

Universidad de Lima  
Facultad de Ingeniería  
Carrera de Ingeniería Civil



# **OPCIONES DE BASES DE DATOS PARA LA DETECCIÓN DE TRABAJADORES APLICADA A LA INGENIERÍA CIVIL**

Tesis para optar el Título Profesional de Ingeniero Civil

**Jesus Hector Suelperes Rodriguez**

**Código 20171513**

**Asesor**

Ana Felicita Luna Torres

Lima – Perú

Mayo del 2025





**OPCIONES DE BASES DE DATOS PARA LA  
DETECCIÓN DE TRABAJADORES  
APLICADA A LA INGENIERÍA CIVIL**

# INDICE DE CONTENIDO

<b>RESUMEN .....</b>	<b>X</b>
<b>ABSTRACT.....</b>	<b>XI</b>
<b>INTRODUCCIÓN .....</b>	<b>1</b>
Planteamiento del problema: .....	1
Justificación:.....	2
Objetivos: .....	4
Hipótesis / Pregunta de Investigación:.....	4
<b>1. MARCO TEÓRICO Y ESTADO DEL ARTE .....</b>	<b>5</b>
1.1 Marco Teórico .....	5
1.2 Estado del Arte .....	17
<b>2. METODOLOGÍA .....</b>	<b>31</b>
<b>3. RESULTADOS DEL ENTRENAMIENTO EN YOLOV4.....</b>	<b>37</b>
<b>4. CONCLUSIONES .....</b>	<b>55</b>
<b>5. REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>56</b>
<b>6. ANEXOS.....</b>	<b>61</b>

# INDICE DE TABLAS

<b>TABLA 1.1</b>	<b><i>ESTADO DEL ARTE – COMPARACIÓN DE MODELOS DE DETECCIÓN Y APLICACIONES DE LA INTELIGENCIA ARTIFICIAL EN LA INGENIERÍA CIVIL</i></b>	
	¡ERROR! MARCADOR NO DEFINIDO.	
<b>TABLA 1.2</b>	<b><i>RESULTADO DE LOS MODELOS ESTUDIADOS</i></b> .....	<b>21</b>
<b>TABLA 1.3</b>	<b><i>RESULTADOS DE DETECCIÓN</i></b> .....	<b>22</b>
<b>TABLA 1.4</b>	<b><i>COMPARACIÓN ENTRE YOLOV3 Y YOLO-DENSE BACKBONE</i></b> ..	<b>24</b>
<b>TABLA 1.5</b>	<b><i>RESULTADOS YOLOV3 VS YOLOV3 CON EL MODELO SPZ</i></b> .....	<b>24</b>
<b>TABLA 2</b>	<b><i>ESPECIFICACIONES DE LAS CÁMARAS</i></b> .....	<b>31</b>
<b>TABLA 3.1</b>	<b><i>RESULTADOS DE %MAP EN YOLOV4</i></b> .....	<b>37</b>
<b>TABLA 3.2</b>	<b><i>NÚMERO DE ELEMENTOS NO DETECTADOS Y PRECISIÓN PROMEDIO EN LA IMAGEN IMG-729</i></b> .....	<b>45</b>
<b>TABLA 3.3</b>	<b><i>COMPARACIÓN ENTRE EL SEGUNDO Y CUARTO ENTRENAMIENTO RESPECTO AL NÚMERO DE ELEMENTOS NO DETECTADOS Y LA PRECISIÓN PROMEDIO</i></b> .....	<b>46</b>
<b>TABLA 3.4</b>	<b><i>RESULTADOS DEL MODELO YOLOV4 DESPUÉS DEL CUARTO ENTRENAMIENTO DESDE LA FIGURA 3.18 HASTA LA FIGURA 3.22</i></b> .....	<b>53</b>

## INDICE DE FIGURAS

<b>FIGURA A</b> <i>PROYECCIÓN DE DEMANDA DE LA INTELIGENCIA ARTIFICIAL EN LOS PRÓXIMOS 3 AÑOS</i> .....	<b>2</b>
<b>FIGURA B</b> <i>PESO EN EL VAB DE LA INDUSTRIA MANUFACTURERA Y DEL SECTOR TIC (EN%)</i> .....	<b>3</b>
<b>FIGURA 1.1</b> <i>ARQUITECTURA DE YOLOV4</i> .....	<b>9</b>
<b>FIGURA 1.2</b> <i>ESQUEMA GRÁFICO DE LA OBTENCIÓN DE LAS COORDENADAS DE UN CUADRO DELIMITADOR</i> .....	<b>10</b>
<b>FIGURA 1.3</b> <i>ESTRUCTURA DE UN PERCEPTRÓN</i> .....	<b>13</b>
<b>FIGURA 1.4</b> <i>ESTRUCTURA DE UNA MULTICAPA PERCEPTRÓN</i> .....	<b>13</b>
<b>FIGURA 1.5</b> <i>ESTRUCTURA DE UNA RED NEURONAL CONVOLUCIONAL</i> ....	<b>14</b>
<b>FIGURA 1.6</b> <i>EXPLICACIÓN GRÁFICA DE UN IOU</i> .....	<b>16</b>
<b>FIGURA 2.1</b> <i>IMÁGENES QUE SERÁN CLASIFICADAS</i> .....	<b>32</b>
<b>FIGURA 2.2</b> <i>CLASIFICACIÓN MANUAL EN LABELME</i> .....	<b>33</b>
<b>FIGURA 2.3</b> <i>CLASIFICACIÓN MANUAL EN LABELIMG</i> .....	<b>33</b>
<b>FIGURA 2.4</b> <i>CLASIFICACIÓN CLASE PERSON</i> .....	<b>34</b>
<b>FIGURA 2.5</b> <i>CLASIFICACIÓN CLASE LEANING_PERSON</i> .....	<b>34</b>
<b>FIGURA 2.6</b> <i>DATOS DE LAS ETIQUETAS EN FORMATO “.TXT”</i> .....	<b>34</b>
<b>FIGURA 2.7</b> <i>DATOS DE LAS ETIQUETAS EN FORMATO “.XML”</i> .....	<b>35</b>
<b>FIGURA 2.8</b> <i>DATOS DE LAS ETIQUETAS EN FORMATO “.JSON”</i> .....	<b>36</b>
<b>FIGURA 3.1</b> <i>YOLOV4 RESULTADOS DEL PRIMER ENTRENAMIENTO</i> .....	<b>38</b>
<b>FIGURA 3.2</b> <i>YOLOV4 RESULTADOS DEL SEGUNDO ENTRENAMIENTO</i> .....	<b>38</b>
<b>FIGURA 3.3</b> <i>YOLOV4 RESULTADOS DEL TERCER ENTRENAMIENTO</i> .....	<b>39</b>
<b>FIGURA 3.4</b> <i>YOLOV4 RESULTADOS DEL CUARTO ENTRENAMIENTO</i> .....	<b>39</b>

<b>FIGURA 3.5</b> <i>EVALUACIÓN EN LA IMAGEN IMG-729 AL FINAL DEL PRIMER ENTRENAMIENTO</i> .....	<b>41</b>
<b>FIGURA 3.6</b> <i>EVALUACIÓN EN LA IMAGEN IMG-729 AL FINAL DEL SEGUNDO ENTRENAMIENTO</i> .....	<b>42</b>
<b>FIGURA 3.7</b> <i>EVALUACIÓN EN LA IMAGEN IMG-729 AL FINAL DEL TERCER ENTRENAMIENTO</i> .....	<b>43</b>
<b>FIGURA 3.8</b> <i>EVALUACIÓN EN LA IMAGEN IMG-729 AL FINAL DEL CUARTO ENTRENAMIENTO</i> .....	<b>44</b>
<b>FIGURA 3.9</b> <i>ELEMENTOS NO DETECTADOS EN LA IMG-729</i> .....	<b>45</b>
<b>FIGURA 3.10</b> <i>EVALUACIÓN EN LA IMAGEN IMG-460 AL FINAL DEL CUARTO ENTRENAMIENTO</i> .....	<b>47</b>
<b>FIGURA 3.11</b> <i>EVALUACIÓN EN LA IMAGEN IMG-460 AL FINAL DEL SEGUNDO ENTRENAMIENTO</i> .....	<b>47</b>
<b>FIGURA 3.12</b> <i>EVALUACIÓN EN LA IMAGEN IMG-894 AL FINAL DEL CUARTO ENTRENAMIENTO</i> .....	<b>48</b>
<b>FIGURA 3.13</b> <i>EVALUACIÓN EN LA IMAGEN IMG-894 AL FINAL DEL SEGUNDO ENTRENAMIENTO</i> .....	<b>48</b>
<b>FIGURA 3.14</b> <i>EVALUACIÓN EN LA IMAGEN IMG-989 AL FINAL DEL CUARTO ENTRENAMIENTO</i> .....	<b>49</b>
<b>FIGURA 3.15</b> <i>EVALUACIÓN EN LA IMAGEN IMG-989 AL FINAL DEL SEGUNDO ENTRENAMIENTO</i> .....	<b>49</b>
<b>FIGURA 3.16</b> <i>EVALUACIÓN EN LA IMAGEN IMG-1058 AL FINAL DEL CUARTO ENTRENAMIENTO</i> .....	<b>50</b>
<b>FIGURA 3.17</b> <i>EVALUACIÓN EN LA IMAGEN IMG-1058 AL FINAL DEL SEGUNDO ENTRENAMIENTO</i> .....	<b>50</b>
<b>FIGURA 3.18</b> <i>EVALUACIÓN DE UN PROYECTO USANDO EL CUARTO ENTRENAMIENTO</i> .....	<b>51</b>
<b>FIGURA 3.19</b> <i>EVALUACIÓN DE UN PROYECTO 2 USANDO EL CUARTO ENTRENAMIENTO</i> .....	<b>51</b>

<b>FIGURA 3.20</b> <i>EVALUACIÓN DE UN PROYECTO 3 USANDO EL CUARTO ENTRENAMIENTO</i> .....	<b>52</b>
<b>FIGURA 3.21</b> <i>EVALUACIÓN DE UN PROYECTO 4 USANDO EL CUARTO ENTRENAMIENTO</i> .....	<b>52</b>
<b>FIGURA 3.22</b> <i>EVALUACIÓN DE UN PROYECTO 5 USANDO EL CUARTO ENTRENAMIENTO</i> .....	<b>52</b>



## INDICE DE ANEXOS

ANEXO 1: IMÁGENES DE LA BASE DE DATOS .....	62
---	----



## RESUMEN

La industria de la construcción a diferencia de otras industrias no tiene muy presente a la industria 4.0, específicamente el uso de inteligencia artificial en busca de la automatización de procesos que permitan ser eficientes en el desarrollo de un proyecto de construcción, debido a la falta de categorización de datos. Hoy en día las distintas metodologías existentes para llevar a cabo un proyecto permiten tener una mejor gestión de toda la información que genera dicho proyecto, y es esa información la que puede ser utilizada como base de datos para la inteligencia artificial que puede ser aplicada en las distintas etapas de un proceso constructivo desde planificación hasta monitoreo y control. Se desarrolló una base de datos en tres distintos formatos “txt”, “PASCAL VOC” y “COCO” en donde se busca identificar a los trabajadores de construcción en distintas posturas con el fin de realizar seguimiento de personal, control de calidad, prevención de accidentes y conflictos, y revisar las condiciones de seguridad en aplicaciones futuras con el fin de garantizar su seguridad y un mejor monitoreo. La base de datos presentada se utilizó para entrenar el algoritmo de clasificación y detección de objetos YOLOv4. La base de datos está compuesta de 1214 imágenes, estas imágenes brindan información importante de los procesos en obra, tomadas por cámaras ubicadas a 8 metros y a 50 metros de distancia del nuevo Centro de Bienestar Universitario (CBU) de la Universidad de Lima.

**Palabras Clave:** Inteligencia Artificial, YOLOv4, Aprendizaje Profundo, Detección a tiempo real, Industria 4.0

## ABSTRACT

The construction industry, unlike other industries, is not very aware of Industry 4.0, specifically the use of artificial intelligence in search of process automation that allows to be efficient in the development of a construction project, due to the lack of data categorization. Nowadays, the different existing methodologies to carry out a project allow to have a better management of all the information generated by such project, and it is that information that can be used as a database for artificial intelligence that can be applied in the different stages of a construction process from planning to monitoring and control. A database was developed in three different formats "txt", "PASCAL VOC" and "COCO" where the aim is to identify construction workers in different positions in order to track personnel, quality control, accident and conflict prevention, and review safety conditions in future applications in order to ensure their safety and better monitoring. The presented database was used to train the YOLOv4 object classification and detection algorithm. The database is composed of 1214 images, these images provide important information of the processes on site, taken by cameras located 8 meters and 50 meters away from the new University Welfare Center (CBU) of the University of Lima.

**Keywords:** Artificial intelligence, YOLOv4, Deep learning, Real-time sensing, Industry 4.0

# INTRODUCCIÓN

## **Planteamiento del problema:**

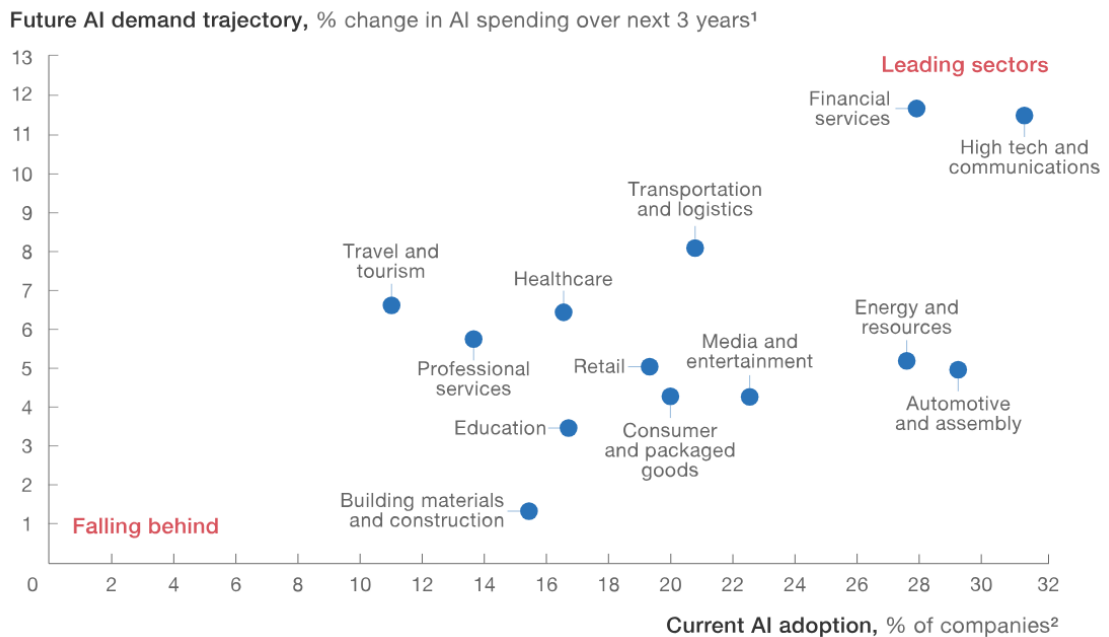
En la industria de la construcción, cada vez hay más aplicaciones de técnicas de Inteligencia Artificial, que de diversa forma pueden ayudar a resolver problemas de ingeniería, y mejorar la eficacia de los procesos. Sin embargo, en la ingeniería civil estas aplicaciones siguen siendo relativamente nuevas (Aleman, 2017). Muchas de esas aplicaciones están relacionadas con el monitoreo de estructuras, procesos de autodiagnóstico, la durabilidad del concreto y predicción de la resistencia a la compresión, a su vez que se han ido desarrollando algoritmos, así como el uso de redes neuronales en búsqueda de optimizar la Inteligencia Artificial en la Ingeniería Civil (Lu et al., 2012).

Las redes neuronales se han utilizado en una gran variedad de aplicaciones de procesamiento de datos, donde se requiere extracción de información y análisis de datos en tiempo real. Teniendo como ventaja que la carga computacional que requiere es solo en la etapa de entrenamiento, una vez entrenada, su operación es relativamente rápida. Las redes neuronales requieren de una significativa cantidad de datos etiquetados y categorizados para un buen análisis y funcionamiento (Ludermir y Yamazaki, 2003). Las redes neuronales han tenido un nivel significativo de participación en la ingeniería civil, brindando soluciones precisas a los problemas presentes en esta industria, el problema está en el poco interés en catalogar la información utilizándolo solo para funciones simples (Flood, 2008).

Como se puede ver en el siguiente gráfico, la demanda en la aplicación de la Inteligencia artificial en la industria de la Ingeniería Civil es muy baja, a diferencia del resto de industrias donde existe una mayor demanda. Existen cinco limitantes, las dos principales son: el Data Labeling, que se refiere a la categorización de los datos, y el Masive training Data Sets, que se refiere a que se requiere de grandes cantidades de información en las bases de datos para un buen funcionamiento.

## Figura A

### Proyección de demanda de la inteligencia artificial en los próximos 3 años



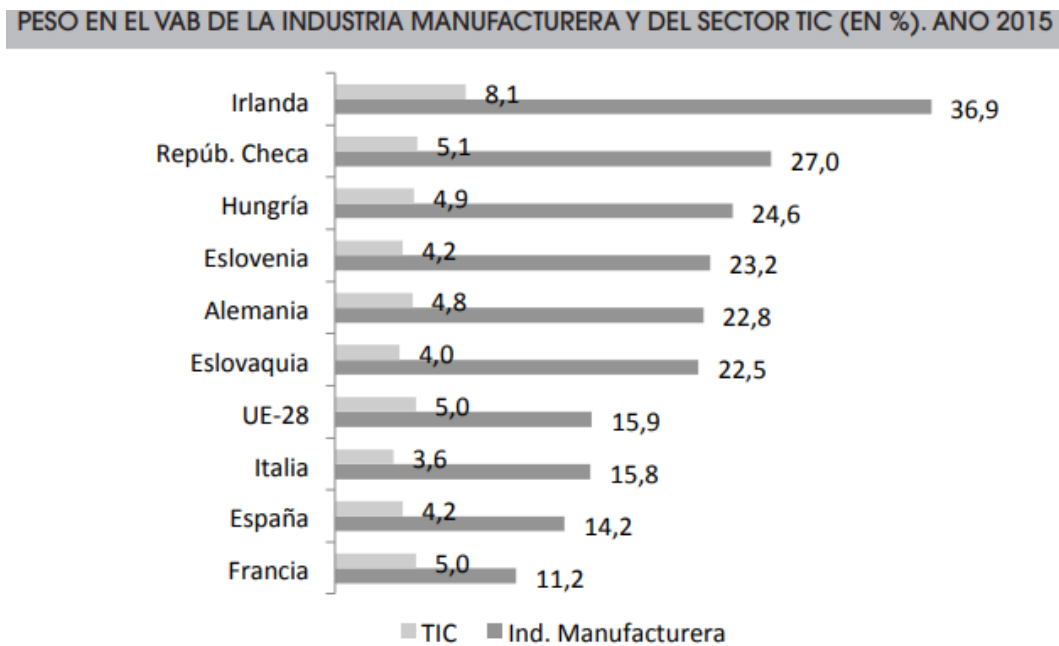
Nota. De “What AI can and can’t do (yet) for your business” por Chui, M., Manyika, J., & Miremadi, M., 2018, *McKinsey Global Institute AI adoption and Survey*, ([https://www.mckinsey.com/capabilities/quantumblack/our-insights/what-ai-can-and-cant-do-yet-for-your-business?cid=other-soc-fce-mip-mck-oth-1801&kui=JM\\_dauoGBH5BzmJfYriUSw&fbclid=IwAR2AhTWXXEPxp9UUWbOZ2aMNvvaTbodjuCPTgZyqsMNo6-IBg-aNU8GpY7U#/](https://www.mckinsey.com/capabilities/quantumblack/our-insights/what-ai-can-and-cant-do-yet-for-your-business?cid=other-soc-fce-mip-mck-oth-1801&kui=JM_dauoGBH5BzmJfYriUSw&fbclid=IwAR2AhTWXXEPxp9UUWbOZ2aMNvvaTbodjuCPTgZyqsMNo6-IBg-aNU8GpY7U#/))

### Justificación:

La industria 4.0 se ha convertido en la clave de la economía industrial, se le conoce como el proceso de transformación digital de la industria. Es el uso de estas tecnologías lo que facilita la productividad y eficiencia de las industrias, su uso está muy presente en la industria manufacturera (Blanco et al., 2017).

## Figura B

Peso en el VAB de la industria manufacturera y del sector TIC (en%)



Nota. De "La industria 4.0: El estado de la cuestión" por R. Blanco, J. Fontodrona, y C. Poveda, 2017, *Economía industrial*, 406, p. 159 (<https://dialnet.unirioja.es/servlet/articulo?codigo=6343649>)

Algunos de los elementos de la industria 4.0 ya están presentes en la industria de la construcción como la simulación, el uso de software como AutoCad y Revit que permite previsualizar el proyecto final y detectar interferencias al revisar los planos de la obra (Pacheco, 2017). Es por ello que este trabajo busca promover el uso de más elementos de la industria 4.0 como es la inteligencia artificial, donde se desarrollará una base de datos a partir de imágenes de procesos constructivos en obra, en busca de catalogar al personal de construcción a 8 metros y a 50 metros de distancia, este tipo de información no siempre se encuentra disponible. Esta base de datos será una herramienta valiosa como parte de diferentes análisis a realizar en los procesos de obra, como condiciones de seguridad, seguimiento de personal, eficiencia en la planificación de recursos, control de calidad, prevención de accidentes y conflictos, y eficiencia en la toma de decisiones. Se utilizará la red neuronal YOLOv4 con el fin de comprobar que la base de datos sea eficiente para que en futuros trabajos tengan datos previos con los cuales trabajar y desarrollar aplicaciones más complejas, así como facilitar la tarea de la clasificación manual. Este desarrollo tendría un aporte para la agenda 2030 correspondiente al plan BIM Perú, centrándose en el desarrollo de la innovación que es el objetivo 9 de dicha agenda (Ministerio de Economía y Finanzas, 2021).

## **Objetivos:**

Objetivo principal:

- Generar una base de datos del personal de construcción, la cual pueda ser utilizada para generar modelos IA que detecten al personal en distintas posiciones.

Objetivos secundarios:

- Utilizar las herramientas de anotación Labelme y Labelimg para etiquetar personal de construcción en las imágenes de la base de datos en formato “.txt”, “.xml” y “.JSON”.
- Analizar los resultados obtenidos del modelo IA entrenado con la base de datos generada para validar su precisión y confiabilidad.
- Identificar la importancia de las bases de datos en la aplicación de la inteligencia artificial en la ingeniería civil.

## **Hipótesis / Pregunta de Investigación:**

Por medio del uso del modelo de detección YOLOv4, basado en inteligencia artificial, entrenado con la base de datos generada, es posible la detección del personal de construcción en un entorno constructivo a una distancia cercana y lejana mediante imágenes o videos a tiempo real por medio de cámaras estratégicamente colocadas en los lados de la obra a una altura conveniente.

Pregunta principal:

¿Cuáles son las opciones de bases de datos para la detección de personal de construcción en un entorno constructivo?

Preguntas específicas:

¿Cómo se crea una base de datos para el entrenamiento de una inteligencia artificial?

¿Puede la base de datos presentada identificar correctamente al personal de construcción en un entorno constructivo?

¿Cuáles son los beneficios que trae la generación de bases de datos en Ingeniería Civil?

# 1. MARCO TEÓRICO Y ESTADO DEL ARTE

Antes del correcto desarrollo de la investigación es necesario entender cada uno de los conceptos que serán empleados para conocer el funcionamiento, entrenamiento y resultados de la inteligencia artificial. Es importante analizar los trabajos previos existentes en los que se obtuvieron resultados positivos o negativos en donde se hace uso de la inteligencia artificial para la detección de objetos.

Los orígenes de la inteligencia artificial se remontan al año 1936 cuando el matemático Alan Turing fue capaz de crear una máquina que podía hacer cálculos definidos que se adaptaban a cualquier escenario. Sin embargo, Turing recién escribió un libro en 1950 relacionado con el tema. En 1956, John McCarthy, Marvin Minsky y Claude Shannon, utilizaron el término “Inteligencia Artificial” dándole el significado de hacer inteligente a las máquinas, en especial aplicaciones de cálculo inteligentes, pero la expansión de la inteligencia artificial no llegaría hasta la década de los 90 (VectorITC, 2018).

## 1.1 Marco Teórico:

**1.1.1 Inteligencia:** Para el presente trabajo de investigación se definirá a la inteligencia como cualquier tipo de sistema que tiene la capacidad de lograr un objetivo o de mantener un comportamiento deseado en condiciones de incertidumbre. Es decir, la inteligencia es la capacidad de adaptación a un entorno para resolver problemas (Mathivet, 2018).

**1.1.2 Los Sistemas Inteligentes:** Según León y García (2008) los sistemas inteligentes se diferencian de los convencionales en que pueden manejar el conocimiento estructurado y el conocimiento empírico, no procesan datos, procesan conocimiento y lo representan de forma adecuada, explican la línea del razonamiento que siguen y justifican sus conclusiones a las que llegan.

**1.1.3 Agentes Inteligentes:** Se caracterizan por actuar en un ambiente determinado y realizar ciertas acciones basadas en percepciones tomadas por el ambiente, hasta cumplir el objetivo por el que fueron diseñados. Un agente es todo lo que percibe en su ambiente mediante sensores. Un agente inteligente es reactivo, pro activo y social, tiene la capacidad de responder a cambios que ocurren en el entorno en el que se encuentra, también es capaz de cumplir los planes y objetivos designados, y además, debe ser capaz de poder comunicarse con otros agentes por medio de un tipo de lenguaje entre agentes (López, 2005).

#### **1.1.4 Características de un Comportamiento Inteligente:** Según López (2005)

- **Autonomía:** Capacidad de actuar de forma independiente de otros sistemas, o humanos. Debe ser capaz de actuar en base a la experiencia que tenga y adaptarse al entorno en el que se encuentre.
- **Sociabilidad:** Los agentes tienen la capacidad de interactuar con humanos u otros agentes mediante algún lenguaje en particular.
- **Reactividad:** Las percepciones captadas de su ambiente producen una acción específica, actúa de acuerdo a los cambios que se producen en determinada situación.
- **Proactividad:** Los agentes tienen capacidad de exhibir un comportamiento particular dependiendo de los objetivos planteados, debe tener la capacidad de controlar sus objetivos planteados sin importar en el entorno en el que se encuentre.
- **Continuidad:** Los agentes están constantemente ejecutando procesos (captando percepciones y ejecutando acciones) y desarrollando la función que le fue designada.
- **Benevolencia:** Capacidad de satisfacer solicitudes, debe tener la capacidad de ayudar a otros agentes si esto no genera conflicto con los intereses que le fueron designados.
- **Racionalidad:** El agente actuará de manera tal de satisfacer sus objetivos, realizará lo que considere correcto en base a los datos que reciba en el entorno en el que se encuentre.

- **Colaboración:** Al interactuar de manera constante con el usuario, los agentes solicitaran colaboración de estos constantemente con la finalidad de ejecutar acciones eficaces y eficientes.

**1.1.5 Prueba de Turing:** Según VectorITC (2018), la prueba Turing se diseñó para proporcionar la definición satisfactoria de inteligencia. El comportamiento inteligente es considerado como la capacidad de lograr un desempeño al mismo nivel que un humano en toda actividad cognitiva. Para aprobar la prueba Turing, tiene que poseer las siguientes capacidades:

- **Procesamiento del Lenguaje Natural:** (Natural Language Processing) capacidad para permitirle comunicarse con éxito en otro idioma humano.
- **Representación del Conocimiento:** (Knowledge Representation) capacidad para almacenar la información proporcionada antes o durante el interrogatorio.
- **Razonamiento Automatizado:** (Automated Reasoning) capacidad para responder preguntas y sacar nuevas conclusiones con la información obtenida.

**1.1.6 Aprendizaje Automático:** (Machine Learning) Según VectorITC (2018), es la capacidad de adaptarse a las nuevas circunstancias, para detectar y extrapolar patrones. Deben ser capaces de aprender por sí solas y auto programarse aprendiendo de su propia experiencia, combinan datos de entrada y situaciones del mundo real. En el aprendizaje automático existen varios métodos:

- **Aprendizaje Dirigido:** También llamado Supervised Learning, en este método los datos que son utilizados durante el entrenamiento incluyen la solución esperada, eso es llamado como “etiquetas” (labels). Por medio de este método, la inteligencia artificial recibe un estímulo, ejemplo o muestra, conocido como parseo de datos. El algoritmo realiza el procesamiento y a partir de esos datos extrae un modelo. Por medio del modelo antes mencionado, cada vez que se coloque un nuevo dato la máquina entrenada está en la capacidad de dar una respuesta.
- **Aprendizaje no Dirigido:** También llamado Unsupervised Learning, en este método los datos que son utilizados durante el entrenamiento no incluyen la

solución, es decir “etiquetas”, por lo que el algoritmo intentará descifrar o clasificar la información por sí mismo.

- **Aprendizaje por Refuerzo:** En este método, el sistema entrenado se convierte en un “agente autónomo”, su función es determinar las acciones que llevará a cabo por medio de prueba y error. Aprende por sí mismo y obtiene recompensas y penalizaciones, por lo que creará la mejor estrategia posible basado en las políticas que se le hayan asignado, para obtener la mayor recompensa que sea posible en tiempo y también en forma. Las políticas que se le asignan son las que van a definir qué acciones son las que serán tomadas en cada situación a la que se enfrente. Este método tiene distintas aplicaciones en diversas áreas como el marketing, la personalización, la ciber seguridad, en el sector financiero, análisis a tiempo real, en el área de la comunicación, motores de búsqueda, segmentación de clientes, para la estimación del consumo energético.

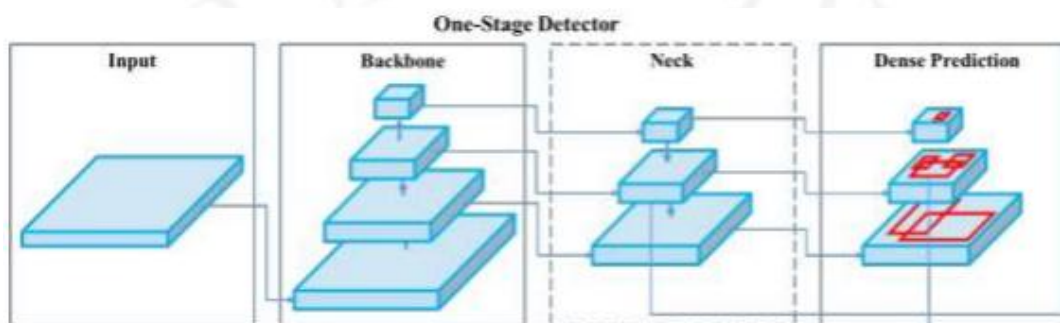
**1.1.7 Aprendizaje Profundo:** También llamado Deep Learning, es un tipo de algoritmo de aprendizaje automático estructurado. Este algoritmo toma modelos existentes para identificar una realidad o predecir el futuro con los datos que tenga a su disposición. La mayor diferencia que existe entre el aprendizaje profundo y el aprendizaje automático es que el aprendizaje profundo tiene un nivel de aprendizaje mucho más detallado. El sistema del aprendizaje profundo va por capas o unidades neuronales, el funcionamiento de estos algoritmos intenta simular el cerebro del ser humano. La predicción de este sistema se realiza por medio del aprendizaje y con reglas establecidas con las que fue programado con anterioridad, normalmente va ligado con el procesamiento de texto, voz, imagen y video, funciona como un sistema de redes neuronales que es capaz de analizar datos en busca de simular el cerebro humano. Este sistema evalúa instrucciones y ejemplos para cambiar el modelo en caso de que se produzca algún error. Estos modelos predictivos tienen mayor uso en el área de diagnósticos médicos y en análisis predictivos de los mercados financieros, pero cada vez se aumenta más su uso en distintos sectores. El aprendizaje profundo es usado en coches autónomos, para el análisis y generación de informes, análisis predictivo, traductores inteligentes, lenguaje natural y escrito, reconocimiento de voz, interpretación semántica, visión computacional y reconocimiento facial (VectorITC, 2018).

**1.1.8 YOLOv4:** Es un sistema de detección de objetos de código abierto que permite identificar objetos en imágenes o videos, es una herramienta que posee un modelo ya preentrenado que le permite detectar objetos en tiempo real, permite que se pueda crear un modelo personalizado (Miranda et al., 2019).

La arquitectura de YOLOv4 se compone de 3 etapas, Backbone, Neck y Head como se observa en la figura 1.1, YOLOv4 está construido sobre YOLOv3, agregando algoritmos que mejoran su rendimiento (Bochkovskiy et al., 2020).

**Figura 1.1**

*Arquitectura de YOLOv4*



*Nota.* De *Estudio de la arquitectura YOLO para la detección de objetos mediante deep learning* (p. 58) por S. Rozada, 2021 (<http://uvadoc.uva.es/handle/10324/45359>).

Después del Input, que son los datos que ingresan de la base de datos, para que entre en la etapa de Backbone, que consta de CSPDarknet53, que es una red convolucional conformada por CSPNet y Darknet53.

CSPNet permite hacer más livianas distintas arquitecturas de redes neuronales, necesita menos recursos para su funcionamiento que aumenta su compatibilidad con dispositivos informáticos de vanguardia, además de encontrar información redundante que le permite optimizar la velocidad sin comprometer la calidad del sistema. CSPNet incorpora el Maxout, es una operación para comprimir los mapas de características generados (Wang et al., 2020).

Darknet53, es una red neuronal convolucional que cuenta con 53 capas de profundidad, esto permite que pueda clasificar múltiples objetos además de poder almacenar gran cantidad de imágenes sin ningún problema. Entonces, CSPDarknet53, permite tener toda la capacidad que posee Darknet53, pero aumentando su velocidad de procesamiento sin comprometer su calidad.

La etapa Neck está compuesta por SPP y PANet. Las siglas de SPP significan Spatial Pyramid Pooling, es una solución para el problema que presentan las redes neuronales convolucionales que requieren de una entrada de tamaño fija. El módulo SPP permite manejar de forma sencilla distintos tamaños y relaciones de aspecto, para el reconocimiento visual con una gran precisión para las tareas de clasificación y detección sin perder precisión agrupando las principales características en una pirámide (He et al., 2015).

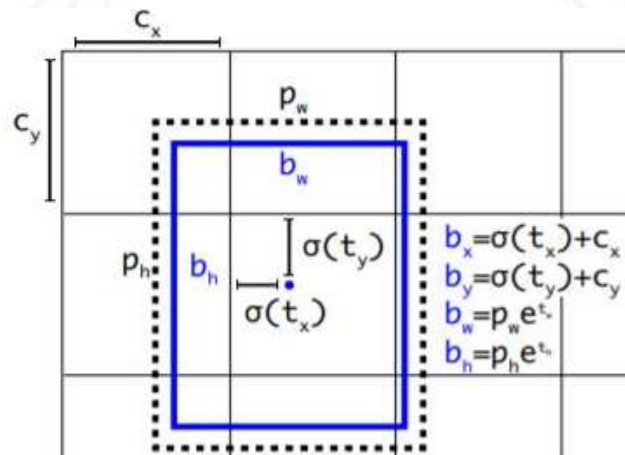
PANet, es una forma de propagar la información obtenida de las bases de datos en las redes neuronales, mejorando la propagación de dicha información mejorando la precisión de las redes neuronales, agrupando las características entre los niveles inferiores y superiores para transmitir la información. PANet se implementa como sustitución de las redes piramidales de características (Liu et al., 2018).

Esta combinación de SPP y PANet, permite que se maneje de mejor forma la transmisión de información de las características de los objetos de los cuadros delimitadores de las anotaciones, a la vez que se pueden incluir imágenes de diferentes tamaños y escalas.

Finalmente, la etapa del Head está conformado por YOLOv3. Para la predicción de cada cuadro delimitador se utilizan 4 coordenadas,  $t_x$ ,  $t_y$ ,  $t_w$ ,  $t_h$ , un alto ( $p_w$ ) y un ancho ( $p_h$ ). En la figura 1.2 se observa que  $c_x$ ,  $c_y$  son el desplazamiento de la celda respecto a la esquina superior izquierda de la imagen,  $b_x$  y  $b_y$  corresponden al centro del cuadro delimitador mientras que  $b_h$  y  $b_w$  corresponden al alto y ancho del mismo.

**Figura 1.2**

*Esquema gráfico de la obtención de las coordenadas de un Cuadro Delimitador*



*Nota.* De Estudio de la arquitectura YOLO para la detección de objetos mediante 10eep learning (p. 57) por S. Rozada, 2021 (<http://uvadoc.uva.es/handle/10324/45359>).

La imagen es dividida en una rejilla, y si la imagen cae dentro de una de las celdas, esa celda se encargará de detectar al objeto. El nivel de confianza se calcula tomando en cuenta el Intersection over Union (IoU), y teniendo en cuenta una serie de probabilidades. Estas probabilidades se basan en las celdas que contienen los objetos, en el momento de la detección la probabilidad de la clase se multiplica por cada cuadro delimitador, esto da la probabilidad de que el objeto este en dicho cuadro y que tanto se ajusta a este (Redmon et al., 2016).

Con respecto al cálculo de la pérdida, YOLO usa la fórmula del error de la suma cuadrada, entre los cuadros delimitadores y los cuadros delimitadores de las anotaciones. La fórmula de la pérdida final consta de la suma de las pérdidas de clasificación, localización, y las pérdidas de confianza.

**1.1.9 Inteligencia Artificial:** Es el estudio de las facultades mentales a través del uso de modelos computacionales, es una disciplina en la que se conjugan diversas ramas como la información, psicología, lingüística y neurofisiología, porque el objetivo de la inteligencia artificial es crear máquinas que piensen y actúen como humanos. La inteligencia artificial se divide en 4 categorías (Alemán, 2017):

- Sistemas que piensan como humanos, estos sistemas tratan de simular el pensamiento humano, un ejemplo de estos sistemas son las redes neuronales artificiales.
- Sistemas que actúan como humanos, estos sistemas buscan simular el actuar de los humanos y de su comportamiento, un ejemplo de estos sistemas es la robótica.
- Sistemas que piensan racionalmente, estos sistemas buscan imitar el pensamiento lógico racional del ser humano, un ejemplo de este sistema es el sistema de expertos.
- Sistemas que actúan racionalmente, estos sistemas buscan simular de forma racional el comportamiento del ser humano, un ejemplo de este sistema son los agentes inteligentes.

**1.1.10 Sistemas de Expertos:** Un sistema de expertos simula el razonamiento de un ser humano experto en algo en concreto mediante una aplicación informática, es un conjunto

de programas que funciona sobre una base de conocimientos formada por uno o más expertos de un área, codificando el conocimiento humano para que una computadora pueda aplicar ese conocimiento en situaciones similares. Sin embargo, un sistema de expertos no sustituye a un humano experto en el área, debido a que los humanos son capaces de aprender de los errores rápidamente, poseen sentido común y pueden mantener una conversación de forma fluida, son características difíciles de lograr por medio de un programa informático (Alemán, 2017).

**1.1.11 Redes Neuronales Artificiales:** Las redes neuronales artificiales son sistemas que procesan información, que simulan la forma en la que funciona el sistema nervioso. Son modelos electrónicos que están basados en la estructura neuronal del cerebro. Estas redes neuronales artificiales son utilizadas para resolver problemas para las tareas de clasificación, identificación, diagnóstico, optimización o predicción, de forma individual o combinada con otros métodos para completar la tarea asignada. En este tipo de sistemas se necesita mayor cantidad de datos, donde hay valor en el tiempo de ejecución y en la tolerancia a errores, en esos casos se adaptan las redes neuronales artificiales dinámicamente haciendo un constante reajuste en los pesos de sus interconexiones. La estructuración de las redes neuronales artificiales está basada en el teorema de Kolmogorov, esta estructura divide a la red neuronal en 3 capas, una de entrada, una capa oculta y una de salida, en las que se pueden aproximar hasta el nivel requerido cualquier función que sea continua en un intervalo que permite que la red neuronal pueda trabajar cualquier cantidad de funciones (Alemán, 2017).

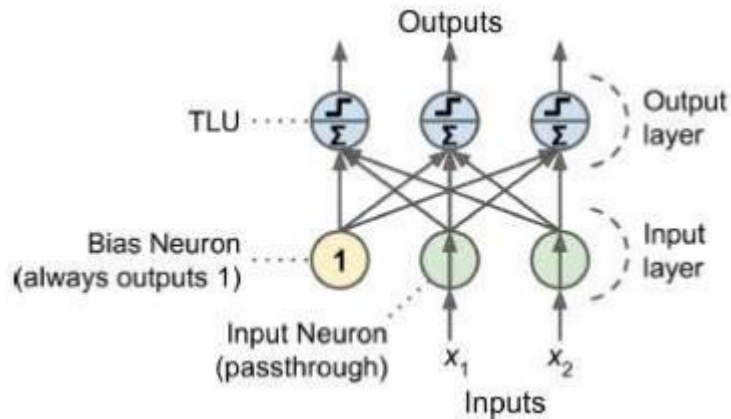
Según Marín (2012), las redes neuronales se crearon para simular los sistemas nerviosos biológicos, constituidos por neuronas y nodos que interconectan las neuronas. Son modelos bastante simplificados de las redes de neuronas que forman el cerebro. Y, al igual que este, intentan “aprender” a partir de los datos que se le suministran. Las principales características que diferencian a las redes neuronales de otras tecnologías de IA son: su capacidad de aprendizaje a partir de la experiencia, la velocidad de respuesta una vez concluido el entrenamiento y la capacidad de generar un número de respuestas correctas incluso si parte de la red neuronal artificial deja de funcionar.

Una neurona artificial, llamada también Thershold Logic Unit (TLU) o unidad lógica de umbral, se compone de entradas y de una salida. Las entradas son asociadas con

unos pesos que son sumados ponderadamente para obtener la salida. Una capa de TLUs con cada TLU conectada con cada entrada forma lo que se le conoce como perceptrón como se observa en la figura 1.3 (Rozada, 2021).

**Figura 1.3**

*Estructura de un Perceptrón*

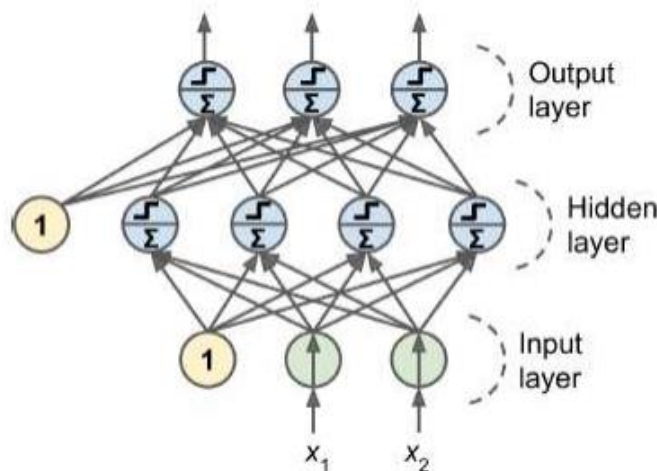


*Nota.* De Estudio de la arquitectura YOLO para la detección de objetos mediante 13eep learning (p. 24) por S. Rozada, 2021 (<http://uvadoc.uva.es/handle/10324/45359>).

La multicapa perceptrón tiene una o más capas añadidas que están ocultas, poseen una capa de entrada y una de salida como se observa en la figura 1.4, las capas cercanas a la entrada se le llaman inferiores y las capas cercanas a la salida se le llaman superiores.

**Figura 1.4**

*Estructura de una multicapa perceptrón*

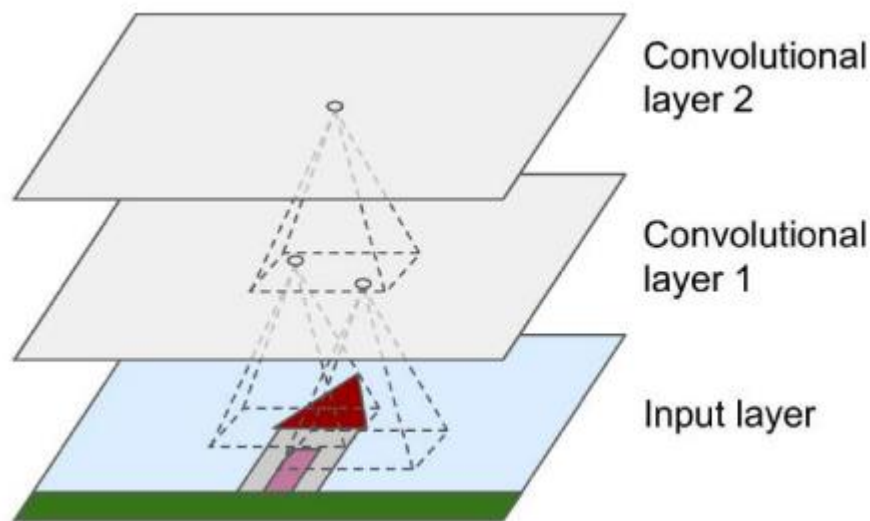


*Nota.* De Estudio de la arquitectura YOLO para la detección de objetos mediante 13eep learning (p. 25) por S. Rozada, 2021 (<http://uvadoc.uva.es/handle/10324/45359>).

En una red neuronal destinada a la detección de objetos, se usan las redes neuronales convolucionales debido al procesamiento de imágenes. Los píxeles de cada imagen actúan como entradas que trabajan con porciones determinadas de la imagen de la cual van extrayendo las características en una capa oculta como se puede apreciar en la figura 1.5 (Rozada, 2021).

**Figura 1.5**

*Estructura de una red neuronal convolucional*



*Nota.* De *Estudio de la arquitectura YOLO para la detección de objetos mediante 14ep learning* (p. 26) por S. Rozada, 2021 (<http://uvadoc.uva.es/handle/10324/45359>).

**1.1.12 Sistemas Borrosos:** Los sistemas borrosos se basan en la lógica difusa, que es un paradigma computacional que permite representar y manipular la información para que simule los procesos de comunicación y el razonamiento humano (Alemán, 2017).

**1.1.13 Algoritmos Genéticos:** Los algoritmos genéticos son procesos recurrentes que operan con un grupo de posibles soluciones llamado población genética. Se genera de forma aleatoria y las soluciones van en constante evolución después de varias etapas de mutaciones y cruzamientos. Cada individuo de la población genética tiene un valor relacionado con su capacidad para resolver el problema, ese valor es obtenido para cada solución potencial. La población genética constituirá una información cuantitativa que usará el algoritmo para guiar la búsqueda, el proceso continúa hasta que llega a un criterio determinado, que puede ser el número máximo de iteraciones (Alemán, 2017).

**1.1.14 Fast R-CNN:** Llamado como Fast Region-based Convolutional Network method, es un método basado en la detección de objetos, que se basa en trabajos previos para realizar una clasificación eficiente de los objetos propuestos utilizando redes convolucionales profundas, Fast R-CNN mejora la velocidad del entrenamiento y la velocidad de comprobación mientras aumenta la precisión de la detección. A diferencia de la clasificación de imágenes, la detección de objetos es más complicado debido a que requiere de métodos más complejos para resolver dicha tarea, por esa complejidad los enfoques actuales entrenan modelos en múltiples capas, lo que lo hace lento. La dificultad en la detección surge porque requiere de una localización precisa de los objetos, eso crea dos problemas, la gran cantidad de ubicaciones de los posibles objetos (propuestas) que son procesados, y el otro problema es que esas propuestas solo brindan una localización aproximada del objetivo. Solucionar esos problemas comprometen mucho la velocidad del sistema, la precisión o la simplicidad. Fast R-CNN optimiza el proceso para la detección de objetos, por medio de un algoritmo de entrenamiento de una sola etapa que aprende en conjunto a las propuestas que debe clasificar y mejora la precisión de las ubicaciones. Fast R-CNN de entrada usa una imagen completa y un conjunto de propuestas de objetos a identificar, procesa toda imagen con agrupaciones convolucionales para producir un mapa de características con las cuales luego realizará la localización (Girshick, 2015).

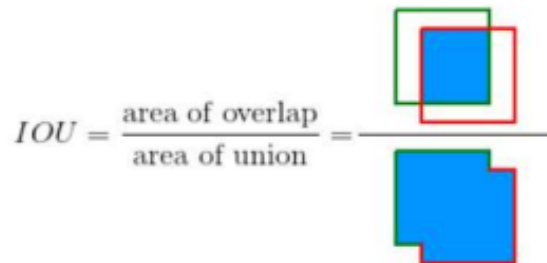
**1.1.15 Métricas de Rendimiento:** Es una serie de métricas que permiten saber si el sistema entrenado funciona, no es suficiente con que el sistema detecte los objetos buscados, sino que se debe evaluar su rendimiento. Las métricas de rendimiento son las siguientes (Rozada, 2021):

- **Intersección sobre Unión (IoU):** También llamado Intersection over Union, esta métrica evalúa qué porcentaje de los cuadros delimitadores (Bounding Box) se superponen. Existen dos cuadros delimitadores, uno llamado cuadro delimitador de las anotaciones (Groundbox), que es la anotación que se ha realizado sobre las imágenes con las cuales se entrena y valida la detección de objetos, y el otro cuadro delimitador se obtiene cuando el sistema realiza la detección de objetos. El IOU es el área entre los cuadros delimitadores en las que coinciden entre el área que unen cuadros delimitadores como se puede

observar en la figura 1.6. A menor sea el número del IoU menos fiable será su detección.

### Figura 1.6

*Explicación gráfica de un IOU*



*Nota.* De Estudio de la arquitectura YOLO para la detección de objetos mediante Deep learning (p. 33) por S. Rozada, 2021 (<http://uvadoc.uva.es/handle/10324/45359>).

- **Verdadero Positivo (TP):** Se produce cuando un objeto es detectado y su IOU es mayor o igual al umbral. Esto se considera una detección correcta.
- **Falso Positivo (FP):** Se produce cuando un objeto es detectado y su IOU es menor al umbral, lo que se le consideraría una detección incorrecta.
- **Falso Negativo (FN):** Se produce cuando existe un cuadro delimitador de la anotación en la imagen, pero no es detectada.
- **Verdadero Negativo (TN):** Los verdaderos negativos no se utilizan debido a que son todos los cuadros delimitadores posibles en los que no hay un objeto a detectar y en los que no los detecta, en la detección de objetos existe una infinidad de cuadros delimitadores que no deben ser detectados.
- **Umbral:** Es el valor establecido para que un objeto sea detectado, este valor generalmente es 50%, 75% o 95%.
- **Precisión:** La precisión es la medida que indica la calidad del modelo propuesto, es la fracción de cuadros delimitadores que contienen el objeto a detectar en relación con todos los objetos que encuentra. El valor ideal de esta medida sería el 1 que indicaría un 100% de precisión.
- **Recall:** El Recall indica la fracción de los objetos de los que verdaderamente hay, son encontrados por el modelo. Si el sistema es capaz de detectar los casos relevantes que existen en la imagen, se puede decir que es un sistema sensible.

- **Precisión Media (mAP):** Es una forma de encontrar si un modelo de detección de objetos es bueno o malo, esta medida se encuentra con la información de la precisión y recall.

## 1.2 Estado del Arte:

Se han realizado diferentes investigaciones y usos aplicando las redes neuronales artificiales para evaluar su funcionamiento, algunos de estos casos fueron llevados al campo de la construcción. Los estudios son los mostrados en la tabla 1.1.

**Tabla 1.1**

*Estado del Arte – Comparación de modelos de detección y aplicaciones de la inteligencia artificial en la ingeniería civil*

ID	TITULO	AUTORES	TIPO DE ESTUDIO	PAIS	AÑO
1	Accuracy and Efficiency Comparison of Object Detection Open-Source Models	Brahim Jabir, Nouredine Falih, Khalid Rahmani	Estudio de caso	Marruecos	2020
2	Contrasting YOLOv5, Transformer, and EfficientDet Detectors for Crop Circle Detection in Desert	Mohamed Lamine Mekhalfi, Carlo Nicolò, Yakoub Bazi, Mohamad Mahmoud Al Rahhal, Norah A. Alsharif, Eslam Al Maghayreh	Estudio de Caso	Reino Unido	2021
3	A remote human activity detection system based on partial-fiber LDV and PTZ camera	Xiyu Han, Tao Lv, Shisong Wu, Yuanyang Li, Bin He	Estudio de Caso	China	2019
4	Pedestrian Detection Using Multispectral Images and a DeepNeural Network	Jason Nataprawira, Yanlei Gu, Igor Goncharenko, Shunsuke Kamijo	Estudio de Caso	Japon	2021

(continúa)

(continuación)

<b>ID</b>	<b>TITULO</b>	<b>AUTORES</b>	<b>TIPO DE ESTUDIO</b>	<b>PAIS</b>	<b>AÑO</b>
5	<b>Social Distancing Detection and Analysis through Computer Vision</b>	<b>Sneha Madane, Dnyanoba Chitre</b>	<b>Estudio de Caso</b>	<b>India</b>	<b>2021</b>
6	<b>Social Distance Capturing and Alerting Tool</b>	<b>Mogula Yeshasvi, Veeramachaneni Bindu, Subetha T</b>	<b>Estudio de Caso</b>	<b>India</b>	<b>2021</b>
7	<b>Helmet Detection Based on Improved YOLO V3 Deep Model</b>	<b>Fan Wu, Guoqing Jin, Mingyu Gao, Zhiwei HE, Yuxiang Yang</b>	<b>Estudio de Caso</b>	<b>China</b>	<b>2019</b>
8	<b>Helmet Detection Algorithm Based on Single Pixel Zoom</b>	<b>Yongheng Zhou, Lifen Jiang, Yan Liang, Chunmei Ma, Huazhi Sun, ShuaiBao Nie, Ying Zuo</b>	<b>Estudio de Caso</b>	<b>China</b>	<b>2020</b>
9	<b>An improved helmet detection method for YOLOv3 on an unbalanced dataset</b>	<b>Rui Geng, Yixuan Ma, Wanhong Huang</b>	<b>Estudio de Caso</b>	<b>China</b>	<b>2020</b>
10	<b>Detection of Workers Without the Helmets in Videos Based on YOLO V3</b>	<b>Jing Hu, Xiaowei Gao, Hefeng Wu, Songhe Gao</b>	<b>Estudio de Caso</b>	<b>China</b>	<b>2019</b>
11	<b>Safety helmet detection method based on YOLO v4</b>	<b>Deng Benyang, Lei Xiaochun, Ye Miao</b>	<b>Estudio de Caso</b>	<b>China</b>	<b>2020</b>
12	<b>Construction worker detection in video frames for initializing vision trackers</b>	<b>Man-Woo Park, Ioannis Brilakis</b>	<b>Estudio de Caso</b>	<b>USA</b>	<b>2012</b>
13	<b>Human-object interaction recognition for automatic construction site safety inspection</b>	<b>Shuai Tang, Dominic Roberts, Mani Golparvar-Fard</b>	<b>Estudio de Caso</b>	<b>USA</b>	<b>2020</b>

(continúa)

(continuación)

<b>ID</b>	<b>TITULO</b>	<b>AUTORES</b>	<b>TIPO DE ESTUDIO</b>	<b>PAIS</b>	<b>AÑO</b>
14	The use of artificial intelligence to identify objects in a construction site	Almeida Del Savio, Ana Luna, Daniel Cárdenas-Salas, Mónica Vergara Olivera, Gianella Urday Ibarra	Estudio de Caso	Perú	2021
15	Automated detection of workers and heavy equipment on construction sites: A convolutional neural network approach	Weili Fang, Lieyun Ding, Botao Zhong, Peter E. D. Love, Hanbin Luo	Estudio de Caso	China	2018
16	CORY-Net: Contrastive Res-YOLOv5 Network for Intelligent Safety Monitoring on Power Grid Construction Sites	Guangyao Peng, Yin Lei, Haolin Li, Di Wu, Junyuan Wang, Fuqiang Liu	Estudio de Caso	China	2021
17	Artificial Intelligence Applied to the Control and Monitoring of Construction Site Personnel	Almeida Del Savio, Ana Luna, Daniel Cárdenas-Salas, Mónica Vergara Olivera, Gianella Urday Ibarra	Estudio de Caso	Perú	2023
18	Artificial intelligence-empowered pipeline for image-based inspection of concrete structures	Jun Kang Chow, Zhaoyu Su, Jimmy Wu, Zhaofeng Li, Pin Siang Tan, Kuan-fu Liu, Xin Mao, Yu-Hsing Wang	Estudio de Caso	China	2020
19	Deep learning-based roadway crack classification using laser-scanned range images: A comparative study on hyperparameter selection	Shanglian Zhou, Wei Song	Estudio de Caso	USA	2020
20	Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks	Young-Jin Cha, Wooram Choi, Oral Büyüköztürk	Estudio de Caso	Canada	2017

(continúa)

(continuación)

ID	TITULO	AUTORES	TIPO DE ESTUDIO	PAIS	AÑO
21	Prediction of Pavement Performance through Neuro - Fuzzy Reasoning	Alessandra Bianchini, Paola Bandini	Estudio de Caso	USA	2010
22	Detection of sealed and unsealed cracks with complex backgrounds using deep convolutional neural network	Ju Huyan, Wei Li, Susan Tighe, Junzhi Zhai, Zhengchao Xu, Yao Chen	Estudio de Caso	China	2019
23	NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naive Bayes Data Fusion	Fu-Chen Chen, Mohammad R. Jahanshahi	Estudio de Caso	China	2017
24	Technology-Related Challenges in Implementing Mixed Reality (MR) Technology in Precast Concrete Production	Mahsa Rezvani, Lloyd Waugh, Zhen Lei	Estudio de Caso	Canada	2022
25	Appearance-based material classification for monitoring of operation-level construction progress using 4D BIM and site photologs	Kevin K. Han, Mani Golparvar-Fard	Estudio de Caso	USA	2015
26	Estudio de Aplicaciones de la Inteligencia Artificial en el Desarrollo de Proyectos de Ingeniería Civil	Benjamín Caro Moreno	Estudio de Caso	Chile	2021

(continúa)

(continuación)

ID	TITULO	AUTORES	TIPO DE ESTUDIO	PAIS	AÑO
27	<b>Building applications for smart and safe construction with the DECENTER Fog Computing and Brokerage Platform</b>	<b>Petar Kochovski, Vlado Stankovski</b>	<b>Estudio de Caso</b>	<b>Slovenia</b>	<b>2021</b>
28	<b>Artificial intelligence in the construction industry: A review of present status, opportunities and future challenges</b>	<b>Sofiat O. Abioye, Lukumon O. Oyedele, Lukman Akanbi, Anuoluwapo Ajayi, Juan Manuel Davila Delgado, Muhammad Bilal, Olugbenga O. Akinade, Ashraf Ahmed</b>	<b>Estudio de Caso</b>	<b>UK</b>	<b>2021</b>

De estos trabajos se destaca lo siguiente:

Se realizó una comparación en la detección de maleza en cultivos con los modelos de detección de código abierto EfficientDet, YOLOv5, Detectron2 y Faster R-CNN. Para la creación de la base de datos se utilizaron 2000 imágenes de círculos de cultivo tomadas en condiciones de mañana y de tarde. Los resultados de la comparación demostraron que YOLOv5 es el modelo más rápido y liviano en el entrenamiento en la detección de objetos con una precisión del 82% como se aprecia en la tabla 1.2 (Jabir et al., 2020).

**Tabla 1.2**

*Resultado de los modelos estudiados*

Model	Type	Average precision on COCO dev-set	Number of training stages	Precision on our dataset	Error On our dataset	Training Time
EfficientDet	One stage detector	55.1	3500 steps	52%	38%	12 hours
YOLOv5	One stage detector	50.1	200 epochs	82%	30%	2 hours

(continúa)

(continuación)

Model	Type	Average precision on COCO dev-set	Number of training stages	Precision on our dataset	Error On our dataset	Training Time
Detectron2	One stage detector	64.31	1500 steps	97%	42%	5 hours
Faster R-CNN	Two stage detector	36.8	10 000 steps	45%	80%	10 hours

*Nota.* De “Accuracy and Efficiency Comparison of Object Detection Open-Source Models” por B. Jabir, N. Falih y K. Rahmani, 2021, *International Journal of Online and Biomedical Engineering (iJOE)*, 17(05), pp. 181 (<https://doi.org/10.3991/ijoe.v17i05.21833>)

Se realizó una comparación de las redes neuronales transformer, EfficientDet y YOLOv5, para la detección de círculos de cultivo de forma remota mediante imágenes obtenidas de Google Earth Pro sobre Egipto y Arabia Saudita a una altura de 20 km. El resultado al realizar dicha comparación demostró que YOLOv5 es capaz de detectar muchos más objetos independientemente de su precisión, demostró su eficacia con respecto a las otras redes neuronales en relación tiempo de detección – precisión como se aprecia en la tabla 1.3 (Melkhalfi et al., 2021).

**Tabla 1.3**

*Resultados de detección*

Method	KSA2EGY		EGY2KSA	
	Recall	Precision	Recall	Precision
DETR	0.68	0.77	0.64	0.68
EfficientDet	0.86	0.86	0.9	0.77
YOLOv5	0.87	0.75	0.85	0.72

*Nota.* De “Contrasting YOLOv5, Transformer, and EfficientDet Detectors for Crop Circle Detection in Desert” por M. L. Mekhalfi, C. Nicolo, Y. Bazi, M. Rahhal y N. Alsharif, 2021, *IEEE Geoscience and Remote Sensing Letters*, 19, p. 4 (<https://doi.org/10.1109/lgrs.2021.3085139>)

De los trabajos antes mencionados destaca YOLO como la mejor red neuronal para la detección de objetos, será utilizado en este trabajo para la detección del personal de construcción en un entorno constructivo.

Se utilizó YOLO para la identificación de personas por medio de cámaras PTZ y se detectó la actividad humana combinando la información visual con la información auditiva usando un LDV de fibra parcial. Para su base de datos se utilizaron imágenes de internet y un modelo pre-entrenado de YOLOv2. Laser Doppler Vibrometer (LDV) es una fibra que tiene la capacidad de detectar vibraciones. Los resultados mostraron una velocidad de reconocimiento de 77 fotogramas por segundo (fps) y una tasa de reconocimiento de 99,89%, por lo cual demostró que las cámaras PTZ son útiles para la detección en tiempo real. (Han et al., 2019).

Se identificó peatones mediante el uso de la red neuronal YOLOv3 entrenada con imágenes multiespectrales, estas imágenes se obtienen separando la imagen original en la composición de colores rojo (R), verde (G) y azul (B), concatenadas con una imagen térmica. Las imágenes multiespectrales son tomadas para la base de datos, estas presentan una solución para la detección de peatones especialmente en condiciones de poca luz. Se propuso una optimización de YOLOv3 para mejorar su precisión de modo que detectaba peatones más pequeños. Esta propuesta de optimización consiste en minimizar la diferencia entre el cuadro delimitador predicho y el real. Como resultado de este cambio se consiguió una mejora en el tiempo de procesamiento manteniendo su precisión (Nataprawira et al., 2021).

Se utilizaron distintos modelos de detección de objetos para comparar su precisión en la detección de personas y medición del distanciamiento social basado en la visión de computadora. Se utilizó una base de datos disponible llamada “INRIA person dataset”, que es una base de datos de imágenes de personas usada para la detección de peatones que consta de 614 personas detectadas (Madane & Chite, 2021).

Se utilizó la red neural YOLOv3 para la detección de personas con el objetivo de calcular la distancia social entre dos personas y emitir una alerta cuando se superen los límites. Para calcular la distancia entre los objetivos se usó visión estéreo por computadora (Computer Stereo Vision), que es la extracción de información 3D de cámaras digitales. La precisión en las imágenes era de 99.7%, en video de 99.3% y en cámara 84.8% (Yeshasvi et al., 2021).

Se utilizó el algoritmo YOLOv3 para la detección de cascos de seguridad a una distancia cercana y se modificó el algoritmo para solucionar problemas de inexactitud. Se demostró que la red optimizada mejora la eficacia y proporciona mejores resultados,

esta mejora consistió en agregar Desnet a la estructura de YOLOv3. Para su entrenamiento se utilizó una base de datos de 402 imágenes del personal de construcción. Se obtuvo un mejor resultado con la mejora realizada a YOLOv3, con una precisión de 2.6% superior al tradicional como se observa en la tabla 1.4 (Wu et al., 2019).

**Tabla 1.4**

*Comparación entre YOLOV3 y YOLO-dense backbone*

	<b>mAP</b>	<b>Recall</b>	<b>Precision</b>
YOLOv3	95.15%	97.53%	95.18%
YOLO-Dense Backbone	97.59%	97.59%	98.78%

*Nota.* De “Helmet Detection Based On Improved YOLO V3 Deep Model”, por F. Wu, G. Jin, M. Gao, Z. He y Y. Yang, 2019, *IEEE 16<sup>th</sup> International Conference on Networking, Sensing and Control (ICNSC)*, pp. 368 (<https://doi.org/10.1109/icnsc.2019.8743246>)

Se utilizó el modelo de detección YOLOv3 añadido con el modelo de zoom de un solo pixel (SPZ-Det) diseñado para la identificación de objetos pequeños para la detección de cascos del personal de construcción a una distancia cercana. Este modelo mejora la información lo que garantiza que las características se mantengan, soluciona la pérdida de información mejorando la precisión y mantiene la velocidad de detección. Se utilizó una base de datos conformado por 7582 imágenes, se incluyó el módulo SPZ en YOLOv3 y se obtuvo una mejora en su precisión de objetos pequeños (cascos) pero una pequeña disminución en la detección de personas como se aprecia en la tabla 1.5 (Zhou et al., 2020).

**Tabla 1.5**

*Resultados YOLOv3 vs YOLOv3 con el modelo SPZ*

<b>Network</b>	<b>AP (hat)</b>	<b>AP (person)</b>	<b>mAP</b>
YOLOv3	86.3%	56.4%	71.4%
YOLOv3+SPZ	91.5%	55.5%	73.5%

*Nota.* De “Helmet Detection Algorithm Based on Single Pixel Zoom” por Y. Zhou, L. Jiang, Y. Liang, C. Ma, H. Sun, S. Nie y Y. Zuo, 2020, *Journal of Physics: Conference Series*, 1682(012021), p. 6 (<https://doi.org/10.1088/1742-6596/1682/1/012021>)

Se utilizó el modelo de detección YOLOv3 para identificar los cascos del personal de construcción. Se utilizaron 7581 imágenes como base de datos de las cuales se probaron en 689 para luego exportar los resultados en un archivo txt. La base de datos incluía imágenes de distintas dimensiones, y se hizo uso de aumento de datos (data augmentation) para incrementar el tamaño de la base de datos. Se realizó una comparación utilizando las imágenes sin alteración, y luego una identificación de las mismas imágenes, pero con desenfoque gaussiano. Las imágenes sin alteración presentaron un nivel de confianza de 0.966, y las imágenes con desenfoque gaussiano presentaron un nivel de confianza de 0.982 (Geng et al., 2020).

Se utilizó el modelo de detección YOLOv3 con un método mejorado para la detección del uso de cascos. Se usó 20554 imágenes obtenidas de los videos de monitoreo de una construcción como base de datos. El método mejorado de YOLOv3 consiste en cambiar las escalas de las imágenes de la base de datos debido a las diferentes resoluciones de las imágenes obtenidas del video. Los resultados cumplieron con la precisión de detección y requisitos a tiempo real (Hu et al., 2019).

Se utilizó el algoritmo YOLOv4 para detectar los cascos del personal de construcción, utilizando un video de construcción complementado por imágenes para la recolección de datos. Se realizó una mejora a YOLOv4 utilizando métodos de entrenamiento multiescala para garantizar una mayor precisión y para aumentar la base de datos variando el tamaño de las imágenes. En este trabajo se hicieron 3 escalas diferentes aparte de la imagen original, cuadruplicando el tamaño de la base de datos. La versión final se utilizó para el entrenamiento de YOLOv4, obteniendo un 92.89% de precisión mientras que el YOLOv4 entrenado sin las imágenes multiescala obtuvo 90.66% de precisión (Benyang et al., 2019).

Se utilizó histogramas de los colores de los EPP en distintas iluminaciones para realizar el seguimiento del personal de construcción a través de cámaras a tiempo real. Se creó la plantilla de histograma de gradientes orientados (HOG) para detectar los colores de los EPP y sus respectivas gradientes para diferenciar al personal de construcción de los peatones. Se utilizó 500 imágenes como base de datos para crear los histogramas y obtuvo un promedio de 99% de precisión. El método presentado permite seguimiento en la secuencia de actividades, mediciones de productividad, análisis y gestión de seguridad del sitio (Park & Brilakis, 2012).

Se utilizó la inteligencia artificial Faster-RCNN para realizar el monitoreo del personal de obra por medio de cámaras para asegurar el cumplimiento de protocolos de seguridad de obra. Se detectó las interacciones de los trabajadores al momento de utilizar las herramientas de trabajo y el uso de equipo de obra, que pueden ser renderizadas para representar las imágenes que están captando de la vida real (Tang et al., 2020).

Se utilizó el modelo de detección YOLO para identificar 8 clases de objetos como volquetes, excavadoras, mixer, camiones y personas. Se entrenó dos redes neuronales utilizando diferentes bases de datos, una base de datos de 1000 imágenes capturadas por drones, y otra de 1046 imágenes capturadas por cámaras estáticas, logrando una precisión entre 78.8% y 82.8% con las imágenes de los drones, y una precisión entre 73.56% y 93.76% con las imágenes de las cámaras estáticas, demostrando la viabilidad del uso de estos algoritmos de clasificación (Del Savio et al., 2021).

Se utilizó la red neuronal Faster-RCNN para realizar la identificación de trabajadores y de excavadoras en tiempo real en una construcción para mejores decisiones en seguridad y planificación. Se utilizó una base de datos de 10,000 imágenes desde distintos puntos de vista, escalas y posiciones. Se logró el objetivo con un tiempo de 0.101 segundos en detección de los objetivos, con una precisión de detección de trabajadores del 91% y un 95% en la detección de excavadoras (Fang et al., 2018).

Se propuso el modelo de detección de objetos CORY-Net para la detección de distintos objetos como casos, elementos de seguridad, torres de alta tensión y elementos que puedan representar un riesgo. Para realizar el monitoreo inteligente se utilizó YOLOv5. Se utilizó una base de datos de 7146 imágenes de una construcción de la red eléctrica. CORY-Net consiguió una detección 0.5% mayor a la detección de YOLOv5 (Peng et al., 2021).

Se utilizó la red neuronal de detección de objetos YOLOv4 para detectar dos objetos, una persona erguida y una persona agachada. En la base de datos se emplearon más de mil imágenes de una construcción y se obtuvo una precisión de 77.98% en promedio de los dos objetos. Se calculó la distancia entre los objetos con el fin de asegurar un adecuado distanciamiento social y en caso del incumplimiento de las medidas de distanciamiento se generó una alerta (Del Savio et al., 2023).

De los trabajos anteriores se concluye, la efectividad que tiene YOLO de reconocer distintos objetos como personas, maquinaria, EPP y redes eléctricas con un

gran nivel de confianza. Sin embargo, en la mayoría de los casos las imágenes que componen las distintas bases de datos fueron tomados a una corta distancia entre la cámara y los trabajadores, por lo que se requiere de una mayor cantidad de imágenes en la base de datos para conseguir ese nivel de confianza en la detección para distancias más alejadas.

Se analizó la viabilidad de técnicas de aprendizaje profundo (Deep Learning) en la inspección de la infraestructura de un proyecto de construcción, el proceso buscó detectar anomalías como grietas, fisuras, desconchado. La base de datos tuvo un tamaño de 200 imágenes de defectos en la infraestructura y se utilizaron 94 imágenes para la validación. El modelo creado genera un mapa de anomalías y lo clasifica según el daño. Se obtuvo un 88.7% de precisión en la detección de fisuras, cumpliendo con lo esperado (Chow et al., 2020).

En el trabajo de Zhou y Song (2020), se mejoró la técnica de detección de grietas en los pavimentos, la inspección de defectos en los puentes y en el revestimiento de los túneles. Algunas perturbaciones en la superficie eran detectadas como falsos positivos, desarrollaron una mejor técnica de detección más consistente y precisa. La base de datos usada estaba compuesta por 32,408 imágenes de grietas en pavimentos,

Se utilizó redes neuronales convolucionales (CNN) para la detección grietas y fisuras en el hormigón. La base de datos constaba de 332 imágenes de grietas y fisuras a distintas condiciones de luz, de las cuales 277 fueron utilizadas para el entrenamiento y la validación, mientras que 55 imágenes fueron utilizadas para las pruebas. Se obtuvo un 98.22% de precisión en la detección de las grietas (Cha et al., 2017).

Se utilizó un modelo neuro-difuso para predecir el rendimiento de pavimentos flexibles teniendo en cuenta las condiciones actuales de la estructura, el tráfico, así como factores climáticos y ambientales. Un modelo neuro-difuso es la combinación de redes neuronales artificiales con lógica difusa. Se creó cinco modelos uno para cada estación climática identificada y logró obtener resultados muy aproximados a los reales (Bianchini & Bandini, 2010).

Se utilizó el modelo de detección CrackDN para detectar fisuras en pavimentos con fondos complejos. La base de datos usada presentaba imágenes de fisuras en 4 fondos distintos, un fondo de pavimento común, un fondo con iluminación variante, un fondo con marcas en la pista (cruceos peatonales), y un fondo con sombras sobre las fisuras.

Se comparó CrackDN con el modelo de detección Faster-RCNN para comprobar su efectividad y demostró mejor precisión en detección de fisuras que Faster-RCNN, (Huyan et al., 2019).

Se utilizó redes neuronales convolucionales (CNN) para detectar grietas en superficies metálicas de centrales nucleares y se utilizó Naïve Bayes (NB-CNN) para analizar videos individuales. Se utilizó más 100000 imágenes de grietas en superficies metálicas como base datos. Se obtuvo una tasa de acierto de 98.3% y es capaz de detectar grietas pequeñas con bajo contraste, brillo y condiciones poco favorables que las hacen poco visibles. (Chen y Jahanshahi, 2018).

Los trabajos anteriores demuestran que por medio de la inteligencia artificial es posible identificar grietas y fisuras en pavimentos, puentes, edificaciones y superficies metálicas con un gran nivel de confianza, pero con un gran tamaño en la base de datos a pesar de la dificultad de las condiciones en las que se encuentren dichas fisuras, mostrando otra capacidad de la inteligencia artificial en la industria de la construcción.

La tecnología de realidad mixta consiste en combinar la información digital con el mundo real para una mejor gestión de la información. Trimble XR10 es una aplicación para los HoloLens 2 de Microsoft, en el cual se integra el modelo BIM del proyecto, lo que le permite visualizar el proyecto en realidad aumentada para un mejor monitoreo y una mejor inspección permitiendo obtener mejores resultados en el sector de la construcción implementando la inteligencia artificial (Rezvani et al., 2022).

Con la información sobre la planificación, cronograma y modelo del proyecto que brinda BIM4D y con la inteligencia artificial se puede monitorear el progreso de una construcción a nivel operativo, utilizando fotos del sitio. Cada elemento BIM se proyecta sobre las fotos tomadas en las que se vea el elemento a comparar, de ese modo se puede realizar la clasificación de los materiales, entrenando a las redes neuronales para que aprendan a detectarlos obteniendo un 94% de precisión (Han & Golparvar-Fard, 2015).

Alice Technologies es la primera plataforma a nivel mundial que simula de proyectos en millones de iteraciones con inteligencia artificial. Haciendo uso de la metodología BIM se facilita la gestión de la información del proyecto permitiendo la integración con la inteligencia artificial para el aprendizaje automático. Alice Technologies permite simular y evaluar escenarios potenciales posibles para llevar a cabo

un proyecto, realizan la planificación de este, de esa manera se reducen los riesgos y se asegura de ser eficiente reduciendo tanto tiempo como costos (Caro, 2021).

La plataforma DECENTER Fog Computing y Brokerage Platform permite la implementación y control de distintas inteligencias artificiales en una misma plataforma. Se necesitan varias aplicaciones inteligentes para solucionar problemas en la construcción. DECENTER presentó buenos resultados como mejor rendimiento en la detección de anomalías, reducción de costos operativos y una mejora en la capacidad de utilizar varios modelos de IA (Kochovski & Stankovski, 2021).

Hay programas que pueden ser aplicados al sector de la construcción, programas que permiten un mejor control y monitoreo con realidad aumentada, incluso un programa que simula proyectos para encontrar la mejor forma de ejecutarlos, proyectando el cronograma, reduciendo riesgos y costos no planeados.

La industria de la construcción presenta problemas en la productividad, problemas que pueden ser resueltos por una IA, debido a que tiene la capacidad de aprovechar los datos que se generan a lo largo del ciclo de vida de un proyecto de construcción, mostrando las oportunidades que presenta y como su aplicación va aumentando paulatinamente en el sector de la construcción (Abioye et al., 2021).

En síntesis, para crear una base de datos es necesario tener información catalogada y clasificada, información que en el sector construcción se pierde debido a la falta de interés en hacerlo. La catalogación requiere de tiempo e información que se pueden obtener de las siguientes formas:

- Videos o imágenes del o los objetos que se quieran identificar.
- Modelo BIM4D, debido a que contiene información del proyecto como programación.
- Plano 3D, para monitorear el avance de obra.

Y las aplicaciones que puede tener la inteligencia artificial en el sector de la construcción son las siguientes:

- Detección de trabajadores, así como cascos y EPP, permitiendo un mejor control y monitoreo del personal de trabajo pudiendo hacerlo incluso de forma remota.

- Detección de lugares inseguros, que en conjunto con el punto anterior se podría mandar una alerta al detectar que un trabajador está cerca de un lugar inseguro sin la indumentaria necesaria para dicha tarea, previniendo accidentes.
- Planificación de obra, es posible con inteligencia artificial encontrar la mejor forma de ejecutar un proyecto incluso ha sido probado en obras reales dando grandes resultados en las tareas que le fueron asignadas.
- Detección de grietas y fisuras, es posible realizar la detección de dichos problemas tanto en concreto armado como en estructuras metálicas. Permite realizar de forma sencilla la tarea de monitoreo a estructuras o pavimentos en proyectos culminados para evaluar si necesitan mantenimiento, así como posterior a un desastre natural los daños a la estructura son significativos permitiendo prevenir un colapso.
- La implementación de la Metodología BIM a la inteligencia artificial, debido a que el modelo BIM tiene la información del proyecto. Se puede realizar un control de avance de obra con el modelado 3D y la situación real de la obra, además que en otro trabajo se realizó la detección de trabajadores con herramientas y maquinaria renderizándolos, creando un modelo 3D de dichos elementos con los que se podría ser más preciso en la ejecución de una obra evitando el problema de un mal replanteo de obra.

## 2. METODOLOGÍA

Para obtener la base de datos se utilizaron 1214 imágenes obtenidas de cámaras motorizadas Panoramic Tilt Zoom (PTZ) y cámaras tipo Bullet estratégicamente colocadas a 50 metros y 8 metros del proyecto. Las cámaras grabaron la construcción del nuevo Centro de Bienestar Universitario (CBU) de la Universidad de Lima para extraer fotogramas de los videos para obtener imágenes como se observa en la figura 2.1, con las que se procederá a realizar la clasificación manual mediante labelme y labelImg (Wada, 2021).

Detalles de las cámaras utilizadas:

**Tabla 2**

*Especificaciones de las cámaras*

	<b>Cámara IP MOTORIZADA PTZ</b>	<b>CÁMARA TIPO BULLET NETWORK</b>
		
<b>Modelo</b>	SD10A848WA-HNF	IPC-HFW2831T-ZS-S2
<b>Fabricante</b>	DAHUA	DAHUA
<b>Sensor</b>	1 / 1.8 " Complementary Metal Oxide Semiconductor (CMOS)	1/2,7" Complementary Metal Oxide Semiconductor (CMOS)
<b>Lentes</b>	5,7 mm – 275 mm	3.7mm ~ 11mm
<b>Zoom</b>	48x	10x
<b>ROM</b>	8GB	
<b>RAM</b>	2GB	<b>512 MB / 128 MB</b>

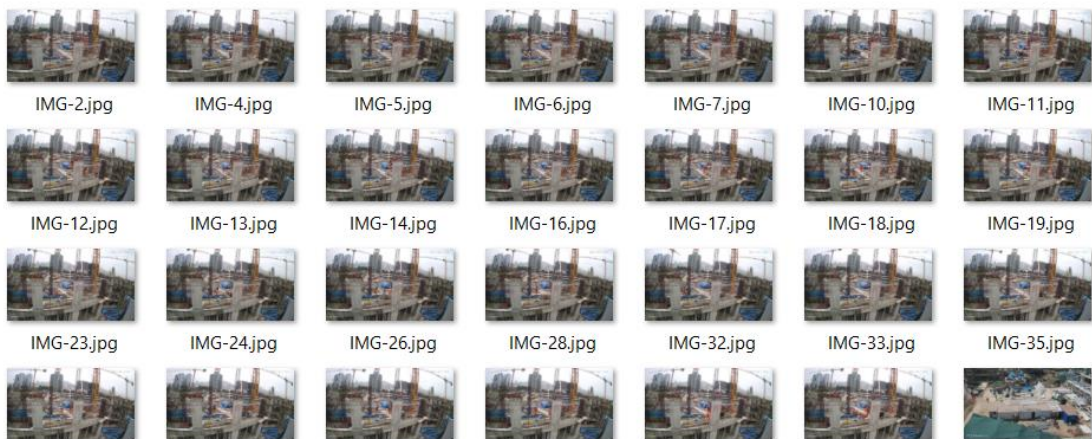
(continúa)

(continuación)

	<b>Cámara IP MOTORIZADA PTZ</b>	<b>CÁMARA TIPO BULLET NETWORK</b>
<b>Resolución</b>	Transmisión principal: 1080p / 1.3 M / 720p (1-50 / 60fps) Transmisión: secundaria 1: D1 / CIF (1-25 / 30fps) Transmisión secundaria 2: 1080p / 1.3 M / 720p (1-25 / 30fps)	4K (3840 × 2160) / 6M (3072 × 2048) / 5M (2560 × 1920) / 3M (2048x1536) / 3M (2304 × 1296) / 1080P (1920 × 1080) / 1.3M (1280x960) / 720P (1280 × 720) / D1 (704 × 576/704 × 480) / VGA (640 × 480) / CIF (352 × 288/352 × 240)
<b>Cuadros por segundos</b>	Velocidad de bits variable / Velocidad de bits constante	4K ( 1- 25/30 fps) Sub Stream: D1 (1 ~ 25 / 30fps)
<b>Control de velocidad de bits</b>	H.264: 32 Kbps ~ 8192 Kbps / H.265: 16 Kbps ~ 8192 Kbps	CBR / VBR
<b>Certificación</b>	FCC: Part15 subpartb, ANSI C63.4-2014	CE (EN 60950:2000) UL:UL60950-1 FCC:FCC Part15 SubpartB

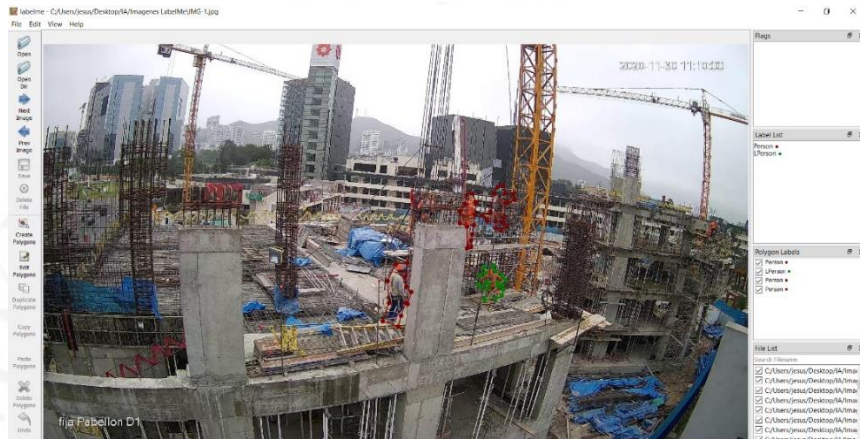
**Figura 2.1**

*Imágenes que serán clasificadas*



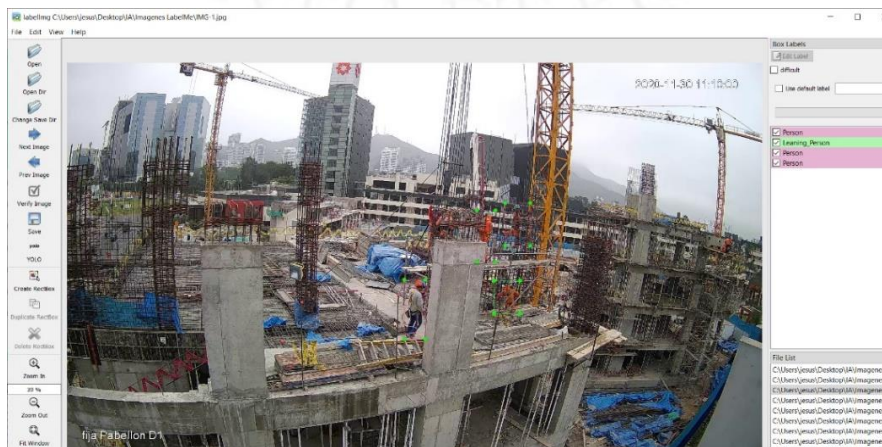
Labelme es una herramienta de anotación que permite realizar la clasificación manual para construir nuestra base de datos para su posterior entrenamiento como se ve en la figura 2.2. Labelme permite crear polígonos y ponerle la etiqueta correspondiente para posteriormente guardar la información en la extensión JavaScript Object Notation (JSON), que sirve para construir la base de datos en formato COCO.

**Figura 2.2**  
*Clasificación Manual en Labelme*



LabelImg es una herramienta de anotación de imágenes que facilita la recolección de datos para la clasificación manual, permite crear rectángulos en los que estén ubicados los elementos a clasificar y colocarles una etiqueta como se observa en la figura 2.3, además que puede almacenar la información en dos formatos distintos, en archivos de texto “.txt” y en archivos XML que sirven para crear una base de datos en formato PASCAL VOC.

**Figura 2.3**  
*Clasificación Manual en LabelImg*



Durante la clasificación manual de las 1214 imágenes se buscará identificar al personal de construcción en 2 posiciones, cuando está de pie (person), como se observa en la figura 2.4; y cuando se encuentra con un grado de inclinación o en otra posición que no sea estar de pie (leaningperson), como se observa en la figura 2.5. Realizada la clasificación manual de las imágenes, la base de datos estará lista. Este mismo proceso se realizará en los formatos “.txt”, “.xml” y “.json”.

**Figura 2.4**

*Clasificación Clase Person*



**Figura 2.5**

*Clasificación Clase Leaning\_Person*



Como se puede observar en la figura 2.6, el primer número de cada fila, es decir, 0 o 1, indica la etiqueta a la que corresponde, el caso 0 representa “leaning\_person”, y 1 representa “person”, y los 4 números a su derecha indicar las coordenadas de los vértices del polígono creado en LabelImg en el que se encontraba el elemento en la imagen.

**Figura 2.6**

*Datos de las etiquetas en formato “.txt”*

IMG-75.txt: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda
1	0.171484	0.217593	0.011198	0.051852
1	0.279167	0.373611	0.014063	0.060185
1	0.329036	0.347685	0.022656	0.059259
1	0.358203	0.345833	0.019531	0.062963
1	0.727865	0.261806	0.009896	0.048611
1	0.945964	0.189120	0.014323	0.055093
0	0.896745	0.178241	0.014323	0.032407
0	0.879818	0.159028	0.009635	0.038426
1	0.624349	0.094907	0.005469	0.031481
1	0.607682	0.080093	0.007031	0.028704
0	0.934635	0.573843	0.025000	0.063426
1	0.766536	0.677315	0.023177	0.092593
1	0.737500	0.675926	0.022917	0.087037

En la figura 2.7 se puede ver la estructura de una clasificación manual en formato XML donde se observan datos como el tamaño y el ancho de la imagen original, el nombre de la etiqueta a la que corresponde cada elemento “Person” y “Leaning\_Person”, y la información de la posición de cada vértice de cada rectángulo creado que contenía el objeto a clasificar, así como información del nombre de la imagen y de la ubicación en la que se encuentra la imagen, y el directorio de la carpeta donde está contenido.

### Figura 2.7

*Datos de las etiquetas en formato “.XML”*



```

IMG-75.xml: Bloc de notas
Archivo Edición Formato Ver Ayuda
<size>
  <width>3840</width>
  <height>2160</height>
  <depth>3</depth>
</size>
<segmented>0</segmented>
<object>
  <name>Person</name>
  <pose>Unspecified</pose>
  <truncated>0</truncated>
  <difficult>0</difficult>
  <bndbox>
    <xmin>637</xmin>
    <ymin>414</ymin>
    <xmax>680</xmax>
    <ymax>526</ymax>
  </bndbox>
</object>

```

En la figura 2.8 se observa la estructura de un archivo de clasificación manual en formato JSON. En comparación con los otros dos formatos, Labelme crea polígonos y no solo rectángulos. La figura 2.8 tiene la información de los vértices del polígono donde se encuentra el objeto a clasificar, así como el nombre de la etiqueta correspondiente, “Person” y “LPerson”, el tipo de forma, la ruta de la imagen, la data de la imagen, y la altura y ancho de la imagen.

## Figura 2.8

*Datos de las etiquetas en formato “.JSON”*

```
IMG-75.json: Bloc de notas
Archivo Edición Formato Ver Ayuda
"shapes": [
  {
    "label": "Person",
    "points": [
      [
        27.058823529411764,
        682.9411764705883
      ],
      [
        19.411764705882355,
        688.2352941176471
      ],
      [
        14.117647058823529,
        717.6470588235294
      ],
      [
        14.117647058823529,
        733.5294117647059
      ],
      [
        27.647058823529413,
        738.8235294117648
      ],
      [
        34.705882352941174,
        760.5882352941177
      ],
    ]
  }
]
```

Una vez la clasificación manual de todas las imágenes se completa, se comenzará a crear la base de datos donde aleatoriamente el 30% de las imágenes son usadas para validación y el 70% restante para el entrenamiento de YOLOv4.

### 3. RESULTADOS DEL ENTRENAMIENTO EN YOLOv4

Se entrenó 4 veces la inteligencia artificial YOLOv4 con las 1214 imágenes con distintas distancias al mismo tiempo para ver su desarrollo y los resultados que presenta en la precisión de la detección, el tiempo de detección (recall) y el Intersection over Union (IoU). Se utilizaron 4000 iteraciones en cada entrenamiento debido a que la recomendación es 2000 por el número de clases que fueron etiquetadas. Se observa una mejora tras cada entrenamiento como se puede observar en la figura 3.1, figura 3.2, figura 3.3 y la figura 3.4. En el segundo y el cuarto entrenamiento se obtuvieron los mejores resultados en la detección promedio de las clases Person y Leaning\_Person como se puede observar en la tabla 3.1.

**TABLA 3.1**

*Resultados de %Map en YOLOv4*

YOLOv4				
%Map	Entrenamiento 1	Entrenamiento 2	Entrenamiento 3	Entrenamiento 4
Person	84.37%	86.16%	85.67%	85.65%
Leaning_Person	74.84%	78.37%	78.27%	78.77%
Ambos	79.61%	82.27%	81.97%	82.21%
IoU	59.67%	61.18%	60.14%	59.68%
recall	0.83	0.84	0.84	0.84

Las figuras 3.1, 3.2, 3.3 y 3.4 representan los resultados de cada entrenamiento. La línea azul representa la pérdida de información que con el paso del entrenamiento se va reduciendo y la línea roja es el map (mean average precision) que es la precisión de la detección. El eje Y representa lo que se quiere versus lo que se obtuvo y el eje X está el número de iteraciones.

Figura 3.1

YOLOv4 Resultados del Primer entrenamiento

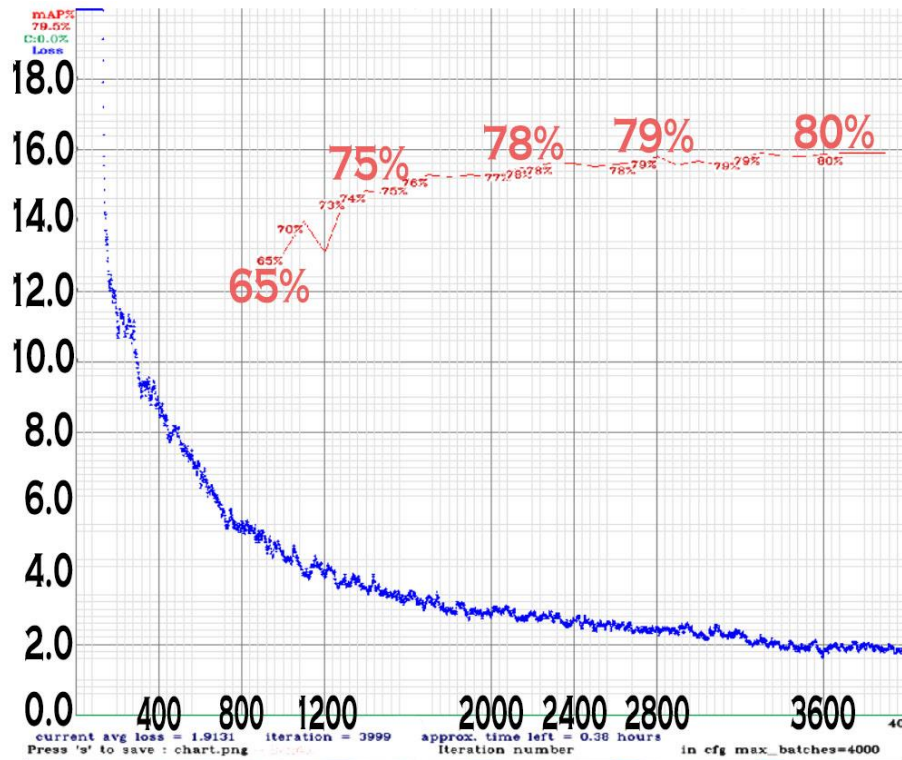


Figura 3.2

YOLOv4 Resultados del Segundo entrenamiento

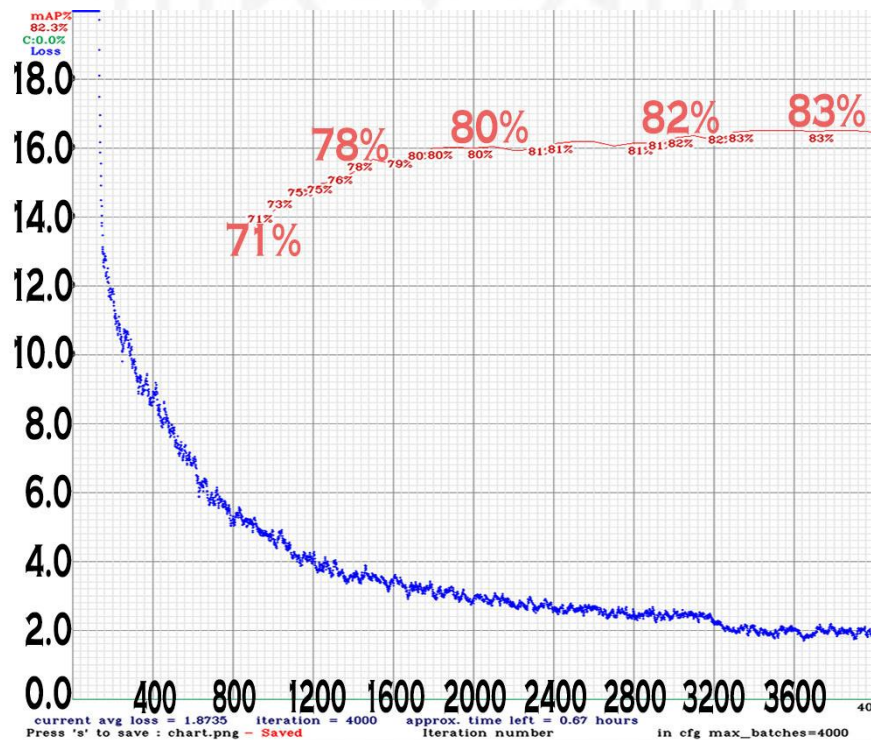


Figura 3.3

YOLOv4 Resultados del Tercer entrenamiento

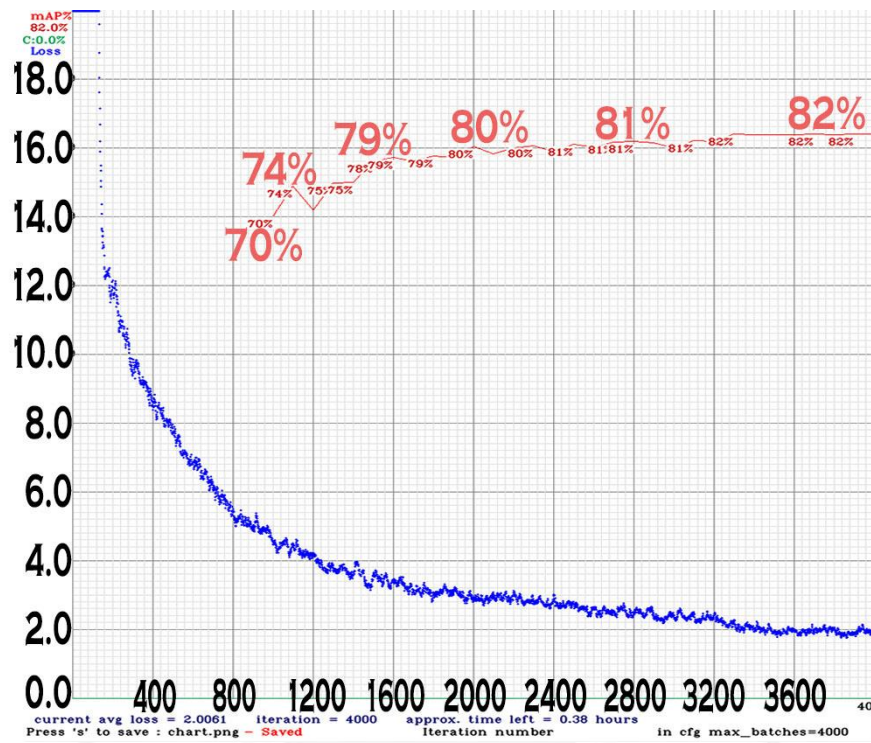
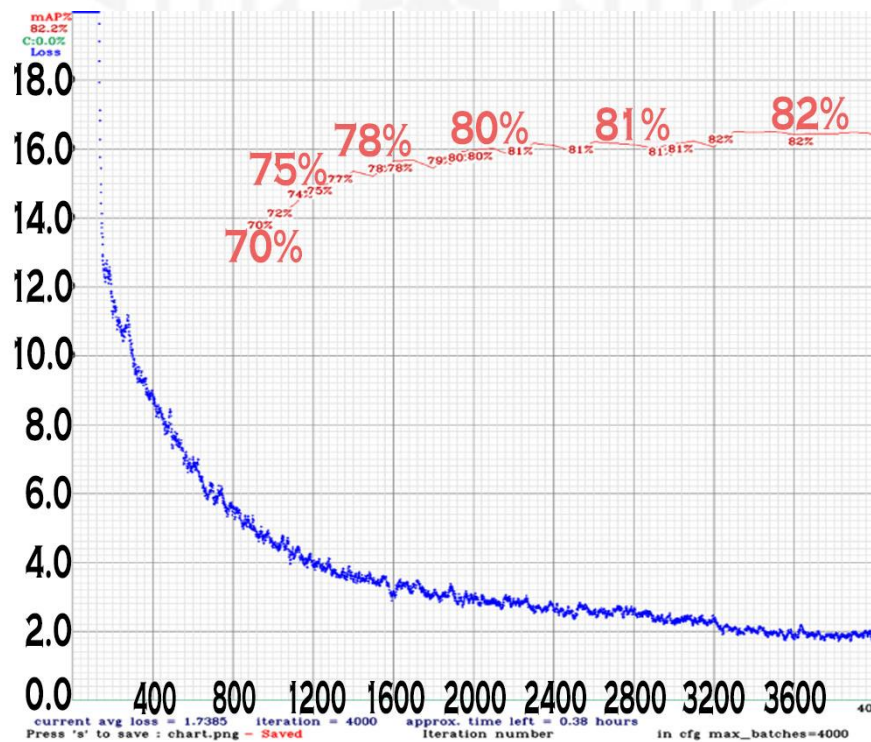


Figura 3.4

YOLOv4 Resultados del Cuarto entrenamiento



Para ver la evolución de la red neuronal YOLOv4 se utilizaron distintas imágenes de la base de datos donde la cámara estaba a 50 metros de distancia de la obra para ver seguridad en la detección de los elementos al final de cada entrenamiento como se observa en las figuras 3.5, 3.6, 3.7 y 3.8. Los elementos marcados de verde son la clase Person y los elementos marcados de morado son la clase Leaning\_Person. El número al lado del nombre de la clase es el nivel de seguridad que tiene el modelo de lo que está detectando, siendo 1.00 igual a 100%.



**Figura 3.5**

*Evaluación en la imagen IMG-729 al final del primer entrenamiento*



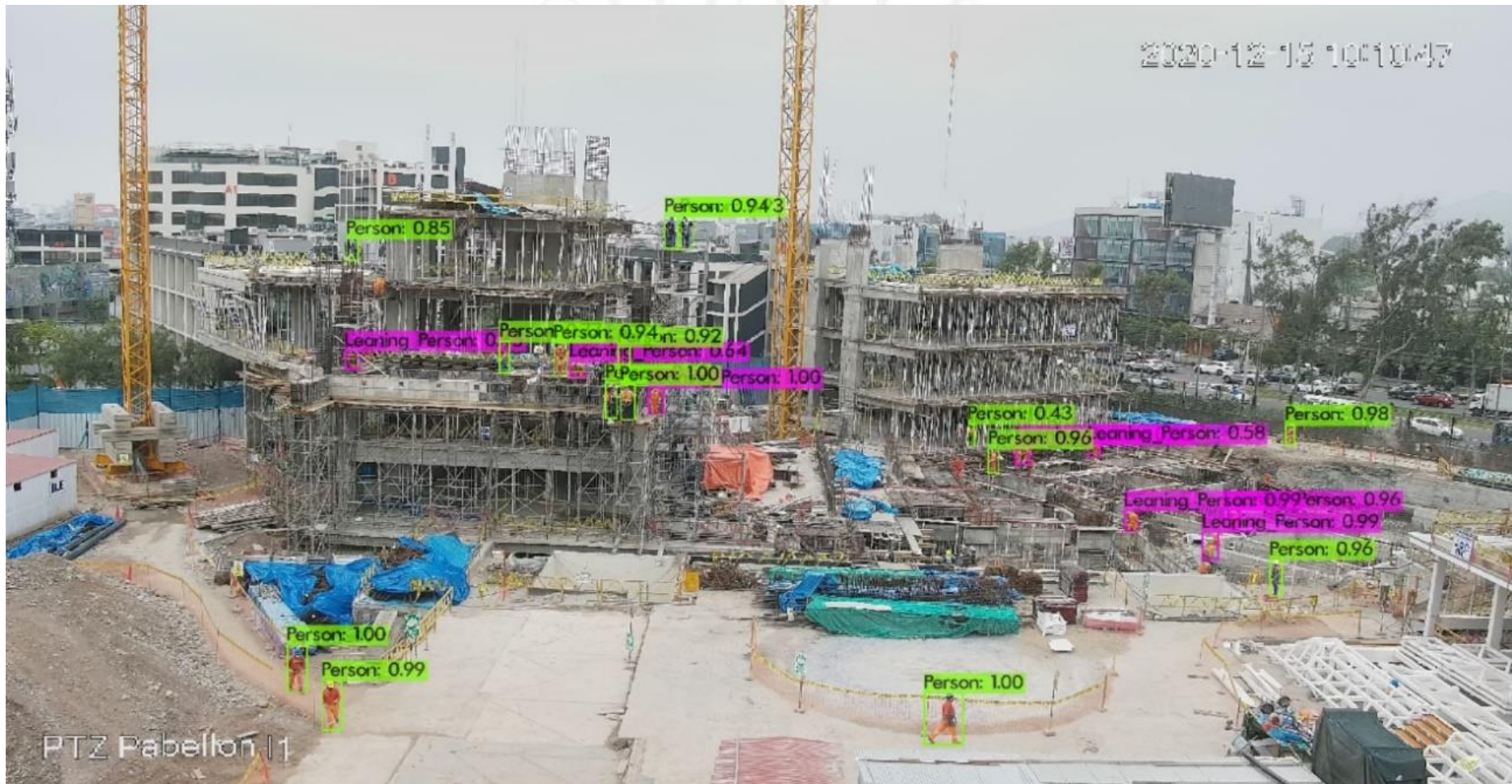
**Figura 3.6**

*Evaluación en la imagen IMG-729 al final del segundo entrenamiento*



**Figura 3.7**

*Evaluación en la imagen IMG-729 al final del tercer entrenamiento*



**Figura 3.8**

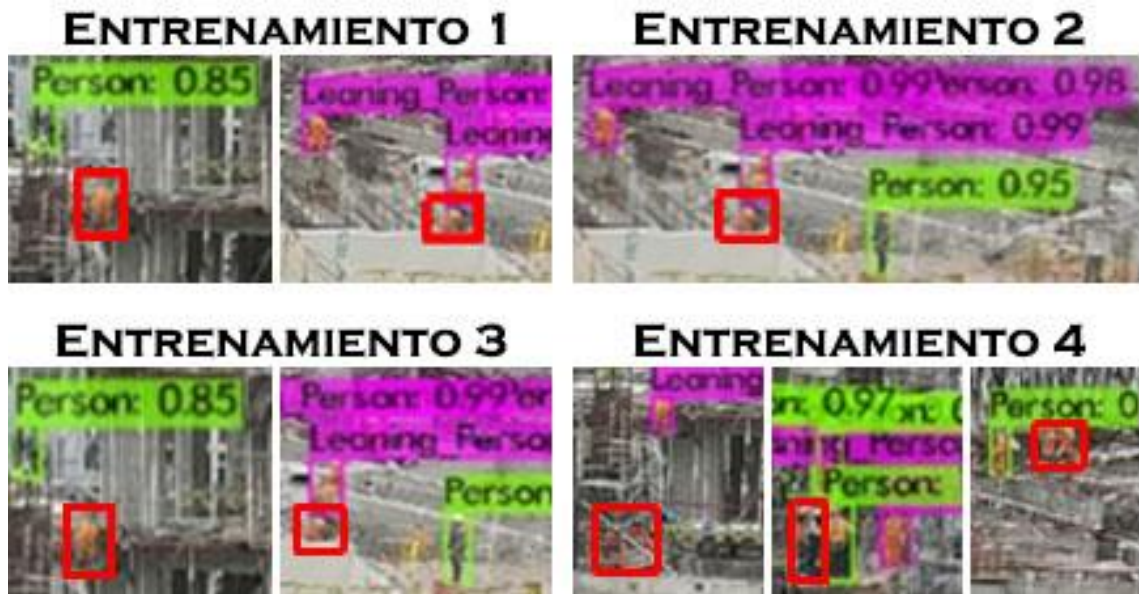
*Evaluación en la imagen IMG-729 al final del cuarto entrenamiento*



El cuarto entrenamiento a diferencia de los otros tres entrenamientos detecta un par de elementos menos como se observa en la figura 3.9 donde los elementos no detectados están dentro del recuadro rojo, pero a la vez detecta los elementos de la clase leaning\_person con una mayor precisión como se observa en la tabla 3.2.

**Figura 3.9**

*Elementos no detectados en la imagen IMG-729*



**TABLA 3.2**

*Número de elementos no detectados y precisión promedio en la imagen IMG-729*

	Entrenamiento 1	Entrenamiento 2	Entrenamiento 3	Entrenamiento 4
Número de elementos no detectados	3	3	5	6
Precisión de la Clase Person	84.85%	88.24%	88.24%	84.85%
Precisión de la Clase Leaning_Person	82.35%	82.35%	75%	88.89%
Recall	72.41%	75.86%	72.41%	75.86%
Precisión de Ambos Elementos	84%	86.28%	84%	86.28%

El segundo entrenamiento también presenta buenos resultados, detecta un par de elementos más que el cuarto entrenamiento, con mayor precisión en la clase person pero menor precisión en la clase leaning\_person. Se comparó el cuarto entrenamiento con el segundo entrenamiento utilizando distintas imágenes a una distancia entre 8 y 50 metros de la obra en distintos ángulos pertenecientes a la base de datos como se observa desde la figura 3.10 hasta la figura 3.17, donde los elementos marcados de verde son la clase Person y los elementos marcados de morado son la clase Leaning\_Person. El número al lado del nombre de la clase es el nivel de seguridad que tiene el modelo de lo que está detectando, siendo 1.00 igual a 100%.

Para la construcción de la tabla 3.3 se consideró a los elementos detectados con un nivel de seguridad menor a 0.5 como elementos no detectados. Los resultados de la cantidad de elementos no detectados de cada entrenamiento se pueden observar en la tabla 3.3, donde el cuarto entrenamiento presento mayor precisión en promedio. Se puede concluir que sin importar el ángulo de observación la precisión de ambas clases se mantiene mayor al 90%.

**TABLA 3.3**

*Comparación entre el segundo y cuarto entrenamiento respecto al número de elementos no detectados y la precisión promedio*

IMAGEN	Entrenamiento 2			Entrenamiento 4		
	Número de elementos no detectados	Recall	Precisión	Número de elementos no detectados	Recall	Precisión
IMG-460	0	100%	100%	1	93.75%	96.77%
IMG-894	1	86.67%	92.86%	1	86.67%	92.86%
IMG-989	1	90.91%	90.91%	0	100%	100%
IMG-1058	6	83.78%	91.18%	5	86.49%	92.75%



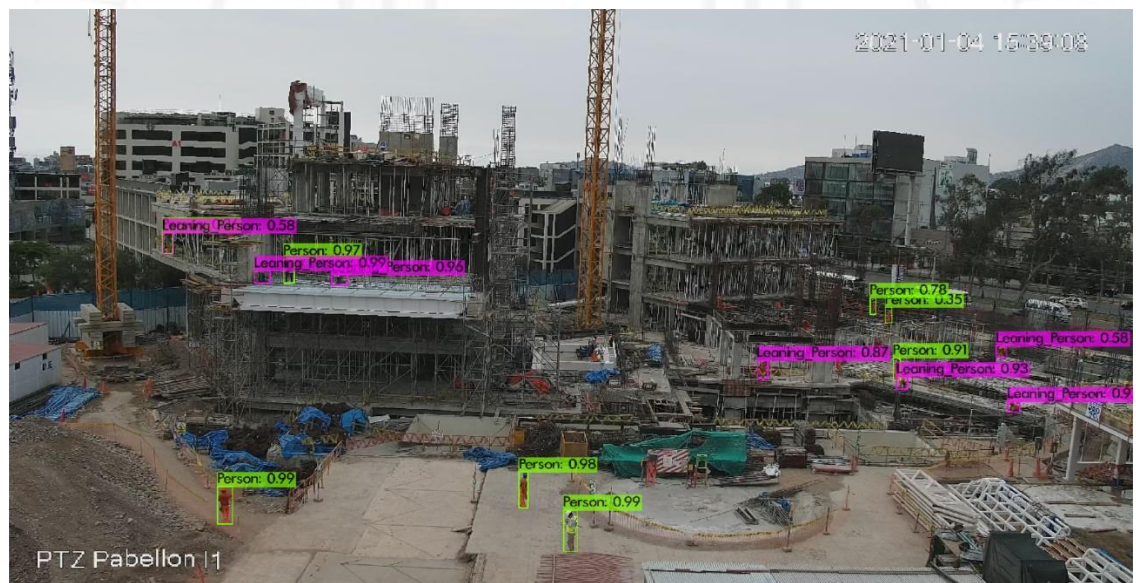
**Figura 3.12**

*Evaluación en la imagen IMG-894 al final del cuarto entrenamiento*



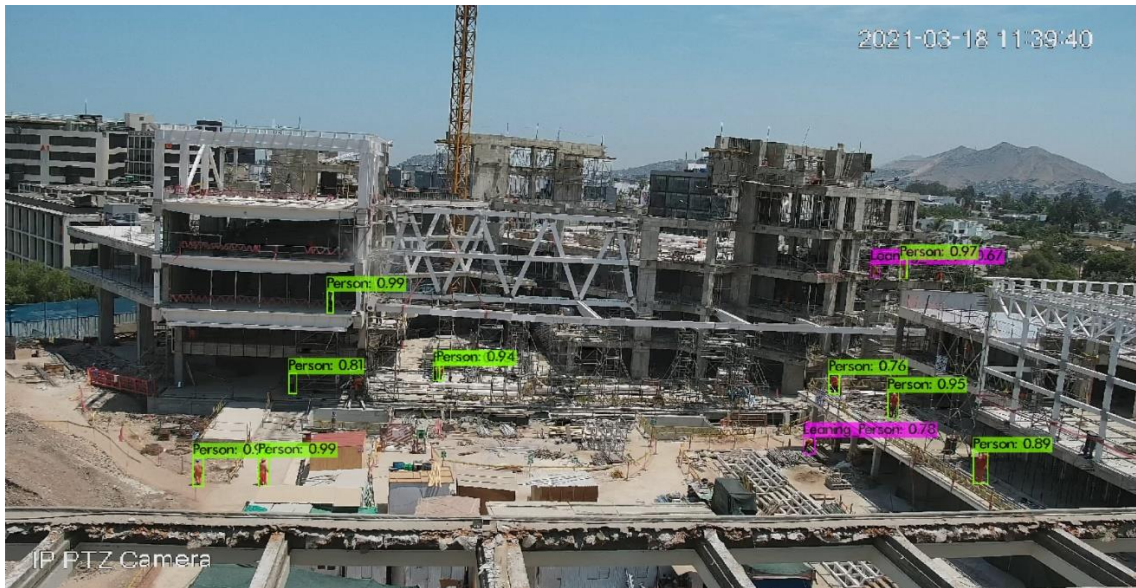
**Figura 3.13**

*Evaluación en la imagen IMG-894 al final del segundo entrenamiento*



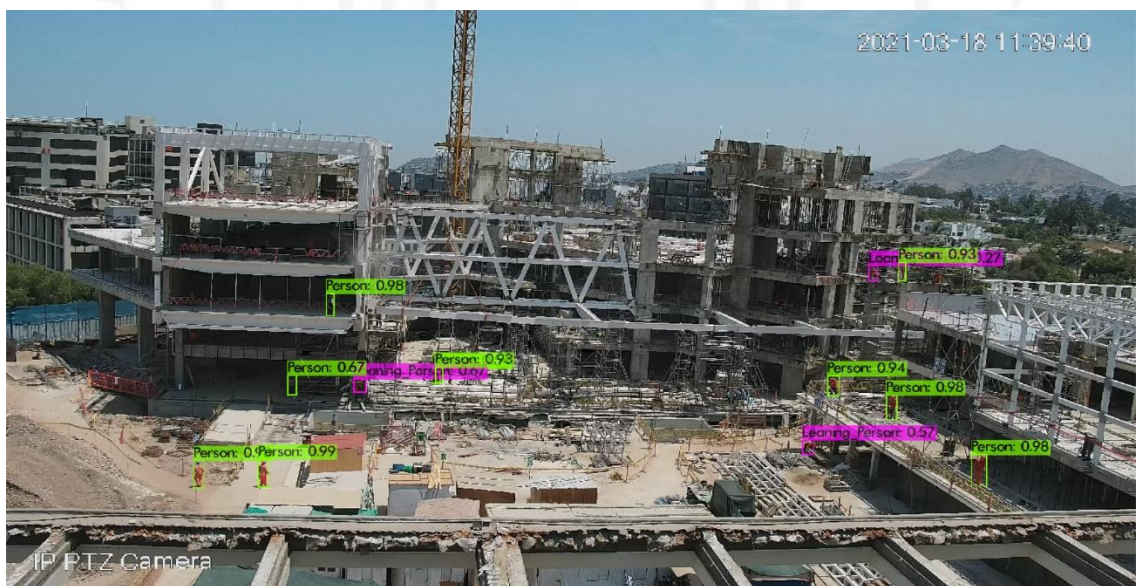
**Figura 3.14**

*Evaluación en la imagen IMG-989 al final del cuarto entrenamiento*



**Figura 3.15**

*Evaluación en la imagen IMG-989 al final del segundo entrenamiento*



**Figura 3.16**

*Evaluación en la imagen IMG-1058 al final del cuarto entrenamiento*



**Figura 3.17**

*Evaluación en la imagen IMG-1058 al final del segundo entrenamiento*



Como la base de datos creada espera ser usada en distintos proyectos se utilizó el modelo YOLOv4 después del cuarto entrenamiento en imágenes recopiladas de internet sobre otros proyectos en estado de construcción para evaluar su correcto funcionamiento. Los resultados pueden ser observados desde la figura 3.17 hasta la figura 3.21, los elementos marcados de verde son la clase Person y los elementos marcados de morado

son la clase Leaning\_Person. El número al lado del nombre de la clase es el nivel de seguridad que tiene el modelo de lo que está detectando, siendo 1.00 igual a 100%.

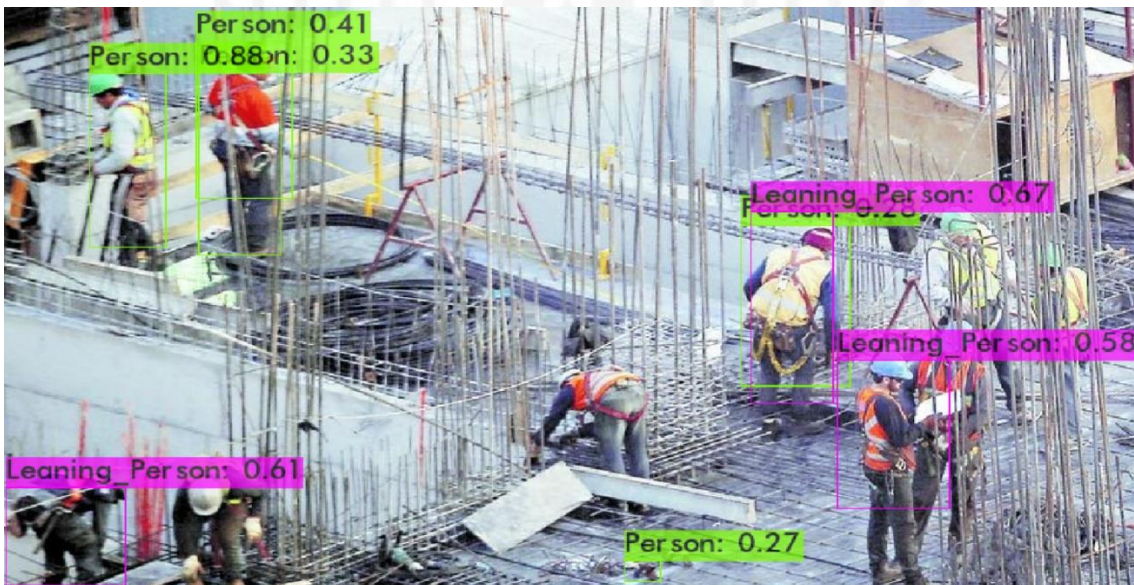
**Figura 3.18**

*Evaluación de un proyecto usando el cuarto entrenamiento*



**Figura 3.19**

*Evaluación de un proyecto 2 usando el cuarto entrenamiento*



**Figura 3.20**

*Evaluación de un proyecto 3 usando el cuarto entrenamiento*



**Figura 3.21**

*Evaluación de un proyecto 4 usando el cuarto entrenamiento*



**Figura 3.22**

*Evaluación de un proyecto 5 usando el cuarto entrenamiento*



**TABLA 3.4**

*Resultados del Modelo YOLOv4 después del cuarto entrenamiento desde la figura 3.18 hasta la figura 3.22*

Figura	Recall	Precisión	Errores en la Detección
Figura 3.18	86.67%	92.86%	
Figura 3.19	44.44%	61.54%	
Figura 3.20	100%	75%	
Figura 3.21	85.71%	88.89%	
Figura 3.22	80%	88.89%	

Para la construcción de la tabla 3.4 se consideró a los elementos detectados con un nivel de seguridad menor a 0.5 como elementos no detectados. Como se observa en la tabla 3.4, la figura 3.18 presenta buenos resultados con una precisión de 92.86%, hay un elemento detectado con 0.32 de seguridad de clase Person debido a que hay un elemento en la construcción que obstruye la vista de la mitad del cuerpo.

La figura 3.19 a comparación del resto de figuras presenta una precisión baja, presenta errores como la detección de un elemento como si fueran dos debido a las varillas de construcción que dificultan la detección de los elementos y en algunos casos atraviesa al elemento que provoca que el modelo detecte doble. El elemento inexistente es detectado debido a que la base de datos generada contiene imágenes a una distancia de 50 metros de la obra y el modelo lo detectó como si fuera una persona a una distancia lejana, pero con un nivel de seguridad de 0.27.

La figura 3.20 también presenta un problema de duplicidad, el elemento es detectado como clase Person y Leaning\_Person al mismo tiempo debido a que el elemento no está lo suficientemente agachado, el modelo lo detectó como Person considerando el casco y como Leaning\_Person sin el casco.

La figura 3.21 presenta una precisión de 88.89%, tiene un problema de duplicidad de elementos debido a que la tabla de madera atraviesa parte del cuerpo lo que provocó que el modelo lo detecte como 2 elementos uno donde toma en cuenta las piernas y otro donde no lo toma en cuenta. El otro error es el elemento detectado con 0.34 de seguridad, error producido debido a que el modelo consideró al casco como parte de otro elemento detectado y la falta del casco provocó la poca seguridad del modelo.

La figura 3.22 presenta una precisión de 88.89% el único error en la detección es un elemento mal detectado con un nivel de seguridad de 0.35 debido a que el cordón colocado en la obra atraviesa el elemento lo que provocó que el modelo detecte solo la parte superior al cable sin embargo tampoco logró detectar el casco.

## 4. CONCLUSIONES

- Las bases de datos contienen imágenes digitales etiquetadas que son imprescindibles para entrenar modelos de inteligencia artificial. Si se quiere implementar la Industria 4.0 en la industria de la ingeniería civil, se necesita de más bases de datos que faciliten y promuevan el desarrollo de nuevas aplicaciones de la IA en el sector de la construcción.
- La base de datos presentada contiene 1214 imágenes etiquetadas del personal de construcción en un entorno constructivo. Esta permitió entrenar un modelo de inteligencia artificial capaz de identificarlos correctamente a una distancia de 8 metros y de 50 metros al mismo tiempo, con un buen porcentaje de confianza y casi en la totalidad de los elementos. Sin embargo, existen algunos casos en donde no llegan a ser identificados por completo o son identificados dos veces como elementos distintos, se recomienda agregar a esta base de datos más imágenes relacionadas con el proyecto a desarrollar o imágenes en donde se encuentren obstrucciones (varillas de acero, cordones, tablas de madera) atravesando a los elementos para obtener un mejor resultado y reducir los fallos o inexactitudes.
- El proceso de construcción de una base de datos como se ha detallado, si bien no es un proceso complicado es un proceso laborioso que demanda tiempo además de una correcta clasificación manual; por eso, el tener información ya catalogada tiene un gran valor en ahorro de tiempo. En ingeniería Civil no existen muchas bases de datos que contengan imágenes etiquetadas de libre uso, por ese motivo se desarrolló esta base de datos en los formatos “.txt”, PASCAL VOC y COCO.

## 5. REFERENCIAS BIBLIOGRÁFICAS

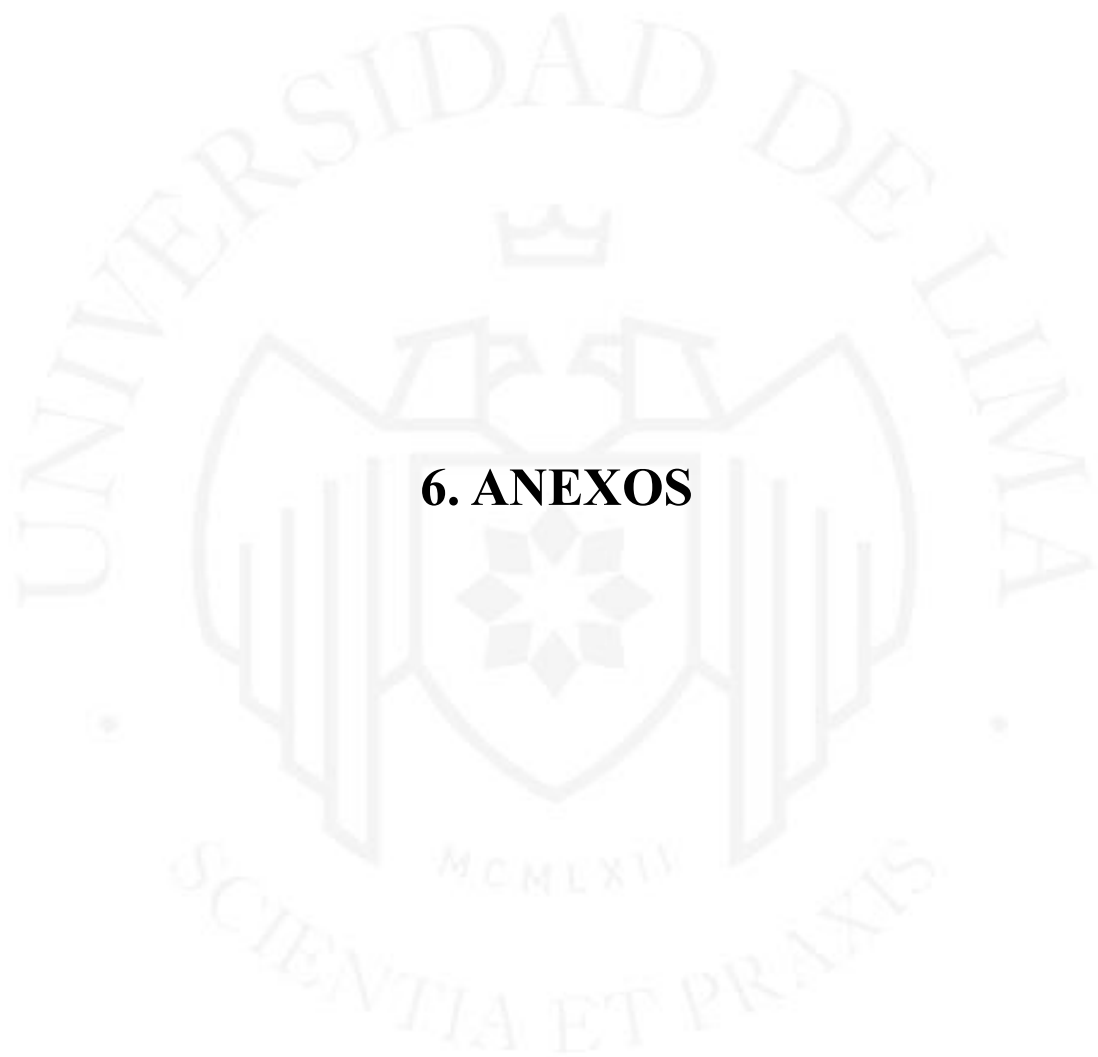
- Abioye, S. O., Oyedele, L. O., Akanbi, L., Ajayi, A., Davila Delgado, J. M., Bilal, M., Akinade, O. O., & Ahmed, A. (2021). Artificial intelligence in the construction industry: A review of present status, opportunities and future challenges. *Journal of Building Engineering*, 44, 103299. <https://doi.org/10.1016/j.jobe.2021.103299>
- Alemán, D. (2017). Técnicas de inteligencia artificial aplicadas a problemas de ingeniería civil. *Revista Arquitectura e Ingeniería*, 11(3), 5. <https://dialnet.unirioja.es/servlet/articulo?codigo=6452841>
- Benyang, D., Xiaochun, L., & Miao, Y. (2020). Safety helmet detection method based on YOLO v4. *2020 16th International Conference on Computational Intelligence and Security (CIS)* (pp. 155-158). Guangxi, China: IEEE. <https://doi.org/10.1109/CIS52066.2020.00041>
- Bianchini, A., & Bandini, P. (2009). Prediction of Pavement Performance through Neuro-Fuzzy Reasoning. *Computer-Aided Civil and Infrastructure Engineering*, 25, 39-54. <https://doi.org/10.1111/j.1467-8667.2009.00615.x>
- Blanco, R., Fontodrona, J., & Poveda, C. (2017). La industria 4.0: El estado de la cuestión. *Economía industrial*(406), 151-164. <https://dialnet.unirioja.es/servlet/articulo?codigo=6343649>
- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv preprint*. <https://doi.org/10.48550/arXiv.2004.10934>
- Caro Moreno, B. A. (2021). *Estudio de aplicaciones de la Inteligencia Artificial en el desarrollo de proyectos de ingeniería civil*. Universidad de Chile. <https://repositorio.uchile.cl/handle/2250/184178>
- Cha, Y.-J., Choi, W., & B. O. (2017). Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, 32, 361-378. <https://doi.org/10.1111/mice.12263>
- Chen, F.-C., & Jahanshahi, M. R. (2018). NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naïve Bayes Data Fusion. *IEEE Transactions on Industrial Electronics*, 65, 4392-4400. <https://doi.org/10.1109/TIE.2017.2764844>
- Chow, J. K., Su, Z., Wu, J., Li, Z., Tan, P. S., Liu, K.-f., Mao, X., & Wang, Y.-H. (2020). Artificial intelligence-empowered pipeline for image-based inspection of concrete structures. *Automation in Construction*, 120, 103372. <https://doi.org/10.1016/j.autcon.2020.103372>

- Chui, M., Manyika, J., & Miremadi, M. (11 de Enero de 2018). What AI can and can't do (yet) for your business. Mckinsey&Company.
- Del Savio, A. A., Luna, A., Cárdenas-Salas, D., Vergara Olivera, M., & Urday Ibarra, G. (2021). The use of artificial intelligence to identify objects in a construction site. *International Conference on Artificial Intelligence and Energy System (ICAIES) in Virtual Mode*. Jaipur, India. <https://doi.org/10.26439/ulima.datasets.13359>
- Del Savio, A. A., Luna, A., Cárdenas-Salas, D., Vergara Olivera, M., & Urday Ibarra, G. (2023). Artificial Intelligence Applied to the Control and Monitoring of Construction Site Personnel. *Advances in Mechanics of Materials for Environmental and Civil Engineering*, 197, 19-29. [https://doi.org/10.1007/978-3-031-37101-1\\_2](https://doi.org/10.1007/978-3-031-37101-1_2)
- Fang, W., Ding, L., Zhong, B., Love, P. E., & Luo, H. (2018). Automated detection of workers and heavy equipment on construction sites: A convolutional neural network approach. *Advanced Engineering Informatics*, 37, 139-149. <https://doi.org/10.1016/j.aei.2018.05.003>
- Flood, I. (2008). Towards the next generation of artificial neural networks for civil engineering. *Advanced Engineering Informatics*, 22, 4-14. <https://doi.org/10.1016/j.aei.2007.07.001>
- Geng, R., Ma, Y., & Huang, W. (2021). An improved helmet detection method for YOLOv3 on an unbalanced dataset. *2021 3rd International Conference on Advances in Computer Technology, Information Science and Communication (CTISC)* (pp. 328-332). Shanghai, China: IEEE. <https://doi.org/10.1109/CTISC52352.2021.00066>.
- Girshick, R. (2015). Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*. Santiago, Chile: IEEE. <https://doi.org/10.1109/ICCV.2015.169>
- Han, K. K., & Golparvar-Fard, M. (2015). Appearance-based material classification for monitoring of operation-level construction progress using 4D BIM and site photologs. *Automation in Construction*, 53, 44-57. <https://doi.org/10.1016/j.autcon.2015.02.007>
- Han, X., Lv, T., Wu, S. W., Li, Y., & He, B. (2019). A remote human activity detection system based on partial-fiber LDV and PTZ camera. *Optics & Laser Technology*, 111, 575-584. <https://doi.org/https://doi.org/10.1016/j.optlastec.2018.10.035>
- 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. <https://doi.org/10.48550/arXiv.1406.4729>
- Hu, J., Gao, X., Wu, H., & Gao, S. (2019). Detection of Workers Without the Helments in Videos Based on YOLO V3. *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. Suzhou, China: IEEE. <https://doi.org/10.1109/CISP-BMEI48845.2019.8966045>

- Huyan, J., Li, W., Tighe, S., Zhai, J., Xu, Z., & Chen, Y. (2019). Detection of sealed and unsealed cracks with complex backgrounds using deep convolutional neural network. *Automation in Construction*, 107, 102946. <https://doi.org/10.1016/j.autcon.2019.102946>
- Jabir, B., Falih, N., & Rahmani, K. (2021). Accuracy and Efficiency Comparison of Object Detection Open-Source Models. *International Journal of Online and Biomedical Engineering (iJOE)*, 17(5), 165-184. <https://doi.org/10.3991/ijoe.v17i05.21833>
- Kochovski, P., & Stankovski, V. (2021). Building applications for smart and safe construction with the DECENTER Fog Computing and Brokerage Platform. *Automation in Construction*, 124, 103562. <https://doi.org/10.1016/j.autcon.2021.103562>
- León, M., & García, Z. (2008). La Inteligencia Artificial en la Informática Educativa. *Revista de Informática Educativa y Medios Audiovisuales*, 5(10), 11-18. <https://studylib.es/doc/6078752/la-inteligencia-artificial-en-la-inform%C3%A1tica-educativa>
- Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path Aggregation Network for Instance Segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA: IEEE. <https://doi.org/10.1109/CVPR.2018.00913>
- López, B. (2005). *Agentes Inteligentes*. Instituto Tecnológico de Nuevo Laredo: <http://www.itnuevolaredo.edu.mx/takeyas/apuntes/Inteligencia%20Artificial>
- Lu, P., Chen, S., & Zheng, Y. (2012). Artificial Intelligence in Civil Engineering. *Mathematical Problems in Engineering*, 2012, 22. <https://doi.org/10.1155/2012/145974>
- Ludermir, T., & Yamazaki, A. (2003). Neural networks for odor recognition in artificial noses. *Proceedings of the International Joint Conference on Neural Networks, 2003*. Portland, OR, USA: IEEE. <https://doi.org/10.1109/IJCNN.2003.1223317>
- Madane, S., & Chitre, D. (2021). Social Distancing Detection and Analysis through Computer Vision. *2021 6th International Conference for Convergence in Technology (I2CT)* (pp. 1-10). Maharashtra, India: IEEE. <https://doi.org/10.1109/I2CT51068.2021.9418195>
- Marín, J. (2012). *Introducción a las Redes Neuronales*. Universidad Carlos III de Madrid. <https://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/Expertos/CursoNN12.pdf>
- MATHIVET, V. (2018). *Inteligencia Artificial para desarrolladores Conceptos e implementación en C# (2ª edición)*. ENI.
- Mekhalfi, M. L., Nicolò, C., Bazi, Y., Rahhal, M. M., Alsharif, N. A., & Maghayreh, E. A. (2021). Contrasting YOLOv5, Transformer, and EfficientDet Detectors for

- Crop Circle Detection in Desert. *IEEE Geoscience and Remote Sensing Letters*, 19, 1-5. <https://doi.org/10.1109/LGRS.2021.3085139>
- Ministerio de Economía y Finanzas. (2021). *Guía Nacional BIM*. [http://www.mef.gob.pe/planbimperu/docs/recursos/guia\\_nacional\\_BIM.pdf](http://www.mef.gob.pe/planbimperu/docs/recursos/guia_nacional_BIM.pdf)
- Miranda, R., Solano, J., & Méndez, A. (2019). Introducción al Aprendizaje Automático con YOLO. *Tecnología Vital*, 2(6). <https://revistas.ulatina.ac.cr/index.php/tecnologiavital/article/view/250>
- Nataprawira, J., Gu, Y., Goncharenko, I., & Kamijo, S. (2021). Pedestrian Detection Using Multispectral Images and a Deep Neural Network. *Sensors*, 21(7), 2536. <https://doi.org/10.3390/s21072536>
- Pacheco, R. (2017). *COMPARACIÓN DEL SISTEMA TRADICIONAL VS LA IMPLEMENTACIÓN DEL BIM (BUILDING INFORMATION MANAGEMENT) EN LA ETAPA DE DISEÑO Y SEGUIMIENTO EN EJECUCIÓN. ANÁLISIS DE UN CASO DE ESTUDIO*. Universidad Católica de Santiago de Guayaquil. <http://repositorio.ucsg.edu.ec/bitstream/3317/7616/1/T-UCSG-PRE-ING-IC-177.pdf>
- Park, M.-W., & Brilakis, I. (2012). Construction worker detection in video frames for initializing vision trackers. *Automation in Construction*, 28, 15-25. <https://doi.org/10.1016/j.autcon.2012.06.001>
- Peng, G., Lei, Y., Li, H., Wu, D., Wang, J., & Liu, F. (2021). CORY-Net: Contrastive Res-YOLOv5 Network for Intelligent Safety Monitoring on Power Grid Construction Sites. *IEEE Access*, 9, 160461-160470. <https://doi.org/10.1109/ACCESS.2021.3132301>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 779-788). <https://doi.org/10.48550/arXiv.1506.02640>
- Rezvani, M., Waugh, L., & Lei, Z. (2022). Technology-Related Challenges in Implementing Mixed Reality (MR) Technology in Precast Concrete Production. *2022: The Digital Reality of Tomorrow*. <https://doi.org/10.57922/tcrc.613>
- Rozada Raneros, S. (2021). Estudio de la arquitectura YOLO para la detección de objetos mediante deep learning. *Universidad de Valladolid. Escuela Técnica de Ingenieros de Telecomunicación*. <http://uvadoc.uva.es/handle/10324/45359>
- Tang, S., Roberts, D., & Golparvar-Fard, M. (2020). Human-object interaction recognition for automatic construction site safety inspection. *Automation in Construction*, 120, 103356. <https://doi.org/10.1016/j.autcon.2020.103356>
- Vector ITC Group. (2018). *Inteligencia Artificial: Pasado, Presente y Futuro*. Digital & Innovation. <https://es.scribd.com/document/453286895/IA-Pasado-presente-y-futuro-Vector-ITC-pdf>

- Wada, K. (2021). Labelme: Image Polygonal Annotation with Python. <https://doi.org/10.5281/zenodo.5711226>
- Wang, C. Y., Liao, H. Y., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A New Backbone that can Enhance Learning Capability of CNN. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, (pp. 390-391). <https://doi.org/10.48550/arXiv.1911.11929>
- Wu, F., Jin, G., Gao, M., HE, Z., & Yang, Y. (2019). Helmet Detection Based On Improved YOLO V3 Deep Model. *2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC)*. Banff, AB, Canada: IEEE. <https://doi.org/10.1109/ICNSC.2019.8743246>
- Yeshasvi, M., Bind, V., & T, S. (2021). Social Distance Capturing and Alerting Tool. *2021 3rd International Conference on Signal Processing and Communication (ICPSC)*. Coimbatore, India: IEEE. <https://doi.org/10.1109/ICSPC51351.2021.9451695>
- Zhou, S., & Song, W. (2020). Deep learning-based roadway crack classification using laser-scanned range images: A comparative study on hyperparameter selection. *Automation in Construction*, *114*, 103171. <https://doi.org/10.1016/j.autcon.2020.103171>
- Zhou, Y., Jiang, L., Liang, Y., Ma, C., Sun, H., Nie, S., & Zuo, Y. (2020). Helmet Detection Algorithm Based on Single Pixel Zoom. *Journal of Physics: Conference Series*, *1682*, 012021. <https://doi.org/10.1088/1742-6596/1682/1/012021>



## **6. ANEXOS**

## Anexo 1: Imágenes de la Base de Datos

Lectura: IMG-X (fecha/ubicación de la cámara/tipo de cámara)



IMG-1 (30-11-2020/Pabellón D1/Bullet)



IMG-82 (18-11-2020/Pabellón I1/PTZ)



IMG-160 (06-01-2021/Pabellón D3/PTZ)



IMG-250 (06-01-2021/Pabellón I1/PTZ)



IMG-400 (07-12-2020/Torre H/PTZ)



IMG-442 (20-11-2020/Pabellón I1/PTZ)



IMG-521 (04-01-2021/Pabellón D1/Bullet)



IMG-684 (07-12-2020/Pabellón I1/PTZ)






IMG-900 (04-01-2020/Pabellón I1/PTZ)



IMG-1165 (08-03-2021/Pabellón I1/PTZ)

# Monica Vergara

## Suelperes

-  Tesis 2025
-  Tesis
-  Universidad de Lima

---

### Document Details

**Submission ID**

trn:oid::1:3231927539

**Submission Date**

Apr 28, 2025, 10:49 AM GMT-5

**Download Date**

Apr 28, 2025, 12:06 PM GMT-5

**File Name**

Tesis\_-\_Jesus\_Suelperes.docx

**File Size**

13.5 MB

**77 Pages**

**13,734 Words**

**76,147 Characters**

# 5% Overall Similarity




The combined total of all matches, including overlapping sources, for each database.

## Filtered from the Report

- ▶ Bibliography
- ▶ Quoted Text
- ▶ Cited Text
- ▶ Small Matches (less than 20 words)

---

## Top Sources

- 5%  Internet sources
- 2%  Publications
- 2%  Submitted works (Student Papers)

---

## Integrity Flags

### 0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

## Top Sources

- 5% Internet sources
- 2% Publications
- 2% Submitted works (Student Papers)

## Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	www.ucavila.es	<1%
2	Internet	docplayer.es	<1%
3	Internet	repositorio.uchile.cl	<1%
4	Internet	repositorio.unh.edu.pe	<1%
5	Internet	www.coursehero.com	<1%
6	Internet	www.semanticscholar.org	<1%
7	Publication	Yongheng Zhou, Lifen Jiang, Yan Liang, Chunmei Ma, Huazhi Sun, ShuaiBao Nie, Y...	<1%
8	Internet	repositorio.ug.edu.ec	<1%
9	Internet	m.dahuasecurity.com	<1%
10	Internet	researchr.org	<1%
11	Student papers	Universidad Cesar Vallejo	<1%

12	Internet	www.bmeguvenlik.com.tr	<1%
13	Student papers	Monash University	<1%
14	Internet	ieeexplore.ieee.org	<1%
15	Student papers	Consortio CIXUG	<1%
16	Student papers	Universidad Rey Juan Carlos	<1%
17	Internet	gestion.pe	<1%