientes muestras de entrenamiento o que usar muestras de otro entorno como el laboratorio en este caso es perjudicial. Otra cosa a investigar es la posibilidad de mejorar los tiempos de ejecución de Faster R-CNN reduciento el número de propuestas de regiones por imágenes. Estos puntos será evaluados en trabajos futuros. Por último, se está trabajando en un proyecto de transferencia para que estas redes sean aplicadas como parte de un sistema de seguridad industrial en una empresa de la ciudad de Córdoba.

REFERENCIAS

Benhimane S., Najafi H., Grundmann M., Genc Y., Navab N., y Malis E. Real-time object detection and tracking for industrial applications. En *VISAPP (2)*, páginas 337–345. Citeseer, 2008.

Dalal N. y Triggs B. Histograms of oriented gradients for human detection. En *international Conference on computer vision & Pattern Recognition (CVPR'05)*, volumen 1, páginas 886–893. IEEE Computer Society, 2005.

Deng J., Dong W., Socher R., Li L.J., Li K., y Fei-Fei L. Imagenet: A large-scale hierarchical image database. En *2009 IEEE conference on computer vision and pattern recognition*, páginas 248–255. Ieee, 2009.

Dunsmore A. Survey of object-oriented defect detection approaches and experience in industry. Informe Técnico, Citeseer, 2000.

Farooq J. Object detection and identification using surf and bow model. En *2016 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, páginas 318–323. IEEE, 2016.

Felzenszwalb P.F., Girshick R.B., y McAllester D. Cascade object detection with deformable part models. En *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, páginas 2241–2248. IEEE, 2010.

Girshick R. Fast r-cnn. En *Proceedings of the IEEE international conference on computer vision*, páginas 1440–1448. 2015.

Girshick R., Donahue J., Darrell T., y Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. En *Proceedings of the IEEE conference on computer vision and pattern recognition*, páginas 580–587. 2014.

Gokberk Cinbis R., Verbeek J., y Schmid C. Segmentation driven object detection with fisher vectors. En *Proceedings of the IEEE International Conference on Computer Vision*, páginas 2968–2975. 2013.

Hirano Y., Garcia C., Sukthankar R., y Hoogs A. Industry and object recognition: Applications, applied research and challenges. En *Toward Category-Level Object Recognition*, páginas 49–64. Springer, 2006.

Lin T.Y., Maire M., Belongie S., Hays J., Perona P., Ramanan D., Dollár P., y Zitnick C.L. Microsoft coco: Common objects in context. En *European conference on computer vision*, páginas 740–755. Springer, 2014.

Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C.Y., y Berg A.C. Ssd: Single shot multibox detector. En *European conference on computer vision*, páginas 21–37. Springer, 2016.

Luan S., Li Y., Wang X., y Zhang B. Object detection and tracking benchmark in industry based on improved correlation filter. *Multimedia Tools and Applications*, 77(22):29919–29932, 2018.

Redmon J., Divvala S., Girshick R., y Farhadi A. You only look once: Unified, real-time object detection. En *Proceedings of the IEEE conference on computer vision and pattern recognition*, páginas 779–788. 2016.

Ren S., He K., Girshick R., y Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. En *Advances in neural information processing systems*, páginas 91–99. 2015.

Sharif Razavian A., Azizpour H., Sullivan J., y Carlsson S. Cnn features off-the-shelf: an astounding baseline for recognition. En *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, páginas 806–813. 2014.

Uijlings J.R., Van De Sande K.E., Gevers T., y Smeulders A.W. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

Viola P., Jones M., et al. Rapid object detection using a boosted cascade of simple features. *CVPR (1)*, 1:511–518, 2001.