SISTEMA INTELIGENTE DE VIDEOVIGILANCIA DE VEHÍCULOS

JACOBO GONZÁLEZ CEPEDA

Tesis depositada en cumplimiento parcial de los requisitos para el grado de Doctor en

Ingeniería Eléctrica, Electrónica y Automática

Universidad Carlos III de Madrid

Director

José María Armingol Moreno

Tutor

José María Armingol Moreno

Julio 2023

"Apprendre à voir est le plus long apprentissage de tous les arts."

Jules de Goncourt

AGRADECIMIENTOS

Después de un largo camino de seis años, parece que esto llega a su fin. Seis años de idas y venidas, de peleas, de sin sabores, de renunciar a momentos y a personas con que vivirlos, de tirarse al mar, empezar nadar y no ver nunca la orilla. Y sin embargo, tras estos seis años únicamente puedo tener y mostrar gratitud hacia todas las personas que me han ayudado a lo largo de esta travesía, un camino imposible de recorrer solo.

En primer lugar, debo agradecer a José María su continua ayuda totalmente desinteresada, un apoyo fundamental sin el cual llegar hasta el final no habría sido posible.

La siguiente persona, clave en este documento es Álvaro, un genio entre genios, o como diría José María, uno entre un millón. Sin él, todo esto no habría sido posible.

Fue Jim Rohn el que dijo que "una persona es el promedio de las cinco personas que te rodean". Por suerte, son más de cinco las personas que me rodean, y con ser simplemente la mitad de ellos ya sería extraordinario. Eso incluye a mis compañeros. Me han dado todas las facilidades y ayudas necesarias, que han sido más de las que me hubiese gustado y merecía.

También incluyo en este grupo a mis amigos, que son el mayor tesoro que puede tener una persona. Siempre dispuestos a ayudarme, apoyarme, a hacerme reír ("cuñadear") y a confiar en los peores momentos.

Por supuesto, mis padres son una parte fundamental. Son los renglones que sujetan este trabajo, que me han hecho avanzar con paso firme independientemente del camino a recorrer. Me han hecho dar este salto, pero también a mantener los pies en el suelo.

Y por último, debo citar a una persona que me ha acompañado en estos últimos años, que ha puesto orden a mi caos, que me ha enseñado que los recursos humanos son más importantes que los materiales (esta investigación es el claro ejemplo) y que las calabazas también sonríen.

RESUMEN

Los sistemas de visión son herramientas esenciales en muchas áreas, incluyendo la seguridad y la prevención de delitos. Una de las áreas de investigación más importantes en este campo es la identificación de vehículos, principalmente a través de las matrículas, que permiten no solo conocer la marca y el modelo del vehículo, sino también al propietario.

Si bien hay muchas soluciones para la detección y lectura automática de matrículas, pueden surgir dificultades si los caracteres están parcialmente ocultos o distorsionados. A pesar de los avances en la inteligencia artificial y las redes neuronales, estos problemas aún no se han resuelto completamente.

La investigación presentada busca abordar este problema utilizando un enfoque basado en la autenticación de dos factores. En este caso, los dos factores son los caracteres de la matrícula del vehículo y la información visual adicional que el vehículo proporciona.

El resultado es un sistema que implementa varias redes neuronales en cascada, que utiliza dos procedimientos para identificar vehículos: uno basado en la lectura de los caracteres de la matrícula y otro que aprovecha las características visuales del vehículo. Esta combinación de dos métodos de identificación muy diferentes proporciona al sistema una mayor versatilidad y soluciona los problemas relacionados con el reconocimiento visual de los vehículos.

PALABRAS CLAVE: sistemas de vigilancia; re – identificación de vehículos; ALPR; aplicaciones en tiempo real; algoritmos de aprendizaje profundo

ABSTRACT

Vision systems are essential tools in many areas, such as security and crime prevention. One of the most important areas of research in this field is vehicle identification, mainly through number plates, which allow not only to know the brand and model of the vehicle, but also the owner.

While there are many solutions for automatic number plate detection and reading, difficulties can arise if characters are partially hidden or distorted. Despite advances in artificial intelligence and neural networks, these problems have not yet been completely solved.

The research presented in this document pursuits to address this problem using an approach based on two-factor authentication. In this case, the two factors are the vehicle's number plate characters and additional visual information provided by the vehicle.

The result is a system that implements several neural networks in cascade, using two procedures to identify vehicles: one based on number plate characters reading and another based on vehicles' visual characteristics. This combination of two very different identification methods provides the system greater versatility and solves the problems related to the visual recognition of vehicles.

KEYWORDS**:** surveillance systems; vehicle re-identification; ALPR; real-time applications; deep learning algorithms

PUBLICACIONES

Algunas ideas, tablas y figuras utilizadas han aparecido previamente en las siguientes publicaciones:

CONTENIDOS PUBLICADOS Y PRESENTADOS:

Artículos en revista:

- González-Cepeda, J., Ramajo, Á., & Armingol, J. M. (2022). Intelligent Video Surveillance Systems for Vehicle Identification Based on Multinet Architecture. *Information*, *13*(7), 325.
  Este artículo ha sido incluido parcialmente en esta tesis en los capítulos 2 y 5. La inclusión en la tesis del material de esta fuente se especifica en una nota de pie de página de cada capítulo en el que se incluye. El material de esta fuente en esta tesis no está señalado con medios tipográficos y referencias.

Artículos en congreso:

- Ramajo Ballester, Á., González Cepeda, J., Armingol Moreno, J. M. (2022). Vehicle re-identification in road environments using deep learning techniques. ITS European Congress.
  Este artículo ha sido incluido parcialmente en esta tesis en el capítulo 5. La inclusión en la tesis del material de esta fuente no se especifica en una nota de pie de página de cada capítulo en el que se incluye. El material de esta fuente en esta tesis no está señalado con medios tipográficos y referencias.
- Ramajo Ballester, Á., González Cepeda, J., Armingol, J. M., & Escalera, A. D. L. (2022). Reidentificación de vehículos mediante técnicas de deep learning. In *XLIII Jornadas de Automática* (pp. 1031-1039). Universidade da Coruña. Servizo de Publicacións.
  Este artículo ha sido incluido parcialmente en esta tesis en el capítulo 5. La inclusión en la tesis del material de esta fuente no se especifica en una nota de pie de página de cada capítulo en el que se incluye. El material de esta fuente en esta tesis no está señalado con medios tipográficos y referencias.

- Ramajo Ballester, Á., González Cepeda, J., & Armingol Moreno, J. M. (2022, November). Deep Learning for Robust Vehicle Identification. In *ROBOT2022:*

Fifth Iberian Robotics Conference: Advances in Robotics, Volume 1 (pp. 346-358). Cham: Springer International Publishing.

Este artículo ha sido incluido parcialmente en esta tesis en el capítulo 5. La inclusión en la tesis del material de esta fuente no se especifica en una nota de pie de página de cada capítulo en el que se incluye. El material de esta fuente en esta tesis no está señalado con medios tipográficos y referencias.

MATERIAL DE TERCEROS

Material escrito por el autor de la tesis:

- Las Figuras 11, 16, 17, 18, 19 y 20 no requieren un permiso especial para reutilizar todo o parte del artículo publicado por MDPI. En el caso de los artículos publicados bajo una licencia Creative Common CC BY 4.0 de acceso abierto, cualquier parte del artículo puede ser reutilizada sin permiso siempre que el artículo original sea claramente citado. La reutilización de un artículo no implica el respaldo de los autores o de MDPI.

OTROS MÉRITOS DE INVESTIGACIÓN

Trabajos de fin de grado tutelados:

- García Serrano, A. Aplicación de Sistemas de Percepción Para la Seguridad Vial; *Departamento de Ingeniería Eléctrica, Electrónica y Automática*, Universidad Carlos III: Madrid, España, 2020.

ÍNDICE DE CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE ECUACIONES

ACRÓNIMOS

ALPR	*Automatic License Plate Reader (*Lector automático de matrículas)
ANN	*Artificial Neural Network* (Red neuronal artificial)
CIoU	*Complete Intersection over Union* (Intersección sobre unión complete)
CNN	*Convolutional Neural Network* (Red Neuronal convolucional)
CUDA	*Compute Unified Device Architecture*
cuDNN	*CUDA Deep Neural Network*
DORI	*Detection, Observation, Recognition, Identification* (Detección, observación, reconocimiento e identificación
FNN	*Feedforward Neural Networks* (Redes neuronales prealimentadas)
FPS	*Frames per Second* (Fotogramas por segundo)
GPS	*Global Positioning System* (Sistema global de posicionamiento)
GPU	*Graphic Processor Unit* (Unidad de procesamiento gráfico)
GSM	*Global System for Mobile Communications* (Sistema global de Comunicaciones móviles)
IBN	*instant batch normalization* (Normalización instantánea por lotes)
KNN	*K nearest neighbours* (K vecinos próximos)
LSTM	*Long Short-Term Memory Networks* (Redes de memoria a corto y largo plazo)
LTE	*Long Term Evolution* (Evolución a largo plazo)
mAP	*Mean Average Precission* (Precisión media)
OCR	*Optical Character Recognition* (Reconocimiento óptico de caracteres)
PANet	*Path Aggregation Network* (Red de agregación de caminos)
PoE	*Power over Ethernet* (Energía sobre Ethernet)
ReLU	*Rectified Lineal Unit* (Unidad lineal rectificada)
RMSProp	*Root Mean Square Propagation* (Propagación de la media de la raíz cuadrada)
RNA	Redes Neuronales de Atención
RNN	*Recurrent Neural Networks* (Redes neuronales recurrentes)
ROI	*Region of Interest* (Región de interés)
SGD	*Stochastic Gradient Descent* (Descenso de gradiente estocástico)
SiLU	*Sigmoid Linear Unit* (Unidad lineal sigmoide)
SNN	*Support Nearest Neighbours* (Apoyo de vecinos más cercanos)
SPP	*Spatial Pyramid Pooling* (Agrupación de pirámides espaciales)
SVM	*Support Vector Machine* (Máquina de soporte vectorial)

I INTRODUCCIÓN

El mantenimiento de la seguridad es una de las principales tareas que todo Estado moderno debe asegurar para garantizar su normal funcionamiento. Por ello, se destinan gran cantidad de recursos tanto humanos como materiales con dicha finalidad. Pero no se trata de una responsabilidad única del Estado, también el ámbito privado tiene necesidad y capacidad de poder ofrecerla. De hecho, la seguridad es tanto un valor en sí mismo como un valor añadido que enriquece los productos y servicios ofrecidos. De ahí que sea muy importante destinar esfuerzos para incrementar y mejorar las capacidades en este campo.

En aras de poder garantizar esa seguridad tanto en personas como en bienes, se recurre a gran cantidad de herramientas, muchas de ellas encuadradas en el ámbito tecnológico. Y dentro de ese ámbito, el empleo de cámaras de vigilancia juega un papel preponderante. Sin embargo, su utilización de forma desproporcionada puede causar un perjuicio más que suponer una ayuda, de ahí la necesidad de hacer un uso inteligente y racional de estos dispositivos. De hecho, lo normal es que su uso vaya acompañado de operadores responsables de su visionado y control.

La cantidad de información que estos dispositivos son capaces de proveer es muy abundante, haciéndose necesario disponer de mecanismos complementarios que permitan analizar y procesar correctamente esta información. Es en este campo donde las herramientas de procesamiento de imágenes gozan de una gran relevancia, y los recientes desarrollos en inteligencia artificial están teniendo un protagonismo especial. Sin embargo, al igual que sucede con el empleo de cámaras en sí mismo, estas herramientas deben desarrollarse para cumplimentar cometidos específicos.

En el campo de la seguridad, las personas son a la vez el principal activo y agente perturbador. Por lo tanto, parece lógico que la mayoría de recursos y avances se centren en su correcta detección e identificación. Sin embargo, las limitaciones lógicas asociadas al empleo de cámaras, provoca que se deban emplear otro tipo de elementos que puedan facilitar dicha identificación. Aquí los vehículos son la principal alternativa. Y es que, a día de hoy, las personas hacemos uso de los medios de transporte para prácticamente cualquier actividad que desarrollamos. Por lo tanto, parece que el empleo de herramientas de detección e identificación de vehículos es también muy importante.

Aunque en este campo ya existen numerosos avances, sobre todo centrados en la correcta lectura de matrículas, todavía queda mucho margen de mejora en herramientas destinadas a la identificación de vehículos, tanto por capacidades como por fiabilidad. Por lo tanto, la presente investigación pretende contribuir al desarrollo de una herramienta versátil, práctica y sobre todo útil, que permita contribuir a la correcta detección e identificación de vehículos en el ámbito de la seguridad, y más concretamente, en el de la seguridad ciudadana. Y para ello, la investigación se va a centrar, por un lado, en analizar y definir el ámbito de actuación, problemas existentes y áreas de mejora, y, por otro lado, en el desarrollo de algoritmos encuadrados en el campo del procesamiento de imágenes que faciliten esa detección e identificación de vehículos.

1.1.- Transformación digital: búsqueda y aplicación útil de la tecnología

El desarrollo y avance de la tecnología tiene como finalidad última contribuir a incrementar las capacidades del ser humano. Puede apreciarse en cualquier aspecto, como por ejemplo las comunicaciones, el ocio, la economía… es en definitiva uno de los principales conductores en la evolución humana. Por lo tanto, tan importante son los avances logrados, como identificar dónde pueden tener un uso práctico y directo. Existen ejemplos en los que el desarrollo de una tecnología tuvo una utilidad práctica con posterioridad a su descubrimiento, teniendo ejemplos tan masivamente empleados a día de hoy como la electricidad, Internet o la inteligencia artificial, o también situaciones en los que una misma tecnología ha tenido multitud de campos en los que utilizarse a pesar de ser muy dispares entre sí (los ejemplos anteriores pueden ser perfectamente válidos).

Precisamente es esa combinación del desarrollo tecnológico con la búsqueda de áreas de aplicación directa la que llega incluso a provocar cambios socioculturales, algunos tan grandes que modifican por completo la estructura y comportamiento de una sociedad, como sucedió durante la Revolución Industrial a lo largo del siglo XIX. La evolución y el desarrollo de la tecnología es constante y cada vez más rápida, lo que está generando una necesidad en la población de ser capaz de asimilar y gestionar cambios en los modelos de manera continua. La velocidad de las comunicaciones, que se van a ver aún más incrementadas con el despliegue completo del 5G de alta velocidad [1] y, sobre todo, una evolución masiva en la inteligencia artificial, contribuye a que todas las organizaciones quieran adoptar como modelo de comportamiento una corriente denominada Transformación Digital.

La transformación digital [2,3] no sólo hace referencia a una "digitalización" de las organizaciones (es decir, la eliminación del papel), ni a la inclusión de la última tecnología como herramientas disponibles. Pretende ser una guía hacia un nuevo modelo que persigue la evolución y adaptación permanente de una organización de tal manera que la capacidad de decisión se encuentre en el lugar más bajo posible, tratando de eliminar las estructuras fuertemente jerarquizadas y potenciando un modelo horizontal. Con ello se busca, por un lado, tratar de mejorar la eficiencia, y por otro, sobrevivir y sacar partido de los cambios tecnológicos que cada vez evolucionan más rápidamente y que pueden llegar a dejar obsoletos modelos de negocio que parecían imposibles de desaparecer. Nokia [4] o Blackberry [5] son el ejemplo de dos compañías punteras en el campo de la telefonía móvil, y que con la aparición de los smartphones se han visto abocados a una presencia puramente testimonial.

La transformación digital también conlleva que a nivel de dirección organizativa y técnica, se tenga un conocimiento de las capacidades tecnológicas y, sobre todo, de cuál es el enfoque correcto que pueden tener dentro de su modelo de negocio y del producto que ofrecen. Un posible ejemplo de mala implementación o de mala lectura sobre cuando implementar determinada tecnología, podría ser la apuesta masiva de Facebook (o mejor dicho, Meta), por una inclusión masiva en el metaverso. De hecho, hay compañías con un alto valor en bolsa que se han desplomado al comprobarse que únicamente contaban con pocos usuarios en dicho metaverso. O el ejemplo de los llamados NFT (*non fungible tokens*) [6], que durante un corto espacio de tiempo se emplearon como medio de especulación para tratar de generar activos virtuales.

Este apartado pretende por tanto poner de manifiesto que es tan importante la evolución tecnológica como saber cómo, dónde y cuándo debe utilizarse y, sobre todo, cuáles son sus implicaciones, capacidades y limitaciones reales. Esta idea es aplicable a casi cualquier campo, pero sobre todo debe serlo en el ámbito de la seguridad.

1.2.- Seguridad

La seguridad es un concepto muy amplio que afecta absolutamente a todos los ámbitos de la sociedad. La propia R.A.E. (Real Academia Española de la Lengua), en el Diccionario, refleja una serie de términos en el que la palabra seguridad lleva asociado un apellido que cambia por completo su significado [7]. Por ejemplo, define la seguridad activa como "características o prestaciones que previenen accidentes (caso de vehículos)"; la seguridad ciudadana como la "situación de tranquilidad pública y de

libre ejercicio de los derechos individuales"; y la seguridad jurídica como una "cualidad del ordenamiento jurídico que implica la certeza de sus normas".

Pero, ¿qué es la seguridad? Se puede considerar un valor fundamental en la sociedad. Maslow [8] lo establece en su pirámide en segundo lugar en importancia (por detrás justo de las necesidades fisiológicas). Además, no habla de la seguridad como un valor concreto, si no que marca diferentes ámbitos, como la seguridad física, de recursos, moral, familiar, de salud y de propiedad privada.

Bajo un prisma legal, se considera un derecho fundamental en las sociedades modernas, tal y como recoge la Constitución Española en el artículo 17.1 [9]: "Toda persona tiene derecho a la libertad y a la seguridad. Nadie puede ser privado de su libertad, sino con la observancia de lo establecido en este artículo y en los casos y en la forma previstos en la ley". Sin embargo, la seguridad no es una competencia exclusiva del ámbito público.

Las empresas privadas también ofrecen seguridad [10], o la emplean como un valor añadido diferenciador de sus competidores. Pueden ser proveedoras de seguridad física en instalaciones, de personas, de bienes, o también pueden dotar sus servicios de determinados grados de seguridad. Por ejemplo, una empresa puede proporcionar de forma específica servicios de seguridad en las comunicaciones (como mecanismos de almacenamiento seguro, comunicaciones cifradas punto a punto, comunicaciones desvinculadas, etc.), o que estas medidas sean un complemento extra a un servicio de comunicaciones que incluya además estas características (como hace un proveedor de telecomunicaciones). Pero también pueden ofrecer como seguridad la garantía de que dichas comunicaciones no se van a perder o a interrumpir. Dependiendo del enfoque, aun estando relacionados entre sí, se ponen de manifiesto tres modelos de negocio con dos conceptos de seguridad, en este caso como sinónimo de garantía. Por un lado, garantía de privacidad en las comunicaciones y por otro, garantía de infraestructura; y ambos casos son ejemplos de seguridad en las comunicaciones.

Se puede por tanto manifestar que la seguridad es un término global con una vertiente genérica y otra más específica. En la presente investigación subyace un objetivo íntimamente ligado a la seguridad de forma específica, y más concretamente, a la seguridad ciudadana. Este matiz es importante de reseñar ya que, si bien los desarrollos realizados tienen cabida perfectamente en otro tipo de ámbitos, ha sido precisamente esa motivación de seguridad ciudadana la que ha permitido identificar un área de mejora,

unos objetivos concretos y una metodología de trabajo acorde a conseguir dichos objetivos.

1.3.- Seguridad ciudadana y la videovigilancia

El preámbulo de la Ley Orgánica 4/2015 [11] define seguridad ciudadana como "la garantía de que los derechos y libertades reconocidos y amparados por las constituciones democráticas puedan ser ejercidos libremente por la ciudadanía" y como "actividades dirigidas a la protección de personas y bienes y al mantenimiento de la tranquilidad ciudadana". Está íntimamente ligada al contenido del artículo 17.1 de la Constitución anteriormente reseñado, y su desarrollo es una garantía del normal funcionamiento de las instituciones u organizaciones.

Para su correcta implementación, debe cubrir tres escenarios (íntimamente ligados a los procedimientos globales de trabajo de las Fuerzas y Cuerpos de Seguridad, también recogidos en [11]): la prevención, la reacción y la investigación.

La prevención, se define como el conjunto de actividades destinadas a evitar la comisión de un delito. Bajo un punto de vista temporal, se situaría en el "antes". Para ello, se emplean medidas disuasorias como, por ejemplo, patrullas de personal uniformado, cámaras de videovigilancia visibles, cerraduras de difícil apertura o protecciones como persianas metálicas o barrotes.

La reacción, son las actividades realizadas cuando se tiene conocimiento de la comisión de un delito o una infracción, se conocen los autores, y se actúa para detener a los mismos y ponerlos a disposición judicial o imponer una sanción al respecto.

Por último, la investigación entra a jugar cuando se ha producido un hecho contrario a la ley, pero aún falta información para esclarecerse correctamente (porque se desconocen los autores o porque se encuentran en paradero desconocido). Por lo tanto, se llevan a cabo acciones para conocer correctamente los hechos, identificar a los autores y conseguir evidencias que puedan ser puestas a disposición de la autoridad correspondiente.

La investigación a su vez también está relacionada con la reacción y la prevención, ya que, en caso de la comisión de determinados delitos, sirve para evitar que se sigan perpetrando. Por ejemplo, cuando se consigue detener a una banda de ladrones de vehículos, se está evitando también que dicha banda continúe delinquiendo.

La prevención, la reacción y la investigación tienen procedimientos comunes, y muchos de ellos se apoyan cada vez más en el uso de la tecnología (tal y como se ha escrito en el preámbulo de la introducción). Existen numerosos ejemplos donde los denominados medios técnicos permiten incrementar las capacidades en materia de seguridad [12], como, por ejemplo, los dispositivos de control de acceso biométrico o las alarmas con sensores magnéticos, térmicos y sísmicos.

Aunque sin duda alguna, el empleo de cámaras es uno de los mecanismos más utilizado en el campo de la seguridad, cumpliendo multitud de funciones que además, son comunes a los tres aspectos de la seguridad ciudadana anteriormente reseñados.

El uso de cámaras puede resultar intimidatorio, ya que entre un establecimiento con cámaras y uno que no las tenga, parece que los criminales se van a decantar por actuar en el que no disponga de dichas cámaras. Es decir, el propio sistema de vigilancia está teniendo una labor disuasoria (prevención). Pero, en caso de que se produzca igualmente el robo, las cámaras van a ayudar a esclarecer dicho delito (investigación). Por lo tanto, se puede deducir que el empleo de cámaras (mejor dicho, de sistemas de videovigilancia), juega un papel fundamental en la preservación integral de la seguridad ciudadana.

De ahí que precisamente, la implementación de cámaras de videovigilancia sea una de las principales herramientas en materia de seguridad gracias a su versatilidad y sus capacidades. Además que, según en qué determinados aspectos, es una de las medidas que bajo el prisma legal menos requisitos exigen si se compara con otras medidas que afectan más directamente a los derechos fundamentales [13] (no obstante, su uso está sometido a numerosa normativa reguladora [14-16]. De ahí que además se convierta en un elemento básico para las labores de prevención y de investigación. Por lo tanto, los desarrollos y avances en este campo pueden tener un alcance elevado debido a la transversalidad de sus aplicaciones.

1.4.- Motivación específica

Como se ha explicado en el apartado anterior, la realización de esta investigación surge de una necesidad práctica existente en el campo de la seguridad en una circunstancia muy concreta, pero a la vez bastante habitual. Sea por una infracción cometida por un usuario o por la comisión de un delito en el que ha participado un vehículo (algo muy habitual), el vehículo tiene información que sirve para ayudar a

identificar al posible autor. Lo habitual es apoyarse en la matrícula, que a nivel legal es el elemento concebido para identificar un vehículo concreto y lo vincula con un propietario [17]. La normativa específica desarrolla las características que debe tener la matrícula en cuanto a tamaño, tipografía y visibilidad de los caracteres. De ahí que la mayoría de herramientas desarrolladas se hayan centrado históricamente en detectar, identificar y leer correctamente la matrícula.

Para poder conseguir este objetivo, es necesario que se produzca una correcta detección y lectura de la matrícula. Sin embargo, estos procesos están supeditados a multitud de factores. El principal y más importante es conseguir una imagen adecuada de la matrícula, y esto es bastante habitual que no suceda.

Un ejemplo son los radares de velocidad. Cuando el radar detecta una medición de velocidad superior a la marcada como referencia, lleva a cabo una o varias tomas en las que se recoja el vehículo infractor [18]. Sin embargo, ¿qué sucede cuándo en la fotografía obtenida, el vehículo infractor está rebasando a otro, y sin embargo la única matrícula reconocible es la del vehículo adelantado? ¿O si hay dos vehículos circulando simultáneamente con exceso de velocidad?

Figura 1: imagen desde la perspectiva de un radar ubicado en el arcén con una oclusión total de una de las matrículas

Este problema se podría solucionar bien cambiando la posición del sensor (como sucede por ejemplo en la Figura 2) o bien mediante el apoyo de un operador humano, capaz de hacer una interpretación correcta de la imagen e identificar (por posición en la imagen, marca y modelo) quién es el infractor.

Figura 2: ejemplo de una toma cenital de una cámara oficial de la DGT

Otro ejemplo análogo sucede cuando se ha producido un robo o un altercado, y las únicas imágenes disponibles son las del vehículo empleado por el infractor, pero no se ha conseguido una reseña completa de la matrícula. En la imagen existe más información que se podría utilizar, pero eso requiere un correcto análisis. Las Figuras 3 y 4 muestran un caso real de como aun disponiendo de dos imágenes claras de un vehículo objetivo, circunstancias como la reflexión de la luz en la matrícula o la falta de resolución de las cámaras impiden identificar correctamente la matrícula.

Figura 3: imagen del vehículo objetivo (lugar donde se producen los hechos)

Figura 4: imagen del vehículo objetivo abandonando el lugar de los hechos

El ejemplo concreto sirve para ilustrar la motivación principal de la investigación. En este caso, las imágenes aportan muchísima información para poder facilitar la identificación, como son la marca, modelo y color del vehículo; es decir, sus características. Estos elementos no son inequívocos de por sí (es decir, existen, como es lógico, varios vehículos que coincidan) aunque si a esa información se le añade un dato adicional (como por ejemplo algún número parcial de la matrícula), se puede conseguir identificar plenamente un vehículo. Sobre todo con un caso como el de las imágenes, en las que el vehículo es muy característico.

Hilvanando con el ejemplo anterior, lo habitual para poder establecer la ruta seguida por dicho vehículo y centrar una posible ubicación o domicilio es recurrir a cualquier tipo de cámara que haya captado imágenes del citado vehículo. Para ello, se suele acotar la búsqueda a cámaras próximas al lugar de los hechos y tratar de centrarla en base a la franja horaria en la que se ha visualizado el vehículo, como es el caso de las Figuras 3 y 4. Pero si no se dispone de alguna ayuda para centrar la búsqueda, puede ser una tarea muy difícil y sobre todo, requerir mucho tiempo. Por lo tanto, la idea detrás de la presente investigación es encontrar una herramienta versátil que facilite esta labor de localizar un vehículo en multitud de escenarios, apoyándose en toda la información que pueda ofrecer una imagen de dicho vehículo para favorecer su identificación. Concretamente, los dos elementos más características y a la vez genéricos: la matrícula y el contorno.

La búsqueda general de una solución que recurra a dos elementos para llevar a cabo una identificación, así como la manera de trabajar sobre uno de estos dos elementos (como se explicará más adelante), se apoya en conceptos utilizados para la identificación de personas. El primero es la aplicación del doble factor de autenticación. El segundo concepto es la re – identificación.

1.4.1.- Doble factor de autenticación

Consiste en la identificación de una persona y/u objeto mediante la verificación combinada de dos agentes o características inequívocas asociadas al elemento identificado [19,20]. Es un término especialmente empleado en la seguridad [21], sobre todo en controles de accesos físicos y virtuales. El ejemplo más habitual del empleo del doble factor de autenticación se puede encontrar en la utilización de un usuario/contraseña y de un elemento biométrico (huella dactilar o reconocimiento ocular) [22] para realizar una operación bancaria o para franquear una barrera física (como una puerta o torno de seguridad).

Como se puede apreciar, este concepto parece estar más centrado en las personas, ya que primero, suelen ser los principalmente susceptibles de ser identificados, y segundo, es más fácil que presenten características inequívocas (como los factores biométricos).

Es un método que pretende aumentar la seguridad al incluir dos elementos que permitan la identificación, no únicamente uno sólo como podría ser el mecanismo usuario y contraseña. Pero precisamente debido a la importancia que se le quiera dar a la seguridad y como no es recomendable depositar plena confianza en los procedimientos automáticos [20] (todo depende del entorno y del grado de seguridad), también existe la figura del operador humano para realizar esa doble o incluso triple autenticación, o como un elemento de control. Esto es habitual cuando se realiza una inspección de documentación, en la que un agente realiza una verificación visual de las credenciales requeridas, de la fotografía asociada y del usuario que la otorga, como sucede en aeropuertos o incluso en estadios de fútbol.

La existencia de un responsable es muy importante, porque las herramientas de seguridad en general (como la que se ha desarrollado en esta investigación), no deja de ser exactamente eso, una herramienta. No pretende sustituir, si no complementar y facilitar una labor que no debe ser desatendida en su totalidad ya que es muy importante recalcar que <u>la fiabilidad del cien por cien no existe</u>.

Por lo tanto, pensando precisamente en la figura de ese responsable de visionar y controlar las imágenes captadas por cámaras de seguridad, se ha analizado cuál suele ser el principal problema: el análisis preciso y en tiempo de la información. Hilvanando con el ejemplo reseñado en las Figuras 3 y 4, para tratar de obtener más datos del vehículo, lo normal sería tratar de analizar otras cámaras próximas. Sin embargo,

dependiendo de otros criterios como la ubicación en el plano de las cámaras, distancia al punto inicial y tiempo transcurrido desde los hechos, puede resultar una tarea casi imposible. Una reinterpretación del doble factor de autenticación, vinculando la matrícula y la silueta del vehículo puede agilizar notablemente estos procedimientos de búsqueda.

La reinterpretación consiste en utilizar estos dos elementos de manera adicional entre sí y no de forma excluyente, de modo que, siendo el operador humano el elemento que realiza la discriminación definitiva, la herramienta detectaría aquellos vehículos que cumplan uno de los parámetros de búsqueda (imagen del vehículo y/o matrícula).

1.4.2.- Re – identificación

En el ámbito de la seguridad se pueden distinguir cuatro niveles de "precisión", denominados "concepto DORI" [23] (acrónimo de detección, observación, reconocimiento e identificación). Este estándar, incluido en la Norma Internacional IEC EN62676-4: 2015, define la resolución en píxeles que debe tener una imagen para que se pueda realizar en ella la detección, observación, reconocimiento e identificación de objetos/personas en imágenes. Es un parámetro que suelen incluir los fabricantes de cámaras de vigilancia en sus fichas técnicas.

Está directamente relacionado con tamaño del objeto incluido en un fotograma y, a su vez, con su distancia al sensor. La Figura 5 muestra visualmente como cuanto más cerca esté el objetivo del captador, más información se tendrá sobre él, avanzando progresivamente por los cuatro niveles de precisión.

La detección (esquina superior izquierda) es la capacidad del sistema para captar algún movimiento, suceso o elemento activo. En este ejemplo concreto, una persona. Sería el primer paso para "activar" diferentes mecanismos, ya que se trata de un dispositivo de alerta temprana.

La observación (esquina superior derecha) corresponde a la capacidad de apreciar los posibles movimientos de este nuevo activo. Este rango de distancia permite analizar o estudiar las posibles "intenciones". Como se muestra en la segunda imagen, suele ser posible cuando el nuevo objetivo se acerca a la cámara, lo que permite verlo con mejor resolución, pero no lo suficiente como para reconocerlo.

El reconocimiento (esquina inferior izquierda) es la capacidad de observar si el activo es conocido previamente o no. En este caso, este término afecta al responsable del visionado de la cámara, ya que es el ente capaz de realizar este reconocimiento.

Por último, la identificación (esquina inferior derecha) es la capacidad de apreciar suficientes elementos característicos en la imagen para que el activo sea reconocible en posteriores ocasiones. La última de las cuatro imágenes contiene suficientes rasgos de la persona para hacerse una idea inequívoca de ella. Estas dos últimas capacidades, reconocimiento e identificación, son los principales objetivos que persigue un sistema de videovigilancia.

Figura 5: ejemplo de DORI (detección, observación, reconocimiento e identificación)

Las tareas de reconocimiento e identificación desarrolladas por los sistemas de videovigilancia suelen realizarse principalmente sobre personas y en menor medida, vehículos. Para identificar a una persona, existen diferentes características biométricas que son individuales e imposibles de replicar, como la cara, la voz, los ojos o incluso la disposición de los vasos sanguíneos [24,25].

Todos estos sistemas necesitan imágenes con una resolución mínima para poder cumplir sus objetivos. De ahí que los mecanismos de iluminación y las ópticas y sensores traten de obtener las imágenes en las condiciones más óptimas, para que el tratamiento de estas señales ofrezca los mejores resultados. A rasgos generales, el procedimiento para realizar una identificación consiste en detectar una cara, extraerla, y cotejarla con una base de datos para saber quién es esa persona. Sin embargo, ¿se trata de una identificación? o, ¿realmente es un reconocimiento?

Debido al gran coste computacional que requieren los algoritmos de inteligencia artificial aplicados, la gran mayoría de los sistemas de identificación en realidad ejecutan tareas de reconocimiento a gran escala [26]. Una identificación puede entenderse como un reconocimiento en el que una muestra "n" se coteja contra una base de datos "N", donde "N" contiene "n" junto con un gran número de elementos.

Por ejemplo, un sistema de identificación facial realmente no “identifica” ninguna cara. Primero codifica una cara como una matriz de características únicas y compara una muestra con una base de datos muy grande (como una base de datos de Documentos de Identidad), en la que los elementos también han sido previamente codificados.

Si se abordase el problema como en el caso explicado en primer lugar, se estaría tratando de una red de clasificación o clasificador, en la que habría tantas categorías como personas entre las que se realiza la identificación. Esto genera un problema. Bajo un enfoque tradicional, para realizar una clasificación correcta, sería necesario disponer de un conjunto de datos muy específico con tantas categorías como elementos (caras) a clasificar. Además, al ser elementos muy similares, también es necesario disponer de un volumen muy grande de imágenes. No es lo mismo que el clasificador distinga entre personas y vehículos, por ejemplo, que sólo entre distintos tipos de vehículos.

Para simplificar este problema se aplica un enfoque diferente. Supongamos un sistema de control de acceso con reconocimiento facial. El sistema, cuando tenga la toma correspondiente a la persona que quiere entrar, comparará esa cara con las que tiene almacenadas en su base de datos. Es decir, indicará si esa cara está o no incluida. A grandes rasgos, este problema puede entenderse como una clasificación binaria, en la que realmente hay dos categorías: “sí” (la imagen corresponde a una de las almacenadas) o “no” (no corresponde). Bajo esta perspectiva, es posible simplificar un problema de clasificación a una mera comparación. Esto se denomina re - identificación [27,28], y es

el enfoque elegido para desarrollar nuestra investigación, aplicándose en este caso a la imagen del propio vehículo.

1.5.- Objetivos

El objetivo final de la investigación es el desarrollo de una herramienta de apoyo en las labores de vigilancia sobre vehículos en el marco de la seguridad ciudadana. Para ello, primero se ha estudiado en detalle:

- Escenarios habituales de uso y aplicación
- Herramientas más utilizadas (concretamente los sistemas de video vigilancia)
- Principales problemas existentes y posibles áreas de mejora
- Ámbito específico de investigación con aplicación directa en este campo

De manera que se ha considerado que la herramienta debe cumplir las siguientes premisas:

- Incrementar las capacidades de sistemas de reconocimientos de vehículos tradicionales
    - Empleo de combinado de capacidades de lectura de matrículas y de reconocimiento visual de vehículos
- Versatilidad para posibilidad de operar en diferentes ecosistemas y situaciones operativas
    - Investigación y desarrollo sobre los mecanismos de procesamiento de las imágenes captadas por los sensores
- Operación en tiempo en tiempo real y en post proceso

Para conseguir estos objetivos, se han desarrollado las siguientes tareas:

- Estudio y análisis del estado del arte en el ámbito de la detección, reconocimiento e identificación de vehículos
- Identificación de los diferentes procedimientos y cuáles podrían ser los de aplicación más directa, centrándose principalmente el procesamiento digital de imágenes mediante algoritmos de inteligencia artificial
- Pruebas de campo de posibles soluciones hasta definir la más adecuada
- Aplicación de mejoras específicas en los algoritmos, centrándose principalmente en su entrenamiento y adaptación
- Análisis de los resultados en los entornos operativos reseñados

1.6.- Estructura

El documento completo consta de 6 capítulos, estructurados con una pequeña introducción y unas conclusiones parciales. El formato es el siguiente:

- Capítulo I: Introducción. Este capítulo versa sobre el trasfondo que motiva los orígenes de la presente investigación, incluyendo el contexto de uso del modelo propuesto y las premisas en las que se apoya.
- Capítulo II: Estado del Arte. Contiene una clasificación de los procedimientos de identificación de vehículos, además de profundizar, desgranar y recopilar las principales tendencias y modelos de identificación basados en elementos visuales.
- Capítulo III: Sistemas de Videovigilancia. Comprende aspectos operacionales que afectan de manera transversal a todos los sistemas de visión utilizados en el ámbito de la seguridad, y que de manera particular han condicionado el transcurso de la presente investigación.
- Capítulo IV: Identificación Visual. Desgrana particularidades teórico prácticas que se han tenido que considerar a la hora de desarrollar y ejecutar el modelo propuesto
- Capítulo V: Sistema de identificación de doble factor. Recoge los avances logrados en la presente investigación, convergiendo en la solución propuesta para abordar los objetivos inicialmente planteados.
- Capítulo VI: Conclusiones y Trabajos futuros. Da fin al presente documento, poniendo de manifiesto las conclusiones derivadas de la realización de este estudio y también reseñando posibles líneas de investigación futuras.

II

ESTADO DEL ARTE

La detección, reconocimiento e identificación de vehículos responde a una serie de necesidades que afectan de manera transversal a diferentes áreas muy dispares entre sí, pudiéndose encontrar ejemplos en transportes [29], infraestructuras [30], industria [31] y por supuesto, seguridad [32]. La finalidad última detrás del empleo y desarrollo de este tipo de herramientas responde a la búsqueda de simplificar y automatizar funciones que tradicionalmente se han realizado de manera más manual, tratando de agilizar procesos para minimizar tiempo. Al final, el tiempo es otro recurso que se pretende emplear de forma más eficiente. Si se analiza el caso concreto de los medios de transporte (que se desplazan a velocidades elevadas), la reducción de tiempos en determinados procedimientos se traduce en ahorro en procesos posteriores [33].

El objetivo principal del trabajo es el desarrollo de un sistema de detección e identificación de vehículos en entornos operativos para el cumplimiento de labores de seguridad, apoyándose para ello en el empleo de técnicas de procesamiento de señal avanzadas. Si se analiza en detalle dicho objetivo, se pueden identificar varios elementos a desgranar para conocer el estado del arte en escenarios concretos que influyen directamente en la presente investigación.

El primer paso es identificar y mostrar de manera global cuáles son las aproximaciones más habituales para realizar las labores de identificación de vehículos. En el ámbito de los transportes, las herramientas de telepeaje son un caso evidente en el cual, la automatización de procesos supone un ahorro de tiempo (y de combustible). Se trata de un dispositivo inalámbrico que identifica al vehículo/usuario que circula a través de una autopista y que, sin necesidad de tener que detenerse, permite abonar (o en su defecto acreditar) al vehículo que ha circulado por el peaje correspondiente. En este ejemplo concreto, esa detección/identificación de vehículo o usuario se ha realizado mediante un dispositivo adicional.

Otro ejemplo de detección/identificación de vehículos, en este caso en el campo de la seguridad, es el módulo e-call que deben tener todos los vehículos desde 2018 en base a la normativa europea [34]. Se trata de un dispositivo alojado en el interior que consta de

un acelerómetro, un módulo de posicionamiento GPS y un módulo de comunicaciones GSM. Este sistema, gracias a la combinación del acelerómetro con el GPS detecta un posible impacto del vehículo usuario y automáticamente realiza una llamada al 112, en la que además de establecer una llamada activa (llamada de voz), transmite también los datos relativos a la ubicación del objetivo y la velocidad en el momento del impacto. Estos datos van asociados al número de bastidor del vehículo, de tal forma que, en combinación con una base de datos que asocie ese número de bastidor al titular (como sucede con las existentes en la Dirección General de Tráfico), permite gestionar de manera eficiente la asistencia sanitaria ya que se puede conocer en tiempo real: ubicación del impacto, posible persona o personas implicadas (si es el titular del vehículo) y gravedad del mismo (según los datos recogidos por el acelerómetro y el GPS).

Figura 6: ejemplo de ubicación de módulo e - call

Los dos ejemplos anteriores sirven para ilustrar dos cuestiones distintas. Por un lado, la versatilidad del empleo de herramientas de detección o identificación de vehículos (en campos tan dispares como los transportes o la seguridad). Por otro lado, que si bien la presente investigación está centrada en el empleo de herramientas de observación visual, existen otro tipo de soluciones igualmente válidas con enfoques muy diferentes y no excluyentes entre sí. Conocer también este tipo de soluciones, aunque sea de manera general, va a ayudar bastante a

entender el punto de situación que motiva la presente investigación y por qué se ha orientado como se ha descrito en este documento.

Analizando el uso de la tecnología destinada a la identificación de vehículos, las técnicas empleadas se pueden dividir en dos categorías: mediante dispositivos electrónicos y mediante herramientas visuales.

La siguiente figura sirve para resumir los diferentes procedimientos:

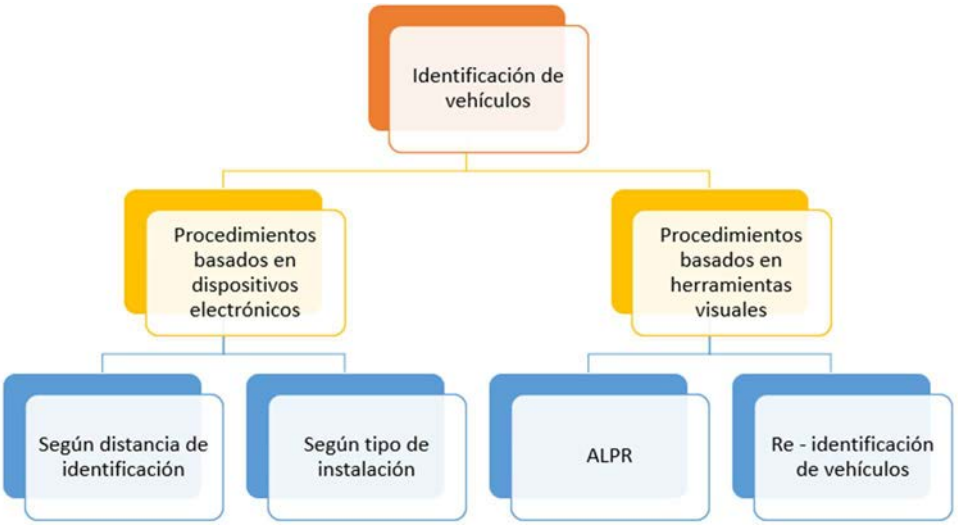


Figura 7: clasificación de procedimientos de identificación de vehículos

2.1.- Procedimientos basados en dispositivos electrónicos

Estos procedimientos, como su propio nombre indica, permiten la identificación de un vehículo mediante el empleo de una serie de dispositivos físicos electrónicos. Se pueden subdividir a su vez según dos criterios: distancia de identificación o tipo de instalación.

2.1.1.- Distancia de identificación

La distancia de identificación está directamente relacionada con el módulo de comunicaciones utilizado por el dispositivo instalado, pudiendo clasificarse como de proximidad y de larga distancia. Cuando se lleva a cabo una identificación, se está produciendo una interacción entre dos elementos: el elemento a reconocer y ante qué o quién se realiza [35]. Por lo tanto, existe una comunicación entre ambos elementos. Al igual que cuando se explique más adelante las circunstancias a las que obedece un sistema de visión, la finalidad es muy relevante cuando se plantea un procedimiento de

identificación de vehículos. Por lo tanto, el hecho de recurrir a mecanismos que actúen en proximidad o que lo hagan a distancia estará supeditado a esa finalidad. Si simplemente la identificación del vehículo responde a un control de accesos, un elemento que actúe en proximidad puede resultar suficiente. En cambio, si se quiere tener un control continuado sobre un objetivo, a priori puede resultar más versátil el empleo de herramientas que actúen a larga distancia.

Clasificar estos elementos como de identificación en proximidad o en distancia lejana es difuso. Bajo un prisma puramente práctico se puede establecer en función de si existe una línea de visión directa (proximidad) o no (larga distancia) entre el punto de origen y de destino de esa comunicación (comunicación punto a punto sin nodos intermedios). Dentro de los dispositivos de proximidad, se pueden encontrar ejemplos como un mando de garaje o un telepeaje [36], que utilizan procedimientos de pregunta – respuesta punto a punto entre el usuario y el elemento de control en un rango de distancia de metros. En cambio, los dispositivos de larga distancia se caracterizan porque su módulo de comunicación utiliza un protocolo que emplea nodos intermedios entre origen y destino, como sucede con la telefonía GSM o la satelital.

La utilización de un determinado tipo de dispositivo específico u otro va a tener una serie de ventajas o de inconvenientes. Por ejemplo, los dispositivos de larga distancia permiten una comunicación "en cualquier circunstancia" entre origen y destino, lo que se aprovecha para poder incluir un módulo de posicionamiento tipo GPS [37]. Esto, a costa de un mayor consumo y de estar normalmente supeditados a una plataforma de gestión. En cambio, los dispositivos de proximidad únicamente van a operar dentro del rango de alcance del módulo de comunicaciones (como se ha dicho anteriormente, de metros) pero con un menor consumo y complejidad. Sobre todo, porque se suelen configurar para ser dispositivos pasivos (telepeaje) o que sea el usuario el que voluntariamente active la comunicación (mando de garaje), estando en un estado latente salvo en el momento que realicen la identificación.

Figura 8: ejemplos de dispositivo de localización en proximidad (izquierda) y a distancia (derecha)

2.1.2.- Tipo de instalación

Esta clasificación diferencia entre los dispositivos ya incorporados directamente en el vehículo y aquellos que requieren una instalación externa. Los diferentes avances en seguridad en los transportes han motivado el desarrollo masivo de tecnología destinada favorecer la interconexión del vehículo con el entorno. Es el ejemplo del módulo e-call anteriormente explicado, que no deja de ser una medida de seguridad. Pero no es el único caso. Los sistemas de *infotainment* [38] de los vehículos permiten hacer funciones de identificación tanto con la propia marca fabricante como con el usuario. Estos sistemas disponen de un módulo de comunicaciones de telefonía (LTE) y un sistema de posicionamiento (normalmente GPS) y permiten no sólo conocer la posición del vehículo, si no también datos asociados (número de bastidor, matrícula, propietario) e incluso interactuar con el propio vehículo controlando el climatizador, la apertura de puertas, etc. Estos dos ejemplos, corresponden a dispositivos de identificación que se encuentran incorporados directamente en el propio vehículo.

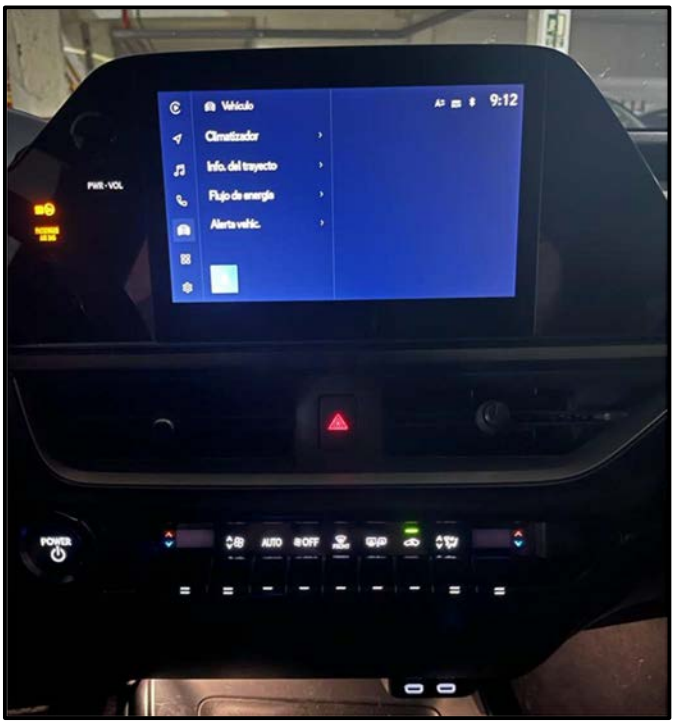


Figura 9: ejemplo de sistema *infotainment* de un vehículo

Respecto a dispositivos que requieren de una instalación externa, el ejemplo más claro y sencillo es el de un rastreador. Estos dispositivos constan normalmente de un sistema de posicionamiento basado en tecnología satelital (GPS, GLONASS, GALILEO, etc.) [39] y de comunicación inalámbrica (como telefonía GSM o satelital) que permite compartir información en tiempo real relativa al vehículo en el que se encuentra instalado dicho dispositivo. Además, este caso concreto también coincide con el de los dispositivos de identificación en larga distancia. En la Figura 8 se puede observar un ejemplo de este tipo de dispositivos.

2.2.- Procedimientos basados en herramientas visuales

Los sistemas de visión son las herramientas más utilizadas para realizar la identificación de vehículos. Un sistema de visión se puede definir como un conjunto de elementos destinados a captar, tratar, almacenar y transmitir información obtenida en forma de fotogramas [40], y dependiendo de su configuración, uso, o capacidades que ofrecen existen multitud de posibilidades. En este caso concreto, la utilización de sistemas de visión persigue captar imágenes que faciliten la identificación del vehículo de forma visual. Para ello, el operador responsable de analizar las imágenes se fijará en los elementos más fácilmente detectables, como por ejemplo la matrícula.

Según la normativa nacional y europea, un vehículo incorpora dos elementos para ser identificado visualmente de forma inequívoca: la matrícula y el número de bastidor. El número de bastidor se asigna a un vehículo concreto cuando sale de fábrica, de tal manera que con dicho número el fabricante puede conocer el modelo, edición, fecha de fabricación y equipamiento completo [41]. Además, se suelen incluir otros datos asociados a componentes del vehículo como el número de serie del motor, de la caja de cambios o de la centralita. Sin embargo, el número de bastidor no es tan fácilmente visible como la matrícula. Se puede encontrar en diferentes lugares como en la parte inferior de la luna delantera, junto al capó, bajo el asiento o en la zona del vano motor, pero no es tan grande ni tan legible desde el exterior como la matrícula, un elemento creado precisamente con esa finalidad.

Figura 10: ejemplo de un número de bastidor

Por ley [17], la matrícula debe ser inequívoca, fácilmente leíble y asociar un titular (propietario) con sus datos con los del vehículo: es decir, nombre y apellidos, dirección y DNI, con marca y modelo del vehículo, número de bastidor y todos aquellos datos que aparecen en la ficha de circulación del vehículo. Además, la normativa regula las características en cuanto a dimensiones, materiales, visibilidad, tipografía y distribución de los caracteres.

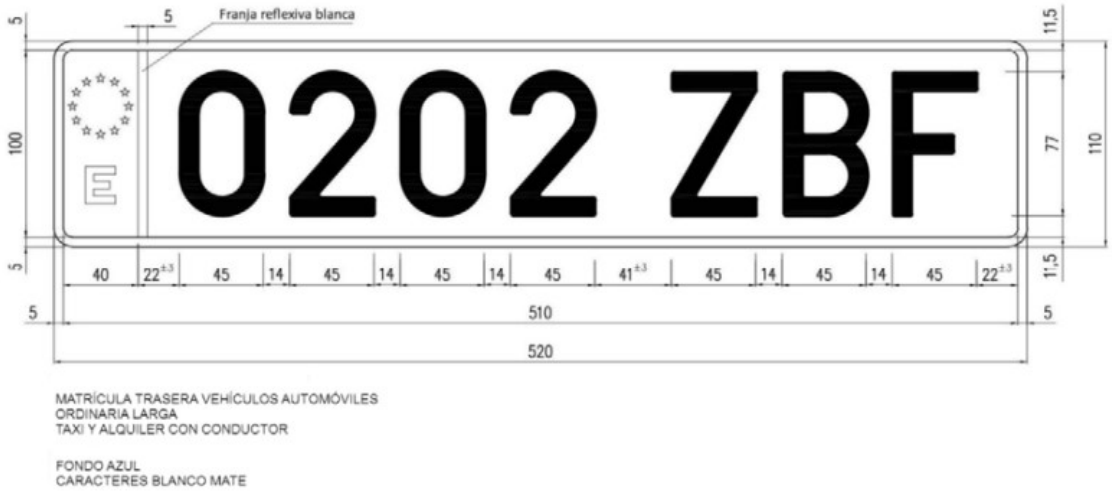


Figura 11: ejemplo de matrícula de un vehículo

Por lo tanto, parece lógico que la mayoría de los procedimientos destinados a la identificación de vehículos se centren en una correcta lectura de la matrícula, aunque también existan algunos para la lectura del número de bastidor [42]. Como se explicará a continuación, comparten puntos en común. No obstante, bajo un prisma puramente visual, una imagen de un vehículo contiene una información asociada que va más allá a la que se pueda obtener de la matrícula. Se puede apreciar su color, la marca y el modelo, e incluso rasgos característicos correspondientes a determinadas versiones específicas del mismo (como las llantas o elementos decorativos).

Esta información, que a priori podría resultar parcial e insuficiente, en ocasiones es de gran utilidad para determinados escenarios, como se muestra en el ejemplo en la Figura 3 y la Figura 4.

Teniendo en cuenta que precisamente el trabajo de investigación pretende afrontar esta doble vertiente, se considera importante que el estado del arte se centre en estos dos campos. Por un lado, el de herramientas basadas en la lectura de matrículas y, por otro lado, aquellas que interpreten la información visual de los vehículos que aparecen en las imágenes.

La gran mayoría de los modelos que se van a mostrar a continuación están íntimamente relacionadas con algoritmos de inteligencia artificial, y más concretamente, con el empleo de redes neuronales. Por lo tanto, posteriormente se incluirá un capítulo exclusivamente dedicado a desgranar gran parte de estos algoritmos, así como consideraciones a tener en cuenta que tendrán influencia directa en la presente investigación.

2.1.1.- ALPR

Estas siglas corresponden a *Automatic License Plate Reader*, o lector automático de placas de matrícula, y es el nombre que habitualmente se utiliza en las investigaciones centradas en la lectura de matrículas para referirse a este tipo de herramientas. Según el procedimiento seguido se pueden distinguir dos tipos de arquitecturas: multifase y de una sola fase [43,44].

2.1.1.1.- Arquitecturas multifase

Las arquitecturas multifase se caracterizan por dividir la lectura de matrícula en tres procesos: detección, preprocesamiento y lectura de los caracteres.

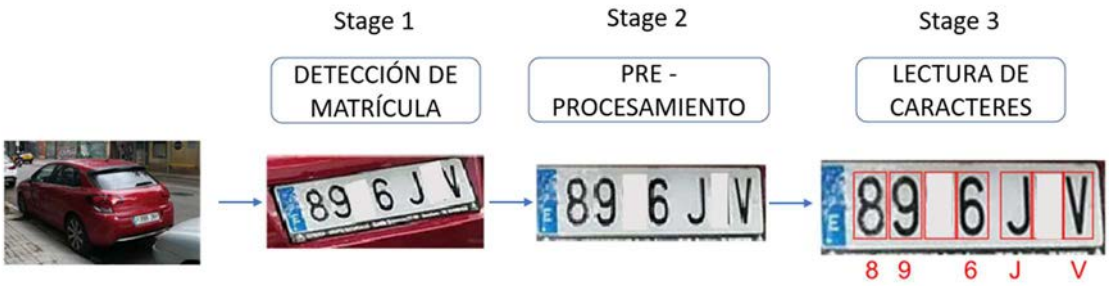


Figura 12: esquema de un ALPR multifase

Este tipo de arquitecturas son las que tienen un recorrido histórico más largo, como se puede apreciar en [45], por lo que se ha decidido situar en primer lugar. En este caso, la detección de la matrícula persigue lo que en procesamiento de imágenes se denomina ROI (*region of interest*) o región de interés. Posteriormente, esa ROI se adecúa bajo determinados procedimientos para que finalmente se puedan interpretar correctamente los caracteres incluidos en dicha ROI.

Detección de la matrícula

Desde un punto de vista práctico, los procedimientos de detección de la matrícula se pueden clasificar entre métodos tradicionales y clasificadores, basado en [43] aunque con alguna ligera modificación, como se puede ver en el siguiente esquema:

Figura 13: esquema de procedimientos de detección de matrículas

Procedimientos tradicionales:

Los métodos más arraigados (que no los más empleados actualmente), están íntimamente ligados a los métodos tradicionales de procesamiento de señales. Esto se debe principalmente a que los primeros procedimientos automatizados de detección de

objetos en imágenes, se apoyaban en patrones geométricos marcados [45]. Las matrículas, precisamente por la normativa que las regula [17], tienen unas dimensiones fijas tanto de contorno como de los propios caracteres. Por lo tanto, una vez implementado un algoritmo para su detección, es asequible que funcione en una gran cantidad de escenarios ya que los contornos a detectar no van a variar. Evidentemente, esto no siempre se va a cumplir debido principalmente a la ubicación del sensor o del objeto a detectar.

La diferencia principal entre los procedimientos tradicionales y los clasificadores es que los primeros se centran en la identificación de un objeto sin entrar a categorizarlo. En cambio, los segundos permiten la detección del objeto porque lo identifican como tal, es decir, lo etiquetan. Además, los primeros van a aprovechar las características de una imagen digital para conseguir determinados efectos que resalten los objetos a localizar.

Una imagen digital es una señal a efectos computacionales y, por ende, puede ser tratada o modificada para conseguir determinados efectos sobre ella. Por ejemplo, reducir el tamaño de la imagen, destacar determinadas características o resaltar objetos. En este caso concreto, la diferencia entre este tipo de procedimientos y los que se van a explicar en el siguiente apartado (basados en inteligencia artificial) radica en la forma de ejecutarse.

Si bien parece que a día de hoy todas las investigaciones giran en torno al empleo de la inteligencia artificial, sobre todo en el tratamiento de imágenes, los procedimientos tradicionales se siguen usando en prácticamente cualquier herramienta de imagen. Primero, porque computacionalmente hablando suelen requerir menos recursos computacionales que aquellos basados en inteligencia artificial. Segundo, porque son relativamente fáciles de implementar (sobre todo gracias a librerías como OpenCV) [46] y tercero, porque son sencillos para operar en tiempo real, como sucede por ejemplo con OpenALPR [47].

Es importante destacar que estos procedimientos tienen una utilidad práctica directa cuando se emplea un sistema de visión, facilitando mucho la labor de un operador. Citando un caso práctico, en un sistema de videovigilancia es habitual que las cámaras muestren imágenes que, debido a malas condiciones de luminosidad, destellos o excesos de objetos hagan que el operador tenga dificultades para interpretar. El empleo de este tipo de herramientas permite resaltar aquello que puede interesar al operador. Por hacer una analogía, es como sucede en un visor térmico. Normalmente las imágenes

producidas por el sensor se proyectan en blanco y negro. Sin embargo, ofrecen otras posibilidades como destacar las imágenes con calor en un color rojo, que destaque sobre un fondo azul más frío [48] (como se aprecia en la Figura 14).

Por lo tanto, los métodos basados en detección de bordes, por color o por textura exploran el empleo de técnicas como la modificación del contraste, del histograma, la eliminación de ruido, la segmentación o las transformaciones morfológicas para buscar formas o patrones en las imágenes que se puedan corresponder con matrículas.

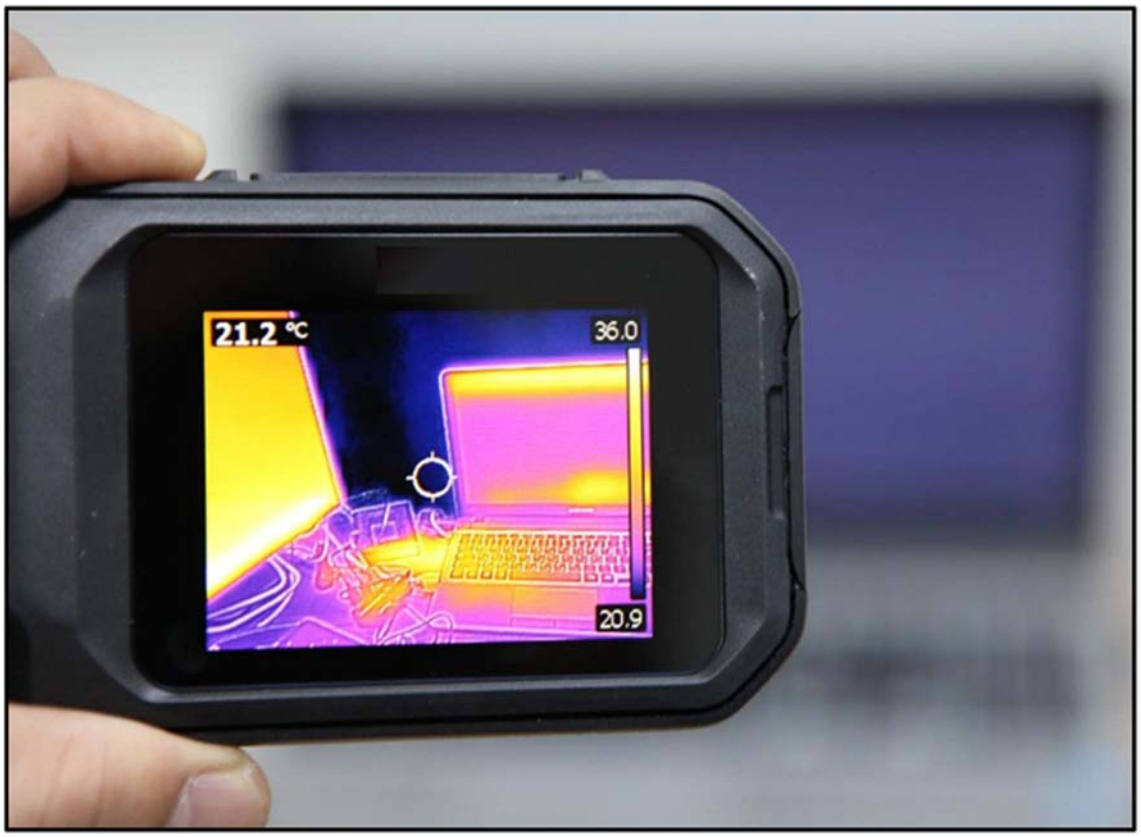


Figura 14: ejemplo de una imagen de un visor térmico

Procedimientos basados en inteligencia artificial

Estos procedimientos, como su propio nombre indica, se basan en algoritmos de inteligencia artificial para detectar la ROI que contiene la matrícula a identificar. Son principalmente clasificadores, es decir, algoritmos que detectan un objeto y lo etiquetan. Los métodos estadísticos como las *support vector machine* (SVM) [49], el algoritmo CAMshift [50] o adaboost [51] o las cascadas de HAAR [52] que se basan en la teoría de la probabilidad, permiten obtener inferencias, predicciones y conclusiones a partir de

datos observados, considerando la incertidumbre y la variabilidad inherente a estos datos.

Sin embargo, la aparición de algoritmos de *deep learning* con capacidad de detectar y clasificar objetos, sobre los basados en redes neuronales convolucionales (que se explicarán más adelante) como es el caso de YOLO [53], han permitido la proliferación masiva de modelos basados en estos procedimientos.

Sus diferentes versiones y evoluciones (especialmente las "Tiny") han reducido el tiempo de inferencia y los costes computacionales, pudiendo operar en tiempo real. De hecho, gracias a estos clasificadores, se está generalizando su uso para pre detectar vehículos en imágenes, reduciendo la cantidad de información procesada.

Pre – procesamiento

Las técnicas de pre – procesamiento tienen como finalidad preparar los caracteres situados en el interior de la ROI previamente detectada para ayudar en la labor de su reconocimiento posterior. Estrictamente hablando no es un paso obligatorio, aunque es muy recomendable dependiendo del tipo de herramienta utilizada.

Al igual que sucede con los procedimientos tradicionales de detección de matrículas, técnicas como la umbralización, la segmentación, la modificación del contraste o las transformaciones geométricas van a ayudar profundamente a resaltar los caracteres del fondo. Aquí juega un papel muy importante el hecho de que la normativa obligue a que exista un contraste natural entre los caracteres y el fondo, de tal manera que muchas de estas técnicas persiguen más bien eliminar posible ruido o información alterada por el paso previo, que reforzar un contraste profundo entre el fondo y los caracteres.

Por lo tanto, parece que la mayoría de los esfuerzos en este apartado se van a incardinar en buscar la orientación más adecuada de la región correspondiente a la matrícula, sobre todo para facilitar el proceso de la lectura de caracteres. Dependiendo del procedimiento posteriormente empleado, puede suceder que precisamente no se logren los resultados perseguidos precisamente por este detalle.

En ocasiones, será necesario realizar fuertes transformaciones geométricas para que nuestra imagen quede lo más horizontal posible. Esto dependerá del método de reconocimiento de caracteres. Aquí es donde de nuevo el aprendizaje profundo está ofreciendo nuevas posibilidades con las "redes de transformación espacial" [54]. Por

ejemplo, WPOD-net [55] emplea la transformación espacial y convierte las imágenes a un plano horizontal.

Lectura de caracteres

Este proceso es el definitivo y el que realmente ejecuta el reconocimiento de los caracteres de la matrícula. Una vez se ha extraído la ROI correspondiente y se ha preparado para facilitar este paso, se realiza un reconocimiento óptico de los caracteres, también llamado OCR (*optical character recognition*). El estudio de [43] muestra tres posibilidades diferentes: comparar todos los valores de píxel de los datos de la imagen en bruto directamente con las plantillas predefinidas (coincidencia de plantillas y patrones); utilizar diferentes técnicas de procesamiento de imágenes y aprendizaje automático para extraer las características antes de clasificar los segmentos (reconocimiento de caracteres mediante extractores de características); y de nuevo técnicas de aprendizaje profundo. En [43], podemos encontrar más información sobre los dos primeros métodos. En este trabajo se va a profundizar en las técnicas de aprendizaje profundo, muy utilizadas en la actualidad.

Tesseract [56] se ha convertido en el buque insignia del OCR. Muchos autores [57-60] han desarrollado varios métodos en los que Tesseract realiza esta función. Esta red fue diseñada para el reconocimiento de caracteres escritos. Es muy fácil de implementar, no tiene mucho coste computacional y funciona bien en entornos controlados, especialmente con textos horizontales con buena segmentación. Sin embargo, presenta varios problemas cuando el texto no está correctamente alineado, o cuando los caracteres están un poco borrosos. Este problema empeora en aplicaciones en tiempo real con circunstancias cambiantes. Por ejemplo, en [61], los resultados arrojan una precisión del 91%. En tiempo real, este resultado desciende al 75% (estos valores sólo se refieren al reconocimiento de caracteres).

Esta solución puede funcionar para modelos sencillos con pocos recursos. Sin embargo, ahora mismo es posible obtener mejores resultados (si tenemos acceso a unidades de procesamiento tipo GPU) con CNNs específicamente entrenadas para realizar reconocimiento de caracteres, como LPRnet [62] (que obtiene un 95% de precisión en matrículas chinas) u OCR-net [55], que obtiene casi un 94% en matrículas europeas (OCR-net es una interpretación de YOLOv2 [63], entrenada para identificar caracteres en imágenes).

2.1.1.2.- Arquitecturas de una sola fase

Estas aproximaciones utilizan una única red neuronal profunda entrenada para la detección, localización y reconocimiento de matrículas en un solo paso. El reconocimiento de matrículas puede considerarse como un caso específico en la detección de objetos. Al igual que los detectores de objetos de una sola etapa, estos modelos pueden aprovechar la alta correlación entre la detección y el reconocimiento de matrículas. Esto permite a los modelos compartir parámetros, reduciendo la cantidad con respecto a los que requiere un modelo típico de dos etapas (detección y reconocimiento por separado). Como resultado, pueden ser más rápidos y eficientes que los métodos de dos etapas comparables [64,65]. Detrás de esto, la idea es el uso de algoritmos de *deep learning* (normalmente redes neuronales convolucionales), como VGG16 [66] o EfficientNet [67], que han sido entrenados para realizar la identificación de caracteres directamente a partir de la imagen.

En este caso, es posible incluir en este grupo a [55] o [62]. De hecho, NVIDIA ofrece LPRnet como una dependencia incluida en su *toolkit* (sólo entrenada para matrículas chinas e indias). Hoy en día, una de las tendencias actuales es crear o adaptar CNNs para extraer directamente los caracteres de las matrículas sin realizar ningún paso previo (o al menos muy rápido). El entrenamiento adecuado de estos sistemas puede permitir la creación de soluciones robustas que se vean menos afectadas por las condiciones cambiantes del entorno.

2.1.1.3.- Soluciones actuales

En esta sección se ha pretendido recopilar distintos enfoques recientes que ayudarán a comprender las tendencias actuales en el reconocimiento de matrículas. Se pueden establecer dos grandes grupos. Los primeros, aquellos desarrollos basados en CNN multietapa, readaptando redes bien conocidas como YOLO (en diferentes versiones) que han sido entrenadas con conjuntos de datos específicos, y métodos de entrenamiento como el aprendizaje por transferencia o *transfer learning* [68-71]. Por ejemplo [68] muestra un método en dos fases punto a punto para caracteres persas, donde el reconocimiento de matrículas es el paso final justo después de la detección de coches y la detección de matrículas, con una precisión del 99,37% en el reconocimiento de caracteres. Otro ejemplo es [69], que es un método de una sola etapa basado también en CNNs para caracteres con diversos tipos de fuentes, con una precisión del 98,13% en varios tipos de vehículos (como coches o motos).

Por otro lado, existe otro enfoque muy interesante que consiste en desplegar estos sistemas para dispositivos de bajos recursos en tiempo real, emulando entornos operativos reales. Por ejemplo, en [72], los autores ejecutan su herramienta en un sistema basado en CPU con 8 GB de RAM, y obtienen una precisión del 66,1% con MobileNet SSDv2 [73] con una tasa de 27,2 FPS; o en [74], con métricas de detección del 90% y una tasa de reconocimiento del 98,73% con sólo Raspberry Pi3B+ como soporte hardware. Yendo más allá, podemos encontrar sistemas basados en Android como [75] o [76], que están diseñados para ser implementados en teléfonos móviles y operar en tiempo real. Estos procedimientos y sus resultados se muestran en la siguiente tabla.

Modelo	Caracteres reconocidos	Exactitud	Requisitos de hardware
Pirgazi et al. [68]	Persas	99.37%	Alto
Kaur et al. [69]	Varios	98.13%	Alto
Hossain et al. [70]	Bangladesh	96.31%	Alto
Zandi et al. [71]	Iraníes	98.86%	Alto
Ashrafee et al. [72]	Bangladesh	66.1%	Bajo
Padmasiri et al. [74]	Chinos (CCPD Dataset)	98.73%	Bajo

Tabla 1: ejemplos de soluciones actuales de reconocimiento de matrículas

2.1.1.4.- *Datasets*

El nombre de *dataset* (o conjunto de datos) hace referencia a los datos de entrada al sistema necesario para realizar el entrenamiento de una red neuronal, aunque este concepto se va a detallar más en profundidad en el Capítulo IV. Uno de los principales problemas para desarrollar y mejorar los ALPR es tener acceso a los *datasets*. La legislación desempeña un papel importante, ya que hace muy difícil compartir conjuntos de datos de matrículas, especialmente en Europa [7777]. De hecho, es más fácil acceder a conjuntos de datos de matrículas asiáticas (como chinas, pakistaníes o indias) (véase la Tabla 2). Por ejemplo, podemos encontrar GAP-LP [78] (turco), con 9.175 matrículas, CCPD [65] (chino, 250.000 matrículas), o UFP-ALPR [79] (chino, 4.500 imágenes).

El mayor conjunto de datos europeo de libre acceso es OpenALPR-EU [80], con sólo 108 imágenes. El conjunto de datos europeo más completo que se ha encontrado es TLPD (THI license plate dataset) [81], con 18.000 matrículas europeas etiquetadas. El problema es que este conjunto de datos no es de acceso público, además que el trabajo que lo desarrolla únicamente se ha centrado en etiquetar los límites de la matrícula, y no los caracteres en su interior.

Si bien a nivel internacional prácticamente todos los vehículos de los países del mundo tienen la obligación de emplear matrículas, no hay una configuración universal. De hecho, incluso en la propia Unión Europea, donde encontramos una mayor uniformidad, existen matrículas de diferentes tamaños y colores (por ejemplo, las matrículas de un ciclomotor).

Figura 15: comparativa entre una matrícula de ciclomotor y una de vehículo ordinaria

Estas variaciones provocan cierta dificultad a la hora de establecer herramientas con capacidad para generalizar las características extraídas, algo fundamental, tal y como se explicará en el Capítulo IV.

Para solucionar esto, es muy importante realizar el entrenamiento con un conjunto de datos específico que cumpla los siguientes requisitos:

- Condiciones de iluminación variables: día, noche, lluvia, niebla...
- Diferentes ángulos y puntos de vista
- Varias matrículas por fotograma
- Diferentes escenarios: calle, carretera, caminos...
- Imágenes con diferentes resoluciones y dimensiones
- Diferentes captadores

El *dataset* debe abarcar el mayor número posible de condiciones reales, y no sólo las ideales. Dicho esto, las matrículas deben ser reconocibles en las diferentes imágenes, y tienen que ser lo suficientemente grandes (esta valoración dependerá de los resultados obtenidos durante el entrenamiento de la red).

Nombre	Región	N.º de Imágenes	Vehículos
GAP-LP [76]	Turquía	9175	Turismos
CCPD [65]	China	250,000	Turismos
UFP-ALPR [79]	China	4500	Varios (turismos, camiones, etc.)
OpenALPR-EU [80]	Europa	108	Varios
TLPD [81]	Europa	18,000	Varios

Tabla 2: principales *datasets* de matrículas

2.1.2.- Re – identificación de vehículos

En el apartado 1.4.2 se hizo una primera aproximación, definiendo la re – identificación de objetos (*Object* ReID) como la capacidad de reconocer un objeto específico que ha sido previamente identificado, o como "la capacidad de asociar un objeto concreto en diferentes observaciones" [82]. Es un concepto que habitualmente está asociado a personas, como la posibilidad de "asociar personas a través de diferentes planos de cámara en distintos lugares y momentos" [83]. La re – identificación se ha utilizado tradicionalmente en rostros más que en coches. Facenet [84], publicado en 2015, es un buen ejemplo. De hecho, tras su desarrollo, muchas empresas empezaron a ofrecer soluciones comerciales de seguridad relacionadas con el reconocimiento facial, como sucede en la identificación de pasaportes.

Aunque los vehículos a priori tienen muchos más patrones característicos que las caras, la re – identificación de vehículos presenta un desafío mayor que la de personas [85]. Por ello, aunque al igual que en el apartado anterior, existen tanto métodos tradicionales como métodos basados en el aprendizaje profundo, la proliferación de estos últimos ha desplazado a los tradicionales. Se puede considerar que es más eficiente, debido a que estos métodos no se centran únicamente en los detalles específicos de los coches, sino también en atributos distribuidos en toda la imagen, seleccionando varias características en el mismo proceso [82,85]. Por ello, esta sección se va a centrar principalmente en describir los procedimientos de *deep learning* más importantes para la re – identificación de vehículos. Para ello, se ha seguido como guía diferentes artículos como [86] o [87].

2.1.2.1.- Métodos basados en características locales (*local features)*

La idea general de estos procedimientos es obtener un mapa de características a la salida de la red, basado en una serie de detalles específicos de la imagen. Estos métodos persiguen captar características visuales únicas en áreas locales y mejorar la percepción,

lo que ayuda en gran medida a distinguir entre distintos vehículos y aumenta la precisión de la re – identificación de vehículos. Mediante CNN, se persigue realizar un reconocimiento de patrones concretos de los vehículos como el emblema de la marca, los espejos retrovisores o la forma de los faros.

Además, muchos métodos intentan combinar características locales con características globales para mejorar la precisión, como sucede en [88]. En este caso, un módulo se centra en los detalles de las luces delanteras y traseras, las ventanillas delanteras y traseras y la marca del vehículo.

Si bien a priori puede ofrecer ciertas ventajas para encontrar similitudes y diferencias entre determinados vehículos, la realidad es que este tipo de herramientas conllevan un coste computacional muy grande, asociado a la profundidad de las redes que son necesarias.

2.1.2.2.- Métodos basados en aprendizaje representativo (*representation learning)*

Uno de los principales problemas de los métodos anteriormente descritos es que no son robustos frente a las alteraciones entre la imagen original y el resto de imágenes correlativas. En el caso de que las imágenes correlativas sean diferentes de la original (lo que es bastante habitual, bien por un cambio en el ángulo de la cámara, bien por haber sido obtenidas por una cámara diferente), la precisión del sistema disminuye significativamente, lo que se traduce en una pérdida de eficacia en escenarios reales.

El aprendizaje de representación se centra en reajustar los pesos de la red. Su objetivo es hacer que la propia red sea capaz de centrarse en los puntos de interés característicos, en lugar de que seamos nosotros los que dirijamos la red a través de las diferentes capas convolucionales. La idea es que al final del entrenamiento, sea capaz de detectar automáticamente aquellos patrones característicos que identifican específicamente la imagen de cada vehículo, pero sin tener un foco predefinido.

Las representaciones se forman por la composición de múltiples transformaciones no lineales de los datos de entrada [89]. En la actualidad, se llevan a cabo principalmente dos tipos de trabajos para las tareas de re – identificación. Uno es considerarlo como un problema de clasificación, lo cual ha día de hoy es muy complicado si se pretende generalizar de forma masiva, ya que únicamente va a reconocer los modelos de vehículos para los que la red haya sido entrenada. La otra posibilidad es hacer un

entrenamiento con un *dataset* que trabaje con parejas de vehículos, de tal manera que la red aprenda los patrones de similitud.

Actualmente, esta aproximación se considera como poco práctica, aunque muy útil cuando no se pretende implementar de manera genérica ya que puede resultar muy robusta.

2.1.2.3.- Métodos basados en aprendizaje mediante métricas (*metric learning*)

Este método se basa en la comparativa de las métricas obtenidas por dos o más conjuntos de imágenes (sus vectores de características) a la salida de una red neuronal convolucional [90], normalmente apoyándose en la distancia euclídea entre ambos vectores. Dado que es necesaria una comparación entre las métricas, está estrechamente vinculado al uso de redes siamesas [91]. Este tipo de procedimiento ha sido ampliamente utilizado, especialmente en la re – identificación de personas.

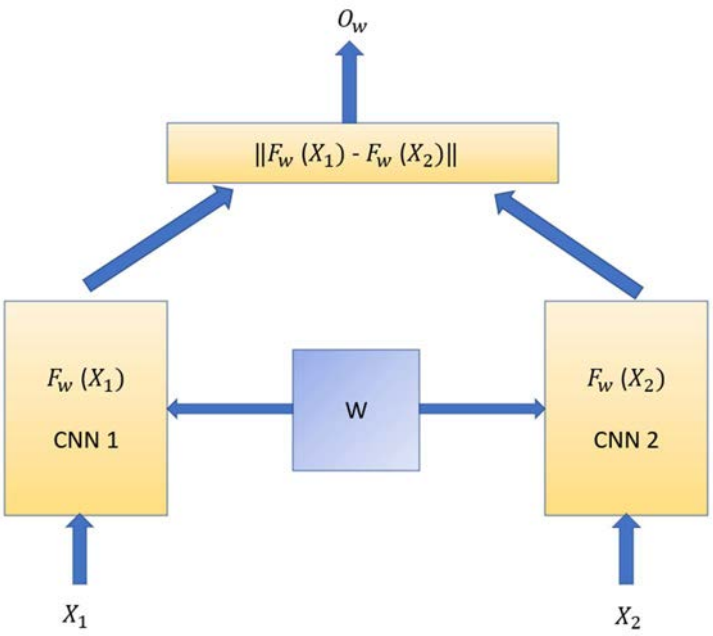


Figura 16: esquema de una red siamesa

Dentro de este grupo de métodos, podemos destacar principalmente dos subtipos:

El primero se basa en la utilización de una función denominada "*contrastive loss*", que está altamente ligada a las redes siamesas. En la capa de salida, se obtienen dos vectores de características. Esta función calcula la pérdida de la red con el valor absoluto de la distancia euclídea entre las dos matrices de cada una de las imágenes de entrada. Así, si

ambas imágenes son similares, el valor de la función de pérdida será similar a 0, y si son diferentes, será cercano a 1.

Se pueden encontrar varios ejemplos del uso de este procedimiento, como [92] o [93].

El otro gran subtipo es el que utiliza *triplet-loss* como hace Facenet [84]. En este caso, en lugar de operar con dos redes, utiliza tres redes convolucionales simultáneamente con tres imágenes de entrada. Una imagen sería la identificada, que se llama "ancla"; una segunda sería prácticamente igual al ancla, que sería la imagen positiva, y una tercera sería la diferente, llamada negativa. Por lo tanto, habría dos distancias euclídeas diferentes, una entre el ancla y la imagen positiva, y otra entre el ancla y la imagen negativa. De este modo, la función de pérdida haría una comparación entre ambas distancias euclídeas, y el aprendizaje de la red ajustaría los pesos para conseguir acortar la distancia entre el ancla y la imagen positiva.

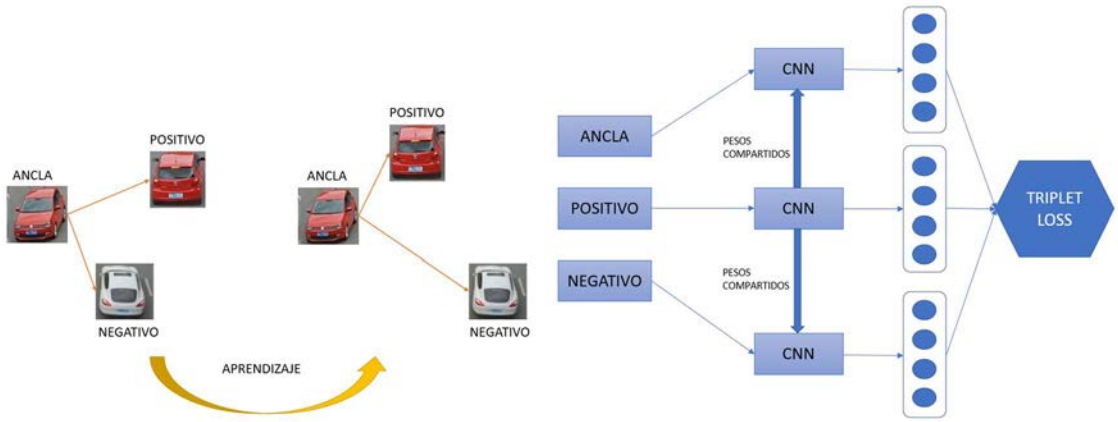


Figura 17: representación del funcionamiento de *triplet loss*

Ejemplos del empleo de este algoritmo podrían ser [94] o [95].

2.1.2.4.- Métodos basados en aprendizaje no supervisado

En este caso, la idea del sistema es utilizar un tipo de CNN de entrenamiento no supervisado, denominadas GAN (*Generative Adversarial Networks*) [96], para realizar tareas de re – identificación. Estas redes tienen dos módulos principales: un generador y un discriminador. El generador crea nuevas imágenes virtuales a partir de imágenes reales, introduciendo patrones pseudoaleatorios que, en algunos casos, son inapreciables para el ojo humano. El discriminador se encarga de discernir si tras esta modificación posterior, la imagen original y la obtenida a la salida del discriminador son similares o

no. Por lo tanto, no es necesario disponer de un conjunto de datos que contenga imágenes etiquetadas.

Este tipo de red se creó inicialmente para protegerse de los ataques esteganográficos (en los que se incluye ruido digital en una imagen, de forma que, si se analiza visualmente, puede ser igual que la original, pero si se analiza su codificación digital, es completamente diferente). Es un tipo de aprendizaje no supervisado (a diferencia del resto de sistemas, que se consideran supervisados), ya que la red no sabe si los pares de imágenes analizados por el discriminador son similares o no entre sí; es la red la que lo hace de forma autónoma. De hecho, este es el objetivo del proceso de aprendizaje.

La idea por tanto en la re – identificación de vehículos, es que el generador induzca modificaciones geométricas para que la red sea capaz de discriminar entre imágenes similares y disímiles.

Figura 18: esquema de una red GAN

Este tipo de redes son especialmente útiles en situaciones en las que hay que identificar el vehículo bajo distintos planos o con distintas cámaras, con lo cual tienen una aplicación práctica muy directa y solventan problemas de procedimientos basados en métodos locales o de aprendizaje representativo. Sin embargo, son muy difíciles de entrenar, ya que son modelos bastante inestables. Ejemplos del empleo de estas técnicas podrían ser [97] y [98].

2.1.2.5.- Métodos basados en mecanismos de atención

Por último, la técnica más innovadora es la aplicación de mecanismos de atención [99] a la salida de una red. Estos mecanismos, considerados como una evolución de las redes neuronales recurrentes (o RNN), se concibieron inicialmente para su uso en el procesamiento del lenguaje natural. La idea general que subyace a las RNN en este campo es que cada palabra se codifica con respecto a las anteriores. Esto presenta problemas con la memoria a largo plazo, debido al arrastre del valor de todos los

gradientes. Los mecanismos de atención resuelven este problema porque pueden utilizar todos los recursos disponibles para operar (el único límite), a través de una secuencia codificador-decodificador (transformadores o *transformers*).

La aplicación de mecanismos de atención en imágenes emula el concepto de RNN en el procesamiento del lenguaje natural. Así, diferentes regiones de la imagen se codifican en relación con esas mismas regiones en etapas anteriores. De esa forma, el sistema otorga de forma automática la importancia a las distintas partes de las imágenes.

La entrada en un transformador es un vector correspondiente a un fragmento de texto (codificador). Si los utilizamos en imágenes, este codificador es un mapa de características que se extrae de la salida de otra red (por ejemplo, una CNN). Como resultado de la aplicación de un mecanismo de atención, la imagen se divide primero en n partes, y calcula las representaciones de cada parte específica de la imagen. Cuando la red genera un detalle de esa imagen, el mecanismo de atención se centra en las partes relevantes, de modo que el descodificador sólo utiliza partes específicas para codificarla en lugar de la imagen completa, obteniendo así los "mapas de atención" o *"attention maps"*.

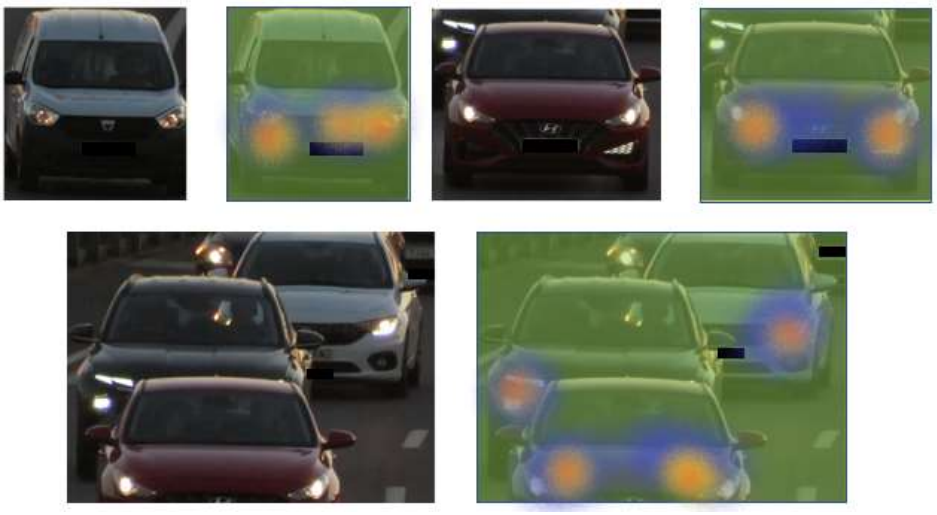


Figura 19: representación gráfica de mapas de atención

Los mecanismos de atención facilitan la re – identificación de una manera parecida a la que lo haría el ser humano, centrando la atención en determinadas regiones características de la imagen.

En la imagen superior se puede apreciar de forma gráfica cómo se obtendrían mapas de atención sobre imágenes de vehículos. Por ejemplo, focalizándose en zonas como

destellos característicos en los parabrisas, pegatinas o pinturas específicas. Sin embargo, al centrarse en detalles llamativos, la capacidad de re – identificación se difumina cuando los detalles son más sutiles, el fondo de la imagen tiene un color parecido al del vehículo o cuando no existe una gran multitud de imágenes correctamente etiquetadas para el entrenamiento. Estos procedimientos se consideran los más actuales a día de hoy dentro del estado del arte en este campo. De hecho, la solución elegida para este ámbito utiliza entre otros elementos mecanismos de atención [100-102].

2.1.2.6.- Soluciones actuales

Al igual que se hizo en el apartado 2.1.1.3, también se van recopilar algunas de las soluciones actuales para la re - identificación de vehículos que se han publicado recientemente. En este caso, los métodos propuestos están encuadrados principalmente en la aplicación de mecanismos de atención y de aprendizaje mediante métricas. [103] es un ejemplo del primer grupo, donde los autores proponen una red de atención adaptativa en tres ramas para la re - identificación de vehículos

En cambio, los trabajos de [104,105] son ejemplos de métodos de aprendizaje mediante métricas. El primero aporta un novedoso punto de vista mediante una doble aplicación de la función *triplet loss* para resolver la re – identificación, cuando lo normal es emplearse únicamente una vez en la etapa final de la red. En este caso concreto se considera una *triplet loss* para las diferencias entre objetos de distintas clases (*intra – view)* y otras similitudes entre objetos de la misma clase (*inter – view*) como etapa final de la red. El segundo trabajo, propone la creación de una nueva función de pérdida que se denomina *support neighbours loss* o SNN *loss*. En este caso se trata de una aplicación del algoritmo KNN (*k nearest neighbours*) a una función de pérdida para conseguir y la re – identificación de personas y vehículos. El estudio de [106] es otro enfoque muy interesante y novedoso, derivado de una combinación de armonizar diferentes perspectivas de imágenes en 3D.

Modelo	**Características**	**Métricas**		
		mAP	**r@ank1**	**r@ank5**
GRMF [103]	*Multi-granularity feature learning*	0.882	0.957	0.991
VARID [104]	*Inter- and intra-view triplet loss*	0.793	0.962	0.992
SN++ [105]	*Support neighbours' loss*	0.757	0.951	0.981
Meng et al. [106]	*3D viewpoint alignment*	0.832	0.987	0.992

Tabla 3: ejemplos de soluciones actuales de re - identificación de vehículos

2.1.2.7.- *Datasets*

En un problema de clasificación, los *dataset* tienen un gran impacto en las métricas que se puedan obtener como resultado. En el caso concreto de la re – identificación, la mayoría de las investigaciones reseñadas en el apartado anterior coinciden en la importancia de disponer de un *dataset* adecuado y específico, siendo el principal requisito, que contengan una gran muestra de imágenes de automóviles bajo puntos de vista muy diferentes.

Haciendo un recorrido histórico, uno de los primeros *dataset* fue publicado en 2013 por la Universidad de Stanford [107]. Su propósito original era poder llevar a cabo una clasificación de objetos tridimensionales muy similares entre sí. Sin embargo, sus usos evolucionaron, sirviendo como punto de partida para varios estudios centrados en el desarrollo de las CNN y evaluación de diferentes clasificadores.

Figura 20: ejemplos de imágenes pertenecientes a *Stanford Cars Dataset* [107]

Posteriormente aparecieron cuatro *datasets* con la idea de hacer posible la re – identificación en vehículos al igual que se hacía con personas: VehicleID [108] , VeRi [109] , VeRi-776 [110] y VeRi-Wild [111] . Por ejemplo, VeRi surge debido a la necesidad de llevar a cabo una vigilancia y rastreo de vehículos en entornos urbanos. De acuerdo con sus autores, el objetivo es crear un ambiente de trabajo que se asemeje lo más posible a un escenario real, en el cual no se dependa únicamente de un catálogo con una sola imagen por marca y modelo de vehículo, sino que se cuenten con varias imágenes del mismo automóvil. Para lograr esto, VeRi incluye imágenes de 619 vehículos distintos, capturadas por 20 cámaras diferentes desde diversos ángulos. Por lo tanto, este conjunto de datos representa un punto de partida crucial para el desarrollo de sistemas de re – identificación de vehículos.

La siguiente evolución a VeRi fue "VeRi-776", un *dataset* que amplía el volumen total de imágenes en un 20%, incluyendo hasta 776 vehículos distintos distribuidos en 50.000 imágenes. El etiquetado además incorpora datos de una relación espacio-temporal combinando la ubicación del captador, el instante en el que se produce, la dirección del vehículo capturado y la trayectoria seguida.

Figura 21: ejemplos de imágenes pertenecientes a VeRi [109] y a VeriWild [111]

La última evolución de estos conjuntos de datos de entrenamiento obtenidos mediante capturas de vídeo en tiempo real es VeRi-Wild. En total, este *dataset* contiene 416.314 imágenes de 40.671 vehículos distintos, con diferentes oclusiones, trayectorias y perspectivas, generadas por 174 videocámaras funcionando 24 horas al día durante un mes completo.

Estos cuatro *dataset* no sólo tienen una gran influencia en la presente investigación, si no que constituyen pilares sobre los que se han desarrollado y se siguen desarrollando a día de hoy gran cantidad de herramientas de detección, clasificación y re – identificación de vehículos, sobre todo cuando se pretende operar en tiempo real.

Dicho esto, existen otros *dataset* (véase Tabla 4) que también tienen una gran influencia en estudios de re - identificación, como “PKU Vehicle” [112] , con decenas de millones de imágenes de vehículos capturadas también por cámaras de vigilancia ubicadas en diferentes ciudades chinas con distintas ubicaciones (autopistas, calles, intersecciones), condiciones climáticas (soleado, lluvioso, con niebla), condiciones de iluminaciones (día y noche), ángulos (frontal, lateral, trasero) y resoluciones (480 píxeles, 640 píxeles o 2K). Otros ejemplos son Vehicle-1M [113], también chino, con 936,051 imágenes de 400 modelos distintos con un total de 55.527 vehículos, o CityFlow [114], que es el primer *dataset* del mundo [115] que contiene seguimiento y re - identificación de automóviles entre cámaras para llevar a cabo labores de seguimiento. Tiene 3,25 horas

de vigilancia de 40 cámaras diferentes en 10 intersecciones de una ciudad estadounidense, tanto en áreas residenciales como en autopistas. El conjunto de datos incluye 229.680 vehículos etiquetados de 666 vehículos diferentes. Cada automóvil ha pasado por al menos dos cámaras y el conjunto de datos proporciona video en bruto, distribución de cámaras y análisis de múltiples vistas.

Como colofón a este apartado, se ha querido reseñar la aparición de un nuevo *dataset* que aporta un punto de vista muy interesante, íntimamente relacionado con la proliferación masiva de vehículos aéreos no tripulados (UAV) y con un enfoque claramente relacionado con la seguridad ciudadana. Se trata de VeRi-UAV [116], un *dataset* focalizado en la re – identificación de vehículos mediante el empleo de imágenes principalmente cenitales adquiridas desde un UAV.

Nombre	N.º de Imágenes	N.º de Vehículos	Características
Stanford Cars Dataset [107]	16.185	196	Fotografías de vehículos
VehicleID [108]	221.763	250	CCTV
VeRi [109]	40.000	619	CCTV
VeRi-776 [110]	49.360	776	CCTV
VeRi-Wild [111]	416,314	40.671	CCTV
PKU Vehicle [112]	10.000.000		CCTV
Vehicle 1M [113]	936.051	26.267	CCTV
CityFlow [114]	229.680	666	CCTV
VeRi-UAV [116]	17.515	454	Imágenes aéreas

Tabla 4: principales *dataset* para la re - identificación de vehículos

2.3.- Conclusiones derivadas del estado del arte

Tras haber recopilado diferentes técnicas y posibilidades de abordar los objetivos perseguidos, en este apartado se pretende hacer un breve resumen de capacidades e inconvenientes, además de reflejar unas conclusiones parciales obtenidas tras analizar la información anteriormente plasmada.

Primero, se va a realizar una comparativa entre los dos grandes grupos de procedimientos de identificación de vehículos reseñados a nivel general, teniendo en cuenta diversos elementos. Como consecuencia de dicha comparativa, aparecen una serie de ventajas en inconvenientes, mayoritariamente prácticas, que aparecen recopiladas en la Tabla 5.

En cuanto a la tendencia general a la hora de abordar problemas relacionados con el procesamiento de imágenes, se puede manifestar que actualmente existe una corriente

muy dominante en materia del empleo de algoritmos de inteligencia artificial, y más concretamente de redes neuronales de varios tipos, pudiéndose organizar en tres grandes grupos:

- Mejoras y evoluciones basadas en realizar pequeñas combinaciones y adaptaciones de modelos anteriores, con la finalidad de mejorar los resultados,
- Investigaciones centradas en la creación y mejora de datasets muy específicos (sobre todo en clasificadores) para mejorar las capacidades de una herramienta ya existente. Es habitual que esto también se realice en combinación de un ajuste de los hiperparámetros vinculados a la red, de tal manera que también se optimicen los resultados obtenidos como consecuencia del entrenamiento
- Por último, creación de soluciones globales que utilizan todo el repertorio disponible para conseguir esa mejora de resultados (es decir, una combinación de los dos tipos de herramientas)

La investigación recogida en el presente documento tiene una aproximación más cercana a las del segundo punto, si bien es verdad que además de aportar mejoras en cuanto a resultados y afrontar la creación de *datasets*, ha supuesto un abordaje novedoso a la hora de realizar una combinación de redes complementarias entre sí, pero que son muy distintas en su concepción inicial.

Procedimientos	Ventajas	Inconvenientes
Basados en dispositivos	Sencillos de implementar (**IMPORTANTE**: **si se tiene acceso al vehículo**)	Más invasivos
	Requieren poca infraestructura	Menor autonomía (según la herramienta empleada)
	Precio contenido	
	Posibilidad de seguimiento en tiempo real	
	Elevada precisión	
Basados en herramientas visuales	Mayor universalidad	Complejidad
	Más escenarios de uso	Costes
	Posibilidad de cubrir varios objetivos de manera simultánea	Diseño e implementación
	Muy poco invasivos	

Tabla 5: comparativa de herramientas reseñadas en el estado del arte

SISTEMAS DE VIDEOVIGILANCIA III

A lo largo de la introducción se han establecido criterios y motivaciones que justifican la presente investigación. Sin embargo, precisar el área específica de trabajo ha requerido tener previamente un conocimiento profundo de cuál es, en última instancia, la herramienta sobre la que se van a implementar los modelos conseguidos: el sistema de videovigilancia. Por lo tanto, es importante también conocer en detalle esta herramienta, sus componentes y sobre todo aquellas cuestiones técnicas y operativas que lo rodean.

Un sistema de videovigilancia se puede considerar como un tipo de sistema de visión artificial, cuya finalidad es establecer un control visual sobre una determinada zona sin necesidad de recurrir a la presencia física de una persona. Esto acarrea una serie factores condicionantes y a su vez provoca que estos sistemas tengan unas características específicas. Por lo tanto, el objetivo de este capítulo es precisamente desgranar cuáles son esos condicionantes y características que afectan a su concepción bajo un prisma práctico, y que han servido para motivar, en última instancia, las decisiones tomadas a lo largo de la presente investigación.

Para ello, este capítulo se va a dividir en tres bloques. El primero pretende mostrar qué es un sistema de visión y cómo los diferentes elementos que lo componen tienen una influencia directa en los resultados finales. El segundo bloque analiza cuáles son los factores y entornos operativos que se deben de considerar y analizar a la hora de diseñar el sistema concreto. Por último, el tercer bloque muestra cómo esos factores confluyen entre sí y, por lo tanto, qué decisiones se deben tomar para lograr la solución más óptima según los requisitos y las necesidades.

3.1.- Sistema de visión artificial e influencia de sus componentes

Un sistema de visión artificial se puede definir como un conjunto de elementos destinados a captar, tratar, almacenar y transmitir información obtenida en forma de fotogramas [117] y consta de los siguientes elementos [118]:

- Iluminación
- Óptica
- Sensor de captación

- Sistema de procesamiento de señal
- Almacenamiento
- Transmisión

Un sistema de visión se debe entender como una cadena, una sucesión de elementos que, aunque no lleguen necesariamente a interactuar entre sí, el comportamiento de cada elemento tiene una influencia directa en el siguiente.

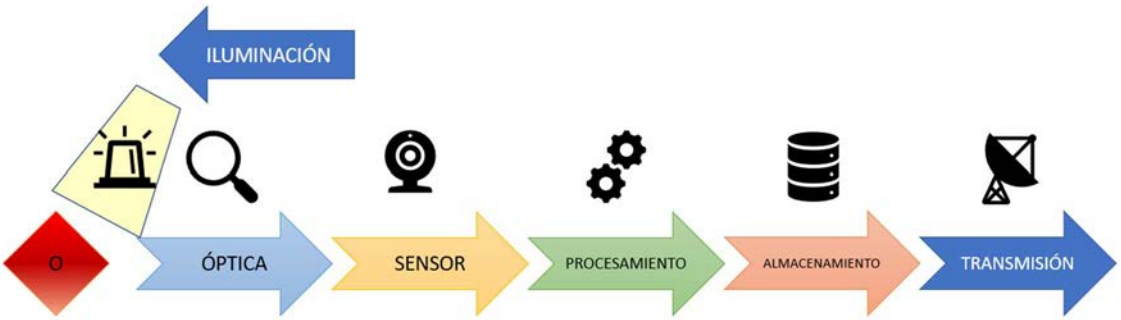


Figura 22: esquema de un sistema de visión

De todos los elementos mencionados, hay dos que necesariamente van a estar presentes: la óptica y el sensor de captación. El sensor, porque es el responsable de generar la imagen (sea analógica o digital) mediante la codificación de una señal luminosa, y la óptica, porque concentra dicha señal luminosa. Este conjunto se suele llamar captador.

La presencia del resto de elementos dependerá de las necesidades que se deban satisfacer y/o de las capacidades que se quieran ofrecer. De manera muy simplificada, si por ejemplo el sistema se encuentra ubicado en entornos con suficiente luz residual, no parece necesario incorporar elementos de iluminación adicionales.

A continuación, se va a desgranar la influencia de los diferentes elementos, con las posibilidades que pueden llegar a ofrecer y las dificultades que a veces se deben subsanar.

3.1.1.- Iluminación, óptica y sensor

La imagen contiene la información de interés generada al emplear un sistema de visión. Por lo que los elementos que tienen una incidencia directa en la creación de las imágenes también van a tener efectos en los resultados finales.

Como se ha explicado anteriormente, el sensor es el lugar donde se generan las imágenes a consecuencia de la excitación de diversos componentes electrónicos al recibir los rayos de luz captados por la óptica. Por lo tanto, la óptica y el sensor (en combinación con la luz), son los dos elementos que siempre van a estar presentes en cualquier sistema de visión y que más pueden afectar al conjunto del sistema.

Las dos características más importantes en una imagen son el tamaño y la resolución. El tamaño de la imagen viene determinado por la resolución (número de píxeles totales del sensor) y por el tamaño del sensor. A priori, una imagen captada por un sensor más grande va a contener una mayor cantidad de información y, por lo tanto, va a permitir precisar mejor determinados detalles de las imágenes. Sin embargo, va a conllevar un mayor coste computacional (además de un coste económico) para poder procesar toda esa información, sobre todo cuando se trata de implementar herramientas que operen en tiempo real.

Otro aspecto importante es la nitidez. Una imagen se considera nítida cuando está "enfocada, parada y correctamente expuesta". El enfoque depende íntegramente de la óptica, que en función de sus características va a determinar la distancia a la que se pueden distinguir los objetos (zoom) y la profundidad de campo (el rango de distancia dentro de una escena en el que los objetos aparecen aceptablemente enfocados y nítidos [119]).

En fotografía, "parar una imagen" generalmente se refiere a la acción de capturar un objeto en movimiento de tal manera que aparece congelado en la fotografía (sin aberraciones ni bordes difuminados). Para ello, el obturador regula el tiempo que el sensor se encuentra expuesto (velocidad de obturación). Una mayor velocidad de obturación permite conseguir una mayor nitidez en movimientos más rápidos de los objetos, pero por el contrario, va a requerir de una mayor cantidad de luz para conseguir una correcta exposición.

Por último, la exposición es la cantidad de luz total captada por el sensor. Está condicionada por varios factores, entre los que se debe mencionar el tamaño del sensor, la velocidad de obturación y la apertura del diafragma de la lente. Es aquí donde los elementos de iluminación pueden jugar un papel fundamental, ya que van a conseguir una exposición correcta cuando las circunstancias operativas y ambientales no lo permitan de forma autónoma.

Teniendo en cuenta los objetivos establecidos en la introducción, el sistema desarrollado en su conjunto debe ser capaz de operar en diferentes ambientes y también en tiempo real, lo que implica elementos de diferentes tamaños (coches y matrículas), con diferentes orientaciones, movimiento y sobre todo condiciones de luz variables.

Para conseguir las mejores imágenes posibles en estas circunstancias, se deben emplear captadores que permitan:

- Una distancia de detección e identificación suficiente (recordar el concepto DORI explicado en el apartado 1.4.2)
- Conseguir en el plano la distancia hiperfocal (distancia a la que se maximiza la profundidad de campo [120])
- Un tamaño de imagen adecuado (que permita al operador al sistema de procesamiento analizar correctamente la información)
- Una correcta exposición

3.1.2.- Sistema de procesamiento de señal

La imagen digital generada en el sensor puede ser almacenada o transmitida sin editar, o se pueden realizar determinadas operaciones sobre ella. Cuando se emplea un visor térmico, las imágenes se representan en escalas cromáticas (normalmente en rojo los puntos de mayor calor, y en azul los de menos). El sistema de procesamiento de señal es el responsable de realizar este tipo de procesos.

Si bien tradicionalmente la función del sistema de procesamiento era únicamente crear las imágenes en base a la información capturada por el sensor, actualmente tienen capacidad de realizar muchísimos más procesos avanzados. Por ejemplo, aplicar configuraciones de detección de movimiento, formatos de compresión o utilizar funciones de analítica de vídeo.

La analítica de vídeo, también conocida como análisis de vídeo inteligente, es el proceso de analizar y extraer la información útil para el propósito del sistema de visión a partir de secuencias de vídeo mediante algoritmos y técnicas de procesamiento de imágenes [121]. Existen multitud de ejemplos de implementación de analítica de vídeo (como los mencionados en el párrafo anterior). Sin embargo, y tal y como se ha desgranado en el estado del arte, actualmente se apoya en gran medida en técnicas de inteligencia artificial y aprendizaje profundo, como las redes neuronales convolucionales (CNN).

3.1.3.- Almacenamiento

Los mecanismos de almacenamiento sirven para conservar la información generada por el sistema en su conjunto. El matiz de en su conjunto aparece porque por norma general, además de los fotogramas o vídeos propiamente dichos, es posible configurar el sistema para que almacene otros datos o únicamente determinados eventos (en lugar del flujo de información al completo).

Dependiendo de la configuración general, puede resultar un recurso limitante en cuanto a las capacidades disponibles. Primero, porque restringe la cantidad máxima de información que se puede almacenar. Y segundo, porque condiciona la velocidad de procesamiento de la señal. Esto es especialmente importante si se pretende almacenar vídeo.

Para hacer un uso eficiente del almacenamiento disponible se suele recurrir al sistema de procesamiento de la señal, grabando únicamente eventos (es decir, variaciones en los fotogramas). Con ello, se va a reducir notablemente la cantidad de información almacenada. Aunque otra posibilidad podría ser emplear herramientas de analítica de vídeo para almacenar solamente fotogramas de interés o la información asociada (como por ejemplo placas de matrícula).

3.1.4.- Transmisión

Los elementos de transmisión o enlace sirven para comunicar la información del sensor con el resto de componentes del sistema para su visualización, almacenamiento o procesar las señales. Aunque se trata de un criterio puramente operacional, normalmente se considera que existen elementos de transmisión cuando los componentes del sistema no están ubicados en un mismo espacio físico y el canal de transmisión se convierte en un factor limitante adicional. Por ejemplo, aunque el grabador esté conectado mediante cable RJ45 (cable Ethernet) al sensor, en el momento en que se pueden generar pérdidas por distancia, se considera que existe un elemento de transmisión (porque es un factor condicionante y/o limitante).

Al igual que sucede con los elementos de almacenamiento, los de transmisión no son estrictamente necesarios. Sin embargo, su uso es bastante habitual, sobre todo debido al empleo de mecanismos de analítica de vídeo y visionado remoto. Esto se debe a que normalmente las herramientas de procesamiento de señal actuales más avanzadas

requieren de un hardware de gran tamaño, coste y consumo, y por economía de medios suelen ser comunes para varios captadores.

Los elementos de transmisión se pueden dividir principalmente en dos categorías: cableados e inalámbricos. Los primeros, como su propio nombre indica, son aquellos en los que un cable físico conecta diferentes elementos del sistema, como el cableado BNC (para señal analógica) o el cableado Ethernet o RJ45 (imagen digital).

Los inalámbricos emplean diferentes tecnologías basadas en la propagación de señales electromagnéticas. Si bien predomina la transmisión basada en 3G/4G y/o WiFi, también destacan otros protocolos como la radiación en VHF (banda de 433 MHz o 868 MHz) o Bluetooth.

Las comunicaciones cableadas ofrecen un mayor ancho de banda, una mayor fiabilidad y un menor consumo, siendo las principales pérdidas de conexión aquellas causadas por un deterioro del cable. Además, en caso de empleo de cable RJ45, existen pares disponibles para proporcionar alimentación (lo que se denomina PoE) junto con la comunicación de datos (lo que simplifica notablemente la infraestructura completa). Como contrapartida, las distancias de transmisión se ven ampliamente reducidas, debido principalmente a la dificultad de disponer de una infraestructura de comunicaciones físicas.

En cambio, los segundos van ofrecer mayor versatilidad en cuanto a distancias de transmisión y necesidad de infraestructura, requiriendo para ello un mayor consumo y ofreciendo una menor fiabilidad, no tanto por fallo de los equipos de transmisión, si no por pérdidas o atenuaciones debidas a multitud de factores ambientales.

La elección tanto de la incorporación de elementos de transmisión como de la tecnología empleada puede alterar el comportamiento completo del sistema, sobre todo si va a operar en tiempo real. Habitualmente los recursos de analítica de vídeo tienden a concentrar varios flujos de vídeo, con lo que una latencia elevada puede traducirse en un funcionamiento inadecuado del sistema al completo.

3.1.5.- Alimentación

La alimentación sirve para proveer energía. Aunque no tenga influencia directa en la obtención y procesamiento de la señal de imagen, condiciona mucho el funcionamiento

general del sistema. Debe estar dimensionada según los requisitos, tanto de duración como de elementos que deben operar de manera simultánea.

Se puede disponer de alimentación mediante la red eléctrica o mediante una fuente de energía de duración temporal, como un generador eléctrico (de combustible o solar), baterías, o una combinación de ambas. En ambos casos, se debe atender al rango de tensión de funcionamiento de los equipos conectados (el voltaje necesario) y al consumo de intensidad de corriente requerido (amperaje).

Probablemente sea el elemento de un sistema de visión que menos relevancia tenga en cuanto a la obtención y tratamiento de la señal. No obstante, es el más limitante en cuanto a operatividad. Una interrupción de corriente, una pérdida de tensión o de intensidad va a causar una caída general del sistema.

3.2.- Factores y entornos operativos

Un sistema de videovigilancia tiene una misión muy específica y a su vez muy amplia: ejercer control visual. Teniendo en cuenta la gran cantidad de elementos que intervienen en un sistema de visión artificial, se puede intuir por consiguiente que puede haber multitud de configuraciones diferentes. Por lo tanto, también es necesario identificar aquellos factores que van a condicionar su diseño.

Si ya en el apartado anterior se pretendió dar un enfoque operativo práctico, en este apartado se van a considerar aún más si cabe cuestiones puramente operativas, es decir, relativas al uso de un sistema de videovigilancia.

A continuación, se van a detallar en cada uno de los siguientes subapartados cuáles son esos factores y cómo se pueden clasificar los sistemas según dichos factores.

3.2.1.- Finalidad

La finalidad, como su propio nombre indica, es el objetivo que motiva la creación del sistema y encuadra todas las capacidades que debe ofrecer. Previamente se ha reseñado que un sistema de vigilancia tiene siempre como finalidad última establecer un control visual sobre una determinada zona; sin embargo, ese control se puede ejercer principalmente de dos maneras distintas: mediante detección de eventos o por detección de elementos.

La detección de eventos implica identificar, analizar y recoger todas aquellas variaciones en los fotogramas captados por el sensor, sin centrarse en qué o quién los produce. En cambio, la detección de elementos supone la vigilancia de objetivos específicos que aparecen dentro del plano de imagen del sistema.

Es decir, el usuario final puede pretender que el apoyo de medios de imagen sirva para la detección de tipos de objetos concretos como armas, maletines, vehículos, animales, personas, etc. O también que además de identificar el tipo de objeto, se puedan individualizar entre sí (personas o vehículos concretos).

La solución más completa es aquella capaz de combinar ambas posibilidades; es decir, la captación de eventos realizados por determinados elementos. Por ejemplo, si únicamente se quiere grabar eventos en general, normalmente es una función implementada por el propio captador (al responder ante estímulos derivados de cambios de iluminación en las imágenes). Sin embargo, en el momento que se quiere actuar sobre objetos, hace falta herramientas que ayuden a distinguir esos objetos, tal y como haría un operador humano.

Aquí es donde la analítica de vídeo juega un papel preponderante. Sin embargo, también se debe tener en cuenta los recursos disponibles. A más capacidades tenga el sistema de procesamiento de señal, mayor va a ser el coste computacional asociado. Por lo tanto, hay que tratar de buscar un equilibrio entre qué se quiere y qué se puede, sobre todo si no se dispone de equipos con el hardware necesario para ejecutar dicha analítica avanzada de vídeo.

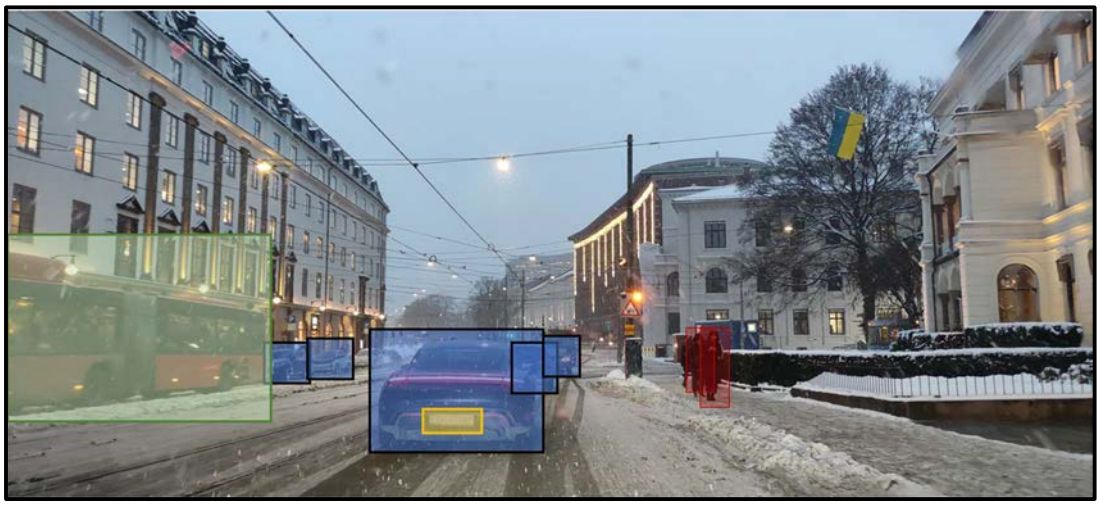


Figura 23: ejemplo de uso de analítica de vídeo

3.2.2.- Condiciones de luz

Como se ha explicado en el apartado 3.1.1, la cantidad de luz disponible condiciona notablemente el funcionamiento del sensor y, por ende, el de todo el sistema. Por lo tanto, el primer criterio que va a afectar al dimensionamiento va a ser la luz existente en la escena a captar.

Los posibles escenarios son: luz diurna o abundante, luz nocturna o escasa y luz variable (una combinación de todas las posibilidades). Además, es habitual que también se manifiesten cambios de iluminación muy grandes y repentinos, como sucede cuando en un escenario nocturno en el que un vehículo se aproxima con las luces encendidas de forma perpendicular al sensor.

Como se detalló en el apartado correspondiente al comportamiento y funciones del sensor, la exposición condiciona la creación de las imágenes. Por lo tanto, un ambiente diurno es más favorables para que opere el sensor. En cambio, en un ambiente nocturno, la disminución de luz residual obliga a utilizar sensores con una fotosensibilidad mayor (del orden de 0’01 lux), o recurrir a elementos de iluminación adicional. Aquí es donde la fotosensibilidad del sensor va a condicionar la necesidad de estos elementos.

Cuando se genera una imagen, es tan importante la ausencia de luz (subexposición) como el exceso (sobreexposición). La incidencia directa de focos de luz puede cegar el sensor saturando las imágenes generadas, como sucede en ocasiones cuando la luz se refleja en las matrículas de los vehículos. En estas ocasiones, es imposible distinguir los caracteres de la matrícula, como se puede apreciar en la Figura 24. Para evitar este problema, una posibilidad es incrementar la luz residual de la escena equilibrando el comportamiento del sensor y reduciendo notablemente la sobreexposición.

Estos condicionantes hacen que se deba valorar correctamente si los elementos de iluminación son necesarios y sí realmente van a tener un efecto positivo (lo que suele ser habitual). Sobre todo, porque su empleo puede generar tres problemas adicionales: un incremento del consumo en el sistema, una mayor facilidad de detección y una ausencia de espacio físico disponible para su instalación.

Figura 24: detalle de una imagen sobreexpuesta por iluminación mal aplicada

3.2.3.- Duración: vigilancia temporal/permanente

Una vigilancia se puede considerar como temporal o permanente dependiendo de dos aspectos: el tiempo total de funcionamiento y la disponibilidad de uso del sistema. Respecto al tiempo de funcionamiento, la diferencia radica en si debe operar ininterrumpidamente, únicamente en determinadas franjas horarias o ante determinados eventos. En cambio, la disponibilidad se refiere al tiempo que la instalación va a estar operativa (independientemente de los instantes de funcionamiento); es decir, cuando se va a poder utilizar.

Clasificar una instalación como temporal o permanente es un concepto difuso. Como se ha explicado con anterioridad, al ser la alimentación un factor muy limitante en el desarrollo de sistemas de video vigilancia, se suele considerar de forma genérica que un sistema es permanente si está conectado a la red eléctrica, y temporal cuando se utiliza una fuente de alimentación finita (baterías o generadores).

La disponibilidad de conexión a la red eléctrica es un factor condicionante en sí mismo porque puede limitar por completo las capacidades del sistema. Aunque se vaya

a desarrollar una vigilancia estática, si las circunstancias de la ubicación seleccionada no ofrecen la posibilidad de disponer de conexión, el sistema se va a tener que diseñar exactamente igual que si se tratase de un sistema móvil. Por lo tanto, la capacidad de almacenamiento, el consumo del equipo y el tamaño de la instalación se van a ver claramente afectadas, como se detallará más adelante.

3.2.4.- Visibilidad de los elementos: vigilancia abierta/encubierta

Este condicionante hace referencia a la apariencia visual. Una vigilancia se clasifica como abierta si alguno de los elementos es perceptible desde el exterior (como por ejemplo las cámaras de control del tráfico).

Figura 25: comparativa entre sistema abierto (izquierda) y encubierto (derecha)

Si la vigilancia es encubierta, todos los componentes del sistema de videovigilancia deben estar ocultos. Esto va a afectar por norma general al tamaño de los componentes y al comportamiento del sensor ante la exposición. Normalmente, si el sistema se encuentra encubierto, la óptica va a permitir una menor entrada de luz, el sensor va a ser más pequeño, producirá imágenes de menor tamaño y probablemente no se van a

poder utilizar elementos de iluminación. Como resultado, se dispondrá de menos información de la escena y de peor calidad.

Otro de los grandes problemas de la vigilancia encubierta es la posibilidad de disponer de acceso a la red eléctrica. En caso de tener que emplear baterías, el principal inconveniente es el tamaño. Imaginando un sistema que tenga un consumo promedio de 1 Ah, un funcionamiento ininterrumpido de 12 horas requerirá como mínimo una batería de más de 12000 mAh de capacidad (hay que tener en cuenta el consumo de pico de arranque de los equipos), con una tensión de 12 voltios. Utilizando como referencia el siguiente ejemplo [122] (dependerá del tipo de batería y del fabricante), el módulo de alimentación tendrá unas dimensiones de 151 x 98 x 95 mm, lo cual es proporcionalmente más grande que el resto de elementos del sistema (dependiendo de los componentes).

Figura 26: imagen de una batería LiFePo

En cambio, si una vigilancia es abierta, al no existir las restricciones de ubicación de los diferentes equipos, se puede conseguir la mejor nitidez y calidad posible de la imagen, pudiendo subsanar esa falta de luz, escasez de profundidad de campo, alcance o amplitud de ángulo de visión.

Aparte de la finalidad última de observar y controlar (que al fin y al cabo es para lo que realmente se instala un sistema de visión), que una vigilancia sea abierta o encubierta puede tener otra finalidad implícita. Si es abierta, la disuasión. Y si es encubierta, la información. El simple hecho de que los elementos de captación sean visibles desde el exterior produce un efecto de intimidación ante posibles agresiones en

la zona vigilada. En cambio, si los medios de captación están ocultos, se va a tratar de conocer quiénes son los posibles agresores, procedimientos y su objetivo final.

3.2.5.- Ubicación de los captadores: vigilancia estática/dinámica

La diferencia entre una vigilancia estática y una dinámica radica en la ubicación de la instalación del sistema. Se denomina estática si la ubicación de los elementos es siempre la misma (importante, una cámara que tenga la capacidad de pivotar para modificar el plano de enfoque, como por ejemplo una cámara domo, no será considerada dinámica si está anclada en un emplazamiento fijo) y dinámica en caso contrario.

Esta configuración va a afectar de manera global al dimensionamiento del sistema, y de manera específica al plano de enfoque. En el primer caso, salvo circunstancias muy puntales en las que los medios de captación se ubiquen en lugares que permitan desplazamientos cortos manteniendo las conexiones (por ejemplo, unos raíles), lo normal es que las vigilancias estáticas sean las únicas preparadas para disponer de alimentación de forma permanente. Por lo tanto, los sistemas destinados a vigilancias dinámicas van a estar supeditadas al uso de baterías, con lo que implica a nivel de capacidades y tamaño total del sistema.

En cuanto al plano de enfoque, parece también evidente que las imágenes que se obtendrán van a ser diferentes si el sistema permanece en movimiento. Las variaciones que se van a producir van a tener una influencia mayor, sobre todo en cuanto a la iluminación y a la profundidad de campo, por lo que el sistema va a tener que ser capaz de poder adaptarse a estos entornos cambiantes.

En este aspecto concreto, es importante reseñar que no se debe confundir una vigilancia dinámica con el hecho de que los elementos que puedan aparecer en el plano de imagen se encuentren en movimiento. Cuando se diseña un sistema, se debe tener preconcebido que van a existir elementos en movimiento. Tras evaluar las primeras imágenes una vez realizada la instalación, se deben realizar las modificaciones pertinentes para cumplir los requisitos marcados, sobre todo pensando en conseguir la distancia hiperfocal.

Empleando un ejemplo, un sistema diseñado para controlar el tráfico se puede concebir sabiendo que los vehículos van a estar en movimiento y a velocidad elevada. Si se quiere realizar el control mediante la lectura de matrículas, se tratará de buscar un ángulo y una profundidad de campo que permita distinguir la matrícula con nitidez el

mayor tiempo posible. Y para ello, se deben hacer diferentes pruebas empíricas hasta hallar la solución definitiva. Se puede por tanto interpretar que en estos casos el principal objetivo va a ser la calidad de los resultados.

En cambio, ante una vigilancia móvil la situación va a ser muy distinta. Al ser un entorno variable, toda la escena puede cambiar, tanto por los elementos que aparezcan en escena como por el plano de enfoque. En estas situaciones, es importante definir claramente la finalidad del sistema y por ende el área de actuación, de tal manera que se debe perseguir principalmente la versatilidad.

3.2.6.- Explotación de las imágenes: vigilancia en tiempo real/diferido

Una vigilancia en tiempo real requiere que se puedan visualizar los eventos en el momento en que se están produciendo. También se podría calificar como vigilancia atendida, ya que la explotación en tiempo real requiere de un operador que esté controlando las imágenes. En cambio, en diferido, únicamente será necesario disponer de la información para su explotación con posterioridad. Se podría considerar una vigilancia desatendida, ya que a priori no sería necesario la presencia física de una persona controlando las imágenes.

Cuando la vigilancia se ejecuta en tiempo real va a ser necesaria la presencia de elementos de transmisión, teniendo en cuenta lo explicado en el apartado correspondiente. En cambio, el visionado en diferido requerirá de mecanismos de almacenamiento que recojan la información y permitan su posterior visionado.

Si bien a priori la visión en tiempo real puede parecer una capacidad muy positiva (sobre todo ante determinadas circunstancias), el empleo de elementos de transmisión tiene un doble efecto negativo. Por un lado, un incremento notable en el consumo eléctrico. De hecho, dependiendo de los mecanismos de procesamiento de señal utilizados, puede constituir el segundo e incluso el primer elemento con mayor consumo. Eso sin entrar a analizar que, dependiendo de factores adicionales como la cobertura, cantidad de transmisión de información, tipo de codificación, etc. ese consumo estándar se puede incrementar. Con lo cual, considerando cuestiones previamente descritas como el tamaño del sistema, la alimentación necesaria y su posible movilidad, lo convierten en un factor muy crítico.

El otro efecto negativo puede ser la aparición de un cuello de botella en la transmisión de la información. Dependiendo de cómo esté concebido el sistema, se puede enviar

todas las imágenes obtenidas, sólo determinados eventos o incluso únicamente datos asociados. De nuevo, dependerá de si existen mecanismos de procesamiento de la señal y de dónde estén ubicados (principalmente si el procesamiento de señal se realiza antes o después de la transmisión). Estas cuestiones están íntimamente relacionadas con la finalidad del sistema.

3.2.7.- Procesamiento de señal: vigilancia tradicional/inteligente

Habitualmente, el visionado de los sistemas de video vigilancia se realiza íntegramente por operadores humanos. Esto es lo que se denominaría vigilancia tradicional, en la que es el operador el responsable de analizar y discretizar la información obtenida. En cambio, los sistemas de vídeo inteligente son aquellos que incorporan elementos de procesamiento de señal.

Bajo el prisma de la seguridad y sobre todo en determinados cometidos, la presencia de un operador humano es obligatoria. Sin embargo, este tipo de tarea puede llegar a ser muy complicada. Los escenarios más complejos son aquellos que requieran un visionado en tiempo real durante un período prolongado de tiempo, sobre todo si existe una gran actividad. Aunque también es especialmente complejo cuando se deben realizar funciones de análisis sobre multitud de imágenes y grabaciones. En estos casos es recomendable la implementación de herramientas de analítica de vídeo, eso sí, como soporte, no como solución definitiva. Ninguno de estos sistemas tiene un cien por cien de precisión, por lo tanto, la figura del operador humano sigue siendo necesaria.

Las ventajas de la incorporación de este tipo de herramientas parecen evidentes, sin embargo, también presentan una serie de inconvenientes. Los equipos responsables de implementar la analítica de vídeo suelen ser más voluminosos que el resto de elementos del sistema y tener asociado un consumo superior. Por lo tanto, en vigilancias dinámicas y encubiertas, va a resultar especialmente complicada su implementación. En estas situaciones lo más recomendable sería utilizar un entorno centralizado en el que se recopilase la información recogida por lo captadores. Pero a su vez, esto también supone un incremento en el consumo de energía en el sistema al ser necesario el uso de elementos de transmisión.

3.3.- Confluencia de factores

A la hora de confeccionar un sistema de videovigilancia, se debe tener un conocimiento de todos los elementos que lo componen y sobre todo de los factores y entornos operativos para su diseño. Sin embargo, estos factores deben ser entendidos como un conjunto y no de forma aislada, ya que salvo circunstancias muy ideales y raras veces existentes, van a condicionarse entre sí. Partiendo de esta base, se debe entender cómo se van a solapar y posteriormente priorizar qué factores son los más importantes y/o los más limitantes.

A priori parece que lo que más peso va a tener va a ser la finalidad del sistema y que por lo tanto todo debe supeditarse a las necesidades que deba satisfacer. Sin embargo, además de la finalidad, es probable que existan otros requisitos de igual importancia y que acaben resultando incompatibles entre sí. Esto es fácil de ver mediante un ejemplo, consistente en un sistema que dispone de todas las capacidades posibles y, por lo tanto, requiera incluir todos los elementos recogidos en el apartado 3.1.

Para suministrar la energía necesaria, seguramente será necesario disponer de alimentación permanente. Con lo cual, desaparece la posibilidad de poder llevar a cabo una vigilancia móvil. Si se pretende que sea móvil, sería necesario usar baterías. Por ende, la vigilancia va a tener una duración temporal; y si se pretende que sea encubierta, el tamaño de las baterías va a tener que ser reducido, disminuyendo aún más el tiempo de uso. Si se quisiera alargar al máximo su vida útil, la única forma sería optimizar el consumo reduciendo la cantidad de elementos existentes en el sistema. Y por lo tanto se perderían funcionalidades.

Es decir, se aprecia claramente cómo la composición de un sistema es un equilibrio entre capacidades y factores operativos, en donde se debe tener muy claro qué se quiere conseguir y a qué se está dispuesto a renunciar.

Esta apreciación es uno de los pilares en los que se ha basado la presente investigación para determinar por qué la concepción de la herramienta aporta un valor añadido superior al ya existente: la posibilidad de incrementar las capacidades que puede ofrecer un sistema sin que suponga alterar su dimensionamiento, al implementar dos procedimientos de analítica de vídeo complementarios.

Volviendo al ejemplo anterior, se puede apreciar la existencia de esa confluencia de factores y cómo condiciona el diseño completo del sistema. Por lo tanto, es importante

reseñar cómo aparecen estas limitaciones y cómo se afectan entre sí. Los detalles que se van a describir a continuación no están ordenados según importancia; esa valoración va a corresponder siempre del usuario final.

3.3.1.- Disponibilidad de alimentación

El acceso o no a alimentación permanente (es decir, conexión a la red eléctrica) actúa como un primer factor condicionante que afecta en cascada al resto de posibles requisitos del sistema. Posteriormente se va a desgranar en los siguientes apartados, pero va a afectar tanto a la movilidad como a su ubicación, al número total de elementos que se pueden incorporar y por último a las capacidades que va a ofrecer.

3.3.2.- Autonomía y Ubicación

Este paradigma suele ser el más habitual. Todo sistema requiere de un mecanismo de alimentación para operar. Como se ha explicado con anterioridad, la mejor forma de conseguir garantizar el máximo tiempo de duración del equipo es disponer de un punto de conexión a la red eléctrica. Sin embargo, esto reduce considerablemente la movilidad, limitándose exclusivamente a una modificación del plano de enfoque.

Si la vigilancia debe ser dinámica, se debe descartar el uso de alimentación continua y por ende habrá que recurrir a baterías. Esto se traduce en temporalidad e incluso en una limitación por consumos, sobre todo los causados por los mecanismos de procesamiento de señal y de transmisión. Además, normalmente el elemento de mayor tamaño en un sistema es la alimentación, con lo que también afecta directamente al encubrimiento del equipo. Por ende, se debe decidir qué es más importante, si el tiempo o el movimiento; y si no se puede renunciar a ninguno de los dos, encontrar un equilibrio. En este caso la mejor solución suele ser emplear una configuración optimizada de encendido y apagado de los diferentes elementos del sistema. Por ejemplo, que el sensor opere en determinadas franjas horarias, que el grabador únicamente capte algunos eventos y que los elementos de transmisión solamente se activen y envíen esos eventos.

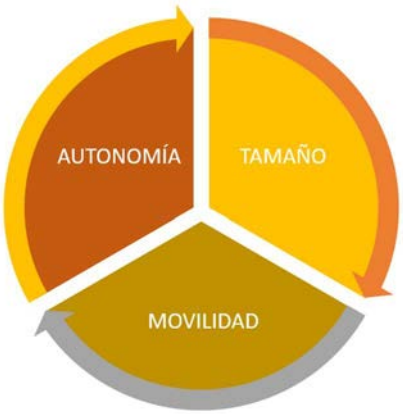


Figura 27: confluencia de factores respecto a la autonomía y la ubicación del sistema

3.3.3.- Visualización, ubicación y duración

De manera análoga al caso anterior, si el sistema debe ser móvil y ofrecer visualización en tiempo real, va a requerir de elementos de transmisión de muy elevado consumo. Vuelve a entrar en juego el factor limitante de las baterías, que afecta todavía más si cabe que en el caso anterior. El problema radica en que la transmisión es una forma activa de radiación de energía (independientemente del protocolo de comunicación que se emplee como WiFi, telefonía móvil, Bluetooth, radiofrecuencia) con una demanda de electricidad permanente. A eso hay que añadir el comportamiento de los equipos según el protocolo de comunicación, que van a tener un consumo adicional al de la transmisión propiamente dicha. Con lo cual, o se incrementa la capacidad de la fuente de alimentación (mayor tamaño, mayor dificultad de encubrimiento y menos agilidad de instalación), o se emplea una configuración optimizada.

Esta cuestión es también un área donde la presente investigación puede resultar trascendente. Una manera de optimizar mucho el consumo de los elementos de transmisión es el envío de la información relevante (qué es o no relevante depende de nuevo del usuario final) o incluso codificada. Como se ha explicado en el apartado correspondiente a los elementos de transmisión, no es lo mismo enviar un flujo continuo de vídeo que únicamente fotogramas concretos, fragmentos de fotograma o incluso un conjunto de datos.

También resulta fundamental saber dónde se va a enviar la información. La ubicación, tanto por cuestiones de cobertura como de distancia, afecta mucho al dimensionamiento del sistema en tamaño y en consumo. Ya se ha mencionado anteriormente que la transmisión de un flujo continuo de vídeo requiere un ancho de

banda elevado. Por distancia y ancho de banda, la solución más versátil suele apoyarse en comunicaciones dentro de la banda de telefonía. Sin embargo, si no se dispone de cobertura con buen ancho de banda (la mejor opción es LTE o superior), o se recurre al denominado pseudo tiempo real (tasas de envío muy bajas que permiten flujos de vídeo por debajo de los 24 fotogramas por segundo [123]), o hay que buscar otra alternativa. Si por ejemplo la distancia entre el sensor y el almacenamiento/visionado no es muy grande (habitualmente inferior al kilómetro), se puede utilizar un radioenlace punto a punto o con algún repetidor intermedio. O si se envían tan sólo pocos fotogramas, la cobertura GSM puede ser suficiente.

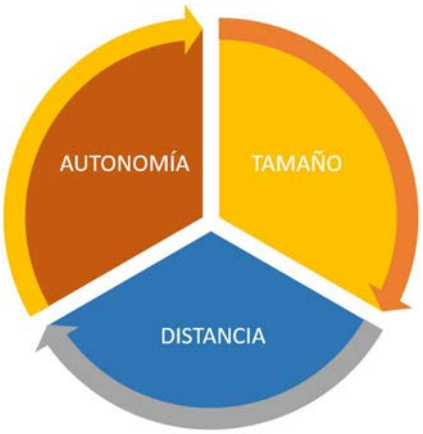


Figura 28: confluencia de factores respecto a la autonomía, la movilidad y la distancia de enlace

Para tratar de incrementar la eficiencia en la transmisión de información, actualmente se están desarrollando herramientas que permiten un flujo inteligente de vídeo [124]. Estos sistemas varían la resolución de las imágenes, o únicamente mandan fragmentos de las imágenes tratando de reducir el ancho de banda de la señal y optimizar el consumo derivado de la transmisión.

3.3.4.-Encubrimiento

El encubrimiento de un equipo afecta al tamaño total del sistema y al empleo de elementos que puedan ser vistos con facilidad desde el exterior. Aunque pueda parecer un contrasentido, suele ser más fácil de encubrir un equipo conectado a la corriente eléctrica que no un sistema alimentado a baterías. Esto es provocado por el tamaño de las propias baterías, que variará según la cantidad de elementos que componga el sistema y por su tiempo de funcionamiento.

Con esto ejemplo se pretende poner de manifiesto que, si el espacio donde debe ubicarse el mecanismo de grabación encubierto no es lo suficientemente grande, puede haber una limitación manifiesta de capacidad de alimentación y por ende, de cómo se debe configurar el sistema globalmente (tanto a cantidad de elementos conectados como de tiempo de funcionamiento).

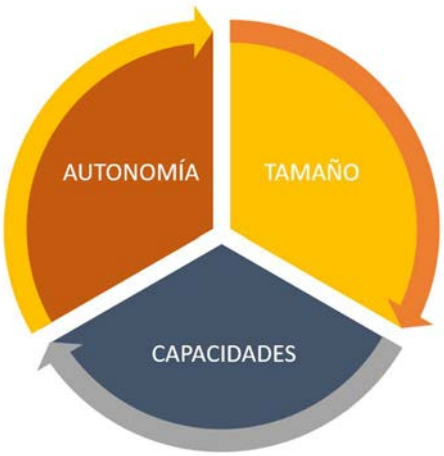


Figura 29: confluencia de factores respecto a la autonomía, el tamaño y las capacidades disponibles

3.3.5.- Finalidad, duración y ubicación

La finalidad del sistema incluye también cuáles son los objetivos que se quieren controlar, sobre qué área se va a establecer un control visual y durante cuánto tiempo debe estar operativo. A priori parece claro entonces que la ubicación de los diferentes elementos debería estar en el lugar que permita cumplir de manera más adecuada con los objetivos establecidos. El problema surge de nuevo si no se dispone de acceso a un punto de conexión eléctrica y se necesita cumplir con una serie de requisitos adicionales, especialmente un tiempo de funcionamiento elevado sumado además a la necesidad de incorporar elementos de transmisión y de procesamiento de la información.

En estos casos es donde se debe priorizar. Si la duración en el tiempo va a ser elevada, probablemente se deba sacrificar un plano de enfoque óptimo en aras de localizar una fuente de alimentación. En cambio, si resulta prácticamente imposible encontrar una ubicación que se adecúe a las necesidades, va a ser necesario limitar tanto el tiempo de funcionamiento como las capacidades ofrecidas (de nada sirve tener acceso ilimitado a energía si no se va a poder hacer un empleo útil del sistema).

No siempre el empleo de un sistema de videovigilancia persigue un control permanente sobre un determinado entorno, en ocasiones simplemente se puede pretender controlar una zona únicamente por espacio corto de tiempo. O en su defecto, se pueden configurar los equipos para operar únicamente bajo determinadas franjas horarias.

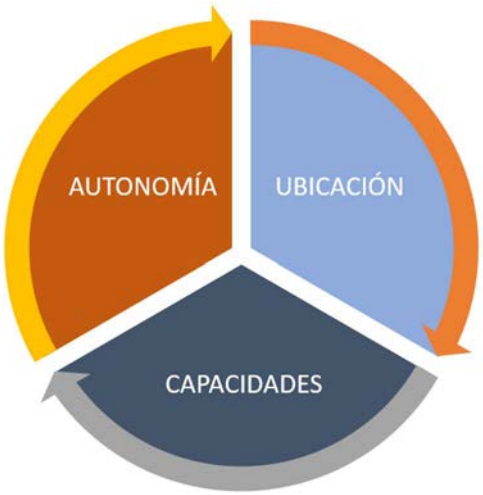


Figura 30: confluencia de factores respecto a la finalidad, la duración y la ubicación de los elementos del sistema

3.4.- Aplicaciones específicas

Un sistema de visión se puede utilizar en una gran cantidad de ámbitos muy diversos. Se puede considerar como una herramienta que permite incrementar las capacidades o agilizar algunas funciones que normalmente realizan las personas.

Por ejemplo, en el ámbito industrial, pueden utilizarse para llevar un control de calidad de la fabricación mediante la realización de mediciones en tiempo real. Disponiendo del patrón de una determinada pieza, el sistema de visión puede capturar imágenes de cada pieza fabricada, realizar las mediciones correspondientes y cotejar si existe algún tipo de defecto [125].

En el sector de la construcción se utilizan para la realización de inspecciones en búsqueda de posibles deterioros o defectos en fachada y estructura [126]. O también para realizar un estudio del relieve y distribución de una determinada zona geográfica [127].

Estos ejemplos sirven para ilustrar el amplísimo campo de aplicación que tienen los sistemas de visión y, en última instancia, cómo de forma directa o indirecta tiene una importante repercusión en el ámbito de la seguridad. Como se ha descrito en la

introducción, es un concepto muy amplio que puede ser aplicado a cualquier otro entorno pudiendo constituir un valor añadido, además de ser un valor o un producto en sí mismo.

Los ejemplos anteriores también ilustran esta idea. Un control de calidad exhaustivo es a la larga una seguridad en la satisfacción de los clientes y por supuesto en la prevención de incidentes o accidentes derivados de la fabricación. O en el caso concreto de la inspección de edificios, se trata una medida de prevención aún más patente si cabe.

Todos estos posibles usos ni pretenden ni pueden sustituir la labor realizada por un ser humano, especialmente si es la seguridad lo que está en juego. Son apoyos muy importantes, complementos a las funciones que deban realizar, pero nunca deben ser un reemplazo. Existen varios motivos por los que actualmente estas herramientas no están preparadas para tener una fiabilidad absoluta y por lo tanto, sobre todo en determinados escenarios, es imposible ceder la responsabilidad de las funciones a un sistema de visión, a pesar de estar dotado de una inteligencia artificial muy elevada y con amplios resultados de fiabilidad. El marco de estudio es un claro ejemplo de esta afirmación.

Los sistemas de visión no se presentan necesariamente de forma aislada, es habitual que convivan varios con diferentes funciones. Unos que contribuyen al cometido principal con el lugar en el que se ubican, y otros específicamente destinados a la seguridad, y que ambos actúen de forma complementaria. Para seguir definiendo de forma específica el marco de la presente investigación, este apartado pretende ofrecer de una forma más gráfica como conviven diferentes sistemas de visión con sistemas de videovigilancia y, a su vez, analizar las particularidades que suelen tener entre ellos, teniendo en cuenta el contenido de los apartados 3.1, 3.2 y 3.3.

3.4.1.- Empleo como herramienta de seguridad

Como se ha reflejado anteriormente, los factores y entornos operativos, unidos a los elementos disponibles permiten una multitud de posibilidades y de configuraciones para el desarrollo de sistemas de visión, independientemente de que existen elementos necesariamente comunes. Los factores operativos reseñados en el apartado 3.3 son los que precisamente guían esas configuraciones, sobre todo cuando se encuadran en el marco de la seguridad.

Para explicar de manera práctica el concepto del párrafo anterior, se va a utilizar como ejemplo un parking público. En este entorno, se pueden identificar diferentes cámaras

que permiten tanto cumplir con el cometido principal (es decir, habilitar el aparcamiento de vehículos), como establecer una seguridad y vigilancia de los vehículos estacionados en el interior. Lo normal es que algunas de estas cámaras estén enfocando al acceso de entrada y salida, captando la matrícula de los vehículos que entran y salen, y así registrar la cantidad de vehículos estacionados y el tiempo que transcurren en el interior del aparcamiento. El resto se suelen emplear como mecanismo de prevención, para evitar robos o daños en los vehículos y poder actuar en caso de incidencia.

El primer tipo de cámaras (las del control de acceso), necesitarán tener la capacidad de identificar y registrar las matrículas de los vehículos que accedan y salgan. Para ello, todos los elementos del sistema estarán destinados a obtener la imagen de dicha matrícula con la mayor calidad y contraste posible. Esto se traduce en disponer de una lente que capte sin aberraciones la matrícula y con la iluminación adecuada, que facilite el contraste entre los números y letras de la misma con respecto del fondo, así como la matrícula con el resto del vehículo.

En este caso concreto, el trabajo principal recae en el sistema de procesamiento de la señal, ya que tiene la finalidad de interpretar ese fotograma de la matrícula, extraer los caracteres, almacenarlos y contabilizar ese vehículo.

Figura 31: ejemplo de una cámara destinada al control de matrículas

El resto de elementos del sistema están elegidos e instalados para facilitar dicha labor. De ahí que la cámara y la iluminación sean grandes, visibles, y enfoquen directamente a la matrícula. Además, se facilita la labor de adquisición de los fotogramas de la matrícula, gracias a que el coche debe permanecer parado hasta que se haya completado el proceso.

Este ejemplo sirve para identificar una configuración de un sistema de visión, en este caso destinado a la detección y reconocimiento de los vehículos que acceden al interior del parking (de forma muy resumida). Ahora toca prestar atención a los elementos de seguridad.

Bajo el prisma de la seguridad ciudadana, como se ha explicado en el apartado 1.3 (que es uno de los pilares que motiva la presente investigación), existen principalmente dos enfoques: la prevención y control y la investigación. Se podría afirmar que la diferencia entre ambos tipos de enfoques es que los destinados a la prevención buscan proteger un lugar o un bien concreto, y por ende tienen utilidad mientras exista el objeto a proteger. En cambio, los destinados a la investigación son útiles mientras consigan obtener información. Y la información suele ser más valiosa si los elementos del sistema se encuentran encubiertos (por razones obvias). En este caso el encubrimiento tiene una gran relevancia, directamente relacionado con la finalidad de implementación del sistema.

Es importante reseñar que no son dos conceptos excluyentes. Es decir, sistemas concebidos para operar en entornos de prevención pueden obtener información válida para llevar a cabo una investigación. Por ejemplo, las cámaras de una gasolinera, que son claramente visibles, permiten obtener información muy útil en caso de robo en la misma o incluso de un vehículo que haya podido estar involucrado en un hecho criminal y que haya repostado en dicha gasolinera.

El ejemplo de la gasolinera coincide con el de la mayoría de los sistemas de seguridad destinados a labores de prevención. Están diseñados para operar en entornos controlados donde se suelen instalar cámaras estáticas, acompañados de objetivos de gran tamaño (que contribuyen a una mejor resolución en la imagen) y una iluminación pensada para eliminar cualquier tipo de aberración. Además, gracias a la visibilidad de sus elementos contribuyen a la disuasión (una de las mejores formas de prevención). Pero esto no significa que no existan otros tipos de sistemas también destinados a la prevención y que tengan una configuración encubierta. Por ejemplo, en los cajeros

automáticos se instalan cámaras que no siempre son del todo visibles, y que sirven para identificar las acciones de un usuario en casos de posible suplantación de identidad o de ejercer violencia física sobre el propio cajero. Normalmente, las diferentes configuraciones en instalaciones físicas (como es el caso) están directamente relacionadas con los denominados círculos de seguridad.

3.5.1.1.- Círculos de seguridad

Las medidas de seguridad física implementadas en una instalación se pueden dividir en tres círculos o anillos [128]: seguridad lejana, próxima e inmediata. Aunque las diferentes medidas de seguridad suelen ser una combinación de varios elementos distintos (como sensores magnéticos, térmicos, infrarrojos, etc.), esta configuración en círculos también se puede encontrar cuando únicamente se empleen sistemas de videovigilancia (no hay que olvidar que al final este tipo de sistemas son una herramienta más de las que se pueden emplear para garantizar la seguridad de una instalación).

Figura 32: esquema visual con los diferentes anillos de seguridad

Continuando con el ejemplo anterior del parking, estos tres círculos se podrían identificar de la siguiente forma: manteniendo funciones de seguridad lejana o perimetral estarían las cámaras de control de matrículas y de control de acceso (normalmente en combinación con otros elementos complementarios como es la propia barrera de acceso o las puertas peatonales con cerradura electrónica). Las cámaras que

controlan los vehículos estacionados (que además son uno de los activos principales) serían las de seguridad próxima. Y, por último, las de seguridad inmediata serían aquellas que controlan directamente las estancias y máquinas de pago, además de contar con un elemento activo como podría ser un vigilante.

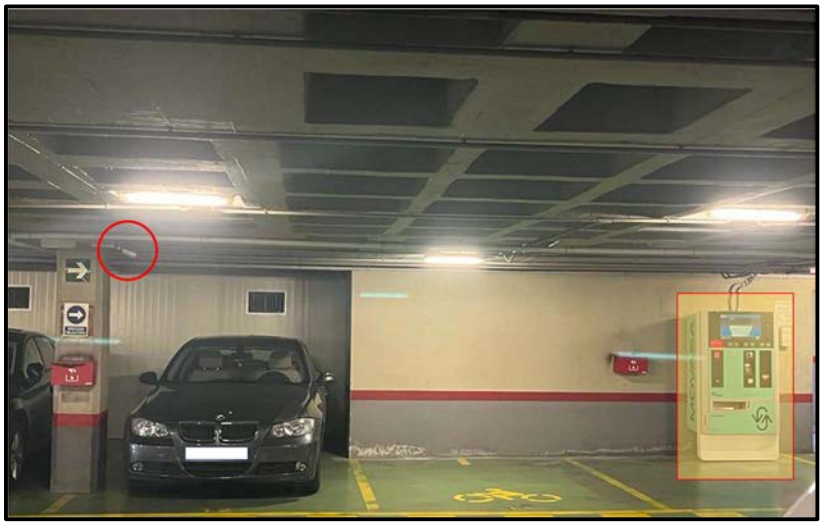


Figura 33: detalle de un círculo de seguridad inmediata en un parking

De nuevo vuelven a aparecer diferencias notables entre los sistemas que operan según el círculo en el que se ubican. Mientras que los de seguridad lejana o perimetral cumplen principalmente esa función de disuasión, los de seguridad próxima y sobre todo seguridad inmediata se centran más en el control que en la disuasión. Por ende, si cuando se explicó la diferencia de sistemas de visión funcionales respecto a los destinados a la videovigilancia, parece entendible que también habrá diferencias entre las herramientas empleadas en cada uno de los círculos. De manera análoga a lo que sucede entre sistemas destinados a la prevención y los destinados a la investigación, el círculo lejano y el próximo serán más parecidos a los concebidos para la investigación. No porque se pretenda obtener información bajo el prisma de labores policiales, si no por una cuestión de minimizar el impacto hacia los empleados y los clientes (al final las cámaras no dejan de tener un efecto intimidatorio).

Por lo tanto, parece que los medios utilizados en la seguridad inmediata van a valorar el encubrimiento casi tanto como cumplir con la función para la que se instalan. La manera más sencilla de encubrir medios es reducir el tamaño de los captadores, trabajando con sensores muy pequeños (por ejemplo, de cuarto de pulgada o 1/2.55") y ópticas de tamaño proporcional al del sensor. El encubrimiento va a acarrear una reducción en la luminosidad que va a penetrar a través del sensor, por lo que la imagen

va a ser de menor calidad (o al menos con menor luminosidad), va a tener menor rango de alcance y va a implicar que las cámaras estén más próximas al sensor.

Las diferencias en las características de los equipos instalados en cada círculo de seguridad, son análogas a las que se pueden identificar en los sistemas concebidos para la prevención y para la investigación. La gran diferencia en ambos tipos de configuración radica principalmente en el tamaño de los equipos (como se refleja en la Figura 25), o en su defecto, en la búsqueda de un encubrimiento o no de los mismos (principalmente de los captadores), sin perjuicio de que el resto de elementos puedan ser muy parecidos o incluso comunes.

3.5.- Conclusiones parciales

Con este capítulo, se ha perseguido un triple objetivo:

- El primero, describir un sistema de videovigilancia, las partes que lo componen y la influencia que puede tener cada componente en los resultados finales de forma aislada.
- El segundo, proporcionar una visión práctica real de los diferentes factores que condicionan la configuración y desarrollo de un sistema de visión, diferente a otros puntos de vista centrados en aspectos más teóricos del funcionamiento.
- Por último, poner de manifiesto por qué ese enfoque práctico es necesario para establecer cómo se debe actuar ante un ecosistema que no es cerrado, que permite multitud de configuraciones y cómo éstas alteran por completo el resultado final.

Estas premisas sirven para apuntalar uno de los objetivos que ha guiado la investigación, y que constituye casi el principal paradigma en el desarrollo de la solución perseguida: la versatilidad. De hecho, viene claramente expresada como uno de las metas reseñadas en el apartado 1.5.

Un error que a veces se comete a la hora de enfocar una investigación es olvidar las posibles aplicaciones directas que puedan tener los resultados conseguidos. En muchas ocasiones se presentan herramientas con grandes capacidades técnicas, muy innovadoras, pero en las que no se ha tenido en cuenta cuáles son las circunstancias reales de trabajo y, por lo tanto, acaban no resultando útiles. Esto se manifiesta bastante actualmente cuando se presentan diferentes soluciones de analítica de vídeo, en las que se muestran resultados a priori muy positivos e innovadores, pero que están encapsulados en entornos completamente estancos (requieren del empleo de captadores

y otros elementos compatibles con el fabricante) y que encima únicamente funcionan correctamente en situaciones muy ideales (es decir, que gran parte del éxito de los resultados proviene de una correcta elección y ubicación del resto de elementos del sistema). La inclusión en el documento de todas las cuestiones y circunstancias operativas pretende evitar estos problemas, pensando precisamente en favorecer la versatilidad.

También se quiere reseñar la influencia a nivel operacional de factores a priori nimios como la alimentación del sistema, que, sin embargo, si se analiza el apartado 3.3 está presente en absolutamente todos los entornos. Tanto si se pretende ganar movilidad, capacidad o reducir tamaño, está totalmente supeditado al uso de electricidad corriente o de baterías. De hecho, esta cuestión es transversal a otros ámbitos, como por ejemplo en el desarrollo del vehículo eléctrico.

Para finalizar, en este capítulo se han visto multitud de factores que podrían ser objeto de una investigación y desarrollo más profundo de manera individual. No obstante, la investigación actual se ha centrado casi en exclusiva en el sistema de procesamiento de señal por varios motivos. Primero, porque como se ha comentado anteriormente en este apartado de conclusiones, se pretende seguir un enfoque práctico y que se desmarque de la tendencia actual a forzar la adquisición de ecosistemas estancos. Y segundo, porque la evolución en la aplicación de inteligencia artificial en este ámbito concreto puede ofrecer avances muy notables y prácticos, sobre todo, como ayuda a los operadores responsables del control. Por lo tanto, al igual que en este capítulo se han reseñado cuestiones teórico – prácticas relativas a los sistemas de visión, el siguiente capítulo pretende hacer lo mismo pero centrado en la herramienta que se ha desarrollado, de una forma más concreta.

IDENTIFICACIÓN VISUAL IV

En el capítulo correspondiente al estado del arte se han ofrecido una gran multitud de soluciones para poder abordar el problema planteado. Sin embargo, únicamente se han tocado de forma tangencial algunos aspectos teóricos. Para poder justificar y explicar correctamente la presente investigación, se considera importante resaltar contenidos teóricos correspondientes principalmente a los algoritmos de inteligencia artificial, ya que primero, son los más utilizados actualmente (tal y como se aprecia en el Capítulo II) y segundo, porque también se han empleado en la presente investigación.

4.1.- Inteligencia artificial: machine learning y deep learning

La inteligencia artificial (IA) es un término que refiere a la capacidad de las máquinas para realizar tareas que normalmente requieren inteligencia humana, como la percepción visual, la interpretación del lenguaje, el razonamiento y la toma de decisiones. La IA tiene una gran cantidad de aplicaciones y campos en los que subyace una combinación de técnicas de aprendizaje automático, procesamiento del lenguaje natural, procesamiento de imágenes y robótica [129-132].

Existen diferentes aproximaciones de aplicación de la inteligencia artificial. Sin embargo, los conceptos que más afectan a esta investigación son el aprendizaje automático o *machine learning* y sobre todo el aprendizaje profundo o *deep learning*. Encontrar una diferencia entre ambos conceptos es en ocasiones difuso. Según determinados autores [133,134], el *machine learning* abarca una amplia variedad de algoritmos y técnicas como la regresión lineal, árboles de decisión, máquinas de vectores de soporte (*support vector machine* o SVM), o algoritmos de agrupamiento, técnicas que tienen un recorrido histórico más largo en cuanto a su uso, debido a tener un coste computacional menor. Estos algoritmos pueden ser de aprendizaje supervisado, no supervisado o semi-supervisado, y a menudo se basan en la extracción manual de características y su posterior selección para representar y procesar los datos [135].

Sin embargo, se puede establecer que el *deep learning* son un conjunto de algoritmos más sofisticados que el *machine learning* [136], principalmente porque los segundos tienen una estructura de aprendizaje más automatizada (ya que los primeros requieren

datos estructurados para realizar las predicciones), basada principalmente en las redes neuronales [137,138]. El aprendizaje profundo es una rama del aprendizaje automático que se centra en el uso de redes neuronales artificiales (*artificial neural networks* o ANN) con múltiples capas ocultas, también conocidas como redes neuronales profundas. Estos modelos son capaces de aprender representaciones jerárquicas y no lineales de los datos de manera automática, sin la necesidad de requerir una extracción manual de características [139]. De manera gráfica, la clasificación de las diferentes técnicas se podría entender de la siguiente manera:

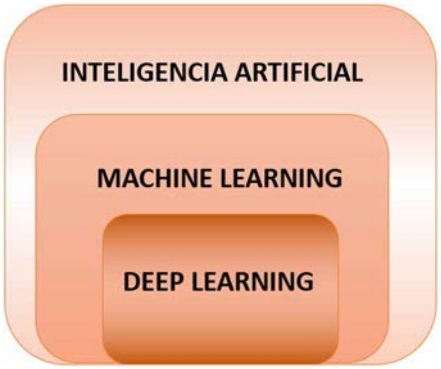


Figura 34: diagrama de *deep learning* frente a *machine learning*

4.2.- Redes neuronales aplicadas al procesamiento de imagen

4.2.1.- Qué es una red neuronal artificial

El algoritmo que mejor representa el *deep learning* es la red neuronal artificial o ANN (*artificial neural network*). Una red neuronal artificial es un modelo matemático que se inspira en la estructura y el funcionamiento del cerebro humano para la resolución de determinados problemas [140]. Se puede utilizar en multitud de ámbitos como el procesamiento de lenguaje natural, la toma de decisiones o el campo que afecta a la presente investigación, el procesamiento de imágenes.

Las redes neuronales cuentan con diferentes componentes [136,141]:

- **Nodos**: las neuronas artificiales, también conocidas como nodos o unidades de procesamiento, son los elementos fundamentales de una red. Están organizadas en capas interconectadas entre sí mediante enlaces ponderados llamados pesos sinápticos.
- **Capas:** se pueden distinguir tres tipos de capas:

- Capa de entrada: es la responsable de recibir los datos de entrada y los distribuye a las neuronas en la siguiente capa. No realiza ningún cálculo en sí misma.
- Capas ocultas: están ubicadas entre la capa de entrada y la capa de salida. Puede haber varias capas ocultas y el número de neuronas en cada capa oculta puede variar. Son responsables de realizar la mayoría de los cálculos y de extraer patrones y características de los datos de entrada.
- Capa de salida: produce la salida final. El número de neuronas en la capa de salida depende de la cantidad de clases o de la naturaleza de la tarea que la red deba realizar. Por ejemplo, si se trata de un clasificador, lo normal es que la última capa tenga tantos nodos como categorías permita el clasificador.

- **Pesos sinápticos**: son los valores numéricos que representan la conexión entre las neuronas en las diferentes capas. Estos pesos son ajustables y determinan la fuerza de la conexión entre las neuronas
- **Funciones de activación**: son funciones matemáticas aplicadas a las neuronas para determinar si deben o no y cómo deben transmitir una señal a las neuronas en la siguiente capa. Algunos ejemplos de funciones de activación habituales son la función sigmoide, la tangente hiperbólica o la función de rectificación lineal (ReLU) [142].

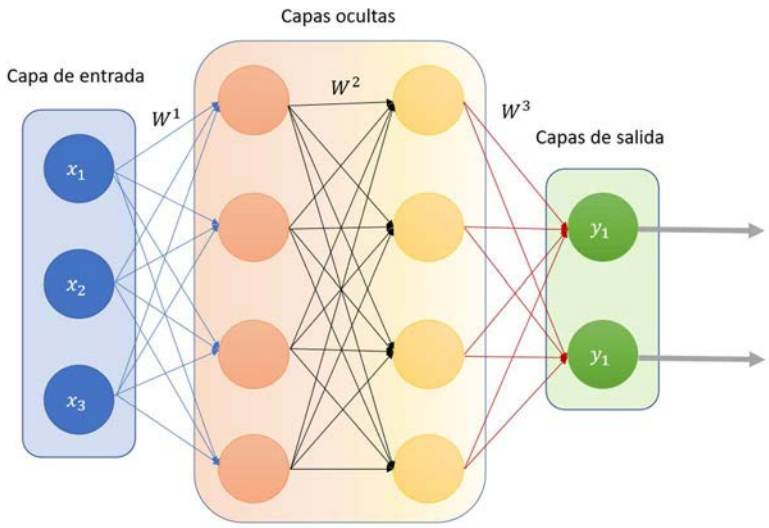


Figura 35: representación de una red neuronal

4.2.2.- Tipos de redes neuronales

Existen multitud de redes neuronales. A continuación, se van a incluir algunos ejemplos más habituales, para posteriormente detallar cuáles son del tipo más relevante asociadas a la presente investigación [140,141]:

- **Redes neuronales prealimentadas (Feedforward Neural Networks, FNN):** Son el modelo más simple de redes neuronales. Se caracterizan porque las conexiones de las neuronas hacen que los datos únicamente fluyan en una sola dirección, desde la capa de entrada, a través de una o más capas ocultas, hasta la capa de salida. No hay conexiones hacia atrás ni bucles. Se suelen utilizar normalmente para resolver problemas de clasificación y regresión.
- **Redes neuronales convolucionales (Convolutional Neural Networks, CNN) [143]:** las redes neuronales convolucionales son un tipo especializado de red diseñado para el procesamiento de imágenes y señales. Utilizan capas convolucionales para aprender características de los datos de entrada. Son ampliamente utilizadas en tareas de clasificación y detección de objetos en imágenes, así como en el análisis de secuencias de tiempo y datos de series temporales.
- **Redes neuronales recurrentes (Recurrent Neural Networks, RNN) [144]:** son redes que se caracterizan por disponer de conexiones hacia nodos anteriores en la secuencia de la red, lo que permite mantener un "estado interno" y procesar secuencias de datos, como si se tratase de un pequeño buffer de memoria. Su evolución ha permitido un gran avance en tareas que conllevan datos secuenciales, como el procesamiento del lenguaje natural, el reconocimiento de voz y la predicción de series temporales.
- **Redes de memoria a corto y largo plazo (Long Short-Term Memory Networks, LSTM) [145]:** son una evolución de las anteriores. Se caracterizan por crear lo que se denomina una "celda de memoria" para aprender y almacenar información a lo largo de secuencias largas. Estas celdas de memoria permiten a las LSTM abordar problemas de dependencias de largo alcance y evitar el desvanecimiento del gradiente, de manera que tienen mayor capacidad que las anteriores y presentan una mayor eficiencia que las RNN tradicionales. Las LSTM se utilizan comúnmente en tareas de procesamiento del lenguaje natural, como la traducción automática y la generación de texto.

- **Redes neuronales de atención (Attention Networks) [99]:** Las redes de atención son un mecanismo que se puede utilizar junto con otras arquitecturas de RNA, como las RNN y las LSTM, para mejorar la capacidad de la red para enfocarse en partes relevantes de la entrada al realizar predicciones. Este enfoque ha demostrado ser especialmente efectivo en tareas de procesamiento del lenguaje natural, como la traducción automática y el resumen de texto. Estos procedimientos ya se explicaron más en detalle en el apartado 2.1.2.5 del estado del arte.
- **Redes generativas adversarias (Generative Adversarial Networks, GAN) [96]:** constan de dos redes neuronales, un generador y un discriminador, que compiten entre sí en un juego de suma cero (como ya se introdujo en el apartado 2.1.2.4). El generador intenta crear muestras sintéticas realistas, mientras que el discriminador intenta distinguir entre las muestras reales y las sintéticas. La evolución de estar redes ha aportado numerosas mejoras en el procesamiento de imagen; entre otras cosas, porque es especialmente útil para evitar ataques esteganográficos en imágenes (es decir, imágenes a priori igual a las originales pero corruptas en algún punto) y porque permite incrementar notablemente la cantidad de imágenes contenida en un dataset [96].
- **Redes neuronales autoencoder (Autoencoder Neural Networks, AE) [146]:** son redes empleadas para aprender representaciones eficientes y comprimidas de los datos de entrada. Constan de dos partes, un codificador que transforma los datos de entrada en una representación de menor dimensión y un decodificador que reconstruye los datos de entrada a partir de la representación comprimida. Los *autoencoders* se utilizan comúnmente en tareas de reducción de dimensionalidad, eliminación de ruido y aprendizaje de características y es habitual que sirvan como elemento de pre procesamiento de las imágenes, como se puede reflejar en alguno de los ejemplos situados en el estado del arte.
- **Redes neuronales de crecimiento (Growing Neural Networks, GNN) [147]:** son redes que pueden cambiar su estructura y tamaño durante el proceso de entrenamiento, agregando o eliminando neuronas y conexiones. Esto permite a la red adaptarse a la complejidad del problema y evitar el sobreajuste. Las GNN se utilizan en una variedad de tareas de aprendizaje automático, como la clasificación, la regresión y el agrupamiento.

4.2.3.- Entrenamiento de una red

El entrenamiento de la red es quizá uno de los procesos más importantes a la hora de trabajar con estos algoritmos [139]. De hecho, gran parte de los esfuerzos y resultados de la presente investigación se han incardinado en esta área. Por lo tanto, es importante definir e introducir los procesos de entrenamiento, que van a cambiar según la configuración y finalidad de la red. El correcto entrenamiento de una red va a condicionar cómo se realizarán las predicciones posteriores, además de permitir validar o descartar el diseño correcto de la red elegida.

Pero para desgranar en mayor detalle cómo se realiza el entrenamiento de una red, primero es necesario tener en cuenta cuáles son los datos y parámetros necesarios para su entrenamiento. A parte de la configuración de la propia red, es muy importante disponer de un *dataset* (o conjunto de datos) adecuado y una correcta configuración de los hiperparámetros.

4.2.3.1.- Hiperparámetros

Son los valores que definen el comportamiento de la red durante el entrenamiento [141,148]. La elección y configuración de los mismos se realiza con anterioridad al comienzo del entrenamiento de la red y en función de los resultados obtenidos de entrenamiento y validación, se suelen alterar para tratar de mejorar los resultados. Los valores más importantes son:

- **Tasa de aprendizaje**: determina la velocidad a la que los parámetros de la red se actualizan durante el entrenamiento. Si se configura una tasa de aprendizaje demasiado alta puede causar inestabilidad en el entrenamiento, ya que se traduce en una variación muy elevada de los pesos de la red, mientras que una tasa de aprendizaje demasiado baja puede resultar en un entrenamiento lento y poco óptimo.
- **Tamaño del lote o *batch size*:** define la cantidad de elementos del *dataset* de entrenamiento que se procesan simultáneamente en cada iteración. Un mayor *batch size* puede proporcionar una estimación más precisa del gradiente, pero también consume muchos recursos computacionales y puede requerir un tiempo de procesamiento mayor e incluso generar una interrupción en el entrenamiento.
- **Número de ciclos o *epochs:*** se trata del número de ciclos completos que todo el *dataset* recorre el algoritmo; en definitiva, el número total de ciclos de

entrenamiento. Teóricamente, un número mayor de *epochs* implica un mejor aprendizaje de la red y una mejor precisión, pero no sólo incrementa el tiempo de entrenamiento, sino que también puede derivar en un sobreajuste u *overfitting*.

- **Regularización:** son técnicas utilizadas para prevenir el sobre entrenamiento y mejorar la generalización del modelo. Existen varias técnicas de regularización, aunque las más utilizadas son: la regularización L1 y L2 [149] , que agregan términos adicionales a la función de pérdida para limitar la magnitud de los parámetros de la red; la regularización por abandono o *dropout*, que implica la eliminación aleatoria de algunas neuronas durante el entrenamiento para evitar la dependencia excesiva de cualquier neurona en particular; la parada repentina del entrenamiento o *early stopping* (esto se realiza cuando el entrenamiento se está monitorizando para evitar un consumo muy grande de recursos cuando ya se está apreciando un posible *overfitting*) y la técnica más relevante y empleada en redes neuronales aplicadas al procesamiento de imágenes, llamada *data augmentation* [150]. Esta última realiza transformaciones geométricas en las imágenes contenidas en el *dataset* para incrementar los datos disponibles durante el entrenamiento.
- **Optimizador:** define el algoritmo utilizado para actualizar los parámetros de la red durante el entrenamiento. Algunos optimizadores populares incluyen el descenso de gradiente estocástico (*stochastic gradient descent* o SGD) [151], Adam [152] o RMSprop (*root mean square propagation)* [153].

4.2.3.2.- Dataset

El *dataset* o conjunto de datos son datos estructurados que se utilizan para el entrenamiento y validación de algoritmos, teniendo como aplicación directa la resolución de un planteamiento previo concreto [154]. Puede contener información en varios formatos, como texto, números, imágenes, audio o video.

Cuando se confecciona un *dataset* para la realización de un entrenamiento de aprendizaje supervisado específico, es tan importante las imágenes recopiladas como el etiquetado que lleven asociado. El etiquetado [155] es el proceso por el cual se indica qué es cada uno de los datos contenidos en el *dataset*. Por ejemplo, si el *dataset* está concebido para entrenar un clasificador, el etiquetado correspondería a imágenes asociadas con una categoría concreta de objeto. Si además fuese un clasificador – detector, el etiquetado no sólo contendría una etiqueta de clase si no también los píxeles de la imagen que

delimitan el *bounding box* o zona de la imagen en la que se ubica el objeto correspondiente.

El etiquetado se puede realizar de manera manual, automática o mediante una combinación de ambos procesos. La elección de un procedimiento u otro depende del tipo de *dataset* y sobre todo de la finalidad última que se quiera perseguir. Cuando se hable de los *datasets* específicos que se han confeccionado en el presente documento, se detallará el procedimiento empleado y el por qué del mismo.

Cuando se confecciona un *dataset*, de cara a realizar el entrenamiento la carga de las imágenes se suele dividir entre conjunto de entrenamiento, de validación y de test. El conjunto de entrenamiento, como su propio nombre indica, sirve para entrenar el modelo y ajustar los hiperparámetros, y constituye la mayor parte del dataset. El conjunto de validación sirve para evaluar el entrenamiento y analizar la precisión del modelo con datos independientes a los del entrenamiento, y constituye el porcentaje minoritario del *dataset*. Por último, el conjunto de test es la prueba definitiva del sistema. En ocasiones (como además sucede en la presente investigación), se puede recurrir a otro *dataset* diferente al de entrenamiento y validación para comprobar el correcto entrenamiento y funcionamiento de la red en escenarios lo más reales posibles.

4.2.3.3.- Procesos producidos durante el entrenamiento

Los pasos del proceso del entrenamiento de la red son los siguientes:

- **Inicialización:** los pesos sinápticos se inicializan con valores aleatorios.
- **Propagación hacia adelante (o *forward pass*):** la red se alimenta con el *dataset*, cuyos datos de entrada van pasando a través de cada nodo de las diferentes capas ocultas y en función de los pesos y de las funciones de activación proporciona un valor en la capa de salida de la red.
- **Cálculo del error**: se compara los datos obtenidos en la capa de salida de la red con los valores reales correspondientes al *dataset* y se calcula el error utilizando una función de pérdida o coste, como el error cuadrático medio o la entropía cruzada.
- **Retropropagación del error (o *backpropagation*)**: el error calculado en el paso previo se propaga hacia atrás desde la capa de salida hasta la capa de entrada, ajustando los pesos sinápticos en función de la contribución de cada neurona al

error total. Este proceso utiliza la derivada de la función de activación y el método de optimización.

- **Iteración y actualización de los pesos**: los pasos 2 a 4 se repiten varias veces utilizando diferentes datos de entrada hasta que los pesos sinápticos converjan y el error total se reduzca a un valor aceptable. El número de veces que se repiten estos pasos se denominan (*epochs*).
- **Validación y ajuste de hiperparámetros**: Después de entrenar la red se evalúa su rendimiento en un conjunto de datos de validación independiente para verificar si el modelo generaliza bien y si es necesario, se ajustan los hiperparámetros.
- **Evaluación del rendimiento**: Se trata de otro de los pasos más importantes en el desarrollo de cualquier investigación relacionada con este tipo de algoritmos. Para verificar los resultados del entrenamiento de la red, se suelen analizar varios resultados que ayudan a evaluar el rendimiento del modelo. Las más habituales son las métricas de rendimiento [156], como la exactitud (*accuracy*), la precisión (*precission),* la sensibilidad (*recall*), el área bajo la curva ROC (AUC-ROC) y la puntuación F1.

4.2.4.- Redes Neuronales Convolucionales aplicadas al procesamiento de imagen

El algoritmo más empleado en la presente investigación está íntimamente ligado a la figura de un clasificador, tal y como se va a detallar más adelante. Debido a que se basa fundamentalmente en una red neuronal convolucional, que además tienen una influencia muy directa en la mayoría de trabajos relacionados con el procesamiento de imágenes constituyendo prácticamente el núcleo de la gran mayoría de los trabajos recopilados en el estado del arte (Capítulo II), a continuación, se va a proceder a desgranar más en detalle qué es una red neuronal convolucional o CNN, así como diversos factores a tener en cuenta [157-159].

4.2.4.1.- Convolución

El apelativo de convolucional viene proporcionado porque la mayoría de operaciones realizadas en las diferentes capas de la red son convoluciones. Una convolución es una operación matemática que se traduce en un producto – suma entre dos matrices.

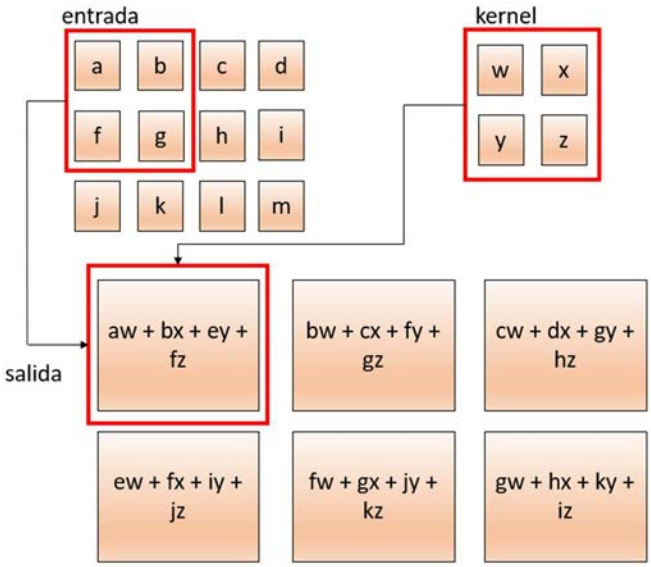


Figura 36: representación de una convolución

Cada una de estas matrices persigue como finalidad la extracción de características de las imágenes de entrada. La agrupación de varias convoluciones en un único paso de la red recibe el nombre de capa convolucional.

4.2.4.2.- Capas convolucionales

Durante el primer paso de la red (*forward propagation*), se aprovecha la distribución de una imagen digital como una matriz de píxeles, de tal manera que los diferentes filtros o *kernel* convolucionan a lo largo de todos los píxeles de la imagen, obteniendo a la salida de la red lo que se denomina *features map* o mapa de características, que precisamente resalta esas características especiales de las imágenes de entrada. A parte del tamaño del filtro y la cantidad de ellos por cada capa, existe otro parámetro muy importante, denominado paso o *stride*, que define el número de píxeles que el filtro se desplaza en cada paso durante la convolución. Un *stride* de 1 significa que el filtro se desplaza un píxel a la vez, mientras que un *stride* de 2 significa que se desplaza dos píxeles a la vez. Un *stride* mayor reduce la dimensionalidad de los mapas de características resultantes y aumenta la eficiencia computacional.

4.2.4.3.- Capas pool

Las siguientes capas de una red convolucional son las capas de *pooling* o agrupación. Estas capas reducen las dimensiones de los *features maps* mediante la selección de determinados valores dentro de una submatriz. Por ejemplo, si se trata de *max pooling*, se va a seleccionar el valor más alto dentro de esa submatriz.

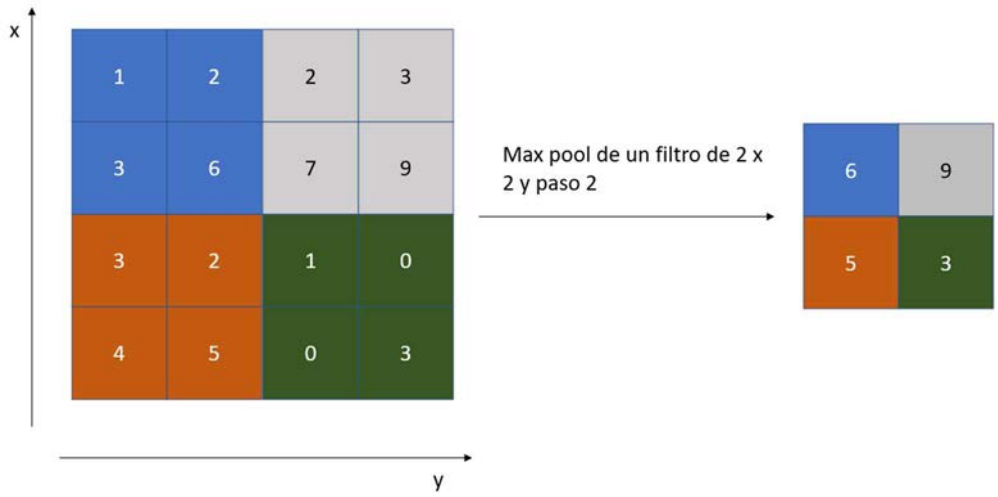


Figura 37: ejemplo de *max pooling*

Normalmente, las capas convolucionales y las de *pooling* se suelen agrupar en grupos. Después de ejecutar varios grupos de operaciones convolucionales y de *pooling*, las características extraídas se aplanan en un vector unidimensional. Este vector se pasa a través de una o varias capas completamente conectadas (también conocidas como capas densas o *fully connected*).

4.2.4.4.- Estructura global de la red

Las CNN son ampliamente empleadas en el procesamiento de imágenes [160]. Probablemente sea el algoritmo actualmente más utilizado para este tipo de entornos, como se ha podido ver en detalle en el capítulo correspondiente al estado del arte. Sin embargo, pueden tener diferentes enfoques o usos que condicionan la estructura de la red. En el caso concreto de la presente investigación, los usos más importantes de este tipo de redes son como extractor de características o *backbone*, como clasificador y como mecanismo de detección y localización de objetos.

Las principales diferencias en la configuración de la red se encuentran sobre todo en la salida de la última capa. Por norma general, todas las CNN presentan una capa *fully connected* posterior a las convoluciones que a efectos prácticos se puede considerar como una codificación inequívoca de la imagen de entrada en forma de tensor. Si el proceso de la red finaliza ahí, se obtiene un codificador de las imágenes de entrada, en la que se resaltan una serie de características específicas. Este esquema en su conjunto suele recibir el nombre de *backbone*, y constituyen el núcleo convolucional de la red en la que el tensor de salida se verá sometido a otra serie de procesos.

En un clasificador, la última capa *fully connected* tendrá tantas neuronas como clases tenga el clasificador, y posteriormente, se aplica la función *softmax* para conseguir una probabilidad normalizada de suma uno en la que la probabilidad más elevada correspondería a la clase que identifique el clasificador.

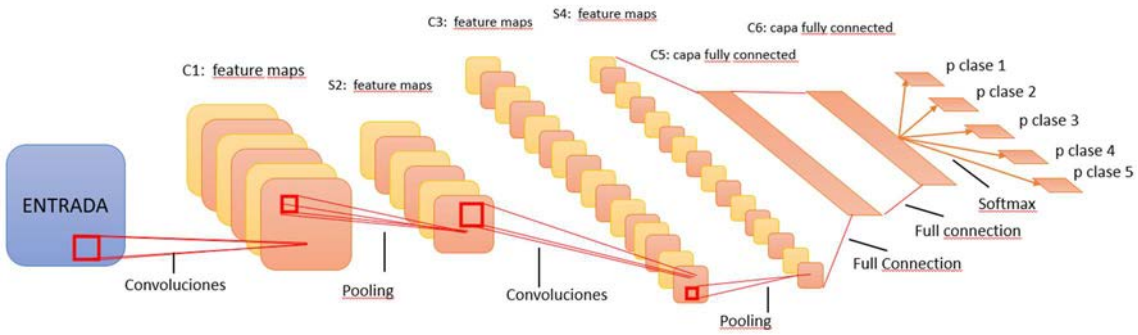


Figura 38: representación de una red neuronal convolucional empleada como clasificador

4.2.4.5.- Tensor

Cuando se ha explicado anteriormente qué es una convolución, se ha hablado de matrices. En realidad, el término que se debe utilizar es el de tensor. Un tensor [161] es una estructura de datos matemática que generaliza las nociones de escalares, vectores y matrices en un objeto multidimensional. Es una tabla de números organizada en una cuadrícula que puede tener múltiples dimensiones o ejes. Los tensores se pueden clasificar según su número de dimensiones o rango:

- **Escalar:** Un escalar es un único número, que se puede considerar como un tensor de rango 0.
- **Vector:** Un vector es una lista unidimensional de números, que se puede considerar como un tensor de rango 1.
- **Matriz:** Una matriz es una tabla bidimensional de números, que se puede considerar como un tensor de rango 2.
- **Tensor de rango N:** Un tensor de rango N es una estructura de datos de N dimensiones, donde N > 2.

En el contexto de la presente investigación, el tensor se utiliza para representar y manipular datos, como las entradas y salidas de una red, así como los parámetros y pesos del modelo. Por ejemplo, en el procesamiento de imágenes, una imagen en color se consideraría un tensor de rango 3, con dimensiones correspondientes a la altura, el ancho y los canales de color (generalmente rojo, verde y azul).

4.2.4.6.- Cuestiones relativas a un *dataset* empleado en una CNN:

Este apartado en cuestión es uno de los más relevantes del documento, ya que tiene influencia directa a la hora de justificar la forma de abordar la investigación. Uno de los principales problemas cuando se debe entrenar un clasificador es el disponer de un *dataset* válido para llevar a cabo el entrenamiento de la red. Es difícil establecer una serie de reglas exactas, ya que a día de hoy los procedimientos para afinar el entrenamiento de una red (lo que se denomina *fine tunning*) son principalmente empíricos. Sin embargo, sí que existen determinados problemas comunes a todos ellos y que se pueden recoger a través de diferentes investigaciones.

El principal problema a la hora de entrenar una red y conseguir resultados adecuados es disponer de una cantidad suficiente de datos [162]. Por norma general, la falta de datos suficientes tiende a provocar la obtención de métricas de precisión deficientes. La cantidad absoluta de datos necesarios para entrenar el clasificador va a depender del número de clases totales y también de la diferencia que exista entre esas clases [141]. No es lo mismo entrenar un clasificador para distinguir por ejemplo entre personas y vehículos, que entre monovolúmenes y turismos. Aunque en ambos casos se trata de un clasificador binaria, es probable que el volumen total de datos necesario en el segundo caso sea muy superior, ya que el clasificador debe extraer características que resultan bastante más parecidos que en el primer caso [163,164].

Otro problema también bastante reseñable es que no exista una cantidad balanceada de muestras de cada clase, ya que puede provocar sesgos en el clasificador [165]. Este problema es más habitual en clasificadores con multitud de clases y en las que algunas resulten parecidas entre sí. Por ejemplo, si se utiliza para que un vehículo autónomo reconozca elementos de la vía como diversos tipos de vehículos (camiones y turismos), peatones y señales de tráfico. En este caso concreto se está introduciendo además dos elementos parejos entre sí con otros dos muy diferentes, lo cual provoca una de las casuísticas más complicadas para trabajar.

También existe el problema contrario, que es una excesiva variabilidad y complejidad de los datos [141]. En este caso, el clasificador puede tener dificultades para aprender las características distintivas de cada clase, lo que puede requerir una arquitectura de modelo más compleja y más tiempo de entrenamiento.

Y por último, aunque ya se ha comentado en el apartado anterior, debe existir una división proporcionada entre los datos de entrenamiento y los de validación. Si la división entre los conjuntos de entrenamiento y validación no es representativa o no mantiene la distribución de clases, los resultados de la validación pueden ser engañosos y no reflejar el rendimiento real del clasificador [166].

Por lo tanto, cuando se emplea una CNN, dependiendo del tipo de problema y la forma de abordarse, puede resultar recomendable confeccionar un *dataset* que se ajuste lo máximo posible al entorno de trabajo. Este paradigma es precisamente el que se ha seguido en la investigación, tal y como se explicará más adelante.

4.2.4.7.- Interpretación de los resultados del modelo:

Las métricas proporcionan información sobre la capacidad del modelo para clasificar correctamente las imágenes en el conjunto de entrenamiento, validación y prueba. En concreto, las más utilizadas normalmente son la precisión (*precission*), la exactitud (*accuracy*), la exhaustividad (*recall*) y la precisión media promedio (*mean average precission* o mAP), tal como se ha podido recopilar en la Tabla 1: ejemplos de soluciones actuales de reconocimiento de matrículas y Tabla 3: ejemplos de soluciones actuales de re - identificación de vehículos correspondientes al estado del arte tanto de lectura de matrículas como de re – identificación de vehículos.

Precisión (*precission):*

Con la métrica de precisión se analiza la calidad del modelo en tareas de clasificación.

Para calcular la precisión usaremos la siguiente fórmula:

$$precission = \frac{TP}{TP + FP}$$

Ecuación 1: cálculo *precission*

Donde TP son los positivos totales y FP son los falsos positivos.

Exactitud (*accuracy*):

La exactitud mide el porcentaje de aciertos del modelo. Se calcula de la siguiente manera:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Ecuación 2 cálculo *accuracy*

Donde TP son los positivos totales, TN los negativos totales, FP son los falsos positivos y FN los falsos negativos.

Es una medida que puede ser engañosa en situaciones en las que las clases están desequilibradas. Por ejemplo, si el 95% de los datos pertenecen a una clase específica, un modelo que siempre predice esa clase tendrá una precisión del 95%, aunque no haya aprendido a distinguir las clases correctamente.

Exhaustividad (*recall*):

La exhaustividad (también conocida como recall o tasa de verdaderos positivos) es una métrica que mide la proporción de verdaderos positivos (TP) con respecto al número total de casos positivos reales (la suma de verdaderos positivos y de falsos negativos o FN):

$$recall = \frac{TP}{TP + FN}$$

Ecuación 3 cálculo *recall*

Se centra en la capacidad del modelo para identificar correctamente los casos positivos, sin tener en cuenta los falsos positivos (FP). Por lo tanto, es especialmente útil en situaciones donde es importante minimizar los falsos negativos, como por ejemplo puede interesar en el escenario descrito en el presente documento.

Precisión media promedio (*mean average precission* o mAP):

Es una combinación entre *precission* y *recall* para proporcionar una puntuación única y completa que mide el rendimiento de un modelo. La precisión es la fracción de predicciones verdaderamente positivas entre todas las predicciones positivas realizadas por el modelo, mientras que el *recall* es la fracción de predicciones verdaderamente positivas entre todas las instancias positivas reales en el conjunto de datos.

Para calcular la mAP, primero se calcula la Precisión Promedio (AP, por sus siglas en inglés) para cada clase en un entorno de múltiples clases. La AP resume la curva de *precission - recall* calculando el promedio de las precisiones en diferentes niveles de *recall*,

generalmente de 0 a 1. Luego, se calcula la media de estos valores de Precisión Promedio en todas las clases, lo que resulta en la Precisión Media Promedio (mAP).

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i$$

Ecuación 4: cálculo de mAP

Donde N es el número total de clases de objetos, y AP_i es la precisión promedio para la clase de objeto i.

La métrica mAP ayuda a comparar diferentes modelos al proporcionar un solo número que refleja tanto la precisión de localización como de clasificación de un modelo. Una mAP más alta indica un mejor rendimiento en la detección y clasificación de objetos en diferentes clases.

Además del uso de las métricas, en ocasiones se hace una visualización de las activaciones y filtros [167] en las capas convolucionales pueden ofrecer información sobre qué características ha aprendido la CNN para representar y clasificar las imágenes. Esto puede ser útil para comprender cómo el modelo está realizando las predicciones e identificar posibles problemas o mejoras en la arquitectura.

Una vez obtenida la precisión del entrenamiento y sobre todo la de validación, se puede valorar si los resultados obtenidos son valorables o aceptables, o si es necesario realizar ajustes no sólo en los hiperparámetros si no también en la configuración del *dataset.* Es aquí donde las técnicas de regularización anteriormente reseñadas juegan un papel muy importante.

Hay una técnica que no se ha mencionado previamente, y que es muy importante, sobre todo en clasificadores. Se trata del *transfer learning*. El aprendizaje por transferencia o *transfer learning* es una técnica empleada en el entrenamiento de las redes que aprovecha modelos previamente entrenados en una tarea específica para mejorar el rendimiento en otra tarea relacionada pero diferente.

La finalidad es reducir notablemente la cantidad de tiempo y recursos necesarios para el entrenamiento y mejorar la precisión del modelo en la nueva tarea.

El *transfer learning* es especialmente útil en situaciones donde se dispone de pocos datos etiquetados para la nueva tarea o cuando el entrenamiento de un modelo desde

cero sería computacionalmente costoso. En lugar de aprender todas las características y parámetros del modelo desde cero, el *transfer learning* permite a los modelos reutilizar las características y parámetros aprendidos en tareas relacionadas como punto de partida.

Se utiliza comúnmente en aplicaciones de aprendizaje profundo, especialmente en tareas de procesamiento de imágenes [168] y lenguaje natural [169]. Por ejemplo, en el reconocimiento y clasificación de imágenes, es común utilizar modelos previamente entrenados en grandes conjuntos de datos, como ImageNet [170], y ajustarlos para tareas más específicas.

4.2.4.8.- Potencia computacional de procesamiento

Uno de los principales planteamientos a la hora de ejecutar el entrenamiento de la red es la capacidad de computación requerida, que va a afectar al tiempo empleado y sobre todo, al hardware necesario.

Como se mencionó con anterioridad, el avance masivo en la proliferación del empleo de herramientas basadas en redes neuronales ha sido el aumento de la potencia de computación disponible y, sobre todo, de las GPU (*graphic processing units* o unidades de procesamiento gráfico) [171,172]. Si bien están diseñados originalmente para procesar vídeos (descargando fuerza de trabajo al microprocesador), son también muy eficientes en la realización de cálculos matriciales y aritmética de coma flotante, fundamentales en el entrenamiento de redes neuronales.

Las principales virtudes que ofrecen las GPU son la paralelización de procesos y la escalabilidad. La paralelización de procesos es una técnica en la que se dividen las tareas computacionales en subprocesos más pequeños que se ejecutan simultáneamente. Por ejemplo, se pueden paralelizar los datos de entrada de la red, los modelos a entrenar (modificando o paralizando determinadas capas intermedias) o los hiperparámetros. Todas estas estrategias, sumado a la posibilidad de utilizar varias GPU de forma simultánea, permite jugar con la distribución de tiempos y de datos a manejar, permitiendo precisamente esa escalabilidad.

Además, existen diferentes bibliotecas de programación y de *frameworks* como TensorFlow [173], PyTorch [174] y cuDNN [175], que están diseñadas y optimizadas específicamente para funcionar con GPU, aprovechando al máximo sus capacidades.

Sin embargo, dependiendo de la profundidad de la red, de los hiperparámetros elegidos y sobre todo del tamaño de los datos del *dataset* y del *batch size*, salvo que se disponga de estaciones de trabajo muy potentes, va a haber ocasiones en las que el entrenamiento de la red va a ser muy lento o incluso imposible de llevar a cabo (ya que se fagocitan todos los recursos disponibles). Además, que estos problemas (aunque en menor medida) pueden suceder cuando se esté ejecutando el algoritmo en tiempo real.

Por lo tanto, tan importante es elegir una configuración de red adecuada como analizar los resultados obtenidos, ya que habrá en ocasiones en que la mejora de los datos de validación va a ser muy pequeña en contraste a los recursos necesarios. Este factor en concreto se va a apreciar también en el capítulo siguiente.

4.3.- Conclusiones parciales

Este capítulo se ha centrado en recoger conceptos teóricos que resultan importantes para situar de manera concreta la investigación realizada y contextualizar qué ha impulsado a dirigirla tal y cómo se ha llevado a cabo. De manera que se pueden establecer las siguientes conclusiones parciales:

- La importancia y versatilidad del empleo de técnicas de *deep learning*, y sobre todo de redes neuronales convolucionales para resolver problemas enfocados bajo el prisma de la visión artificial.
- Una vez elegida la configuración del algoritmo, establecer cuáles van a ser los hiperparámetros elegidos para realizar el entrenamiento.
- Realizar una correcta elección y configuración del *dataset* que se ajuste lo máximo posible al problema real que se pretende abordar.
- Llevar a cabo una interpretación correcta de los resultados, atendiendo a las métricas adecuadas.
- En caso de que los resultados no sean satisfactorios, establecer si el problema se debe al *dataset*, a la configuración de los hiperparámetros o al modelo en sí (falta de profundidad), para reorientar el problema.
- Tener en cuenta el equilibrio de coste/rendimiento en función de los resultados obtenidos, ya que en este caso tan importante es conseguir métricas adecuadas como que el sistema pueda operar en tiempo real.

SISTEMA DE IDENTIFICACIÓN DE DOBLE FACTOR

V

Tras haber definido tanto el marco teórico como cuáles van a ser los entornos de uso del sistema, este capítulo se va a centrar en desgranar el modelo propuesto para llevar a cabo la identificación de vehículos.

En el Capítulo II se recopilaron y mostraron, entre otros, diferentes aproximaciones para la identificación de vehículos basadas en el procesamiento inteligente de imágenes. Y en el Capítulo III, se llevó a cabo un proceso análogo sobre la influencia de los elementos que forman un sistema de videovigilancia y los factores operativos que se deben considerar. Como resultado, se dedujo que uno de los elementos con más posibilidades para incrementar las capacidades es el sistema de procesamiento de señal. En particular, mediante el desarrollo de una herramienta de analítica de vídeo.

Tras evaluar de forma conjunta el estado del arte y las características de un sistema de videovigilancia, se ha considerado que una solución actual debería apoyarse en herramientas de inteligencia artificial, y más concretamente, en algoritmos de *deep learning*. De ahí precisamente la relevancia del Capítulo IV, en el que además se han destacado términos y consideraciones que tienen una influencia directa en la presente investigación.

A lo largo de todo el documento se ha hecho hincapié en la necesidad de que la herramienta fuese versátil y robusta, permitiendo llevar a cabo la identificación de vehículos en el mayor número de escenarios posibles. La existencia de diferentes escenarios de operación provoca una disminución de los resultados que se obtendrían en entornos controlados, independientemente de la evolución de la técnica (tal y como se ha reflejado en el estado del arte). Si además se tiene en consideración que en última instancia se trata de una herramienta de seguridad, y que por lo tanto debe ofrecer las máximas garantías posibles en la realización de las funciones encomendadas, se ha perseguido que la identificación no tuviese un único punto de apoyo, si no que ofreciese la posibilidad de aprovechar dos elementos de identificación diferentes, tal y como se va a mostrar a continuación.

5.1.- Arquitectura del modelo

Teniendo en cuenta el marco teórico, el modelo diseñado para realizar la identificación de vehículos es un sistema multi red en cascada que consta de varias etapas. La primera etapa, es un clasificador que detecta la presencia de vehículos. A partir de ahí, el sistema se divide en dos ramas. Una rama es responsable de realizar las funciones de ALPR. Tomando las referencias del Capítulo II, se trata de un sistema multifase, que consta de un algoritmo de detección de las matrículas y otro que opera como OCR (es decir, se omite la parte de pre procesamiento de las imágenes, la etapa intermedia). La otra rama se encarga de realizar el reconocimiento de vehículos basado en sus características visuales, mediante el empleo de la re – identificación.

Los algoritmos utilizados para la detección inicial de los vehículos y para las funciones de ALPR están basados en YOLO v5 [176]; en cambio, la re – identificación se realiza mediante otro algoritmo denominado FastReID [102].

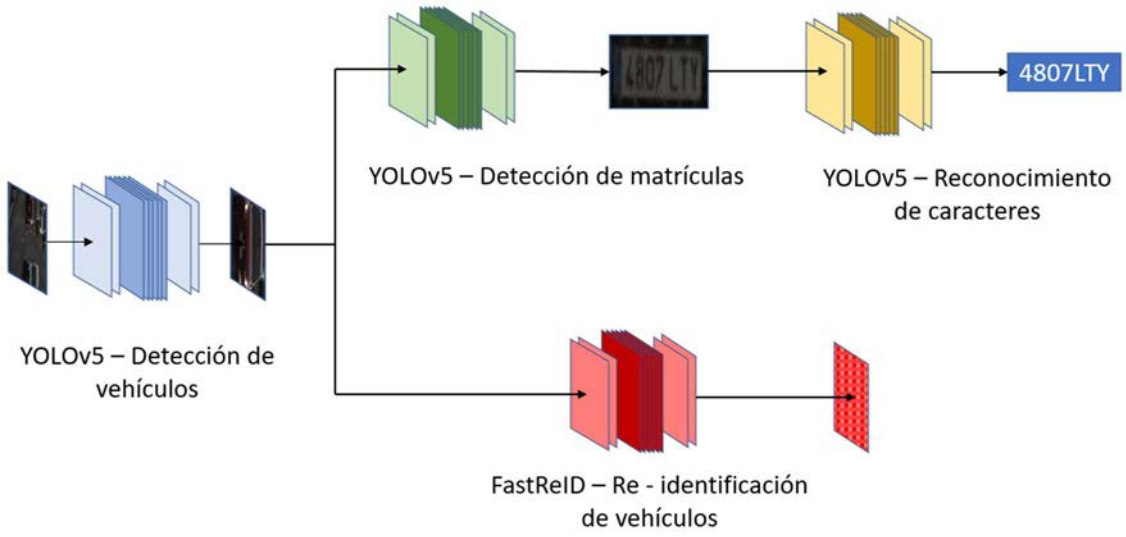


Figura 39: arquitectura de la herramienta

El capítulo se estructura de la siguiente manera. En un primer bloque se explicará la rama superior del sistema; es decir, la parte de detección de vehículos y el ALPR. Ambas partes se basan principalmente en YOLOv5, por lo que primero se pretende aprovechar los conceptos del Capítulo IV para describir su funcionamiento. A continuación, se desgranará el *dataset* creado exprofeso para el entrenamiento del ALPR. Posteriormente, se aportarán las técnicas empleadas durante el entrenamiento y las métricas obtenidas.

De igual manera con la rama correspondiente a la re – identificación de vehículos, se detallará el funcionamiento de FastReID así como otros dos *datasets* creados en este caso para realizar su testeo emulado entornos operacionales reales.

Por último, se desgranará cómo se lleva a cabo la implementación conjunta de la solución.

5.2.- YOLO v5

Gran parte del desarrollo de la investigación se ha apoyado en este algoritmo. Para motivar la elección de este sistema, primero se debe explicar exactamente en qué consiste [177].

YOLO v5 es la quinta versión del algoritmo YOLO (You Only Look Once) [52] un clasificador de objetos en tiempo real basado en una red neuronal convolucional. La principal revolución respecto a otros clasificadores de objetos previos fue que, mientras que estos otros algoritmos dividían una imagen en regiones y luego detectaban objetos en cada región [178], YOLO utiliza una sola red neuronal que descompone la imagen en regiones y establece las probabilidades de predicción de cada clase de objeto en cada región. YOLOv5 es una versión avanzada mucho más rápida en realizar la inferencia que el YOLO original.

5.2.1.- Arquitectura de YOLOv5

Un detector de objetos de una sola etapa consta de la siguiente estructura:

- Un modelo *backbone* que extrae las características de las imágenes y reduce la resolución espacial y aumenta los canales.
- Un "cuello" que se utiliza para extraer "pirámides de características". Con esto se consigue incrementar la generalización del modelo frente a objetos de distintos tamaños y escalas.
- Una fase final o cabecera, que es la parte superior de la red o *head of the network* que crea los *bounding boxes* o cuadros delimitadores a los objetos detectados y que además indica a qué clase pertenecen y con qué probabilidad.

A la hora de trabajar, YOLOv5 ofrece cinco configuraciones diferentes de red y tres tamaños de imagen de entrada (320 x 320, 640 x 640 y 1280 x 1280 píxeles).

Las principales diferencias entre los modelos radican sobre todo en el número de capas intermedias totales y en el número de parámetros, como se indica en el cuadro siguiente:

Modelo	Tamaño	$mAP^{val}_{0.5:0.95}$	$mAP^{val}_{0.5}$	Velocidad CPU (ms)	Velocidad GPU (Tesla V100)	Parámetros (en miles)
YOLOv5n	640	28,0	45,7	45	6,3	1,9
YOLOv5s	640	37,4	56,8	98	6,4	7,2
YOLOv5m	640	45,4	64,1	224	8,2	21,2
YOLOv5l	640	49,0	67,3	430	10,1	46,5
YOLOv5x	640	50,7	68,9	766	12,1	86,7

Tabla 6: datos oficiales de YOLOv5 [176]

La arquitectura global de la red es:

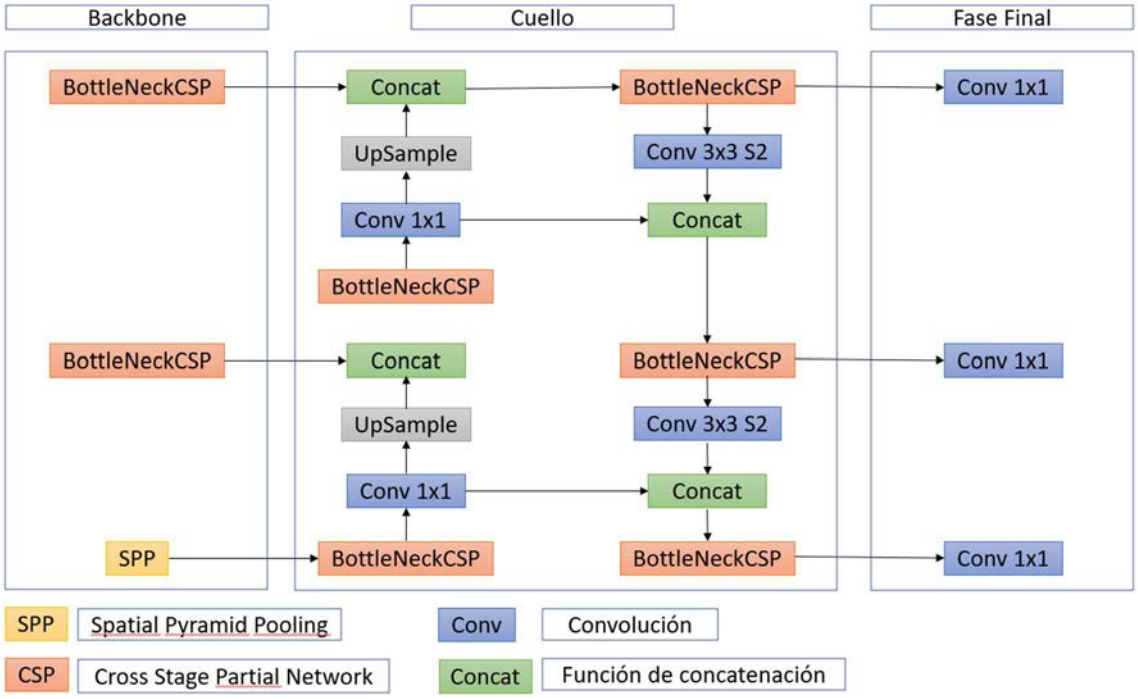


Figura 40: arquitectura de YOLOv5 [179]

5.2.1.1.- Backbone

YOLOv5 utiliza una arquitectura de red neuronal como *backbone* denominada CSP-Darknet 53. Se trata de la red convolucional Darknet53 (empleada previamente como columna vertebral de YOLOv3 [180]) a la que se aplica adicionalmente la red CSP (*Cross*

Stage Partial) [181], para mejorar la eficiencia y el rendimiento del modelo. La idea principal detrás de CSP es dividir la red en dos partes y que cada parte aprenda características de forma independiente. Finalmente, se combinan las características aprendidas por ambas partes para obtener una "representación" de los datos de entrada más variada y diversa.

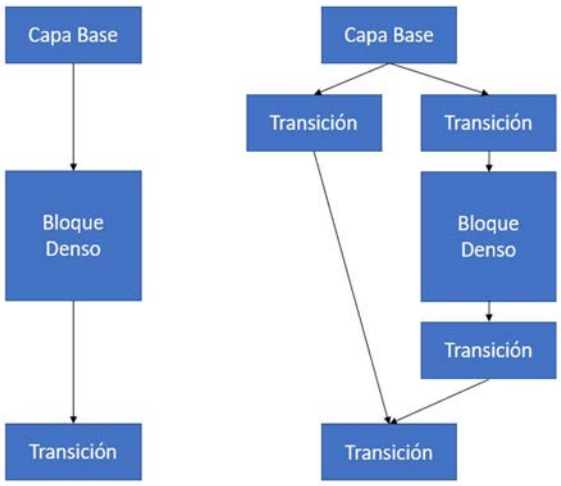


Figura 41: red densa convencional y red densa con CSP [181]

De tal manera, que en YOLOv5 se aplica de la siguiente forma en los denominados BottleNeckCSP:

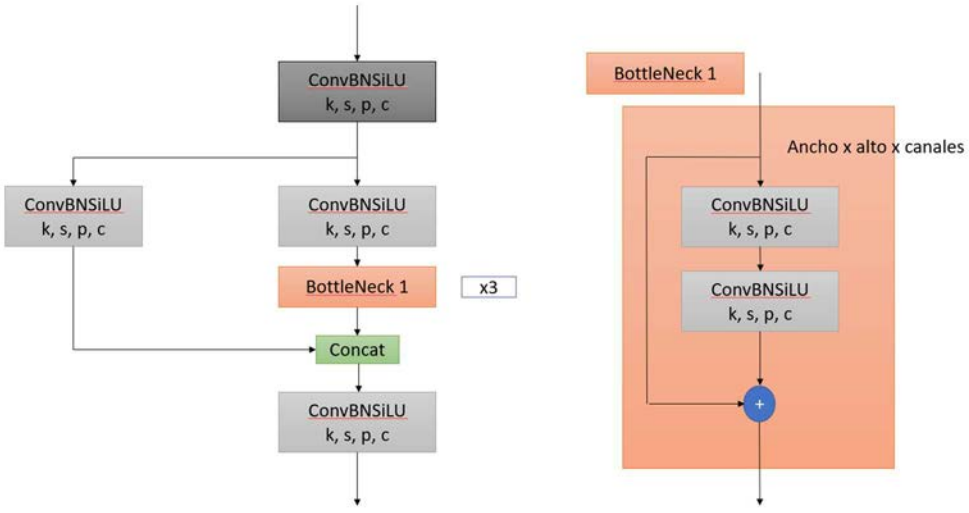


Figura 42: arquitectura BottleNeckCSP

La utilización de CSPNet ayuda a reducir el número de parámetros que va arrastrando la red en cada paso, disminuyendo la potencia computacional necesaria para ejecutar el algoritmo y la velocidad de procesamiento, lo que aumenta la velocidad de inferencia (parámetro crucial para la detección de objetos en tiempo real).

5.2.1.2.- Extractor de pirámides de características

Los principales avances en este apartado son la utilización de la *Spatial Pyramid Pooling* (SPP) [182], y una modificación de *Path Aggregation Network* (PANet) [183] incorporando el BottleNeckCSP en su arquitectura.

La idea central detrás de PANet es mejorar la extracción de características en diferentes niveles de la red neuronal, utilizando una estructura de conexión ascendente y descendente para permitir una comunicación más eficiente entre las capas de diferentes niveles de resolución en la red. Esto se traduce en una mejor capacidad para capturar información y detalles más precisos del contexto de la imagen.

Por otro lado, el bloque SPP consta de varias capas de agrupamiento (*pooling*) con diferentes tamaños de ventana y pasos (*strides*), lo que permite capturar información contextual en diferentes escalas. Después de aplicar las capas de agrupamiento, las características extraídas se concatenan y se pasan a través de una capa convolucional adicional para fusionarlas en una única representación de características.

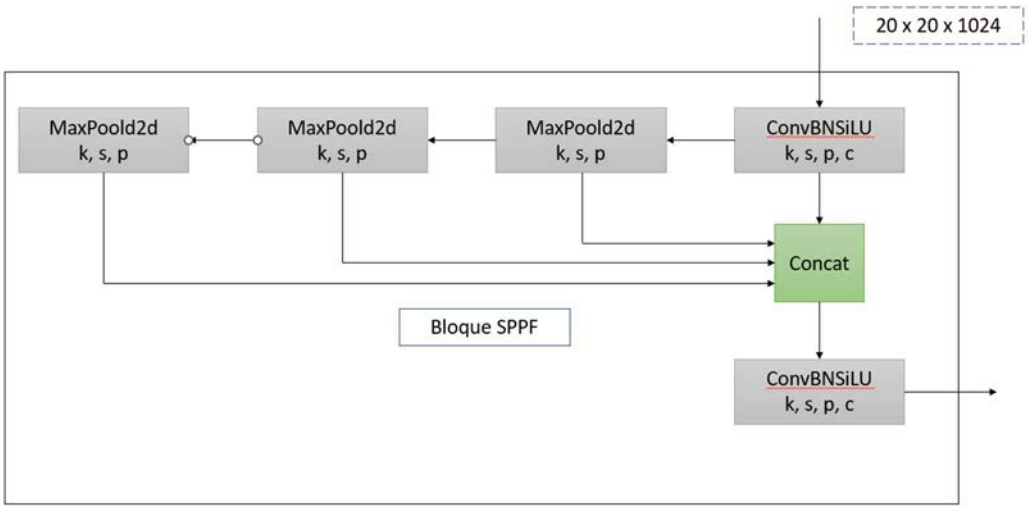


Figura 43: estructura del bloque SPPF [176]

5.2.1.3.- Parte superior

Se compone de tres capas de convolución que predicen la ubicación de los *bounding boxes* (x, y, altura, anchura), las puntuaciones y las clases de objetos. La red YOLOv5 determina los siguiente parámetros para cada caja delimitadora:

- Coordenadas del centro del *bounding box* (t_x, t_y)
- Dimensiones del *bounding box* (t_w, t_h)
- Probabilidad de objeto (t_{obj})
- Probabilidades de clase condicionales (t_{c1}, t_{c2}, ..., t_{cn}), donde N es el número de clases

Las predicciones de los *bounding boxes* se calculan a partir de las características aprendidas por la red neuronal. Para obtener las coordenadas y dimensiones finales, se realizan las siguientes operaciones:

Coordenadas del centro del *bounding box*:

$$b_x = (2\sigma(t_x) - 0.5) + c_x$$

$$b_y = (2\sigma(t_x) - 0.5) + c_y$$

Ecuación 5: cálculo de coordenadas del centro del *bounding box*

Donde b_x y b_y son las coordenadas del centro de la caja delimitadora en la imagen, c_x y c_y son las coordenadas de la celda de la cuadrícula en la que se encuentra la caja.

Dimensiones del *bounding box*:

$$b_w = p_w\left(2\sigma(t_w)\right)^2$$

$$b_h = p_h\left(2\sigma(t_h)\right)^2$$

Ecuación 6: ecuación del cálculo de las dimensiones del *bounding box*

Donde b_wy b_h son el ancho y alto de la caja delimitadora, respectivamente, y p_w y p_h son las dimensiones del *anchor box* (caja ancla) correspondiente.

Probabilidad de objeto:

$$obj = \sigma(t_{obj})$$

Ecuación 7: cálculo de la probabilidad de que exista un objeto en la cuadrícula

Donde obj es la probabilidad de que haya un objeto en la cuadrícula.

Probabilidades de clase condicionales:

$$ci = \sigma(t_{ci})$$

Ecuación 8: cálculo de la probabilidad de clase

Donde ci es la probabilidad condicional de que la cuadrícula contenga la clase i, y t_{ci} es la salida de la red neuronal para esa clase.

Una vez que se han calculado las coordenadas, dimensiones y probabilidades, se utilizan métodos de post-procesamiento como la supresión no máxima (NMS) para eliminar cajas delimitadoras superpuestas y seleccionar las detecciones finales.

5.2.2.- Funciones de activación

YOLOv5 utiliza dos funciones de activación. Para las convoluciones de las capas ocultas, emplea la función SiLU (*Sigmoid Linear Unit*) [184]. Se define como:

$$SiLU\ (x) = x\sigma(x)$$

Ecuación 9: función SiLU

Por otro lado, la función de activación *Sigmoid* o sigmoide se utiliza en las operaciones de convolución en la capa de salida.

5.2.3.- Función de pérdida

A la salida de YOLOv5 se presentan tres resultados: las clases de los objetos detectados, sus *bounding boxes* y la puntuación (probabilidad) de cada objeto. Así, utiliza por un lado BCE (*Binary Cross Entropy*) para calcular la pérdida relacionada con la ubicación en el plano y las dimensiones de los *bounding boxes*. Por otro lado, se emplea la pérdida CIoU (*Complete Intersection over Union*) para calcular la pérdida relacionada con la distancia entre los centros del *bounding box* real (el del etiquetado de los objetos) y el que predice YOLOv5. La fórmula completa de la función de pérdida viene dada por la siguiente ecuación:

$$Loss = \lambda_1 L_{class} + \lambda_2 L_{obj} + \lambda_3 L_{loc}$$

Ecuación 10: función de pérdida de YOLOv5

Donde L_{class} es la pérdida relativa a la clasificación correcta de la categoría del objeto detectado en el *bounding box*, L_{obj} es la pérdida relacionada con la existencia o no de un objeto en el *bounding box* detectado y L_{loc} es la pérdida respecto a la ubicación del objeto predicho en la imagen. Por otro lado, λ_1, λ_2 y λ_3 son hiperparámetros que ponderan la relevancia de cada una de las pérdidas a la hora de configurar el aprendizaje de la red.

5.3.- Spanish ALPR Dataset (SAD)

Tanto en el Capítulo II como en el Capítulo IV se referenció el concepto de *dataset* y su crucial importancia a la hora de conseguir resultados óptimos y que concuerden con los objetivos perseguidos.

En el apartado 2.1.1.4 se reseñaron diferentes *datasets* que, si bien podrían resultar válidos a la hora de entrenar un detector de matrículas (que no para la parte de OCR), debido a las diferencias de tamaño y color con respecto a las matrículas europeas no sirven para recrear un escenario que se ajuste al objeto de estudio.

Figura 44: comparativa entre matrículas europea, china y pakistaní

Aunque no se ha mencionado durante la introducción, la finalidad última de poder implantar la herramienta desarrollada es su posible uso en el territorio nacional. Con lo cual, se hacía imprescindible disponer de un *dataset* de matrículas al menos europeas (ya que por normativa [17] presentan patrones comunes como mínimo respecto a las dimensiones). Como aparece indicado en el apartado 2.1.1.4, únicamente se pudieron localizar dos repositorios, de los cuáles, uno de ellos no contenía suficientes matrículas disponibles [80] y al otro [81], no se pudo tener acceso. Por lo tanto, se estableció la necesidad de crear un *dataset* propio que además cumpliese con los requisitos mencionados tanto en el apartado 2.1.1.4 como en el apartado 4.2.3.2. De aquí, nace "Spanish ALPR Dataset".

5.3.1.- Características

Para cumplir con las premisas de obtener imágenes con diferentes resoluciones, tamaños, distancia y grados de luminosidad, se emplearon tres captadores diferentes en todo tipo de situaciones y condiciones de luminosidad.

El *dataset* consta de dos partes. La primera está concebida para entrenar el sistema en la detección de matrículas y la segunda para la lectura de caracteres (es decir, la parte de OCR).

Figura 45: ejemplos de imágenes obtenidas

5.3.1.1.- Detección de matrículas

En total, el *dataset* consta de 2,521 matrículas extraídas de 1,977 imágenes de diferentes resoluciones, de las cuales 2,161 son diurnas (D) y 360 nocturnas (D), con diferentes variedades de vehículos tal y como recoge la normativa correspondiente [17]:

Figura 46: tipos de matrículas españolas

• Tipo A: 2475 muestras del tipo más común, largo, una fila con fondo

• Tipo B: 28 muestras de motocicletas de doble fila con fondo blanco

• Tipo C: 1 muestra de motocicleta ligera de una fila con fondo amarillo

• Tipo D: 1 muestra de motocicleta ligera de triple fila con fondo amarillo

• Tipo E: 11 muestras de taxis y VTC (acrónimo en español de vehículo de alquiler con conductor) con fondo azul

• Tipo F: 6 muestras de remolques con caracteres negros y fondo rojo

El etiquetado de las matrículas se realizó de manera manual, de tal manera que en un primer paso se delimita un polígono que contenga el borde de la matrícula lo más preciso posible. A continuación, se llevó a cabo una rectificación de la matrícula para confeccionar la imagen de entrada de la red, a la cual se asociaron las medidas relativas a los píxeles correspondientes al centro del recuadro o *bounding box* y sus dimensiones (centro eje x, centro eje y, tamaño eje x, tamaño eje y).

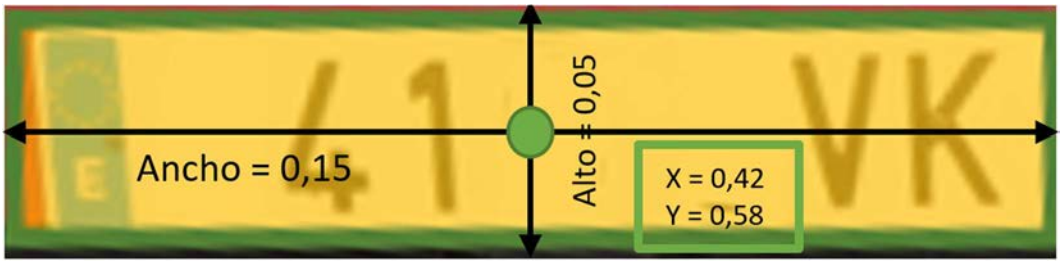


Figura 47: ejemplo del etiquetado de una matrícula en la imagen

5.3.1.2.- OCR

Para la lectura de los caracteres de la matrícula, se creó un segundo etiquetado más específico. Primero se recortaron las imágenes originales utilizando como referencia los datos del etiquetado anterior, de tal manera que la nueva imagen es únicamente una placa de matrícula.

En esa nueva imagen se empleó como etiqueta un cuadro delimitador ortogonal (como se aprecia en la siguiente imagen), extrayendo de las 2,521 matrículas 12,620 nuevas etiquetas diferentes que contienen las 37 letras y números posibles (letras de la A a la Z y números del 0 al 9), lo que permite el entrenamiento de modelos para la identificación efectiva de matrículas.

Figura 48: detalle del etiquetado visual de los caracteres de la matrícula

En la siguiente figura se puede apreciar la distribución de datos de estos caracteres según su frecuencia de aparición. Es importante reseñar que el principal escollo es la presencia de vocales en las matrículas, ya que desaparecieron a partir de la modificación de la legislación a nivel europeo. Hay que recordar como originalmente sí que se incluían vocales en las matrículas, tanto para indicar la provincia de matriculación del vehículo como en las dos letras que acompañaban a las cuatro cifras.

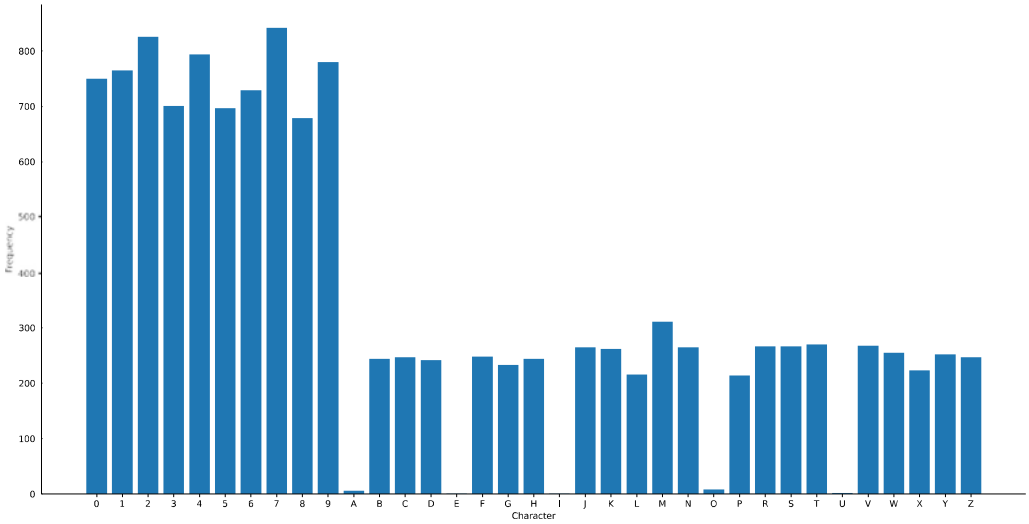


Figura 49: representación de la distribución de los diferentes caracteres

5.4.- Entrenamientos de la red (YOLOv5)

En este apartado se va a detallar cómo se ha llevado a cabo el entrenamiento de la red. Como se ha explicado anteriormente, este clasificador realiza tres detecciones en cascada. La primera, para detectar vehículos. La segunda, para las placas de matrícula. Y la tercera para los caracteres de la matrícula.

5.4.1.- Detección de vehículos

Para la detección de vehículos se ha empleado un modelo pre entrenado disponible en [176] correspondiente al modelo YOLOv5s, que además es la arquitectura utilizada como base para el resto de partes del sistema.

Este entrenamiento fue realizado con el *dataset* de dominio público COCO [185], que contiene en total 123,287 imágenes con 886,284 objetos etiquetados de 80 categorías de objetos en total. Se trata de uno de los *dataset* más importantes para el entrenamiento y evaluación de clasificadores. En el caso concreto de los vehículos, contiene en total 12,786 etiquetas.

5.4.2.- Detección de matrículas

Para el entrenamiento de la detección de matrículas, lo primero que se ha querido valorar ha sido la influencia de los formatos de "carga de datos" del *dataset*, es decir, tamaño de *batch size* y tamaño de las imágenes. Si bien a priori pudiera parecer que un *batch size* mayor permitiría acelerar el proceso de entrenamiento, es posible que se genere

el efecto contrario comprometiendo la capacidad de generalización del modelo, como se muestra a continuación.

Por ejemplo, *batch sizes* más grandes pueden dar lugar a mínimos "más planos" en la optimización de la función de pérdida (es decir, una tasa de cambio baja en los pesos), lo que puede hacer que el modelo converja a una solución subóptima con menor capacidad de generalización.

En cambio, el tamaño de las imágenes de entrada del *dataset* sí contribuye a mejorar el rendimiento del modelo como se muestra en la siguiente imagen. A priori este comportamiento tiene sentido ya que, a mayor resolución, se puede extraer más detalles de la imagen.

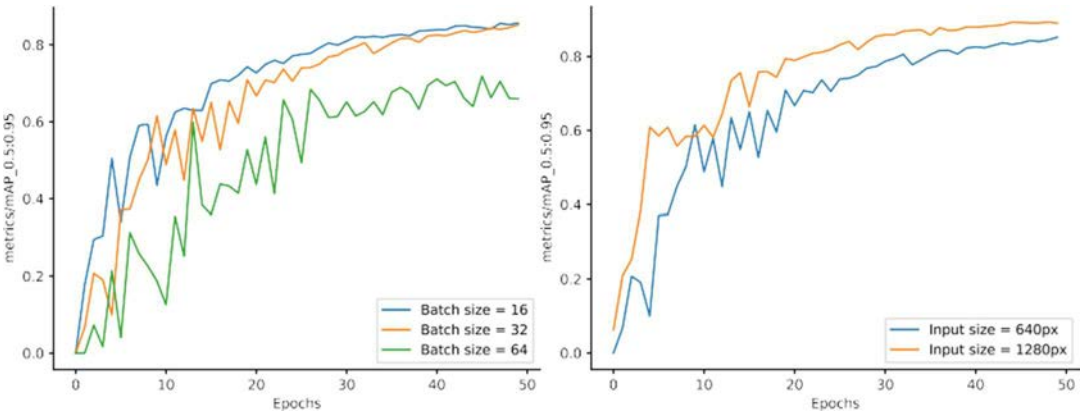


Figura 50: comparativa de la influencia en el entrenamiento del *batch size* (izquierda) y del tamaño de imagen de entrada (derecha) – detección de matrículas

La siguiente comparativa se centra en el análisis de la elección de la función de optimización. En este caso, se comparó el comportamiento durante el entrenamiento empleando la función SGD [135], Adam [185] y AdamW [187], todos ellos optimizadores ampliamente utilizados en el entrenamiento de clasificadores basados en CNNs.

SGD permite a priori actualizaciones rápidas de los pesos con menor coste computacional, ya que realiza dichas actualizaciones tomando un subconjunto de datos de entrenamiento y refresca los pesos en función de los gradientes de dicho subconjunto. Adam se trataría de una evolución de SGD al aplicar dos mecánicas, el momento (que promedia los gradientes más pesados) y la tasa de aprendizaje adaptativa (ajusta la tasa de aprendizaje a cada parámetro del modelo de manera individual). Por último, AdamW es una variante del anterior, que se diferencia porque al "desacoplar" la regularización

del decaimiento de peso de los momentos, aplicándolo directamente a los pesos, para, a priori, mejorar la generalización del modelo.

Pues bien, en este caso concreto, SGD manifiesta un rendimiento ligeramente superior que AdamW y que Adam, aunque son diferencias relativamente pequeñas. Parece que en este entrenamiento de forma específica la elección del optimizador no tiene una influencia decisiva en los resultados, probablemente debido a que el tamaño de los datos en este entrenamiento en concreto (2521 imágenes) no es tan significativo como para que se manifiesten los beneficios que ofrecen Adam y AdamW, pensados para conjuntos de datos más elevados. De hecho, si se analiza también una comparativa con modelos de YOLOv5 más grandes, como se muestra en la siguiente figura, los modelos tienden a sobre ajustarse y a ofrecer peores métricas que por ejemplo YOLOv5s (más pequeño).

Por lo anteriormente reseñado y en base a los resultados obtenidos, para la implementación definitiva se ha decidido utilizar el entrenamiento apoyado en YOLOv5s debido a su facilidad para operar en tiempo real al requerir un menor coste computacional que otros modelos más complejos.

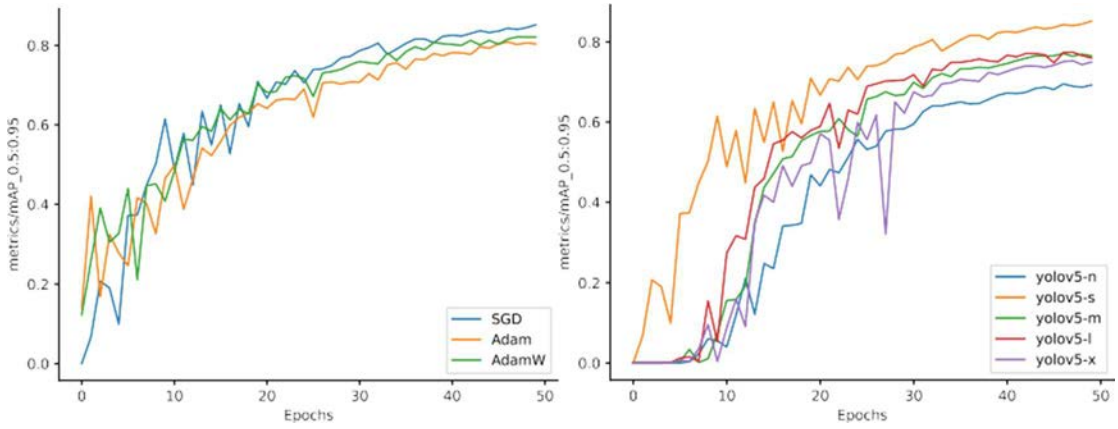


Figura 51: comparativa de la influencia de los optimizadores (izquierda) y de los diferentes modelos de red posibles (derecha) durante el entrenamiento – detección de matrículas

5.4.3.- OCR

En la tarea de OCR se aplicaron los mismos ensayos de entrenamiento que en el apartado 5.4.2. En la Figura 50, se refleja el impacto del *batch size* y del tamaño de la imagen de entrada. En este caso concreto, los efectos del *batch size* son similares al escenario anterior, aunque sobre todo se manifiestan en la velocidad de entrenamiento, ya que los resultados finales son bastante parecidos. Un tamaño de lote reducido puede hacer que el modelo converja más rápidamente, porque los gradientes se actualizan con

mayor frecuencia, lo que puede conducir a un uso más eficiente de los recursos computacionales.

En cuanto a la influencia de la resolución de la imagen de entrada, en este caso concreto parece no haber diferencias significativas. Esto podría deberse a que no muchas de las matrículas extraídas tienen dimensiones superiores a 640 píxeles, por lo que aumentar su tamaño no parece proporcionar una visión más detallada de la imagen.

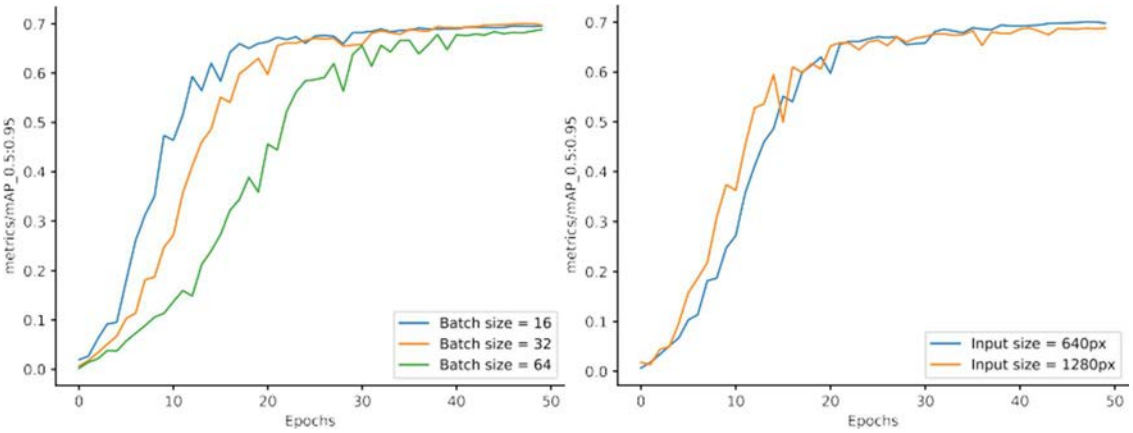


Figura 52: comparativa de la influencia en el entrenamiento del *batch size* (izquierda) y del tamaño de imagen de entrada (derecha) – OCR

Respecto a la influencia de la función de optimización, al contrario que en el apartado anterior, se puede apreciar una caída del rendimiento cuando se utiliza SGD en favor de optimizadores adaptativos. Esto puede deberse a que en este escenario existe una mayor cantidad de imágenes de entrada en el entrenamiento correspondiente al OCR, con lo cual, se manifiestan los efectos de otros optimizadores concebidos para grandes cantidades de datos, como se puede apreciar en este caso. Respecto a la comparativa de los diferentes modelos posibles de YOLOv5, aparte del ajuste insuficiente del modelo YOLOv5n (el más pequeño de todos), el resto parecen arrojar unos resultados parecidos.

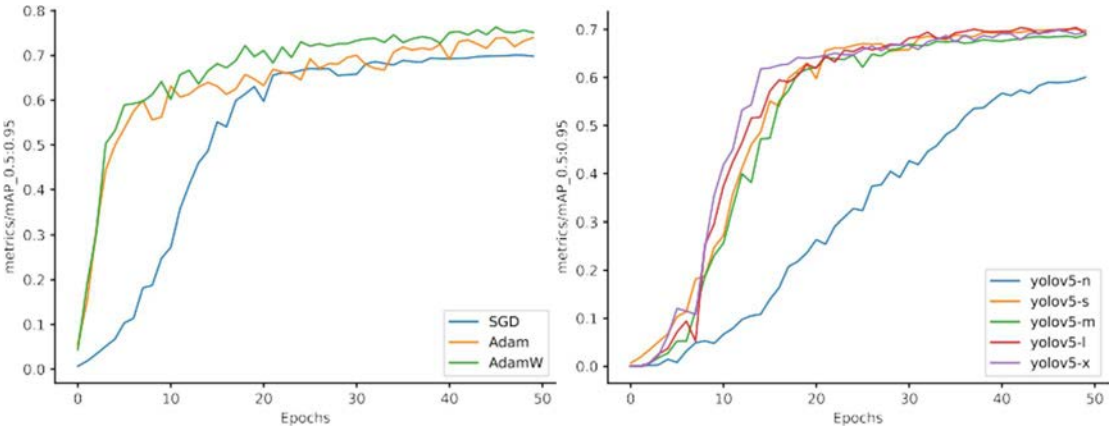


Figura 53: comparativa de la influencia de los optimizadores (izquierda) y de los diferentes modelos de red posibles (derecha) durante el entrenamiento – OCR

5.5.- Resultados obtenidos

Como consecuencia de los entrenamientos realizados, se han obtenido los siguientes resultados.

5.5.1.- Detección de vehículos

Como se ha explicado anteriormente, para la detección de vehículos se ha utilizado un modelo pre entrenado mediante el *dataset* COCO que arroja los siguientes resultados:

Modelo	Tamaño	$mAP^{val}_{0.5:0.95}$	$mAP^{val}_{0.5}$	Velocidad CPU (ms)	Velocidad GPU (Tesla V100)	Parámetros (en miles)
YOLOv5s	640	37,4	56,8	98	6,4	7,2

En este caso la métrica mAP 0.5:0.95 es un valor comúnmente utilizado para evaluar un modelo de detección de objetos en *datasets* como COCO. El mAP da una medida de la precisión del modelo en todos los umbrales de detección, como se explicó en el apartado 4.2.4.7. El valor "0.5:0.95" indica que el mAP se calcula utilizando varios niveles de i*ntersection over union* (IoU) desde 0.5 hasta 0.95 con un paso de 0.05. La IoU mide la superposición entre dos áreas; en el caso de la detección de objetos, se utiliza para determinar la precisión entre el *bounding box* real y el calculado por el modelo.

El valor de mAP de 37.4 es un valor de por sí muy bueno, si se tiene en cuenta que es un modelo muy ligero. No obstante, y aunque a continuación se va a aportar un ejemplo visual de cómo se lleva a cabo la detección de vehículos, no se llegaría a reflejar ya que se trata de un paso inicial y además no ha sido objeto de estudio en la presente investigación.

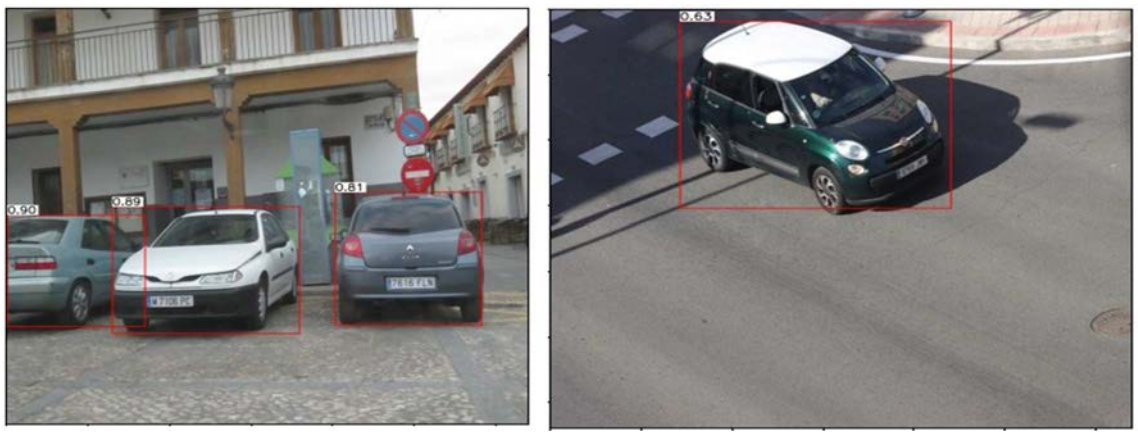


Figura 54: ejemplo de la detección de vehículos

5.5.2.- Detección de matrículas

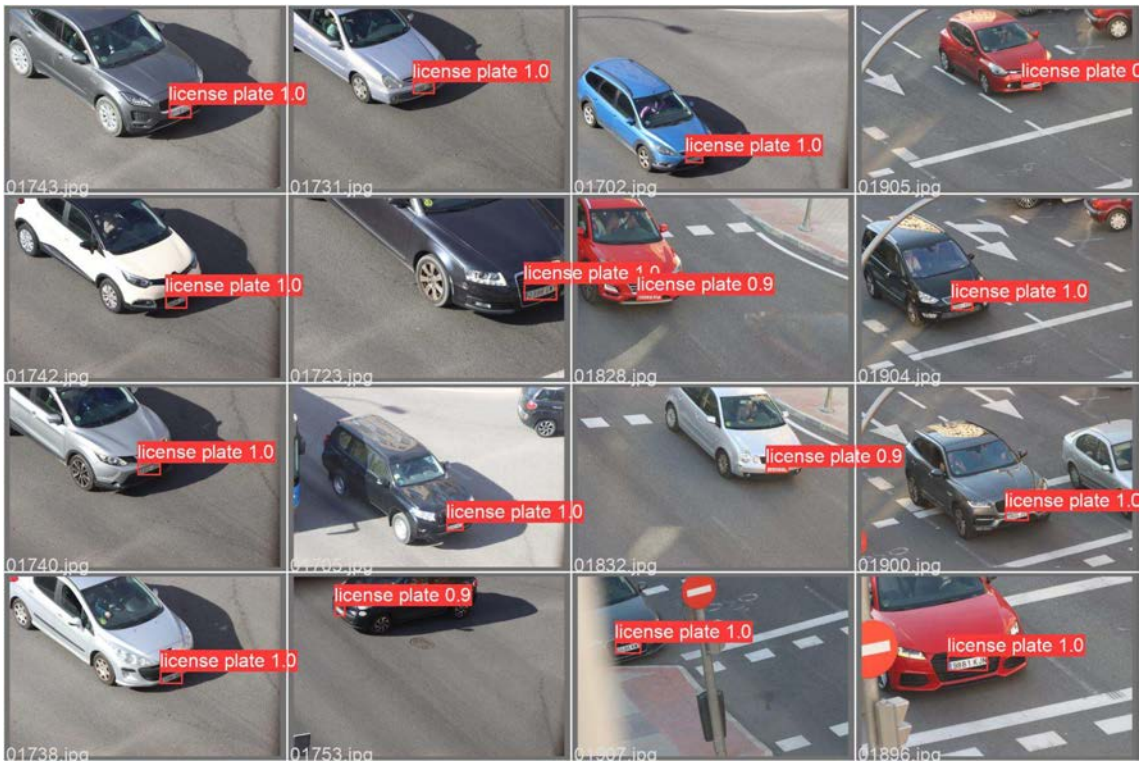
Los resultados de los modelos de detección de matrículas muestran una gama de rendimientos según las distintas configuraciones probadas. El modelo con mejor rendimiento alcanzó una precisión media (mAP) de 0,893 con un umbral de intersección sobre la unión (IoU) de 0,5 a 0,95 y una mAP de 0,988 con un IoU de 0,5. Cabe destacar que el tamaño de imagen de 1280 píxeles produjo los mejores resultados, mientras que el mayor *batch size*, el de 64 imágenes, ofreció el peor rendimiento. Esto sugiere que los lotes más pequeños pueden ser más eficaces para entrenar modelos de detección de matrículas. Curiosamente, el modelo de referencia (o *baseline*) obtuvo resultados similares a los de los modelos Adam, AdamW y Yolov5-l, lo que indica que los modelos más complejos y las estrategias de optimización pueden no conducir necesariamente a un mejor rendimiento en esta tarea. En general, estos resultados destacan la importancia de analizar cuidadosamente las configuraciones de los modelos y optimizar los hiperparámetros para lograr un rendimiento óptimo en la detección de matrículas.

La siguiente tabla muestra los resultados según las variaciones realizadas respecto al modelo estándar de YOLOv5s o baseline, que consta de un batch size de 32, optimizador SGD y un tamaño de entrada de imagen de 640 píxeles. El resto de modificaciones realizadas respecto a dicho modelo son las que se reflejan en cada fila.

Modelo	mAP@0.5:0.95	mAP@0.5	precision	recall	F1
Tamaño de imagen = 1280px	**0,893**	**0,988**	**0,961**	**0,965**	**0,963**
Batch size = 16	0,856	0,985	0,953	0,959	0,956
Baseline	0,852	0,982	0,941	**0,965**	0,953
Optimizador AdamW	0,821	0,975	0,922	0,952	0,937
Optimizador Adam	0,809	0,971	0,935	0,939	0,937
YOLOv5l	0,774	0,956	0,924	0,911	0,918
YOLOv5m	0,771	0,944	0,920	0,911	0,916
YOLOv5x	0,753	0,943	0,907	0,899	0,903
YOLOv5n	0,695	0,917	0,930	0,867	0,897
Batch size = 64	0,684	0,968	0,929	0,948	0,938

Tabla 7: comparativa de métricas del entrenamiento del módulo de detección de matrículas

A continuación se muestra gráficamente ejemplos prácticos que plasman el resultado de la detección de matrículas:



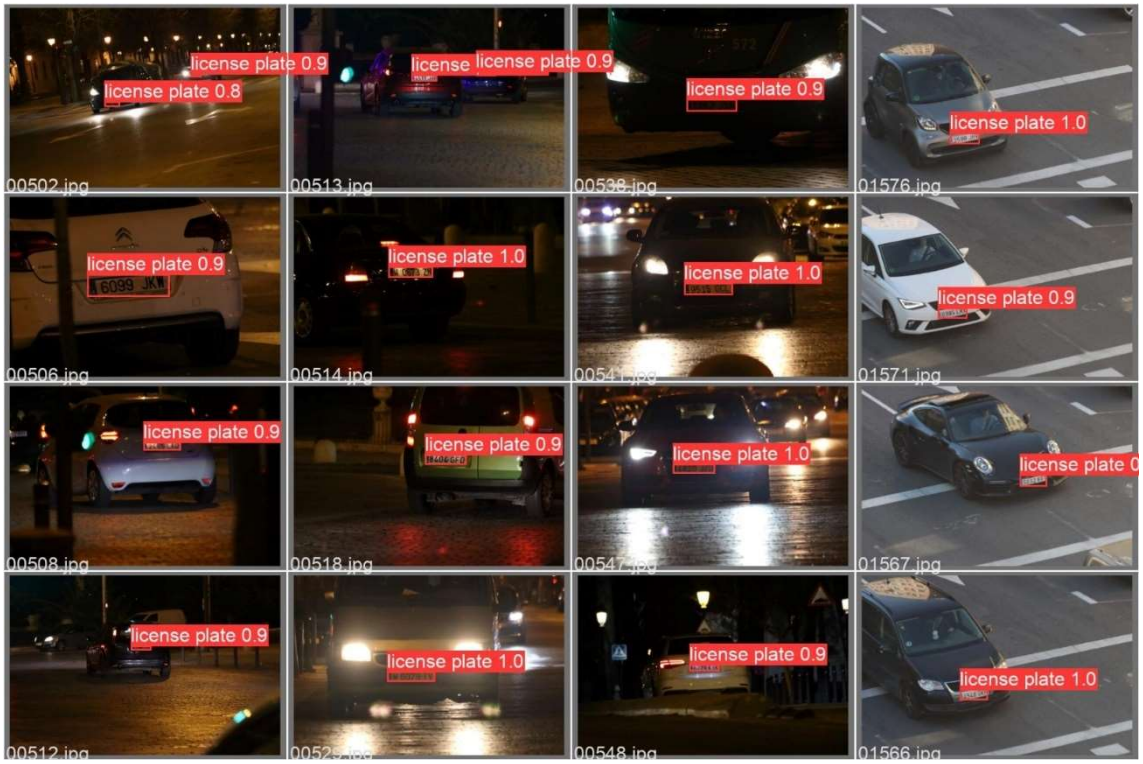


Figura 55: detalle del proceso de detección de matrículas

Como se puede apreciar, una de las principales ventajas de la herramienta es la robustez frente todo tipo de movimiento, condiciones lumínicas y sobre todo variaciones en la orientación geométrica.

5.5.3.- OCR

Los resultados del entrenamiento del OCR reflejan que el modelo entrenado con el optimizador AdamW obtuvo las mejores métricas de mAP 0,5:0,95, alcanzando una puntuación de 0,764. El modelo entrenado con el optimizador Adam también alcanzó buenos resultados, con un mAP de 0,74. En cuanto a la comparativa entre las arquitecturas de YOLOv5l, YOLOv5s y YOLOv5x se lograron puntuaciones mAP similares, con valores que oscilaron entre 0,699 y 0,704. Como puede verse en la tabla, al aumentar el *batch size* de 16 a 64 y el tamaño de la imagen a 1280 píxeles no se llegaron a producir mejoras significativas en el rendimiento, como se explicó anteriormente. En general, los resultados sugieren que los modelos con optimizadores de momento adaptativo son los más eficaces para el reconocimiento de caracteres de matrículas, probablemente causado por disponer de una mayor cantidad de datos para el entrenamiento.

Modelo	mAP@0.5:0.95	mAP@0.5	precision	recall	F1
Optimizador AdamW	**0,764**	**0,976**	0,943	**0,972**	**0,957**
Optimizador Adam	0,740	0,962	0,979	0,914	0,946
YOLOv5l	0,704	0,926	0,988	0,884	0,933
Baseline	0,701	0,911	0,988	0,889	0,936
YOLOv5x	0,699	0,922	0,989	0,885	0,934
Batch size = 16	0,696	0,904	**0,990**	0,890	0,937
Batch size = 64	0,689	0,901	0,986	0,885	0,933
Tamaño de imagen = 1280px	0,689	0,901	0,985	0,882	0,931
Yolov5-m	0,688	0,909	0,988	0,878	0,930
Yolov5-n	0,601	0,816	0,737	0,818	0,775

Tabla 8: comparativa de métricas del entrenamiento del módulo OCR

Al igual que en los casos anteriores, a continuación se aportan imágenes correspondientes a los resultados de la ejecución del modelo específicamente entrenado para la detección de caracteres. A priori los resultados son muy positivos bajo diferentes condiciones de iluminación y con resoluciones de imagen muy variadas.

Figura 56: ejemplos de la detección de caracteres

5.6.- FastReID

FastReID [102] es una red neuronal multietapa configurable, creado como un marco de investigación y con desarrollo de código abierto desarrollado para realizar tareas de re – identificación en objetos y personas. Implementado íntegramente en PyTorch, dispone de algunos modelos pre entrenados para la re – identificación de personas y también de vehículos (objetivo perseguido en la presente investigación).

5.6.1.- Arquitectura de la red

La red consta de cuatro módulos: preprocesamiento de imágenes, *backbone*, capas de agregación (*aggregation)* y cabecera. La característica más interesante de FastReID es la versatilidad para implementar diferentes configuraciones en cada una de los módulos, en función del objetivo final perseguido.

5.6.1.1.- Módulo de preprocesamiento

El primer paso es un redimensionamiento de las imágenes de entrada, donde se convierten a un tamaño de 256 x 256 píxeles. Posteriormente, se implementan técnicas de *data augmentation* para introducir otro tipo de variaciones en las imágenes que alberga el *dataset*, con la finalidad de favorecer la posterior generalización del modelo.

5.6.1.2.- Backbone

Para la obtención del *features map*, se pueden emplean tres posibles redes convolucionales como ResNet [188], ResNeXt [189] y ResNeSt [190] eliminando las capas de *average pooling.* Además, también se añade un módulo de atención no local [191] y un módulo de normalización de lotes de instancias, normalmente llamada *instant batch normalization* (IBN) [192] para ejecutar un aprendizaje de características más robustas.

El módulo de atención, permite a la red ponderar y seleccionar información relevante de todo el *dataset* en lugar de depender únicamente de las imágenes de entrada locales o próximas, mediante el cálculo de las relaciones entre diferentes ubicaciones dentro del *dataset* y utilizando estas relaciones para ponderar la importancia de cada ubicación en función de su relevancia para la tarea en cuestión.

La aplicación de *instant batch normalization* persigue mejorar el rendimiento y la estabilidad del entrenamiento permitiendo a la red adaptarse a diferentes niveles de

abstracción y mantener las características específicas de las imágenes de entrada originales. Esta técnica es especialmente útil cuando se aplica, como es el caso, *data augmentation* o *transfer learning*.

5.6.1.3.- Capa de agregación

En estas capas se reducen las dimensiones de los *features maps* generados a la salida del *backbone*, mediante diferentes tipos de *pooling* que van a favorecer destacar determinadas características de las imágenes de entrada.

Se pueden emplear cuatro capas de agregación diferentes, en concreto cuatro técnicas de *pooling*, a saber: *max pooling, average pooling, generalized mean pooling* y *attention pooling*.

La fórmula de GMP (que es la que se va a implementar en el modelo) es:

$$GMP: f_c = \left(\frac{1}{|X_c|}\sum_{x \in X_c} x^{\alpha}\right)^{\frac{1}{\alpha}}$$

Ecuación 11: fórmula de *generalized mean pooling*

Donde f_c correspondería al nuevo *feature map* obtenido, X_c es el *feature map* generado a la salida del *backbone* y α es un coeficiente de control.

5.6.1.4.- Cabecera

A la salida del módulo anterior, se incluyen tres posibles cabeceras: una lineal, una basada en *batch normalization* (BN) y una de decisión. La lineal únicamente contiene una capa final de decisión o *decisión layer*; la BN contiene una capa BN y una capa de decisión; y el *head* o cabecera de reducción contiene una operación consistente en una convolución, un BN, una capa *ReLU (rectified lineal unit)* y una capa de *dropout*, seguida de otra capa de reducción (que disminuye la dimensionalidad del vector de características) y una capa de decisión.

5.6.2.- Funciones de pérdida

FastReID implementa cuatro posibles funciones de pérdida: entropía cruzada o *cross-entropy loss, arcface loss, triplet loss* (ya explicada anteriormente en el apartado 2.1.2.3) y *circle loss.*

La entropía cruzada mide la diferencia entre dos distribuciones de probabilidad: la distribución real y la distribución estimada por el modelo. En decir, compara cuán similar es la salida del modelo a la verdad fundamental, y busca minimizar esta diferencia para mejorar el rendimiento del modelo. Esta función penaliza las predicciones incorrectas al asignarles un valor de pérdida alto. Cuando la predicción es correcta, el valor de la pérdida es bajo. Durante el entrenamiento, se busca minimizar la pérdida de entropía cruzada, lo que significa que el modelo se ajustará para producir predicciones más cercanas a las etiquetas verdaderas.

Arcface loss [193] modifica la función de pérdida softmax para incluir un margen angular entre las características extraídas de las imágenes y los pesos de las clases. Esto permite que el modelo aprenda características más discriminativas y separables. Durante el entrenamiento, la pérdida de ArcFace busca minimizar esa distancia angular entre las características de la misma clase y maximizar la distancia entre las características de diferentes clases. El margen angular aditivo 'm' proporciona una separación adicional entre las clases, lo que resulta en un mejor rendimiento de clasificación y generalización del modelo.

Por último, *circle loss* [194] minimiza las similitudes intraclases (entre pares de muestras de la misma clase) y maximiza las similitudes entre clases (entre pares de muestras de diferentes clases). La idea principal es representar las similitudes de pares como puntos en un círculo, de modo que la función de pérdida pueda optimizar las similitudes de pares de manera más eficiente.

5.6.3-. Modelo para la re – identificación de vehículos

Como se ha explicado a lo largo de los apartados, la red está pensada para poder configurarse bajo diferentes posibilidades, de tal manera que se pueda adaptar a diferentes tareas de re – identificación.

En este caso el modelo empleado presenta el siguiente esquema:

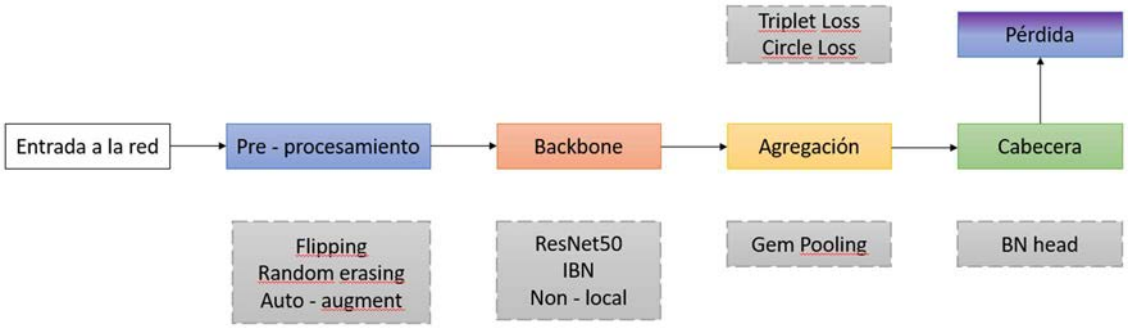


Figura 57: esquema de implementación de FastReID para la re - identificación de vehículos

5.7.- Datasets para el testeo de la re – identificación

Cuando se estudiaron diferentes posibilidades de sistemas para la re – identificación de vehículos, la decisión de utilizar FastReID vino motivada por su versatilidad, su presumible robustez en base a las métricas aportadas (que posteriormente se reseñarán) y por su facilidad de implementación.

Sin embargo, tal y como se refleja en el propio documento que introduce FastReID, se hace necesario poder verificar la viabilidad de la herramienta mediante determinados test. Para ello, se debe incluir una búsqueda (o *query*) e igualmente comprobar los resultados. Por lo tanto, de manera análoga a como se hizo en la detección de matrículas y la lectura de caracteres, se decidió crear dos *datasets* diferentes, con la particularidad de que en este caso no están destinado a labores de entrenamiento si no de validación.

5.7.1.- Highway Gantry Dataset (HGD)

Highway Gantry Dataset (HGD) es el nombre que recibe el primer conjunto de datos. Contiene imágenes captadas desde un pórtico en una autopista con un ángulo elevado, oblicuo y posterior, emulando las imágenes que captarían las cámaras de vigilancia emplazadas por la Dirección General de Tráfico (DGT) para el control de carreteras. Este *dataset* se caracteriza por tener imágenes de la misma categoría (vehículos) con un gran parecido entre sí, ya que presentan la misma perspectiva, una iluminación uniforme y ningún obstáculo. Su finalidad es utilizarse como una primera etapa en el proceso de evaluación, ya que es comparativamente menos exigente y se espera que arroje resultados más positivos. Contine 458 imágenes de 200 modelos de vehículos distintos. En este caso, el etiquetado también se realizó de manera manual, incluyendo el *bounding box* en coordenadas relativas de dónde se ubica el vehículo dentro del fotograma, así

como la marca, modelo, color y edición del mismo, para poder realizar posteriormente las búsquedas en las imágenes.

Figura 58: ejemplos de imágenes de HGD

5.7.2.- Operational Urban Dataset (OUD)

Este segundo *dataset* pretende emular escenarios operativos que se presentan habitualmente durante la realización de determinadas video vigilancias sobre vehículos. En este caso se han obtenido imágenes correspondientes a escenas de tráfico captadas en intersecciones, donde es habitual que no sea posible obtener una lectura correcta de una placa de matrícula.

El escenario se divide en dos lugares de grabación distintos (v_1y v_2), con una variedad de perspectivas y oclusiones entre vehículos y vegetación. Cada escena ha sido captada simultáneamente por dos cámaras (c_1y c_2) y representa un alto grado de dificultad. Disponer de dos fuentes de entrada permite buscar vehículos anotados de una cámara en la otra con una perspectiva diferente, que es el objetivo principal de este estudio. El conjunto de datos incluye un total de 1.255 imágenes y 69 clases con criterios de anotación ligeramente diferentes. v_1 contiene todas las anotaciones posibles de vehículos, incluidas las vistas muy lejanas y parciales, mientras que en v_2 sólo se anotaron vehículos completos con un tamaño mínimo reconocible.

Al igual que en el anterior *dataset*, también se incluyen el *bounding box* seguido de la marca, modelo, edición y color de los vehículos.

Figura 59: ejemplos de imágenes de OUD

5.7.- Test realizados

Para validar la implementación de la arquitectura de FastReID propuesta para la re – identificación de vehículos, se decidió hacer dos planteamientos. El primero, preparar un modelo con una estructura similar pero con otro tipo de particularidades y evaluar su rendimiento con alguno de los *datasets* públicos utilizados en el entrenamiento del modelo de FastReID, como son VeRi y VeRi - Wild.

El segundo planteamiento consistió en comparar las métricas de los modelos definitivos mediante la ejecución de búsquedas, implementando para ello la función de la distancia euclídea (anteriormente descrita) con un determinado intervalo de confianza para ejecutar re – identificaciones como sucedería en un caso real.

5.7.1.- Preparación de un modelo comparativo

Para la creación del modelo comparativo se empleó como *backbone* redes neuronales de la familia EfficientNet [67], implementando adicionalmente capas de agregación tipo *max pooling*, un optimizador *SGD* y en este caso la función de pérdida *categorical crossentropy*, con la finalidad de emular y comparar ambos sistemas, actuando este último como si se tratase de un clasificador.

El primer entrenamiento se llevó a cabo con *Stanford Cars Dataset* [107], muy utilizado en la valoración de algoritmos basados en CNN para clasificación de vehículos, como se reseñó en el estado del arte. Se empleó una versión reducida del *dataset*

(aproximadamente el 10% del total de las imágenes) para ajustar la tasa de abandono (*dropout*) y aprendizaje. El abandono o *dropout* es un término empleado cuando durante el entrenamiento se procede a "apagar" algunas neuronas en ciertas capas de manera aleatoria, para favorecer la extracción de características a través de varios caminos y contribuyendo a una mejor generalización del modelo. En cambio, la tasa de aprendizaje (como se explicó anteriormente) indica la velocidad a la que se actualizan los pesos. Un valor reducido permite realizar más actualizaciones de los pesos, pero a costa de un mayor tiempo de entrenamiento, por lo que es necesario encontrar un equilibrio (optimización).

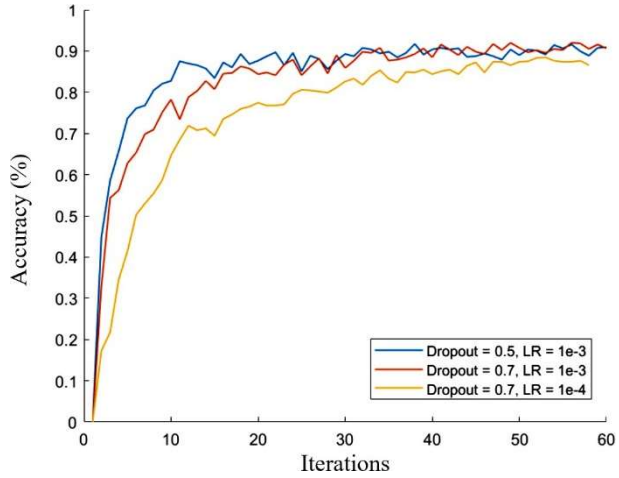


Figura 60: comparativa de la influencia del *dropout* y del *learning rate*

La gráfica permite apreciar dos resultados reseñables. En primer lugar, que un *dropout* de 0,7 muestran una generalización ligeramente superior que las de 0,5. A pesar de tardar algo más en las primeras fases del entrenamiento, se alcanza un máximo inferior con 0,5. Respeto al *learning rate*, una tasa de aprendizaje de 0,001 parece ser la más adecuada en este caso, ya que maximiza la precisión más rápidamente.

Tras estas pruebas, se evaluó el rendimiento de tres versiones de la red EfficientNet (B0, B3 y B7). La salida se configuró añadiendo una capa de agregación de *max pooling* global para cada filtro de salida y una capa de clasificación densa con el valor de *dropout* previamente ajustado.

Además, se utilizó el conjunto de datos VeRi [109] para realizar otra ronda de entrenamiento con las mismas redes. Sin embargo, para simplificar el proceso de entrenamiento, se modificaron ligeramente las clases de salida. La red se diseñó originalmente con fines de clasificación, por lo que se eliminó la última capa *softmax*, dejando que el modelo codificara la imagen de entrada con la salida de la penúltima

capa. Como puede observarse, la precisión es similar entre los tres modelos, por lo que se elige EfficientNetB0 por las mismas razones.

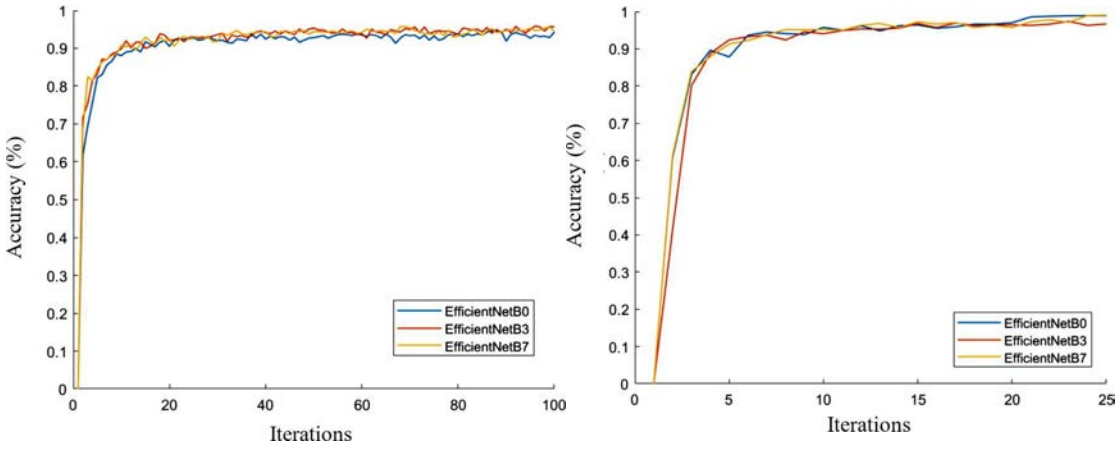


Figura 61: comparativa de los tres modelos de EfficientNet con Standford Cars Dataset (izquierda) y VeRi (derecha)

Dado el mayor tamaño y la menor velocidad de procesamiento de los modelos más grandes, tiene sentido elegir el modelo B0 para utilizarlo como red de caracterización, cuando posteriormente se compare con los datos de FastReID.

5.7.2.- Comparativa de resultados

La siguiente tabla muestra la exactitud (*accuracy*) de los dos modelos entrenados (EfficientNetB0) frente a los modelos FastReid pre entrenados en los *dataset* públicos. Para realizar la comparativa de ambos modelos, se ha medido la exactitud correspondiente a una prueba de cotejo entre pares positivo - negativo al llevar a cabo una búsqueda (es decir, si la imagen de entrada se corresponde con el vehículo objetivo o no), mediante la incorporación de la distancia euclídea bajo un margen de confianza determinado. Cada par positivo-negativo se ha creado con cada imagen del conjunto de evaluación, una imagen de su clase (positiva) y una imagen aleatoria del resto de clases (negativa). De estos resultados se podría extraer que el modelo FastReID pre entrenado con VeRi-Wild (un *dataset* con mayor cantidad de datos), es un mejor candidato que el modelo pre entrenado con VeRi.

Modelo	Stanford-Cars	VeRi	VeRi-Wild
EfficientNetB0 (Stanford-Cars)	**0,835**	0,625	0,727
EfficientNetB0 (VeRi)	0,591	0,772	0,797
FastReid (VeRi)	0,606	**0,968**	0,908
FastReid (VeRi-Wild)	0,676	0,905	**0,995**

Tabla 9: comparativa de resultados con *datasets* públicos

Sin embargo, se consideró someter de nuevo estos modelos a los dos *datasets* creados específicamente para recrear las situaciones reales a las que se podría enfrentar el sistema; es decir, con HGD y con OUD. Los resultados tras ejecutar estos nuevos test ponen de manifiesto que el modelo más favorable pasa a ser el FastReID entrenado con VeRi. Por lo tanto, e hilvanando con el apartado 5.6.3, finalmente se decidió que la parte de re – identificación se iba a llevar a cabo mediante la implementación directa de este algoritmo.

Modelo	HGD	OUD v1c1	OUD v2c1	HGD v1	HGD v2
EfficientNetB0 (Stanford-Cars)	0,851	0,736	0,841	0,625	0,619
EfficientNetB0 (VeRi-776)	0,966	0,882	0,916	0,715	0,781
FastReid (VeRi)	**0,979**	**0,940**	**0,966**	**0,878**	**0,915**
FastReid (VeRi-Wild)	0,967	0,909	0,897	0,788	0,825

Tabla 10: comparativa de resultados con los *datasets* propios

El resultado se puede apreciar de forma visual en la siguiente figura:

Figura 62: ejemplo de la re - identificación de vehículos

5.8.- Implementación del modelo

Finalmente, tras los resultados obtenidos, se va a proceder a desgranar la implementación completa de la herramienta desarrollada. Los trabajos que se han llevado a cabo durante la presente investigación han sido:

- Creación de un *dataset* específico para la detección y lectura de matrículas (Spanish ALPR Dataset)
- Entrenamiento de la red YOLOv5 en la variante "s" tanto en detección de matrículas como en lectura de caracteres
- Desarrollo de dos *dataset* para entrenamientos y test en labores de re – identificación de vehículos (Highway Gantry Dataset y Operational Urban Dataset)
- Testeo de diferentes herramientas de re – identificación de vehículos
- Implementación del sistema completo, formado por:
    - Red de detección de vehículos
    - Red de detección de matrículas
    - Red de lectura de caracteres
    - Red de re – identificación de vehículos

5.8.1.- Entorno de trabajo

5.8.1.1.- Elementos Hardware

Para llevar a cabo todas tareas anteriormente reseñadas se han empleado los siguientes elementos:

Servidor con:

- Procesador AMD Ryzen 7 3700X 8-Core de 3,6 GHz
- 64 GB de memoria RAM
- 2 tarjetas gráficas NVIDIA RTX 3090 de 24 GB por unidad (48 GB en total)
- Disco duro 1 TB SSD

Para las imágenes de los diferentes *datasets*, se emplearon los siguientes medios:

- Canon P90D con diferentes objetivos
- Canon 720sx
- Xiaomi Mi 10T
- iPhone 7
- BQ Aquaris X5
- Cámara instalada en pórtico

A efectos de despliegue y tomando como referencia las características operativas desgranadas en el Capítulo III, se trataría de un sistema centralizado que bebería de captadores ubicados en entornos diferentes. Principalmente porque a nivel de hardware, aunque los requisitos para ejecutarse son menores que para su entrenamiento, son equipos que presentan un consumo elevado (sobre todo si se quiere sacar partido al empleo de una GPU). Además que a efectos prácticos, donde mayor rendimiento se puede sacar a la solución es empleándose en un entorno desatendido y/o como herramienta de post procesamiento de la información.

No obstante, la versatilidad de la herramienta permitiría, con algunas modificaciones y con el hardware adecuado, o bien implementar un equipo por cada captador, o bien centralizar varios flujos de vídeo (hablando en este caso de funcionamiento en tiempo real). La diferencia radicaría principalmente en escalar la capacidad de proceso del hardware instalado.

5.8.1.2.- Entorno software

A la hora de implementar los diferentes códigos, se ha trabajado en un ecosistema que requiere operar de manera unísona para poder explotar las capacidades de las tarjetas gráficas o GPU, fundamentales para poder llevar a cabo los diferentes entrenamientos.

Por un lado, el sistema operativo utilizado en el servidor es Ubuntu, una distribución de Linux.

El lenguaje de programación elegido ha sido Python, en su versión 3, un lenguaje de programación orientado a objetos de alto nivel que permite la implementación de librerías específicas para simplificar la programación de algoritmos y funciones.

Para la implementación de las funciones específicas de las arquitecturas de las redes neuronales se ha utilizado Pytorch, una biblioteca de código abierto para Python que permite simplificar la programación de las operaciones con tensores. Concretamente, se ha empleado la versión 1.13.1.

Por último, para poder ejecutar los diferentes procesos mediante las dos GPUs se utilizó CUDA y CuDNN, herramientas ambas desarrolladas por NVIDIA (misma marca fabricante de las dos tarjetas gráficas). CUDA (Compute Unified Device Architecture) es una plataforma que habilita precisamente el uso de GPUs, responsable de la paralelización de los procesos. Mientras que CuDNN (CUDA Deep Neural Network *library*) es una librería de software que facilita y acelera el desarrollo de aplicaciones de aprendizaje profundo utilizando la plataforma CUDA.

Es importante reseñar que este *framework* opera como un ecosistema conjunto, en el cuál debe haber una sinergia perfecta entre las diferentes versiones de Pytorch, CUDA, CuDNN y *drivers* instalados en la tarjeta gráfica. Si no, y esto es fácil que suceda, no va a ser posible poder emplear las GPUs para el entrenamiento, y esto es un fallo crucial, ya que si no es prácticamente imposible trabajar con redes neuronales bastante profundas o con *datasets* que incorporen gran cantidad de imágenes.

En este caso en concreto, se ha utilizado la versión de CUDA 11.6.r11.6, CuDNN versión 11.4 y los *drivers* 470.182.03.

5.8.2.- Implementación del código

Para la ejecución del código, lo primero que se hace es cargar los cuatro modelos anteriormente reseñados con sus entrenamientos pertinentes, es decir:

- YOLOv5s: modelo estándar pre entrenado en COCO, con un tamaño de imagen de entrada de 640 x 640 píxeles, responsable de la detección de vehículos
- YOLOv5s_LP: modelo entrenado con Spanish ALPR Dataset, responsable de la detección de matrículas
- YOLOv5s_OCR: modelo entrenado con Spanish ALPR Dataset, responsable de la lectura de caracteres

- VeRi_FastReID: modelo pre entrenado con VeRi, con imágenes con un tamaño de entrada de 256 x 256 píxeles, responsable de llevar a cabo la re – identificación de vehículos

El algoritmo se implementa de tal manera que inicialmente sólo está operando YOLOv5s realizando funciones de detección de vehículos. En caso de detección positiva, automáticamente empiezan a funcionar de manera simultánea otras dos redes, por un lado, YOLOv5s_LP para detectar la matrícula en esa imagen, y por otro lado VeRi_FastReID para llevar a cabo la re – identificación de los vehículos. Si YOLOv5s_LP consigue detectar una matrícula, entonces se activa YOLOv5s_OCR para extraer los caracteres de dentro de la matrícula.

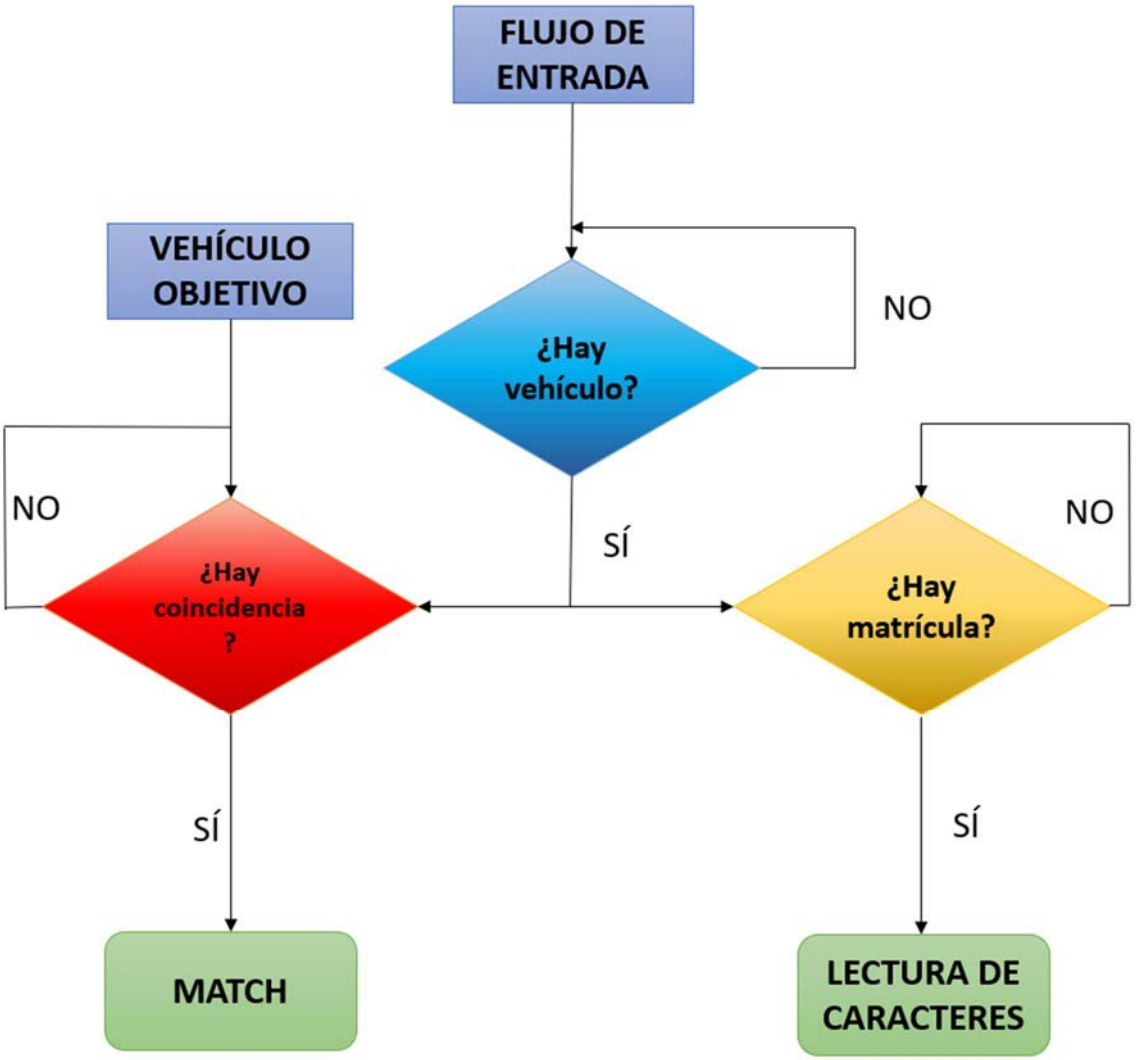


Figura 63: diagrama de flujo del funcionamiento del modelo

El siguiente algoritmo permite ver de forma resumida el funcionamiento completo del sistema:

 # Carga los modelos anteriormente reseñados

Detección de vehículos: veh_model = 'YOLOv5s.pt'

Detección de matrículas: lp_model = 'YOLOv5s_LP.pt'

OCR: ocr_model = 'YOLOv5s_OCR.pt'

Re – identificación: reid_model = 'VeRi_FastReID.pt'

 # Carga del vídeo de entrada y de la imagen objetivo

video = 'video_a_analizar.mp4'

objv_image = 'imagen_objetivo.jpg'

 # Extraer características

objv_features = extract_features(objv_image, reid_model)

while video:

    # Lectura de los fotogramas

    ret, frame = video.read()

    if not ret:

    break

    # Una vez cargados los fotogramas, se realiza la detección de vehículos en base al primer modelo

    vehicles = detect(frame, veh_model)

    if len(vehicles) > 0:

        # Si se consigue detectar un vehículo, automáticamente se recorta esa parte de la imagen donde se alberga el vehículo y se carga para realizar la detección de matrículas con el segundo modelo

        license_plate = detect(frame, lp_model)

        if len(license_plate) > 0:

            # Si se ha detectado una matrícula, se emplea el tercer modelo para extraer los caracteres

            characters = detect(frame, ocr_model)

            # Procesar detecciones de la tercera categoría, si es necesario

            if len(characters) > 0:

                # Procesar cada detección (dibujar un rectángulo alrededor del objeto)

```
            # Se extrae la imagen si se ha conseguido detectar un vehículo
            for det in vehicles:
                vehicle = frame[det.ymin:det.ymax, det.xmin:det.xmax]

            # Se calcula el vector de características del objeto detectado usando FastReID
            detected_features = extract_features(vehicle, reid_model)

            # Se calcula la similitud entre las características de la imagen de referencia y el objeto detectado
            similarity = calculate_similarity(ref_features, detected_features)

            # Si la similitud supera un umbral definido
            if similarity > threshold:
                # Realizar acciones necesarias, por ejemplo, recuadrar el objeto en color verde
        end
    end
end
```

Como resultado, se obtendrían fotogramas en los que se podría apreciar los caracteres de las matrículas y los vehículos detectados, siendo el vehículo objetivo el que aparece resaltado en color verde.

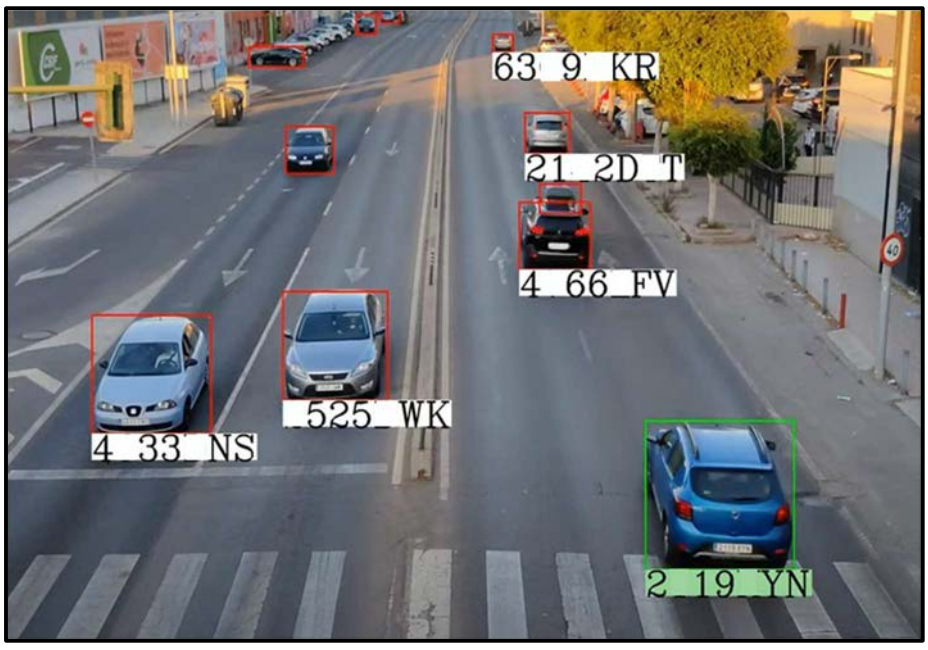


Figura 64: resultado de la implementación de la herramienta

CONCLUSIONES Y TRABAJOS FUTUROS VI

Para finalizar el documento, en este capítulo se van a abordar, por un lado, las metas alcanzadas a lo largo de la presente investigación y, por otro lado, las posibles líneas de investigación futuras.

6.1.- Conclusiones

En el apartado 1.5 se plantearon una serie de objetivos que han servido de guía para la ejecución y desarrollo de toda la investigación. Atendiendo una por una las principales premisas, se puede concluir que se han logrado las siguientes aportaciones:

- Se ha conseguido incrementar las capacidades de los sistemas de reconocimiento de vehículos tradicionales. Esto ha sido posible al unificar, de forma novedosa, dos ecosistemas que, aunque comparten una finalidad común, no se habían hecho operar previamente de manera conjunta. De esta forma, se consigue tener una visión integral de los elementos del vehículo para poder llevar a cabo su identificación.
- Se ha presentado un enfoque novedoso a la hora de realizar la lectura de los caracteres de la matrícula, rompiendo con el uso convencional de redes neuronales específicamente destinadas a la realización de labores de OCR. El empleo de un algoritmo de detección como YOLOv5 consigue una mayor robustez frente a variaciones espaciales como diferentes enfoques, giros y oclusiones parciales en las imágenes, ya que analiza los caracteres de manera individualizada y no como un conjunto de elementos agrupados.
- Se ha desarrollado una herramienta versátil, capaz de funcionar con cualquier captador y en cualquier ecosistema, sin estar supeditado necesariamente a la elección de determinados elementos. Además, es totalmente compatible para trabajar tanto en tiempo real como realizando post – procesado de imágenes.
- Se han creado tres *datasets* específicos que recrean de manera efectiva situaciones operacionales reales y que son novedosos tanto en su desarrollo como en su finalidad.

En definitiva, se considera que la presente investigación ha logrado alcanzar los resultados perseguidos en los objetivos propuestos, demostrando que la metodología empleada y los enfoques desarrollados fueron efectivos para abordar las problemáticas planteadas. Los hallazgos obtenidos no solo han permitido obtener una solución satisfactoria a las cuestiones planteadas, sino que también ha proporcionado una comprensión más profunda de los mecanismos subyacentes (la aplicación de metodologías específicas de *deep learning* aplicadas al procesamiento de imágenes) y de los condicionantes derivados del entorno real de operación.

Sin embargo, si algo se debe reseñar con especial atención además de haber cumplido con los objetivos previamente establecidos, es que este trabajo ha permitido sentar las bases para una nueva línea de investigación. La exploración de las implicaciones y aplicaciones prácticas de un doble factor de autenticación de elementos visuales, así como la identificación de dos técnicas de identificación diferentes pero complementarias, han generado un camino prometedor para el avance en esta área de conocimiento. Y todo esto teniendo presente las dificultades y requisitos del usuario final.

Esta nueva línea de investigación tiene el potencial de ampliar los límites del campo, ofreciendo oportunidades para desarrollar enfoques innovadores, mejorar las tecnologías existentes y abordar nuevos desafíos. Prueba de ello es la confección de los *datasets*, algo que sin duda es muy laborioso tanto por la recopilación de los datos como para realizar su etiquetado. Se espera por tanto que en un futuro a corto plazo se puedan superar los resultados obtenidos, ya que la nueva línea de investigación marcada no sólo ofrece varias de áreas de mejora, sino que además muestra qué caminos se pueden seguir.

6.2.- Trabajos futuros

Como consecuencia de la investigación realizada, se han visualizado varias evoluciones que podrían proporcionar mejoras susceptibles en las capacidades del sistema y que constituirían una secuencia lógica en las líneas de investigación definidas. Estas mejoras se pueden encontrar en los tres aspectos que se han trabajado, es decir: la detección y lectura de matrículas, la re – identificación de vehículos y la creación y mejora de los diferentes *datasets.*

En la lectura de matrículas, la primera aportación podría ser reemplazar la infraestructura basada en YOLOv5 por la de YOLOv8 [195] , recientemente lanzada y

que ofrece mejoras en cuanto a rendimiento y resultados. Además, existen también variantes del algoritmo que son más ligeras y se pueden implementar en equipos tipo Jetson NANO de NVIDIA, dispositivos con buena capacidad gráfica y un menor consumo y precio que permitirían agilizar la construcción de una infraestructura basada en procesamiento local.

Respecto a la re – identificación de vehículos, si bien FastReID ha dado muy buenos resultados, probablemente estos se podrían mejorar modificando la etapa de *backbone* de la red, sustituyéndola por algún otro tipo de CNN como alguna de las ramas de Efficientnet o similares, e integrándola con el resto de la infraestructura manteniendo la configuración original, o incluso haciendo diferentes pruebas alterando las funciones de pérdida o implementando algún tipo de capa de agregación diferente. Con esto se perseguiría incrementar la optimización de la red, mejorando incluso los resultados y reduciendo el coste computacional de ejecución de los algoritmos. No hay que olvidar que precisamente FastReID está concebido por módulos para implementar mejoras y evoluciones de manera relativamente sencilla.

También sería interesante entrenar el clasificador de entrada de la red única y exclusivamente en la detección de vehículos, con la finalidad de evaluar si existe un incremento en el rendimiento de la herramienta.

En cuanto a los *datasets*, la primera mejora que se hace fundamental es la de aumentar el número de imágenes disponibles de placas de matrículas. Probablemente permitiría una mejor generalización del modelo y se potenciarían las métricas obtenidas. Para ello, se podría aprovechar la red ya entrenada de detección de matrículas para facilitar la labor del etiquetado, que es uno de los principales problemas a la hora de realizar un *dataset* (por el tiempo requerido y por la precisión de la zona delimitada). Además, habría que enriquecer el *dataset* con una mayor variedad de caracteres (vocales) y de tipos de matrículas, especialmente aquellas menos habituales y con otra gama cromática como las de vehículos importados temporales (con fondo rojo), o matrículas correspondientes a taxis y licencia VTC (con fondo azul).

La otra gran tarea pendiente sería la confección de un *dataset* que combinase de forma conjunta un etiquetado de vehículos y matrículas, de tal manera que se pudiese parametrizar los resultados obtenidos de la aplicación de este doble factor de autenticación visual.

Para concluir, existen otras posibilidades relacionadas íntegramente con las capacidades operativas del sistema. Por un lado, para poder poner en funcionamiento la herramienta se debería crear un entorno de uso amigable y sencillo, que no sólo permita la realización de búsquedas si no también recopilar la información obtenida. Por otro lado, quedaría pendiente la implementación de un mecanismo de alerta, con la consiguiente infraestructura de transmisión de la información. Por último, se podría llevar a cabo una evaluación totalmente práctica del sistema mediante la búsqueda y prueba de diferentes captadores, intentando reproducir el entorno operativo que ofrezca los mejores resultados.

BIBLIOGRAFÍA

1. Conde-Zhingre, L. E., Quezada-Sarmiento, P. A., & Labanda, M. (2018, June). The new generation of mobile networks: 5G technology and its application in the e-education context. In *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-4). IEEE.
2. Vial, G. (2019). Understanding digital transformation: A review and a research agenda. *The journal of strategic information systems*, 28(2), 118-144.
3. del Val Román, J. L. (2016, March). Industria 4.0: la transformación digital de la industria. In Valencia: *Conferencia de Directores y Decanos de Ingeniería Informática, Informes* CODDII.
4. Bouwman, H., Carlsson, C., Carlsson, J., Nikou, S., Sell, A., & Walden, P. (2014). How Nokia failed to nail the Smartphone market.
5. Moussi, A., & van Amsterdam, U. (2017). Mini-Case Study: The Downfall of Blackberry. *Universteit van Amsterdam*, *5*, 185-206.
6. Wang, Q., Li, R., Wang, Q., & Chen, S. (2021). Non-fungible token (NFT): Overview, evaluation, opportunities and challenges. arXiv preprint arXiv:2105.07447.
7. Real Academia Española. (2023). Seguridad. En Diccionario de la lengua española (23.a ed.). https://dle.rae.es (consultado el 20 de mayo de 2023)
8. Maslow, A. H. (1943). A theory of human motivation. *Psychological review*, 50(4), 370.
9. Constitución española. (1978). *Boletín Oficial. Del Estado*, *311*, 29313-29424.
10. (2014). Ley 5/2014, de 4 de abril, de Seguridad Privada. *Boletín Oficial del Estado, núm. 83, de 5 de abril de 2014*, 27309 a 27350.
11. Ley Orgánica 4/2015, de 30 de marzo, de Protección de la Seguridad Ciudadana. Boletín Oficial del Estado, de 31 de marzo de 2015, núm. 77, páginas 27216 a 27243.
12. González, R. D. S. (2008). *Manual de medios técnicos de Seguridad*. Seguridad y Defensa.
13. Ley de Enjuiciamiento Criminal (1882). Real decreto de 14 de septiembre de 1882 por el que se aprueba la Ley de Enjuiciamiento Criminal.
14. Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales. *Boletín Oficial del Estado, núm. 294, de 6 de diciembre de 2018*, páginas 119788 a 119857.
15. Agencia Española de Protección de Datos. (2006). Instrucción 1/2006, de 8 de noviembre, sobre el tratamiento de datos personales con fines de vigilancia a través de sistemas de cámaras o videocámaras. *Boletín Oficial del Estado, núm. 296, de 11 de diciembre de 2006*, páginas 43243 a 43248.

16. Ley Orgánica 4/1997, de 4 de agosto, por la que se regula la utilización de videocámaras por las Fuerzas y Cuerpos de Seguridad en lugares públicos. *Boletín Oficial del Estado, núm. 185, de 5 de agosto de 1997*, páginas 23988 a 23990.

17. Real Decreto 885/2020, de 6 de octubre, por el que se establecen los requisitos para la comercialización y puesta en servicio de placas de matrícula para vehículos de motor y remolques, y por el que se modifica el Reglamento General de Vehículos. *Boletín Oficial del Estado, núm. 272, de 7 de octubre de 2020*, páginas 75933 a 75961.

18. Huidobro, J. M. (2016). Radares para el control del tráfico.

19. Mail, A. O. L., & Box, D. (2017). Two factor authentication.

20. Schneier, B. (2005). Two-factor authentication: too little, too late. *Communications of the ACM*, *48*(4), 136.

21. Reese, K., Smith, T., Dutson, J., Armknecht, J., Cameron, J., & Seamons, K. (2019, July). A usability study of five two-factor authentication methods. In *Proceedings of the Fifteenth Symposium on Usable Privacy and Security*.

22. Gope, P., & Sikdar, B. (2018). Lightweight and privacy-preserving two-factor authentication scheme for IoT devices. *IEEE Internet of Things Journal*, *6*(1), 580-589.

23. *IEC EN62676-4*; Video surveillance systems for use in security applications - Part 4: Application guidelines; International Standard, 2015.

24. Bouchrika, I. A survey of using biometrics for smart visual surveillance: Gait recognition. In *Surveillance in Action*; Springer: Cham, Switzerland, 2018; pp. 3–23.

25. Devasena, C.L.; Revathí, R.; Hemalatha, M. Video Surveillance Systems—A Survey. *Int. J. Comput. Sci. Issues (IJCSI)* **2011**, *8*, 635–642.

26. Renninger, L. W., & Malik, J. (2004). When is scene identification just texture recognition?. *Vision research*, *44*(19), 2301-2311.

27. Gong, S., Xiang, T., Gong, S., & Xiang, T. (2011). *Person re-identification* (pp. 301-313). Springer London.

28. Layne, R., Hospedales, T. M., Gong, S., & Mary, Q. (2012, September). Person re-identification by attributes. In *Bmvc* (Vol. 2, No. 3, p. 8).

29. Sharma, P., Singh, A., Singh, K. K., & Dhull, A. (2021). Vehicle identification using modified region-based convolution network for intelligent transportation system. *Multimedia Tools and Applications*, 1-25.

30. Li, R., Liu, Z., & Zhang, R. (2018). Studying the benefits of carpooling in an urban area using automatic vehicle identification data. *Transportation Research Part C: Emerging Technologies*, *93*, 367-380.

31. Persad, K., Walton, C. M., & Hussain, S. (2007). *Electronic vehicle identification: Industry standards, performance, and privacy issues* (No. Product 0-5217-P2).

32. Nakanishi, Y. J., & Western, J. (2005). Ensuring the security of transportation facilities: evaluation of advanced vehicle identification technologies. *Transportation research record*, *1938*(1), 9-16.

33. Day, C. M., Brennan, T. M., Hainen, A. M., Remias, S. M., & Bullock, D. M. (2012). Roadway system assessment using Bluetooth-based automatic vehicle identification travel time data.

34. Parlamento Europeo y Consejo de la Unión Europea. (2015). Reglamento (UE) 2015/758 del Parlamento Europeo y del Consejo, de 29 de abril de 2015, relativo a los requisitos de homologación de tipo para el despliegue del sistema eCall basado en el número 112 integrado en los vehículos y por el que se modifica la Directiva 2007/46/CE. *Diario Oficial de la Unión Europea*, L 123/77.

35. Mohammed, I. A. (2017). Systematic review of Identity Access Management in information security. *International Journal of Innovations in Engineering Research and Technology*, *4*(7), 1-7.

36. Bip&Drive. (s. f.). Bip&Drive: Telepeaje. Recuperado el 20 de mayo de 2023, de https://www.bipdrive.com/telepeaje/

37. Amazon.es. (s.f.). Traqueur - Localizador GPS para coches, vehículos y motos, Extra Long Standby. Recuperado el 20 de mayo de 2023, de https://www.amazon.es/Traqueur-v%C3%A9hicules-voitures-Localisateur-Extra-Long/dp/B074QNDSYJ

38. Garzon, S. R. (2012, June). Intelligent in-car-infotainment systems: A contextual personalized approach. In *2012 Eighth International Conference on Intelligent Environments* (pp. 315-318). IEEE.

39. Motakabber, S. M. A., Alam, A. Z., Wafa, S. A. F., & Francis, M. R. M. (2022). GPS and GSM Based Vehicle Tracker. *Asian Journal of Electrical and Electronic Engineering*, *2*(1), 17-24.

40. Forsyth, D. A., & Ponce, J. (2002). *Computer vision: a modern approach*. prentice hall professional technical reference.

41. Asociación Española de Normalización y Certificación (AENOR). (2011). UNE-ISO 3779:2011 Vehículos de carretera. Número de identificación de los vehículos (VIN). Contenido y estructura.

42. Shah, P., Karamchandani, S., Nadkar, T., Gulechha, N., Koli, K., & Lad, K. (2009, November). OCR-based chassis-number recognition using artificial neural networks. In *2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES)* (pp. 31-34). IEEE.

43. Shashirangana, J.; Padmasiri, H.; Meedeniya, D.; Perera, C. Automated license plate recognition: A survey on methods and techniques. *IEEE Access* **2020**, *9*, 11203–11225.

44. González-Cepeda, J., Ramajo, Á., & Armingol, J. M. (2022). Intelligent Video Surveillance Systems for Vehicle Identification Based on Multinet Architecture. *Information*, *13*(7), 325.

45. Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2012). Automatic license plate recognition (ALPR): A state-of-the-art review. *IEEE Transactions on circuits and systems for video technology*, *23*(2), 311-325.

46. Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.".

47. OpenALPR Technology Inc. (s. f.). OpenALPR: Automatic License Plate Recognition. Recuperado el 20 de mayo de 2023, de en https://www.openalpr.com/

48. Ejemplo de cámara térmica FLIR T865, Recuperado el 23 de mayo de https://www.flir.es/products/t865/?vertical=condition%20monitoring&segment=solutions

49. Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*, 273-297.

50. Bradski, G. R. (1998). Computer vision face tracking for use in a perceptual user interface. *Intel technology journal*.

51. Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, *55*(1), 119-139.

52. Viola, P., & Jones, M. (2001, December). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001* (Vol. 1, pp. I-I). IEEE.

53. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

54. Jaderberg, M., Simonyan, K., & Zisserman, A. (2015). Spatial transformer networks. *Advances in neural information processing systems*, *28*.

55. Silva, S. M., & Jung, C. R. (2018). License plate detection and recognition in unconstrained scenarios. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 580-596).

56. Smith, R. (2007, September). An overview of the Tesseract OCR engine. In *Ninth international conference on document analysis and recognition (ICDAR 2007)* (Vol. 2, pp. 629-633). IEEE.

57. Patel, C.; Patel, A.; Patel, D. Optical character recognition by open-source OCR tool tesseract: A case study. *Int. J. Comput. Appl.* **2012**, *55*, 50–56.

58. Singh, J., & Bhushan, B. (2019, October). Real time Indian license plate detection using deep neural networks and optical character recognition using LSTM tesseract. In *2019 international conference on computing, communication, and intelligent systems (ICCCIS)* (pp. 347-352). IEEE.

59. Goel, T., Tripathi, K. C., & Sharma, M. L. (2020). Single Line License Plate Detection Using OPENCV and tesseract. *International Research Journal of Engineering and Technology (IRJET)*, (05).

60. Dias, C., Jagetiya, A., & Chaurasia, S. (2019, September). Anonymous vehicle detection for secure campuses: A framework for license plate recognition using deep learning. In *2019 2nd international conference on intelligent communication and computational techniques (ICCT)* (pp. 79-82). IEEE.

61. García Serrano, A. Aplicación de Sistemas de Percepción Para la Seguridad Vial; *Departamento de Ingeniería Eléctrica, Electrónica y Automática*, Universidad Carlos III: Madrid, Spain, 2020.

62. Zherzdev, S., & Gruzdev, A. (2018). Lprnet: License plate recognition via deep neural networks. *arXiv preprint arXiv:1806.10447*.

63. Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263-7271).

64. Li, H., Wang, P., & Shen, C. (2018). Toward end-to-end car license plate detection and recognition with deep neural networks. *IEEE Transactions on Intelligent Transportation Systems*, *20*(3), 1126-1136.

65. Xu, Z., Yang, W., Meng, A., Lu, N., Huang, H., Ying, C., & Huang, L. (2018). Towards end-to-end license plate detection and recognition: A large dataset and baseline. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 255-271).

66. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

67. Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105-6114). PMLR.

68. Pirgazi, J., Pourhashem Kallehbasti, M. M., & Ghanbari Sorkhi, A. (2022). An end-to-end deep learning approach for plate recognition in intelligent transportation systems. *Wireless Communications and Mobile Computing*, *2022*, 1-13.

69. Kaur, P., Kumar, Y., Ahmed, S., Alhumam, A., Singla, R., & Ijaz, M. F. (2022). Automatic license plate recognition system for vehicles using a cnn. *Computers, Materials & Continua*, *71*(1), 35-50.

70. Hossain, S. N., Hassan, M. Z., & Masba, M. M. A. (2022). Automatic License Plate Recognition System for Bangladeshi Vehicles Using Deep Neural Network. In *Proceedings of the International Conference on Big Data, IoT, and Machine Learning: BIM 2021* (pp. 91-102). Springer Singapore.

71. Shahidi Zandi, M., & Rajabi, R. (2022). Deep learning based framework for Iranian license plate detection and recognition. *Multimedia Tools and Applications*, *81*(11), 15841-15858.

72. Ashrafee, A., Khan, A. M., Irbaz, M. S., Nasim, A., & Abdullah, M. D. (2022). Real-time bangla license plate recognition system for low resource video-based applications. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 479-488).

73. Chiu, Y. C., Tsai, C. Y., Ruan, M. D., Shen, G. Y., & Lee, T. T. (2020, August). Mobilenet-SSDv2: An improved object detection model for embedded systems. In *2020 International conference on system science and engineering (ICSSE)* (pp. 1-5). IEEE.

74. Padmasiri, H., Shashirangana, J., Meedeniya, D., Rana, O., & Perera, C. (2022). Automated license plate recognition for resource-constrained environments. *Sensors*, *22*(4), 1434.

75. Ali, F., Rathor, H., & Akram, W. (2021, March). License plate recognition system. In *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)* (pp. 1053-1055). IEEE.

76. Yang, C., & Zhou, L. (2022). Design and Implementation of License Plate Recognition System Based on Android. In *Proceedings of the 11th International Conference on Computer Engineering and Networks* (pp. 211-219). Springer Singapore.

77. Unión Europea. (2016). Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos (Reglamento General de Protección de Datos). http://data.europa.eu/eli/reg/2016/679/oj

78. Kessentini, Y., Besbes, M. D., Ammar, S., & Chabbouh, A. (2019). A two-stage deep neural network for multi-norm license plate detection and recognition. *Expert systems with applications*, *136*, 159-170.

79. Laroca, R., Severo, E., Zanlorensi, L. A., Oliveira, L. S., Gonçalves, G. R., Schwartz, W. R., & Menotti, D. (2018, July). A robust real-time automatic license plate recognition based on the YOLO detector. In *2018 international joint conference on neural networks (ijcnn)* (pp. 1-10). IEEE.

80. OpenALPR. Openalpr-Eu Dataset. (2016). Recuperado el 21 de mayo de 2023 en: https://github.com/openalpr/benchmarks/tree/master/endtoend/eu (accessed on 4 July 2022).

81. Chan, L. Y., Zimmer, A., da Silva, J. L., & Brandmeier, T. (2020, September). European union dataset and annotation tool for real time automatic license plate detection and blurring. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1-6). IEEE.

82. Yang, H., Cai, J., Zhu, M., Liu, C., & Wang, Y. (2022). Traffic-informed multi-camera sensing (TIMS) system based on vehicle re-identification. *IEEE Transactions on Intelligent Transportation Systems*, *23*(10), 17189-17200.

83. Fernandes, A. O., Moreira, L. F., & Mata, J. M. (2011, December). Machine vision applications and development aspects. In *2011 9th IEEE international conference on control and automation (ICCA)* (pp. 1274-1278). IEEE.

84. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815-823).

85. Wang, Y. (2022, February). Deep learning technology for re-identification of people and vehicles. In *2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)* (pp. 972-975). IEEE.

86. Khan, S. D., & Ullah, H. (2019). A survey of advances in vision-based vehicle re-identification. *Computer Vision and Image Understanding*, *182*, 50-63.

87. Wang, H.; Hou, J.; Chen, N. A survey of vehicle re-identification based on deep learning. *IEEE Access* **2019**, *7*, 172443–172469.

88. He, B., Li, J., Zhao, Y., & Tian, Y. (2019). Part-regularized near-duplicate vehicle re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3997-4005).

89. Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1798-1828.

90. Xing, E., Jordan, M., Russell, S. J., & Ng, A. (2002). Distance metric learning with application to clustering with side-information. *Advances in neural information processing systems*, *15*.

91. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., & Shah, R. (1993). Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, *6*.

92. Cai, J., Deng, J., Aftab, M. U., Khokhar, M. S., & Kumar, R. (2019). Efficient and deep vehicle re-identification using multi-level feature extraction. *Applied Sciences*, *9*(7), 1291.

93. Liu, X., Liu, W., Mei, T., & Ma, H. (2016). A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14* (pp. 869-884). Springer International Publishing.

94. Zhang, Y., Liu, D., & Zha, Z. J. (2017, July). Improving triplet-wise training of convolutional neural network for vehicle re-identification. In *2017 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1386-1391). IEEE.

95. Li, Y., Li, Y., Yan, H., & Liu, J. (2017, September). Deep joint discriminative learning for vehicle re-identification and retrieval. In *2017 IEEE International Conference on Image Processing (ICIP)* (pp. 395-399). IEEE.

96. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative Adversarial Networks. Advances in Neural Information Processing Systems.

97. Lou, Y., Bai, Y., Liu, J., Wang, S., & Duan, L. Y. (2019). Embedding adversarial learning for vehicle re-identification. *IEEE Transactions on Image Processing*, *28*(8), 3794-3807.

98. Zhou, Y., & Shao, L. (2017, September). Cross-view GAN based vehicle generation for re-identification. In *BMVC* (Vol. 1, pp. 1-12).

99. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

100. Zheng, Z.; Ruan, T.; Wei, Y.; Yang, Y.; Mei, T. VehicleNet: Learning robust visual representation for vehicle re-identification. *IEEE Trans. Multimed.* **2020**, *23*, 2683–2693.

101. Bai, S., Zheng, Z., Wang, X., Lin, J., Zhang, Z., Zhou, C., ... & Yang, Y. (2021). Connecting language and vision for natural language-based vehicle retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4034-4043).

102. He, L., Liao, X., Liu, W., Liu, X., Cheng, P., & Mei, T. (2020). Fastreid: A pytorch toolbox for general instance re-identification. *arXiv preprint arXiv:2006.02631*.

103. Tian, X., Pang, X., Jiang, G., Meng, Q., & Zheng, Y. (2022). Vehicle Re-Identification Based on Global Relational Attention and Multi-Granularity Feature Learning. *IEEE Access*, *10*, 17674-17682.

104. Li, Y., Liu, K., Jin, Y., Wang, T., & Lin, W. (2020). VARID: Viewpoint-aware re-identification of vehicle based on triplet loss. *IEEE Transactions on Intelligent Transportation Systems*, *23*(2), 1381-1390.

105. Li, K., Ding, Z., Li, K., Zhang, Y., & Fu, Y. (2020). Vehicle and person re-identification with support neighbor loss. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(2), 826-838.

106. Meng, D., Li, L., Liu, X., Gao, L., & Huang, Q. (2022). Viewpoint Alignment and Discriminative Parts Enhancement in 3D Space for Vehicle ReID. *IEEE Transactions on Multimedia*.

107. Krause, J., Stark, M., Deng, J., & Fei-Fei, L. (2013). 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops* (pp. 554-561).

108. Liu, H., Tian, Y., Yang, Y., Pang, L., & Huang, T. (2016). Deep relative distance learning: Tell the difference between similar vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2167-2175).

109. Liu, X., Liu, W., Ma, H., & Fu, H. (2016, July). Large-scale vehicle re-identification in urban surveillance videos. In *2016 IEEE international conference on multimedia and expo (ICME)* (pp. 1-6). IEEE.

110. Liu, X., Liu, W., Mei, T., & Ma, H. (2016). A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14* (pp. 869-884). Springer International Publishing.

111. Lou, Y., Bai, Y., Liu, J., Wang, S., & Duan, L. (2019). Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3235-3243).

112. Bai, Y., Lou, Y., Gao, F., Wang, S., Wu, Y., & Duan, L. Y. (2018). Group-sensitive triplet embedding for vehicle reidentification. *IEEE Transactions on Multimedia*, *20*(9), 2385-2399.

113. Guo, H., Zhao, C., Liu, Z., Wang, J., & Lu, H. (2018, April). Learning coarse-to-fine structured feature embedding for vehicle re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 32, No. 1).

114. Tang, Z., Naphade, M., Liu, M. Y., Yang, X., Birchfield, S., Wang, S., ... & Hwang, J. N. (2019). Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8797-8806).

115. ElRashidy, A., Ghoneima, M., Abd El Munim, H. E., & Hammad, S. (2021, December). Recent Advances in Vision-based Vehicle Re-identification Datasets and Methods. In *2021 16th International Conference on Computer Engineering and Systems (ICCES)* (pp. 1-6). IEEE.

116. Song, Y., Liu, C., Zhang, W., Nie, Z., & Chen, L. (2020, July). View-Decision Based Compound Match Learning for Vehicle Re-identification in UAV Surveillance. In *2020 39th Chinese Control Conference (CCC)* (pp. 6594-6601). IEEE.

117. Szeliski, R. (2022). *Computer vision: algorithms and applications*. Springer Nature.

118. Armingol Moreno, J. M., De la Escalera Hueso, A., & Garcia Fernández, F. (2017). Procesamiento de imagen por computador. *Universidad Carlos III,* Leganés, Madrid.

119. Merklinger, H. M. (1996). View Camera Focus and Depth of Field—Part II.

120. Merklinger, H. M. (2002). The ins and outs of focus. *Internet Edition*.

121. Smeulders, A. W., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, *22*(12), 1349-1380.

122. BateriasOnline. (s.f.) Batería LiFePO4 12V 14.5Ah INNPO - Baterías Bicicletas Eléctricas. Recuperado el 21 de mayo de 2023 de https://bateriasonline.com/es/baterias-bicicletas-electricas/bateria-lifepo4-12v-145ah-innpo-baterias-bicicletas-electricas.html

123. Liu, Y., & Dey, S. (2003). A New Perspective on Video Streaming Over the Internet: A Congestion-Adaptive Framework for Real-Time Video Streaming. *IEEE Transactions on Multimedia*, 5(3), 356-369.

124. Gao, C., Wang, Y., Han, Y., Chen, W., & Zhang, L. (2022). IVP: An Intelligent Video Processing Architecture for Video Streaming. *IEEE Transactions on Computers*, *72*(1), 264-277.

125. Yu, W., & Zhao, C. (2019). Broad convolutional neural network based industrial process fault diagnosis with incremental learning capability. *IEEE Transactions on Industrial Electronics*, *67*(6), 5081-5091.

126. Xu, S., Wang, J., Shou, W., Ngo, T., Sadick, A. M., & Wang, X. (2021). Computer vision techniques in construction: a critical review. *Archives of Computational Methods in Engineering*, *28*, 3383-3397.

127. Cappelle, C., El Najjar, M. E. B., Pomorski, D., & Charpillet, F. (2007, June). Localisation in urban environment using GPS and INS aided by monocular vision system and 3D geographical model. In *2007 IEEE Intelligent Vehicles Symposium* (pp. 811-816). IEEE.

128. Crowther, K. G. (2004). Implementing a Perimeter Security Strategy: 3-Layer Perimeter Defense. *SANS Institute Reading Room*.

129. Russell, S., & Norvig, P. (2010). Artificial intelligence: A modern approach. *Prentice Hall Press*.

130. Nilsson, N. J. (2014). Principles of Artificial Intelligence. *Morgan Kaufmann Publishers*.

131. McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (2006). A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine*, *27*(4), 12-12.

132. Kurzweil, R. (2005). The singularity is near: When humans transcend biology. *Penguin Books.*

133. Mitchell, T. M. (1997). Machine Learning. *McGraw-Hill.*

134. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. *Springer*.

135. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. *Springe*r.

136. Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

137. Murty, M. N., & Devi, V. S. (2015). *Introduction to pattern recognition and machine learning* (Vol. 5). World Scientific.

138. Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255-260.

139. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436-444.

140. Haykin, S. (2009). Neural Networks and Learning Machines. *Pearson Education*.

141. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

142. Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (pp. 807-814).

143. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

144. Elman, J. L. (1990). Finding structure in time. *Cognitive science*, *14*(2), 179-211.

145. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

146. Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning* (pp. 1096-1103).

147. Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

148. Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc.".

149. Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). New York: springer.

150. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, *6*(1), 1-48.

151. Bottou, L. (2012). Stochastic gradient descent tricks. *Neural Networks: Tricks of the Trade: Second Edition*, 421-436.

152. Zhang, Z. (2018, June). Improved Adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)* (pp. 1-2). IEEE.

153. Zou, F., Shen, L., Jie, Z., Zhang, W., & Liu, W. (2019). A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11127-11135).

154. Barbedo, J. G. A. (2018). Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Computers and electronics in agriculture*, *153*, 46-53.

155. Russell, S. J. (2010). *Artificial intelligence a modern approach*. Pearson Education, Inc.

156. Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427-437.

157. O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

158. Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)* (pp. 1-6). IEEE.

159. Ajit, A., Acharya, K., & Samanta, A. (2020, February). A review of convolutional neural networks. In *2020 international conference on emerging trends in information technology and engineering (ic-ETITE)* (pp. 1-5). IEEE.

160. Xin, R., Zhang, J., & Shao, Y. (2020). Complex network classification with convolutional neural network. *Tsinghua Science and technology*, *25*(4), 447-457.

161. Loomis, L. H., & Sternberg, S. (2018). Advanced Calculus. *World Scientific Publishing Company*.

162. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321-357.

163. Bengio, Y., & Grandvalet, Y. (2004). No Unbiased Estimator of the Variance of K-Fold Cross-Validation. *Journal of Machine Learning Research*, 5, 1089-1105.

164. Guyon, I., & Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3, 1157-1182.

165. Fernández, A., García, S., Herrera, F., & Chawla, N. V. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. Journal of Artificial Intelligence Research, 61, 863-905.

166. Kohavi, R. (1995, August). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai* (Vol. 14, No. 2, pp. 1137-1145).

167. Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13* (pp. 818-833). Springer International Publishing.

168. Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1717-1724).

169. Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

170. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248-255). IEEE.

171. Dean, J., Corrado, G.S., Monga, R., Chen, K., Devin, M., Le, Q.V., Mao, M.Z., Ranzato, M., Senior, A.W., Tucker, P.A., Yang, K., & Ng, A. (2012). Large Scale Distributed Deep Networks. *NIPS*.

172. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84-90.

173. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P.A., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., & Zhang, X. (2016). TensorFlow: A system for large-scale machine learning. *ArXiv, abs/1605.08695*.

174. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. *ArXiv, abs/1912.01703*. Chetlur, S., Woolley, C., Vandermersch, P., Cohen, J., Tran, J., Catanzaro, B., & Shelhamer, E. (2014). cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*.

175. Chetlur, S., Woolley, C., Vandermersch, P., Cohen, J., Tran, J., Catanzaro, B., & Shelhamer, E. (2014). cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*.

176. Jocher G. YoloV5 by Ultralytics. Avaliable online: https://github.com/ultralytics/yolov5 (accesed on 4 July 2022).

177. Cherifi, I. (s.f.) YOLO v5 model architecture [Explained]. Recuperado el 21 de mayo de 2023 de https://iq.opengenus.org/yolov5/

178. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).

179. Xu, R., Lin, H., Lu, K., Cao, L., & Liu, Y. (2021). A forest fire detection system based on ensemble learning. *Forests*, *12*(2), 217.

180. Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

181. Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 390-391).

182. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, *37*(9), 1904-1916.

183. Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8759-8768).

184. Ramachandran, P., Zoph, B., & Le, Q. V. (2017). Searching for Activation Functions. arXiv preprint arXiv:1710.05941. [https://arxiv.org/abs/1710.05941].

185. Lin, T., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C.L. (2014). Microsoft COCO: Common Objects in Context. *European Conference on Computer Vision*.

186. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

187. Loshchilov, I., & Hutter, F. (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

188. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

189. Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1492-1500).

190. Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., ... & Smola, A. (2022). Resnest: Split-attention networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2736-2746).

191. Wang, X., Girshick, R., Gupta, A., & He, K. (2018). Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7794-7803).

192. Pan, X., Luo, P., Shi, J., & Tang, X. (2018). Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 464-479).

193. Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4690-4699).

194. Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., & Wei, Y. (2020). Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6398-6407).

195. Jocher, G., Chaurasia, A., Qiu, J. (2023). YOLO by Ultralytics. *Ultralytics.* Recuperado el 21 de mayo de 2023 de https://github.com/ultralytics/ultralytics

196. Zhang, T., Aftab, W., Mihaylova, L., Langran-Wheeler, C., Rigby, S., Fletcher, D., ... & Bosworth, G. (2022). Recent advances in video analytics for rail network surveillance for security, trespass and suicide prevention—A survey. *Sensors*, *22*(12), 4324.

197. Balasundaram, A., & Chellappan, C. (2020). An intelligent video analytics model for abnormal event detection in online surveillance video. *Journal of Real-Time Image Processing*, *17*(4), 915-930.

198. Tu, P., Wheeler, F., Krahnstoever, N., Sebastian, T., Rittscher, J., Liu, X., Perera, A.G., & Doretto, G. (2007). Surveillance video analytics for large camera networks. *SPIE Newsroom*.

199. Andonie, R. (2019). Hyperparameter optimization in learning systems. *Journal of Membrane Computing*, *1*(4), 279-291.

200. Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295-316.

201. Wang, Q., Ma, Y., Zhao, K., & Tian, Y. (2020). A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, 1-26.

202. Ji, Y., Zhang, H., Zhang, Z., & Liu, M. (2021). CNN-based encoder-decoder networks for salient object detection: A comprehensive review and recent advances. *Information Sciences*, *546*, 835-857.

203. Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*.

204. Arata, M. J. (2006). *Perimeter security*. Mcgraw-hill.

205. Griffin, S., Brierley, D., & Waters, M. (2004). Perimeter security: fences, hedges and gates. *Practical Professional Child Care*, *1*(2).