

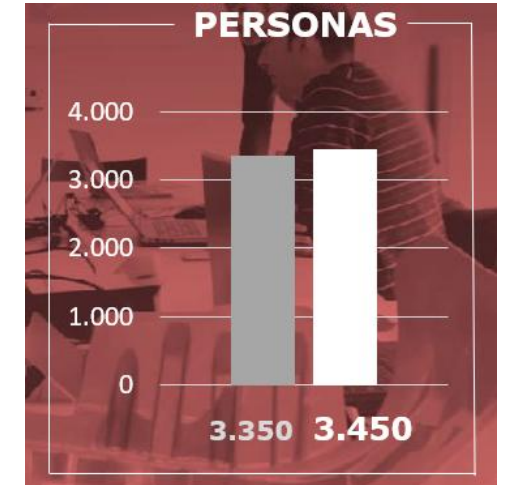
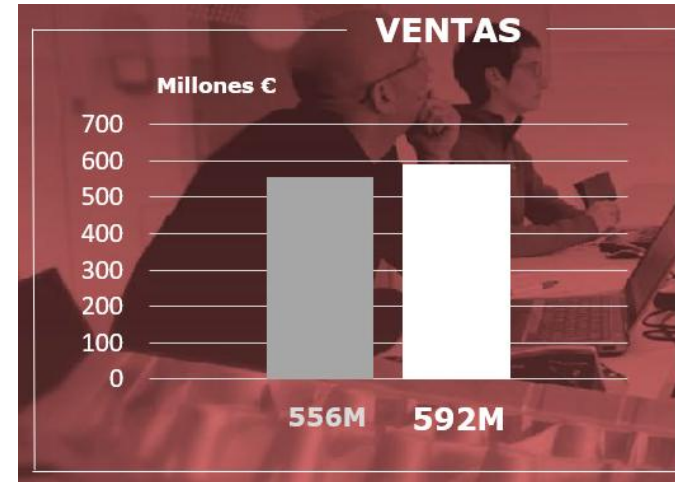
DELICADA:

Revolucionando el Control de Calidad 100% complementándolo con IA

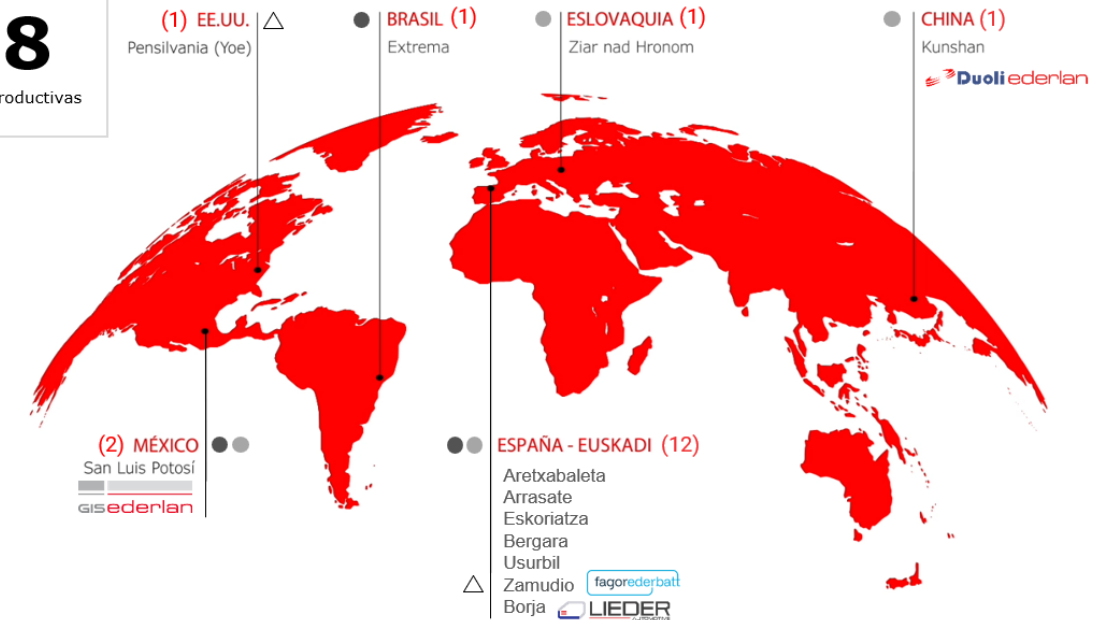
LÍDERES GLOBALES EN AUTOMOCIÓN

EDERTEK

fagorederlandgroup



18
Plantas productivas



EXPERTOS EN FUNDICIÓN

EDERTEK

fagoredenlangroup



FUNDICIÓN HIERRO & MECANIZADO

190K
Tn/año
fundidas

20M
Un./año
mecanizadas

8 plantas
6 EUROPA
1 BRASIL
1 MEXICO



ALUMINIO HPDC & MECANIZADO

35K
Tn/año
fundidas

4M
Un./año
mecanizadas

3 plantas
3 EUROPA

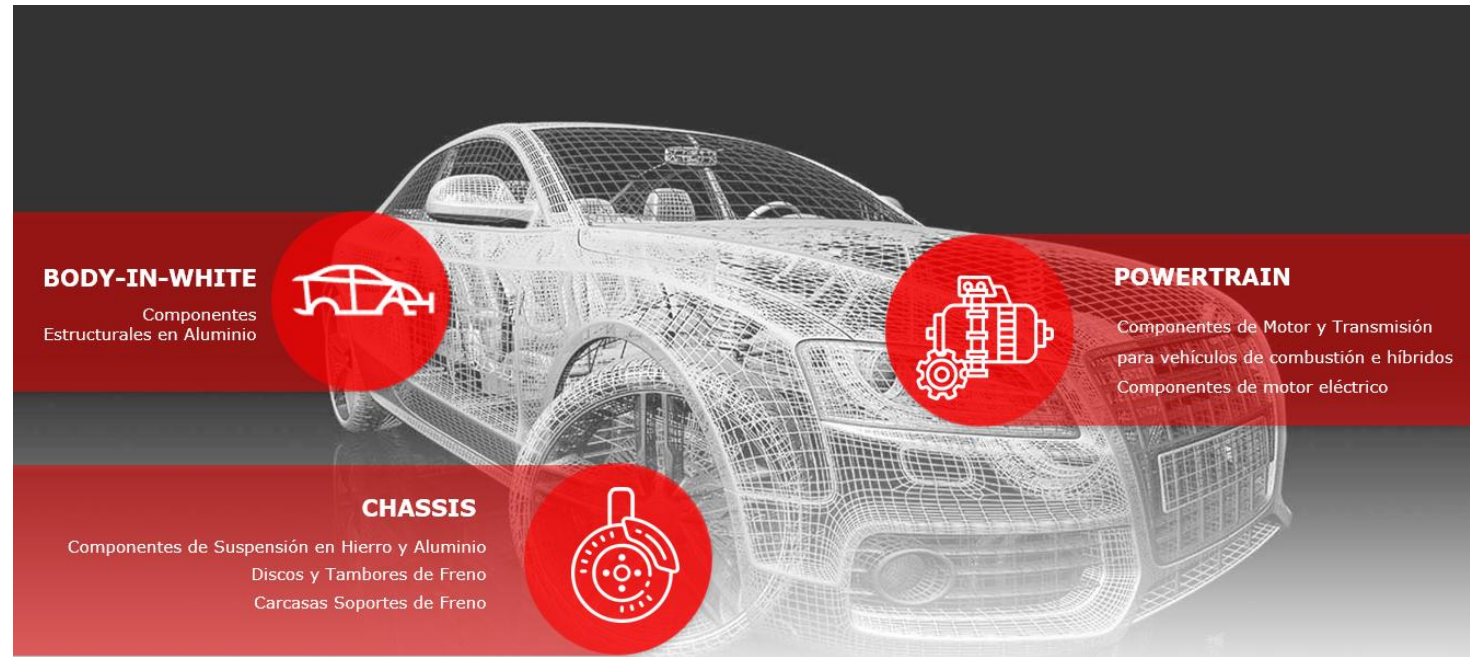


ALUMINIO LPDC & MECANIZADO

30K
Tn/año
fundidas

9M
Un./año
mecanizadas

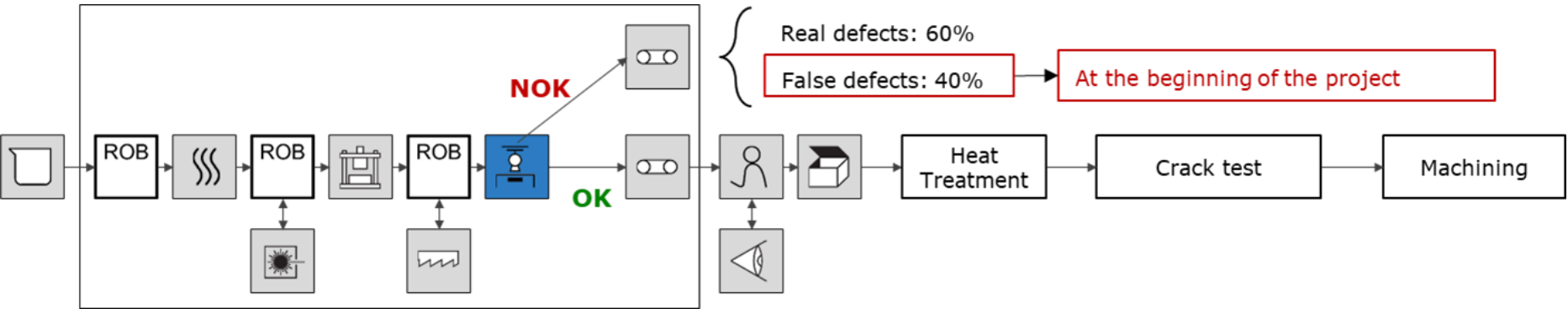
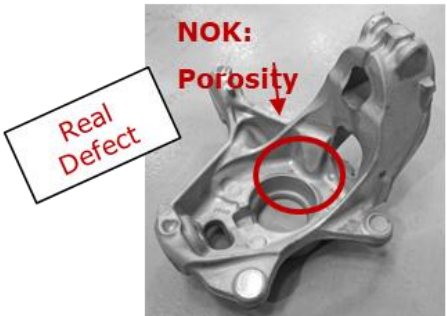
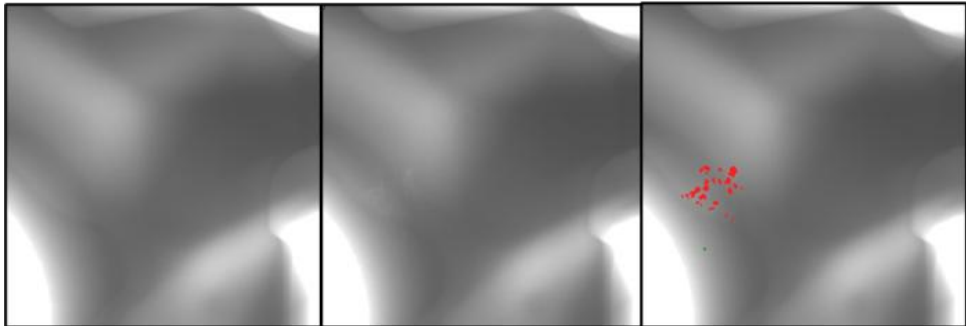
4 plantas
2 EUROPA
1 CHINA
1 MÉXICO



COMPONENTES DE SEGURIDAD

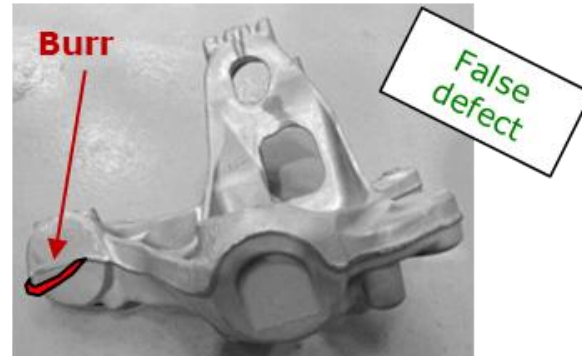
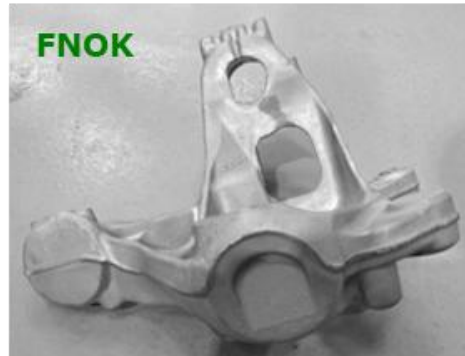
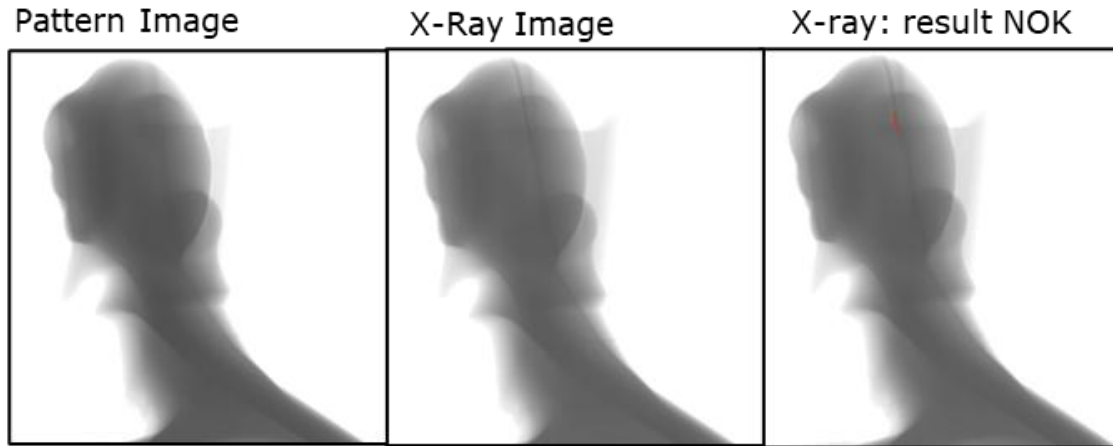
CONTROL 100% RAYOS X

Pattern Image X-Ray Image X-ray: result NOK



COMPONENTES DE SEGURIDAD

CONTROL 100% RAYOS X



Parts that, although be good, X-ray classifies as defective.

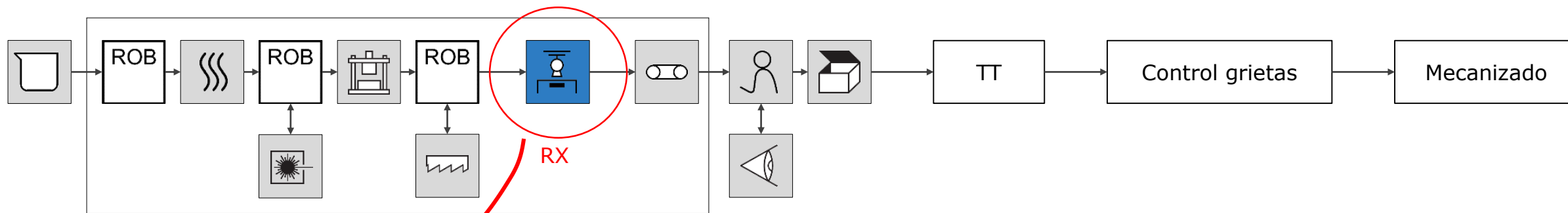


COMPONENTES DE SEGURIDAD

Objetivo:

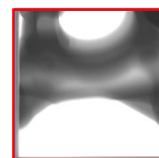
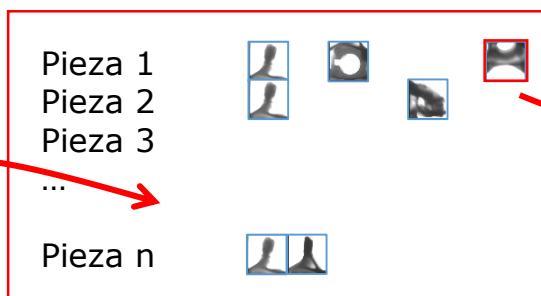
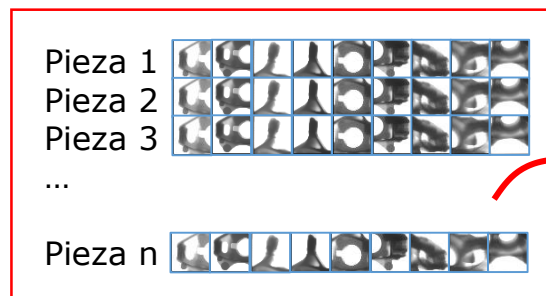
Aplicar técnicas de Inteligencia Artificial “Deep learning” en el proceso de detección de piezas defectuosas mediante rayos X que permita mejorar el control de calidad de las piezas de aluminio fabricadas por LPDC y evitar falsos defectivos.





Todas las imágenes de RX:

Imágenes NOK de RX:



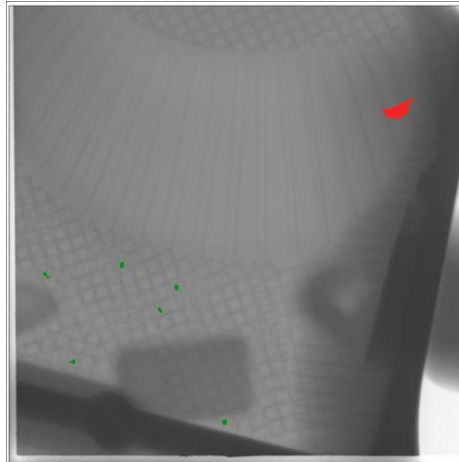
1º paso: Se etiqueta cada imagen NOK de RX (NOK/FNOK) para crear el dataset.

Entradas:
Cada imagen NOK de RX

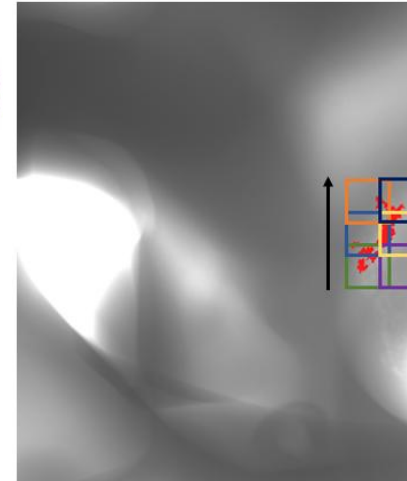
Modelo "deep learning"

Salidas:
NOK (defectivo)
FNOK (falso defectivo)

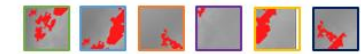
2º paso: Se entrenan diferentes algoritmos de IA hasta encontrar un modelo que sepa discernir entre defectos reales y falsos defectos.



*Imagen con defecto
marcado en rojo por
programa de RX*



*Pre-procesamiento
de la imagen*

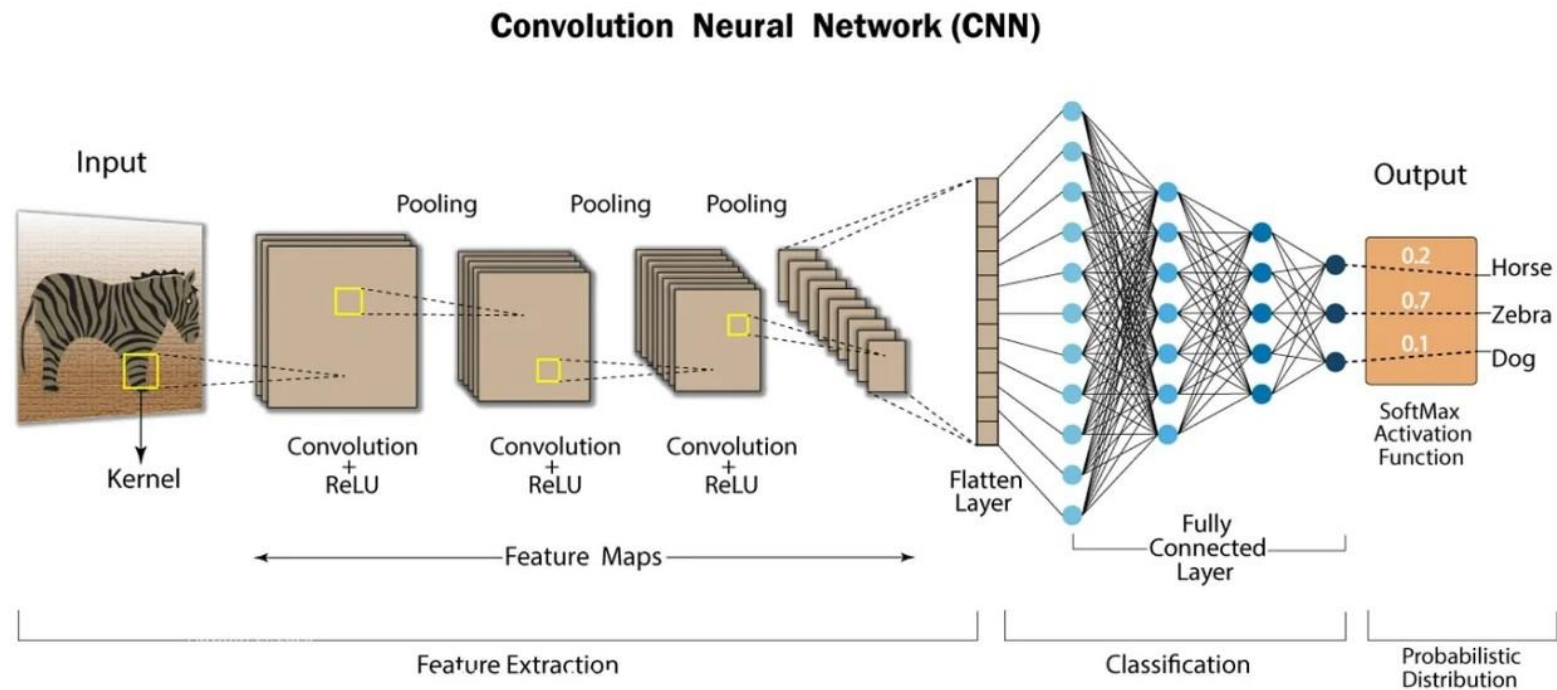


*Imágenes de
entrada para el
modelo*

La **imagen original de RX** es **demasiado grande** para el algoritmo y necesitaría **mucho tiempo** de **procesamiento**.

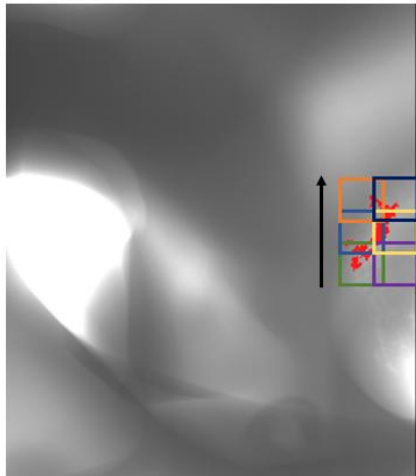
Se necesitan **extraer subregiones** en la zona donde se encuentra el defecto. Cada una de estas subregiones conforma una **imagen** que es la **entrada al modelo** de IA.

El algoritmo de predicción de Delicada consiste en un **modelo de inteligencia artificial** en concreto una **red neuronal convolucional profunda** (Deep Convolutional Neural Network). Las redes neuronales convolucionales, se centran en **extraer las características más importantes de las imágenes** de entrada, mediante el uso de capas de [convolución](#), [pooling](#) y [activaciones no lineales](#), lo que **permite reconocer patrones y realizar tareas de clasificación** y detección con alta precisión.



Para más información sobre redes neuronales convolucionales: [What is a convolutional neural network](#) - LinkedIn

CÁLCULO DEL RESULTADO FINAL

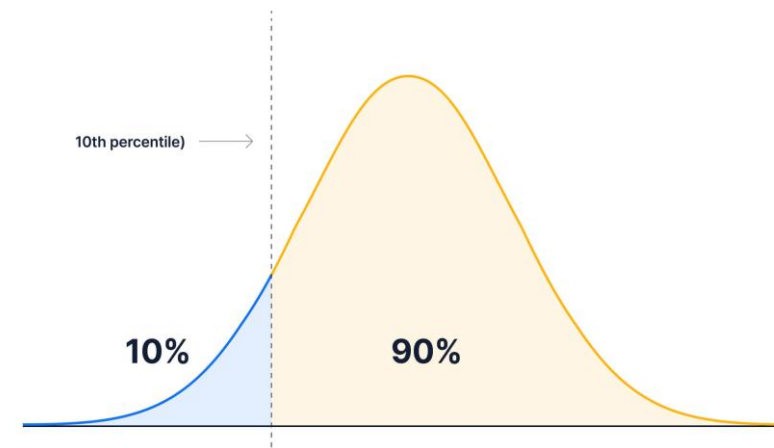


*Resultado de salida
del modelo para
cada sub-imagen*

El **algoritmo** nos da un **resultado para cada sub-imagen** de entrada, pero nosotros necesitamos una **única predicción** para toda la imagen.



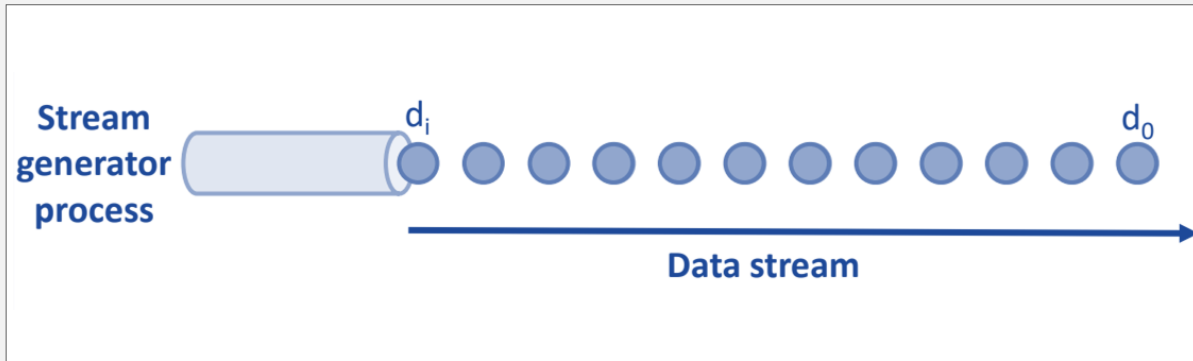
Si el **10% de las sub-imágenes** se clasifican como **NOK**, la **imagen** (y por lo tanto la pieza) se da como **NOK**.



CONCEPT DRIFT

Nuestros **procesos industriales** generan **datos** cuya **distribución se ve alterada** con el tiempo, de forma inesperada, fenómeno **concept drift**.

SIN CONCEPT DRIFT



CON CONCEPT DRIFT

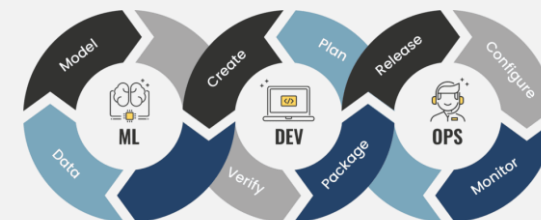


Rendimiento degradante de los modelos entrenados con la distribución original de los datos.

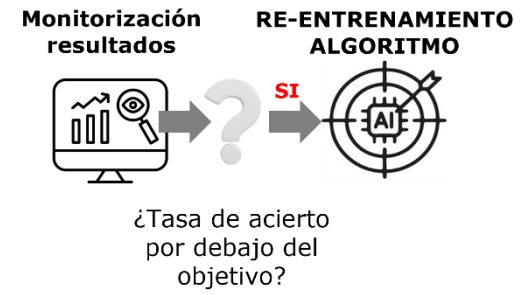
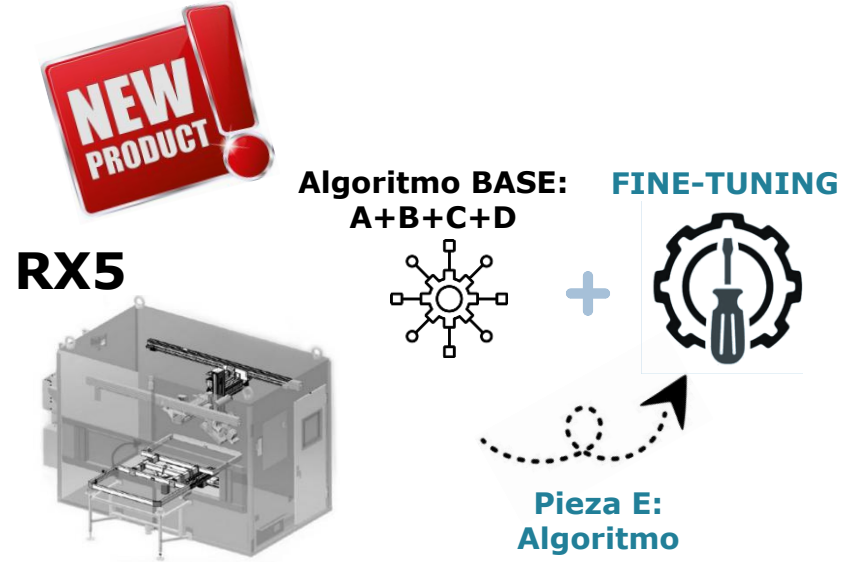
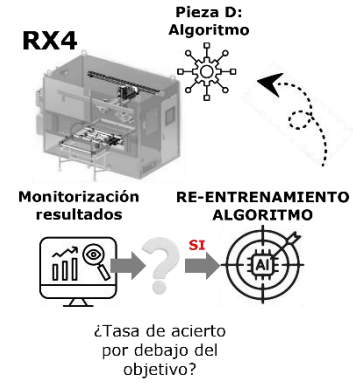
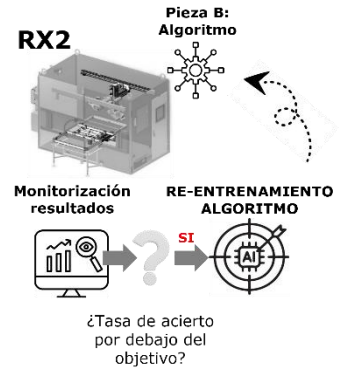
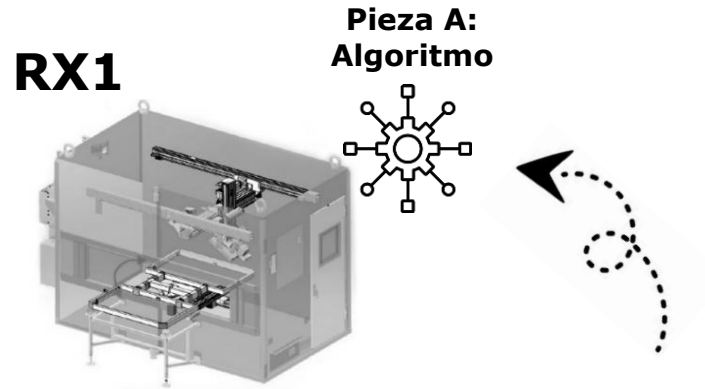
Para mantener la tasa de acierto del algoritmo, ha sido necesario implementar en producción re-entrenamientos automáticos con los nuevos datos, una vez que se detecta de forma automática que la tasa de acierto está por debajo del objetivo.

MLOps

**Automatización de todo el ciclo de vida de la IA:
Desarrollo+Validación+Despliegue+Re-entrenamiento**



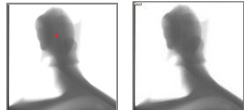
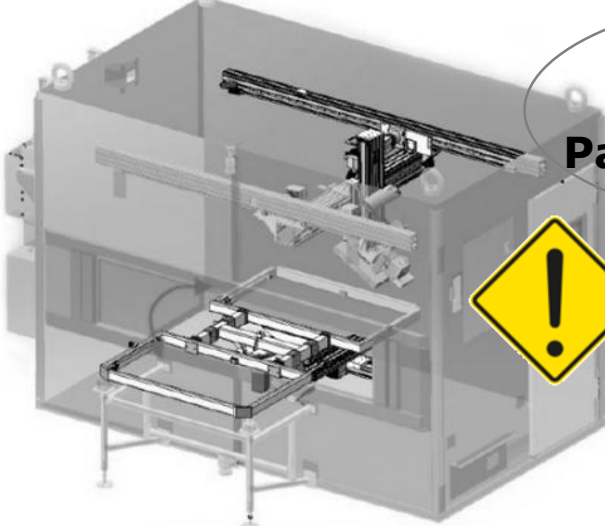
RE-ENTRENAMIENTO AUTOMÁTICO Y ADAPTACIÓN A NUEVOS PRODUCTO



SUCCESS STORY



False Defect



Pattern <-> Part -> NOK



A % of the parts rejected were OK parts.

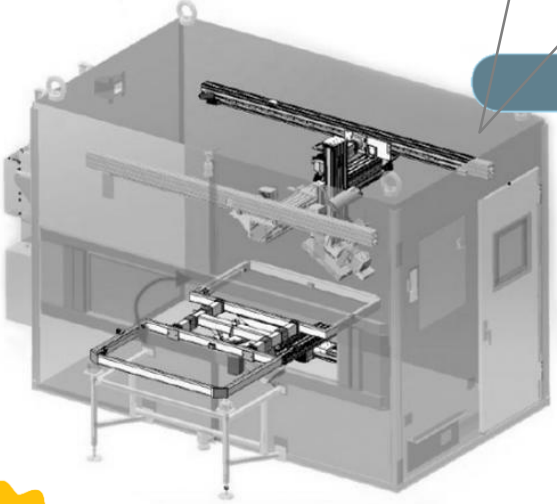
BEFORE...

A collection of technology logos including Kubeflow, TensorFlow, docker, kubernetes, python, django, PostgreSQL, HTML, JS, CSS, and a logo with a hand holding a gear.

COMPUTER VISION

Pattern <-> Part -> NOK or OK

NEURAL NETWORKS



False NOK Removed

NEW

NOWADAYS

EN POCAS PALABRAS

