Análisis de técnicas de aumento de datos y entrenamiento en YOLOv3 para detección de objetos en imágenes RGB y TIR del UMA-SAR Dataset

Álvaro Galán-Cuenca, Ricardo Vázquez-Martín, Anthony Mandow, Jesús Morales
y Alfonso García-Cerezo
Universidad de Málaga, Andalucía Tech, Departamento de Ingeniería de Sistemas y Automática
amandow@uma.es

Resumen

El uso de imágenes de los espectros visible (RGB) e infrarrojo térmico (TIR) para la detección de objetos puede resultar crucial en aplicaciones donde las condiciones de visibilidad están limitadas, como la robótica para búsqueda y rescate en catástrofes. Para ello resulta beneficioso analizar cómo las técnicas de aprendizaje profundo basadas en redes neuronales convolucionales (CNN) pueden aplicarse a ambas modalidades. En este artículo se analizan diferentes configuraciones y parámetros para el entrenamiento de CNN tanto para imágenes térmicas como para imágenes equivalentes del espectro visible. En concreto, se aborda el problema del sobre-entrenamiento para determinar una configuración eficaz de técnicas de aumento de datos y parada temprana. El caso de estudio se ha realizado con la red de código abierto YOLOv3, pre-entrenada con el dataset RGB COCO y optimizada (o re-entrenada) con el conjunto público de datos UMA-SAR dataset, que incluye pares de imágenes RGB y TIR obtenidas en ejercicios realistas de rescate.

Palabras clave: visión por computador, aprendizaje profundo, redes neuronales convolucionales, YOLO, imágenes térmicas.

1. INTRODUCCIÓN

La visión por computador puede contribuir decisivamente a aplicaciones complejas de la robótica como la respuesta a desastres, incluyendo el reconocimiento y cartografía, búsqueda de supervivientes, logística, primera asistencia médica, evacuación de víctimas y apoyo para la percepción cooperativa. Sin embargo, los sistemas de visión de un vehículo terrestre (UGV) o aéreo (UAV) pueden producir gigabytes de imágenes [15]. A esto se añaden las limitaciones en la calidad de las imágenes recibidas, la diversidad de los objetos de interés y lo desestructurado de los entornos [3]. De ahí que se necesiten nuevas herramientas de inteligencia artificial para identificar información útil [9][8].

En especial, las imágenes térmicas pueden ofrecer información relevante en entornos de baja visibilidad o para supervisión de incendios [1], pero en general aportan menor resolución, mayor ruido y contornos difusos [11]. En este sentido, la redes neuronales convolucionales de aprendizaje profundo (CNN) pueden ofrecer resultados eficaces tanto para el espectro visible como para el térmico infrarrojo (TIR) [5].

Un problema para la adopción de CNNs en nuevas aplicaciones es la escasez de imágenes convenientemente etiquetadas para casos de aplicación o modalidades específicas [2]. En este sentido, las técnicas de transferencia de conocimiento han demostrado eficacia en distintas aplicaciones [7] y se pueden combinar con operaciones de aumento de datos para conseguir buenos resultados a partir de conjuntos de datos reducidos [8].

La arquitectura de código abierto YOLO, preentrenada con imágenes RGB del dataset COCO [12], ofrece un buen compromiso entre velocidad y precisión, y ha ofrecido buenos resultados al reentrenarla con datos específicos (transferencia de conocimiento) en aplicaciones en el espectro visible, como la detección de congestiones del tráfico [10], la detección de objetos en entornos rurales [4], el desarrollo aplicaciones de inteligencia artificial (AI) en internet de las cosas (IoT) en un chip [19] o la detección desde cámaras vestibles [18].

En un trabajo previo [5], evaluamos la capacidad de la arquitectura CNN de YOLO para entrenar dos redes correspondientes a imágenes térmicas infrarrojas y en el espectro visible con objeto de detectar cuatro clases representativas en escenas de búsqueda y rescate (SAR) que comprendían personas (civiles e intervinientes) y vehículos (automóvil civil y vehículo de respuesta).

En este trabajo se propone un análisis de la red YOLOv3 [17] para la identificación de objetos. Se comienza analizando el problema del sobreentrenamiento mediante técnicas de aumento de datos y parada temprana. Después se evalúa el efecto de los dos hiper-parámetros del algoritmo de aprendizaje profundo más relevantes, la tasa de aprendizaje y el tamaño de lote. Además, se analiza la respuesta de esta red RGB para detectar ob-

https://doi.org/10.17979/spudc.9788497498043.686

jetos en las imágenes TIR. Con este fin, se utiliza un conjunto de datos recogidos durante ejercicios realistas de respuesta a emergencias que contiene pares de imágenes RGB y TIR.

El resto del artículo se organiza de la siguiente manera. La sección 2 revisa brevemente el conjunto de datos públicos utilizado en este trabajo. En la sección 3 se definen los casos de análisis para evitar el sobre-entrenamiento de la red. La sección 4 presenta los resultados para diferentes procesos de entrenamiento de la red en base a diferentes valores de sus hiper-parámetros. Finalmente, se presentan las conclusiones e ideas para desarrollos futuros.

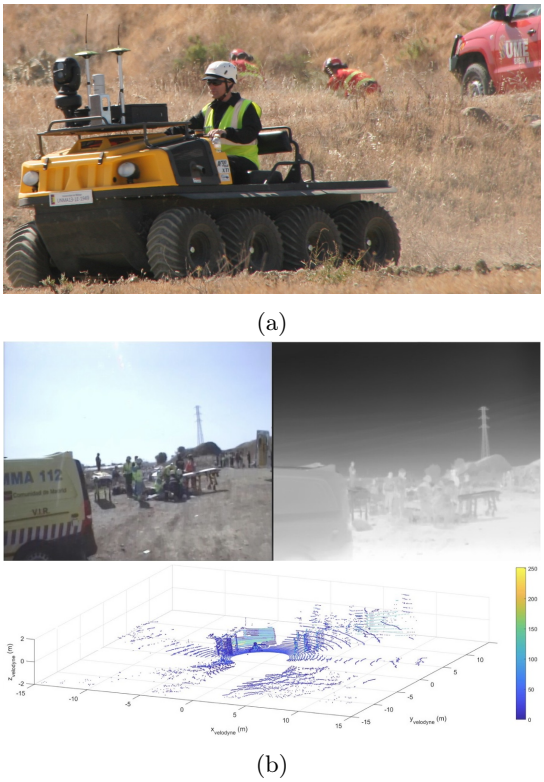
2. METODOLOGÍA

2.1. El dataset UMA-SAR

Para el entrenamiento hemos utilizado nuestro conjunto de datos *UMA-SAR Dataset* [14], disponible públicamente en www.uma.es/robotics-and-mechatronics/sar-datasets.

Este *dataset* recopila información sensorial multimodal capturada desde un vehículo tripulado todoterreno durante ejercicios realistas de búsqueda y rescate celebrados en 2018 y 2019 en el Área de Experimentación en Nuevas Tecnologías para la Intervención en Emergencias de la Universidad de Málaga [20]. En la Fig. 1(a) se muestra el conjunto sensorial, formado por dos cámaras monoculares sincronizadas de luz visible (RGB) e infrarrojo térmico (TIR), un lidar tridimensional (3D) Velodyne HDL-32, así como una unidad de medición inercial (IMU) y dos receptores del sistema de posicionamiento global (GPS) con los que obtener el *ground truth*.

Nuestra misión en los ejercicios fue recoger una amplia gama de datos del dominio SAR (*Search and Rescue*), incluyendo personas, vehículos, escombros y actividad SAR en terreno no estructurado. En concreto, se recogieron cuatro secuencias de datos siguiendo rutas cerradas durante los ejercicios, con una longitud total de la trayectoria de 5,2 km y un tiempo total de 77 minutos. Además, proporcionamos tres secuencias más del lugar vacío (es decir, antes o después del ejercicio) con fines de comparación (4,9 km adicionales y 46 min). Adicionalmente, los datos se ofrecen tanto en formato legible para el ser humano como en formato de archivos *rosbag*, y se proporcionan dos herramientas de software específicas para extraer y adaptar este conjunto de datos a la preferencia de los usuarios. Un ejemplo de imágenes y nubes de puntos 3D capturadas durante el ejercicio de 2019 se presenta en la Fig. 1(b).



(a)

(b)

Figura 1: (a) Vehículo todoterreno y sistema sensorial, (b) Imágenes de cámaras RGB/TIR y nube de puntos 3D del LIDAR.

2.2. Modelo de datos

El dataset empleado para el entrenamiento de la red ha sido construido con 1125 imágenes etiquetadas con cinco clases: $\mathcal{C}$= {*civil, interviniente, víctima-yaciente, automóvil-civil, vehículo-respuesta*}. A las cuatro clases utilizadas en [5], que comprenden personas (civiles e intervinientes) y vehículos (automóvil civil y vehículo de respuesta), se ha añadido la clase *víctima-yaciente*. De estas imágenes, 775 proceden del UMA-SAR dataset. Con objeto de disponer de imágenes adicionales con suficiente número de víctimas yacientes, se han añadido 350 imágenes del conjunto publicado por por Oliveira *et al.* [16], compuesto únicamente por imágenes RGB. Puesto que YOLO requiere que todas las imágenes tengan las mismas dimensiones, se han escalado todas al tamaño 416×416.

Así, en este trabajo se considera una única red YOLOv3 re-entrenada con imágenes RGB. Del el número total de imágenes en el dataset etiquetado, se ha utilizado el 80 % para el entrenamiento, el 15 % para validación y el 5 % para la fase de testeo.

https://doi.org/10.17979/spudc.9788497498043.686

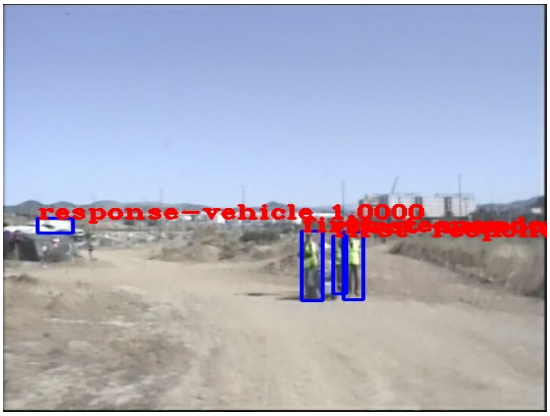


Figura 2: Imagen original del UMA-SAR dataset.

3. ANÁLISIS DE TÉCNICAS PARA EVITAR SOBRE-ENTRENAMIENTO

3.1. Aumento de datos

El aumento de datos amplia el conjunto de datos que se usa para el entrenamiento por medio de transformaciones a las imágenes existentes y a sus respectivos cuadros delimitadores si fuera necesario. Posteriormente se añaden dichas imágenes transformadas al dataset junto a las originales. De esta forma se consigue aumentar la cantidad de datos etiquetados para el entrenamiento de la red. Además, gracias a las transformaciones sobre las imágenes, se crean nuevos ejemplos ampliando la variedad de patrones, que ayuda a la generalización de la red durante el proceso de aprendizaje.

3.1.1. Transformaciones

Se analizan cuatro transformaciones a las imágenes del dataset: i) aplicar un volteo horizontal a la imagen, ii) modificar su saturación, iii) modificar el brillo y iv) cambiar el contraste. En la Figura 2 se muestra una imagen de ejemplo del UMA-SAR dataset con las etiquetas de las clases presentes en la imagen. Como se puede apreciar en la imagen, aparece un coche de primeros auxilios en la parte izquierda y tres intervinientes en la parte derecha. La Figura 3 muestra las cuatro transformaciones bajo análisis mencionadas anteriormente, aplicadas a la imagen original mostrada en la Figura 2.

3.1.2. Análisis experimental de los resultados

En esta sección se analiza cómo afecta el aumento de datos al dataset del trabajo a fin de determinar qué transformación es más eficiente. Para los entrenamientos realizados en este análisis, se han

Tabla 1: Parámetros de configuración iniciales de la red.

Parámetro	Valor
Tamaño de lote	8
Tasa de aprendizaje	$1 \cdot 10^{-2}$
Épocas	80

Tabla 2: Resultados de aplicar diferentes casos de aumento de datos.

Transformación	Núm. de imágenes	mAP
ST	900	34.87 %
VH	1800	25.53 %
SAT	1800	26.09 %
BR	1800	16.02 %
CON	1800	35.30 %

fijado los cuatro parámetros mostrados en la tabla 1 para los cinco casos de estudio planteados.

Los casos de estudio son:

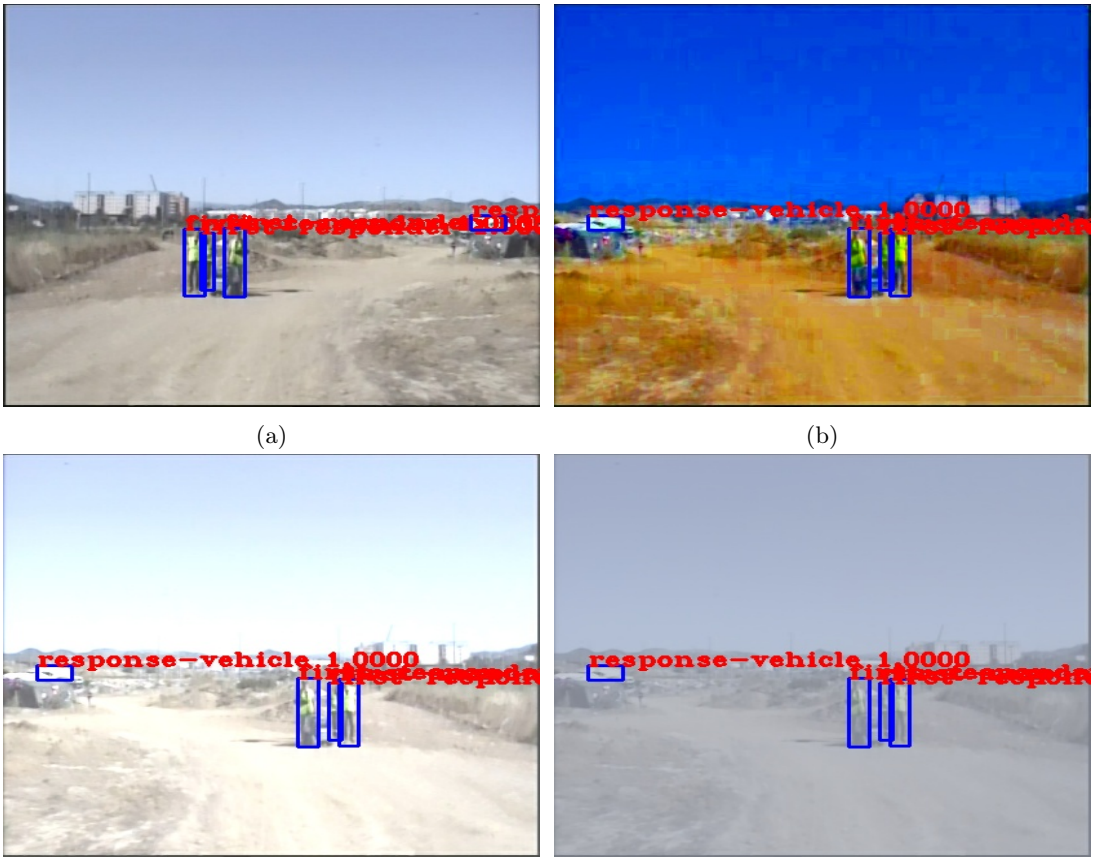
- Caso 1: Sin aplicar ninguna transformación (ST).
- Caso 2: Aplicar únicamente un volteo horizontal de imagen (VH).
- Caso 3: Cambiar la saturación de las imágenes (SAT).
- Caso 4: Modificar el brillo de las imágenes (BR).
- Caso 5: Cambiar el contraste de las imágenes (CON).

La Tabla 2 recoge el *mean Average Precision* (mAP) conseguido para cada uno de los casos de estudio, así como el número total de imágenes empleadas en el entrenamiento que se obtiene tras el aumento de datos. Estos resultados indican un porcentaje de acierto mayor cuando no se aplica ninguna transformación. Esto podría explicarse debido a que la red ejecuta 80 épocas (ver Tabla 1), de manera que al aumentar el número de imágenes la red converge antes pero aun así debe realizar ese número de épocas, lo que implica que la red sufra de sobre-entrenamiento. Por este motivo, se necesita establecer un protocolo de parada temprana que permita detener el entrenamiento antes de que la red comience a memorizar en lugar de aprender.

3.2. Parada temprana

Esta técnica consiste en detener el entrenamiento cuando los errores de entrenamiento y validación empiecen a separarse. Para detener el entrenamiento se debe indicar el número de épocas p

https://doi.org/10.17979/spudc.9788497498043.686



(a) (b)

(c) (d)

Figura 3: (a) Volteo horizontal, (b) cambio en la saturación, (c) modificación del brillo y (d) cambio de contraste.

Tabla 3: Resultados de los cinco casos de aumento de datos para $p = 3$.

Transformación	Número de imágenes	Número de épocas	mAP
ST	900	12	24.30 %
VH	1800	7	38.33 %
SAT	1800	8	36.73 %
BR	1800	9	42.82 %
CON	1800	9	36.25 %

que pueden transcurrir desde que ambos errores empiezan a divergir. En este trabajo se han analizado los siguientes valores para el parámetro p: 3, 5 , 10 y 20.

3.3. Análisis conjunto de aumento de datos y parada temprana

Las Tablas 3, 4, 5 y 6 recogen los resultados de combinar la técnica de parada temprana con los cinco casos de estudio de aumento de datos. Estos resultados indican mejores resultados con las transformaciones de volteo horizontal (VH) y cambio de brillo (BR).

Tabla 4: Resultados de los cinco casos de aumento de datos para $p = 5$.

Transformación	Número de imágenes	Número de épocas	mAP
ST	900	15	42.46 %
VH	1800	17	60.72 %
SAT	1800	13	54.72 %
BR	1800	10	59.12 %
CON	1800	13	58.62 %

Tabla 5: Resultados de los cinco casos de aumento de datos para $p = 10$.

Transformación	Número de imágenes	Número de épocas	mAP
ST	900	16	51.01 %
VH	1800	16	63.27 %
SAT	1800	13	50.25 %
BR	1800	14	51.98 %
CON	1800	18	55.09 %

3.4. Combinación de técnicas de aumento de datos

A continuación, se estudia la combinación de técnicas de aumento de datos para producir un mayor incremento del número de imágenes de entrenamiento. En particular, se evalúan tres combina-

https://doi.org/10.17979/spudc.9788497498043.686

Tabla 6: Resultados de los cinco casos de aumento de datos para $p = 20$.

Transformación	Número de imágenes	Número de épocas	mAP
ST	900	35	48.08 %
VH	1800	31	49.76 %
SAT	1800	23	23.39 %
BR	1800	25	44.88 %
CON	1800	26	57.07 %

ciones:

- Caso 6: Combinación del caso 2 y del caso 4, triplicando el número de datos de entrenamiento (VH+BR).
- Caso 7: Aplicar las cuatro transformaciones por separado, de esta manera el número de imágenes de entrada se multiplica por cinco (VH+SAT+BR+CON).
- Caso 8: Combinación aleatoria de todas las transformaciones hasta conseguir aumentar el dataset a un número de datos considerable. En este caso se hacen primero las cuatro transformaciones de manera separada como en el caso 7 y luego se realizan combinaciones de ellas, como cambiar el brillo y hacer un volteo horizontal a la vez, consiguiendo así por cada imagen de entrada generar siete imágenes nuevas (COMB).

Los resultados de los ochos casos de estudio del aumento de datos para los cuatro posibles valores de p están representados en la gráfica de la figura 4. En la gráfica el eje de abscisas representa el número de épocas tras el punto de parada temprana, p, y el de ordenadas el porcentaje de acierto, el mAP. En dicha gráfica se puede ver que para el caso 7 (VH+SAT+BR+CON) con $p = 5$ se consigue el mejor resultado, que es de un 69.55 % tras 16 épocas.

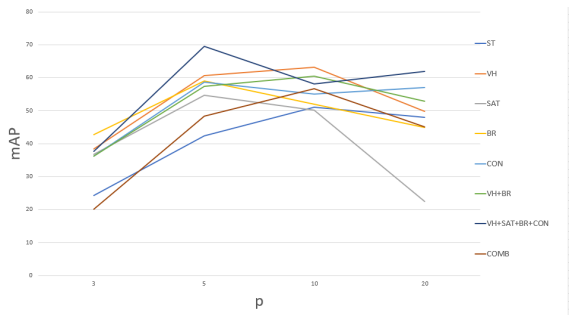


Figura 4: Gráfica representativa de los ocho casos de estudio del aumento de datos para los cuatro posibles valores de p.

Tabla 7: Resultados de modificar el valor de la tasa de aprendizaje.

Tasa de aprendizaje	mAP	Número de épocas	Tiempo
$1 \cdot 10^{-1}$	8.62 %	7	11m4s
$1 \cdot 10^{-2}$	69.55 %	16	10m50s
$1 \cdot 10^{-3}$	82.75 %	6	9m59s
$1 \cdot 10^{-4}$	79.06 %	6	8m55s
$1 \cdot 10^{-5}$	57.62 %	8	12m32s
$1 \cdot 10^{-6}$	57.84 %	26	49m31s

Tabla 8: Resultados de modificar el valor del tamaño de lote.

Tasa de aprendizaje	mAP	Número de épocas	Tiempo
8	82.75 %	6	10m50s
12	84.70 %	6	8m53s
16	81.37 %	6	9m39s
32	83.13 %	6	9m51s
64	75.46 %	9	15m11s
128	60.13 %	12	20m22s

4. ANÁLISIS DE PARÁMETROS DE ENTRENAMIENTO

El ajuste de los parámetros del descenso del gradiente presente en los algoritmos de aprendizaje profundo mejorar la precisión de la red. En esta sección, se analizarán experimentalmente la tasa de aprendizaje o *learning rate* y el tamaño de lote o *batch size*.

4.1. Tasa de aprendizaje

Este parámetro permite regular cómo de rápido o lento aprende la red y cómo se adapta ésta frente nuevos patrones de entrada [6]. Su valor es positivo y comprendido entre 0 y 1, aunque los valores más empleados para este parámetro son: 0.1, 0.01, 0.001, 0.0001, 0.00001 y 0.000001.

Se ha entrenado la red con la combinación de transformaciones para el aumento de datos del caso 7 (VH+SAT+BR+CON) y $p = 5$, pero esta vez alternando el valor de la tasa de aprendizaje para analizar su efecto. Los resultados se muestran en la tabla 7, que recoge el valor del *mean Average Precision*, el número de épocas y el tiempo que tarda la red en converger para cada uno de los valores de la tasa de aprendizaje. En vista a estos resultados, se puede concluir que mientras menor sea el valor de la tasa de aprendizaje, más tarda la red en converger; mientras mayor sea su valor, peor es el porcentaje de acierto, y que para nuestro caso de estudio, el mejor valor de la tasa de aprendizaje es de $1 \cdot 10^{-3}$.

4.2. Tamaño de lote

Este parámetro indica el número de muestras (imágenes) del dataset de entrenamiento que se emplea en cada iteración. El número de iteraciones que se realizan en cada época (I_E) es inver-

https://doi.org/10.17979/spudc.9788497498043.686

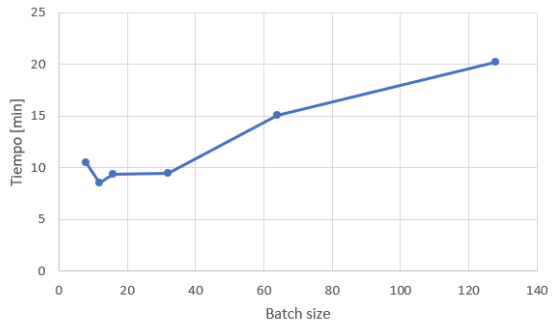


Figura 5: Identificación de objetos mediante CNN en pares de imágenes.

Tabla 9: Condiciones de la red neuronal de este trabajo.

Transferencia de conocimiento	Sí
Tamaño de lote (B)	12
Tasa de aprendizaje	$1 \cdot 10^{-3}$
Tamaño de las imágenes	416×416
p	5
Aumento de datos	VH+SAT+BR+CO

samente proporcional al tamaño del lote (B), es decir:

$$I_E = \frac{N_T}{B}, \tag{1}$$

donde N_T es el número total de imágenes usadas para el entrenamiento.

En este análisis se entrena la red con las mismas transformaciones que en el apartado anterior, un valor de la tasa de aprendizaje de $1 \cdot 10^{-3}$, y se modifica el valor del tamaño de lote. Los valores evaluados para este parámetro son: 8, 12, 16, 32, 64 y 128, que son considerados habituales [6] [13]. Los resultados se recogen en la Tabla 8, donde un tamaño de lote igual a 12 proporciona los mejores resultados. Esto significa que en cada una de las seis épocas se han realizado 375 iteraciones (se tienen $N_T = 4500$ imágenes después de aplicar el aumento de datos) con 12 imágenes en cada iteración. Se consigue así alcanzar un mAP del 84.70 % de media para las cinco clases. La gráfica de la Figura 5, muestra como a medida que aumenta el tamaño de lotes, el tiempo que tarda la red neuronal en converger también aumenta.

5. RESULTADOS

El análisis experimental permite establecer las condiciones de entrenamiento mostradas en la Tabla 9 para el conjunto de clases $\mathcal{C}$= {*civil, interviniente, víctima-yaciente, automóvil-civil, vehículo-respuesta*}. Los resultados obtenidos en Average Precision (AP) para cada clase para imágenes RGB se muestran en la Tabla 10. Como se puede observar, se alcanza un porcentaje muy elevado en la detección de víctimas, así como precisiones adecuadas en el resto de clases. La alta precisión en la detección de la clase víctima se ha alcanzado gracias al uso de un dataset específico de víctimas [16], que ha permitido añadir 350 imágenes donde está presente dicha clase.

Tabla 10: Resultados para imágenes del espectro visible.

Clase	AP
civiles	77.29 %
automóvil civil	76.84 %
intervinientes	85.27 %
vehículo de respuesta	84.56 %
victimas yacentes	99.62 %
mAP	84.70 %

Tabla 11: Resultado de la red RGB para imágenes de prueba TIR

Clase	AP
civiles	21.66 %
automóvil civil	13.51 %
intervinientes	11.45 %
vehículos de respuesta	26.48 %
mAP	18.37 %

En la literatura se pueden encontrar datasets con imágenes térmicas, pero son fundamentalmente de peatones. Por tanto, no existen datasets de víctimas en el espectro TIR, como las que se han utilizado para imágenes RGB. En la Figura 6 se muestran dos ejemplos de detección en imágenes TIR. Se puede apreciar como en condiciones de baja visibilidad con este tipo de imágenes se pueden detectar las clases presentes cuando no es posible que sean detectadas empleando imágenes RGB.

En un trabajo previo hemos comprobado la viabilidad de utilizar un re-entrenamiento de imágenes TIR sobre una red YOLO [5]. Resulta interesante estudiar el rendimiento de la red RGB entrenada en este trabajo para detectar objetos en imágenes TIR del dataset UMA-SAR [14]. Los resultados obtenidos se ofrecen en la Tabla 11, donde no se ha incluido la clase víctima ya que su presencia no es representativa en en las imágenes de prueba. Como se puede apreciar, los resultados son muy inferiores a los que se alcanzan con imágenes RGB, ya que la red no ha sido entrenada especialmente para el espectro termográfico. Este resultado refleja la necesidad de realizar un entrenamiento específico de la red para imágenes TIR, como el realizado en [5]. En cualquier caso, dada

https://doi.org/10.17979/spudc.9788497498043.686

(a) (b)

Figura 6: Detección en imágenes TIR: (a) imagen en exterior con buena visibilidad y (b) imagen con baja visibilidad.

la poca disponibilidad de imágenes TIR en datasets públicos y con imágenes etiquetadas, resulta relevante que se pueda utilizar la transferencia de conocimiento de redes pre-entrenadas con imágenes RGB para ser optimizadas con imágenes térmicas, como también se concluye en [5].

6. CONCLUSIONES

En este artículo se ha realizado un análisis de diferentes herramientas y parámetros para el entrenamiento de una red neuronal convolucional (CNN) utilizando imágenes RGB del dataset público multimodal *UMA-SAR Dataset*, obtenido durante ejercicios realistas de búsqueda y rescate en catástrofes. En concreto se ha abordado el problema del sobre-entrenamiento para determinar una configuración eficaz de técnicas de aumento de datos y parada temprana. La operación volteo horizontal a las imágenes originales ha producido el mejor resultado de entrenamiento. En cuanto a la técnica de parada temprana, el momento óptimo para detener el entrenamiento se ha obtenido tras 5 épocas transcurridas desde que el error de validación comienza a crecer y se aleja del error de entrenamiento. Respecto a los hiperparámetros, se ha concluido que utilizar una tasa de entrenamiento de $1 \cdot 10^{-3}$ y un tamaño de lote de 12 son los valores que proporcionan un buen rendimiento en el entrenamiento de la red.

Los resultados muestran una buena precisión en las cinco clases tras la optimización de la red preentrenada de YOLO para imágenes RGB, y se alcanzan precisiones equivalentes al trabajo previo [5], donde se definían cuatro clases.

Como trabajo futuro, resulta necesario de incorporar nuevos datos públicos donde esté presente la clase víctima. Por último, también es interesante realizar un análisis análogo al presentado en este trabajo sobre el aumento de datos en imágenes TIR, para poder determinar las transformaciones más relevantes para esta modalidad, así como aumentar el tipo de transformaciones empleadas.

Agradecimientos

Este trabajo ha recibido financiación del proyecto nacional RTI2018-093421-B-I00, la Universidad de Málaga (Andalucía Tech).

English summary

Analysis of data augmentation and training techniques for YOLOv3 object detection in RGB and TIR images from the UMA-SAR Dataset

Abstract

The combination of imaging of visible (RGB) and thermal infrared (TIR) modalities can be crucial for object detection in applications where visibility conditions are limited, such as search and rescue robotics. For this, it is beneficial to analyze how deep learning techniques based on convolutional neural networks (CNN) can be applied to these modalities. This article discusses different settings and parameters for CNN training for two equivalent sets of thermal and RGB images. Specifically, we address the problem of overfitting and determine an effective configuration of

https://doi.org/10.17979/spudc.9788497498043.686

data augmentation and early stop techniques. The case study has been carried out with the open source network YOLOv3, pre-trained with the RGB COCO dataset, and optimised with the UMA-SAR dataset, which includes pairs of RGB and TIR images obtained in realistic rescue exercises.

Keywords: Computer vision, deep learning, convolutional neural networks, YOLO, thermal imaging

Referencias

[1] Al-Kaff, A., Madridano, I., Campos, S., GarcÍa, F., MartÍn, D. and de la Escalera, A.: 2020, Emergency support unmanned aerial vehicle for forest fire surveillance, *Electronics* **9**(2).

[2] Alonso, I., Yuval, M., Eyal, G., Treibitz, T. and Murillo, A. C.: 2019, CoralSeg: learning coral segmentation from sparse annotations, *Journal of Field Robotics* **36**(8), 1456–1477.

[3] Arnold, S., Ohno, K., Hamada, R. and Yamazaki, K.: 2019, An image recognition system aimed at search activities using cyber search and rescue dogs, *Journal of Field Robotics* **36**(4), 677–695.

[4] Barba-Guaman, L., Naranjo, J. E., Ortiz, A. and Gonzalez, J. G. P.: 2021, Object detection in rural roads through SSD and YOLO framework, *Advances in Intelligent Systems and Computing* **1365 AIST**, 176–185.

[5] Bañuls, A., Mandow, A., Vázquez-Martín, R., Morales, J. and García-Cerezo, A.: 2020, Object detection from thermal infrared and visible light cameras in search and rescue scenes, *IEEE International Symposium on Safety, Security, and Rescue Robotics*, pp. 380–386.

[6] Bengio, Y.: 2012, Practical recommendations for gradient-based training of deep architectures, *Neural Networks: Tricks of the Trade* p. 437–478.

[7] Blanco-Medina, P., Fidalgo, E., Alegre, E., Vasco-Carofilis, R. A., Jañez-Martino, F. and Villar, V. F.: 2021, Detecting vulnerabilities in critical infrastructures by classifying exposed industrial control systems using deep learning, *Applied Sciences* **11**(1), 1–14.

[8] Cebollada, S., Payá, L., Flores, M., Peidró, A. and Reinoso, O.: 2021, A state-of-the-art review on mobile robotics tasks using artificial intelligence and visual data, *Expert Systems with Applications* **167**.

[9] Chaves, D., Saikia, S., Fernández-Robles, L., Alegre, E. and Trujillo, M.: 2018, A systematic review on object localisation methods in images [Una revisión sistemática de métodos para localizar automáticamente objetos en imágenes], *Revista Iberoamericana de Automática e Informática Industrial* **15**(3), 231–242.

[10] Gan, H. M., Fernando, S. and Molina-Solana, M.: 2021, Scalable object detection pipeline for traffic cameras: Application to Tfl JamCams, *Expert Systems with Applications* **182**.

[11] Hinojosa, S., Pajares, G., Cuevas, E. and Ortega-Sanchez, N.: 2018, Thermal image segmentation using evolutionary computation techniques, *Studies in Computational Intelligence* **730**, 63–88.

[12] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L.: 2014, Microsoft COCO: Common objects in context, *European Conference on Computer Vision*, Springer International Publishing, pp. 740–755.

[13] Masters, D. and Luschi, C.: 2018, Revisiting small batch training for deep neural networks, *arXiv* (1804.07612).
URL: *https://arxiv.org/abs/1804.07612v1*

[14] Morales, J., Vázquez-Martín, R., Mandow, A., Morilla-Cabello, D. and García-Cerezo, A.: 2021, The UMA-SAR dataset: Multimodal data collection from a ground vehicle during outdoor disaster response training exercises, *The International Journal of Robotics Research* **40**(6-7), 835–847.

[15] Murphy, R. R., Tadokoro, S. and Kleiner, A.: 2016, *Springer Handbook of Robotics*, Springer, Cham, chapter Search and Rescue Robotics, pp. 1151–1173.

[16] Oliveira, G., Valada, A., Bollen, C., Burgard, W. and Brox, T.: 2016, Deep learning for human part discovery in images, *IEEE International Conference on Robotics and Automation (ICRA)*.

[17] Redmon, J. and Farhadi, A.: 2018, YOLOv3: an incremental improvement, *arXiv* (1804.02767).
URL: *https://arxiv.org/abs/1804.02767v1*

[18] Sabater, A., Montesano, L. and Murillo, A. C.: 2019, Performance of object recognition in wearable videos, *IEEE International Conference on Emerging Technologies and Factory Automation,*, pp. 1813–1820.

https://doi.org/10.17979/spudc.9788497498043.686

[19] Torres-Sanchez, E., Alastruey-Benede, J. and Torres-Moreno, E.: 2020, Developing an AI IoT application with open software on a RISC-V SoC, *Conference on Design of Circuits and Integrated Systems*.

[20] Universidad de Málaga: 2021 (Consultado en julio de 2021), Área de experimentación en nuevas tecnologías para la intervención en emergencias (LAENTIEC).
URL: *www.uma.es/LAENTIEC*

https://doi.org/10.17979/spudc.9788497498043.686