Curso Profesional de

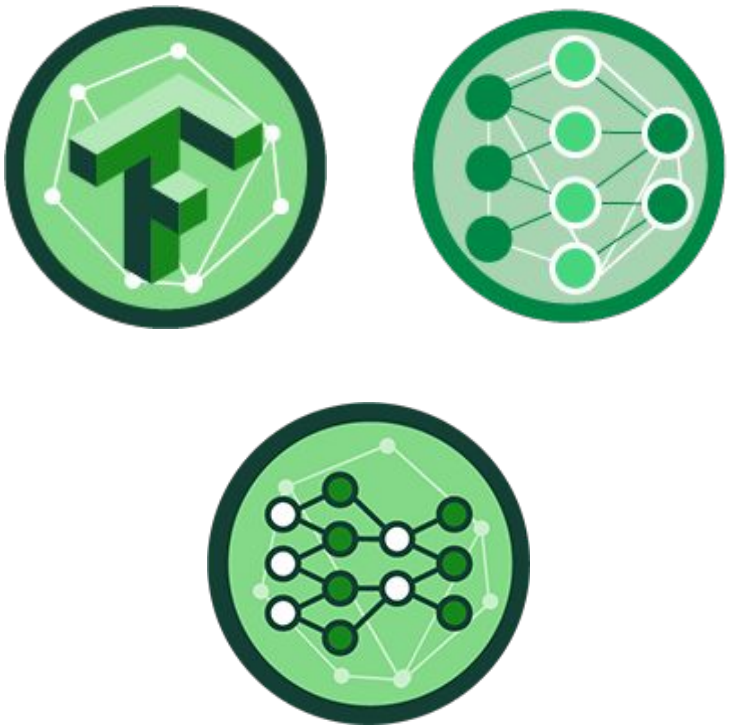
Computer vision con TensorFlow

Adonaí Vera
Co-Founder y CTO en SwitchAI

www.switchai.co

Requisitos del curso

- Fundamentos de redes neuronales con Python y Keras.
- Redes neuronales con TensorFlow.
- Redes neuronales convolucionales.



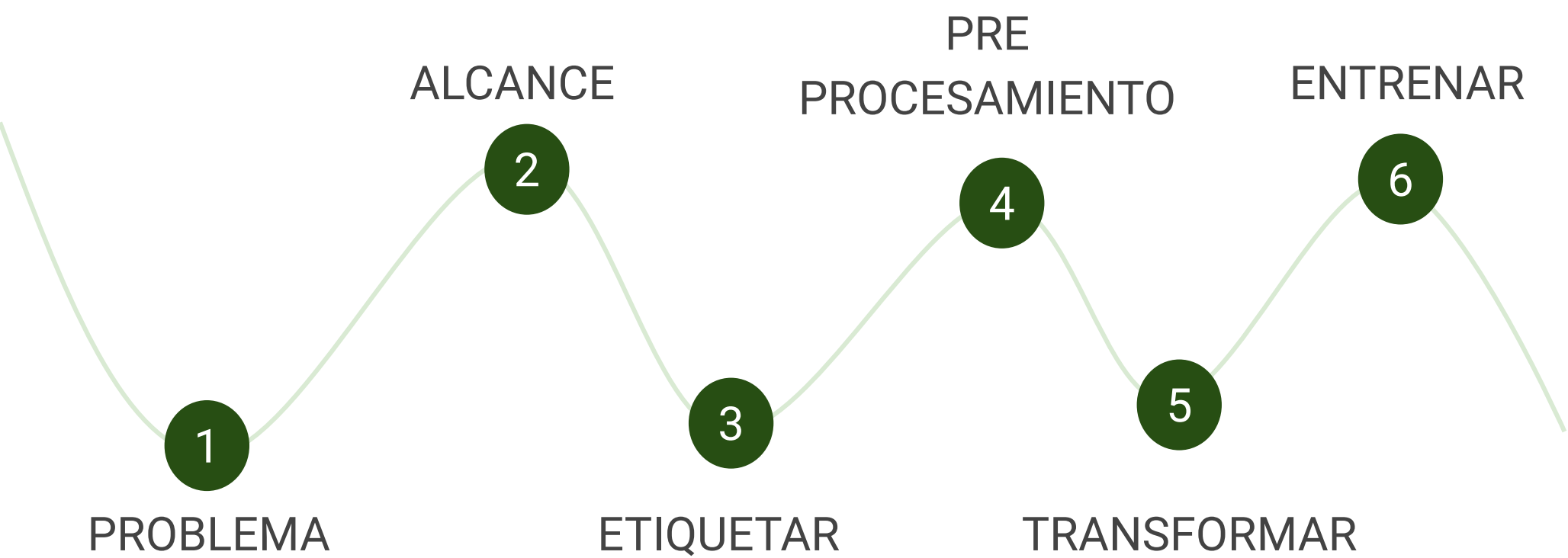
Por qué debes tomar el curso

- Pasar de crear un modelo de inteligencia artificial a crear un producto para vender.
- Dimensionar el costo y alcance de tu proyecto.
- Entender a fondo la visión computarizada.
- Crear tu propio modelo de object detection.

Por qué debes tomar el curso

- Convertir múltiples formatos a TFRecord.
- Pasar de clasificar un objeto a localizarlo y clasificarlo.
- Poner en producción tu modelo.
- Tienes una buena idea con IA y no sabes cómo empezar.

¿Qué aprenderás?



¿Qué aprenderás?

7 EVALUAR RESULTADO

8 OPTIMIZAR

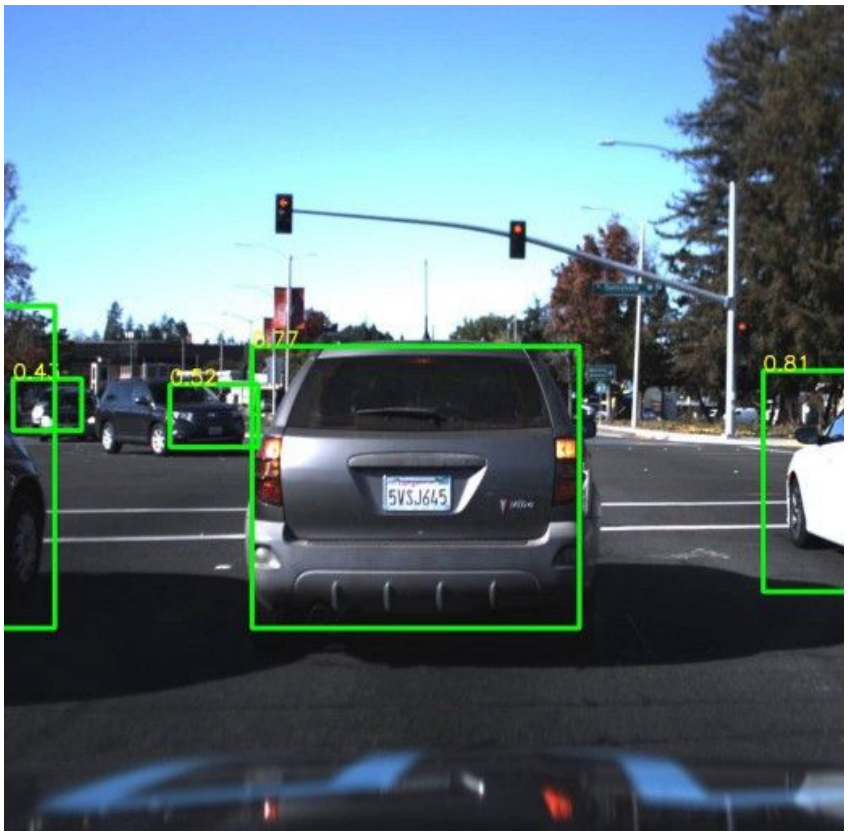
9 DESPLEGAR GOOGLE

10 MONITORIZAR

11 CIERRE BUCLE

Proyecto del curso

- Detección.
- Clasificación.
- Seguimiento.
- Conteo de vehículos.



Vamos a pasar de...

Clasificación de objetos

→

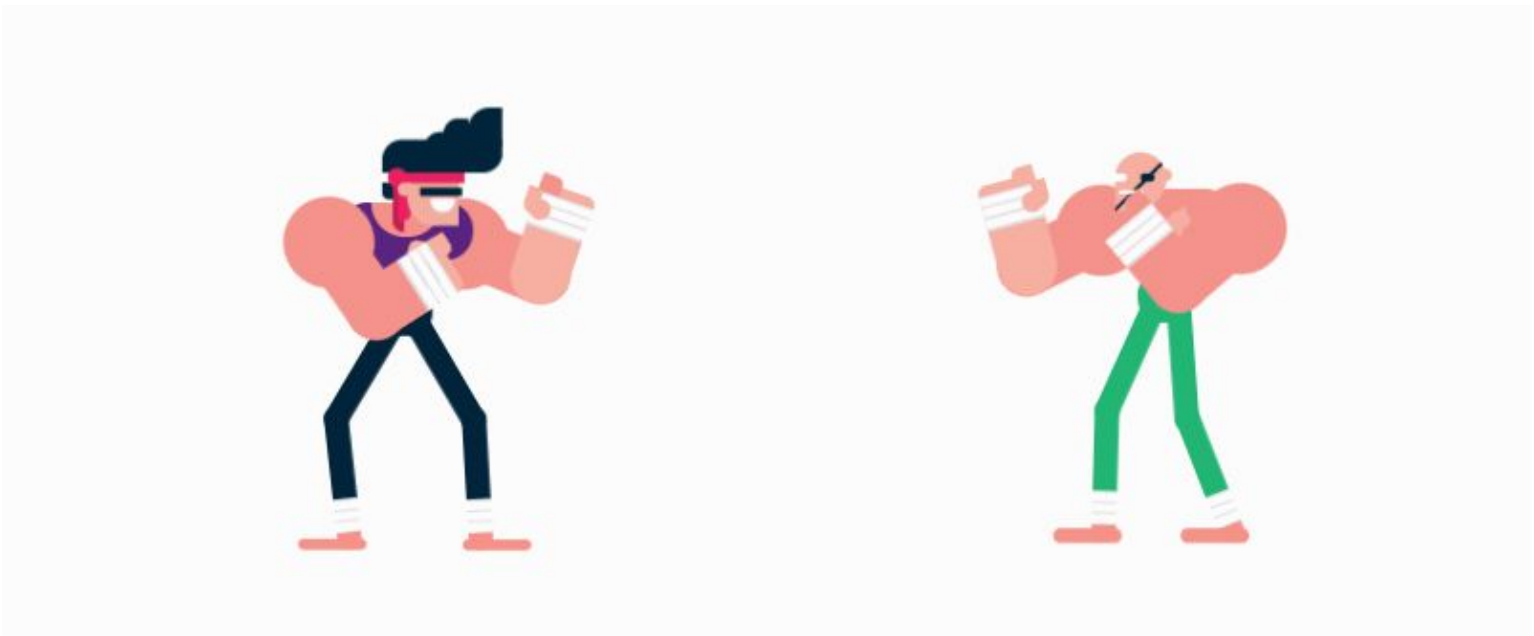
Detección de objetos.

Segmentación de objetos.

Segmentación de instancias de objetos.

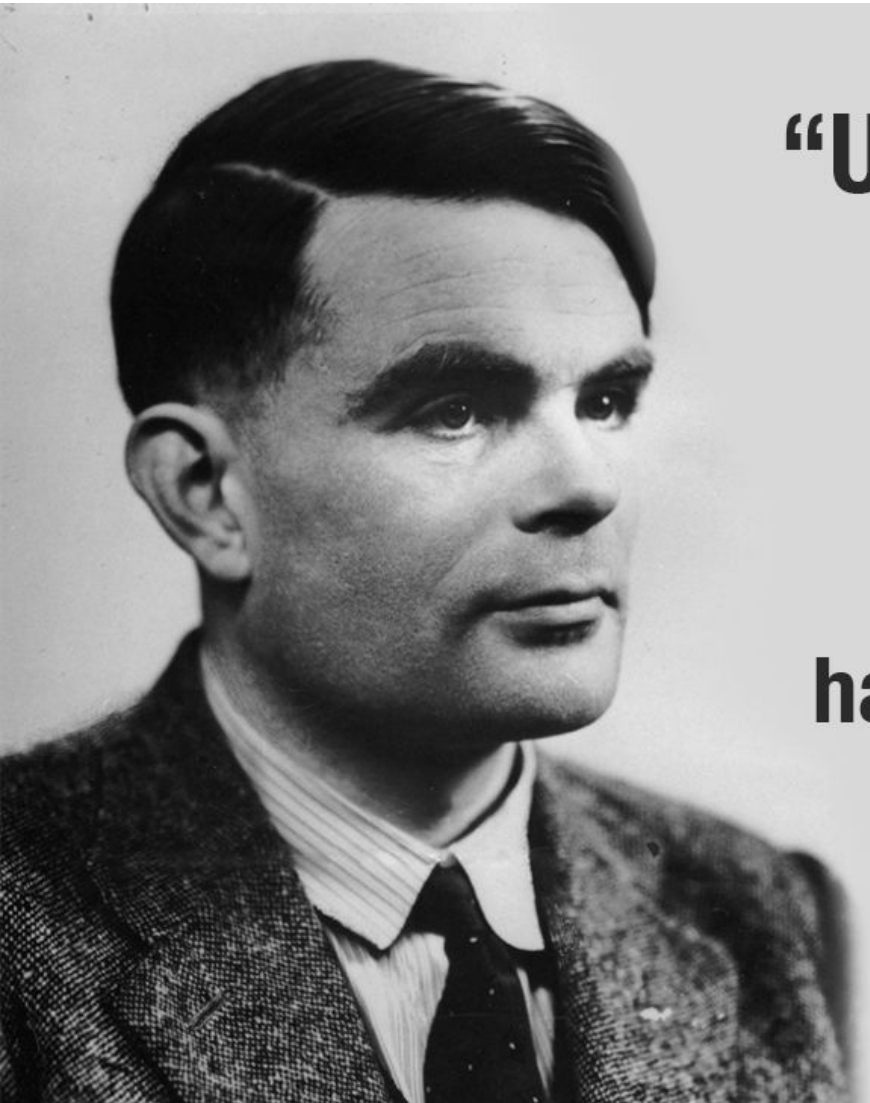
Segmentación panóptica.

¿Por qué decidimos crear este curso?



¿Qué es la visión computarizada?

¿Qué es inteligencia artificial?

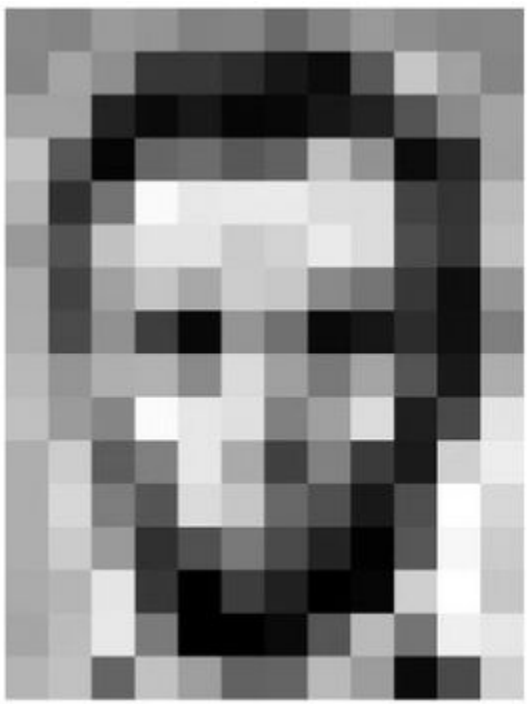


"Una computadora puede ser llamada **inteligente** si logra engañar a una persona haciéndole creer que es un **humano**"

Alan Mathison Turing
1912–1954

¿Qué es la visión computarizada?

¿Qué es una imagen?



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

¿Qué tener en cuenta en una imagen?

- Resolución en pixeles.
- Escala de colores (RGB, BGR, HSV, escala de grises).
- Tamaño de la imagen.
- Frames per second.

Ejemplos de visión computarizada

Control de calidad

Caracterizar clientes

Ejemplos de visión computarizada

Detección enfermedades en plantas

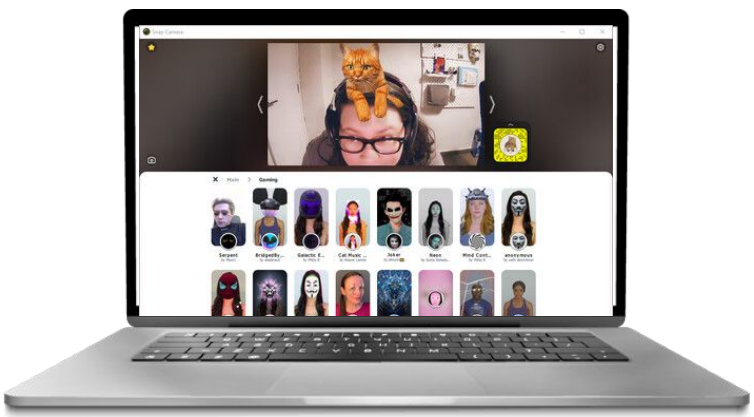
Conteo de vehículos

Ejemplos de visión computarizada

Clasificación cavings

Control de riesgos

Ejemplos de visión computarizada



Filtros inteligentes

Clasificar sentimientos

Tipos de visión computarizada

Clasificación objetos

PERSONA

Clasificación multiclase

PERSONA | PERRO

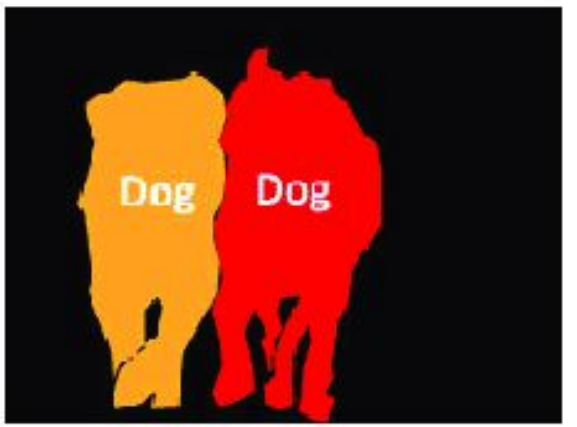
Clasificación de múltiples etiquetas

Detección de objetos

Segmentación de imágenes

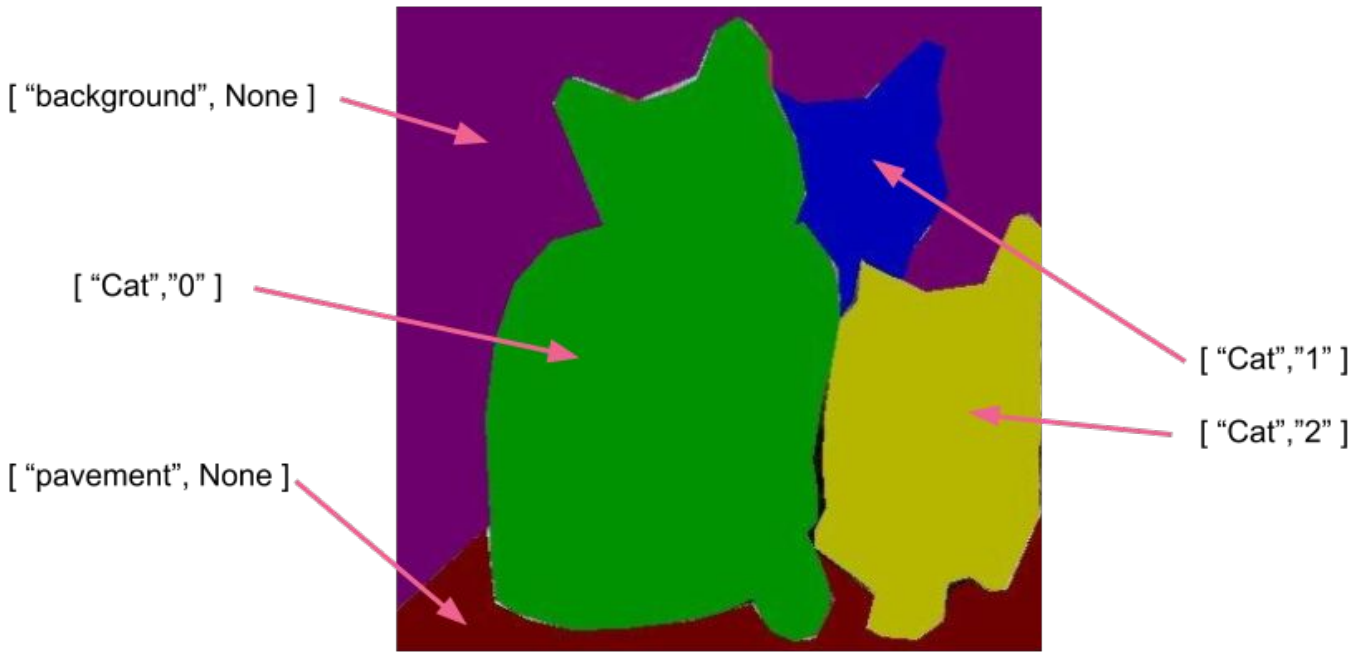
Original

Segmentación semántica



Segmentación de instancias

Segmentación panóptica



Segmentación semántica

+

Segmentación de instancias

Introducción a object detection

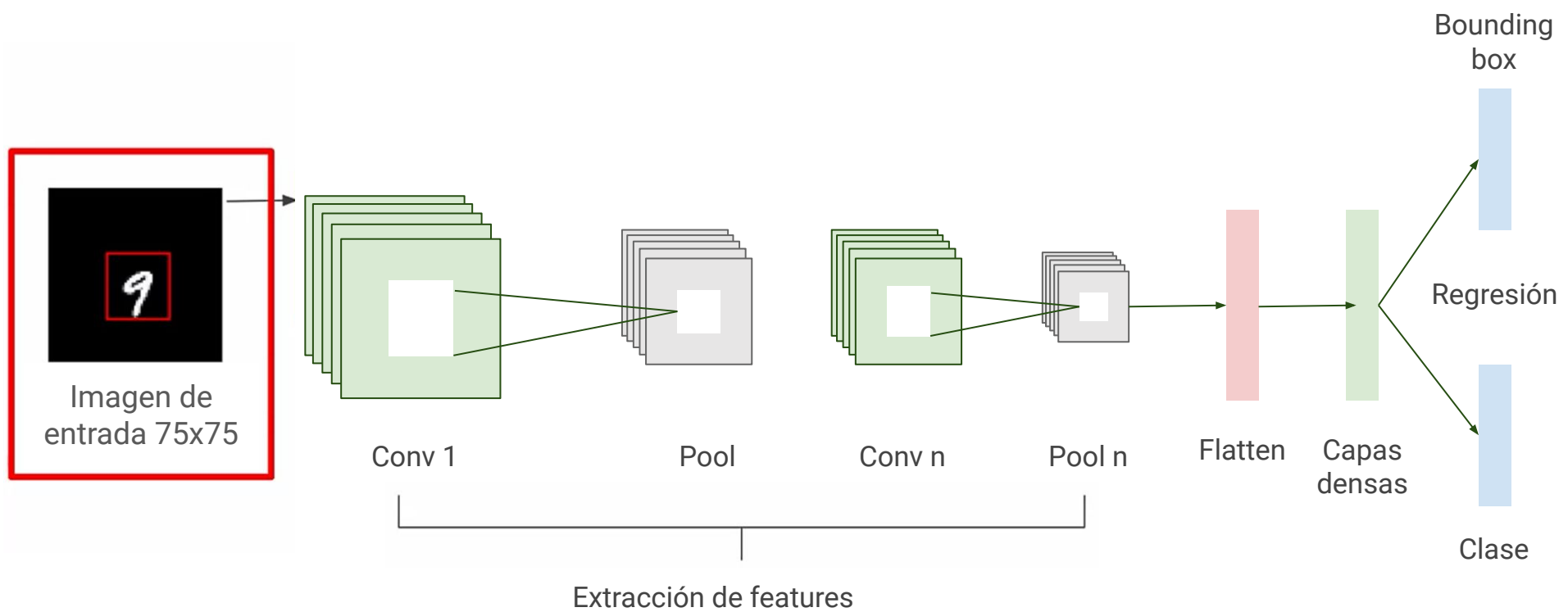
Clasificación y localización + Seguimiento

Composición de la detección de objetos

Localización del objeto

Detección del objeto

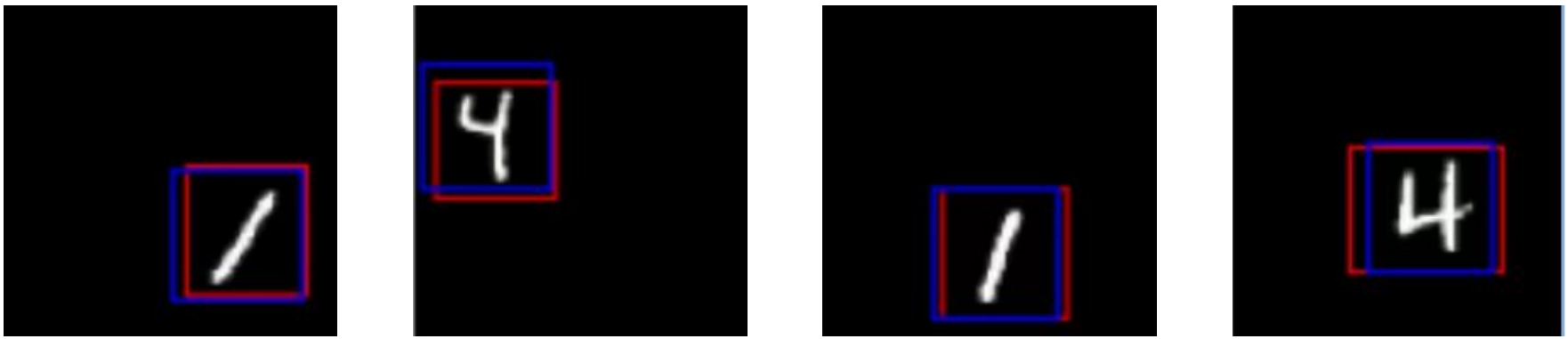
Arquitectura de detección de objetos

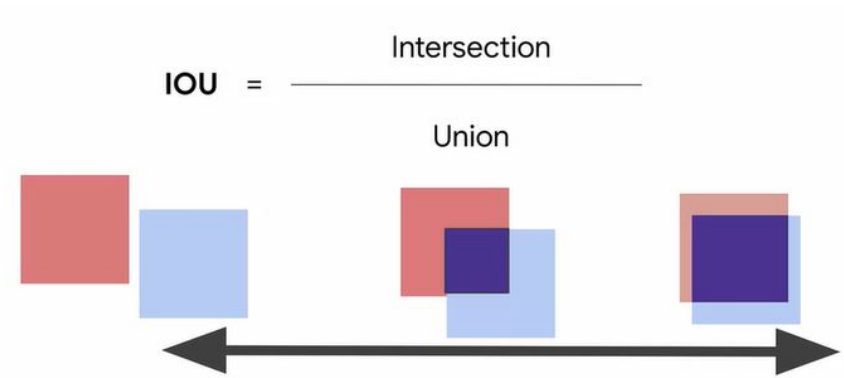


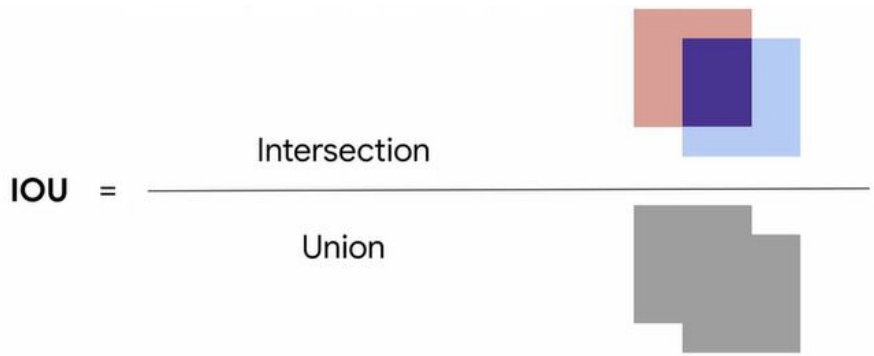
Métricas relevantes

- Intersección entre uniones (IoU)
- Accuracy
- Loss

Intersección entre uniones IoU

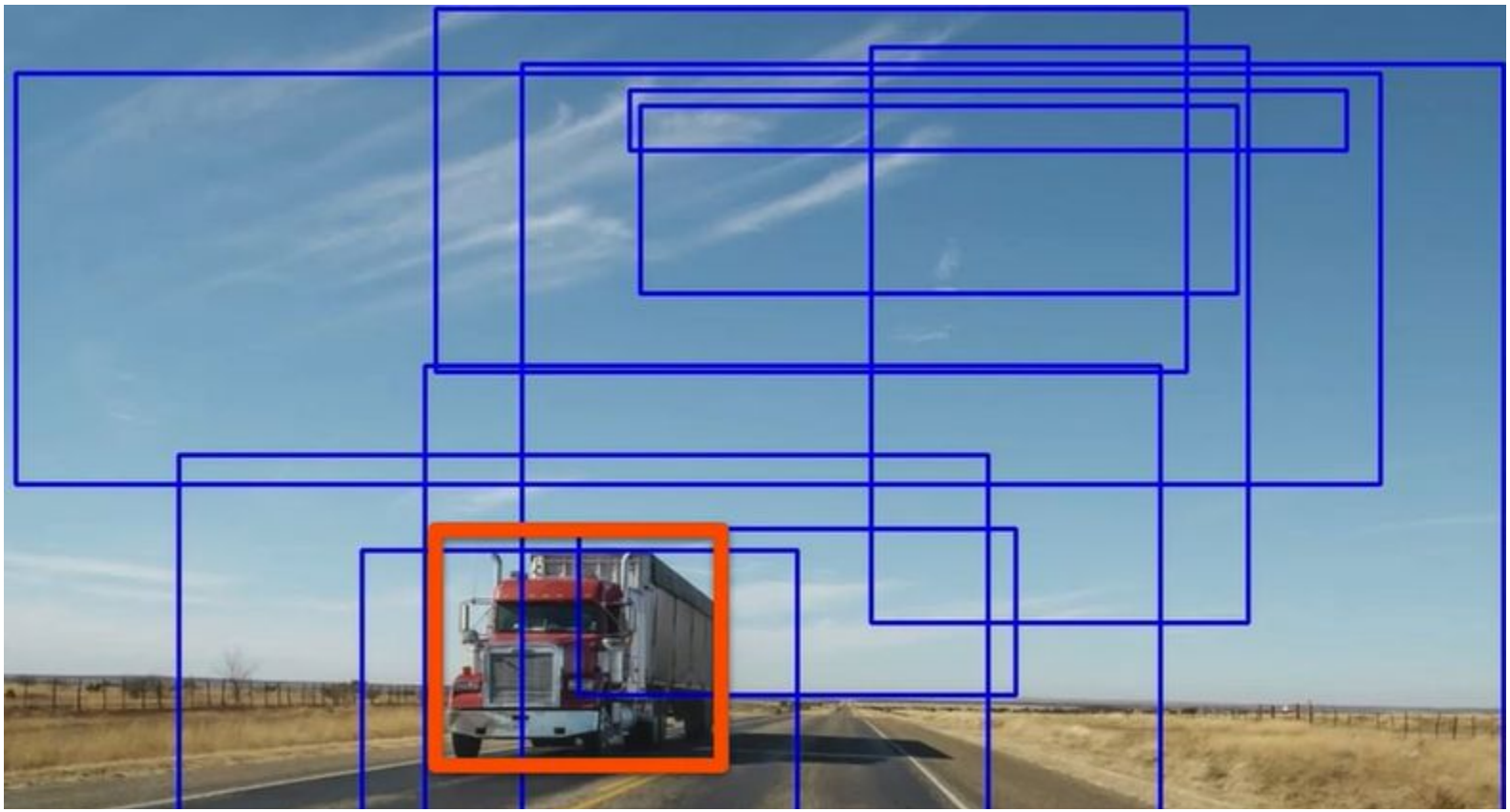






Object detection and sliding windows

Object detection and selective search



Non-maximum suppression (NMS)

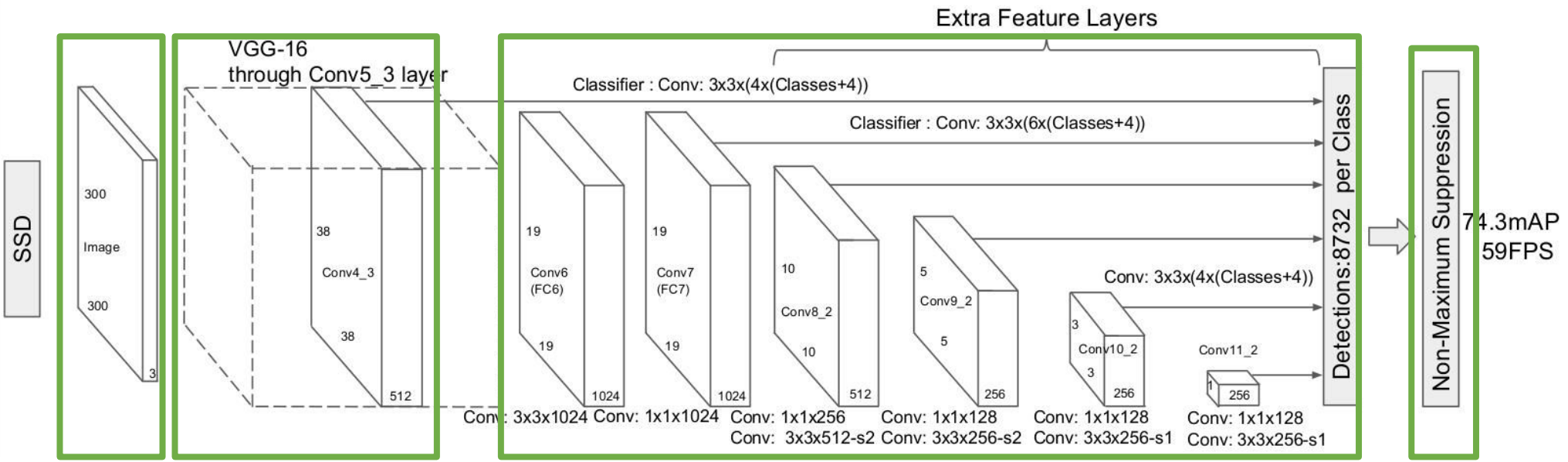
Antes NMS

Después NMS
IoU > 0.51

Algoritmos más utilizados

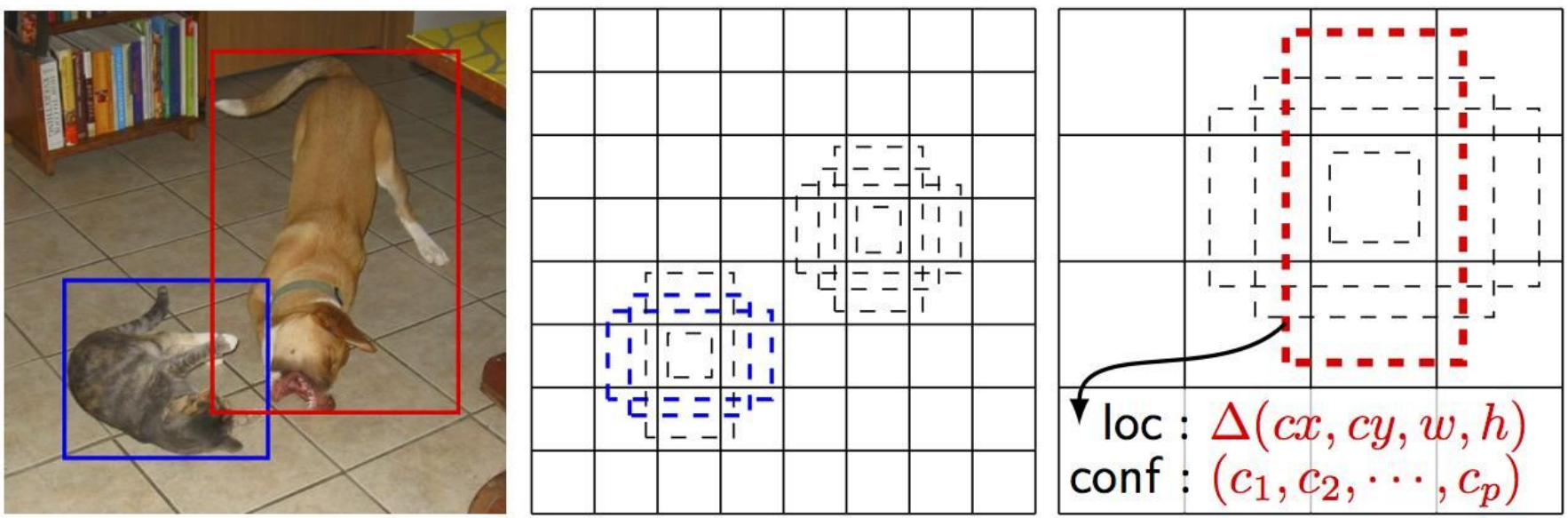
R-CNN
FASTER R-CNN
YOLO
SSD

Single Shot Detector



Referencias. Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg (2015). SSD: Single Shot MultiBox Detector. 10.1007/978-3-319-46448-0_2.

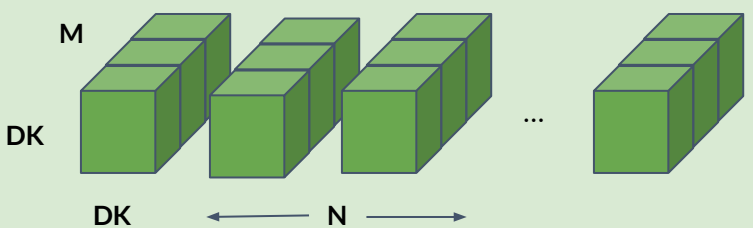
Feature Map



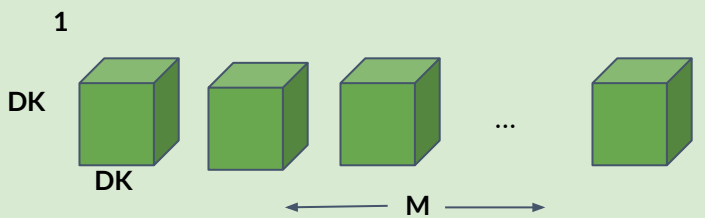
(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

Referencias. Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg (2015). SSD: Single Shot MultiBox Detector. 10.1007/978-3-319-46448-0_2.

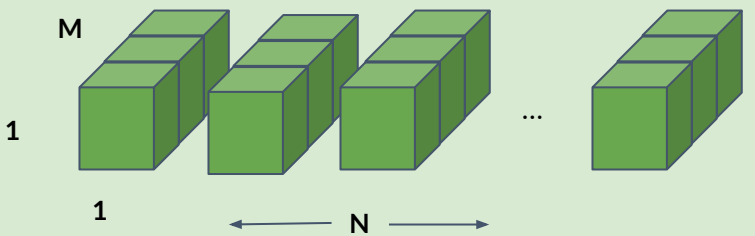
MobileNet V1



Filtros convolucionales Estándares



Filtros convolucionales DepthWise



1x1 PointWise convolucionales

MobileNet V1

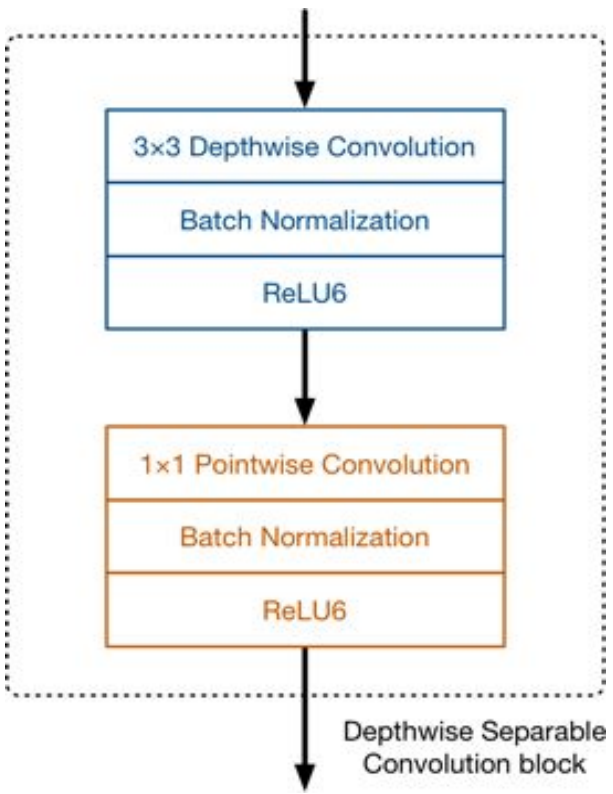
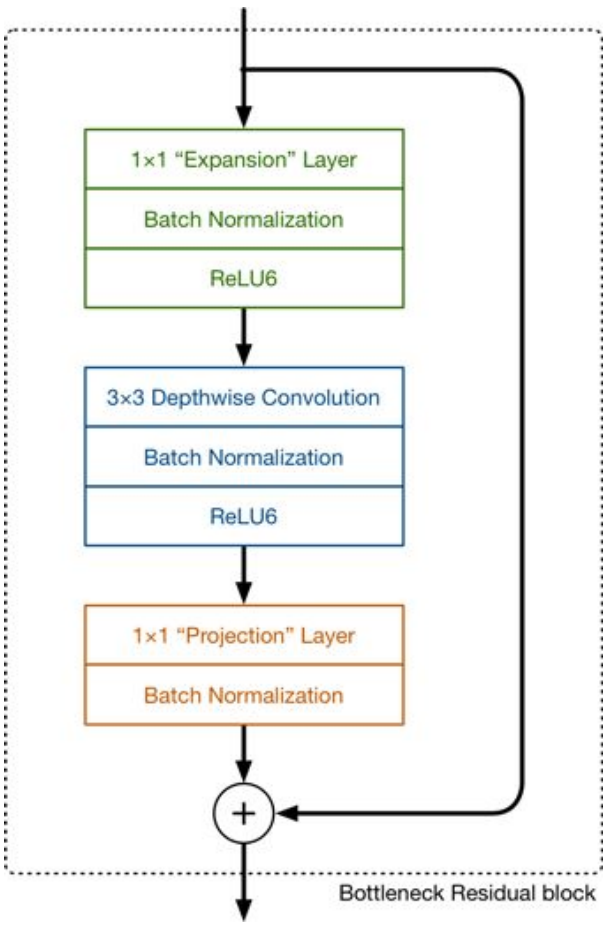


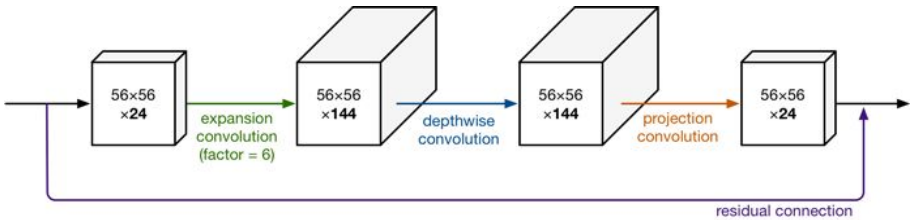
Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Referencias. Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv:1704.04861

MobileNet V2

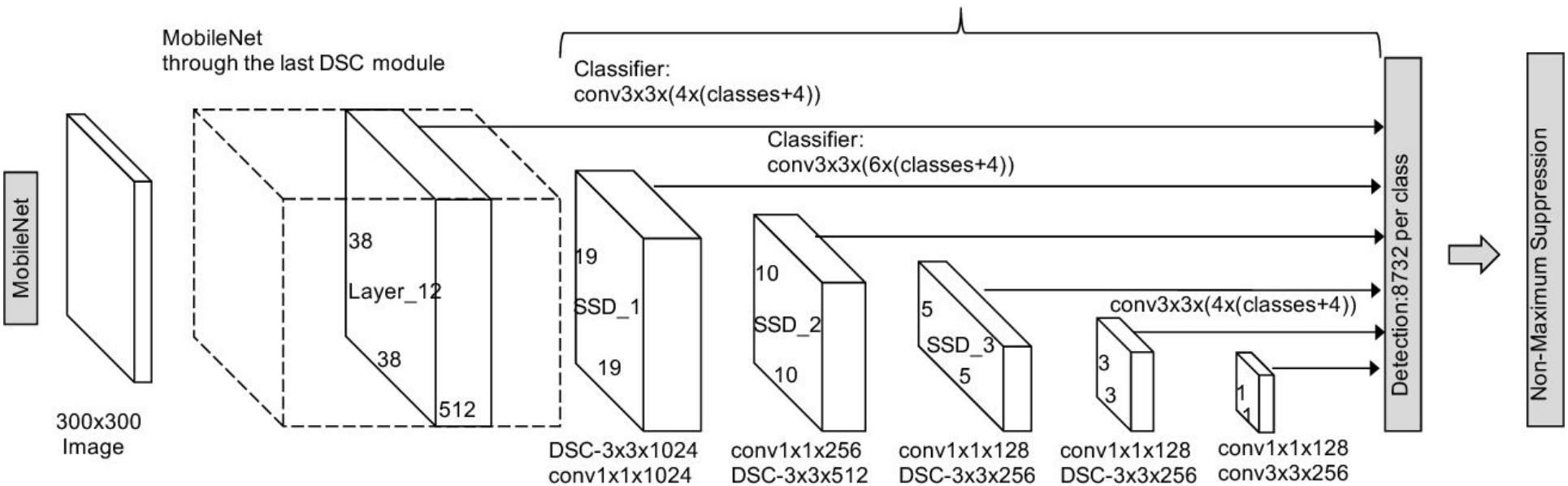




Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Referencias. Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. arXiv:1801.04381.

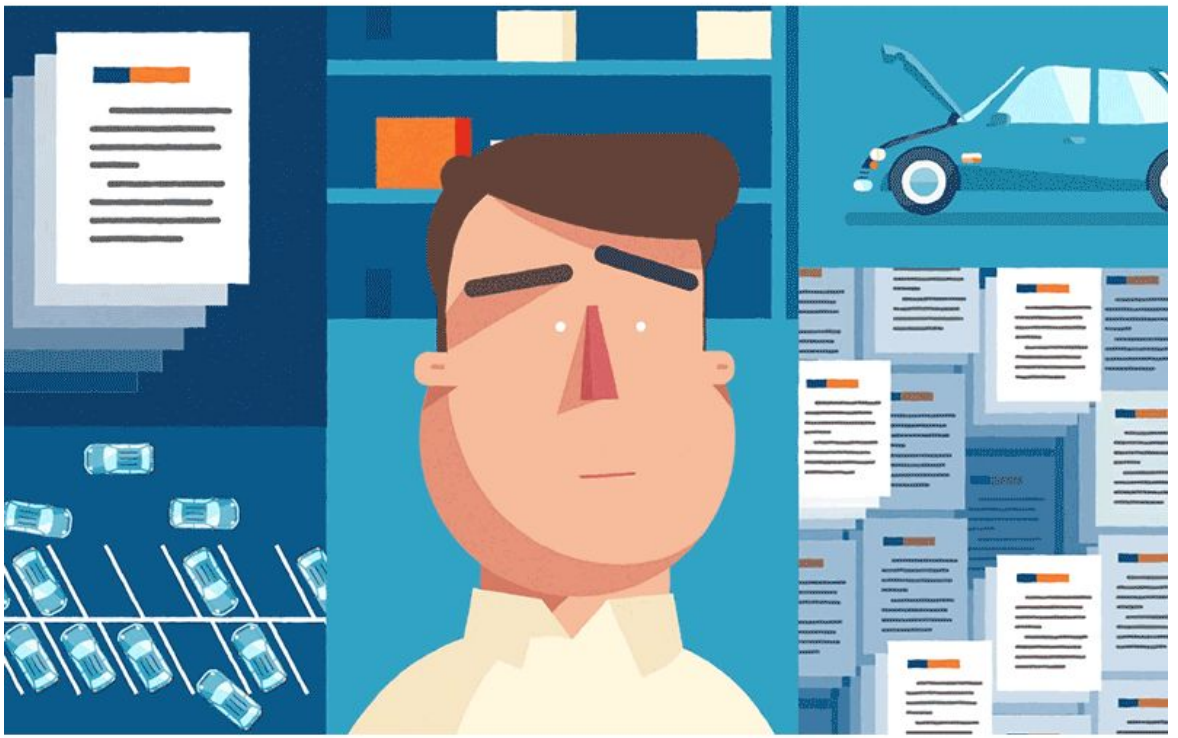
MobileNetV2 + SSD



Referencias. Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. arXiv:1801.04381.

Aprende a identificar problemas

Modo detección: ON



Referencias. Allen Laseter. (2015). Autosoft pt2

Identificar problemas

Encuentra problemas

Entiende cuál es el resultado que buscas

Conoce usuarios

La solución NO siempre es IA.

“

Enamórate del problema, no de la solución

”

Uri Levine, el creador de Waze

Nuestro caso de estudio

Todos los días me toma alrededor de 2 horas para llegar a mi casa.

CONGESTIÓN VEHICULAR

Nuestro caso de estudio

No existen métricas en tiempo real que nos permitan monitorizar y analizar la distribución de vehículos en mi ciudad.

Nuestro caso de estudio

Las ciudades mexicanas pierden **94 mil millones de pesos o 4.6 mil millones de USD** por el tráfico al año.

Fuente: IMCO Instituto Mexicano para la Competitividad A.C.

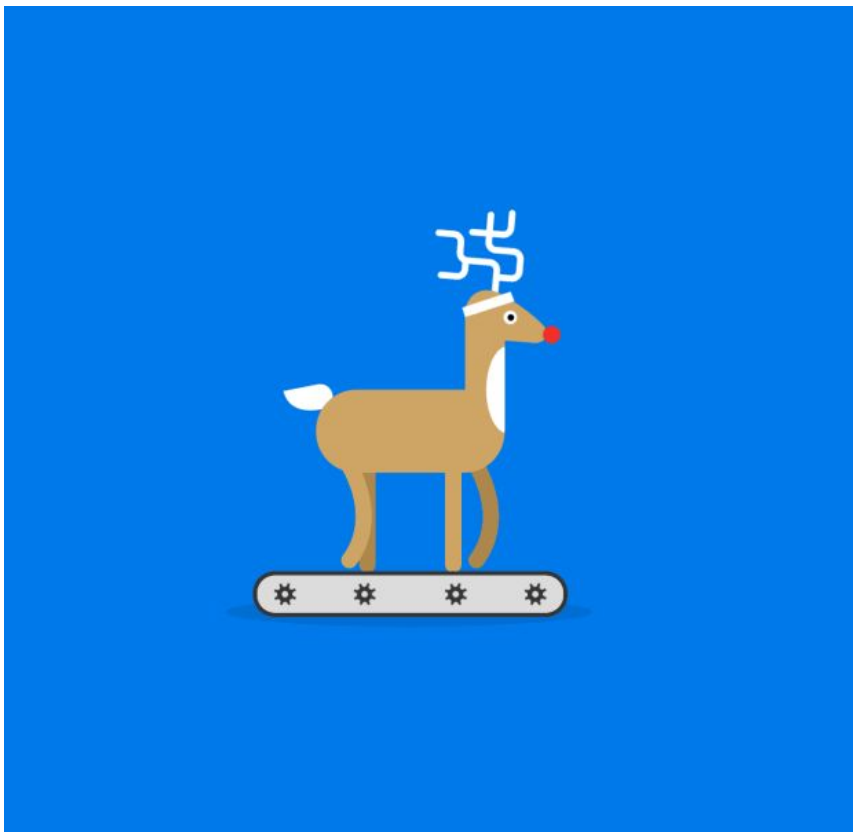
Cómo definir los tiempos de entrega de tu proyecto

En este módulo aprenderemos

- Cómo definir el alcance de tu proyecto: tiempos y costos.
- Identificar roles para tu proyecto.
- Definir el producto mínimo viable.

Tiempos externos

¿Existe alguna fecha externa **obligatoria** que debas cumplir?



Define el alcance

Must have

Should have

Nice to have

Cómo costear tu proyecto

Recomendaciones

- No existe una única forma de costear.
- Crea tres escenarios.
- Agrega a los tres escenarios un margen de falla.
- Agrega costos de servidores o herramientas externas.

Recomendaciones

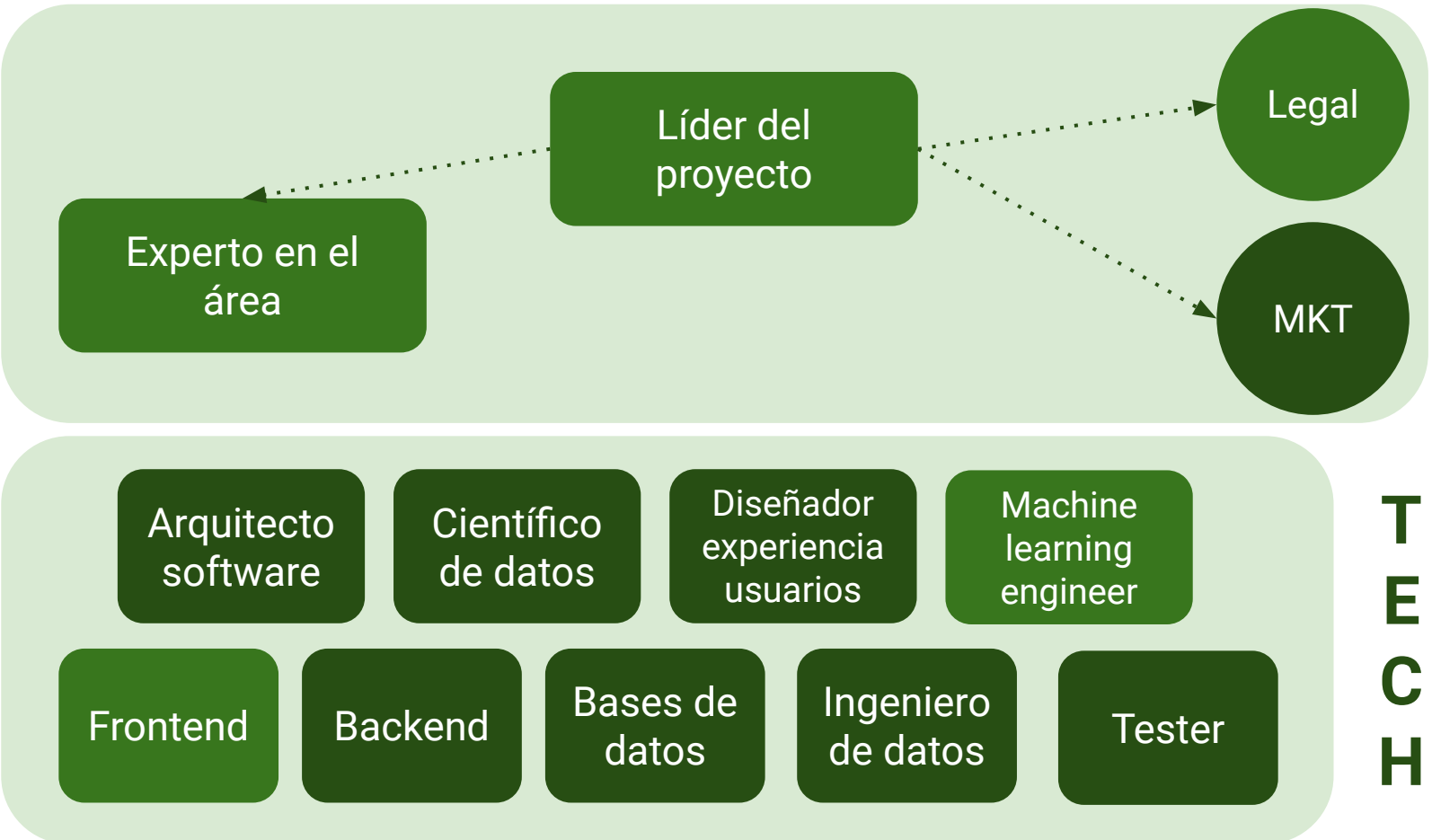
- Divide tu proyecto por entregas mensuales.
- Valida y firma el alcance propuesto: cambios representa costos adicionales.
- Analiza la metodología de cobro.
- QUE NO TE DE MIEDO COBRAR.

Cómo identificar los roles para tu proyecto

Fusión

Experto
+
ML team

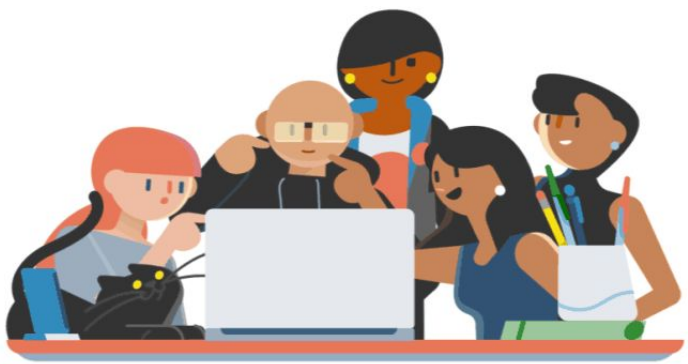
Roles


Líder del proyecto
Legal
Experto en el área
MKT
Arquitecto software
Científico de datos
Diseñador experiencia usuarios
Machine learning engineer
Frontend
Backend
Bases de datos
Ingeniero de datos
Tester
TECH

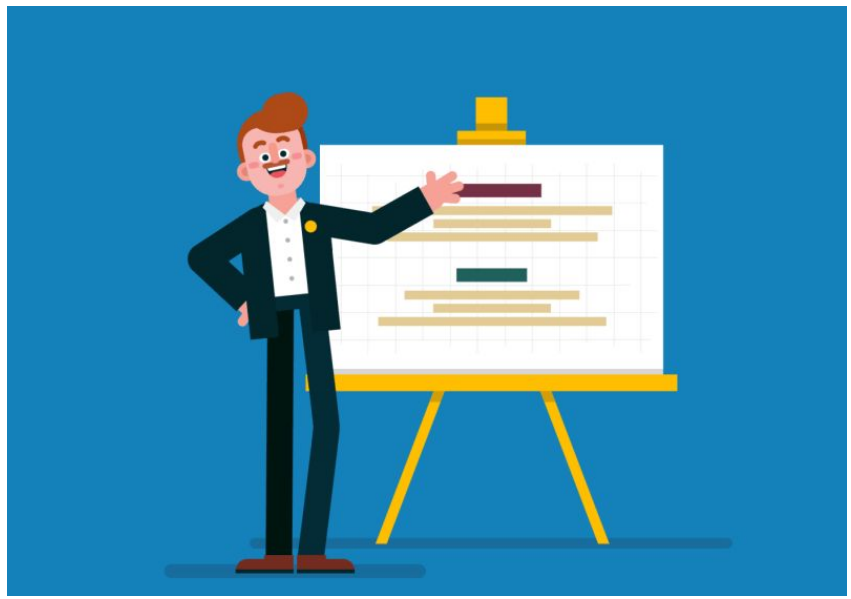
Ingeniero de ML es experiencia

- Busca tener un par de expertos en tu equipo.
- Busca consultores externos para cubrir los vacíos de conocimiento.
- Integra un Full Stack a tu proyecto.
- Busca diversidad en tu equipo.

Escoge tu camino



Contratar personal capacitado

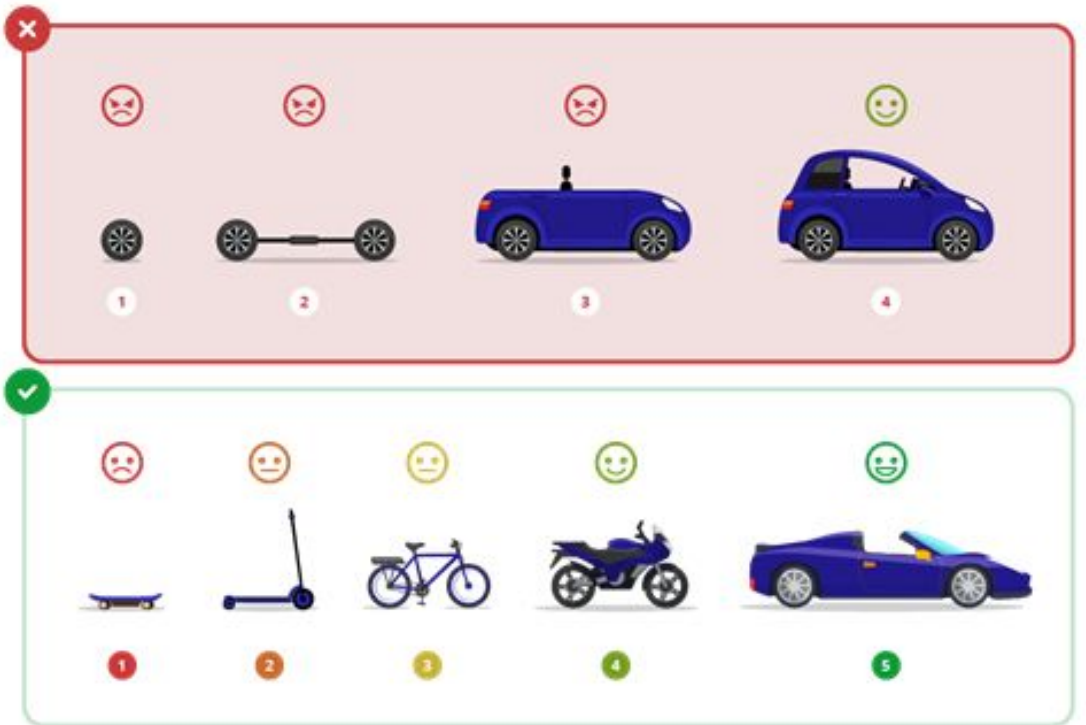


Enseñar a personal que está iniciando

Producto mínimo viable en CV

Producto Mínimo Viable (PMV)

- Try small.
- Fail fast.
- Fail cheap.
- Fail forward.
- Don’t aim for perfection.



1 mes

Equipo y alcance

- Crear un algoritmo de detección de objetos que localice y clasifique carros y motos.
- Seguimiento a los objetos.
- Generar un umbral para contar dirección y cantidad.

Obtención de datos para tu proyecto

En este módulo aprenderemos

- Recomendaciones al crear tu base de datos.
- Obtención de datos para tu proyecto.
- Etiquetado de base de datos.

En este módulo aprenderemos

- Limpiar base de datos (preprocesamiento).
- Aplicar data augmentation.
- Transformar la base de datos.
- Distribuir la base de datos.

Genera la mayor cantidad de escenarios

1. ¿Qué pasa si el vehículo va a 100 km/h?
2. ¿Qué pasa si un camión tapa a un vehículo?
3. ¿El algoritmo funcionaría en la noche?
4. ¿Qué sucede si llueve?
5. ¿En dónde se integrará nuestra tecnología?
6. ¿Quién tiene acceso a la base de datos?

¿Dónde buscar?

Fuente del cliente/empresa

kaggle

Google Dataset Search Beta

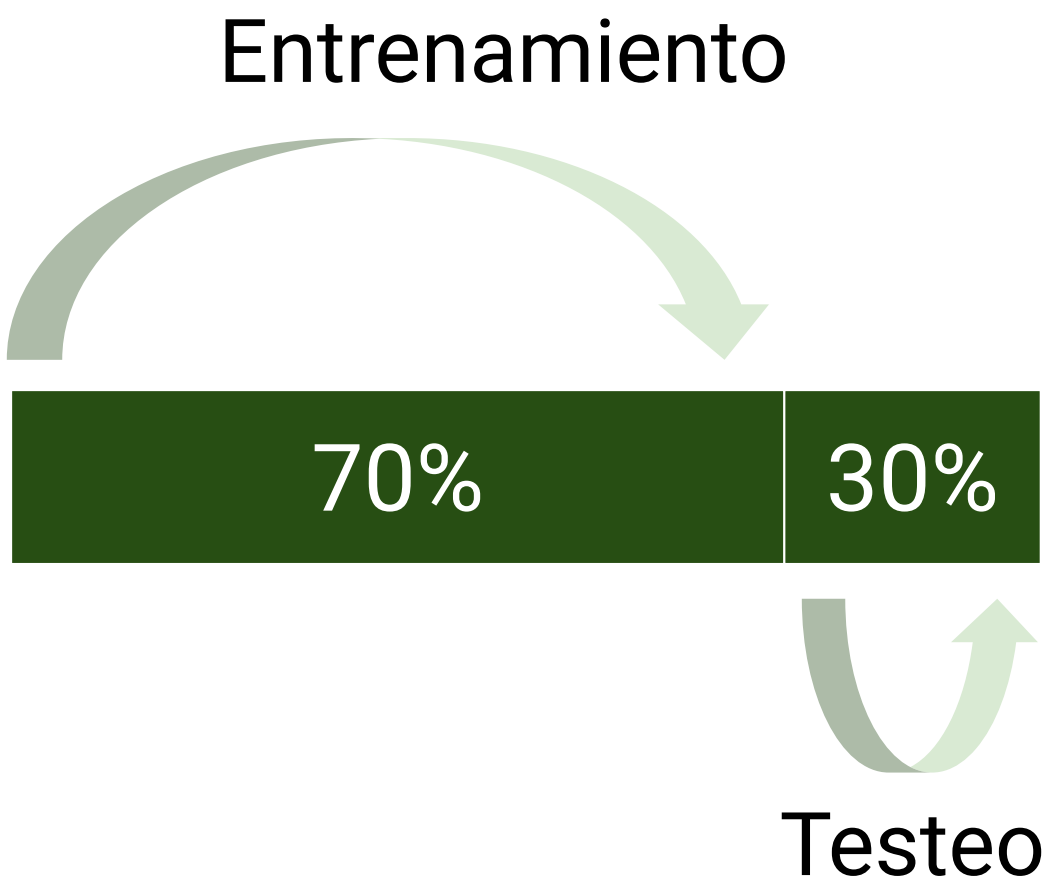
Creada manualmente

Limpieza de la base de datos

Navega por tu base de datos

Distribución de datos en entrenamiento y testeo

Distribución de los datos

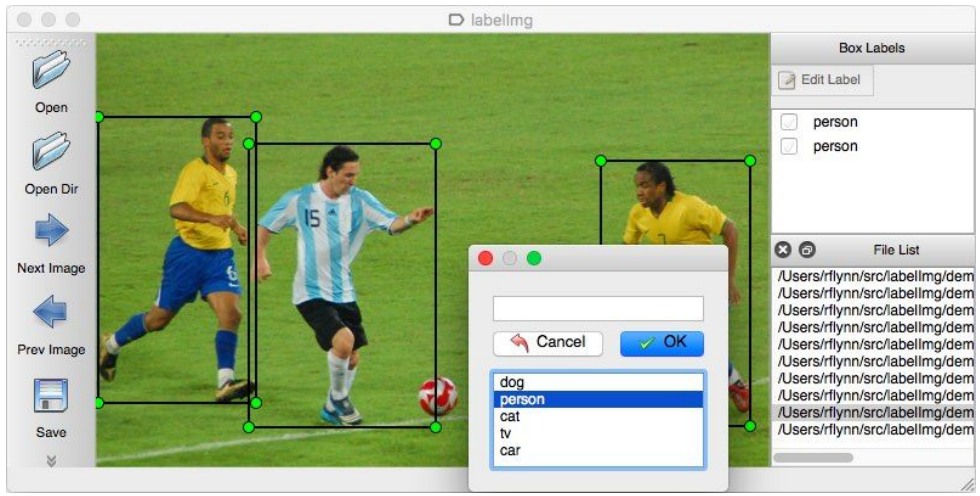


Etiquetado de las bases de datos

Herramientas para etiquetar

Supervisely

LabelImg



LinkedAI

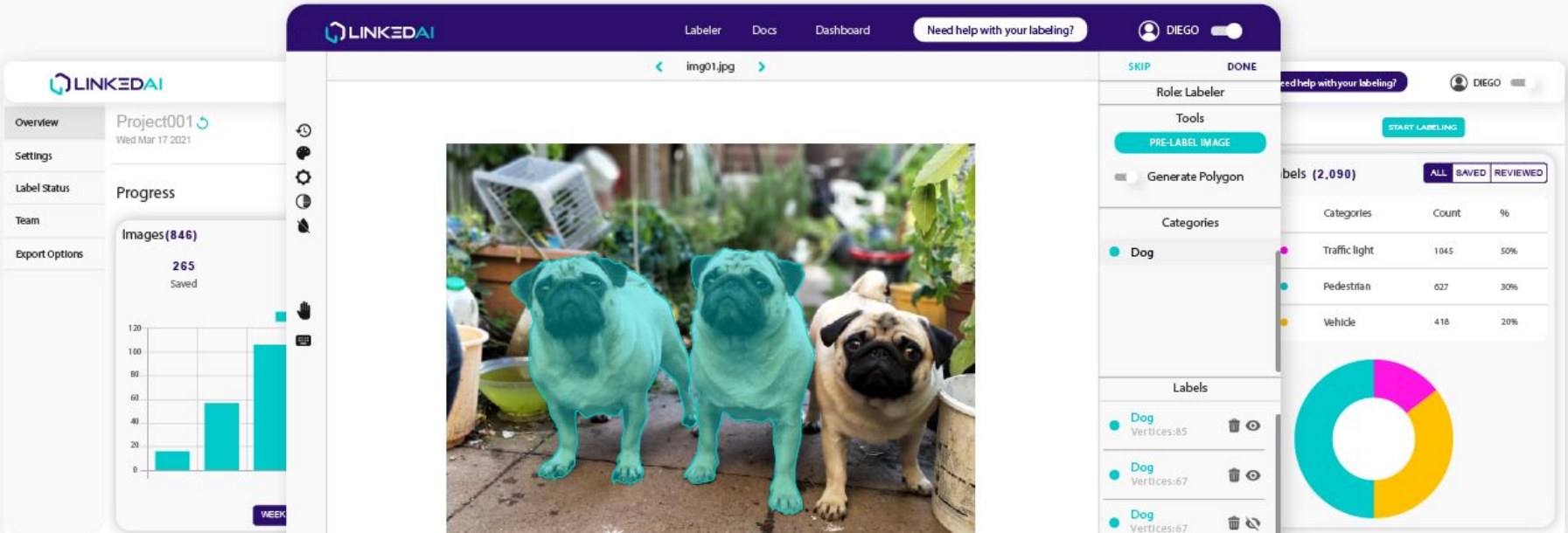
LINKEDAI

Home Product Pricing Blog Try It Free REQUEST DEMO

Simplifying AI Training Data

A complete solution for taking control of your training data, with fast labeling tools, human workforce, data management, and automation features.

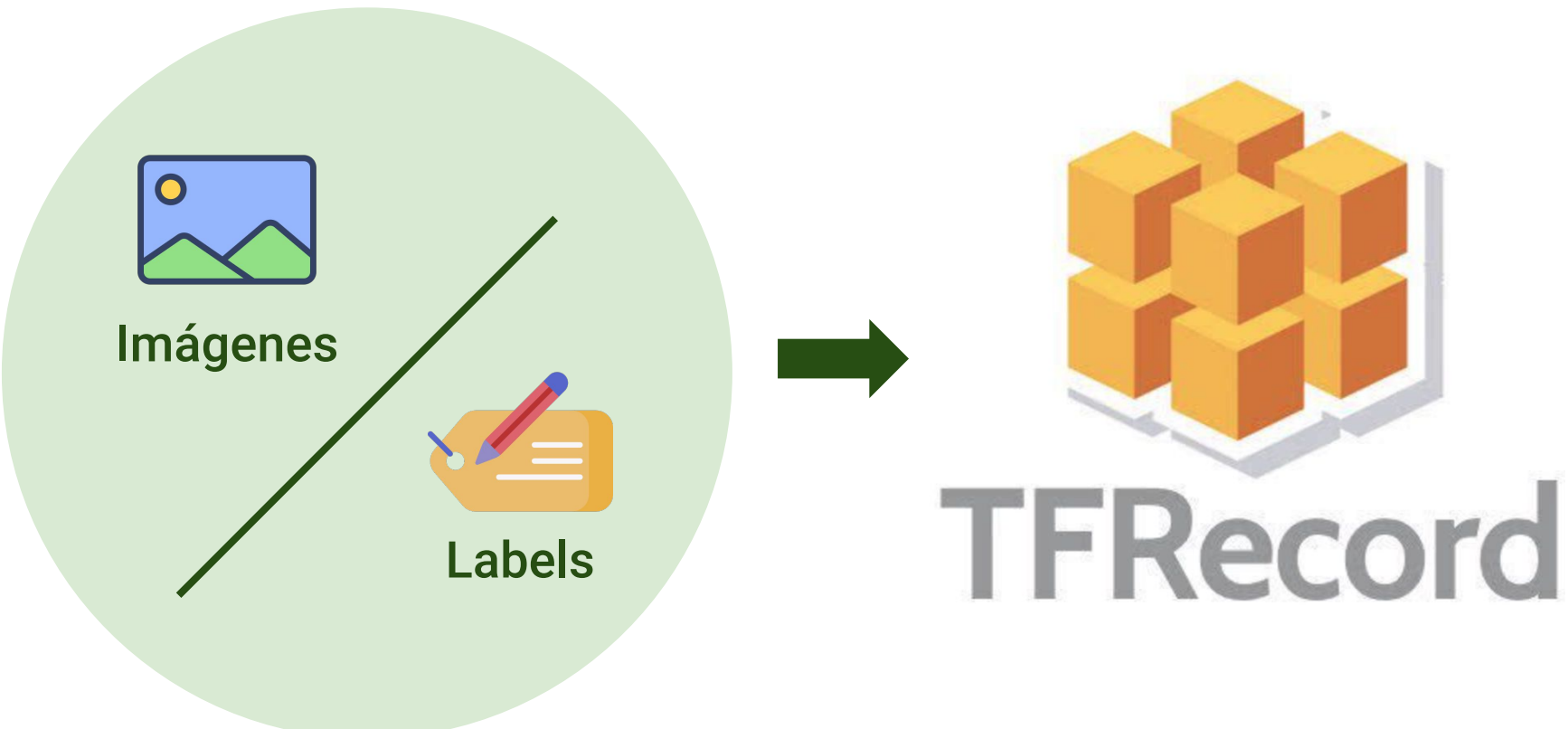
Try it Free Request Demo



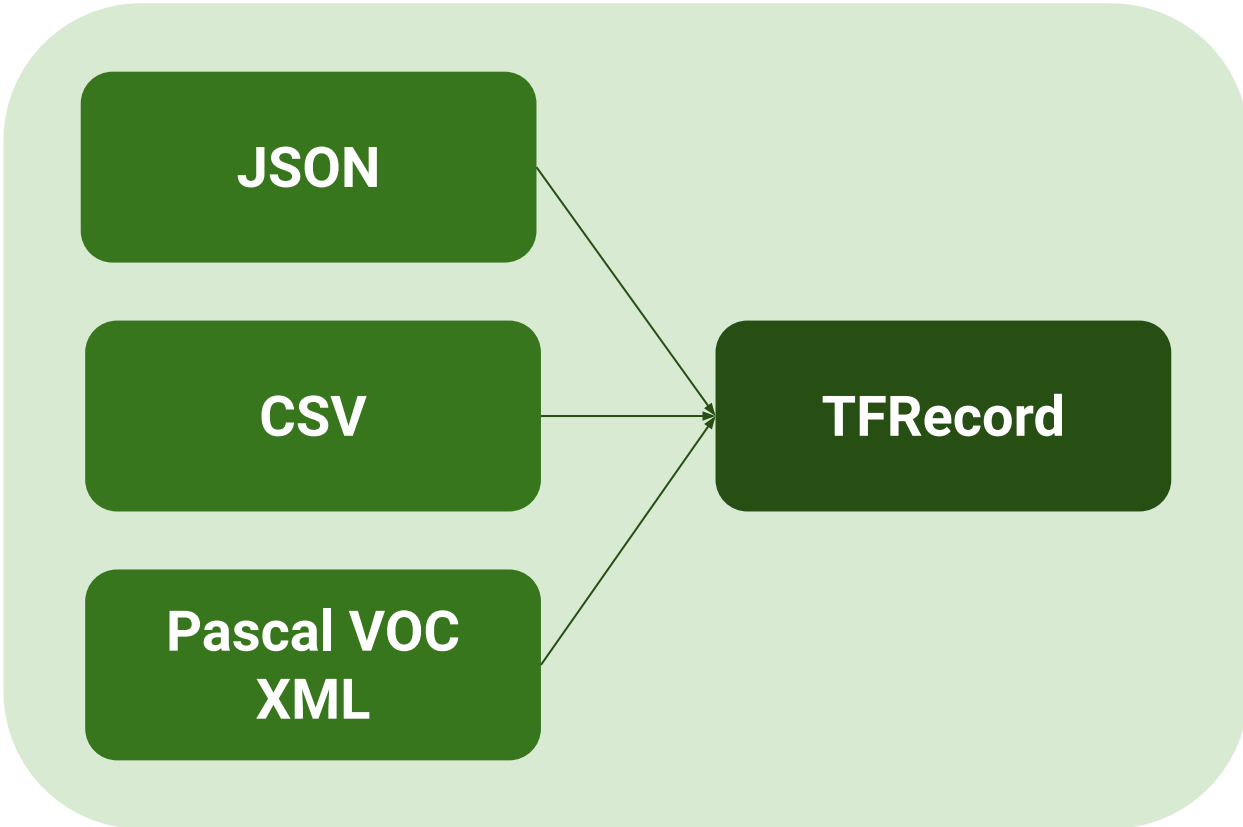
Etiquetado de los datos de train

Transforma tu base de datos a TFRecord

TFRecord



Formatos



CSV a TFRecord

Fase de entrenamiento del modelo

En este módulo aprenderemos

- Qué elementos se necesitan para entrenar tu modelo.
- Aplicar aprendizaje por transferencia.
- Evaluar y optimizar con TensorBoard.
- Seguimiento de centroides con OpenCV.
- Crear un ciclo de entrenamiento.

¿Qué compone el entrenamiento?

LabelMap

TFRecords Train

Modelo pre-entrenado de TF

Model Config pipeline.config

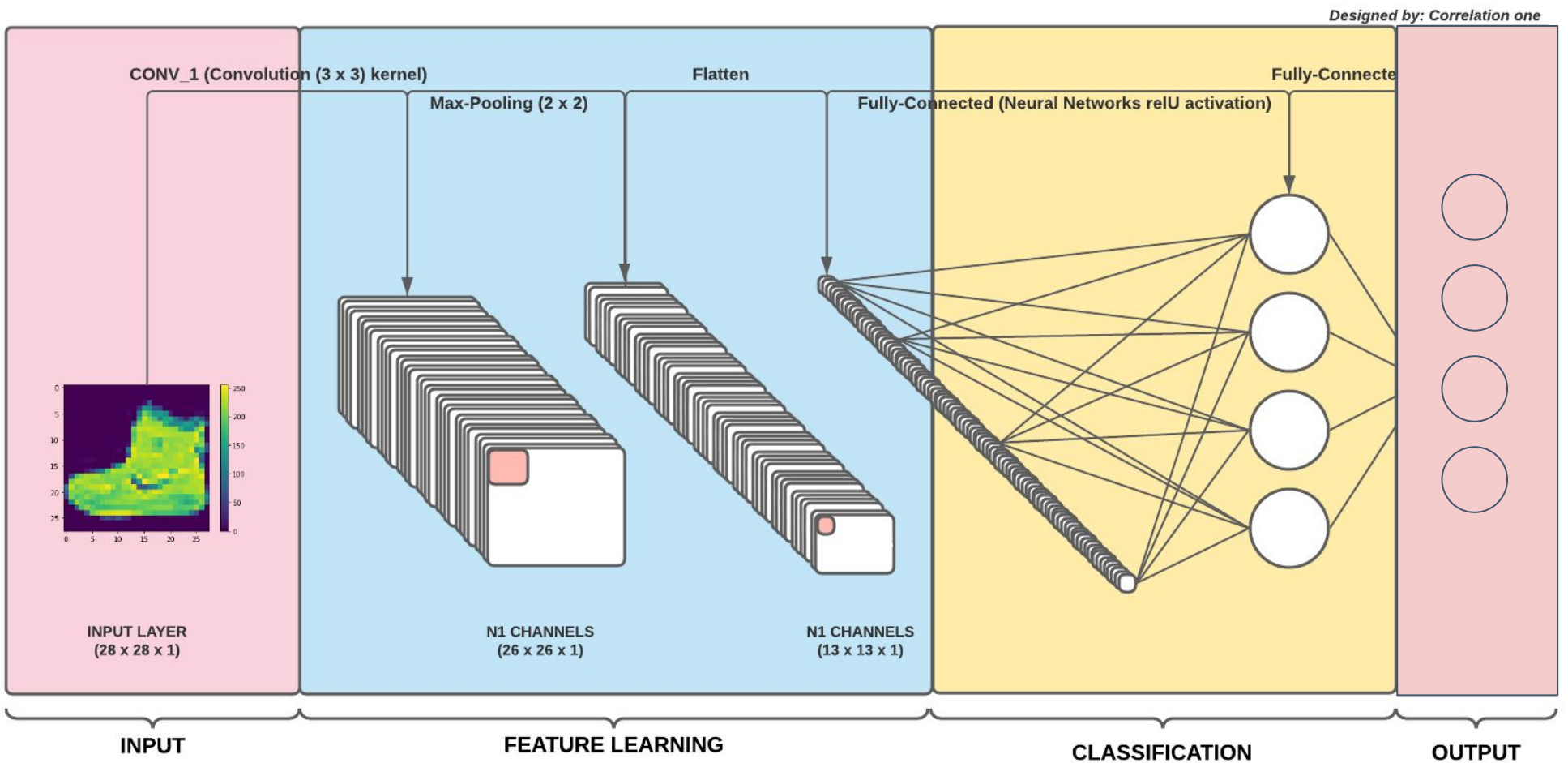
TFRecords Test

Transferir características principales

Inicialización de pesos

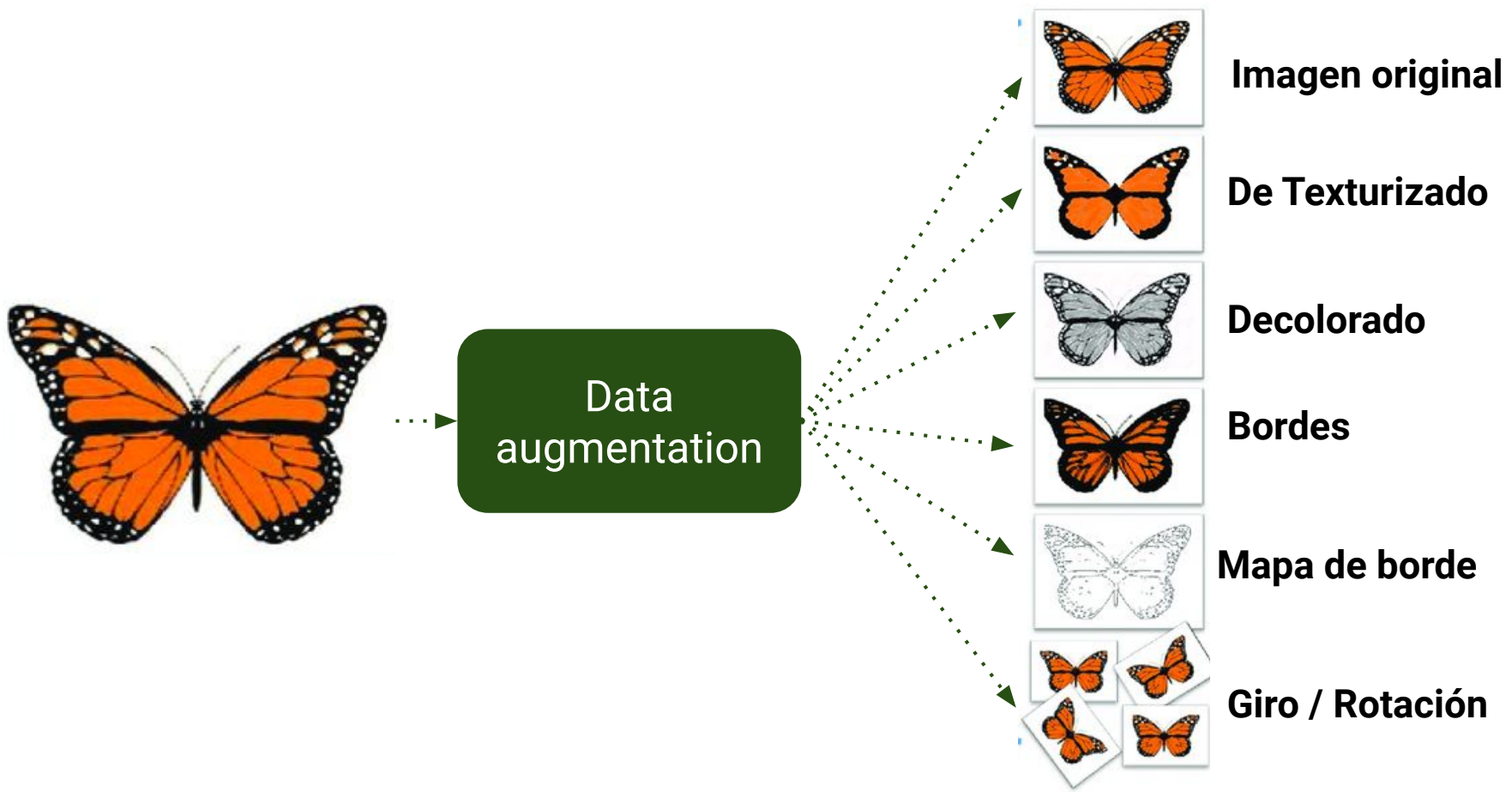
Inicialización de pesos

Aprendizaje por transferencia



Balanceo y data augmentation

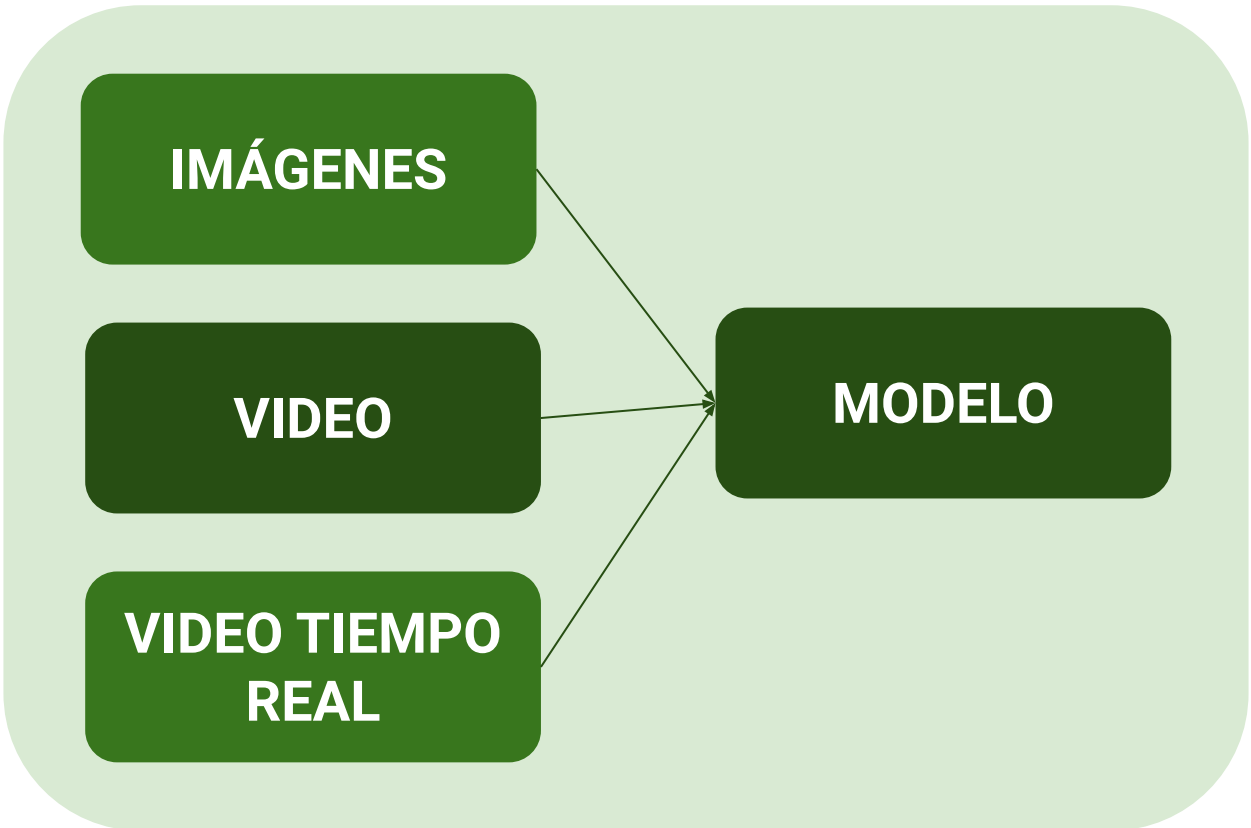
Data augmentation



Entrena, evalúa y optimiza con TensorBoard

Validación tu modelo en un entorno de ejecución

¿Inferencia de tu modelo?



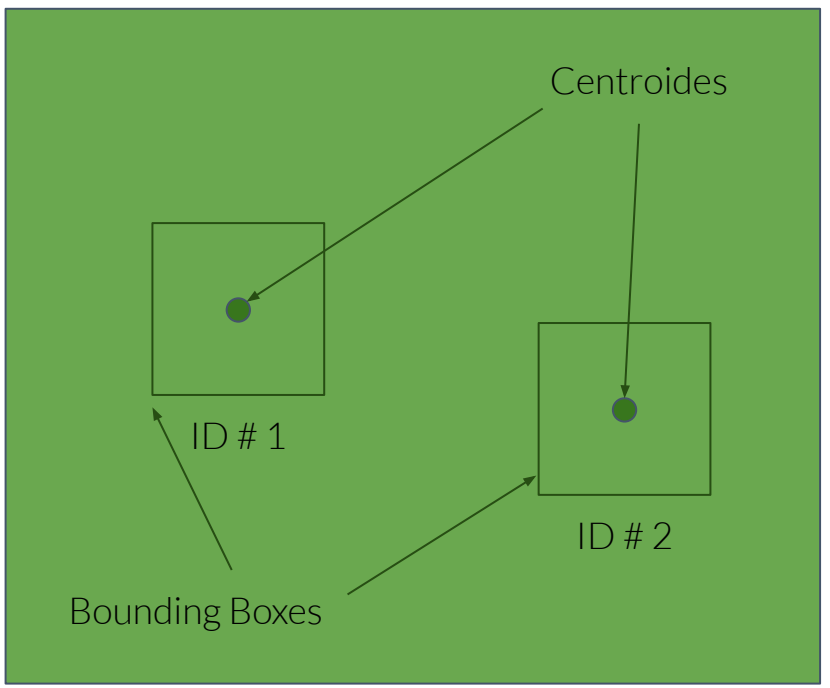
Re-entrenamiento del modelo

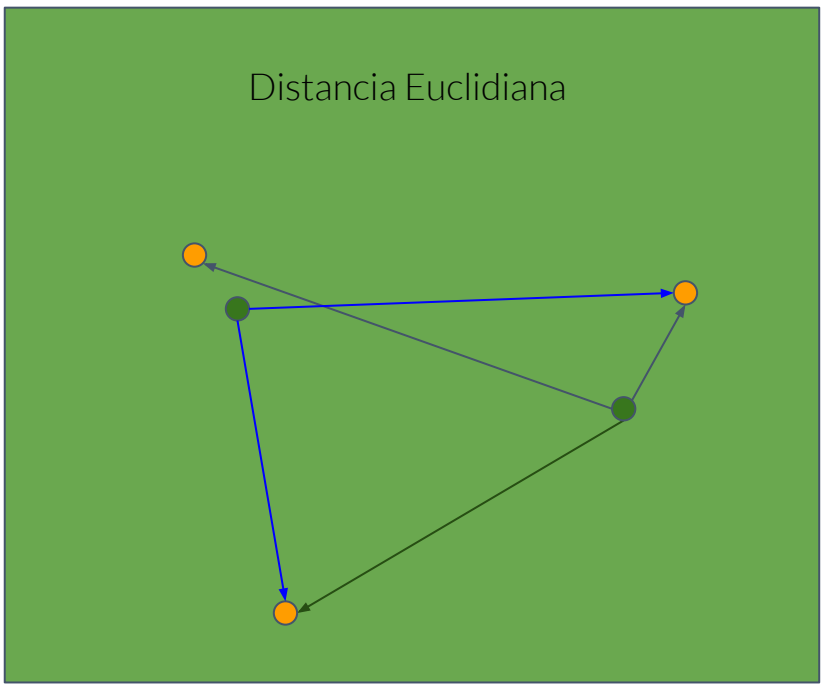
Repite el proceso

1. Descargar la nueva base de datos.
2. Limpiarla manualmente.
3. Utilizar el script distribucion-bases-datos y dividir la base de datos.
4. LinkedAI → Etiquetar.
5. Descargar el el JSON.
6. Utilizar el script json-tfRecord
7. Entrenamiento-modelos (Modelo, pbtxt).

Seguimiento de centroides con OpenCV

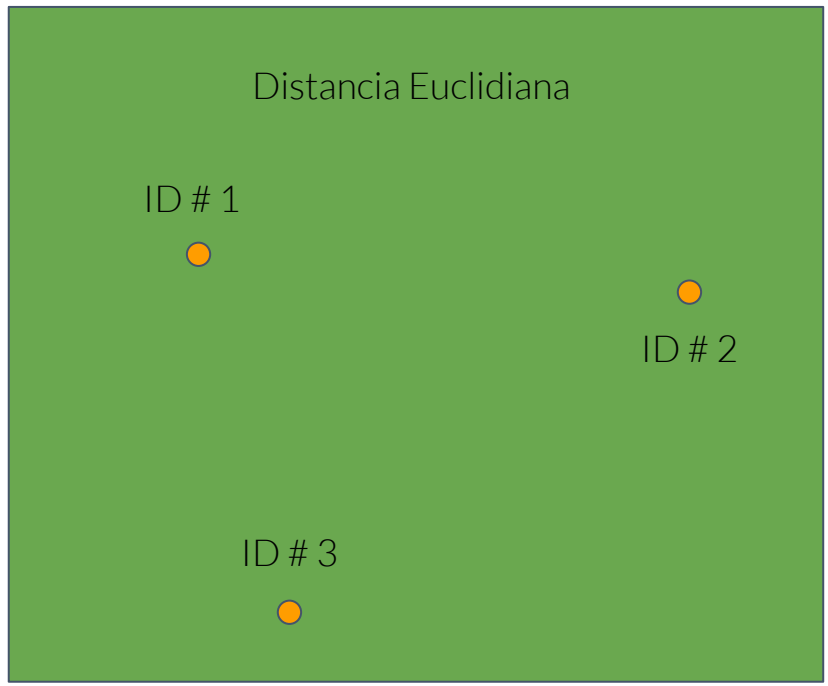
¿Cómo funciona?





Referencias. Rosebrock, Adrian (23 de Julio de 2018). Simple object tracking with OpenCV. pyimagesearch. https://www.pyimagesearch.com/2018/07/23/simple-object-tracking-with-opencv/

¿Cómo funciona?



Referencias. Rosebrock, Adrian (23 de Julio de 2018). Simple object tracking with OpenCV. pyimagesearch. https://www.pyimagesearch.com/2018/07/23/simple-object-tracking-with-opencv/

Crea un ciclo de entrenamiento de tu modelo

DevOps

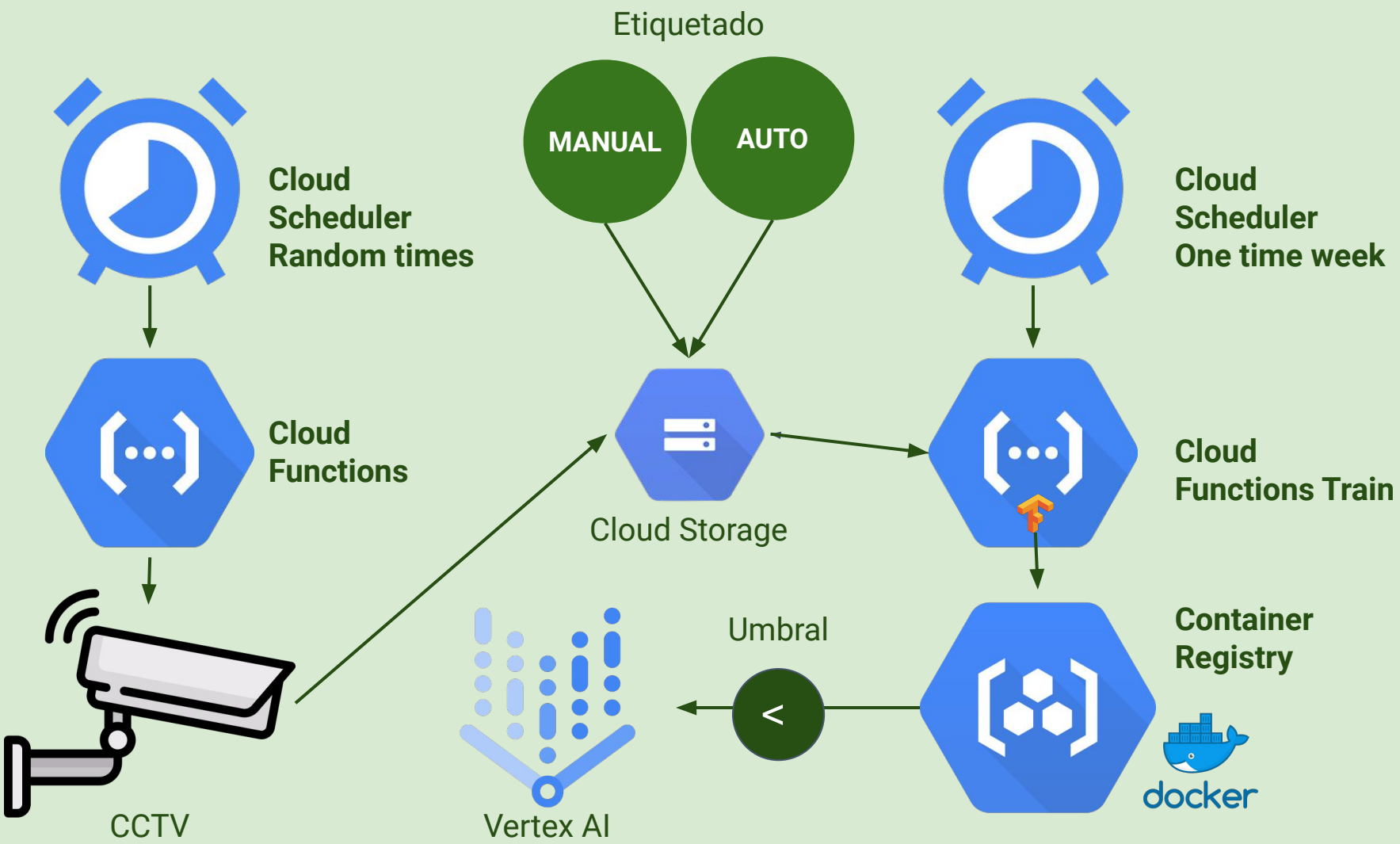
Optimiza desde modelo de negocio

1. Obtener retroalimentación de los clientes.
2. Sigue un proceso de re-entrenamiento: diario, semanal, mensual, trimestral o anual.
3. ¿Podemos automatizar la llegada de nuevos datos?

Optimiza desde modelo de negocio

4. ¿Puedo etiquetar mi base de datos de manera automática?
5. ¿Puedo re-entrenar mi modelo de manera automática?

Crea el ciclo



Prepara tu entorno en Google Cloud Platform

Ejemplos de producción

Nube

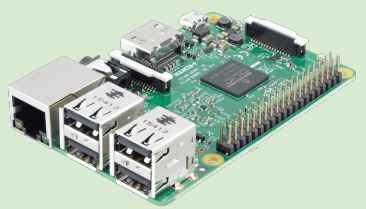
Google Cloud

Azure

aws

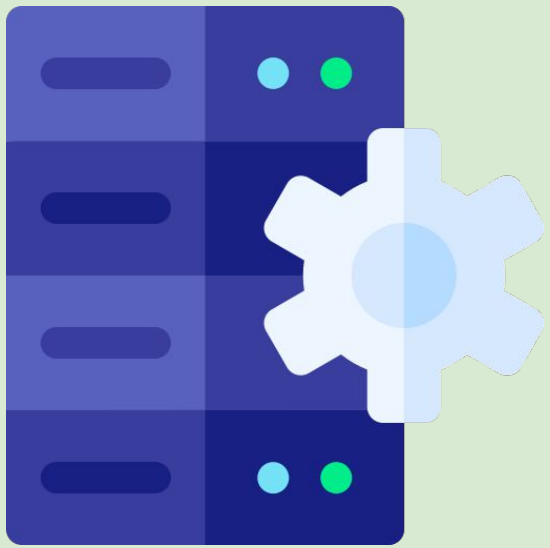
ORACLE CLOUD

Sistemas embebidos



Ejemplos de producción

Local



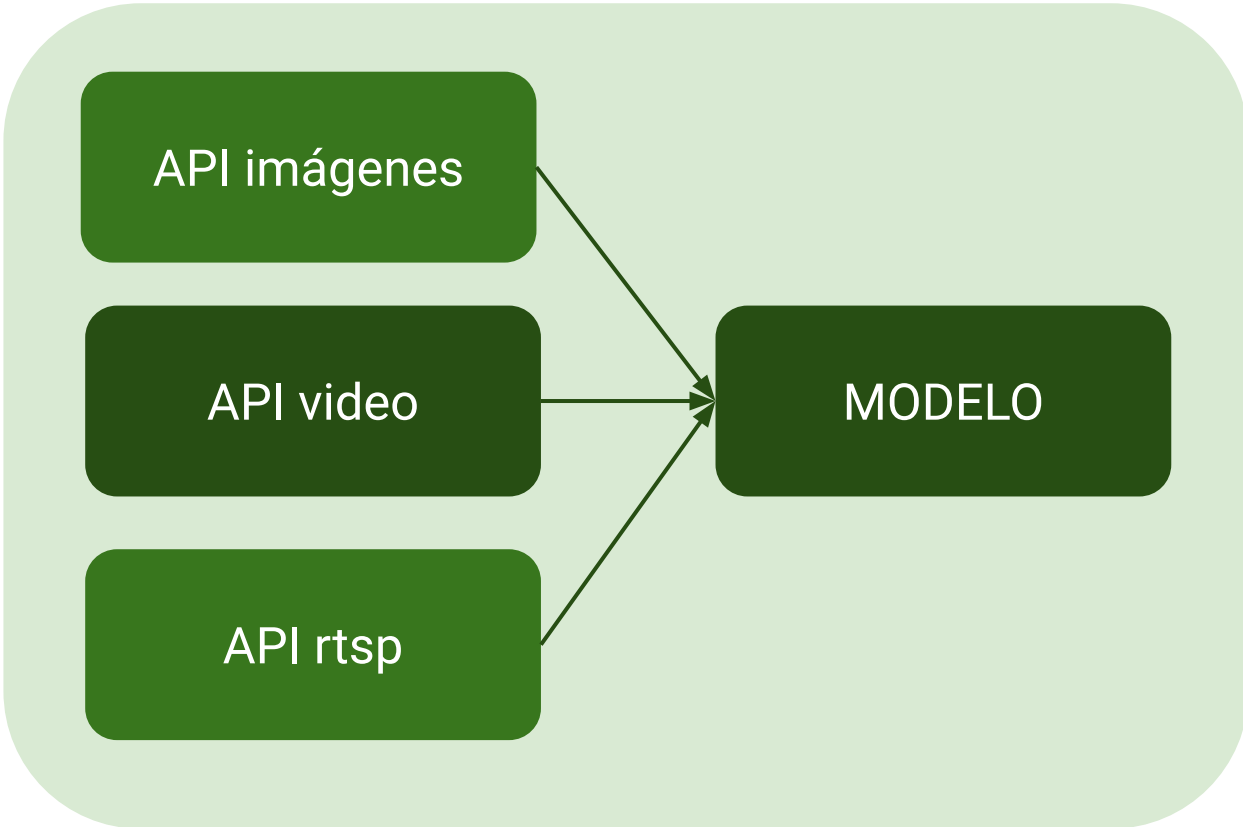
USB Accelerator

Google Cloud Platform

¿Cuáles servicios utilizaremos?

Carga, preprocesamiento y postprocesamiento de modelos

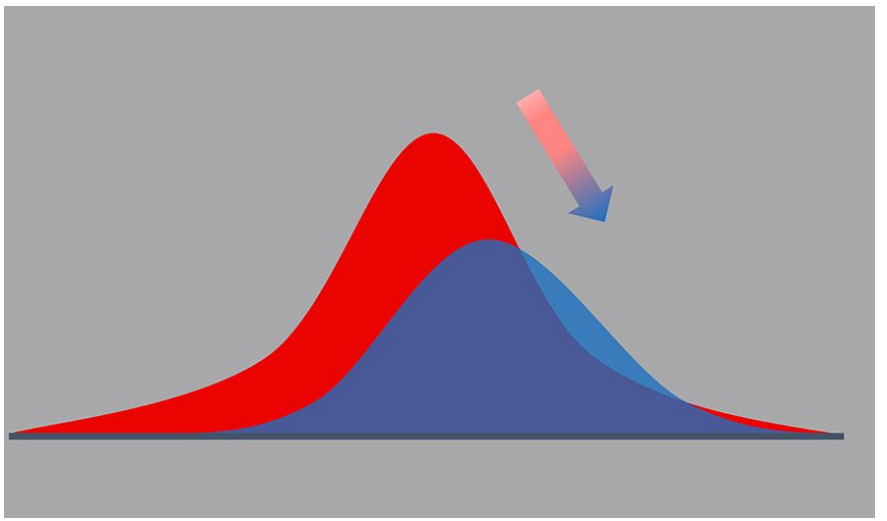
Inferencia de tu modelo en la nube



Consume tu modelo en producción

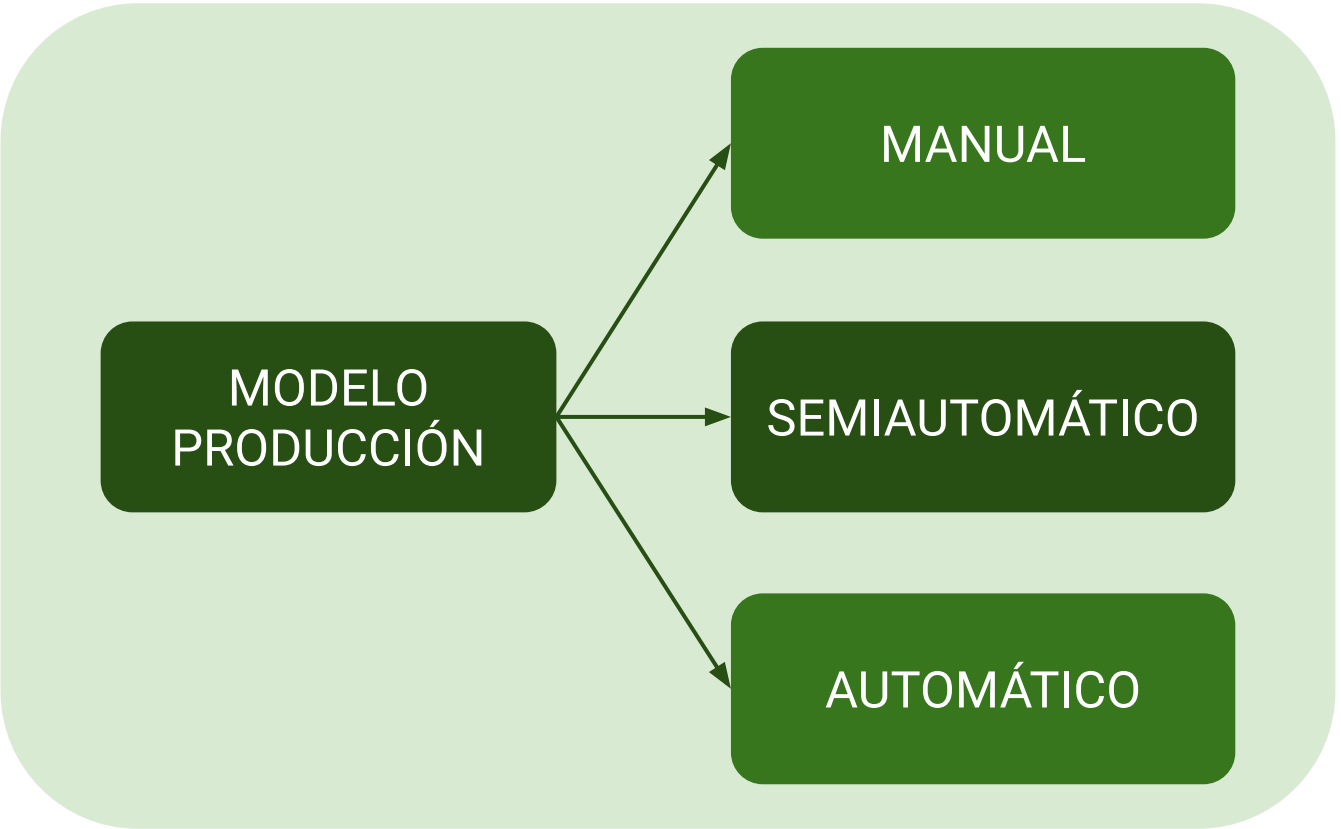
DataDrift

Nuevas imágenes



Sesgo en el modelo

Monitorizar tus modelos

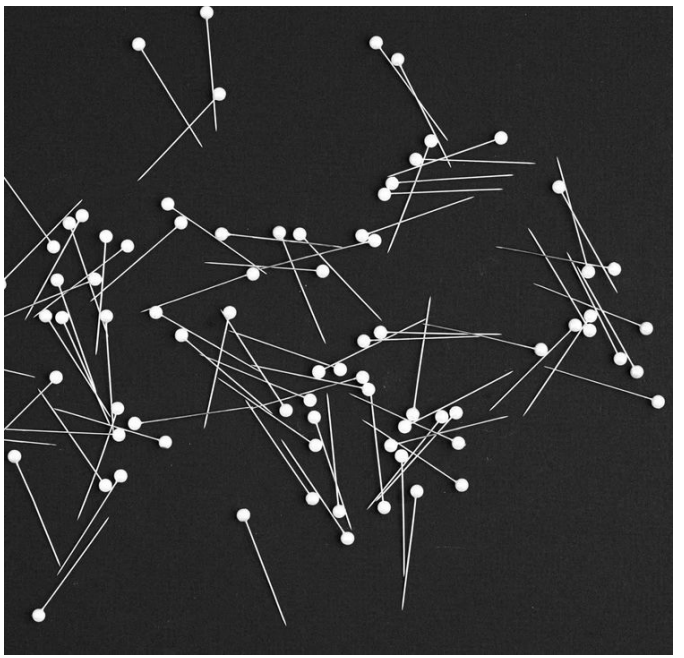


Recuerda apagar tus recursos de GCP

Siguientes pasos en inteligencia artificial

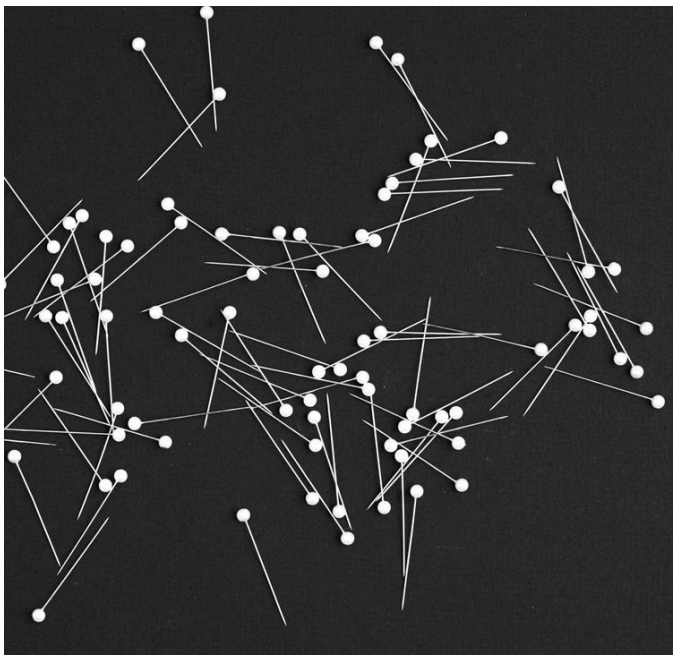
Qué más aprender

- Despliegue de modelos a producción.
- MLOps (data, model, code).
- Monitorizar modelo avanzado.



Qué más aprender

- Uso avanzado de GCP u otras herramientas cloud.
- Incluso crear tu propia startup con IA.



www.switchai.co