PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

VISIÓN COMPUTACIONAL EN LA INDUSTRIA DE LA CONSTRUCCIÓN: IDENTIFICACIÓN DE EQUIPOS DE SEGURIDAD EN OBRAS MEDIANTE EL USO DE *DEEP LEARNING*

Tesis para obtener el título profesional de Ingeniero Civil

AUTOR:

Miguel Moisés Gutiérrez Torres

ASESOR:

Danny Eduardo Murguía Sánchez PhD.

Lima, Octubre, 2022

RESUMEN

La industria de la construcción es uno de los sectores que expone la vida de los operarios en constante peligro debido a las condiciones laborales que esta demanda como el trabajo en alturas, manejo de maquinaria pesada, entre otros. El uso de equipos de protección colectiva y personal es una medida de seguridad para resguardar la vida de los operarios frente a caídas, colisiones, entre otros accidentes. No obstante, en campo existen actitudes inapropiadas por parte del personal de obra pues estos tienden a retirarse los equipos de seguridad, debido a la disconformidad que produce su peso, el cambio de temperatura, entre otros factores. En efecto, actualmente el control de estos comportamientos es exhaustivo, pues involucra monitorear múltiples actividades proactivamente a lo largo de la jornada laboral.

Este estudio propone evaluar la efectividad de la tecnología *deep learning* en automatizar el reconocimiento de estos equipos de seguridad para comunicar a los supervisores de campos sobre el uso inapropiado de estos objetos y, de esta manera, controlar los accidentes de obra. En consecuencia, se desarrolló una base de datos que comprende imágenes de equipos de seguridad en obra bajo diferentes condiciones visuales: variedad de intraclase (posturas, color, contexturas, estaturas, etc.), intensidades de iluminación, oclusiones, aglomeraciones, entre otros efectos. Este entregable se justifica debido a que en comparación con la literatura se analizó una mayor variedad de equipos de seguridad y se empleó para entrenar y evaluar tres algoritmos más recurridos en la bibliografía (VGG-16, Resnet-18 y Inception-V3), debido al desempeño de sus resultados. Específicamente, el performance del prototipo Inception-V3 alcanzó un valor de 84% en *accuracy* empleando el set de datos de escala regular. Este desempeño indica que las metodologías en aprendizaje profundo pueden contribuir a monitorear equipos de seguridad de obra al disponer de mayor datos, seleccionando modelos más sofisticados y siguiendo

las recomendaciones en este documento para evitar confusiones en la clasificación de objetos.

Asimismo, existen dos contribuciones adicionales. En primer lugar, se realizó un resumen del estado del arte sobre las aplicaciones actuales de la visión computacional en el sector construcción con el objetivo de orientar a otros proyectos a seleccionar un tema de estudio, identificar los logros alcanzados, responder a las limitaciones encontradas y reconocer buenas prácticas. En segundo lugar, el set de base de datos desarrollado presenta una mayor variedad de tipos de EPP's y EPC's, respecto a la literatura, y está disponible a solicitud con el objetivo de estandarizar la existencia de un set de datos propio para el sector construcción y facilitar la aplicación de la visión computacional en esta industria.

Dedicado a mis padres por su apoyo y esfuerzo incondicional.

A mis hermanos por su compañía durante los últimos años de la carrera.

Agradezco al profesor Murguía por su iniciativa y guía en el desarrollo del documento.

Al equipo de investigación que colaboró con los datos de entrada del proyecto.

Contenido

LISTA DE FIGURAS

LISTA DE TABLAS

1. GENERALIDADES

Este primer capítulo presenta la necesidad y beneficios de recurrir a una metodología, en un marco tecnológico, para el control de la seguridad del personal de obra. Además, los objetivos, alcances e hipótesis son descritos en este apartado.

1.1 Introducción

La industria de la construcción es uno de los sectores que mayor expone la vida de los trabajadores en constante peligro. Por ejemplo, el Ministerio de Trabajo y Promoción del Empleo (2019, p. 229) reportó que en el 2019 la industria de la construcción demandó el 11.27% de la mano de obra total en el Perú, contabilizando el 11.58% de accidentes totales en el año. Además, de acuerdo al Bureau of Labor Statistics (2020), ocurren 1000 accidentes mortales cada año en los Estados Unidos. De acuerdo con Fang et al. (2020), las dos principales causas que originan el 90% de los accidentes son los comportamientos inadecuados de los operarios y condiciones de trabajo inapropiadas (áreas congestionadas, incorrecta instalación de equipos temporales como grúa torre y andamios, entre otras). Respecto a la primera categoría, Li et al. (2017) estimó en un estudio que alrededor del 80% de los operarios que sufrieron lesiones en la cabeza no estaban equipados con cascos de protección. Frente a esta circunstancia, Huang & Hinze (2003) determinaron los motivos que incitan al personal de obra a retirarse los EPP y EPC: sensaciones incómodas que produce su peso, incremento del calor corporal en reacción a la temperatura del ambiente, interferencia con la flexibilidad del trabajo, entre otras razones. En este contexto, Fang et al. (2018) consideran necesario el desarrollo de una tecnología que asista a la supervisión a monitorear el uso apropiado de los equipos de protección personal y colectiva en obras de construcción.

En la actualidad, las técnicas convencionales para el control de riesgos en obra se realizan principalmente mediante capacitaciones y supervisiones (Cai, 2020). En las primeras, se entrenan a las cuadrillas a identificar peligros potenciales. Sin embargo, este ejercicio no resulta eficiente, puesto que los trabajadores tienden a ignorar las indicaciones de seguridad, debido a distracciones, humor, fatiga u olvido. Además, Jeelani et al. (2021) declaran que no existe una correlación positiva entre la adopción de estos entrenamientos y aumento de la seguridad del personal. En ese sentido, la estrategia actual consiste en realizar observaciones manuales en el área de trabajo para verificar el cumplimiento de los protocolos de seguridad. No obstante, esta práctica tradicional está limitada por la naturaleza dinámica y proactiva de las obras, debido a que se exige al supervisor monitorear múltiples actividades, que ocurren en simultáneo y durante toda la jornada laboral. En consecuencia, la supervisión actual es propensa a cometer errores y, por ende, generar retrasos de la programación, elevar el costo del proyecto, incumplir estándares de calidad, entre otros efectos negativos.

Asimismo, al sector construcción se le ha calificado como una de las industrias menos digitalizadas (McKinsey, 2016), por lo que resulta ser una razón adicional para orientar las nuevas metodologías en un marco tecnológico. En este contexto, diversas investigaciones se han enfocado en desarrollar propuestas en base a visión computacional, debido a que sus funciones permiten automatizar actividades que requieran de emplear la visión humana, ahorrando tiempo y costos; no obstruye ni interfiere con el confort y/o libertad de trabajo de los operarios, a comparación con otras tecnologías como internet de las cosas (IoT), equipos de radio frecuencia, entre otras; y representa un sistema proactivo, es decir, trabaja de manera continua y precisa (Konstantinou, 2018).

Entre los componentes empleados por la comunidad de investigadores para elaborar sistemas en visión computacional se comprende la planificación de equipos (cámaras y

ordenadores); diseño del modelo con una función específica (identificar, localizar, monitorear, etc.); y la aplicación del sistema en proyectos reales, para evaluar su desempeño verdadero (Xu et al., 2020). En esencia, los primeros estudios en *computer vision*, aplicados en la construcción civil, se basaron en modelos tradicionales en *machine learning* para la extracción de caracteres y reconocimiento de patrones. Sin embargo, estas metodologías convencionales no resultan eficientes para procesar imágenes bajo condiciones complejas: desenfoques, debido a vibraciones; variaciones de iluminación solar; ráfagas de viento; polvo; texturas de los materiales de construcción; entre otros (Fan et al., 2018). En efecto, las investigaciones más recientes emplean métodos en *deep learning* para contrarrestar los factores mencionados. Por lo tanto, la presente tesis investigará los antecedentes de esta herramienta para validar la efectividad de este sistema en identificar equipos de seguridad y, en base a ello, promover su aplicación con el objetivo de controlar los riesgos en obras de construcción.

1.2 Formulación de preguntas de investigación

En referencia al apartado anterior, se plantea la siguiente pregunta para el control de riesgos en los proyectos de construcción civil.

¿Es la técnica *Deep learning* una herramienta robusta para identificar equipos de seguridad en obras de construcción?

Preguntas específicas

¿Cuáles son los principales algoritmos de la visión computacional en el sector de la construcción?

¿Qué información procesarán los algoritmos para validar su desempeño?

¿Cuál es el desempeño del aprendizaje profundo en la industria de la construcción?

1.3 Objetivos

1.1.1 Objetivo general

Validar la efectividad de la tecnología *deep learning* en la identificación de equipos de protección personal y colectiva en obras de construcción.

1.3.1 Objetivos específicos

- Sintetizar los algoritmos en visión computacional aplicados en la industria de la construcción.
- Desarrollar una base de datos que comprenda equipos de protección propios del sector construcción.
- Evaluar el desempeño de tres algoritmos más recurrentes en la literatura sobre la base de datos propuesta.

1.4 Alcances

El dominio de la visión computacional abarca diversas tareas y las principales se denotan por clasificar la clase del objeto, localización de objetos, monitorear el flujo óptico del mismo, reconstrucción tridimensional, estimar posturas y acciones, entre otras aplicaciones (Szeliski, 2021). En efecto, la presente investigación de pre grado se enfocará en la función fundamental de *computer vision*: clasificación de imágenes. Además, esta tecnología resulta ser la función base que sirve de raíz para elaborar las otras tareas de mayor complejidad. Por tanto, los resultados de la presente investigación permiten extenderse a otros temas de investigación.

1.5 Hipótesis

La tecnología *deep learning* es capaz de identificar los equipos de seguridad con una precisión mayor al 70% en un set de datos de escala regular, promedio de 1000 ejemplares por objeto sin aumento de data.

1.6 Justificación

La presente investigación busca extender la aplicación de los prototipos de mejor desempeño en detección de objetos en el campo de la construcción civil. En efecto, las investigaciones previas se enfocaron en evaluar estos modelos de visión computacional en otros sectores industriales o proyectos que permiten capturar imágenes con caracteres menos complejos, en comparación con los factores propios de obras de construcción que dificultan el desempeño de esta tecnología*:* desenfoques, debido a las ráfagas de viento y vibraciones; la variación de la iluminación solar; oclusiones entre operarios y maquinarias; entre otros. Por tanto, reside la incertidumbre de conocer el desempeño real de los sistemas de mayor eficiencia en el rubro de la construcción civil.

Asimismo, actualmente existen bases de datos públicas que comprenden objetos generales de la vida diaria y que se emplean para entrenar modelos neuronales. Sin embargo, existen pocos sets de imágenes con acceso abierto que comprendan entidades de construcción civil. En efecto, este hecho se debe a que la elaboración de una base de datos demanda grandes cantidades de horas hombres para la recolección de imágenes in-situ y etiquetado de las mismas. Por tanto, el desarrollo de sets de entrenamiento que comprendan elementos propios de la construcción civil permitirá agilizar y promover la elaboración de proyectos en visión computacional aplicados en la industria de la construcción.

Asimismo, el desarrollo de herramientas prácticas asociadas al control de riesgos en los proyectos de construcción permite apoyar al área de supervisión en sus actividades

laborales con el objetivo de eliminar o disminuir los errores: retrasos en la programación y aumento del costo del proyecto, debido al ausentismo de los operarios accidentados y jubilación temprana de los mismos (producto de los accidentes que afectan permanentemente su postura y capacidad física); entre otros factores.

Por último, existen escasas síntesis sobre las aplicaciones de estos modelos, por lo que, la elaboración de un resumen actual permite identificar las metodologías modernas, buenas prácticas y oportunidades de investigación.

1.7 Metodología

Se plantea realizar las siguientes actividades:

- Se identificarán las metodologías modernas y tradicionales de la visión computacional.
- Se investigarán las etapas que conforman la metodología moderna de la visión computacional.
- Se sintetizarán las aplicaciones actuales de los modelos en la industria de la construcción para identificar las mejores prácticas, oportunidades de investigación y los principales modelos empleados en la literatura.
- En referencia a la base de datos, se recolectarán imágenes de campo que comprendan diferentes condiciones visuales como variaciones en iluminación, clase de objetos, escalas, oclusiones, desenfoques, entre otras características, para elaborar un modelo capaz de generalizar el reconocimiento de objetos en diferentes escenarios de construcción civil.

- Finalmente, se desarrollarán y evaluarán tres modelos más recurrentes en la literatura en base a *deep learning* empleando la base de datos propuesta.

2. ESTADO DEL ARTE

En este capítulo se viajará en el tiempo para concebir la idea del origen de la capacidad visual y se detallarán los hitos históricos más relevantes sobre la visión humana. Además, se enmarcarán los conceptos técnicos necesarios para aplicar la metodología de aprendizaje profundo. Seguidamente, se comprenderá el proceso que involucra desarrollar proyectos en *computer vision* aplicados en la construcción civil.

2.1 Visión computacional: historia y fundamentos

2.1.1 Evolución de la capacidad visual

De acuerdo a Parker (2004), en la teoría de la explosión cámbrica, el origen de la capacidad visual data alrededor de 543 millones de años atrás con la creación de los primeros órganos visuales en fósiles trilobites. En efecto, el autor indica que las especies depredadoras presentaron un comportamiento ineficaz en esa época debido a la ausencia de capacidades sensoriales. Este carácter desencadenó el desarrollo de un estímulo externo: se considera a los ojos como órganos capaces de interpretar las ondas de radiación solar y producir imágenes a través de ellas. Además, en este mismo trabajo se explica que la existencia de dos ojos en lugar de uno se debe a propósitos dimensionales.

En años posteriores se desarrollaron hipótesis que intentaron explicar el proceso que realiza el ojo para capturar la información visual: proyección de ondas en la retina, la segmentación de la información en la corteza cerebral, transmisión de datos hacia otras zonas del cerebro, entre otras etapas. Evidencia de estos proyectos, se presentaron en los años 1500 cuando Leonardo da Vinci desarrolló modelos artificiales para replicar funcionamiento de los ojos (Kenneth, 1955). En esencia, en la transcripción de Kenneth (1955) se comenta que el proyecto *camera obscura* significó la creación de una de las

primeras cámaras fotográficas en la historia de la humanidad. El flujo de trabajo consistió en el ingreso de luz hacia una caja oscura a través de un orificio equipado con una esfera de cristal llena de agua, representando la pupila. Por lo visto, el investigador concluye que cuando los rayos de luz se fuerzan al ingresar a través de un espacio reducido (el agujero de la caja) se genera una proyección de la imagen exterior distorsionada, tanto vertical como horizontalmente, en la cara más profunda de la caja oscura. Por lo que se sabe, este proceso ocurre similarmente en la corteza estriada del cerebro, según las respuestas celulares observadas en ensayos en animales al emitir rayos de luz hacia sus ojos (Hubel & Wiesel, 2004). Asimismo, los investigadores del ensayo indican que, posteriormente, la información fluye a otras regiones cerebrales para corregir la distorsión de las imágenes.

En la década de 1960, se concibió la idea de dotar a las computadoras con capacidad visual. La primera tesis en visión computacional, conocida como *block world*, consistió en extraer la información tridimensional de las aristas de sólidos graficados en un dibujo de papel (Lawrence, 1963). Hacia el año 2001, se desarrolló el primer modelo en ingeniería, *Haar Cascades*, que permitió a los ordenadores identificar rostros de personas (Viola & Jones, 2001). En el 2005, el modelo matemático *Histogram of Oriented Gradients* facilitó la identificación de objetos de la vida diaria con mayor precisión (Dalal & Triggs, 2005). El siguiente hito histórico más reciente data en el año 2012 con la propuesta de redes neuronales artificiales, comúnmente referidas como *convolutional neural networks* (CNNs). Esta metodología opera imitando el mecanismo que realiza la corteza cerebral y su desempeño disminuyó el error de clasificación de los modelos tradicionales en un 50% aproximadamente en el desafío *ImageNet* de *Stanford University* (clasificar 1000 clases de objetos en 1.2 millones de imágenes) (Krizhevsky et al., 2017). En el año 2014, se marca el inicio de la aplicación de los prototipos CNNs en la industria de la construcción y, hasta

la actualidad, son las herramientas representativas de la visión computacional (Akinosho et al., 2020).

2.1.2 Metodologías: aprendizaje de máquina y procesamiento de imágenes

El concepto general de la inteligencia artificial consiste en dotar a las máquinas de capacidades e inteligencia humana para que sean capaces de solucionar problemas de la vida diaria (Gollapudi, 2019). En efecto, la visión computacional representa el mecanismo que permite a las máquinas procesar, analizar y comprender la información visual del mundo real con el objetivo de entregar información numérica o simbólica (Mohammad, 2016). Ejemplos de este entregable son la identidad de un objeto, la localización, velocidad, trayectoria, cantidad, entre otras. Asimismo, según Szeliski (2021), la visión computacional emplea de 2 herramientas para cumplir con estos objetivos: *machine learning*, específicamente para la identificación de patrones, y procesamiento digital de imágenes, técnicas que manipulan la imagen para estandarizar el análisis de ellas por los algoritmos. La Tabla 1 presenta los objetivos principales de la visión computacional, según la tecnología *machine learning* empleada, mientras que la Figura 1 busca diferencias los objetivos de *computer vision* respecto al procesamiento de imágenes.

Tabla 1. *Funciones de la visión computacional.*

Tecnología			Metodología
Tipo	Descripción	Objetivo	
Clasificación de imágenes	Es la primera tarea que elaboraron los modelos iniciales.	i). Identifica la clase del objeto.	*Convolutional neural networks*
Detección de objetos	Requiere del clasificador de imágenes para cumplir con su objetivo.	i). Identifica la clase del objeto. ii). Localiza el objeto en la imagen.	*Two-stage detectors: R-CNN, mask R-CNN, etc.*

			One-stage detectors: YOLO, SSD, etc.
Monitoreo de objetos	Resulta posible extraer las velocidades, trayectorias, entre otros parámetros del elemento.	i). Identifica la clase del objeto. ii). Localiza el objeto. iii). Monitorea la entidad de estudio	*Mutiple object tracking*
Reconocimiento de actividades	Consiste en entrenar el clasificador a identificar caracteres espaciales (representación del objeto) y temporales (movimiento del objeto) en secuencias de imágenes para reconocer acciones.		*Description based method* *Hidden markov models*
Estimación de posturas humanas	Esta herramienta busca estimar las posturas humanas a través de imágenes y videos. Además, su aplicación se presta para reconocer actividades, analizar posturas ergonómicas, entre otras.		*Pictorial structures* *Deep learning: stacked hourglass network*

Nota. Tomado de "Taxonomy, state of the art, challenges and applications of visual understanding: A review" por Khanday & Sofi, 2021.

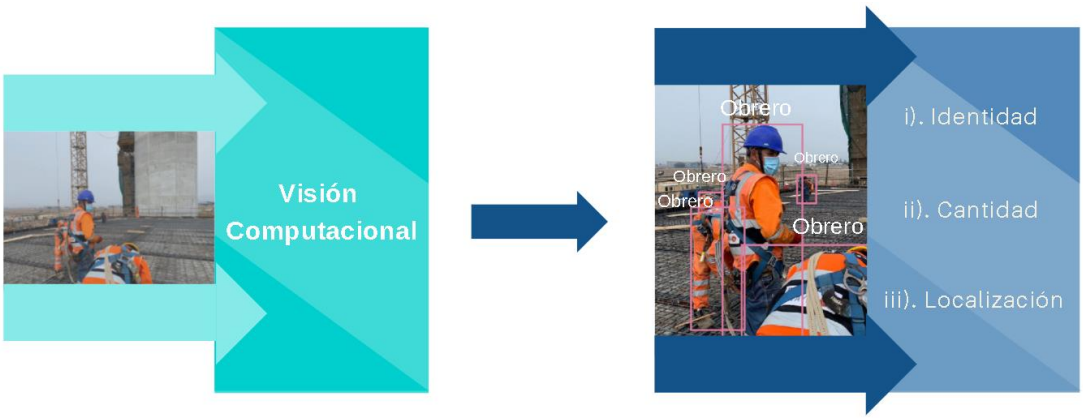


Figura 1. *Procesamiento de imágenes versus visión computacional.*

Nota. Tomado de "Computer Vision: Algorithms and Applications", por Szeliski, 2021.

Asimismo, Szeliski (2021) menciona que el aprendizaje de máquina presenta tres diferentes enfoques, según la interacción entre la persona y la máquina. En primer lugar, el modo supervisado consiste en la participación de la persona en corregir los errores del sistema durante la fase de aprendizaje por medio de tareas de etiquetación (ilustrar elementos verdaderos y falsos). Por lo contrario, se explica que el enfoque no supervisado refiere a un sistema independiente que no requiere de un instructor para aprender a identificar patrones. Asimismo, el autor refiere al último caso como un enfoque semi supervisado que representa la combinación de los dos modelos anteriores: durante una fracción de la fase de entrenamiento el modelo se entrena con el apoyo de la persona, mientras que en el tiempo restante el prototipo comprende patrones individualmente.

2.1.2.1 Aprendizaje de máquina tradicional

La comunidad de investigadores en visión computacional ha desarrollado diferentes metodologías a lo largo del tiempo, como se aprecia en la Figura 2. En consecuencia, estas herramientas se clasifican en dos perspectivas: enfoque tradicional y moderno. Según Joshi & Patel (2020), ambas metodologías se enfocan en realizar dos principales tareas denominadas *feature extraction* y *image classification.* Respecto a los modelos tradicionales, estos realizan la extracción de caracteres por medio de técnicas *feature*

detection y *feature description. Feature detection* consiste identificar las características del objeto (curvas, contornos, esquinas, etc.) a una escala local (opera dentro de la etiqueta) o global (se obtiene información de toda la imagen). Ejemplos de estos algoritmos tradicionales son SIFT, SURF, FAST, BRIEF, ORB, entre otros. Asimismo, Joshi & Patel (2020) mencionan que *feature description* se encarga de transformar los caracteres identificados en un formato compatible por el clasificador, por lo general, un vector. Ejemplo de descriptores ilustrados en su trabajo son SIFT, SURF (trabajan extracción y descripción), BRISK, FREAK, HoG. En referencia a la tarea *image classification*, Szeliski (2021) menciona que el propósito de esta tarea es la identificación de patrones, presentes en la configuración de los formatos descriptivos (vectores), y correlacionarlos con una etiqueta que represente la clase del objeto. Uno de los primeros ejemplares, mencionados por el autor y que efectúan esta tarea, es la técnica de regresión logística. Esta consiste en generar una ecuación lineal que mejor acomode a los vectores descriptivos. Sin embargo, este modelo no permite clasificar data que no sea separable linealmente. Por tanto, el autor ilustra otras técnicas que han sido desarrolladas para compensar las limitaciones de los primeros modelos: la máquina de soporte de vectores, es capaz de generar hiperplanos que agrupan distribuciones complejas de los formatos descriptivos*; clustering,* empleado para desarrollar clasificadores no supervisados; *tree´s decisión; forest decisión; principal component analysis,* entre otros.

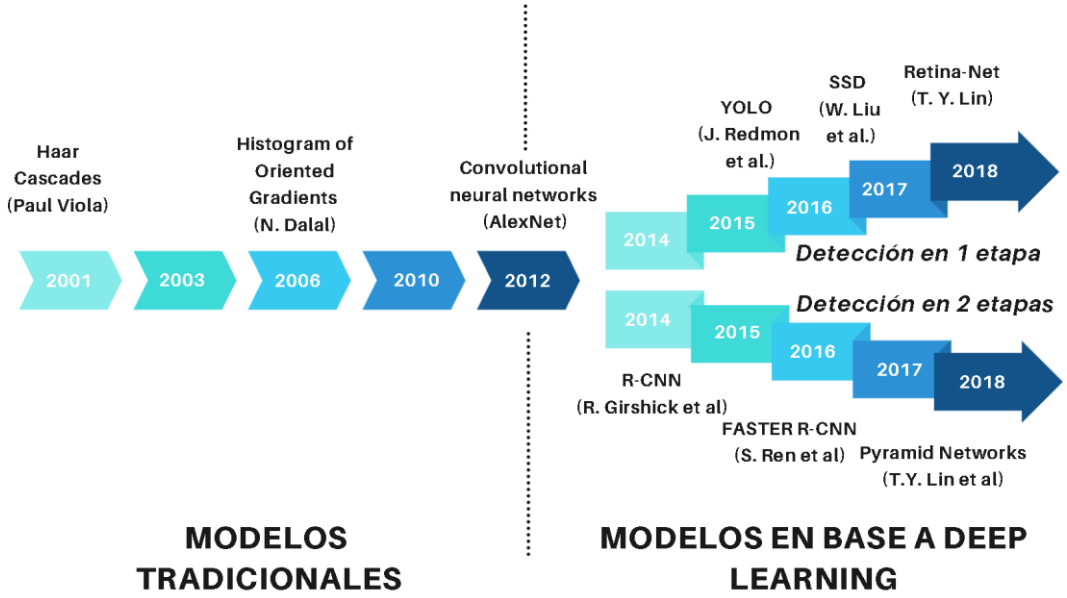


Figura 2. *Contexto histórico de las metodologías en visión computacional.*

Nota. Tomado de "Rapid Object Detection Using a Boosted Cascade of Simple Features" por P. Viola, 2001; "Histograms of Oriented Gradients for Human Detection" por Dalal, 2006; "Deep learning in the construction industry: A review of present status and future innovations", por Akinosho et al., 2020.

2.1.2.2 Aprendizaje de máquina neuronal: *deep learning*

Según Fan et al. (2017), *deep learning* refiere a una colección de algoritmos en *machine learning* capaces de identificar patrones, en data compleja y no lineal, imitando el proceso que realizar la corteza cerebral humana. La Figura 3 ilustra la configuración de uno de los primeros algoritmos bajo este enfoque denominado *Multilayer Perceptron* (Gonzalez et al., 2018). En esencia, mediante esta metodología la extracción de caracteres simples, o *feature extraction*, (bordes, curvas, formas, etc.) se efectúan en las capas de entrada, mientras que la elaboración de las partes del objeto se realiza en las capas intermedias. Respecto a la última capa, esta se encarga predecir la clase del objeto a partir de las características identificadas. Además, según Gonzalez et al., (2018), este enfoque emplea técnicas en *backpropagation* para reducir los errores de clasificación.

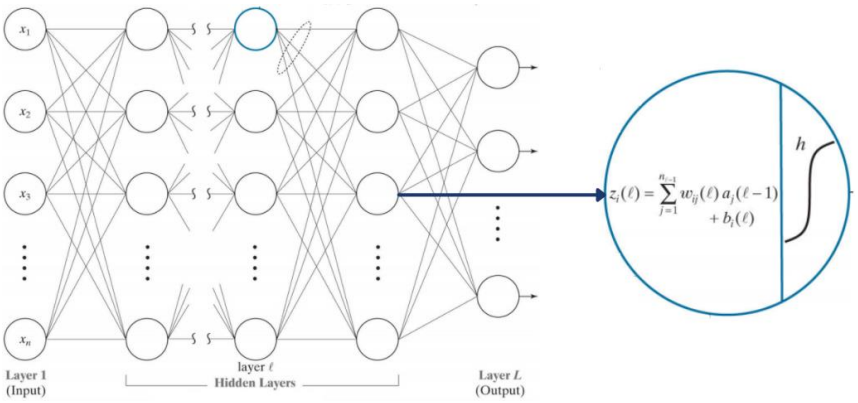


Figura 3. *Idealización de un algoritmo en aprendizaje profundo en base a la corteza cerebral humana: neuronas, capas de entrada, capas ocultas y capas de salida.*

Nota. Tomado de "Digital Image Processing", por Gonzalez et al., 2018.

Actualmente, las principales estructuras en *deep learning* empleadas en la literatura son las siguientes: *convolutional neural networks*, dispone de mejor desempeño en *image classification, object detection* y *segmentation*; *recurrent neural networks*, principalmente empleadas en las ramas de *speech recognition* y *natural language processing*; *autoencoder networks*, se desarrollan bajo entrenamientos no supervisados; *generative adversial networks*; entre otros (Akinosho et al., 2020). En este contexto, Brownlee (2016) identifica que los algoritmos de clasificación en base a la configuración CNNs poseen tres tipos de capas: convolucionales, *pooling* y conexión total. En efecto, el autor menciona que las dos primeras capas se encargan de reducir la presencia de caracteres en la imagen, prevaleciendo la información más relevante, mientras que la tercera capa es localizada, generalmente, al final de la red para realizar predicciones de clases. Un ejemplo de este flujo de trabajo es ilustrado en la Figura 4. De igual manera, Brownlee (2016) define tres funciones importantes al final de la estructural neuronal: función de activación, transforma los valores obtenidos en probabilidades de clases; función de pérdidas, se encarga de estimar el error del modelo al final de cada ciclo; y la función de optimización, se enfoca en actualizar los parámetros del modelo con el objetivo de reducir la función de pérdidas en cada siguiente iteración.

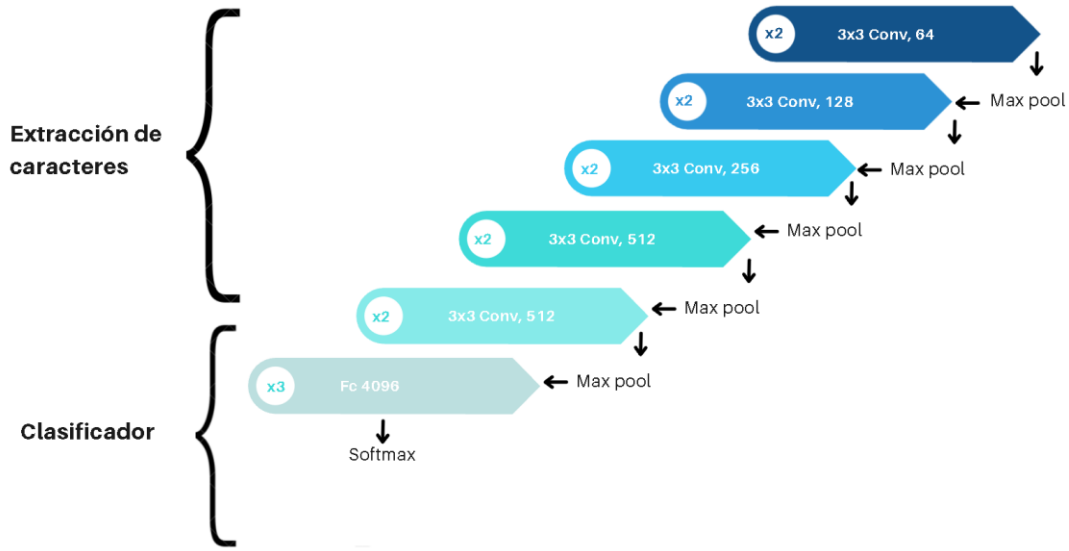


Figura 4. *Flujo de trabajo de la arquitectura convolucional VGG-16.*

Nota. Tomado de "Very Deep Convolutional Networks for Large-Scale Image Recognition", por Simonyan & Zisserman, 2014.

Por otro lado, la Figura 5 ilustra algunos modelos en detección de objetos, compuestos por clasificación y localización, y, por lo general, esta tarea se efectúa mediante una o dos etapas (Fang et al., 2020). Respecto a "*one-stage",* la primera versión propuesta en la literatura se denomina *Region Fully Convolutional Network* (R-CNN). En efecto, este prototipo consiste en ejecutar una búsqueda selectiva de 2000 regiones de interés (posible localización del objeto en estudio) en la imagen y, luego, emplea la red neuronal convencional para realizar la tarea de clasificación (Girshick et al., 2014). Las limitaciones de este modelo se exhiben en la lentitud del procesamiento para describir todas las regiones propuestas, por lo que, las versiones mejoradas del algoritmo en orden cronológico son las siguientes: Fast R-CNN, Faster R-CNN y Mask R-CNN (Ren et al., 2017). Respecto a los prototipos "*two-stage",* se reconoce que *You Only Look Once* (YOLO) dispone de mayor velocidad que la clase R-CNN para efectuar el proceso de identificación de objetos. Sin embargo, presenta complicaciones al identificar objetos diminutos en la imagen, por lo que, sus valores de *accuracy* son inferiores al modelo R-CNN (Fang et al., 2018). Un segundo ejemplar, empleado frecuentemente en la literatura, en base a la convención "*two-stage"*

es *Single Shot Multibox Detector* (SSD) y, en esencia, no presenta las desventajas de los modelos anteriores (Fang et al., 2018).

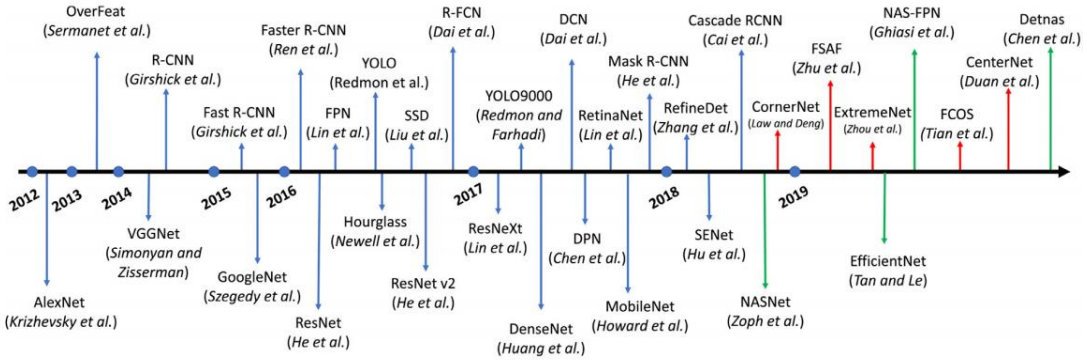


Figura 5. *Evolución de los modelos en detección de objetos en base a la red convolucional.*

Nota. Tomado de "Taxonomy, state of the art, challenges and applications of visual understanding: A review" por Khanday & Sofi, 2021.

Asimismo, Rosebrock (2017a) menciona que existen diversas ventajas que prevalecen en el aprendizaje de máquina moderno respecto al formato tradicional. Por ejemplo, los prototipos en *deep learning* materializan el espacio de memoria de las fotos dentro de sus hiperparámetros, mientras que los sistemas clásicos requieren de almacenar las imágenes en el computador cada vez que se requiera aplicar el modelo. Además, los algoritmos en aprendizaje profundo tienden a incrementar su precisión al procesar mayor data en la etapa de entrenamiento o aumentar las capas del modelo. Por lo contrario, la precisión de los algoritmos, bajo el enfoque tradicional, no se optimiza a pesar de continuar incrementando los ejemplares de entrenamiento. Asimismo, el autor recomienda emplear modelos en *deep learning*, puesto que son capaces de trabajar directamente con los pixeles de las imágenes, en lugar de emplear algoritmos matemáticos que transformen imágenes en vectores. Adicionalmente, según Singaravel et al. (2018), los prototipos en *deep learning* se asemejan más a un trabajo colaborativo, puesto que resulta factible de reutilizar las capas de la red por diferentes miembros de la organización. De esta manera, es posible disminuir el tiempo de entrenamiento de sus modelos en nuevos sets de datos.

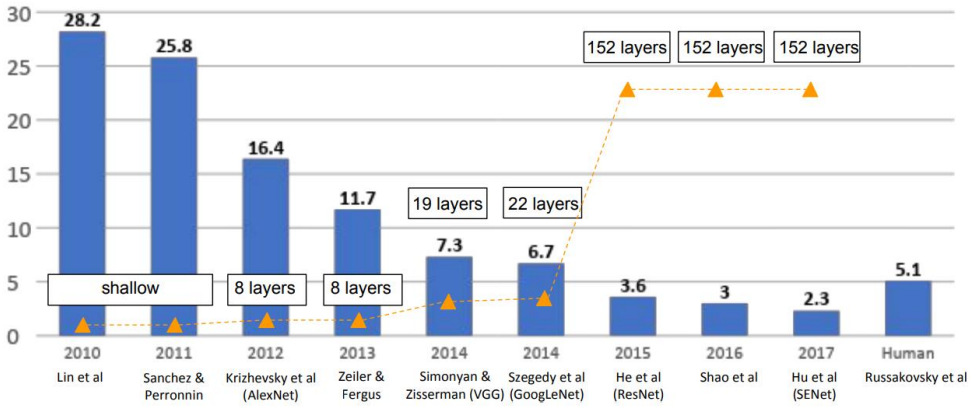


Figura 6. *Desempeño histórico entre modelos tradicionales y en base a deep learning realizados en la base ImageNet (año vs porcentaje de error de clasificación).*

Nota. Tomado de Fuente: Li, Johnson & Yeung (2019)

2.1.3 Etapas del proyecto en visión computacional

En el apartado anterior, se comprendió que los modelos de visión computacional en base a *deep learning* resultan ser técnicas superiores, respecto a las herramientas tradicionales en *machine learning*, en cuanto al aprendizaje de patrones y, por ende, solucionar problemas propuestos. A continuación, se pretende concebir las etapas que conforman la aplicación de un sistema en visión computacional en el sector construcción.

2.1.3.1 Colección y etiquetado de la data

De acuerdo a Yao (2018), esta etapa resulta ser una de las mayores complicaciones en los proyectos de visión computacional, debido a que la elaboración de un set de datos consume grandes cantidades de hora hombre. Por tanto, una alternativa es seleccionar un set de datos público. Por ejemplo, Golparvar-Fard et al. (2013) presentaron el primer set de videos que comprende excavadoras y volquetes para entrenar modelos en *action recognition*. Similarmente Tajeen & Zhu (2014) compartieron 2000 imágenes que contienen 5 tipos de equipos pesados empleados para el movimiento de tierras. Posteriormente, los autores Xiao & Kang (2020) extendieron el número de clase de maquinarias a 10 para entrenar modelos en *object recognition*. De igual manera, Nath et al. (2020) propusieron un set de

datos, denominado *Pictor-v3*, que contiene 4700 instancias de obreros equipados con cascos y chalecos reflectivos. No obstante, según Yao (2018) hasta la actualidad no existe una plataforma que permita a la comunidad de ingenieros civiles compartir la data recolectada en su investigación. Asimismo, si se desea analizar nuevos objetos propios de la construcción lo más probable es optar por desarrollar un set de datos nuevo.

De esta manera, el desarrollo de un set de dato comprende dos etapas: recolección de imágenes y etiquetado. Respecto a la primera fase, Xu et al. (2020) indican que la práctica común para la recolección de información es el uso de cámaras 2D (celulares, tabletas, cámaras de seguridad) y equipos 3D (*depth cameras, laser scanners*, entre otros) situados en obra, de manera estática o dinámica. En efecto, según Seo et al. (2015) la data tridimensional permite alcanzar mayores niveles de precisión en los modelos de *machine learning*, debido a que capturan mayor información. En la práctica estática, Xu et al. (2020) comentan que la mejor localización de los equipos corresponde a posiciones elevadas para cubrir mayor rango visual. Por ejemplo, la instalación de los dispositivos en la cabina de la grúa torre, andamios y otras estructuras son alternativas empleadas en la literatura. Respecto al modo dinámico, se menciona que la práctica general consiste en incorporar cámaras en drones y equipos de protección personal (lentes de seguridad, cascos, etc.). En efecto, esta alternativa incluye un mayor costo, pues se debe adquirir los dispositivos electrónicos para cada obrero, y un menor desempeño en los algoritmos, debido a que la calidad de las imágenes se distorsiona por la vibración del objeto volador no tripulado o incluso movimientos bruscos del operario. De igual manera, Kim et al. (2019) indican una serie de pautas a seguir para optimizar la instalación de las cámaras en cuanto a localización, cantidad de equipos, orientación y tipos de cámaras según la accesibilidad, oclusiones y espacio geométrico del área de trabajo. Otros investigadores se valieron de la

tecnología *Building Information Modeling* para determinar la mejor ubicación de los dispositivos de video (Albahri & Hammad, 2017).

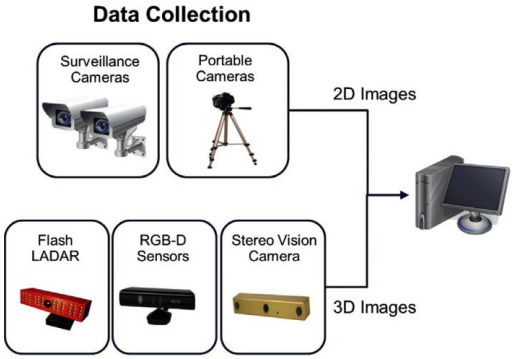


Figura 7. *Tipos de dispositivos para capturar la información en campo.*

Nota. Tomado de "Computer Vision Techniques for Construction Safety and Health Monitoring", por Seo et al., 2015.

Asimismo, Xiao & Kang (2020) ofrecen una alternativa diferente para la colección de la información. Esta consiste en recopilar datos disponibles en internet a través de búsquedas manuales o de manera automatizada por medio de minería de datos en internet. Sin embargo, esta herramienta presenta la limitación de que está expuesta a recopilar datos repetidos, sin variedad e información irrelevante debido a la polisemia. Frente a esta problemática, Yao (2018) propuso una metodología para elaborar set de datos de calidad a través de sitios web y emplearlos para entrenar algoritmos de clasificación sin necesidad de etiquetar la data.

La segunda fase para desarrollar un set de datos consiste en etiquetar las imágenes, solo si es que se trabaja con sistemas de *machine learning* en modo supervisado. En efecto, esta tarea refiere a enmarcar los objetos de estudio presentes en las imágenes y, por lo general, hasta este instante se suele emplear más del 50% del tiempo total del proyecto en visión computacional (Cognilytica, 2020). Específicamente, los criterios de anotación varían y dependen de la función del modelo de *computer vision* (*object recognition, object tracking,*

action estimation, etc). La técnica más recurrente es la identificación del área de interés mediante una etiqueta rectangular denominada *bounding box*. Otras etiquetas más complejas consisten en la zonificación mediante *keypoints*, *masking*, *cuboids*, etc. (Cloudfactory, 2020). En esencia, estas herramientas están incluidas en aplicaciones web o softwares como *labelme*, *super annotate* y *supervisely*, que permiten realizar este proceso de manera manual. Una segunda alternativa, es tercerizar la tarea a servicios de confianza. Por ejemplo, gla plataforma de *Amazon*, denominada MTurk, es la recomendada por la literatura (Xiao & Kang, 2020).

En este instante es importante resaltar que la comparación correcta entre algoritmos de visión computacional debe desarrollarse empleando el mismo set de datos. Caso contrario, es probable que algunos modelos se estén analizando en un set de imágenes con características más complejas (Fang et al., 2020). Algunos de estos factores se describen en la Tabla 2.

Tabla 2. *Principales atributos desafiantes en el set de entrenamiento.*

Occlusion	El objeto se encuentra parcial o completamente cubierto por otro elemento.
Motion blur	Los desenfoques son la pérdida de detalle de información en la imagen.
Background clutter	El entorno alrededor del objeto presenta textura, color y características similares a él.
Illumination variation	El espacio de color RGB por su naturaleza no resulta adecuado para capturar grandes longitudes de ondas.
Out-of-view	Objetos truncados que presentan una porción no visible en la imagen.
Scale variation	Los modelos en visión computacional deben ser capaces de reconocer al mismo objeto en sus diferentes presentaciones de escala.
Intra-class variation	En referencia a los diferentes diseños y estereotipos de la clase en estudio
Deformation	En referencia a las diferentes posturas o molduras que puede presentar el objeto en estudio.

Nota. Tomado de "Two-Dimensional Visual Tracking in Construction Scenarios: A Comparative Study", por Xiao & Zhu, 2018.

2.1.3.2 Procesado de la información

Esta siguiente etapa existe con el propósito de extraer la información relevante dentro de las imágenes. En efecto, Rosebrock (2016) comparte algunas de las técnicas para corregir los caracteres defectuosos presentes en las imágenes (como desenfoques, oclusiones, *background clutters*, iluminaciones, entre otros defectos generados por vibraciones, vientos, polvo, exceso de luz solar, sombras, lluvias y demás factores característicos de obra). Por ejemplo, las tácticas más frecuentes son la transformación del espacio de color de la imagen (RGB, escala de grises, etc.); *masking*, para enfocar el objeto de estudio; entre otras. Además, el autor menciona que una tarea indispensable es el escalamiento de las imágenes a una dimensión estándar para facilitar las operaciones matemáticas concebidas por la red neuronal. Asimismo, según Satya (2016), en esta fase también se suelen aplicar técnicas de *data augmentation* para incrementar la cantidad de imágenes en el set de entrenamiento. De esta manera, se evita el fenómeno *overfitting*, que consiste en la inhabilidad del prototipo para generalizar la clasificación de objetos en un set diferente.

Adicionalmente, Rosebrock (2017b) indica que en esta etapa se debe realizar una separación de las imágenes en 2 categorías: set de entrenamiento y set de evaluación. En el primer grupo, la información es destinada para ajustar los parámetros del modelo que le permitirán reconocer patrones similares presentes en otros escenarios. La segunda categoría se emplea para evaluar el desempeño del algoritmo al realizar predicciones en imágenes no visualizadas con anterioridad. En efecto, el autor menciona que la selección de ratios comúnmente se distribuye en ¾ para el *training set* y el resto en el *testing set.*

De igual manera, en esta sección se define la selección del modelo en *machine learning* a desarrollar según las funciones de *computer vision* que se deseen aplicar en la data procesada: *image clasification, object detection, object tracking, face detection, posture estimation,* entre otras. En este contexto, la literatura acostumbra a emplear tres principales indicadores para identificar el desempeño de la función *object detection* en su aplicación: *precision,* es el ratio de las predicciones correctas efectuadas respecto al total de predicciones realizadas; *recall*, refiere a los objetos identificados correctamente respecto al total de objetos que existen en la data; y, *accuracy*, es la combinación de los índices anteriores (Fang et al., 2018).

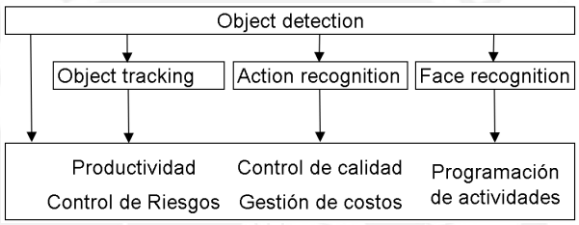


Figura 8. *Enfoque para abordar aplicaciones en el sector construcción empleando las tecnologías de la visión computacional.*

Nota. Tomado de "Computer Vision Techniques for Construction Safety and Health Monitoring", por Seo et al., 2015.

Tabla 3. *Indicadores de algoritmos en detección y localización de objetos.*

Accuracy	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
Precision	$Precision = \frac{TP}{TP + FP}$
Recall	$Recall = \frac{TP}{TP + FN}$

Nota. TP: verdaderos positivos equivale a la cantidad de objetos identificados correctamente. TN: verdaderos negativos refieren al hecho de no realizar predicciones pues no se presentaron determinados objetos en la imagen. FP: falsos positivos refiere al número de objetos identificados de manera incorrecta. FN: falsos negativos significa las predicciones que debieron realizarse pues sí se presentaron objetos en la imagen. Tomado de "A Deep Learning-Based Method for Detecting Non-Certified Work on Construction sites", por Fang et al., 2018.

Del mismo modo, Angah & Chen (2020) proponen evaluar el desempeño de la tarea *object tracking* mediante el indicador MOTA para identificar el error de los modelos en correlacionar trayectorias incorrectas de obreros al ocluirse entre ellos, presentar trayectorias desfasadas respecto a la ruta real, entre otros. Además, según Taha & Hanbury (2015) el indicador esencial para evaluar el performance de modelos neuronal en *image segmentation* se concibe por *dice similarity coefficient* (SDC). En efecto, este valor mide la proximidad que existe entre la máscara de segmentación predicha y la etiquetada manualmente. En general, se desean valores elevados en todas las métricas expuestas para representar un modelo confiable.

2.1.3.3 Inferencia semántica

Este apartado refiere a la fusión de las funciones de la visión computacional con nuevas herramientas para elaborar aplicaciones más complejas. En efecto, Gollapudi (2019) reconoce que la mayor parte de las funciones de la visión computacional se elaboran en base a la tarea *object detection*. Por ejemplo, según Angah & Chen (2020) el proceso de *object tracking* inicia por aplicar *object detection* para localizar al objeto (operario, postura del operario, maquinaria, etc.) y, luego, se realiza el registro continuo en las imágenes para extraer información relevante como velocidades, trayectoria y flujo de movimiento. Asimismo, Luo et al. (2019) indican que esta herramienta permite analizar acciones simples (transportar material, doblado de acero, vibrado de concreto, etc.), debido a que es factible entrenar el clasificador a identificar caracteres espaciales (representación del objeto) y temporales (movimiento del objeto) en secuencias de imágenes mediante técnicas en *optical flow*.

De igual manera, Luo et al. (2018c) analizaron actividades (encofrar, armado del acero de refuerzo, vaciado de concreto, movimiento de tierras, etc.) en base al reconocimiento de

acciones simples. En esencia, los autores incorporaron al sistema modelos estocástico (gráficas probabilísticas), como *hidden markov models* y *conditional random fields*, para correlacionar las posibles secuencias de acciones y predecir las actividades. Otra alternativa empleada comúnmente en la literatura es el enfoque *description based method*, que consiste en identificar actividades mediante un suceso de eventos (acciones) que cumplan relaciones temporales y espaciales previamente definidas (Bügler et al., 2017). En efecto, esta práctica es empleada con mayor frecuencia en las tareas que realizan los equipos de movimiento de tierra. Sin embargo, se limita en las partidas que operan netamente operarios, debido a la diversidad de acciones improvisadas que puede presentar la mano de obra.

Por otro lado, la Figura 9 representa los resultados de detección de objetos en el reconocimiento de extremidades. De esta manera, es posible materializar la información como modelos esqueléticos (Xu et al., 2020). En efecto, este mecanismo también posibilita analizar actividades, por medio de las acciones individuales que se registran en el esqueleto; facilita la identificación de posturas anormales de los operarios que provocan daños temporales o permanentes; entre otras aplicaciones.

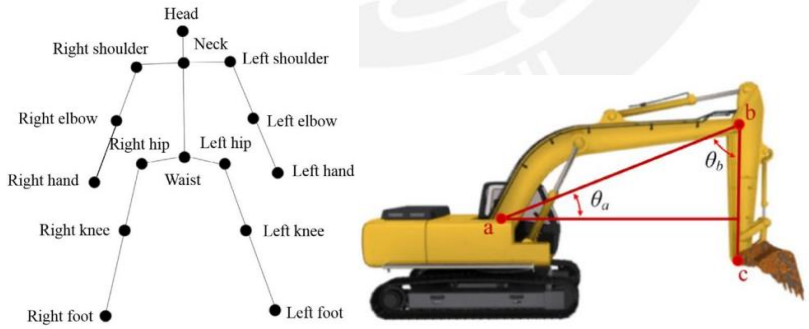


Figura 9. *Modelo esquelético de personas y equipos pesados.*

Nota. Tomado de "Computer Vision Techniques in Construction: A Critical Review", por Xu et al., 2020.

2.1.3.4 Aplicación

El módulo final del proyecto en visión computacional refiere a la habilidad del ingeniero en transformar las relaciones semánticas de la visión computacional *computer*

vision en valor agregado para el proyecto de construcción. Por lo general, se incluyen capas lógicas al sistema para elaborar instrucciones específicas. Por ejemplo, Luo et al. (2019) emplearon los resultados de la tarea *action recognition* para clasificar las acciones en trabajo productivo, contributorio y no contributorio. De esta manera, se contabilizaron los estados productivos de cada obrero en un intervalo de 2 minutos para graficar una regresión lineal y monitorear la fatiga de la mano de obra. Asimismo, la tarea *object recognition* permite identificar escenarios que signifiquen un riesgo para el proyecto (fisuras, uso inapropiado de equipos de protección, posturas inadecuadas, etc.). Por lo tanto, la incorporación de una capa lógica que permita emitir alertas en tiempo real a los supervisores y prevenir impactos negativos al proyecto es un ejemplo de valor agregado (W. Fang et al., 2020). Además, una aplicación similar resulta válido de incorporarse en los resultados de *object tracking* para identificar en tiempo real velocidades excedidas por lo equipos pesados en el área de trabajo, analizar distancias críticas entre operarios y maquinaria, entre otros escenarios (Seo et al., 2015).

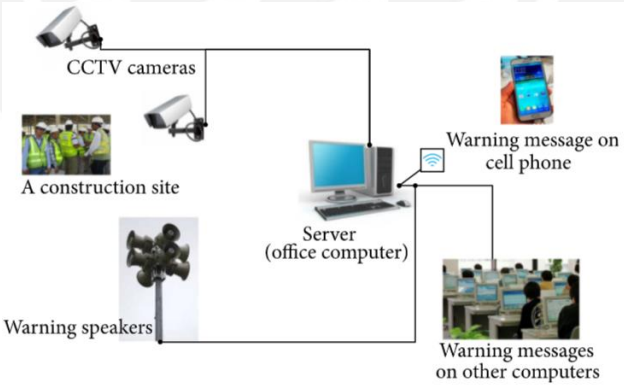


Figura 10. *Esquema de un sistema de alertas incorporado en modelos de visión computacional.*

Nota. Tomado de "Hard-Hat Detection for Construction Safety Visualisation", por Shrestha et al., 2015.

2.2 Aplicaciones de la visión computacional en la construcción civil

Como se comentó anteriormente, la visión computacional provee la capacidad de analizar información capturada en videos e imágenes. En el contexto del sector

construcción, la visión computacional permite emplear esta data para analizar la productividad laboral, controlar riesgos, monitorear la calidad de las estructuras, entre otras aplicaciones.

2.2.1 Productividad laboral

Según Konstantinou (2018), los ratios de productividad son indispensables para planificar adecuadamente la programación del proyecto y reducir los costos del mismo. En efecto, Konstantinou (2018) reconoce que las técnicas actuales en *computer vision* permiten obtener las métricas mencionadas por medio del desarrollo de modelos en *action recognition* y, de esta manera, analizar las actividades de la mano de obra y equipos de movimiento de tierra. Por ejemplo, Luo et al. (2018) desarrollaron un método en base a tres variantes de CNN para clasificar las acciones en tres estados: trabajo productivo, trabajo contributorio y trabajo no contributorio. La metodología contempla el uso de una tecnología en *single object tracking,* denominada MDNet, para la generación de *bounding boxes* consecutivas; un modelo en *optical flow* (red neuronal FlowNet 2.0), para realizar el proceso de *feature extraction* en videos y entregar caracteres espaciales (formato RGB que identifica al obrero), como temporales (formato en escala de grises que registra movimiento); y el reconocimiento de acciones por medio del prototipo TSN, que emplea los resultados anteriores. Además, el desempeño promedio del sistema resulta de 80.5 %, en *accuracy,* al reconocer las acciones distribuidas en 6 que corresponden a partidas de encofrado, 7 para el armado de acero y 3 acciones en común. Asimismo, el recurso de entrada al sistema consistió en 76 videoclips de duraciones variadas entre 1 a 15 minutos. Respecto a las desventajas de la investigación, los autores obtuvieron una métrica de *accuracy* por debajo del 90% debido a dos principales factores: la variación de intraclase de los obreros (variedad de gestos para una acción y acciones que presentan gestos en común) y el empleo del modelo en *object tracking* que no registrar movimientos de

múltiples unidades en simultáneo. Por lo contrario, el valor agregado a los proyectos de construcción consiste en realizar una carta de balance considerando toda la muestra de obreros, en vez de una minoría como sucede en la práctica convencional.

Del mismo modo, el equipo de investigación anterior fusionó la metodología mencionada, TSN en el reconocimiento de acciones, con gráficas probabilísticas (una variante de *hidden markov model* denominada HDP-HMM) para identificar estados más complejos en el vaciado del concreto (Luo et al., 2019). Una característica importante del modelo TSN es que permite operar en videos que capturan información a larga distancia, por lo que, se les permitió instalar una cámara de video por debajo de la cabina de la grúa torre (a 20 m. de altura sobre el nivel de trabajo) mientras que en el ensayo anterior la cámara se instaló a una altura de 15 m en los andamios. Asimismo, la fuente de datos del sistema consistió de 540 video clips de 3 segundos de duración. Respecto al desempeño de la metodología, se exhibe un valor de 84% en *average accuracy* al reconocer 7 acciones características de la partida de vaciado de concreto, mientras que el gráfico probabilístico formuló 13 actividades, producto de las interacciones entre las acciones de cada obrero (compactar concreto con vibrador, nivelar la superficie del concreto con un rastrillo, etc). Respecto a las limitaciones, los autores recomiendan desarrollar tecnologías en *object tracking* que registren múltiples objetos en simultáneo de manera precisa y generar más base de datos que comprendan acciones de obreros en otras partidas de construcción.

En referencia a las tecnologías *object tracking*, Angah & Chen (2020) elaboran una técnica de *multiple object tracking* (MOT) en base a *mask* R-CNN para monitorear múltiples trayectorias de operarios en simultáneo, alcanzando una exactitud MOTA de 81.8%. Asimismo, Gard et al. (2018) aportaron al estudio de la productividad laboral con el desarrollo de un sistema CNN en *human tracking* que predice las posturas de los operarios mediante planos antropométricos. En efecto, los autores concibieron esta idea

para contrarrestar el problema de trabajar con una sola cámara: la información de velocidades y deformaciones de las articulaciones se distorsionan respecto a la realidad, debido al efecto de capturar objetos cerca o alejados de la cámara. En esencia, se comenta que la metodología sucede en dos etapas: se detectan modelos esqueléticos, junto a las etiquetas de las articulaciones, y luego se transforman las articulaciones a sus correspondientes planos antropométricos (planos a nivel de la cabeza, hombros, cintura, rodillas y pies) para monitorear las trayectorias mediante la técnica *Kalman filter*. Las limitaciones residen en la pérdida de información del objeto debido a las oclusiones y las alturas de los planos antropométricos diseñadas para capturar articulaciones en base a la estatura promedio de un residente de América del Norte.

De igual manera, Luo et al. (2018) propusieron un método que combina tres CNN's (VGG-16, CNN convencional en escala de grises y *optical Flow* CNN) para el reconocimiento de tres acciones relacionadas al doblado de acero: transportar material, desplazarse y doblado de acero. El recurso de entrada al sistema consistió en 654 video clips de 2.2 segundos de duración cada uno. Además, la precisión promedio del modelo se limita a 85% principalmente por el impacto de las oclusiones en el monitoreo de los obreros y la poca cantidad de clips para entrenar los modelos. Un tercer problema reside en no incorporar reglas de decisión que definan el comienzo y fin de la partida en estudio. Un enfoque diferente se presenta en el trabajo de Luo et al. (2018) al no considerar modelos en *object tracking* y trabajar netamente con algoritmos en *object recognition*, Faster R-CNN y ResNet-50. Específicamente, los autores emplearon solo la función fundamental de *computer vision* para identificar 22 clases de objetos y clasificarlos en 4 grupos: operarios, materiales, herramientas y vehículos pesados. El proceso clave para identificar acciones consistió en la propuesta de *relevance networks,* que consiste en generar patrones de actividades compuestas por relaciones semánticas, representando la probabilidad de los

elementos en coexistir en una misma actividad, y correlaciones espaciales, que indican el trabajo colaborativo de las entidades según las distancias entre sus *bounding boxes.* Asimismo, los recursos de entrada al sistema comprendieron 7790 imágenes, destinando un tiempo estimado de etiquetado de objetos en 200 horas de trabajo. Respecto al desempeño del modelo, se exhibe una precisión de 62.4% y *recall* de 87.3% al reconocer 17 acciones repartidas entre la partida de encofrado (columnas, vigas, losas y escaleras), vaciado del concreto e instalación del acero de refuerzo. En esencia, un factor negativo corresponde al uso de distancias 2D, en lugar de distancias 3D, para formular las correlaciones espaciales entre las *bounding boxes* de las unidades de estudio. En efecto, los autores asumieron que las imágenes se encuentran orientadas en ángulos verticales para formular las relaciones espaciales. Sin embargo, existen imágenes desenfocadas con ángulos diferentes a la convención. Frente a esta problemática, los investigadores recomiendan que los modelos en *object recognition* se trabajen con por lo menos 3 cámaras para realizar una triangulación visual y obtener distancias tridimensionales entre las unidades de estudio. Por último, la metodología está limitada a trabajar con imágenes y no en videos, pues no se diseñó para captura la información temporal de los fotogramas.

Asimismo, Liu et al., (2017) estudiaron las actividades de la mano de obra, relacionadas al corte del acero de refuerzo y tareas de mampostería, mediante una red CNN (*Faster* R-CNN) que estima las posturas humanas. De esta manera, los autores buscan promover el análisis de posturas ergonómicos y productividad laboral. La limitación de la metodología se representa en menores valores de *accuracy* al identificar la cintura del operario, al sujetar herramientas de trabajo, y sus cabezas (debido a la oclusión que genera el casco de seguridad).

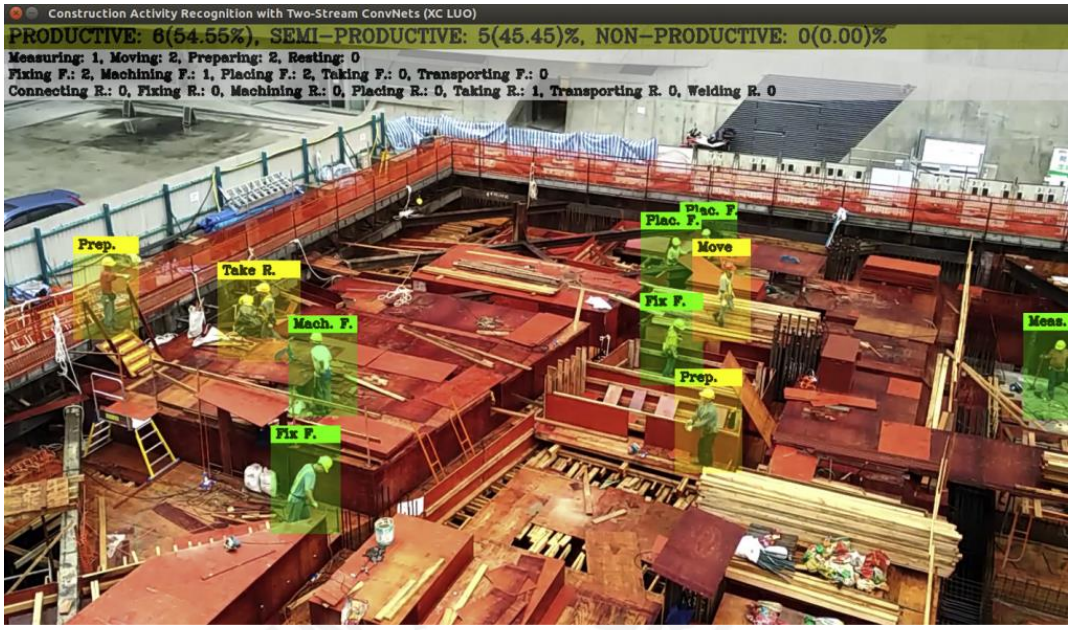


Figura 11. *Monitoreo de la productividad laboral de la mano de obra: clasificación entre actividades productivas, contributorias y no contributorias.*

Nota. Tomado de "Convolutional Neural Networks: Computer Vision-Based Workforce Activity Assessment in Construction", por Luo et al., 2018.

Adicionalmente al estudio de la productividad de la mano de obra, existen investigaciones que se enfocan a reconocer las actividades de los equipos de movimiento de tierras con el objetivo de minimizar sus tiempos no productivos, reducir las emisiones de gases de efecto invernadero, evitar colisiones con los operarios, uso económico del combustible, entre otras aplicaciones. En efecto, Rashid & Louis (2019) evaluaron una red RNN, denominada *long short term memory*, al monitorear de 9 y 10 actividades de excavadoras y cargadores frontales, respectivamente. Además, los valores de *accuracy* del modelo corresponden a 59.7%, sin técnicas en *data augmentation*, y 96.7%, empleando técnicas de rotación, escalamiento, *time-warping* y *jittering*. De igual manera, Slaton et al. (2020) diseñaron un modelo híbrido que comprende una red CNN convencional y RNN, denominada *Long Short Term Memory*, para predecir actividades de equipos pesados (rodillos compactadores y excavadoras) en base a acelerómetros. Un estudio similar es elaborado por Kim & Chi (2019) al proponer una metodología que identifica excavadoras, por medio del algoritmo *faster R-CNN*; monitorea sus trayectorias, en base a la técnica

Tracking-Learning-Detection; y reconoce sus actividades, a través de un modelo que combina una red CNN convencional y *Double-layer Long Short Term Memory.* Además, los recursos de entrada en el sistema consistieron en 70000 imágenes y 121 minutos de videos, y generaron resultados de *accuracy* mayores a 90% en las tres funcionalidades*: object detection, tracking* y *action recognition*.

Del mismo modo, Roberts & Golparvar-Fard (2019) desarrollaron el modelo *Resnet-101* para identificar excavadoras interactuando con varios volquetes*;* monitorearon sus trayectorias mediante la red *Tubelets* CNN; y modelaron las transiciones de sus actividades, en base al método estocástico *Hidden Markov Models.* Asimismo, Fang et al. (2018) presentaron un prototipo, denominado IFaster R-CNN, que automatiza la detección de equipos de construcción y operarios en tiempo real para promover las aplicaciones de análisis de productividad y colisiones entre las entidades de estudio. De igual manera, Chen et al. (2020) ejecutaron las aplicaciones de productividad laboral a través del monitoreo de las actividades de excavadoras (excavación, carga, oscilamiento, etc.). En esencia, los autores emplearon el modelo *Faster* R-CNN, para identificar los objetos desde videos; la tecnología *Deep* SORT para monitorear múltiples entidades en simultáneo; algoritmos que representan estados inactivos o paralizados; y la red 3D ResNet, para identificar las actividades de la excavadora. Además, el set de entrenamiento del sistema se compuso de 351 video clips con 600 segundos de duración en promedio, produciendo un valor de 92.5% y 87.6% en *accuracy* al identificar las excavadoras y reconocer sus actividades, respectivamente. En referencia a las limitaciones de la investigación, los autores comentan que el desempeño del modelo es afectado por la superposición de *bounding boxes*, cuando las excavadoras se ubican próximas unas de las otras, y el set de entrenamiento contiene objetos capturados desde posiciones elevadas, por lo que, el sistema presentaría dificultades al evaluarse en una base de datos capturadas a nivel del terreno.

Figura 12. *Monitoreo de la productividad en equipos de movimiento de tierra.*

Nota. Tomado de "End-to-end vision-based detection, tracking and activity analysis of earthmoving equipment filmed at ground level", por Roberts & Golparvar-Fard, 2019.

Asimismo, las investigaciones mencionadas hasta el momento han sido desarrolladas en escenarios que contienen específicamente las actividades en estudio, ignorando la coexistencia de actividades irrelevantes, diferentes equipos de trabajo y maquinarias en campo. Frente a esta problemática, Cai et al. (2019) propusieron una metodología que identifica cuadrillas de trabajo correspondiente a una actividad específica, en entornos de trabajo que comprendan múltiples tareas en simultáneo. En efecto, los autores aplicaron el modelo *Long Short Term Memory*, junto a señales de posición (distancias entre obreros) y atención (direcciones relativas de la cabeza y mentón, orientación del cuerpo, etc.), para diferenciar actividades de pavimentación, excavación, supervisión, entre otras. Además, los recursos del algoritmo consistieron en 10000 videoclips, de 5 segundos de duración cada uno, repartidos entre 80% y 20% para propósitos de entrenamiento y evaluación, respectivamente. En esencia, el desempeño del modelo resulta en 85% en *accuracy*, al identificar cuadrillas y sus respectivas actividades, sin aplicar señales de atención. Por otro lado, el performance del sistema incrementa a 96.3% al incorporar señales de atención en el análisis de actividades. Entre las limitaciones del proyecto, los autores afirman que se analizaron actividades sencillas compuestas por pocos operarios y una máquina. Asimismo, el modelo considera importante las orientaciones del cuerpo de los obreros, puesto que si

no se direccionan entre la cuadrilla el prototipo presentará complicaciones en distinguir los grupos de trabajo.

Por otro lado, Yu et al. (2019) reconocen que la productividad laboral es afectada por el rendimiento físico de los operarios al trabajar durante horas prolongadas, bajo condiciones de temperatura elevadas, diferente resistencia y musculatura entre operarios, entre otros factores. En efecto, los autores propusieron un modelo en base a CNN, denominado *Stacked Hourglass Networks*, para analizar la fatiga física de los obreros en las actividades de asentado de ladrillos, transporte de materiales e instalación de acero de refuerzo. En esencia, el prototipo, entrenado en la data pública de posturas MPII, es aplicado para identificar articulaciones específicas de los operarios (cuellos, hombros, codos, rodillas, tobillos y cintura) y, luego, inferir sus posturas tridimensionales mediante modelos esqueléticos. Además, los científicos emplearon dispositivos inerciales IMU, sujetados en el cuerpo de los obreros, para comparar las posturas reales con los resultados del modelo neuronal. En consecuencia, las desviaciones obtenidas prevalecieron con valores entre 1.32 y 4.93 cm de error en las actividades estudiadas. Por lo tanto, el sistema no estima las posturas eficientemente. Frente a esta problemática, los autores recomiendan incrementar las imágenes de entrenamiento considerando el efecto de las oclusiones. Asimismo, la información cinemática, obtenidas con las dos herramientas mencionadas, se utilizaron para calcular el torque de las respectivas articulaciones y, mediante un índice de fatiga, se determinaron las capacidades disponibles de cada articulación de la mano de obra. Entre otras limitaciones de la metodología, se contempla el requisito de realizar mediciones de los pesos de los materiales y herramientas que manipulará el operario antes de aplicar el modelo (importante para determinar el torque de las articulaciones). Además, los autores asumieron tiempos de descanso específicos, en referencia a las acciones que no ejercen torques en articulaciones, para determinar las capacidades disponibles. En efecto, los

científicos recomiendan identificar los tiempos de descanso articular mediante el análisis de posturas en modelos esqueléticos. En la Tabla 4 se presenta un breve resumen sobre los artículos revisados.

En referencia a los modelos tradicionales en *machine learning*, el estudio de Gong & Caldas (2010) representa la primera propuesta en analizar actividades (relacionadas al vaciado de concreto) mediante el enfoque *description-based method.* En esencia, la metodología consiste en especificar zonas de interés en el video que significan subtareas y un flujo de trabajo productivo se identifica como el cumplimiento de las entidades de construcción a seguir un orden secuencial en las regiones predefinidas. Sin embargo, la limitación de la metodología reside en conocer con anticipación el flujo de trabajo de la partida y emplear horas hombres extras para localizar las zonas de estudio en el video. De igual manera, Yang et al. (2016) estudiaron las acciones del personal de obra mediante el modelo *dense trajectories,* que se compone de *Motion Boundary Histogram,* para efectuar la tarea de *feature extraction,* y el clasificador *support vector machines.* En esencia, el performance del modelo alcanza un valor de 59% en *average accuracy* al ser evaluado en 1176 video clips, de 6.8 segundos de duración en promedio, que contienen 1 acción por video. Por tanto, la aplicabilidad del modelo tradicional en *machine learning* está limitado a no identificar múltiples acciones en simultáneo. Asimismo, Golparvar (2013) empleó *support vector machines* para clasificar las acciones que efectúan las excavadoras con una métrica de *average accuracy* igual a 86.33%.

Tabla 4. *Aplicaciones de modelos en deep learning para el análisis de la productividad laboral en el sector construcción.*

Referencia	Aplicaciones	Tecnologías		Limitaciones
Luo et al. (2018)	Reconocimiento de actividades de construcción y carta de balance a partir de los estados TP, TC y TNC	CNN	MDNet FlowNet 2.0 TSN	Variación de la intraclase de los operarios al realizar sus acciones laborales.
Luo et al. (2019)		CNN	TSN	Empleo de tecnología *single object tracking* en lugar de monitoreo múltiple de objetos (MOT).
		Otras	HDP-HMM	

Angah & Chen (2020)	Monitorean múltiples operarios (MOT)	CNN	Mask R-CNN ResNet-18	
Luo et al. (2018)	Reconocimiento de actividades de construcción relacionadas a la instalación del acero de refuerzo	CNN	VGG-16	No se desarrollan reglas de decisión para identificar inicio y fin de una actividad. Los videos solo capturan una única partida de construcción.
Luo et al. (2018)	Reconocimiento de actividades de construcción relacionadas al encofrado, vaciado e instalación de acero de refuerzo	CNN	Faster R-CNN ResNet-50	La metodología trabaja con imágenes y no procesa videos. Sistema se basa solo en *object recognition* por lo que depende de información espacial. No se emplearon imágenes 3D que resultan crítico en el desempeño del modelo.
(M. Liu et al., 2017)	Estudian las actividades de la mano de obra en base a las posturas esqueléticas 2D	CNN	Faster R-CNN *Stacked Hourglass Networks*	Dificultades del modelo al identificar caderas y cabezas debido a las oclusiones que generan las herramientas de trabajo y cascos de seguridad.
Rashid & Louis (2019)	Monitorean las actividades de excavadoras y cargadores frontales	RNN Otras	LSTM Sensor inercial IMU	Se evalúa el modelo en dos tipos de maquinarias. No se emplea la información obtenida para el análisis de la productividad, uso económico del combustible, entre otros.
Slaton et al. (2020)	Monitorean actividades de excavadoras y rodillos de compactación	RNN Otras	LSTM Acelerómetros	Resta evaluar la capacidad del modelo en generalizar actividades de otros tipos de maquinarias.
Kim & Chi (2019)	Se monitorean las actividades de las excavadoras	CNN RNN MOT	Faster R-CNN DLSTM TLD	Excesivas imágenes de entrenamiento para obtener valores de *accuracy* mayor al 90%. Pérdida de información 3D al capturar la maquinaria como movimientos verticales respecto a la cámara. No se emplea la información obtenida para el análisis de la productividad
Roberts & Golparvar-Fard (2019)	Identifican, monitorean y analizan las actividades de excavadoras	CNN Otras	Resnet-101 *Tubelets CNN* *Hidden Markov Model*	Se identifican actividades individuales de los equipos pesados, mas no las interacciones entre ellos. El set de entrenamiento comprende información capturada a nivel del terreno (mas no a otras alturas) y una excavadora por video.
Fang et al. (2018)	Automatizan la detección de equipos de construcción y operarios para promover análisis de productividad y colisiones	CNN	IFaster R-CNN	Se requiere capturar más objetos bajo efectos de oclusión y escalas. Áreas de estudio limitadas debido a la localización de cámaras estáticas.

Chen et al. (2020)	Identifican, monitorean y analizan las actividades de las excavadoras para determinar su productividad laboral	CNN MOT	Faster R-CNN 3D ResNet Deep SORT	Confusión entre las actividades de los equipos al monitorear excavadoras próximas unas de las otras. La data en el set de entrenamiento es capturada desde elevaciones elevadas mas no a nivel del terreno. Se requieren videos con mayor tiempo de duración para analizar ciclos de productividad
Cai et al. (2019)	Reconocen cuadrillas específicas correspondientes a actividades de pavimentación y excavación en entornos colaborativos	RNN	LSTM	Error en reconocer grupos de trabajo cuando los operarios se retiran e ingresan a la zona de trabajo. Se analizan actividades sencillas compuestas por pocos operarios y una máquina. El modelo presenta complicaciones en distinguir grupos de trabajo si las orientaciones de los cuerpos de los obreros no se direccionan entre la cuadrilla.
Gard et al., (2018)	Monitorean las articulaciones de operarios en base a planos antropométricos	CNN Otras	VGG-16 *Kalman filter*	Las alturas de los planos antropométricos corresponden a medidas de una persona americana. Pérdida de información debido a las oclusiones.
Yu et al. (2019)	Monitorean la fatiga física de los operarios en partidas de asentado de ladrillo, tarrajeo y transporte de materiales	CNN Otras	*Stacked Hourglass Networks* Sensor inercial IMU	El prototipo determina posturas con un error de hasta 5 cm. Requisito de determinar el peso de las herramientas que manipulará el operario antes de aplicar la metodología. Se asume manualmente tiempos de descansos específicos para aligerar la tensión articular en el modelo.

2.2.2 Seguridad y control de riesgos

Las aplicaciones de la visión computacional para el control de riesgos y salud en el trabajo se clasifican en 2 grupos de acuerdo a la convención de Akinosho et al. (2020): metodologías que identifican comportamientos inadecuados del personal de obra y condiciones de trabajo inapropiadas. Respecto a la primera categoría, un caso práctico consiste en el desarrollo de modelos en *object detection* para verificar el uso de equipos de protección personal y colectiva. En efecto, Fang et al. (2018) identificaron operarios sin el uso de cascos de protección mediante la red convolucional *Faster* R-CNN. En esencia, el

sistema exhibe una precisión y *recall* de 95.7% y 94.9%, respectivamente, después de entrenarse en 81 000 imágenes. Sin embargo, el modelo no permite identificar la identidad del operario que infringe la norma de seguridad. Asimismo, actualmente *Faster* R-CNN es reemplazada por *mask* R-CNN y otros prototipos en cuanto a performance. Además, el algoritmo no reconoce individualmente a los cascos de seguridad, por lo que, se limita a no distinguir los colores de los cascos e identificar roles del personal. En esencia, Wu et al. (2019) desarrollan un modelo en base a *single shot multibox detector* (SSD) que permite identificar individualmente cascos y obreros, así como el color del equipo de seguridad. Además, los autores contribuyeron con la recolección y distribución pública de 3174 imágenes, que produjeron una precisión mAP igual a 83.89% en su análisis. En efecto, los autores sugieren aplicar estrategias de *semantic segmentation* para mejorar el performance de modelos enfocados en la misma tarea. Asimismo, los resultados de su investigación demostraron que las dimensiones de las imágenes son un factor importante que se debe definir para obtener desempeños exitosos en detección en tiempo real mediante el modelo SSD.

De igual manera, Xie et al. (2018) evaluaron 4 modelos en *object recognition* (R-FCN, SSD, FPN y YOLO) para identificar cascos de protección en imágenes de construcción civil y se corroboró que YOLO representa el mejor desempeño con un valor *mAP* equivalente a 53.8%. Posteriormente, los autores elaboraron una capa lógica para verificar el uso del EPP, que consistió en calcular las áreas de intersección (IOU) entre la *bounding box* de los cascos y operarios para confirmar el equipamiento del elemento de seguridad. Respecto a las técnicas tradicionales en *machine learning*, Mneymneh et al. (2017) evaluaron los algoritmos SURF, BRISK y FAST junto al clasificador *cascade,* obteniendo una precisión igual a 86.79% en una base de datos de 239 imágenes. Sin embargo, la velocidad de procesado resulta de 2 *frame* por segundo, por lo que, no resulta factible

aplicar el modelo en tiempo real. De igual manera, Wu et al. (2019) fusionaron *color-based hybrid descriptor* y *hierarchical support vector machines* para entrenar el modelo en 6631 imágenes. Las métricas del proyecto en *accuracy* resultaron de 90.3% y probablemente se asocia a una base de datos sin escenarios complejos.

Por otro lado, Nath et al. (2020) proponen tres diferentes enfoques en base a CNNs para reconocer el uso de más de un EPP: prendas reflectivas y cascos de protección. El primer método emplea YOLO-v3 (previamente entrenado en una base de datos ajena al sector construcción), para identificar los objetos, y árboles de decisión, que ejecutan reglas lógicas para inferir el uso de cada EPP por el usuario; el segundo método aplica YOLO-v3 directamente para identificar y verificar el uso del objeto, obteniendo el menor performance; y el tercer enfoque es similar al primero con la diferencia de emplear *transfer learning* (VGG-16, ResNet-50 y Xception) en lugar del árbol de decisión. En esencia, el mejor desempeño se obtuvo en la tercera configuración alcanzando un valor mAP equivalente a 85.6%. Asimismo, la característica importante de este trabajo es que se permite verificar el uso de varios EPP, por lo que, los autores incitan a extender la aplicación del sistema con otras unidades de seguridad.

Asimismo, Fang et al. (2018) se enfocaron en estudiar equipos de protección personal que evitan accidentes letales en trabajos en altura. Por lo tanto, alternativamente a las sesiones de capacitación y normas que penalizan la ignorancia del uso del arnés de seguridad, los autores propusieron 2 modelos CNN para identificar el uso de este accesorio. En efecto, el algoritmo Faster R-CNN es destinado a identificar individualmente a los obreros, mientras que una variante de la arquitectura CNN convencional se emplea para vincular el equipamiento de los arneses de seguridad con el personal de obra. Las métricas de precisión para cada etapa residen en 99% y 80%, respectivamente, al ser evaluadas en 700 imágenes. Efectivamente, la limitación del modelo reside en emplear una base de datos

insuficiente para contrarrestar el efecto de *background clutter* (colores similares entre el arnés de seguridad y la vestimenta del operario) y las oclusiones del EPP con el chaleco reflectivo.

De igual manera, Fang et al. (2018) estudiaron la detección de arneses de seguridad, junto a la línea de vida, y cascos de protección para controlar la seguridad en partidas que requieran de emplear plataformas suspendidas. En efecto, la metodología corresponde al empleo del modelo *single shot detector* (SSD) para identificar el uso de los EPP solo cuando el personal se localiza en el área suspendida y así evitar emitir alertas falsas cuando no existe riesgo de caída. Asimismo, los autores incorporaron el algoritmo *simple online real time tracking* (SORT) para monitorear los EPP en los videos. Además, el recurso de entrada consistió de 38000 imágenes de entrenamiento y 3748 video clips para el set de evaluación. Respecto al desempeño del modelo, se corroboró una precisión de 96% al ser evaluado en un set sin oclusiones, mientras que la precisión disminuyó a 50% al evaluarse en video clips que contienen los EPP casi totalmente ocluidos. Un efecto negativo de la metodología es que requiere localizar una cámara por habitación debido a que representan la zona de ingreso a la plataforma suspendida. Una segunda limitación se presenta en el proceso de identificación del arnés de seguridad junto a la línea de vida. En efecto, el etiquetado con *bounding boxes* implica cubrir el espacio del obrero y de la línea de vida con formas rectangulares, por lo que, se capturan áreas no útiles y caracteres falsos que generan ruido en la tarea de *feature extraction*. Asimismo, hasta el momento, los esfuerzos de aplicar *object detection* para verificar el uso de EPP o EPC no se han enfocado en identificar la identidad del operario que infringe la norma de seguridad.

Figura 13. *Ejemplos de resultados de detección de equipos de protección personal*.

Nota. Tomado de "Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset", por Wu et al., 2019*; "Detecting non-hardhat-use by a deep learning method from far-field surveillance videos", por* Fang et al., 2018.

Asimismo, Kolar et al. (2018) propusieron un diferente enfoque para prevenir las caídas desde alturas. En efecto, los autores desarrollaron una red CNN, denominada *VGG-16*, para verificar la instalación de barandas de seguridad en niveles elevados, plataformas y andamios. Además, el sistema se entrenó con 2000 imágenes propias de campo y se evaluó en 2000 fotos capturadas desde internet. Respecto al desempeño del modelo, el sistema obtuvo un valor de 96.5% en *accuracy*. Entre las limitaciones del proyecto, los autores mencionan que la red neuronal presenta confusiones al distinguir entre las barandas de seguridad y las barras de refuerzo, así como algunos elementos del encofrado, debido a la escasez de falsos negativos comprendidos en el set de entrenamiento. Además, la data recolectada comprende un solo diseño de baranda de seguridad mientras que en el sector comercial existen diferentes configuraciones y materiales. Por último, los autores no consideraron barandas ocluidas en el *training set.* De igual manera, Siddula et al. (2016) propusieron una red CNN no supervisada para reconocer cubiertas, operarios, barandas y arneses de seguridad. En efecto, el set de entrenamiento se compuso de 1200 imágenes, capturando 1 solo objeto de interés en cada una. Por lo tanto, no se consideran efectos de

oclusión en el proyecto y el desempeño del modelo alcanza un valor de 96.67% en *accuracy.*

Un segundo comportamiento inadecuado que se busca monitorear en obra es verificar que la mano de obra se encuentre ejecutando tareas relacionadas a su área de especialización. Esto resulta importante debido a que aproximadamente más del 40% de accidentes ocurren cuando los operarios intentan ejecutar trabajos de alto riesgo (trabajo en alturas, manejar circuitos eléctricos, manipular maquinaria pesada, entre otros) sin disponer del entrenamiento y experiencia suficiente (Umeokafor et al., 2014). En efecto, Fang et al. (2018) propone una metodología basada en tres tecnologías de visión computacional: la primera herramienta corresponde a Faster R-CNN, que efectúa la tarea de *object detection* para reconocer obreros, equipos y materiales a partir de videos; El segundo componente corresponde SORT, que refiere a la función *object tracking* para localizar a los objetos a lo largo del video; y la tercera tecnología refiere a MTCNN, desarrollada para ejecutar la tarea de *face recognition,* es decir, reconocer la identidad de los operarios. Asimismo, el reconocimiento de actividades se efectúa por medio de correlacionar espacial y temporalmente la información de los equipos, materiales y obreros. De esta manera, con la información obtenida los autores desarrollaron una capa lógica en el sistema para comparar las actividades en progreso con las establecidas en el servidor. En referencia a los recursos del proyecto, se recolectaron 8000 imágenes para entrenar el modelo Faster R-CNN, mientras que el algoritmo en *face recognition* se preparó con bases de datos públicas. Asimismo, el set de evaluación consistió en 60 video clips de 120 segundos de duración, mientras que para reconocer la identidad del personal de obra se emplearon 10 fotos del rostro de cada uno. Respecto al performance del sistema, se obtuvo un valor de 83.20% de *average precision* y 83.4% de *average recall*. En esencia, los autores recomiendan emplear más de una sola cámara de estudio para capturar diferentes ángulos que permitan localizar

el rostro de la cuadrilla en cada videoclip. Asimismo, el sistema no dispone de un mecanismo que notifique en tiempo real el incumplimiento de ejecutar trabajos ajenos a la especialidad del obrero.

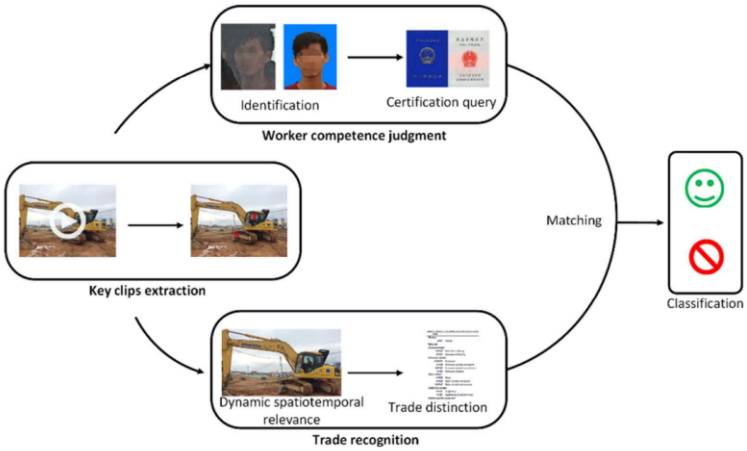


Figura 14. *Ejemplo de flujo de trabajo para inspeccionar trabajos autorizados por personal calificado.*

Nota. Tomado de "Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset", por Wu et al., 2019*;*

Una tercera actitud inapropiada estudiada en la literatura refiere a los desórdenes musculoesqueléticos que experimenta la mano de obra al sobre esforzarse físicamente durante horas prolongadas (cargando pesos elevados, trabajar con posturas anormales, etc.). Por lo general, los proyectos pertenecientes a esta categoría emplean dispositivos electrónicos (WIMUs), para obtener información cinemática de la persona, y un modelo en *machine learning*, para clasificar las actividades en estudio (Fang et al., 2019). En efecto, Yang et al. (2020) integraron una variante de *recurrent neural networks*, denominada *long short-term memory* (Bi-LSTM), y sensores inerciales, atados en el tobillo, para reconocer cuatro niveles de esfuerzo físico en el transporte de ladrillos: 0, 2, 4 y hasta 6 ladrillos de concreto transportados por un solo operario. En efecto, el sensor inercial WIMU se empleó para recolectar información sobre la aceleración y velocidad angular, a través del contacto de los pies con el terreno. Asimismo, los autores aplicaron el modelo neuronal para

clasificar los niveles de carga a partir de las respuestas del dispositivo inercial y videoclips de 1.2 segundos que comprenden el transporte del material. Específicamente, se dispuso de 2196 instancias divididas en 80% para entrenamiento y 20% para evaluación del algoritmo. Respecto al desempeño del modelo, se obtuvo un valor de 98.6% en *accuracy* al identificar cualquier escenario de carga de ladrillos y disminuyó a 74.60% al clasificar los 4 tipos de esfuerzo físico. En cuanto a las limitaciones del método, se conoce que las partidas de construcción no solo comprenden tareas de transporte de materiales, por lo que el sistema no permite evaluar las condiciones físicas en los demás escenarios de construcción. Además, el sistema no se orientó a identificar las posturas anormales ni se consideró la influencia de la fatiga en los operarios para elaborar un modelo que prevenga en su totalidad daños musculoesqueléticos.

Adicionalmente, Zhang et al. (2018) proponen una arquitectura CNN denominada *convolutional pose machine* que estima cuerpos esqueléticos 3D para identificar posturas ergonómicas. En efecto, los autores emplearon dispositivos sensoriales de inercia (IMU) para capturar los ángulos de flexión entre las extremidades del cuerpo para representarlos en el modelo esquelético. Respecto al desempeño de la metodología, se reconocieron los movimientos del brazo (debajo y por encima del hombro), piernas (sentadillas, arrodillado y firme) y espalda (rotado, inclinado y firme) con valores de *accuracy* entre 93.9% a 94.6%. Asimismo, se comenta que el efecto de oclusión es el principal obstáculo de la metodología debido a que si no se perciben las extremidades del operario no es posible distinguir determinados movimientos del cuerpo. Por ejemplo, en el ensayo de los autores los operarios que trabajaban de espalda hacia la cámara no se les visualizaban las piernas.

Figura 15. *Ilustraciones de estimación de modelos esqueléticos 3D para identificar posturas inseguras.*

Nota. Tomado de "Ergonomic posture recognition using 3D view-invariant features from a single ordinary camera", por Zhang et al., 2018; *"Deep learning-based classification of work-related physical load levels in construction", por* Yang et al., 2020.

Asimismo, existen otros comportamientos específicos en referencia al incumplimiento de procedimientos, reglas y normas establecidos. En efecto, Ding et al. (2018) desarrollaron un modelo híbrido que emplea red CNN (*inception v-3* diseñado por *google*), para extraer los información temporal-espacial en los videos, y una red RNN (*long short-term memory)* para reconocer 4 acciones peligrosas relacionadas a las caídas desde escaleras. Entre los recursos del proyecto, los autores emplearon 200 videoclips de 8 segundos cada uno comprendiendo ascensos y descensos por la escalera de espaldas, cargando objetos, entre otras manipulaciones. Además, el performance del modelo alcanza valores de 97% y 92% en *accuracy* al reconocer comportamientos seguros e inseguros, respectivamente. Del mismo modo, Chen et al. (2019) estudiaron 8 comportamientos inseguros en trabajos que requieran de emplear escaleras mediante sensores UWB (para analizar las posiciones) y modelos esqueléticos 3D en base a sistemas neuronales (para determinar las posturas de los operarios). Además, el desempeño del modelo presentó un valor de 0.83 en *accuracy* y el recurso de entrada consistió en la base pública *Human3.6m* que contiene 3.6 millones de muestras de posturas.

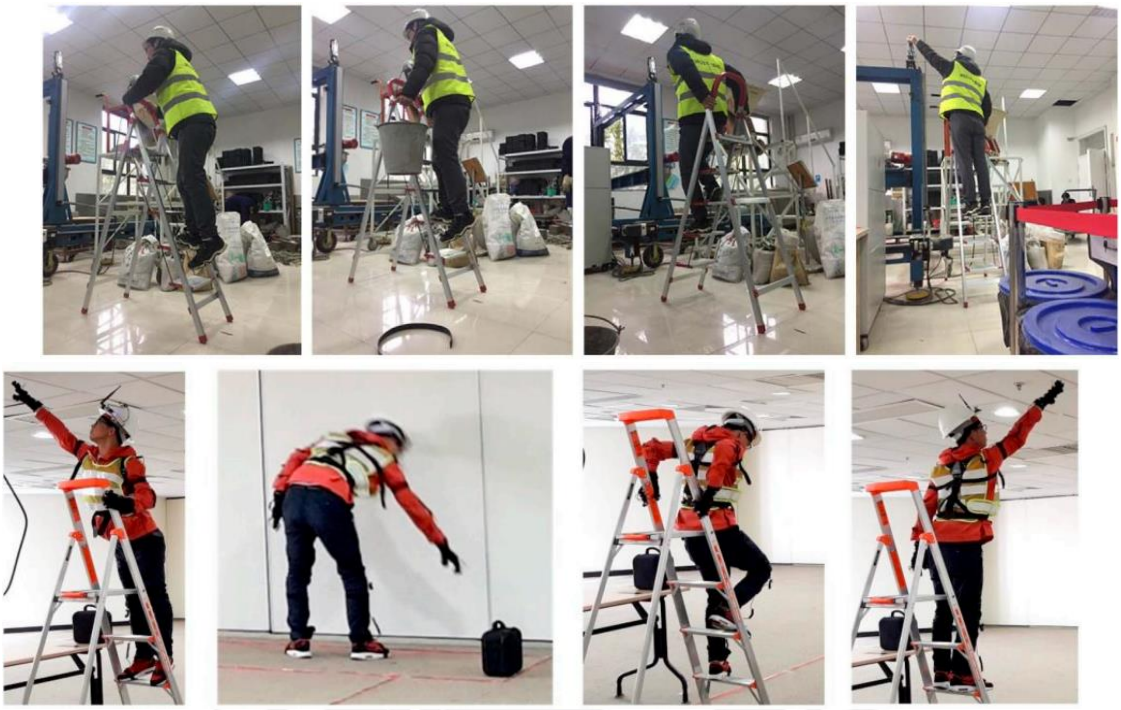


Figura 16. *Ejemplos de movimientos inseguros al trabajar con escaleras.*

Nota. Tomado de "A proactive workers' safety risk evaluation framework based on position and posture data fusion", por Chen et al., 2019*; "A Deep hybrid learning model to detect unsafe behavior: Integrating convolutional neurol networks and long short-term memory", por* Ding et al., 2018.

De igual manera, Fang et al. (2019) desarrollaron un modelo en base a *mask* R-CNN que identifica operarios sin arnés de seguridad al desplazarse sobre estructuras de soporte, suspendidas a alturas considerables, para disminuir la trayectoria de sus recorridos. En efecto, la red neuronal (previamente entrenada en MS COCO) es empleada para identificar objetos (obreros, estructura de acero y concreto), mientras que el algoritmo *overlapping detection module* reconoce los comportamientos inadecuados del operario, a través de la superposición de las áreas o pixeles *mask* obtenidas para cada entidad de construcción. Además, los autores recolectaron 2018 imágenes, que se repartieron entre 70% y 30% para el entrenamiento y evaluación del prototipo, respectivamente. En base a ello, el performance del modelo, en identificar los objetos de estudio, resultó en una precisión y *recall* igual a 99% y 74%, respectivamente. Asimismo, el reconocimiento de comportamientos inadecuados permaneció con un desempeño equivalente a un rango entre

75% y 90%, en términos de precisión y *recall*. En esencia, los autores recomiendan extender el set de entrenamiento, pues la data solo comprende la acción de caminar sobre las estructuras de soporte.

Por otro lado, las aplicaciones en referencia a la segunda categoría, condiciones de trabajo inapropiadas, es la identificación de escenarios que presenten un peligro a la vida de los operarios. Por ejemplo, determinar distancias críticas entre equipos de construcción y obreros con el objetivo de evitar colisiones. En efecto, Luo et al. (2018) proponen una metodología, en base a 3 redes CNN (*Stacked Hourglass Network, Cascaded Pyramid Network* y HG-CPN), que estima modelos esqueléticos de equipos de excavación para determinar el movimiento de sus articulaciones y, por ende, identificar la seguridad de sus operaciones. En esencia, el modelo alcanza un valor de *accuracy* equivalente a 93.43%, en base a 6405 imágenes de entrenamiento y sin emplear dispositivos inerciales adheridos en las maquinarias. Además, los autores recomiendan capturar más puntos de visualización durante la operación de excavación para evitar las oclusiones de las articulaciones de la excavadora. De igual manera, Jeelani et al. (2021) desarrollaron una metodología que localiza obreros e infiere potenciales riesgos en sus alrededores, con el objetivo de anticipar fatalidades y registrar las causas de los accidentes. En esencia, los autores emplearon la red neuronal Faster R-CNN para identificar a los obreros mediante *bounding boxes,* mientras que la red Mask R-CNN se elaboró para segmentar los pixeles dentro de las *bounding boxes* y reconocer escenarios peligrosos. Respecto al desempeño de la metodología, el algoritmo generalizó sus objetivos con una precisión del 93%. Entre las limitaciones de la investigación, se presenta el hecho de que no existe gran cantidad de data etiquetada sobre escenarios peligrosos (caídas por aberturas en losas, bordes de una excavación, restricción en zonas inseguras, etc.) disponibles para entrenar modelos neuronales, por lo que se debe desarrollar un nuevo set de imágenes. En efecto, actualmente este algoritmo solo identifica

6 escenarios peligrosos. Además, los autores señalan que la presente metodología no permite calcular la distancia entre el obrero y los objetos dinámicos próximos a él, mas sí respecto a objetos estáticos. Por tanto, los autores sugieren emplear más de 1 cámara para obtener mayor información espacial y contrarrestar la liberación de falsas alarmas. Por último, este sistema requiere de un servidor GPU, al menos una cámara instalada en obra y un ordenador para realizar *streaming.* En esencia, estos equipos involucran un costo adicional al implementar esta metodología en campo.

Asimismo, Guo et al. (2020) desarrollaron un modelo denominado *orientation-aware SSD* que emplea configuraciones de la red VGG-16 e identifica vehículos de construcción, a través de imágenes áreas y técnicas de *orientation-aware bounding box.* Además, el desempeño del prototipo resultó en un valor mínimo de 89.4% de precisión al evaluarse en 240 imágenes. Entre las limitaciones del modelo se presenta la dificultad de conectar el modelo a los equipos UAV para realizar operaciones en tiempo real.

Figura 17. *Ejemplos de estimación de las articulaciones de excavadoras.*

Nota. Tomado de "Dense construction vehicle detection based on orientation-aware feature fusion convolutional neural network", por Guo et al., 2020.

A partir de la Tabla 5, se observa que las metodologías son elaboradas para enfocarse únicamente a atender un problema en específico con métricas, *accuracy*, elevadas. Sin embargo, hasta el momento no existe una propuesta que permita a un único modelo atender y desempeñarse óptimamente en todas las aplicaciones revisadas en este documento.

Tabla 5. *Aplicaciones de modelos en deep learning para el control de riesgos en el sector construcción.*

Referencia	Aplicaciones	Tecnologías		Limitaciones
Fang et al. (2018)	Detección del uso del casco de protección	CNN	Faster R-CNN	Modelo no identifica la identidad del operario que infringe la norma de seguridad. El modelo no reconoce el EPP de manera individual.
Wu et al. (2019)	Detección del uso del casco y reconocimiento de su color	CNN	SSD VGG-16 Resnet-50	Modelo no identifica la identidad del operario que infringe la norma de seguridad.
Xie et al. (2018)	Detección del uso de cascos de protección	CNN	YOLO v-1	El algoritmo se ha empleado para identificar 1 tipo de EPP. YOLO presenta errores al identificar objetos superpuestos.
Nath et al. (2020)	Detección del uso de cascos y prendas reflectivas	CNN	YOLO v-3 VGG-16 ResNet-50	El algoritmo se ha empleado para verificar 2 tipos de EPP's.
Fang et al. (2018)	Detección del uso de arnés de seguridad	CNN	Faster R-CNN	Escasez de imágenes recolectadas sobre operarios trabajando en alturas. No se identifica el estrobo amortiguador que conecta el arnés a la línea de vida.
Fang et al. (2018)	Detección del uso de cascos, arnés y conexión del arnés a la línea de vida	CNN Otras	SSD SORT	El desempeño del modelo decrece al evaluarse en imágenes con EPP's casi totalmente ocluidos. Técnica *bounding box* resulta ineficiente en el etiquetado del estrobo amortiguador. Algoritmo SSD emite errores al detectar objetos superpuestos.
Kolar et al. (2018)	Verificar la instalación de barandas de seguridad en los precipicios	CNN	VGG-16	Se estudia un solo diseño de baranda de seguridad. Modelo confunde el elemento de seguridad con acero de refuerzo y encofrados. Se ignoraron los elementos ocluidos en el *training set.* El modelo opera con imágenes y no en videos.
Siddula et al. (2016)	Detección de barandas de seguridad, cubiertas y arneses de seguridad	CNN	CNN convencional	Se ignoraron los elementos ocluidos en el *training set.* Solo se identifican elementos, pero no se estudian las correlaciones entre ellos (operario equipado con arnés y laborando en

				un área sin cubiertas ni barandas, etc.)
Fang et al. (2018)	Verificar que los trabajos sean ejecutados por personal autorizado	CNN Otras	Faster R-CNN MTCNN SORT	No se ha desarrollado un mecanismo de alertas en tiempo real.
Yang et al. (2020)	Clasifican niveles de esfuerzo físico al transportar un número determinado de ladrillos	RNN Otras	Bi-LSTM Sensor Inercial	No es posible analizar condiciones físicas en partidas que comprendan otras tareas diferentes al transporte de materiales. Método ignora el análisis de postura inadecuadas. No se considera la fatiga en los operarios como fuente de error. Los aparatos sensoriales obstruyen las operaciones manuales y la productividad.
Zhang et al. (2018)	Identifican posturas ergonómicas mediante modelo esquelético	CNN Otras	CPM Sensor Inercial	Las oclusiones afectan el desempeño del modelo. Los aparatos sensoriales obstruyen las operaciones manuales y la productividad.
Ding et al. (2018)	Reconocen comportamientos inadecuados asociados a trabajos con escaleras	CNN RNN	Inception V3 LSTM	Se reconocen solo 4 movimientos asociados a actitudes inapropiadas en trabajos con escaleras.
Chen et al., (2019)	Reconocen comportamientos inadecuados asociados a trabajos con escaleras	CNN Otras	PAF's Sensor UWB	Los aparatos sensoriales obstruyen las operaciones manuales y su performance se reduce en espacios cerrados. Solo se evalúan comportamientos inseguros en tareas con escaleras.
Fang et al. (2019)	Reconocen comportamientos inapropiados asociados al desplazamiento sobre estructuras suspendidas	CNN	Mask R-CNN	El modelo analiza un solo comportamiento inadecuado. Se requiere extender el set de entrenamiento para lidiar con los efectos de oclusión.
Luo et al. (2018)	Estiman los movimientos de las articulaciones de excavadoras para promover el análisis de estados seguros de sus operaciones y productividad laboral	CNN	*Stacked Hourglass Network* *Cascaded Pyramid Network*	Tiempo de detección se duplica al combinar los dos modelos. Data de entrenamiento es limitada debido a que demanda altas horas hombres y esfuerzos de trabajo. Las operaciones de excavación ocluyen por lo menos uno de los 6 puntos que conforman el modelo esquelético.
Jeelani et al. (2021)	Identifica potenciales peligros en el entorno más próximo a los operarios.	CNN	Faster RCNN Mask RCNN	El modelo solo reconoce solo 6 escenarios peligrosos. El algoritmo no calcula la distancia entre el obrero y el objeto dinámico próximo a él. La implementación en tiempo real requiere de cámaras, servidor GPU y ordenador de transmisión.

2.2.3 Control de la calidad

Los estudios de visión computacional investigados en el ámbito de la calidad se enfocan principalmente en dos aspectos: verificar la calidad de los materiales y la construcción de los elementos acorde a los planos de diseño (correcta ubicación, dimensiones, superficies planas y otras características geométricas) (Xu et al., 2020). Respecto a la primera categoría, diversos autores se enfocan en identificar grietas, corrosiones, infiltraciones, descascaramientos, huecos, eflorescencia, entre otros defectos superficiales de los materiales, debido a que representan visualmente el estado de servicio y durabilidad de la estructura (Hoskere et al., 2020). En esencia, algunos autores consideran apropiado emplear la tecnología *semantic segmentation* sobre *object detection* para capturar los caracteres de los daños, debido a que, estos presentan distribuciones amorfas. Por ejemplo, Kalfarisi et al. (2020) estudiaron la detección automatizada de grietas y fisuras en diversas infraestructuras (puentes, pavimentos de carretera, túneles, edificios y torres de agua), debido a efectos de expansión y contracción por temperatura, esfuerzos de fatiga, entre otros factores. En esencia, los autores evaluaron dos modelos en *object detection*: el primero integra la red Faster R-CNN (para localización de grietas) y la técnica *structured random forest edge detection* (para segmentar las griestas), mientras que el segundo prototipo corresponde a *Mask* R-CNN (ejecuta las tareas de detección y segmentación por sí mismo). Además, los recursos del proyecto consistieron en 1000 y 250 imágenes para entrenamiento y evaluación, respectivamente, comprendiendo fracturas capturadas desde una distancia entre 0.5 y 10 metros. Asimismo, la técnica de etiquetado para entrenar el primer modelo corresponde a *bounding boxes*, mientras que el segundo algoritmo utilizó la práctica *segmentation mask.* Respecto al desempeño de los modelos, Mask-RCNN presenta un mejor performance para la identificación de fisuras, a comparación de Faster R-CNN, alcanzando un valor de 78% en *average precision.* Sin embargo, la métrica del primer

algoritmo respecto a *image segmentation* es inferior y equivalente a 54%. Asimismo, los autores emplearon los resultados de los modelos (grietas identificadas por medio de máscaras de segmentación) para generar una fotogrametría 3D de la estructura y facilitar la localización de las grietas. Además, los autores prefieren trabajar con el modelo Mask-RCNN pues no emplean de etiquetas al estilo *bounding box*, por lo que se evita cubrir áreas no útiles que producen ruido (caracteres falsos) y confunden al sistema.

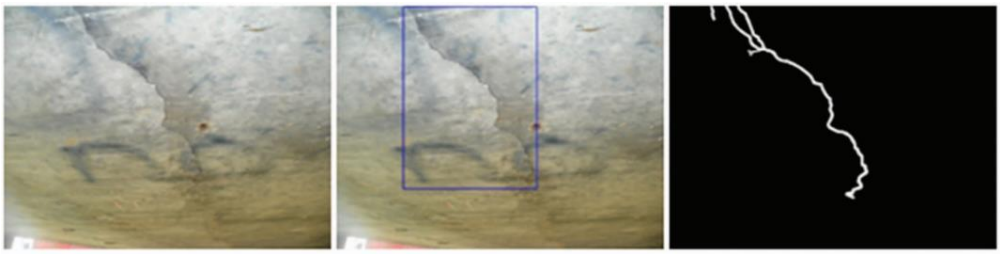


Figura 18. *Etiquetado de imágenes de entrenamiento por medio de bounding boxes y segmentation mask.*

Nota. Tomado de "Crack detection and Segmentation Using Deep Learning with 3D Reality Mesh Model for Quantitative Assessment and Integrated Visualization", por Kalfarisi et al., 2020.

Del mismo modo, Chen & Jahanshahi (2018) desarrollaron la red NB-CNN para identificar grietas en superficies metálicas, a través de videos, y obtuvieron una eficacia de 98.3% en *accuracy*. No obstante, su prototipo solo localiza las grietas mas no cuantifica sus propiedades geométricas. Asimismo, los autores recomiendan incorporar técnicas de *image segmentation* para obtener los anchos y largos de los defectos en estudio. Un estudio similar se elaboró por Dung & Anh (2019) al proponer un método que identifica y segmenta las grietas en superficies de concreto con una precisión de 90%. Los autores evaluaron tres modelos (VGG-16, ResNet50 y InceptionV3), empleando 40000 imágenes correspondientes a una base de dato pública, y se obtuvo que la red VGG-16 presentó el mejor desempeño. Asimismo, Dais et al. (2021) implementaron, por primera vez, diferentes modelos para la identificación de grietas en superficies de muros de albañilería, a nivel de píxeles segmentados. En esencia, los autores obtuvieron el mejor desempeño, 95.3% en *accuracy*, empleando la red *MobileNet*, junto a la técnica *transfer learning*, y un menor

desempeño (89%) al evitar la transferencia de aprendizaje. Además, los investigadores comentaron que el rendimiento del modelo disminuye en más del 5% al identificar grietas en otros tipos de superficies (concreto, asfalto y madera).

Una metodología adicional, en base a VGG-16 y ZF Net, para identificar efectos de corrosión en superficies metálicas fue propuesta por Atha & Jahanshahi (2018). En efecto, los autores obtuvieron un desempeño de 93.50% en *accuracy,* al emplear 926 imágenes de estudio. Sin embargo, el prototipo no se diseñó para clasificar el grado de la corrosión en el material ni determinar el área de la sección afectada.

De igual manera, Li et al. (2019) desarrollan una red CNN, denominada ResNet18, para identificar y clasificar 7 defectos presentes en tuberías de alcantarillado (depósitos sedimentados, deformación y rotura de la tubería, etc.) debido al desgaste por aguas residuales, presiones hidráulicas, entre otros factores. De esta manera, los autores buscan automatizar la inspección y rehabilitación de las instalaciones de alcantarillado con el propósito de evitar la infiltración de aguas negras al terreno, contaminación de la red de agua potable, entre otros objetivos. En efecto, los científicos emplearon técnicas de *data augmentation* para obtener 172840 imágenes de entrenamiento y evaluación, a partir de capturar 18333 imágenes en campo. Además, el modelo obtuvo un desempeño de 83.2% en *accuracy* y presentó dificultades en clasificar los defectos de la tubería individualmente, debido a que, por lo general, coexisten en simultáneo (depósitos sedimentados junto a rotura de tubería, entre otros.). Un estudio similar se elaboró por Kumar et al. (2018) al proponer una red CNN convencional, entrenada en un set de 12000 imágenes, que clasifica los defectos mencionados en las tuberías con un porcentaje de 86.22% en *accuracy*. En esencia, los autores emplearon dispositivos *closed circuit televisión* (CCTV) para inspeccionar y capturar imágenes sobre las condiciones de las tuberías de alcantarillado. Entre las limitaciones del prototipo se presentan la inhabilidad de distinguir defectos entre

las subclases (grietas en espiral respecto a las longitudinales); determinar su localización respecto a la circunferencia de la tubería, para identificar defectos críticos; y el entrenamiento del modelo en base a imágenes estáticas, por lo que se limita el desempeño del prototipo al no considerar información temporal que permita diferenciar entre siluetas complicadas.

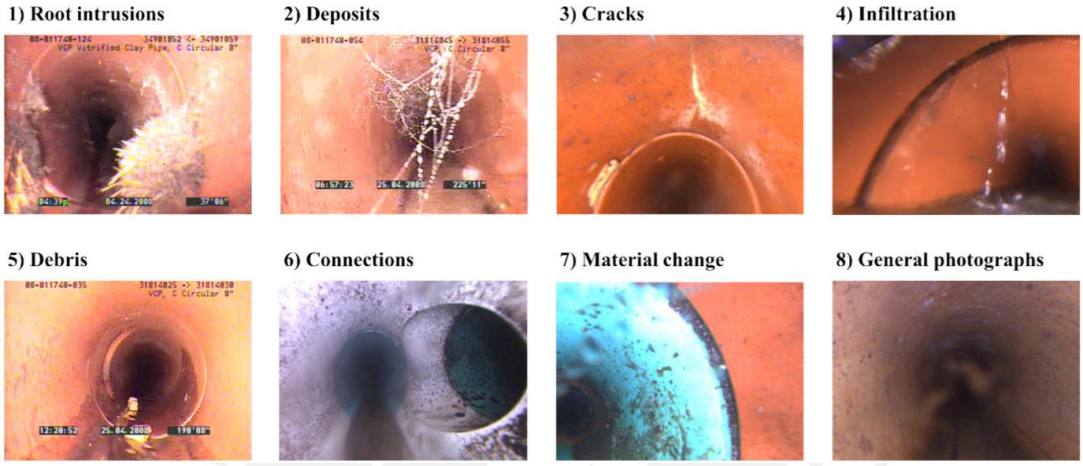


Figura 19. *Clasificación de defectos en tuberías de alcantarillado.*

Nota. Tomado de "Automated defect classification in sewer closed circuit television inspections using deep convolutional neural networks", por Kumar et al., 2018.

Asimismo, (Fan et al., 2019), motivados por la inspección y mantenimiento de autopistas, desarrollaron una red CNN convencional para el reconocimiento de grietas y fisuras en superficies de asfalto con un rendimiento de 99.92% en *accuracy*. Por lo que se conoce, la metodología consistió en usar la red neuronal para clasificar las grietas y la detección se realizó por medio de un algoritmo *hand-craft feature*, denominado *k-mean clustering.* Además, los autores señalan que el sistema se limita a identificar defectos que se presenten en un área menor a 100 píxeles. De igual manera, Fan et al. (2018) propusieron un modelo supervisado en base a una red CNN convencional para identificar la geometría de las grietas en superficies de asfalto. Los autores emplearon 2 bases de datos públicas (AigleRN y CFD), que comprenden 156 imágenes en conjunto, y la librería *TensorFlow* para programar el modelo. Además, entre las limitaciones presentes en este proyecto se

presenta el juicio crítico de las personas para realizar la etiquetación manual de las grietas. Es decir, si se ignora parte de la extensión geométrica de estas, debido a razones de visibilidad (anchos pequeños), los resultados de detección se afectan drásticamente. Asimismo, Liu et al. (2019) diseñaron la red *DeepCrack* para realizar segmentación de grietas en superficies de asfalto y concreto. En efecto, los autores produjeron 537 imágenes anotadas manualmente para elaborar el prototipo propuesto. Además, los investigadores comentan que la limitación principal del proyecto reside en no desarrollar técnicas de post procesado para extraer los caracteres de las grietas y cuantificarlas geométricamente (obtener el ancho, extensión de ellas, etc.).

Del mismo modo, Yang et al. (2018) se limitaron a elaborar una metodología en base a VGG-19 para ejecutar específicamente la detección de grietas en superficies de concreto armado. En efecto, los resultados de la investigación demuestran que su prototipo no requiere de técnica de post procesado para la obtención de longitudes y anchos de las grietas. Sin embargo, el modelo presenta dificultades al reconocer grietas localizadas en esquinas de superficies debido a la pérdida de información. Además, los autores recomiendan desarrollar prototipos que integren su área de estudio junto a la detección de otros defectos (corrosión en metales, huecos en superficies, infiltraciones, etc.) para promover la aplicación universal del modelo. En efecto, Li et al. (2019) propusieron una metodología denominada DenseNet-121, desarrollada mediante la librería *TensorFlow*, para identificar cuatro defectos en superficies de concreto: grietas, descascaramiento, eflorescencia y huecos. Por lo que se conoce, los investigadores recopilaron 1375 imágenes, etiquetadas manualmente, y obtuvieron una precisión promedio de 91.59%. Asimismo, la red MaDnet fue propuesta por Hoskere et al. (2020) para identificar simultáneamente materiales (concreto, acero y asfalto) y defectos en sus superficies (grietas, aceros de refuerzo visibles, corrosión y descascaramiento). Entre los resultados de la investigación,

se concluyó que el modelo de dos etapas identifica los defectos un 58.3% más rápido, computacionalmente, y aproximadamente 2% más preciso respecto a los prototipos de una sola fase.

Xu et al. (2020) estudiaron los efectos de fatiga que se presentan en puentes de acero. Además, alternativamente a modelos basados en CNN, los autores emplearon *Autoencoder (Restricted Boltzman machine)* para identificar las grietas en la estructura. Asimismo, se obtuvo un valor en *average accuracy* igual a 90.95%. Los autores comentan que la principal limitación se debe a la baja resolución de las imágenes. De igual manera, Rubio et al. (2019) estudiaron la inspección automatizada de la exposición del acero de refuerzo en puentes de concreto, a través de la red VGG-16. Entre los resultados de la investigación, se concluyó que existen caracteres falsos (como rastros de suciedad, otros materiales oxidados, sombras intensas, entre otros) que limitaron el desempeño del prototipo a un valor de 78.4% en accuracy. Un estudio similar se realizó por Cha et al. (2017) al emplear *deep learning* para identificar defectos en la infraestructura civil: grietas, corrosiones y delaminación del acero. Por último, en la Tabla 6 se elaboró un breve resumen sobre los artículos revisados.

Tabla 6. *Aplicaciones de modelos en deep learning para el control de calidad en el sector construcción.*

Referencia	Aplicaciones	Tecnologías		Limitaciones
Kalfarisi et al. (2020)	Detección de grietas y fisuras en infraestructuras	CNN	Faster R-CNN Mask R-CNN	Técnica *bounding box* captura áreas no útiles que confunden al sistema.
Chen & Jahanshahi (2018)	Detección de grietas en superficies metálicas en plantas nucleares	CNN	NB-CNN	El prototipo no determina el ancho ni largo de las grietas. Se necesita GPU para ejecutar el modelo.
Dung & Anh (2019)	Reconocimiento de grietas en superficies de concreto	CNN	VGG-16	No se cuantifican las dimensiones de las grietas.
Yang et al. (2018)	Identificación de grietas en superficies de concreto	CNN	VGG19	Modelo presenta dificultades en reconocer grietas interceptadas y otras ubicadas en esquinas debido a la pérdida de información.
Dais et al. (2021)	Detección de grietas en superficies de	CNN	MobileNet	Set de imágenes insuficientes. Modelo reduce su desempeño al identificar grietas en otros tipos

	muros de albañilería			de superficies (concreto, asfalto y madera).
Atha & Jahanshahi (2018)	Detección de corrosión en superficies metálicas correspondientes a sistemas estructurales	CNN	VGG-16 ZF Net	El sistema no diferencia entre los diferentes tipos de corrosión. No se determina el área de la sección corroída.
Li et al. (2019)	Reconocimiento de roturas, deformación y depósitos sedimentados en tuberías de alcantarilla	CNN Otras	Resnet18 CCTV	Escasez de imágenes recolectadas sobre operarios trabajando en alturas. No se identifica el estrobo amortiguador que conecta el arnés a la línea de vida.
Kumar et al. (2018)	Reconocimiento de roturas, deformación y depósitos sedimentados en tuberías de alcantarilla	CNN Otras	CNN convencional CCTV	No se distinguen defectos entre subclases. Modelo no identifica la localización de los defectos respecto a la circunferencia de la tubería. El prototipo no captura información en videos.
Fan et al. (2019)	Reconocimiento de grietas en superficies de asfalto	CNN	CNN convencional	No se identifican grietas y fisuras con un área menor a 100 píxeles en la base de datos.
Fan et al. (2018)	Reconocimiento de grietas en superficies de asfalto	CNN	CNN convencional	Las etiquetas manuales de las grietas influencia drásticamente los resultados de detección.
Liu et al. (2019)	Segmentación de grietas en superficies de asfalto y concreto	CNN	DeepCrack	Se requieren técnicas de post procesado para obtener la segmentación completa de fisuras y cuantificar sus características (espesor, largo, etc.).
Li et al. (2019)	Detección de grietas, descascaramiento, eflorescencia y huecos	CNN	DenseNet121	Redes neuronales no permiten analizar la profundidad del daño de la superficie debido al empleo de imágenes 2D.
Hoskere et al. (2020)	Reconocimiento de grietas, descascaramiento, corrosión y acero de refuerzo visible	CNN	MaDnet	No se comentaron limitaciones.
Xu et al. (2020)	Reconocimiento de grietas en puentes de acero.	Auto-enco-der	*Restricted Boltzman machine*	La calidad de las imágenes incide en el desempeño del modelo.
Rubio et al. (2019)	Identificación de acero de refuerzo expuestos en puentes de concreto	CNN	VGG-16	Prototipo confunde la textura del acero expuesto con otros materiales oxidados, sombras intensas y rastros de suciedad.

2.2.4 Progreso de la obra y gestión de costos

En la práctica, se destinan horas hombres para inspeccionar el avance de la obra con la finalidad de identificar retrasos u oportunidades en el cronograma del proyecto, predecir la fecha fin de obra, verificar la calidad, entre otras aplicaciones. En este contexto la función *object detection*, propia de la visión computacional, puede automatizar el monitoreo continuo de los cambios constructivos efectuados en los ambientes de obra. De esta manera, al percibir los avances (mediante la aparición de columnas, vigas o muros en un nivel específico) resulta factible cuantificar el progreso de la obra. Por ejemplo, Zhu & Brilakis (2010) estudiaron la detección de columnas de concreto mediante un algoritmo tradicional en *object detection* que comprende el extractor *edge detection* para inferir caracteres de contorno. Por lo que se conoce, el modelo ignora caracteres de color y textura debido a que no permiten diferenciar a los elementos estructurales cuando son construidos con el mismo material (concreto armado). Además, los autores mencionan que la metodología depende de una regla de decisión asumida en el sistema: representan las columnas como elementos que disponen de una relación de base entre altura menor a 1, para diferenciarse de los muros y placas estructurales. Entre las limitaciones del proyecto, el modelo no reconoce columnas alejadas del foco de la cámara, debido a que su relación de altura entre base resulta mayor a 1. Asimismo, algunos muros capturados en la imagen o video con ratio menor a 1 son confundidos por columnas. Similares estudios en base a aprendizaje profundo para la detección de elementos estructurales no se han registrado en la literatura hasta el momento.

Un diferente enfoque consiste en realizar comparaciones entre escáneres 3D y modelos BIM 4D para monitorear el progreso de la obra. Por ejemplo, Pučko et al. (2018) generaron modelos 4D *as-built* del proyecto a través de escáneres 3D localizados en los cascos de seguridad de los obreros. De esta manera, el sistema realiza comparaciones respecto al

modelo BIM *as-design* para identificar progresos en el proyecto o actividades retrasadas según la programación. No obstante, esta metodología no requiere de herramientas en *deep learning.* Entre las limitaciones de la propuesta, se presentan el sobrepeso actuante en el casco de seguridad, el sobrecosto de los dispositivos y la dificultad de conexión al servidor en ambientes cerrados. De igual manera, Han et al. (2015) propusieron una metodología para inferir el avance de ciertas actividades a partir de información 3D *as-built* incompleta, debido a oclusiones y limitaciones de visibilidad en campo. Por lo que se conoce, los autores incorporan un criterio de correlaciones entre los elementos estructurales: elementos atados, soportados, embebidos, cubiertos y bordeados. Sin embargo, la investigación no hace referencia a técnicas de *deep learning.* Un estudio similar se llevó a cabo por (Son et al., 2017) al desarrollar un sistema que actualiza el cronograma de obra en el software *Microsoft Project,* a través de comparar la data *as-built* y el modelo BIM. No obstante, los autores comentan que la primera limitación corresponde a la colección manual de data 3D en campo. En general, los estudios de esta categoría se han limitado a identificar el progreso de la obra respecto a trabajos estructurales mientras que no se han enfocado reconocer avances en actividades de arquitectura, mecánica, instalaciones sanitarias y eléctricas.

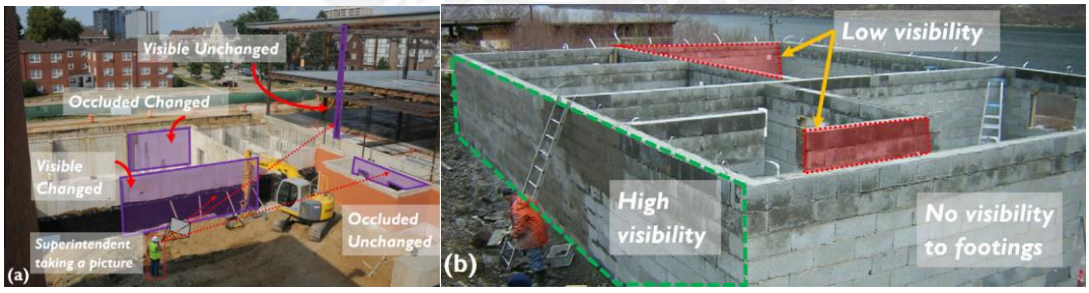


Figura 20. *Ilustración de diferentes tipos de visibilidad en elementos estructurales a ser empleados en el análisis del monitoreo del avance de obra.*

Nota. Tomado de "Formalized knowledge of construction sequencing ofr visual monitoring of work-in-progress via incomplete point clouds and low-LoD 4D BIMs", por Han et al., 2015.

Por otro lado, Turkan et al. (2013) monitorearon el avance del proyecto y transformaron los objetos identificados en términos de valor ganado, mediante un modelo 4D en *object*

recognition. En efecto, la metodología aplica algoritmos algebraicos denominados *iterative closest points* y *surface-based recognition* para reconocer los elementos de campo, representarlos en un entorno BIM 3D y realizar una comparación con el modelo BIM del proyecto. Respecto al performance de la metodología, se exhiben valores de 93% y 83% en *average accuracy* y *recall*, respectivamente. Asimismo, los autores indican que el nivel de detalle trabajado en el modelo 3D representa una limitación, debido a que en campo se percibieron encofrados, puntales y otros elementos no definidos virtualmente. Además, el modelo exige recolectar imágenes 3D en varias localizaciones para evitar extraer fotogrametría con datos incompletos. Adicionalmente, los investigadores mencionan que la convención de cálculo concebida en el valor ganado limita el monitoreo del avance del proyecto para algunos elementos. por ejemplo, en estructuras de acero los costos registrados se realizan en términos de toneladas de acero ejecutadas, en lugar del elemento completamente construido. En general, todas las aplicaciones destinadas a monitorear el progreso de la obra proporcionan información útil para gestionar los costos del proyecto mediante la incorporación de la teoría del valor ganado. En esencia, esta consiste en vincular cada elemento del proyecto con su respectivo valor ganado (Xu et al., 2020).

2.2.5 Gestión de activos

Una de las primeras aplicaciones está enfocada en el ahorro de energía que consumen los activos de la edificación y es logrado mediante la determinación cuantitativa de la ocupación de sus ambientes en tiempo real. De esta manera, es posible regular las cargas de calor y frío en la ventilación dentro de la edificación, por lo que se reducen los gases de efecto invernadero; automatizar el apagado de luces al no identificar la presencia de personas en ambientes específicos; determinar rutas de evacuación y rescate en eventos de emergencia; identificar las horas pico de mayor consumo de energía; entre otros usos. Por ejemplo, Chen & Jiang (2018) propusieron una metodología, en base a la red *Generative*

Adversial Network, que determina el tiempo de llegada, tiempo de salida, duración de ocupación del ambiente y la cantidad de transiciones que el ocupante realiza entre los ambientes. En efecto, el aspecto novedoso de su trabajo consistió en no recurrir a consideraciones iniciales que relacionan los comportamientos de los ocupantes. Es decir, no se dividen a los ocupantes en grupos o eventos (trabajando en la oficina, almorzando, en servicios higiénicos, etc.) para determinar los patrones de los ocupantes. Sin embargo, el modelo se evaluó en el campus de la Universidad de Florida, por lo que reside la incertidumbre de cómo se desempeña en otros tipos de edificaciones.

Otras aplicaciones de la tecnología *deep learning* en la gestión de activos está orientadas a procesar datos que no necesariamente son píxeles. Por ejemplo, Singaravel et al. (2018) asumieron parámetros de ocupación para determinar el diseño eficiente de consumo energético (demandas térmicas mensuales) a partir de la fase de concepción del edificio. De esta manera, la metodología indica potenciales elementos estructurales a incluir en el diseño para cumplir con demandas energéticas deseadas. En efecto, los autores evaluaron diferentes modelos en base a la red *Long Short Term Memory.* Entre las limitaciones de la investigación, se presentan los espacios geométricos empleados para entrenar el sistema, que corresponden a una estructura de 3 pisos, puesto que existen diferentes configuraciones en altura y área en otros edificios. Asimismo, la verificación de los resultados se efectúa mediante una simulación en el software *EnergyPlus,* por lo que los datos empleados, respecto a la calefacción, no corresponden a fuentes netamente reales. Un estudio similar es presentado por Fan et al. (2017) al desarrollar un modelo que predice el consumo diario de energía para alimentar los equipos de refrigeración de los edificios (ventiladores, aire acondicionado y calentadores), a partir de la data histórica de las últimas 24 horas (temperatura exterior, humedad exterior, entre otros datos). En efecto, los resultados del proyecto demuestran que la cantidad óptima de capas ocultas a emplear en la red neuronal

para realizar predicciones exitosas es de 2, mientras que un aumento en ellas no incrementa significativamente el desempeño del modelo.

De igual manera, Rahman & Smith (2018) evalúan modelos en base a la red *Recurrent Neural Networks* para estimar las demandas de calor que requieren los tanques térmicos de centros comerciales, en diferentes escalas de tiempo, para proveer agua caliente. De esta manera, se busca optimizar el consumo de energía de estos dispositivos. Adicionalmente, los autores buscan proporcionar bases de diseño en el cálculo de la capacidad de los tanques. Sin embargo, la metodología presenta limitaciones al predecir las demandas de calor a largo plazo pues asume parámetros del clima que no consideran eventos aleatorios a corto plazo. Una segunda limitación consiste en actualizar el set de entrenamiento para considerar los cambios en la relación de consumo de agua caliente respecto a la ocupación del centro comercial. Un estudio similar es elaborado por Rahman et al. (2018) al diseñar una metodología en base a redes RNN, que comprende el modelo *Long Short Term Memory*, para predecir las demandas de energía del sistema HVAC. En efecto, los resultados del proyecto demostraron que el modelo RNN efectúa mejor desempeño respecto a la red CNN.

ET LUX IN TENEBRIS LUCET
MCMXVII

3. IMPLEMENTACIÓN DEL APRENDIZAJE PROFUNDO

De acuerdo al capítulo anterior, en relación al control de riesgos en obra, existen diversos estudios que se enfocaron en la identificación de ciertos equipos de protección personal y colectiva: cascos, arnés de seguridad, obreros y maquinarias. Sin embargo, en obra existen una inmensa variedad de objetos que se correlacionan con la protección de la salud y vida del personal. En ese contexto, este proyecto de investigación validará la eficacia de las metodologías, en base a *deep learning*, empleadas por la comunidad de la visión computacional, considerando una mayor variedad de equipos de seguridad. En ese sentido, el siguiente flujo de trabajo se resume en recolectar imágenes con los objetos en estudio, procesar la información y emplearla para entrenar los modelos propuestos.

3.3 Generación de data

3.3.1 Criterios y atributos de etiquetación

En este apartado se emplearon tres definiciones importantes para continuar con el proceso de etiquetación de imágenes: clases, refiere a una etiqueta que representa el objeto en estudio; *bounding boxes,* representa una caja que bordea toda la extensión del objeto visible en la imagen; y *tags,* alude a los criterios de anotación que sirven de soporte a ciertos modelos, en la fase de entrenamiento, para seleccionar las imágenes que resultan más adecuadas de procesar (Everingham et al., 2010). En esencia, los diferentes *tags* incluidos en este proyecto se presentan en la Tabla 7 y la Figura 21 ilustra el uso de estas etiquetas.

Tabla 7. *Atributos y criterios de etiquetado.*

Elemento	**Criterio**
Clases	Representa a un objeto en estudio.
Bounding Boxes	Solo se remarca el área visible del objeto en la imagen (no su extensión sobre zonas ocluidas) con una tolerancia máxima de 5 píxeles.

Tags	*Occluded*	Si la entidad de estudio presenta un área ocluida entre 15 a 75% dentro de la *bounding box*, esta categoría indica que el artículo no está totalmente visible. Sin embargo, no se consideran a las vestimentas (polo reflectivo, pantalón reflectivo, etc.) como elementos que ocluyan a la entidad en estudio.
	Truncated	Si el objeto de análisis presenta un área superior al 15% fuera de la *bounding box*, esta anotación indica que la caja de etiqueta no cubre la extensión total del elemento. Es decir, el objeto no ha sido capturado en su totalidad en la foto.
	Difficult	Se refieren a las *bounding box* que presentan un área mayor al 75% del elemento como truncado y/o ocluido. Adicionalmente, se consideran las etiquetas que engloban zonas con poca iluminación, baja calidad y, en general, efectos que dificultan el reconocimiento del elemento.

Nota. Tomado de "The Pascal Visual Object Classes (VOC) challenge", por Everingham et al., 2010; *"Development of an Image Data Set of Construction Machines of Deep Learning Object Detection", por Xiao & Kang, 2020.*

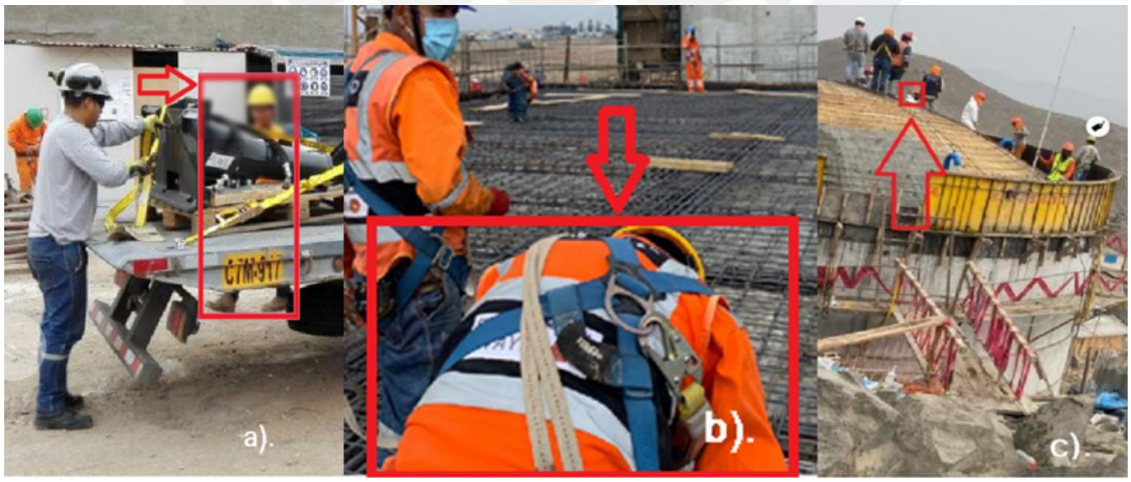


Figura 21. *a). Ilustración de etiqueta ocluida, pues el operario presenta un área oculta entre 15% a 75%. b) Etiqueta truncada, debido a que no se captura la parte inferior del obrero. c). Etiqueta difícil pues existe pérdida de información para el reconocimiento del elemento.*

3.3.2 Recolección de fotos

El proceso de recolección de imágenes se realizó de manera *in-situ* y *online* por el equipo de investigación conformado por: Yuri Gabriel Pila, Danny Murguía (asesor), Edson Zanabria, Rosario Barriga, Edson Bécker Arias, Gianmarco Murguía, Liliana Ruiz Llanos, Miguel Ibarra Navarro, Shirley Corilla y Augusto Ríos. Respecto a la recolección *in-situ*, se obtuvieron 859 imágenes, en un intervalo de tiempo de 2 meses, que capturan operarios usando equipos de protección colectiva y personal, señales de seguridad y estructuras temporales. En efecto, se reconoce que el principal dispositivo empleado para esta tarea consistió en cámaras de celulares. Además, se consideraron diferentes condiciones visuales en las imágenes, es decir, factores que dificulten el desempeño de los modelos de visión computacional: variación de las posturas, variedad de intraclase (estaturas, tez de color, contexturas, etc.), diferentes intensidades de iluminación, oclusiones, aglomeraciones, entre otros efectos. Ejemplos de estas condiciones se aprecian en la Figura 22, Figura 23, Figura 24, Figura 25, Figura 26 y Figura 27. Bajo esta misma premisa, la segunda técnica consistió en recolectar data disponible en internet, obteniéndose 104 imágenes en total.

Figura 22. *El set de datos propuesto comprende distintas variaciones de posturas.*

Figura 23. *Se contemplan imágenes a corta distancia.*

Figura 24. *El set de datos comprende imágenes a mediana y larga distancia.*

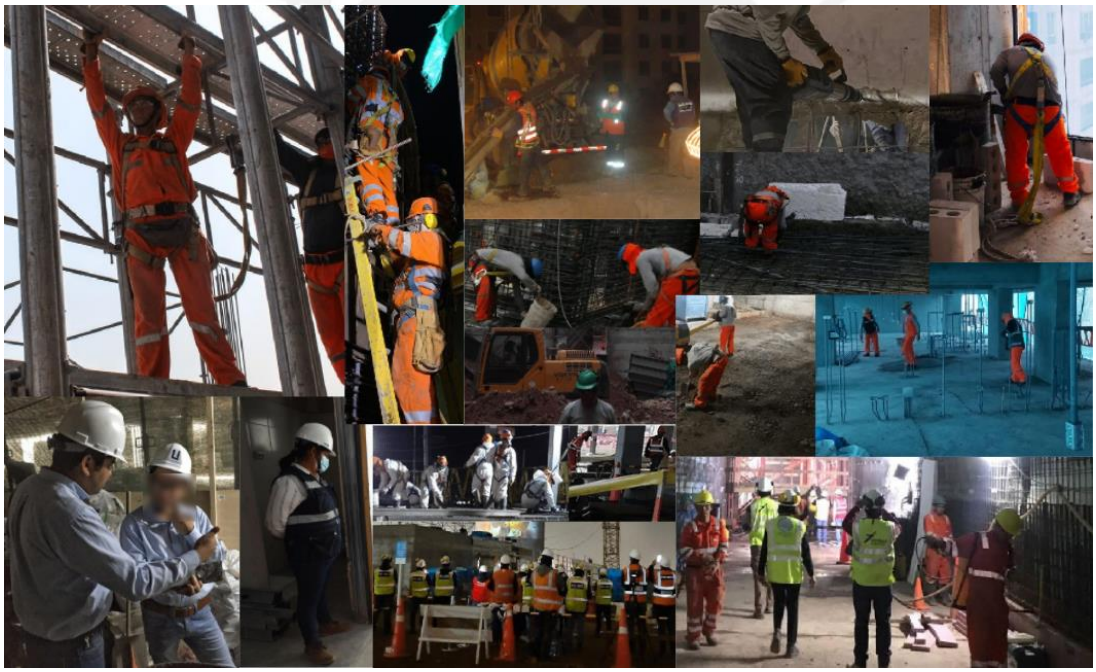


Figura 25. *El set de datos comprende variaciones de intensidad de luz.*

Figura 26. *El set de datos comprende escenas aglomeradas, es decir, existen texturas idénticas entre los objetos.*

Figura 27. *El set de datos comprende objetos ocluidos y truncados.*

3.3.3 Etiquetado de imágenes

Dado que el enfoque de este proyecto es colaborar con investigaciones actuales y futuras en desarrollar un set de datos con gran variedad de objetos de construcción bajo diferentes condiciones visuales, se procedió a identificar la mayor cantidad de elementos posibles relacionados con la seguridad en obra. En consecuencia, se obtuvieron 33 clases,

representadas en la Figura 30, distribuidas de la siguiente manera: 25 objetos corresponden a los accesorios de EPP, EPC y señales de prevención, mientras que los 8 restantes corresponden a rangos del staff de obra y tipos de maquinaria pesada. En base a ello, se procedió a capturar los objetos en las imágenes mediante etiquetas rectangulares y manualmente por el autor de la tesis. Además, se siguieron tres estándares establecidos en la base de datos *VOC2007*: consistencia, en referencia a seleccionar las etiquetas adecuadas y ubicar correctamente las *bounding boxes*; precisión, se busca evitar todo tipo de errores en el proceso de etiquetado como la otorgación de anotaciones incorrectas a los objetos; y completado, respecto al trabajo exhaustivo de etiquetar todo objeto existente en cada imagen a pesar de la presencia de desenfoques y objetos en pequeña escala. La Figura 28 ilustra el carácter exhaustivo y preciso del trabajo de etiquetado.

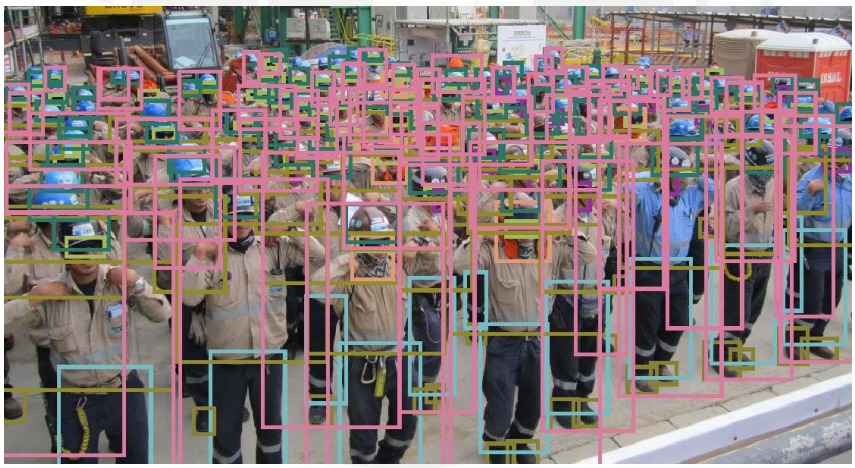


Figura 28. *Ilustración de etiquetado exhaustivo y preciso.*

Además, en este proyecto se trabajó con la plataforma *Supervisely* para realizar el proceso de etiquetación. La Figura 29 ilustra la interfaz de esta herramienta. En esencia, el tiempo de anotación promedio resultó en aproximadamente 10 minutos para cada 40 objetos. Específicamente, este tiempo está distribuido en la identificación de los objetos en la imagen y correlacionar con 1 de las 30 etiquetas disponibles; procurar que la *bounding box* encaje adecuadamente en la extensión del elemento; analizar los pixeles visibles del elemento para aplicar los *tags* de difícil, ocluido o truncado; encender y apagar las etiquetas

actuales para reconocer las que se enmarcaron y las que restan por anotar; entre otras acciones para controlar la plataforma. En este contexto, el tiempo total estimado para esta etapa resultó aproximadamente de 100 horas de trabajo para un total de 24 102 objetos.

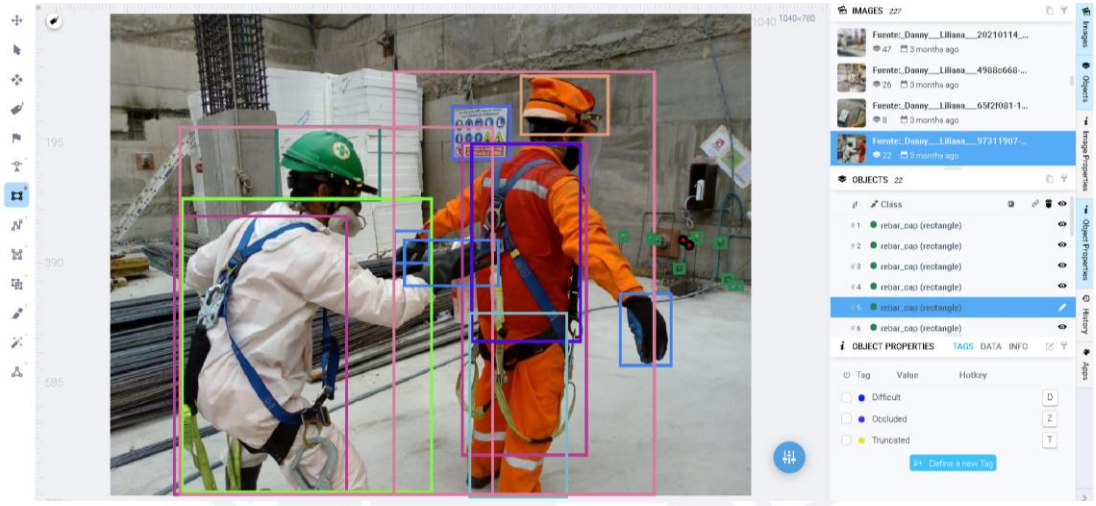


Figura 29. *Ilustración de la interface de la plataforma Supervisely.*

3.3.4 Estadísticas de la data

Al final del proceso de etiquetación, se determinó el número de ejemplares perteneciente a cada clase de objeto. En efecto, en la Figura 30 se aprecia que no todos los objetos de estudio presentan las mismas cantidades de ejemplares, debido al ángulo de la captura y consistencia de objetos en obra. Por tanto, se decidió trabajar con 11 de las 33 clases disponibles debido a que se busca proporcionar la mayor cantidad de información posible a los modelos en visión computacional. Además, no se considerarán los objetos etiquetados con la categoría “difícil”, pues no todos los ejemplares a evaluar disponen de cantidades mayores a 1000 unidades, recomendable para controlar este efecto. De esta manera, la Figura 31 presenta el número de ejemplares originales para las clases propuestas a analizar.


Cascos 3388
Obreros 3309
Botas puntas de acero 2346
Pantalon reflectivo 1699
Guantes 1328
Chaleco reflectivo 1202
Cortaviento 1099
Arnés 927
Tapa de acero 908
Máscara respiratoria 875
Polo reflectivo 863
Ingeniero 793
Conos 668
Barbiquejo 647
Lentes 597
Orejeras 514
Botas impermeables 386
Señales prevención 380
Cachacos de madera 362

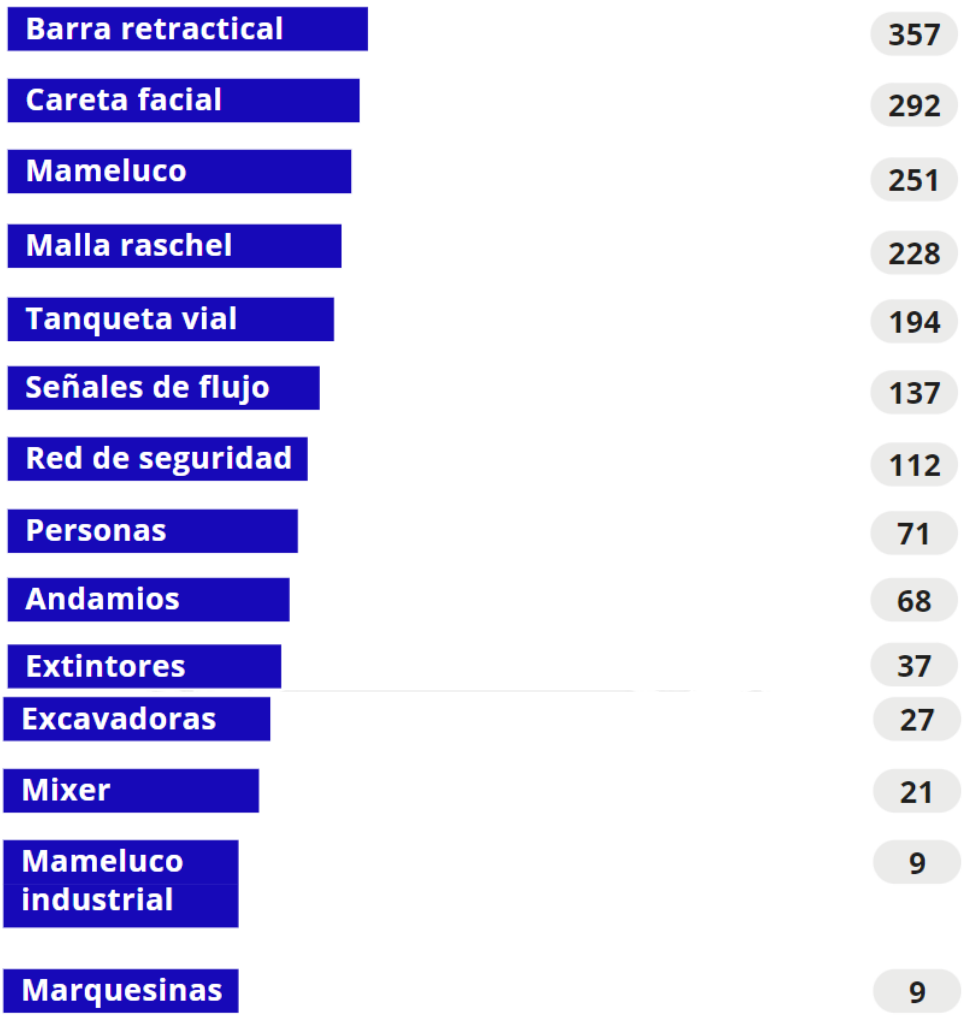


Figura 30. *Estadísticas del set de datos.*

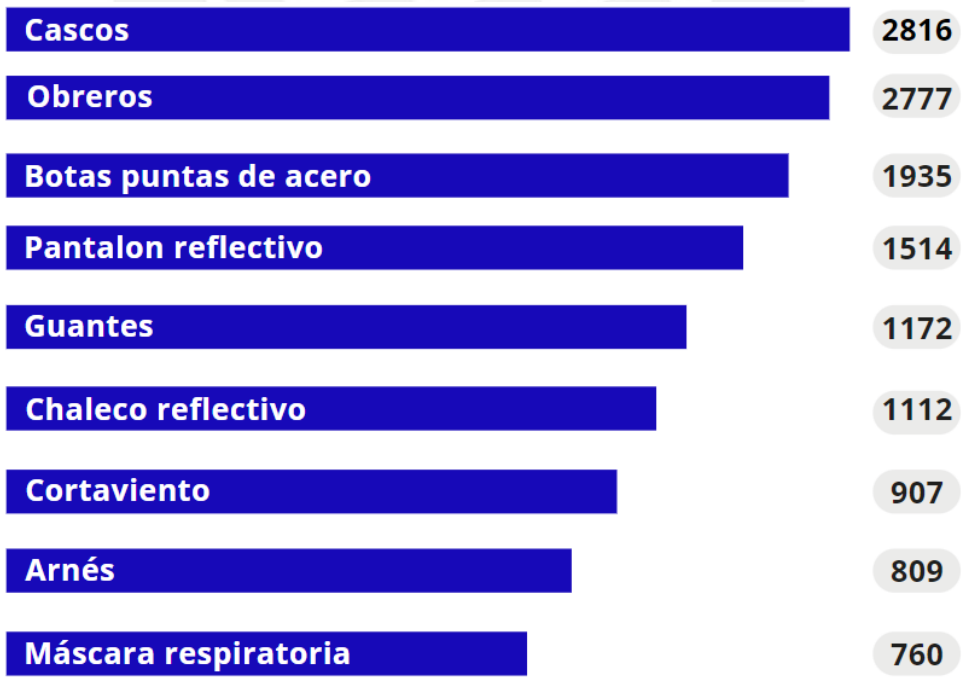


Figura 31. *Número de objetos seleccionados por cada clase sin considerar la etiqueta "Difficult".*

3.3.5 Procesamiento de etiquetas

Una vez culminado el etiquetado manual de los objetos, se procedió a extraer los rectángulos como imágenes individuales en diferentes escalas. Este proceso se llevó a cabo mediante la API de la plataforma *Supervisely* y el código se adjunta en el Anexo 01. Asimismo, como se comprendió en el apartado anterior, los datos a analizar no presentan una distribución equitativa de ejemplares, por lo que se uniformizó el set de datos seleccionando 4 estrategias de aumento de datos: inversión vertical, inversión horizontal, inversión vertical y horizontalmente, y escala de grises. Ejemplos de estas transformaciones se aprecian en la Figura 32 y la cantidad de ejemplares totales se resume en la Figura 33. De esta manera, se evita que los modelos de clasificación de imágenes prioricen el reconocimiento de una respecto a la otra.

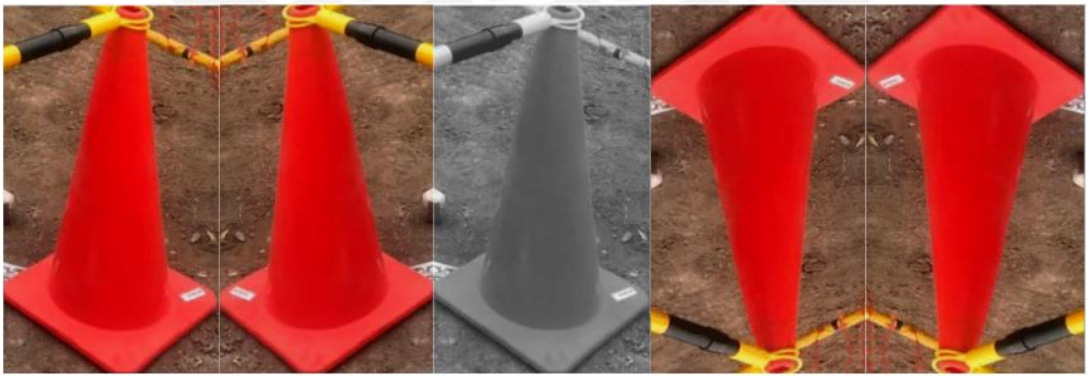


Figura 32. *Ilustración de aumentación de data para una imagen.*

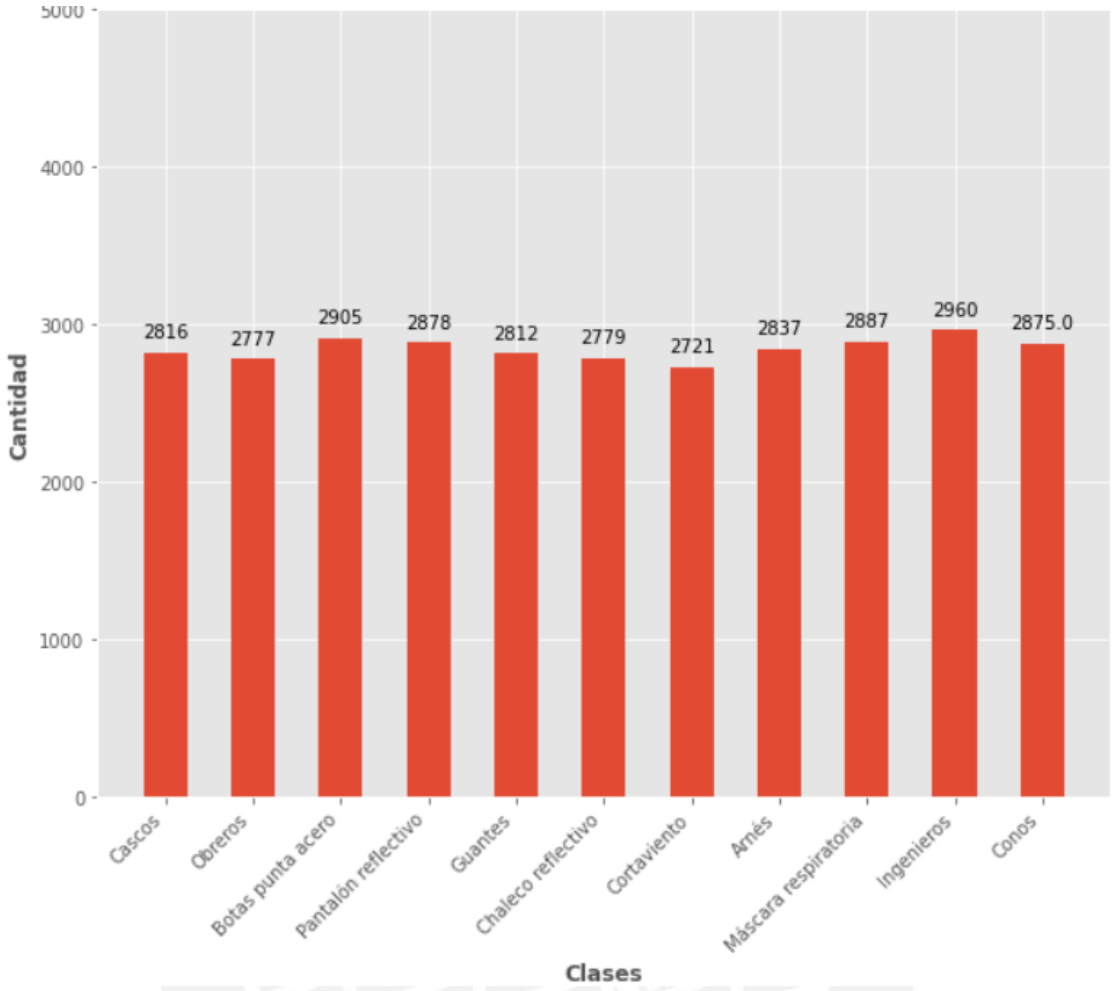


Figura 33. *Estadística de la data uniformizada.*

Adicionalmente, en este trabajo de investigación se aplicarán dos estrategias adicionales de preprocesado comúnmente empleadas en la literatura: escalar las imágenes a las mismas dimensiones, normalizar el rango y estandarizar los valores. Respecto a la primera técnica, las escalas seleccionadas comprenden dimensiones de 224x224 y 299x299, debido a que son requisitos por los algoritmos seleccionados. En relación al segundo pre-procesado, los píxeles originales comprenden valores en el rango de 0 a 255, por lo que se decidió uniformizar el rango a 0 y 1, a través de una división entre 255. Esta técnica es recomendada en la literatura pues las operaciones del algoritmo resultan ser menos engorrosas. Por último, se realizó una estandarización de los valores de los píxeles con el objetivo de enlazar los datos de todas las imágenes recortadas. Por ello, se inició a extraer el valor promedio y la desviación estándar de todo el conjunto de imágenes.

Posteriormente, a cada valor de las imágenes recortadas se sustrajo el promedio y se dividió entre la desviación estándar.

3.4 Generación de los algoritmos

En este proyecto, se trabajó con el enfoque *transfer learning*, que consiste en emplear algoritmos previamente entrenados en un set de datos que podría o no guardar relación con el conjunto de datos propuesto. En efecto, se ha demostrado que la transferencia de los parámetros aprendidos, desde un entorno externo al campo de estudio actual, permite obtener mejores desempeños y capacidades de generalización (Shin et al., 2016). En este contexto, se evaluaron tres redes pre entrenadas en el set de datos externo *ImageNet*: VGG-16 (Simonyan & Zisserman, 2014), Resnet 18 (He et al., 2015) y InceptionV3 (Szegedy et al., 2015). Estos algoritmos fueron seleccionados debido a su uso y desempeño en la literatura, comprendido en el capítulo 2.

3.4.1 Configuración del sistema

Los modelos de clasificación desarrollados en este proyecto han sido implementados en el ambiente de programación *GoogleColab*, bajo el uso de la librería PyTorch. En efecto, se revisó la documentación de PyTorch, OpenCV, Scikit-Learn, entre otros para elaborar el código de cada modelo. Además, la tecnología GPU utilizada corresponde a los modelos Nvidia K80s, T4s, P4s y P100s ofrecidos aleatoriamente por el servicio gratuito de *GoogleColab*. Además, se emplearon 32 *batches* por cada 30 *epochs*, debido al uso limitado de GPU por *GoogleColab*. Asimismo, se determinó emplear una tasa de aprendizaje a 0.001 después de realizar varios ensayos con la curva de aprendizaje. Las características del ordenador presentan las siguientes características: marca Aspire VX5-591G; procesador Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz; memoria RAM de 12GB; y tarjeta de video GeForce GTX 1050.

3.4.2 Desarrollo de la arquitectura

En este proyecto se respetó la arquitectura original de cada modelo en cuanto a capas convolucionales (Conv), *pooling* y las de conexión completa. En esencia, los parámetros que se definieron corresponden a la función de activación, la función de pérdidas, la función de optimización y el afinamiento del número de clases de objetos a identificar (11).

La función de activación seleccionada corresponde a la versión multi categórica de *Sigmoid Activation* denominada *Softmax,* debido a que se lidia con un problema de clasificación de más de 2 clases. Su formulación está definida en la ecuación 1, donde x_i representa el vector de salida de la capa neuronal anterior para el índice "í" en un set de datos con "k" ejemplares. Además, se conoce que el denominador cumple la función de normalizar el resultado en rango de 0 a 1 y, por ende, entrega un valor en términos de probabilidad (Zhang et al., 2021).

$$Softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}$$

Ecuación 1*. Definición matemática de la función Softmax.*

Nota. Tomado de "Dive into Deep Learning", por Zhang et al., 2021.

Por otro lado, se aplicó la definición de Shannon (1948) al emplear la función *Entropy,* pues desde un enfoque informático se conoce que este indicador logarítmico representa la estrategia más adecuada para medir el valor esperado de información recibida, sin importar necesariamente la distribución de los pixeles en la imágen. En efecto, este concepto está incorporado en la función *Cross Entropy,* por lo que es la seleccionada en este proyecto. Es decir, esta función permitirá medir el error de la información obtenida en la imagen por la red neuronal.

$$H[P] = -\sum_{j} P(j) log\ P(j)$$

$$L = \frac{-1}{n} x \sum_{i=1}^{n} \sum_{j=1}^{m} y_{ij} Log\ \hat{y}_{ij}$$

Ecuación 2*. Expresión matemática de la función Cross Entropy.*

Nota. Tomado de "Dive into Deep Learning", por Zhang et al., 2021.

Asimismo, las técnicas más utilizadas como función de optimización son *Stochastic Gradient Descent* (SGD), RMSProp, AdaGrad y *Adaptive Moment Estimation* (ADAM). En efecto, en este proyecto se optó por emplear la función ADAM, debido a que consume menor memoria y combina los factores claves del optimizador RMSProp y AdaGrad respecto a la minimización de la función de pérdidas (Kingma & Lei Ba, 2017). Por ejemplo, esta técnica replica la tendencia promedio de los pesos exponenciales en cada *minibatch* del proceso SGD, entre otras características. En efecto, las constantes de la expresión de ADAM se definieron con los siguientes valores: α=0.001, β_1=0.9, β_2=0.999 y ϵ=10^{-8}.

Obtención del gradiente al intervalo t: $g_t = \nabla_\theta x f_t(\theta_{t-1})$

Estimación del primer momento del gradiente: $m_t = \beta_1 - m_{t-1} + (1 - \beta_1) g_t$

Estimación del segundo momento del gradiente: $v_t = \beta_2 - v_{t-1} + (1 - \beta_2) g_t^2$

Normalización del primer momento: $\widehat{m}_t = \frac{m_t}{(1-\beta_1^t)}$

Normalización del segundo momento: $\hat{v}_t = \frac{v_t}{(1-\beta_2^t)}$

Actualización de los parámetros del modelo neuronal: $\theta_t = \theta_{t-1} - \frac{\alpha . \widehat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$

Ecuación 3*. Expresiones matemáticas para obtener la formulación que actualiza los parámetros del sistema neuronal (Kingma & Lei Ba, 2017).*

Tabla 8. *Arquitectura del modelo VGG16, Resnet y Inception V3.*

VGG16	
Capa	Kernel size / Stride / Padding
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Maxpool	2x2 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1

Resnet18	
Capa	Kernel size / Stride / Padding
Conv	7x7 / 2x2 / 3x3
Maxpool	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 1x1 /1x1

Maxpool	2x2 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Maxpool	2x2 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Maxpool	2x2 / 2x2
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Maxpool	2x2 / 2x2
Fc linear	(Inputs: 25088 Outputs: 4096)
Fc linear	(Inputs: 4096 Outputs: 4096)
Fc linear	(11outputs)

Conv	3x3 / 1x1 /1x1
Conv	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 /1x1
Conv	1x1 /2x2
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 /1x1
Conv	1x1 /2x2
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 2x2 / 1x1
Conv	3x3 / 1x1 /1x1
Conv	1x1 /2x2
Conv	3x3 / 1x1 /1x1
Conv	3x3 / 1x1 /1x1
Fc linear	(11 outputs)

Inception V3	
Capa	Kernel size / Stride / Padding
Conv	3x3 / 2x2
Conv	3x3 / 1x1
Conv	3x3 / 1x1 / 1x1
Maxpool	3x3 / 2x2
Conv	1x1 / 1x1
Conv	3x3 / 1x1
Maxpool	3x3 / 2x2
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	5x5 / 1x1 / 2x2
Conv	1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	5x5 / 1x1 / 2x2
Conv	1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1

Continúa…	
Capa	Kernel size / Stride / Padding
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x1 / 1x1
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x1 / 1x1
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	5x5 / 1x1

Conv	5x5 / 1x1 / 2x2
Conv	1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x1 / 1x1
Conv	3x3 / 2x2
Conv	1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	3x3 / 2x2
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x1 / 1x1
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x1 / 1x1
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	1x7 / 1x1 / 0x3

Fc linear	(11 outputs)
Conv	1x1 / 1x1
Conv	3x3 / 2x2
Conv	1x1 / 1x1
Conv	1x7 / 1x1 / 0x3
Conv	7x1 / 1x1 / 3x0
Conv	3x3 / 2x2
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x3 / 1x1 / 0x1
Conv	3x1 / 1x1 / 1x0
Conv	1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x3 / 1x1 / 0x1
Conv	3x1 / 1x1 / 1x0
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x1 / 1x1
Conv	1x3 / 1x1 / 0x1
Conv	3x1 / 1x1 / 1x0
Conv	1x1 / 1x1
Conv	3x3 / 1x1 / 1x1
Conv	1x3 / 1x1 / 0x1
Conv	3x1 / 1x1 / 1x0
Conv	1x1 / 1x1
Avg pool	
Fc linear	(11 outputs)

1. Nota. Tomado de "Rethinking the Inception Architecture for Computer Vision", por Szegedy et al., 2015.

3.4.3 Pruebas de algoritmos e hiperparámetros

Para el entrenamiento y comparación de los algoritmos se emplea principalmente el set de datos propuesto, debido a que se pretende corroborar la efectividad del mismo, conservar los rasgos y etnicidad nacional de los trabajadores, desarrollar la experiencia completa del proyecto en visión computacional, no prescindir de set de datos externos con posibles errores de etiquetado, entre otros factores. Además, como se comprendió en el apartado anterior, el entrenamiento de las redes neuronales propuestas se resume en minimizar la

función de pérdidas *Cross Entropy* empleando el optimizador *Adam*. De esta manera, los resultados se plasman en la Figura 34, Figura 35 y Figura 36. Estas comprenden curvas de aprendizaje útiles para identificar el performance de cada algoritmo conforme incrementa el número de ciclos.

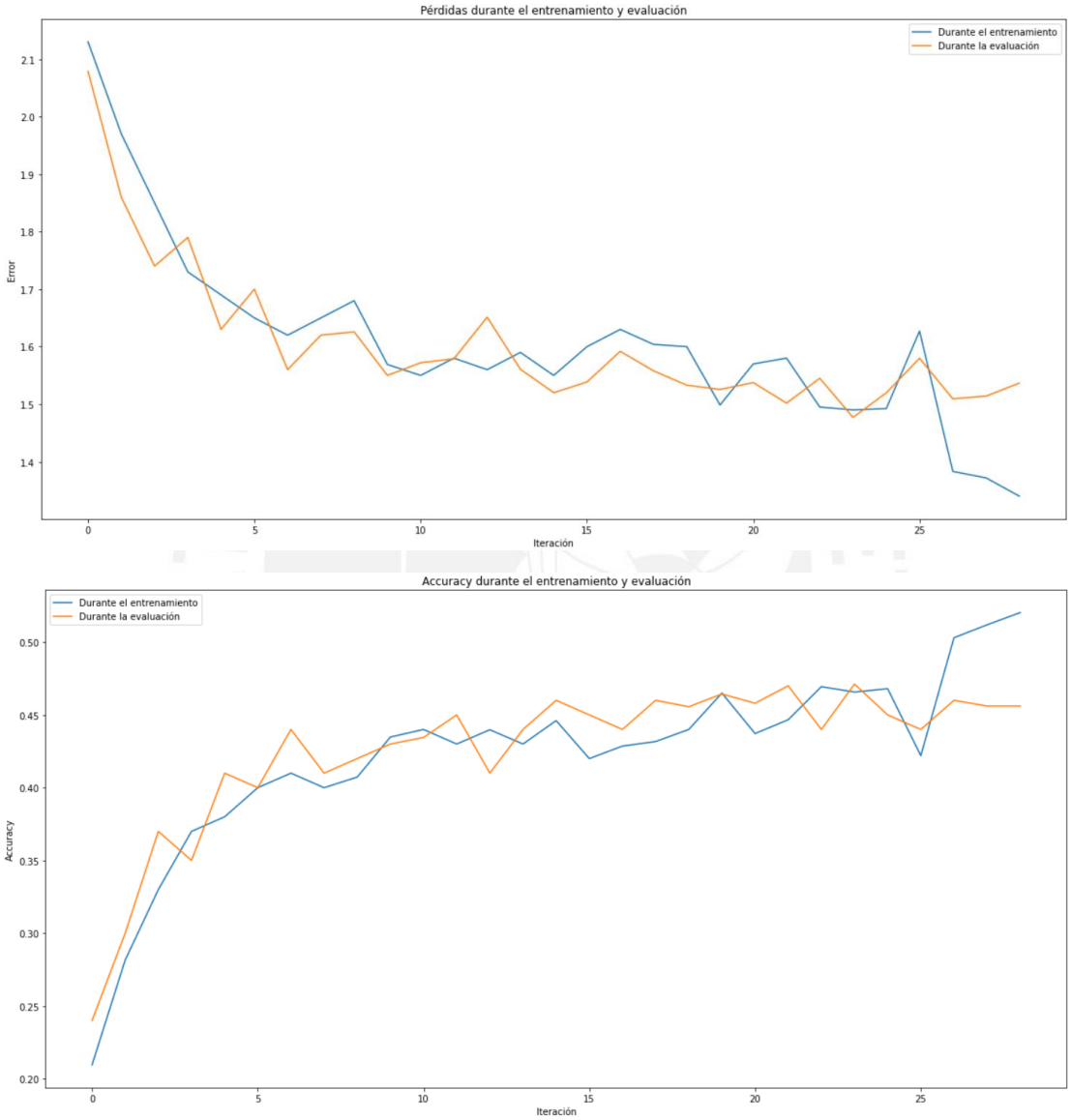


Figura 34. *Curvas de aprendizaje del algoritmo VGG16.*

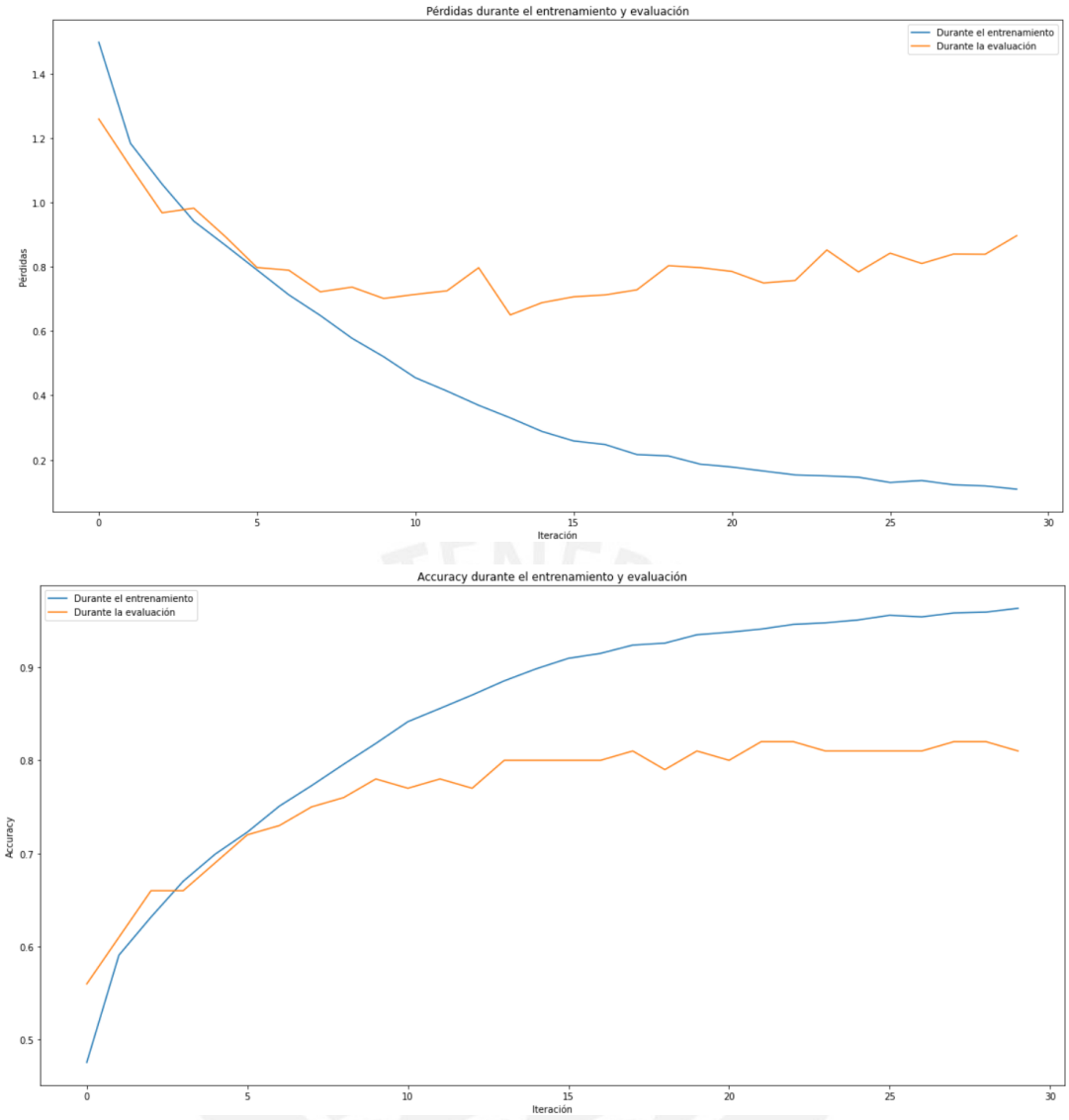
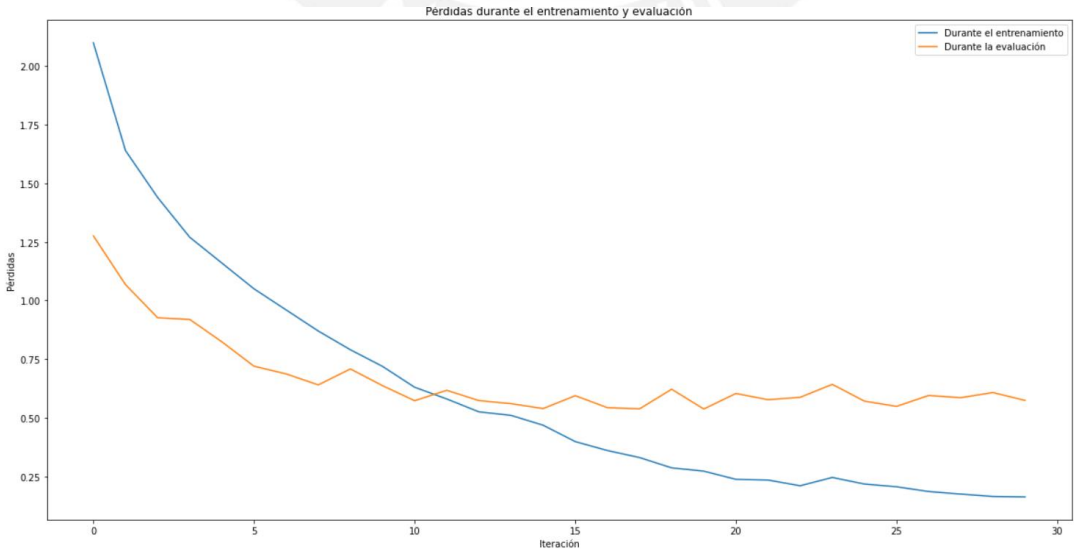


Figura 35. *Curvas de aprendizaje del algoritmo Resnet18.*



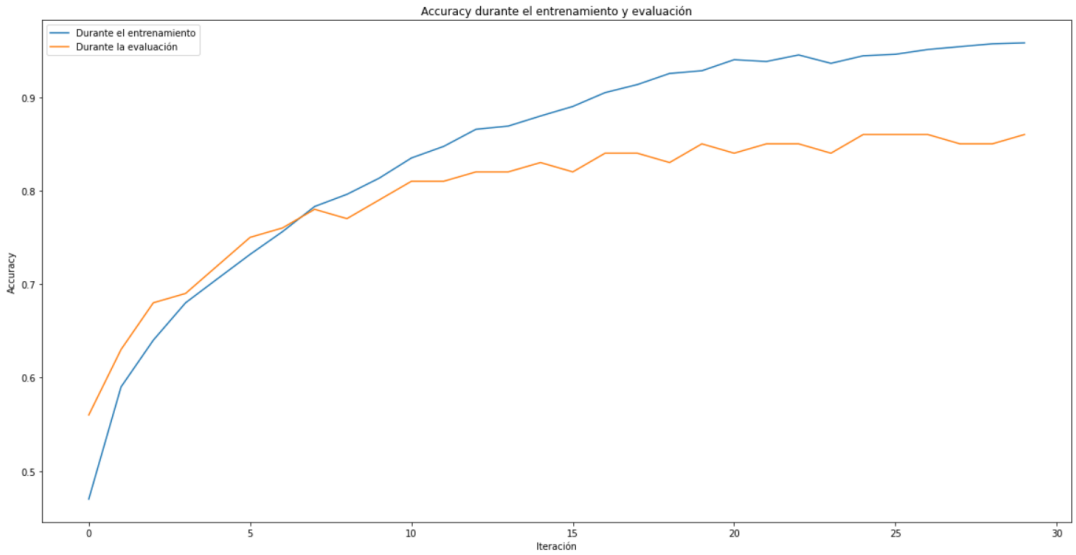


Figura 36. *Curvas de aprendizaje del algoritmo InceptionV3.*

En efecto, se aprecia que las curvas de los modelos Resnet18 y InceptionV3 resultan las más adecuadas, puesto que la precisión del modelo incrementa en cada iteración, mientras que el error del modelo decrece en cada ciclo. Asimismo, según Andrew Ng (2018) en las etapas finales de las curvas de aprendizaje se debe presentar una plataforma horizontal y una abertura mínima entre las curvas de entrenamiento y evaluación para indicar que el modelo es capaz de generalizar nueva información con un error mínimo. En efecto, a partir del gráfico pérdidas versus iteraciones del prototipo Resnet18 se observa que las curvas de evaluación y entrenamiento no idealizan una plataforma horizontal, sino existe una leve pendiente positiva para la primera y pendiente negativa para la segunda. Respecto al algoritmo InceptionV3, su gráfico de pérdidas adopta el escenario correcto, debido a que las curvas se asientan horizontalmente manteniendo constante una abertura mínima entre ellas. Finalmente, las gráficas de VGG-16 indican que este algoritmo no resulta adecuado para representar los datos obtenidos del sector construcción.

Asimismo, la Tabla 9 ilustra los valores de *accuracy* de cada modelo, así como el tiempo empleado durante la etapa de entrenamiento. Cabe resaltar que la variación entre los tiempos de entrenamiento se debe a que el servicio gratuito de *GoogleCollab* ofrecía

aleatoriamente diferentes tipos de procesadores gráficos (GPU). En efecto, el valor más alto de *accuracy* resultó en 85.79%, como era de esperarse, debido a que no se evaluó la etiqueta de mayor dificultad y la cantidad de datos procesados no fueron suficientes para contrarrestar la pérdida de información que generan los efectos de oclusión y truncado. En conclusión, considerando los resultados de la Tabla 9 y el análisis efectuado en las curvas de aprendizaje, esta investigación evidencia que los prototipos de clasificación de imágenes, específicamente InceptionV3, presentan un gran potencial para generalizar el reconocimiento de objetos de construcción civil bajo distintas condiciones visuales

Tabla 9. *Desempeño y tiempo de entrenamiento requerido para cada modelo.*

Modelo		Accuracy	Transfer Learning Accuracy	Tiempo de Entrenamiento
Año	Red neuronal			
2014	VGG16	47.12%	9.70%	14h 9m 6s
2015	Resnet18	81.88%	10.79%	**2h 48m 9s**
2015	**InceptionV3**	**85.79%**	11.25%	11h 1m 27s

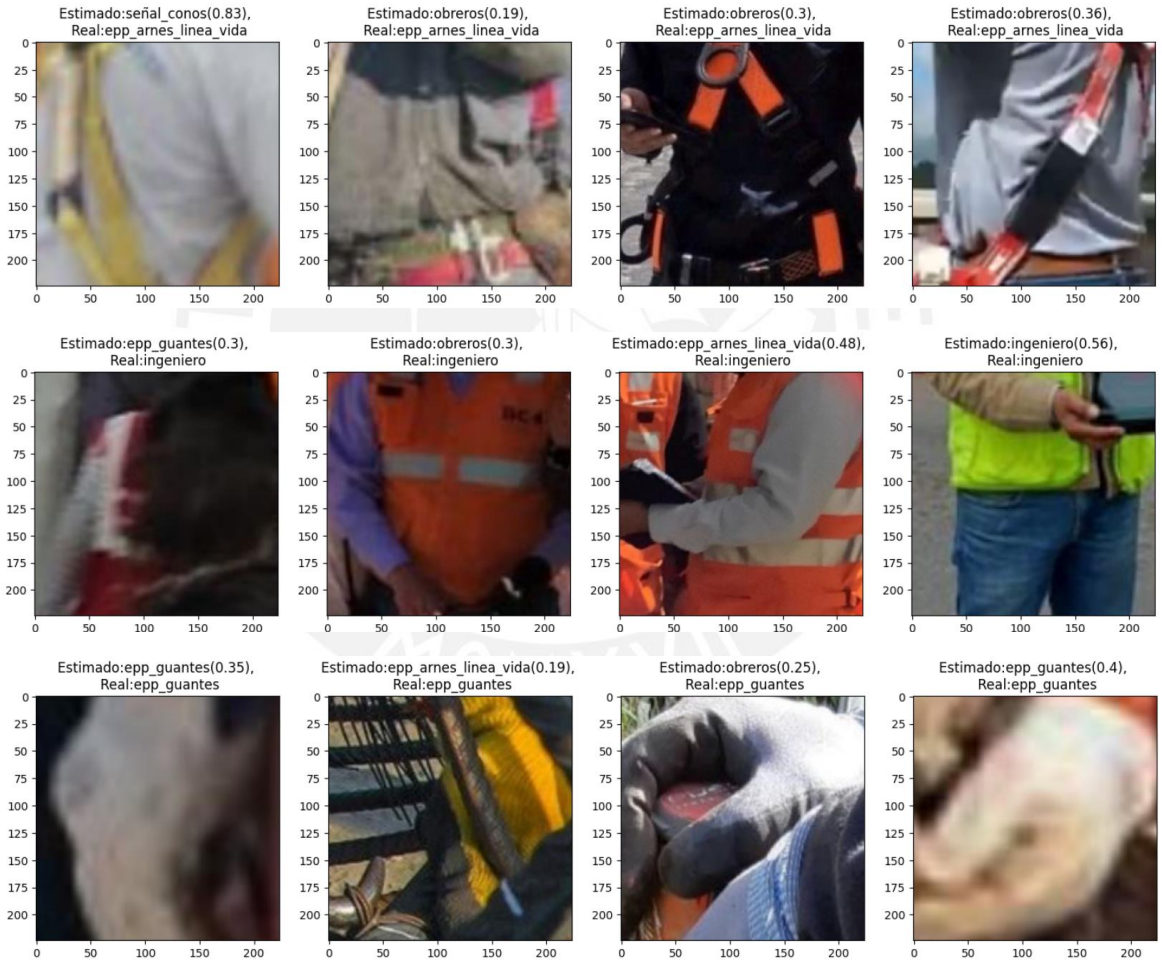
Adicionalmente, la columna de transferencia de aprendizaje representa el performance de cada modelo empleando los parámetros aprendidos en el set de datos *ImageNet* del año 2012, que consiste en 1000 clases de objetos distribuidos en 1.28 millones de imágenes. En efecto, se observa que el desempeño promedio reside en un valor de 10% aproximadamente. Por tanto, estos resultados corroboran que el set de datos propuesto presenta una complejidad 10 veces superior respecto a las condiciones visuales de *ImageNet.*

3.5 Evaluación de los modelos

En esta sección se emplea una data adicional para verificar la capacidad de las redes neuronales en generalizar nueva información. En efecto, la métrica seleccionada corresponde a uno de los indicadores más recurridos por la comunidad de visión computacional.

3.5.1 Matriz de confusiones

Por matriz de confusiones se entiende por una tabla que identifica el número de veces que el modelo se equivoca al clasificar objetos durante la clasificación. En efecto, las filas representan las clases verdaderas, mientras que las columnas aluden a los objetos estimados. Las siguientes figuras presentan algunos ejemplos de clasificación de objetos de protección, así como del staff de obra, realizados por los tres algoritmos estudiados y las matrices de confusión respectivas.



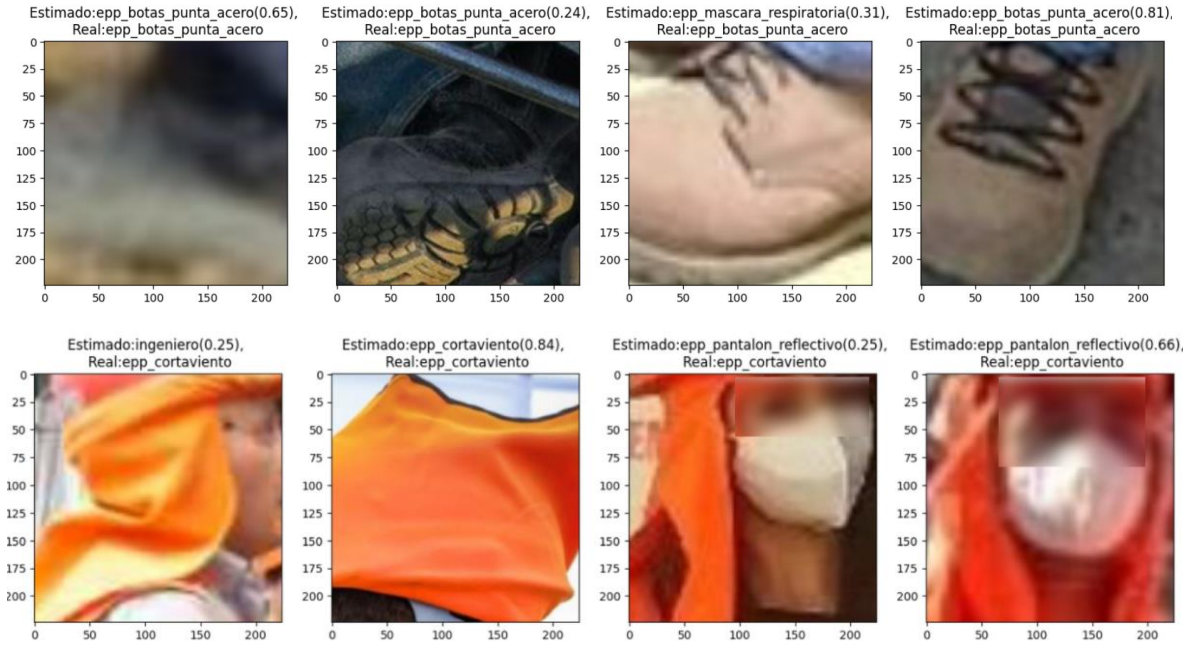


Figura 37. *Ejemplares estimados por el algoritmo VGG16.*

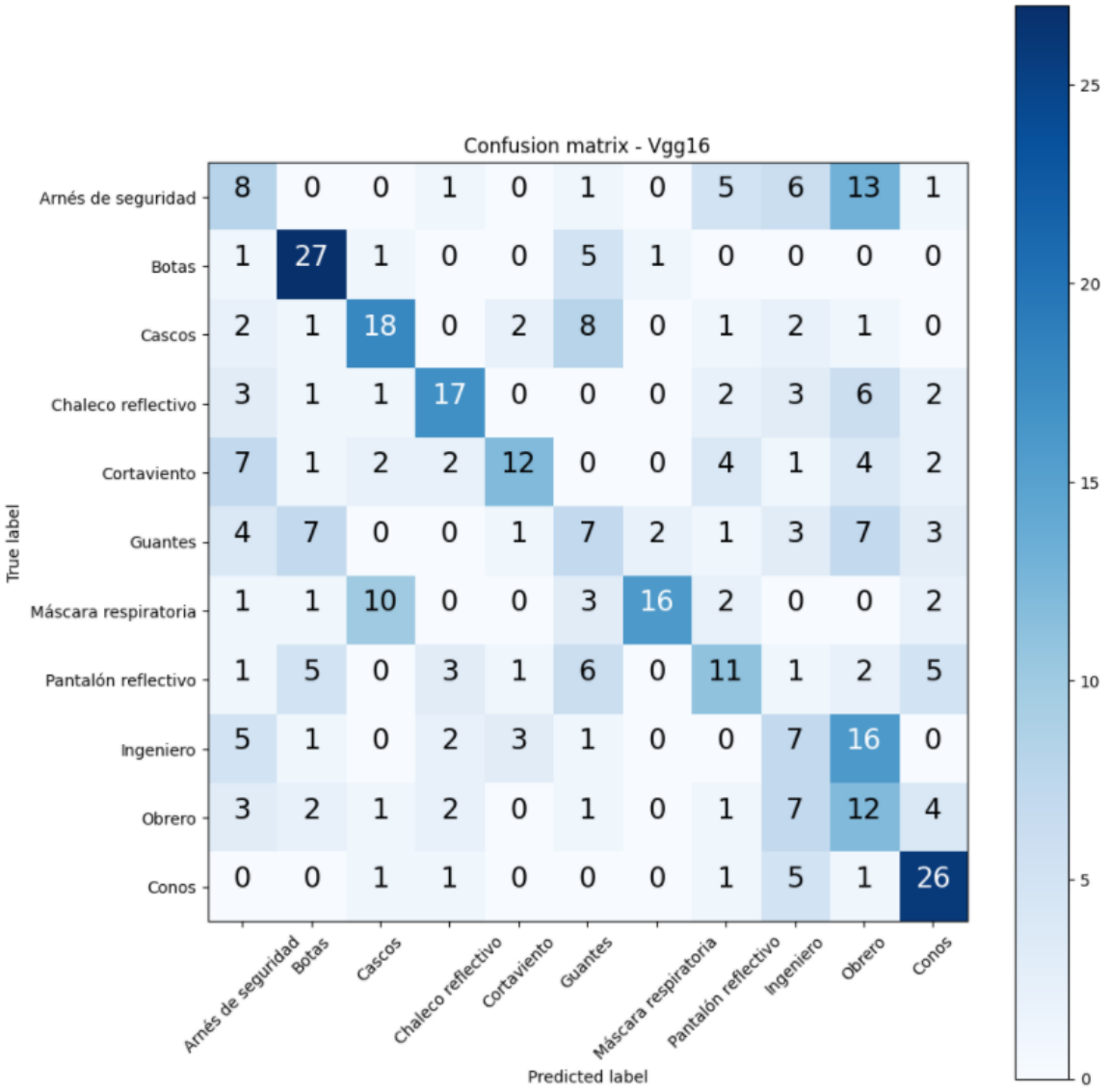
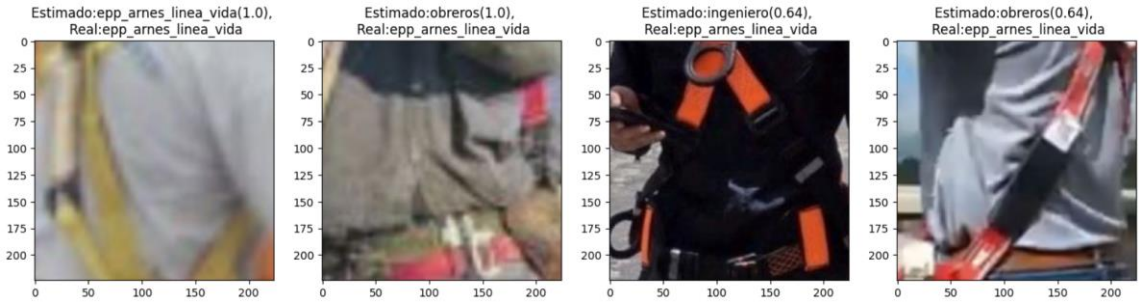


Figura 38. *Matriz de confusión en modelo VGG16.*



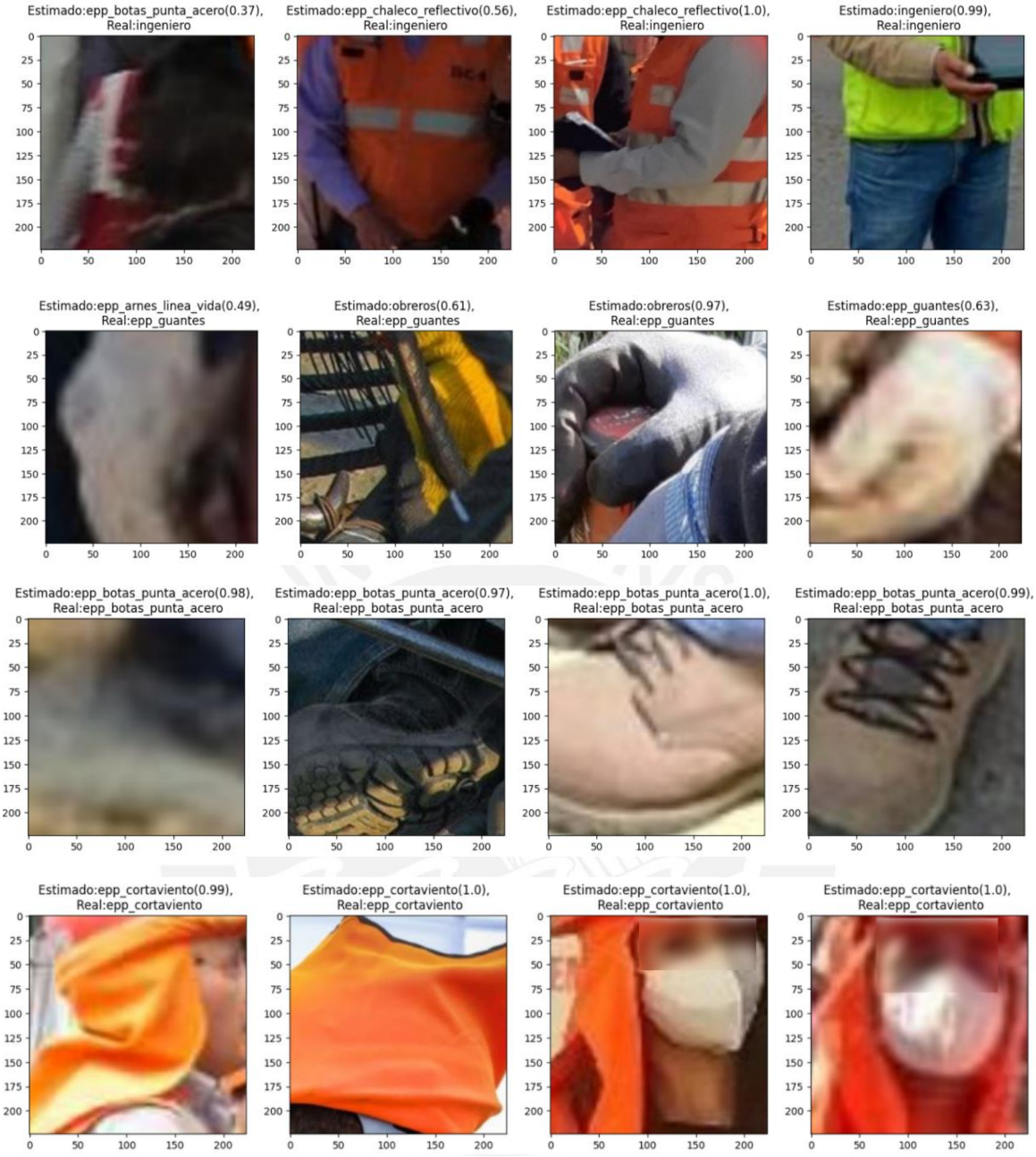


Figura 39. *Ejemplares estimados por el algoritmo Resnet18.*

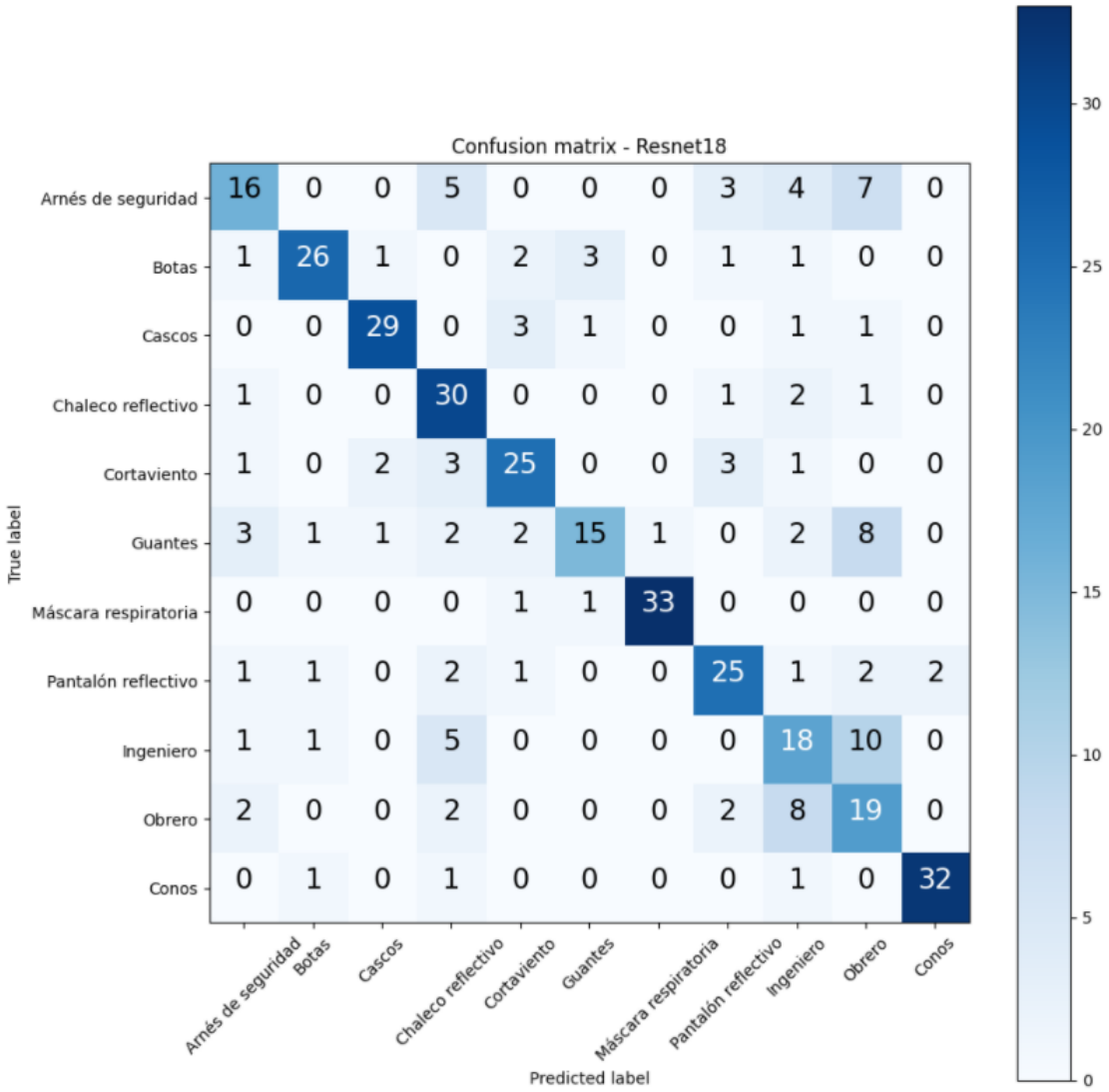


Figura 40. *Matriz de confusión en modelo Resnet18.*

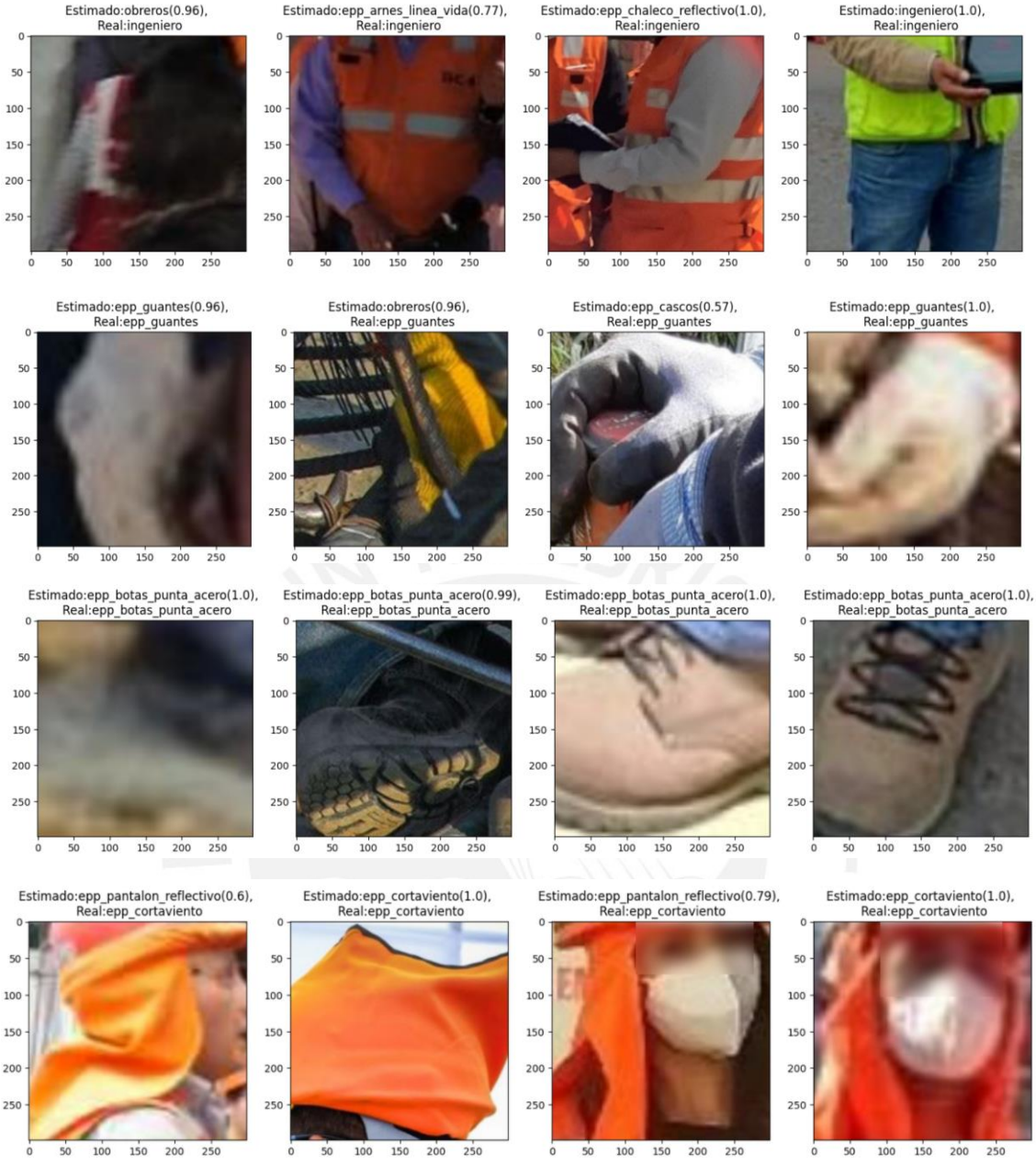


Figura 41. *Ejemplares estimados por el algoritmo InceptionV3.*

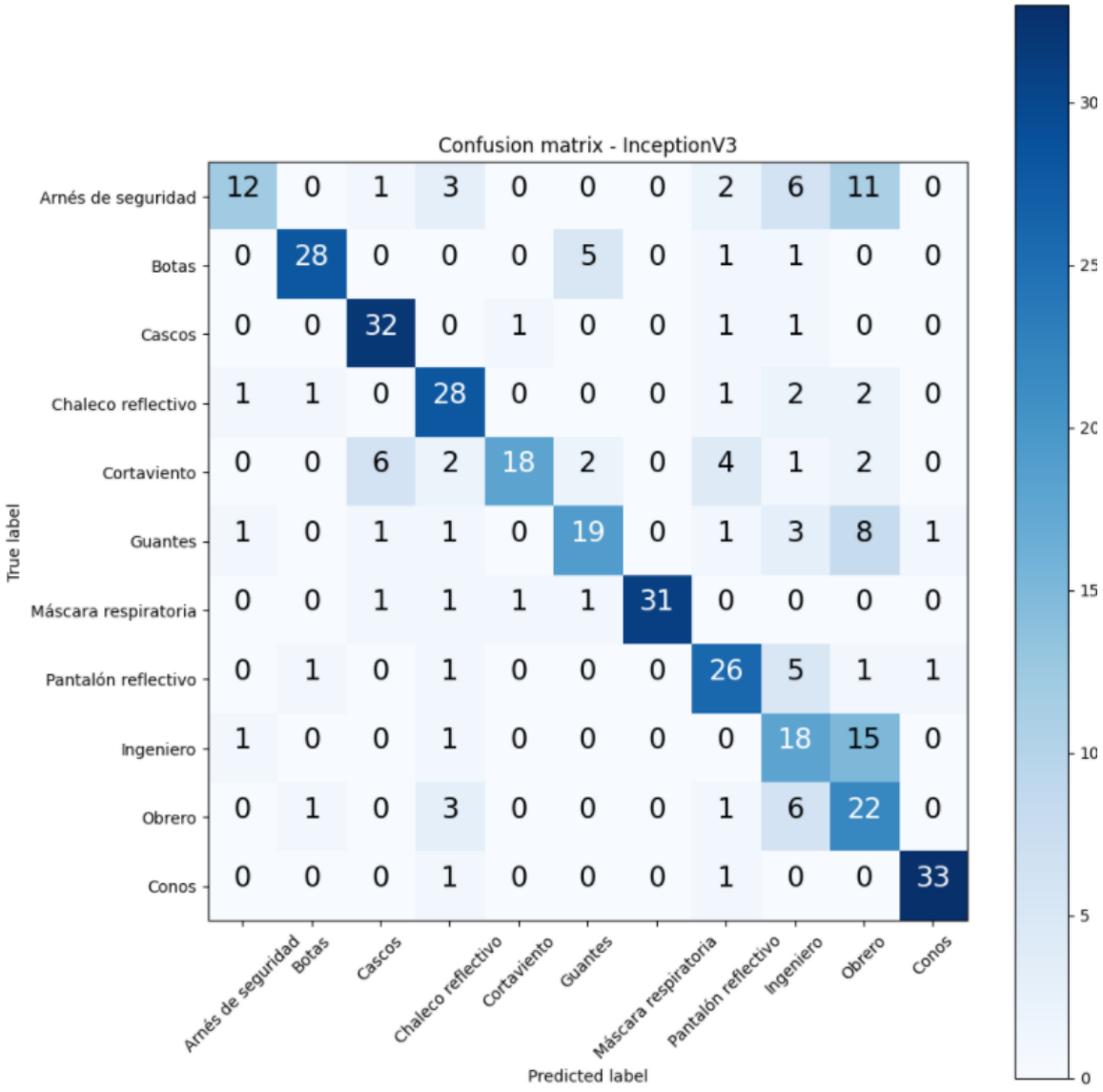


Figura 42. *Matriz de confusión en modelo InceptionV3.*

A través de las matrices de confusión, se puede observar que en general los dos últimos modelos clasifican mejor los 11 objetos de estudio. Además, en todos los gráficos resulta que las mayores confusiones se presentaron entre el arnés de seguridad con obrero, guantes con obrero e ingenieros junto a obreros. Por ejemplo, respecto a la primera relación, la matriz del modelo InceptionV3 se equivocó aproximadamente el 30% de las veces (11 ejemplares) en reconocer al arnés de seguridad como obrero, mientras que en la matriz de Resnet18 se presentaron errores el 20% de los casos (7 instancias). La causa más probable que produce este fenómeno está correlacionada con la selección de la herramienta de

etiquetación (etiquetas rectangulares). En efecto, la geometría de este equipo de seguridad resulta relativamente menor al área que comprende toda la zona de la *bounding box.* Incluso, al ser un elemento que se ubica en la zona pectoral y alrededor de la cintura, la etiqueta captura los píxeles del chaleco de seguridad, obrero, entre otros elementos en la misma ubicación. Por tanto, es de esperar que el modelo encuentre dificultad en reconocer este elemento cuando se está entregando información adicional y en exceso. En este contexto, como lo recomendó la literatura, la solución viable consiste en emplear etiquetas de segmentación por píxeles.

De igual manera, los guantes se tienden a confundir con obreros. Una posible explicación a este efecto se exhibe en las condiciones visuales capturadas en las imágenes. Por ejemplo, el set de datos propuesto presenta algunas instancias de obreros equipados sin guantes. Por tanto, si la textura o el material de este equipo de seguridad no es totalmente nítido (desgastado, empolvado, oscurecido) los caracteres y curvas del guante tienden a interpretarse como geometría de manos descubiertas del obrero. En este escenario, la solución se presenta en recolectar mayor data para que la máquina pueda diferenciar entre manos descubiertas y guantes de seguridad. Asimismo, la clase “ingenieros” se confunde con la entidad de obreros aproximadamente el 40% de los casos al emplear el modelo InceptionV3, mientras que existe una frecuencia del 30% de los eventos al procesar la data con Resnet18. Estos falsos positivos se producen debido a que los únicos caracteres que realizan la distinción entre estas dos clases son el pantalón reflectivo (por parte de los operarios), polo reflectivo, mamelucos y el color del casco. En esencia, en Figura 41, Figura 39 y Figura 37 se observan algunas imágenes donde no están presentes estos equipos de seguridad y, por lo tanto, existe cierto nivel de complejidad en diferenciar un obrero y un ingeniero.

4. CONCLUSIONES

El uso de equipos de protección colectiva y personal permite resguardar la vida de los operarios ante cualquier accidente que se presente en las obras de construcción. No obstante, en campo existen comportamientos inadecuados por parte del personal de obra, pues tienden a retirarse los equipos de seguridad, debido a la disconformidad que produce su peso, el cambio de temperatura corporal, entre otros factores. En esencia, actualmente el control de estas actitudes es exhaustivo pues involucra monitorear múltiples actividades proactivamente a lo largo de la jornada laboral. Este estudio propone evaluar la efectividad de la tecnología *deep learning* para automatizar la detección de estos equipos de seguridad. En esencia, se evaluaron tres algoritmos (VGG-16, Resnet-18 y Inception-V3) y se observó que los modelos más actuales tendieron a generalizar mejor la clasificación de los objetos de estudio en diferentes condiciones visuales. En efecto, los resultados del prototipo Inception-V3 alcanzaron valores de *accuracy* de 84% empleando un set de datos de escala regular y considerando una mayor variedad de equipos de seguridad respecto a la literatura. Este desempeño indica que las metodologías en aprendizaje profundo pueden contribuir a monitorear la seguridad del personal de obra, debido a que es posible incrementar el performance obtenido al emplear mayor data, modificando la estructura de la red neuronal (*dropout, batchnormalization,* incluir regularizadores, entre otras técnicas) o seleccionando un modelo más complejo. Además, actualmente existe la tarea de explorar las funcionalidades de *object tracking* y *object location* en monitorear los equipos de protección propuestos en tiempo real.

4.1 Contribuciones

Asimismo, existen dos contribuciones adicionales. En primer lugar, se realizó un breve listado sobre las investigaciones más recientes de la visión computacional realizadas en la industria de la construcción civil con el objetivo de orientar a otros proyectos a seleccionar

un tema de estudio, identificar los logros alcanzados, responder a las limitaciones encontradas y reconocer buenas prácticas. En segundo lugar, se elaboró una base de datos que comprende una variedad de condiciones visuales presentes en escenarios de construcción, incluidas cambios de iluminación, rango de vista, posturas individuales, oclusiones, aglomeraciones, calidad de la imagen, entre otros efectos. Por tanto, esta información contribuye con el desarrollo de modelos en visión computacional para identificar y, en un futuro trabajo, monitorear equipos de seguridad. Adicionalmente, a diferencia de la literatura este set de datos dispone de una mayor extensión de EPP's y EPC's, debido a que el enfoque de este proyecto es colaborar con futuras investigaciones para estandarizar un set de datos propio del sector construcción, además de servir de *benchmark* para comparar el desempeño entre diferentes modelos en visión computacional.

4.2 Limitaciones

En la presente investigación permanecen ciertas limitaciones. Por ejemplo, la literatura sugirió etiquetar determinados objetos mediante máscaras segmentadas frente a etiquetas rectangulares. En efecto, las matrices de confusiones corroboraron esta premisa, debido a que los modelos tendieron a confundir el reconocimiento del arnés de seguridad. Esto se debe a que la etiqueta rectangular no solo captura la geometría de este elemento, sino que incorpora información extra, como chaleco, pantalón o al mismo obrero, e incita a cometer errores de clasificación. Si bien esta nueva etiqueta significa una solución para contrarrestar este fenómeno, existe la contraparte de incrementar el tiempo en la etapa de etiquetación de imágenes y trabajar con un costo computacional mayor al momento de aplicar el modelo en tiempo real.

Asimismo, los algoritmos se analizaron sin considerar la etiqueta difícil, que representa un objeto con pérdida de información de más de 75%, debido a que no se dispuso de una

gran cantidad de ejemplares para lidiar con este fenómeno, aunque sí se estudiaron los efectos de oclusión y truncado. En efecto, estos dos últimos factores resultan ser la causa de no obtener una precisión perfecta o al menos superior al 90%. En este contexto, los beneficios de esta tecnología se pueden maximizar si se consideran las pautas, comentadas en el capítulo 2, en cuanto a la óptima planificación de cámaras en campo para capturar imágenes sin pérdida de información. De esta manera, se reduciría el trabajo exhaustivo de coleccionar más data de la necesaria para contrarrestar efectos de oclusiones y truncado.

Una desventaja de la tecnología clasificación de imágenes es que solo permite capturar un objeto en una imagen. Por lo tanto, resulta necesario incorporar estos algoritmos en una metodología que comprenda clasificación, detección de objetos, e, incluso, reconocimiento facial para clasificar múltiples objetos en una imagen y procesar la información en tiempo real de forma práctica. Sin embargo, a través de la literatura se comprendió que la implementación en campo involucra un costo adicional, debido a que se necesita adquirir un servidor con procesadores GPU, al menos 1 cámara de video y un ordenador que capture la transmisión en vivo. Por tanto, se debe promover el desarrollo de investigaciones que aspiren a obtener un performance cada vez mayor para que el costo de mantenimiento de equipos se contrarreste con el beneficio de controlar la seguridad de los operarios.

Por otro lado, los resultados de la presente investigación presentan algunas similitudes respecto a otros estudios basados en modelos CNN. Por ejemplo, Nath et al. (2020) evaluaron el clasificador VGG-16, Resnet-50 y Xception para identificar 2 equipos de seguridad (cascos y chalecos reflectivos), obteniendo desempeños de 78.2%, 77.8% y 76.8%, respectivamente. Estos resultados se lograron empleando 9305 ejemplares de obreros y equipos de protección. Asimismo, es importante mencionar que los autores optaron por incorporar capas de detección y localización a cada modelo para operar en tiempo real, obteniendo el mejor performance (64.2%) al emplear Resnet-50.

De igual manera, Wu et al. (2019) incorporaron el clasificador VGG16 en el detector *Single Shot Detector* (SSD) para el monitoreo de cascos de seguridad, alcanzando una precisión de 83.89%; Ding et al. (2018) incluyeron el clasificador Incepton-V3 en el algoritmo LSTM para identificar y monitorear 3 peligros en el trabajo con escaleras alcanzando una precisión de 92%; Kolar et al. (2018) aplicaron el clasificador VGG16 para identificar la presencia de barandas de seguridad en los bordes del abismo y luego monitorearlas mediante el modelo MLP, obteniendo un performance de 86%; entre otros casos de estudio. En esencia, el presente trabajo es superior a la literatura en cuanto a clasificar un mayor número de equipos de seguridad.

4.3 Recomendaciones

En un siguiente trabajo, se buscará expandir el set de datos propuesto dado que se evaluaron los modelos de visión computacional en un número específico de equipos de protección. En ese sentido, se considerarán las referencias citadas en el capítulo dos para determinar la localización óptima de los dispositivos de video y capturar imágenes de calidad. Además, se contempla la alternativa de adquirir servicios de etiquetación para acelerar el proceso de anotaciones.

Asimismo, posteriormente se buscará desarrollar clasificadores más robustos, debido a que una mayor precisión permitirá, a largo plazo, realizar un mayor número de acciones correctas empleando menor data (se disminuyen los esfuerzo en recolectar imágenes) y facilitar su aplicación en campo. En este segundo escenario, se requerirán de procesadores potentes, Nvidia CUDA, para habilitar la ejecución de los programas en tiempo real. Adicionalmente, se considerará integrar la tecnología de reconocimiento de rostros para identificar el obrero que infringe el uso del equipo de seguridad y realizar el seguimiento correspondiente.

En un futuro trabajo, se planea incorporar nuevas categorías de etiquetado que permitan reconocer objetos incorrectamente equipados, elementos desgastados, color de los cascos para diferenciar rangos entre el staff de obra, entre otras aplicaciones.

5. REFERENCIAS

Akinosho, T. D., Oyedele, L. O., Bilal, M., Ajayi, A. O., Delgado, M. D., Akinade, O. O., & Ahmed, A. A. (2020). Deep learning in the construction industry: A review of present status and future innovations. *Journal of Building Engineering*, *32*. https://doi.org/10.1016/j.jobe.2020.101827

Angah, O., & Chen, A. Y. (2020). Tracking multiple construction workers through deep learning and the gradient based method with re-matching based on multi-object tracking accuracy. *Automation in Construction*, *119*. https://doi.org/10.1016/j.autcon.2020.103308

Atha, D. J., & Jahanshahi, M. R. (2018). Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection. *Structural Health Monitoring*, *17*(5). https://doi.org/10.1177/1475921717737051

Brownlee, J. (2016). *Deep Learning With Python Develop Deep Learning Models On Theano And TensorFlow Using Keras* (Machine Learning Mastery, Ed.; 1.7).

Bügler, M., Borrmann, A., Ogunmakin, G., Vela, P. A., & Teizer, J. (2017). Fusion of Photogrammetry and Video Analysis for Productivity Assessment of Earthwork Processes. *Computer-Aided Civil and Infrastructure Engineering*, *32*(2). https://doi.org/10.1111/mice.12235

Bureau of Labor Statistics. (2020, December 16). *Table 3. Fatal occupational injuries for selected occupations, 2015-19*. U.S. Department of Labor. https://www.bls.gov/news.release/cfoi.t03.htm

Cai, J. (2020). *DATA-DRIVEN APPROACH TO HOLISTIC SITUATIONAL AWARENESS IN CONSTRUCTION SITE SAFETY MANAGEMENT*. https://doi.org/https://doi.org/10.25394/PGS.12412808.v1

Cai, J., Zhang, Y., & Cai, H. (2019). Two-step long short-term memory method for identifying construction activities through positional and attentional cues. *Automation in Construction*, *106*. https://doi.org/10.1016/j.autcon.2019.102886

Chen, C., Zhu, Z., & Hammad, A. (2020). Automated excavators activity recognition and productivity analysis from construction site surveillance videos. *Automation in Construction*, *110*. https://doi.org/10.1016/j.autcon.2019.103045

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*(5). https://doi.org/10.1109/TIE.2017.2764844

Chen, H., Luo, X., Zheng, Z., & Ke, J. (2019). A proactive workers' safety risk evaluation framework based on position and posture data fusion. *Automation in Construction*, *98*. https://doi.org/10.1016/j.autcon.2018.11.026

Chen, Z., & Jiang, C. (2018). Building occupancy modeling using generative adversarial network. *Energy and Buildings*, *174*. https://doi.org/10.1016/j.enbuild.2018.06.029

CLOUDFACTORY. (2020). *Image Annotation for Computer Vision A Guide to Labeling Visual Data for Your Machine Learning Project*. CLOUDFACTORY. https://www.cloudfactory.com/image-annotation-guide

Cognilytica. (2020, January 31). *Data Preparation & Labeling for AI 2020 | Cognilytica*. https://www.cognilytica.com/2020/01/31/data-preparation-labeling-for-ai-2020/

Dais, D., Bal, İ. E., Smyrou, E., & Sarhosis, V. (2021). Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning. *Automation in Construction*, *125*. https://doi.org/10.1016/j.autcon.2021.103606

Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. https://doi.org/10.1109/CVPR.2005.177

Ding, L., Fang, W., Luo, H., Love, P. E. D., Zhong, B., & Ouyang, X. (2018). A deep hybrid learning model to detect unsafe behavior: Integrating convolution neural networks and long short-term memory. *Automation in Construction*, *86*. https://doi.org/10.1016/j.autcon.2017.11.002

Dung, C. V., & Anh, L. D. (2019). Autonomous concrete crack detection using deep fully convolutional neural network. *Automation in Construction*, *99*. https://doi.org/10.1016/j.autcon.2018.11.028

Everingham, M., van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, *88*(2). https://doi.org/10.1007/s11263-009-0275-4

Fan, C., Xiao, F., & Zhao, Y. (2017). A short-term building cooling load prediction method using deep learning algorithms. *Applied Energy*, *195*. https://doi.org/10.1016/j.apenergy.2017.03.064

Fan, R., Bocus, M. J., Zhu, Y., Jiao, J., Wang, L., Ma, F., Cheng, S., & Liu, M. (2019, June). Road Crack Detection Using Deep Convolutional Neural Network and Adaptive Thresholding. *2019 IEEE Intelligent Vehicles Symposium (IV)*. https://doi.org/10.1109/IVS.2019.8814000

Fan, Z., Wu, Y., Lu, J., & Li, W. (2018). *Automatic Pavement Crack Detection Based on Structured Prediction with the Convolutional Neural Network*. http://arxiv.org/abs/1802.02208

Fang, Q., Li, H., Luo, X., Ding, L., Luo, H., & Li, C. (2018a). Computer vision aided inspection on falling prevention measures for steeplejacks in an aerial environment. *Automation in Construction*, *93*. https://doi.org/10.1016/j.autcon.2018.05.022

Fang, Q., Li, H., Luo, X., Ding, L., Luo, H., Rose, T. M., & An, W. (2018b). Detecting non-hardhat-use by a deep learning method from far-field surveillance videos. *Automation in Construction*, *85*. https://doi.org/10.1016/j.autcon.2017.09.018

Fang, Q., Li, H., Luo, X., Ding, L., Rose, T. M., An, W., & Yu, Y. (2018). A deep learning-based method for detecting non-certified work on construction sites. *Advanced Engineering Informatics*, *35*. https://doi.org/10.1016/j.aei.2018.01.001

Fang, W., Ding, L., Love, P. E. D., Luo, H., Li, H., Peña-Mora, F., Zhong, B., & Zhou, C. (2020). Computer vision applications in construction safety assurance. *Automation in Construction*, *110*. https://doi.org/10.1016/j.autcon.2019.103013

Fang, W., Ding, L., Luo, H., & Love, P. E. D. (2018). Falls from heights: A computer vision-based approach for safety harness detection. *Automation in Construction*, *91*. https://doi.org/10.1016/j.autcon.2018.02.018

Fang, W., Ding, L., Zhong, B., Love, P. E. D., & Luo, H. (2018). Automated detection of workers and heavy equipment on construction sites: A convolutional neural network approach. *Advanced Engineering Informatics*, *37*. https://doi.org/10.1016/j.aei.2018.05.003

Fang, W., Zhong, B., Zhao, N., Love, P. E. D., Luo, H., Xue, J., & Xu, S. (2019). A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network. *Advanced Engineering Informatics*, *39*. https://doi.org/10.1016/j.aei.2018.12.005

Gard, N. A., Chen, J., Tang, P., & Yilmaz, A. (2018). Deep learning and anthropometric plane based workflow monitoring by detecting and tracking workers. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLII–1*. https://doi.org/10.5194/isprs-archives-XLII-1-149-2018

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014, June). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2014.81

Gollapudi, S. (2019). Learn Computer Vision Using OpenCV. In *Learn Computer Vision Using OpenCV*. Apress. https://doi.org/10.1007/978-1-4842-4261-2

Golparvar-Fard, M., Heydarian, A., & Niebles, J. C. (2013). Vision-based action recognition of earthmoving equipment using spatio-temporal features and support vector machine classifiers. *Advanced Engineering Informatics*, *27*(4). https://doi.org/10.1016/j.aei.2013.09.001

Gong, J., & Caldas, C. H. (2010). Computer Vision-Based Video Interpretation Model for Automated Productivity Analysis of Construction Operations. *Journal of Computing in Civil Engineering*, *24*(3). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000027

Gonzalez, R. C., Woods, R. E., Partridge, J., & Bai, J. (2018). *Digital Image Processing*. www.pearsoned.com/

Guo, Y., Xu, Y., & Li, S. (2020). Dense construction vehicle detection based on orientation-aware feature fusion convolutional neural network. *Automation in Construction*, *112*. https://doi.org/10.1016/j.autcon.2020.103124

Han, K. K., Cline, D., & Golparvar-Fard, M. (2015). Formalized knowledge of construction sequencing for visual monitoring of work-in-progress via incomplete point clouds and low-LoD 4D BIMs. *Advanced Engineering Informatics*, *29*(4). https://doi.org/10.1016/j.aei.2015.10.006

Hoskere, V., Narazaki, Y., Hoang, T. A., & Spencer, B. F. (2020). MaDnet: multi-task semantic segmentation of multiple types of structural materials and damage in images of civil infrastructure. *Journal of Civil Structural Health Monitoring*, *10*(5). https://doi.org/10.1007/s13349-020-00409-0

Huang, X., & Hinze, J. (2003). Analysis of Construction Worker Fall Accidents. *Journal of Construction Engineering and Management*, *129*(3). https://doi.org/10.1061/(ASCE)0733-9364(2003)129:3(262)

Hubel, D., & Wiesel, T. (2004). *Brain and Visual Perception*. OUP USA.

Jeelani, I., Asadi, K., Ramshankar, H., Han, K., & Albert, A. (2021). Real-time vision-based worker localization & hazard detection for construction. *Automation in Construction*, *121*. https://doi.org/10.1016/j.autcon.2020.103448

Joshi, K., & Patel, M. I. (2020). Recent advances in local feature detector and descriptor: a literature survey. *International Journal of Multimedia Information Retrieval*, *9*(4), 231–247. https://doi.org/10.1007/s13735-020-00200-3

Kalfarisi, R., Wu, Z. Y., & Soh, K. (2020). Crack Detection and Segmentation Using Deep Learning with 3D Reality Mesh Model for Quantitative Assessment and Integrated Visualization. *Journal of Computing in Civil Engineering*, *34*(3). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000890

KEELE K. D. (1955). *Leonardo da Vinci on vision: Vol. 48(5)*. Proceedings of the Royal Society of Medicine.

Khanday, N. Y., & Sofi, S. A. (2021). Taxonomy, state-of-the-art, challenges and applications of visual understanding: A review. *Computer Science Review*, *40*. https://doi.org/10.1016/j.cosrev.2021.100374

Kim, J., & Chi, S. (2019). Action recognition of earthmoving excavators based on sequential pattern analysis of visual features and operation cycles. *Automation in Construction*, *104*. https://doi.org/10.1016/j.autcon.2019.03.025

Kim, J., Ham, Y., Chung, Y., & Chi, S. (2019). Systematic Camera Placement Framework for Operation-Level Visual Monitoring on Construction Jobsites. *Journal of Construction Engineering and Management*, *145*(4). https://doi.org/10.1061/(ASCE)CO.1943-7862.0001636

Kingma, D. P., & Lei Ba, J. (2017). *ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION*.

Kolar, Z., Chen, H., & Luo, X. (2018). Transfer learning and deep convolutional neural networks for safety guardrail detection in 2D images. *Automation in Construction*, *89*. https://doi.org/10.1016/j.autcon.2018.01.003

Konstantinou, E. (2018). *Vision-Based Construction Worker Task Productivity Monitoring*. https://doi.org/https://doi.org/10.17863/CAM.20613

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6). https://doi.org/10.1145/3065386

Kumar, S. S., Abraham, D. M., Jahanshahi, M. R., Iseley, T., & Starr, J. (2018). Automated defect classification in sewer closed circuit television inspections using deep convolutional neural networks. *Automation in Construction*, *91*. https://doi.org/10.1016/j.autcon.2018.03.028

Lawrence, R. (1963). *Machine Perception of three-dimensional solids*. http://hdl.handle.net/1721.1/11589

Li, D., Cong, A., & Guo, S. (2019). Sewer damage detection from imbalanced CCTV inspection data using deep convolutional neural networks with hierarchical classification. *Automation in Construction*, *101*. https://doi.org/10.1016/j.autcon.2019.01.017

Li, H., Li, X., Luo, X., & Siebert, J. (2017). Investigation of the causality patterns of non-helmet use behavior of construction workers. *Automation in Construction*, *80*. https://doi.org/10.1016/j.autcon.2017.02.006

Li, S., Zhao, X., & Zhou, G. (2019). Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network. *Computer-Aided Civil and Infrastructure Engineering*, *34*(7). https://doi.org/10.1111/mice.12433

Liu, M., Han, S., & Lee, S. (2017, June 22). Potential of Convolutional Neural Network-Based 2D Human Pose Estimation for On-Site Activity Analysis of Construction Workers. *Computing in Civil Engineering 2017*. https://doi.org/10.1061/9780784480847.018

Liu, Y., Yao, J., Lu, X., Xie, R., & Li, L. (2019). DeepCrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing*, *338*. https://doi.org/10.1016/j.neucom.2019.01.036

Luo, H., Wang, M., Wong, P. K.-Y., & Cheng, J. C. P. (2020). Full body pose estimation of construction equipment using computer vision and deep learning techniques. *Automation in Construction*, *110*. https://doi.org/10.1016/j.autcon.2019.103016

Luo, H., Xiong, C., Fang, W., Love, P. E. D., Zhang, B., & Ouyang, X. (2018). Convolutional neural networks: Computer vision-based workforce activity assessment in construction. *Automation in Construction*, *94*. https://doi.org/10.1016/j.autcon.2018.06.007

Luo, X., Li, H., Cao, D., Dai, F., Seo, J., & Lee, S. (2018). Recognizing Diverse Construction Activities in Site Images via Relevance Networks of Construction-Related Objects Detected by Convolutional Neural Networks. *Journal of Computing in Civil Engineering*, *32*(3). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000756

Luo, X., Li, H., Cao, D., Yu, Y., Yang, X., & Huang, T. (2018). Towards efficient and objective work sampling: Recognizing workers' activities in site surveillance videos with two-stream convolutional networks. *Automation in Construction*, *94*. https://doi.org/10.1016/j.autcon.2018.07.011

Luo, X., Li, H., Yang, X., Yu, Y., & Cao, D. (2019). Capturing and Understanding Workers' Activities in Far-Field Surveillance Videos with Deep Action Recognition and Bayesian Nonparametric Learning. *Computer-Aided Civil and Infrastructure Engineering*, *34*(4). https://doi.org/10.1111/mice.12419

McKinsey. (2016, June 24). *Imagining construction's digital future | McKinsey*. https://www.mckinsey.com/business-functions/operations/our-insights/imagining-constructions-digital-future

Ministerio de Trabajo y Promoción del Empleo. (2019). *2019 Anuario ESTADÍSTICO SECTORIAL*. https://cdn.www.gob.pe/uploads/document/file/920578/ANUARIO_2019_.pdf

Mneymneh, B. E., Abbas, M., & Khoury, H. (2017). Automated Hardhat Detection for Construction Safety Applications. *Procedia Engineering*, *196*. https://doi.org/10.1016/j.proeng.2017.08.022

Nath, N. D., Behzadan, A. H., & Paal, S. G. (2020). Deep learning for site safety: Real-time detection of personal protective equipment. *Automation in Construction*, *112*. https://doi.org/10.1016/j.autcon.2020.103085

Nnedinma Umeokafor, Kostis Evaggelinos, Shaun Lundy, David Isaac, & Stuart Allan. (2014). The Pattern of Occupational Accidents, Injuries, Accident Causal Factors and Intervention in Nigerian Factories. *Developing Country Studies*, *4*.

Parker, A. (2004). *In The Blink Of An Eye*. Basic Books.

Pučko, Z., Šuman, N., & Rebolj, D. (2018). Automated continuous construction progress monitoring using multiple workplace real time 3D scans. *Advanced Engineering Informatics*, *38*. https://doi.org/10.1016/j.aei.2018.06.001

Rahman, A., & Smith, A. D. (2018). Predicting heating demand and sizing a stratified thermal storage tank using deep learning algorithms. *Applied Energy*, *228*. https://doi.org/10.1016/j.apenergy.2018.06.064

Rahman, A., Srikumar, V., & Smith, A. D. (2018). Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks. *Applied Energy*, *212*. https://doi.org/10.1016/j.apenergy.2017.12.051

Rashid, K. M., & Louis, J. (2019). Times-series data augmentation and deep learning for construction equipment activity recognition. *Advanced Engineering Informatics*, *42*. https://doi.org/10.1016/j.aei.2019.100944

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(6). https://doi.org/10.1109/TPAMI.2016.2577031

Roberts, D., & Golparvar-Fard, M. (2019). End-to-end vision-based detection, tracking and activity analysis of earthmoving equipment filmed at ground level. *Automation in Construction*, *105*. https://doi.org/10.1016/j.autcon.2019.04.006

Rosebrock, A. (2016). *Practical Python and OpenCV: An Introductory, Example Driven Guide to Image Processing and Computer Vision 3rd Edition* (A. Rosebrock, Ed.; 3rd ed.). pyimagesearch. https://www.pyimagesearch.com/practical-

Rosebrock, A. (2017a). *Deep Learning for Computer Vision with Python* (1st ed.). pyimagesearch.

Rosebrock, A. (2017b). *Deep Learning for Computer Vision with Python Starter Bundle 1st Edition (1.1.0)* (1st ed.). PyImageSearch.

Rubio, J. J., Kashiwa, T., Laiteerapong, T., Deng, W., Nagai, K., Escalera, S., Nakayama, K., Matsuo, Y., & Prendinger, H. (2019). Multi-class structural damage segmentation using fully convolutional networks. *Computers in Industry*, *112*. https://doi.org/10.1016/j.compind.2019.08.002

Satya, M. (2016, November 14). *Image Recognition and Object Detection : Part 1 | Learn OpenCV*. https://learnopencv.com/image-recognition-and-object-detection-part1/

Seo, J., Han, S., Lee, S., & Kim, H. (2015). Computer vision techniques for construction safety and health monitoring. *Advanced Engineering Informatics*, *29*(2). https://doi.org/10.1016/j.aei.2015.02.001

Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, *27*(3). https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Shrestha, K., Shrestha, P. P., Bajracharya, D., & Yfantis, E. A. (2015). Hard-Hat Detection for Construction Safety Visualization. *Journal of Construction Engineering*, *2015*. https://doi.org/10.1155/2015/721380

Siddula, M., Dai, F., Ye, Y., & Fan, J. (2016). Unsupervised Feature Learning for Objects of Interest Detection in Cluttered Construction Roof Site Images. *Procedia Engineering*, *145*. https://doi.org/10.1016/j.proeng.2016.04.010

Singaravel, S., Suykens, J., & Geyer, P. (2018). Deep-learning neural-network architectures and methods: Using component-based models in building-design energy prediction. *Advanced Engineering Informatics*, *38*. https://doi.org/10.1016/j.aei.2018.06.004

Slaton, T., Hernandez, C., & Akhavian, R. (2020). Construction activity recognition with convolutional recurrent networks. *Automation in Construction*, *113*. https://doi.org/10.1016/j.autcon.2020.103138

Son, H., Kim, C., & Kwon Cho, Y. (2017). Automated Schedule Updates Using As-Built Data and a 4D Building Information Model. *Journal of Management in Engineering*, *33*(4). https://doi.org/10.1061/(ASCE)ME.1943-5479.0000528

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). *Rethinking the Inception Architecture for Computer Vision*.

Szeliski, R. (2021). *Computer Vision: Algorithms and Applications 2nd Edition* (2nd ed.). Springer. http://szeliski.org/Book/,

Taha, A. A., & Hanbury, A. (2015). Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC Medical Imaging*, *15*(1). https://doi.org/10.1186/s12880-015-0068-x

Tajeen, H., & Zhu, Z. (2014). Image dataset development for measuring construction equipment recognition performance. *Automation in Construction*, *48*. https://doi.org/10.1016/j.autcon.2014.07.006

Turkan, Y., Bosché, F., Haas, C. T., & Haas, R. (2013). Toward Automated Earned Value Tracking Using 3D Imaging Tools. *Journal of Construction Engineering and Management*, *139*(4). https://doi.org/10.1061/(ASCE)CO.1943-7862.0000629

Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. https://doi.org/10.1109/CVPR.2001.990517

Wu, H., & Zhao, J. (2018). An intelligent vision-based approach for helmet identification for work safety. *Computers in Industry*, *100*. https://doi.org/10.1016/j.compind.2018.03.037

Wu, J., Cai, N., Chen, W., Wang, H., & Wang, G. (2019). Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset. *Automation in Construction*, *106*. https://doi.org/10.1016/j.autcon.2019.102894

Xiao, B., & Kang, S.-C. (2020). *Development of an Image Data Set of Construction Machines for Deep Learning Object Detection*. https://doi.org/10.1061/(ASCE)CP.1943

Xiao, B., & Zhu, Z. (2018). Two-Dimensional Visual Tracking in Construction Scenarios: A Comparative Study. *Journal of Computing in Civil Engineering*, *32*(3). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000738

Xie, Z., Liu, H., Li, Z., & He, Y. (2018, December). A convolutional neural network based approach towards real-time hard hat detection. *2018 IEEE International Conference on Progress in Informatics and Computing (PIC)*. https://doi.org/10.1109/PIC.2018.8706269

Xu, S., Wang, J., Shou, W., Ngo, T., Sadick, A.-M., & Wang, X. (2020). Computer Vision Techniques in Construction: A Critical Review. *Archives of Computational Methods in Engineering*. https://doi.org/10.1007/s11831-020-09504-3

Xu, Y., Li, S., Zhang, D., Jin, Y., Zhang, F., Li, N., & Li, H. (2018). Identification framework for cracks on a steel structure surface by a restricted Boltzmann machines algorithm based on consumer-grade camera images. *Structural Control and Health Monitoring*, *25*(2). https://doi.org/10.1002/stc.2075

Yang, J., Shi, Z., & Wu, Z. (2016). Vision-based action recognition of construction workers using dense trajectories. *Advanced Engineering Informatics*, *30*(3). https://doi.org/10.1016/j.aei.2016.04.009

Yang, K., Ahn, C. R., & Kim, H. (2020). Deep learning-based classification of work-related physical load levels in construction. *Advanced Engineering Informatics*, *45*. https://doi.org/10.1016/j.aei.2020.101104

Yang, X., Li, H., Yu, Y., Luo, X., Huang, T., & Yang, X. (2018). Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network. *Computer-Aided Civil and Infrastructure Engineering*, *33*(12). https://doi.org/10.1111/mice.12412

Yao, Y. (2018). *Towards Automatic Construction of Diverse, High-quality Image Dataset*.

Yu, Y., Li, H., Yang, X., Kong, L., Luo, X., & Wong, A. Y. L. (2019). An automatic and non-invasive physical fatigue assessment method for construction workers. *Automation in Construction*, *103*. https://doi.org/10.1016/j.autcon.2019.02.020

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). *Dive into Deep Learning*.

Zhang, H., Yan, X., & Li, H. (2018). Ergonomic posture recognition using 3D view-invariant features from single ordinary camera. *Automation in Construction*, *94*. https://doi.org/10.1016/j.autcon.2018.05.033

Zhu, Z., & Brilakis, I. (2010). Concrete Column Recognition in Images and Videos. *Journal of Computing in Civil Engineering*, *24*(6). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000053