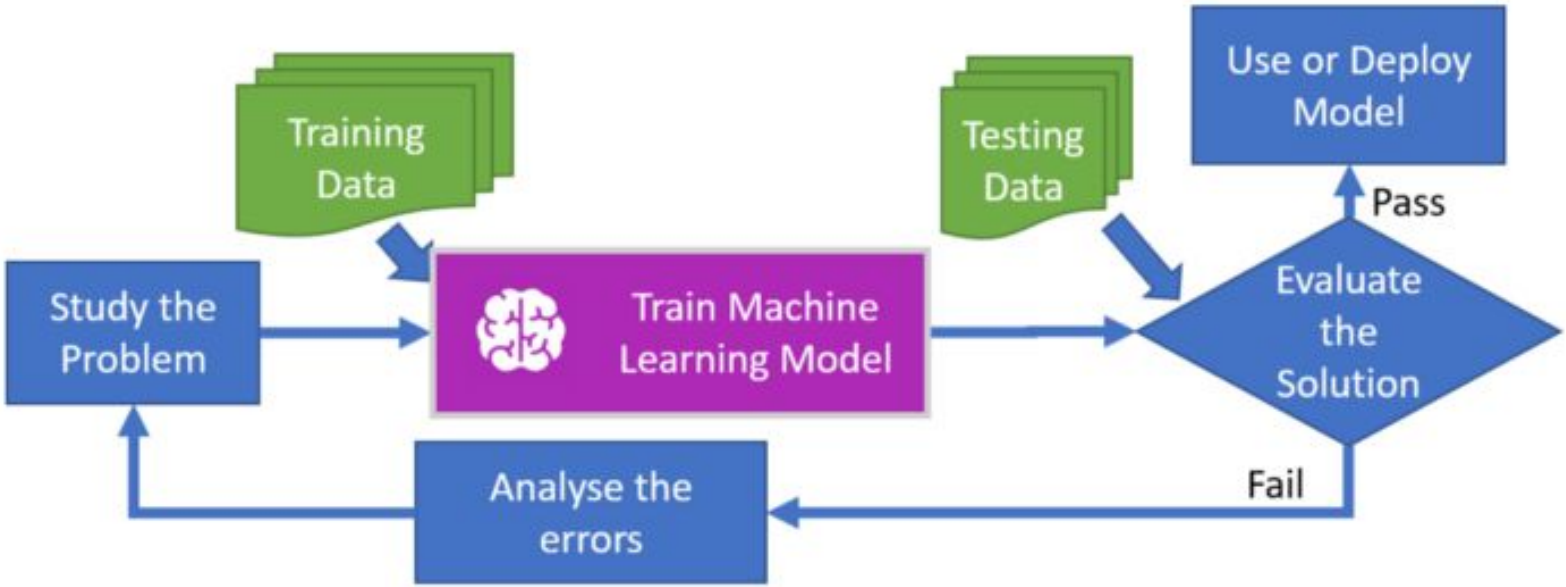
Deep Learning Aplicado à Visão para Processamento Real-Time em Borda

Gabriel Salomon

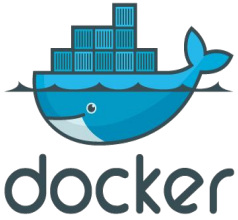
Estrutura da Apresentação

1. Fluxo de um projeto de Deep Learning
2. Introdução aos problemas clássicos de visão
3. Métodos para resolver os problemas clássicos
4. Desafios em sistemas reais
5. Diferença entre borda e cloud
6. Processamento e limitações da borda
7. Aplicações Reais

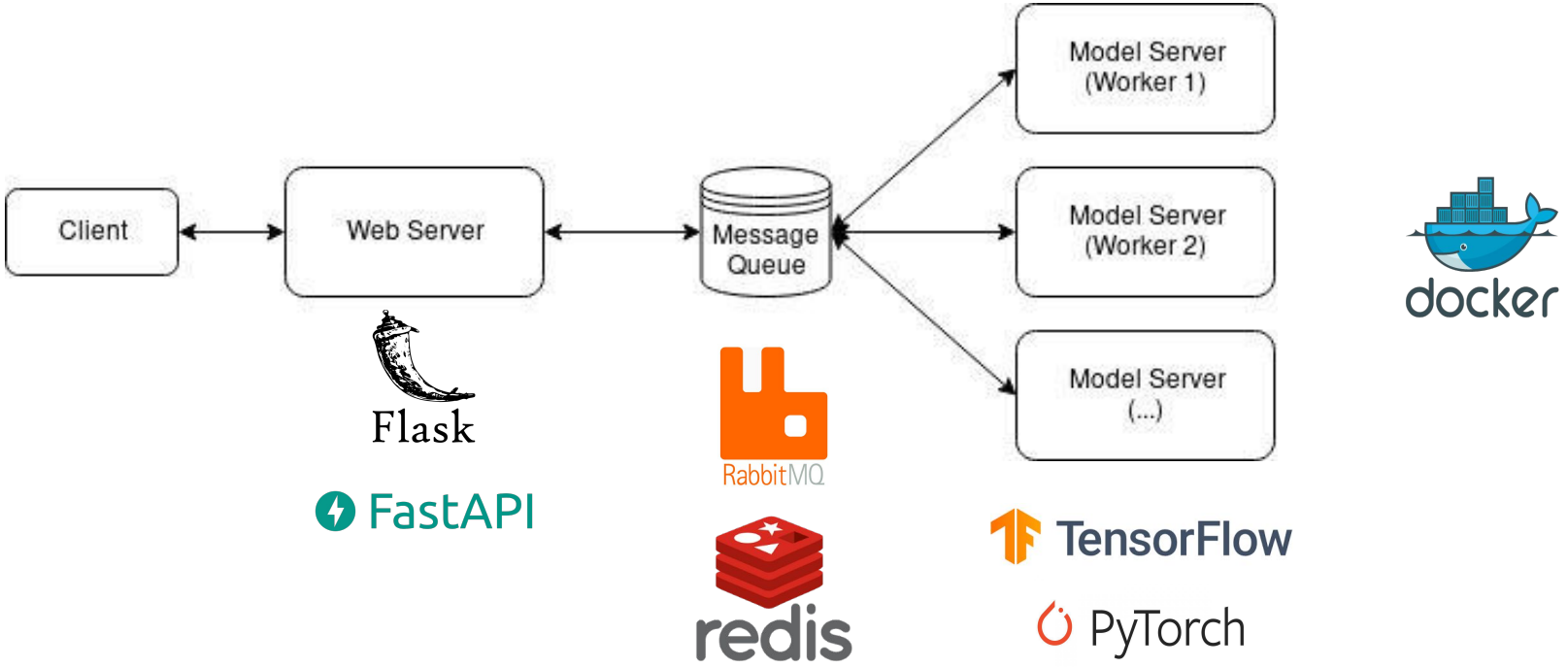
Fluxo de Projetos de DL



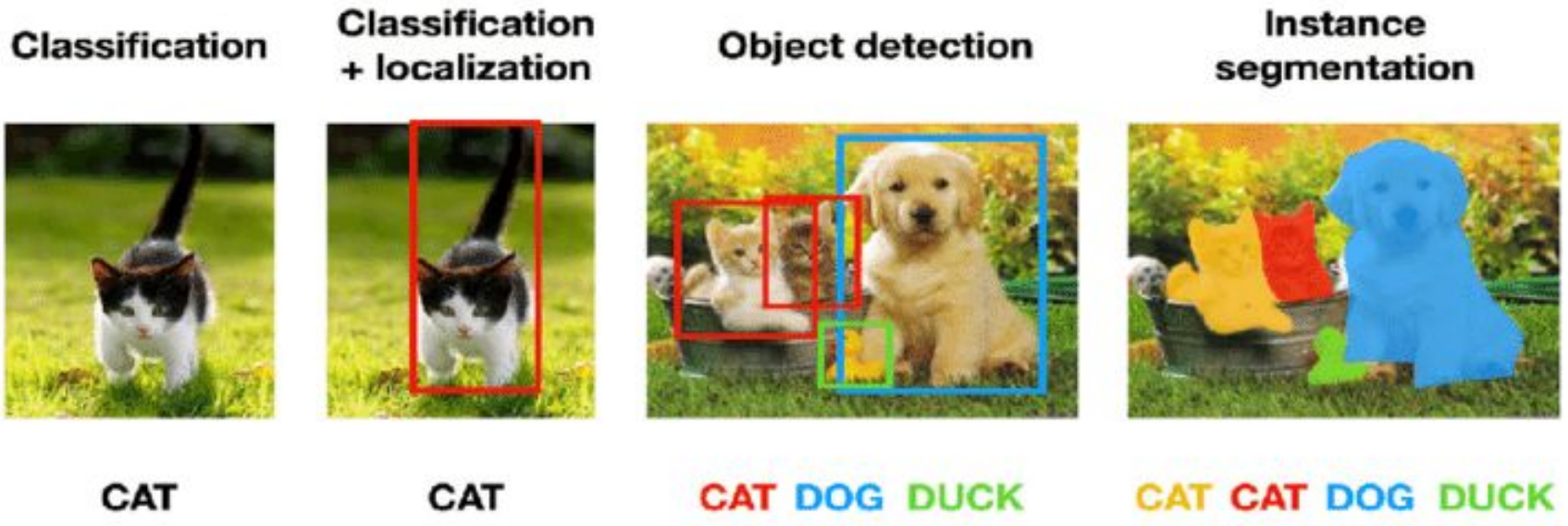
Tecnologias Úteis



Deploy de Modelos

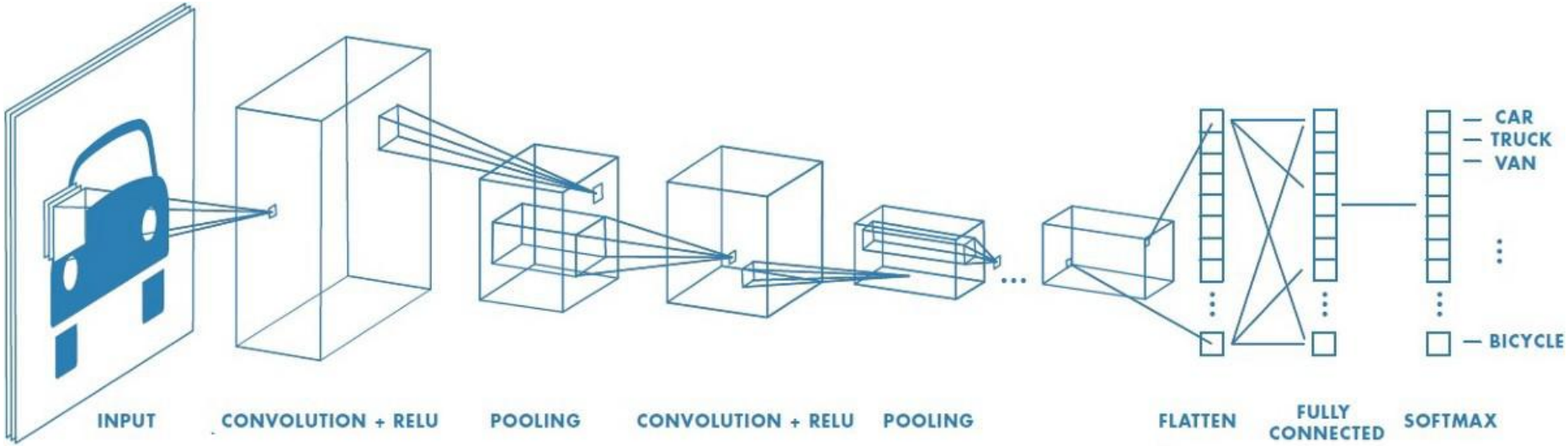


Problemas Clássicos de Visão



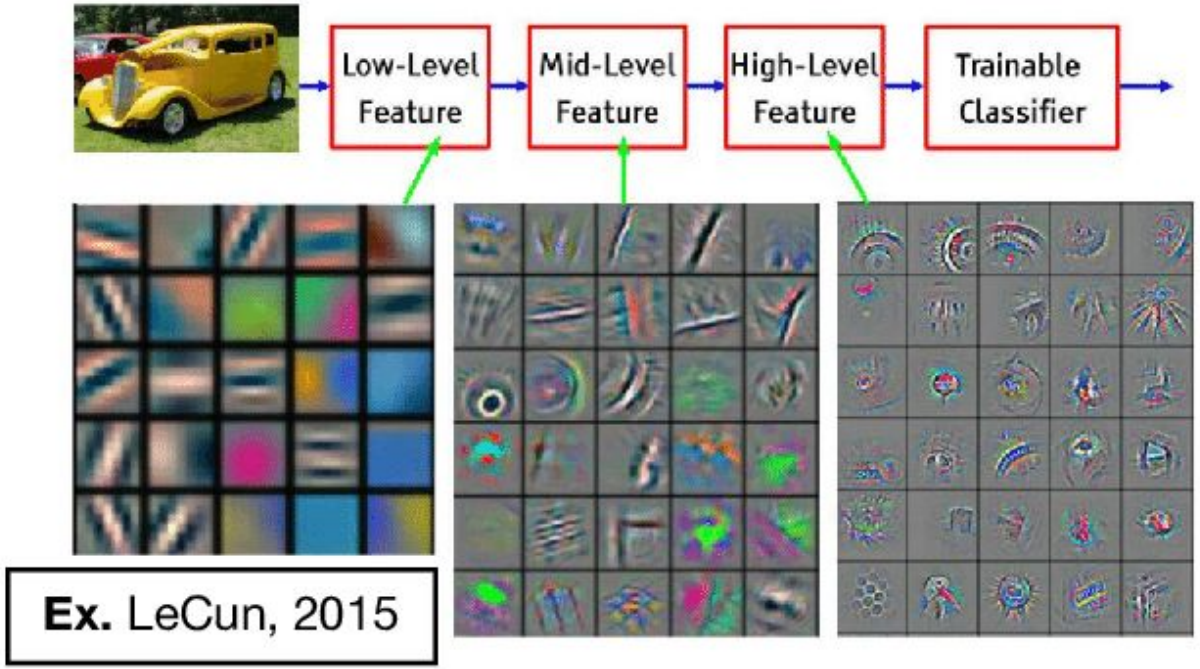
Classificação

Modelo CNN



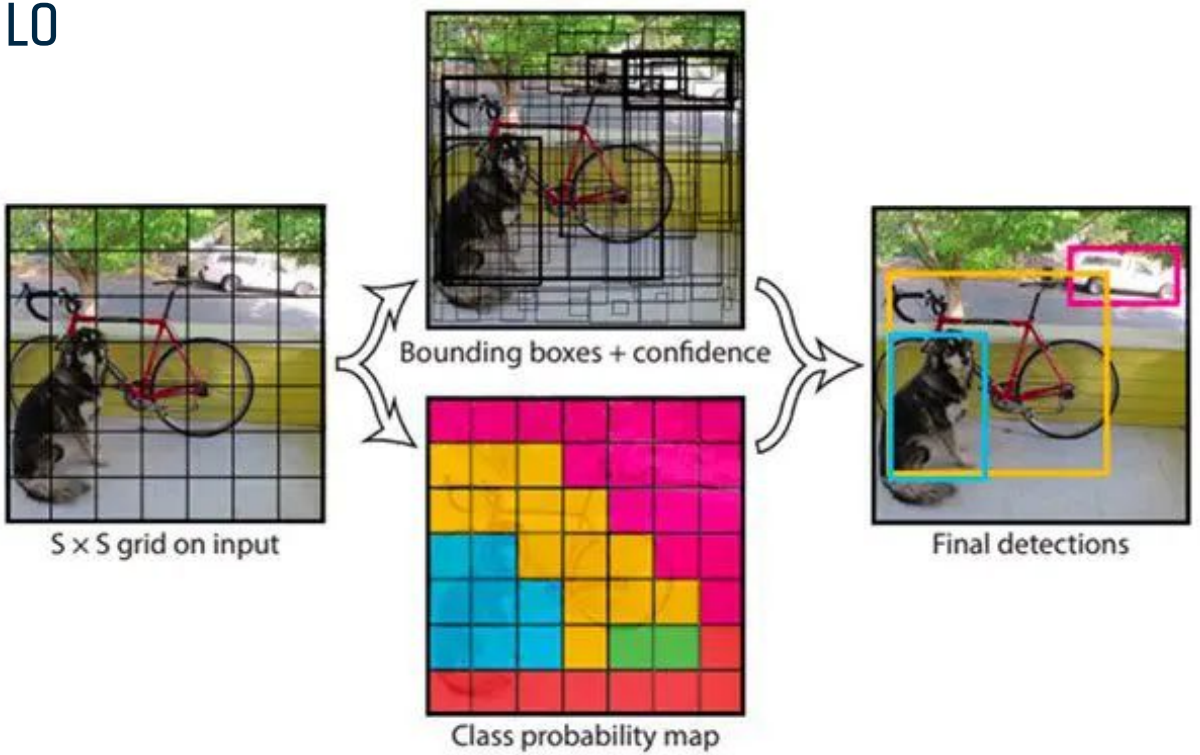
Classificação

Modelo CNN



Detecção de Objetos

Modelo YOLO



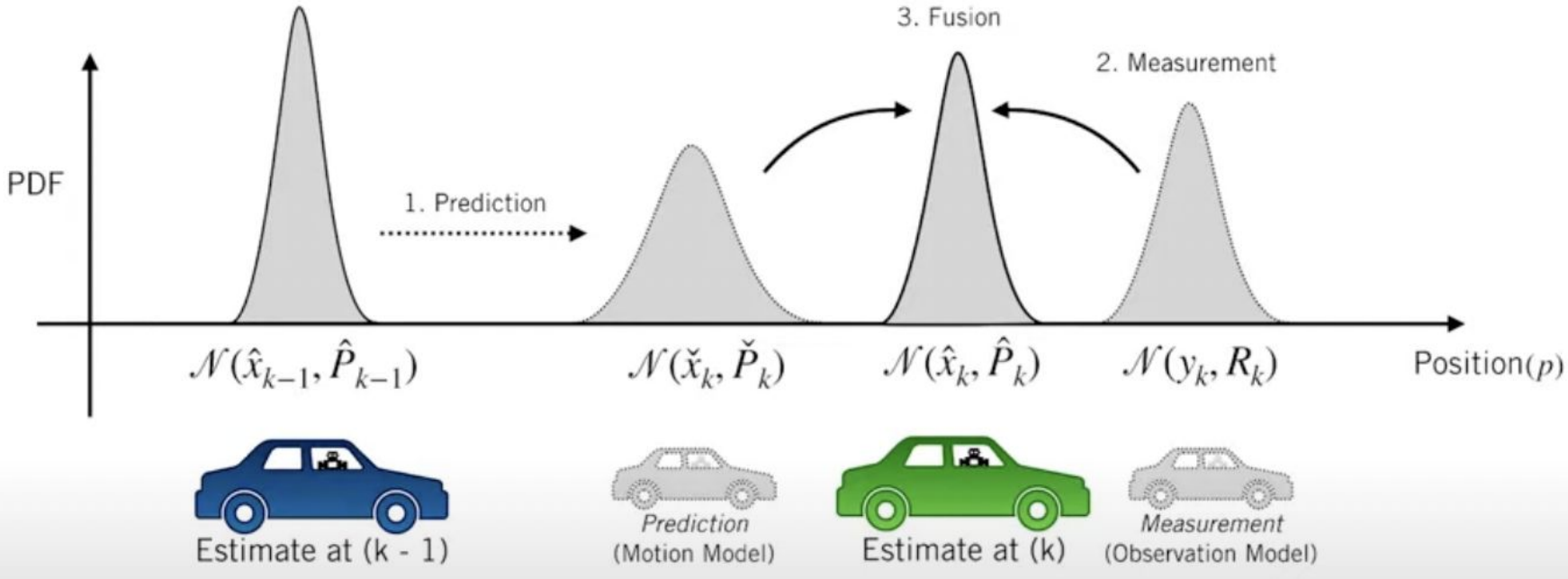
Tracking de Objetos

Pipeline de Tracking

Pipeline de Tracking – Exemplo

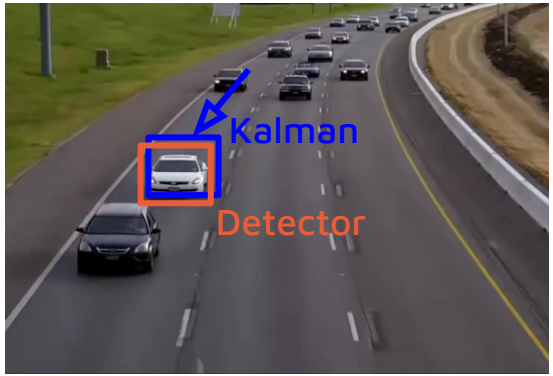
Tracking

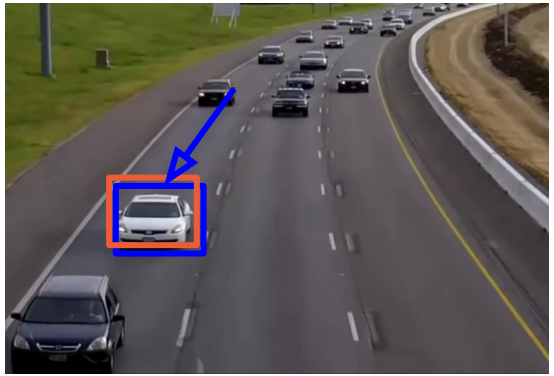
Filtro de Kalman

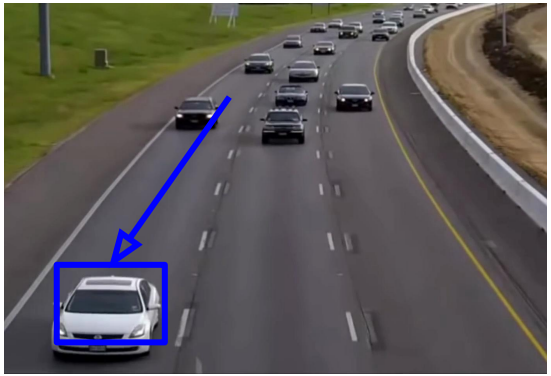


Tracking

Filtro de Kalman

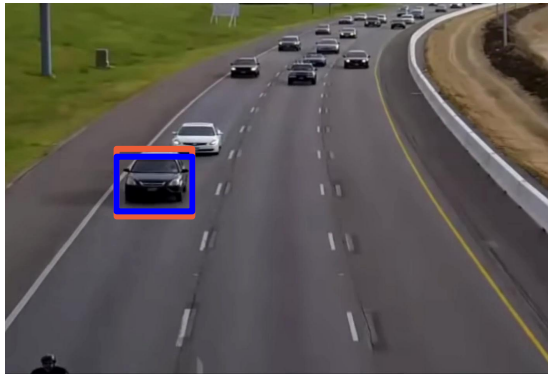


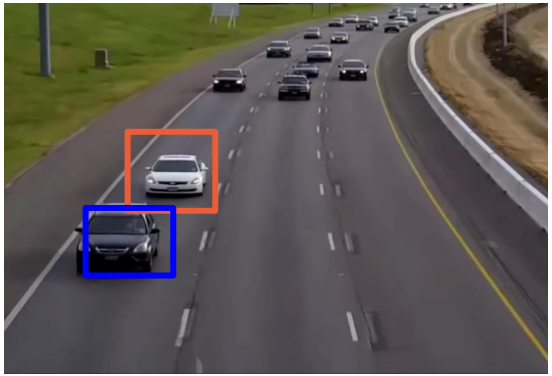


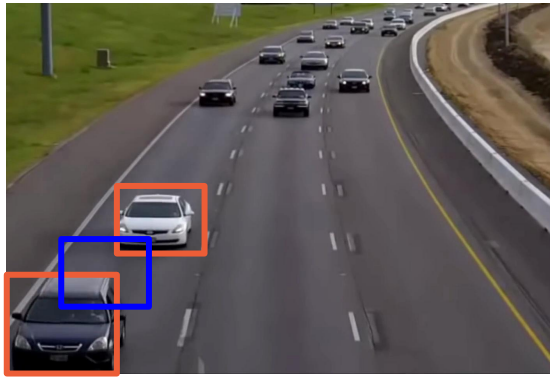


Tracking

Método Húngaro – Problema







Tracking

Método Húngaro – Matching

- Foca em minimizar uma função de custo
- O custo pode ser a distância entre as detecções e as estimativas de Kalman (IoU, Euclidiana, Manhattan)
- Se há mais detecções do que estimativas há um possível objeto novo
- Se há menos detecções do que estimativas um objeto saiu de cena

Tracking

Método Húngaro - Matching

Step 1

$$\begin{bmatrix} 0 & 1 & 2 \\ 0 & 2 & 4 \\ 0 & 3 & 6 \end{bmatrix}$$

Step 2

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 2 & 4 \end{bmatrix}$$

Step 3

$$\begin{bmatrix} \cancel{0} & \cancel{0} & \cancel{0} \\ \cancel{0} & 1 & 2 \\ \cancel{0} & 2 & 4 \end{bmatrix}$$

Step 4

$$\begin{bmatrix} 2 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 2 & 4 \end{bmatrix}$$

Step 5

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 3 \end{bmatrix}$$

Step 6

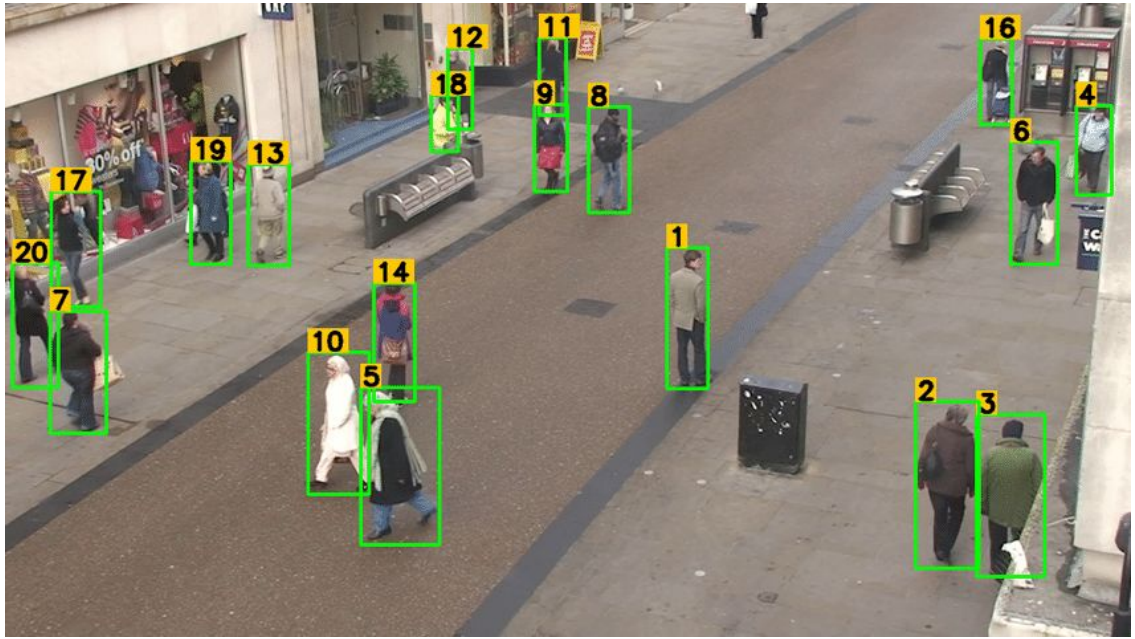
$$\begin{bmatrix} \cancel{1} & \cancel{0} & \cancel{0} \\ \cancel{0} & \cancel{0} & \cancel{1} \\ \cancel{0} & \cancel{1} & \cancel{3} \end{bmatrix}$$

Tracking

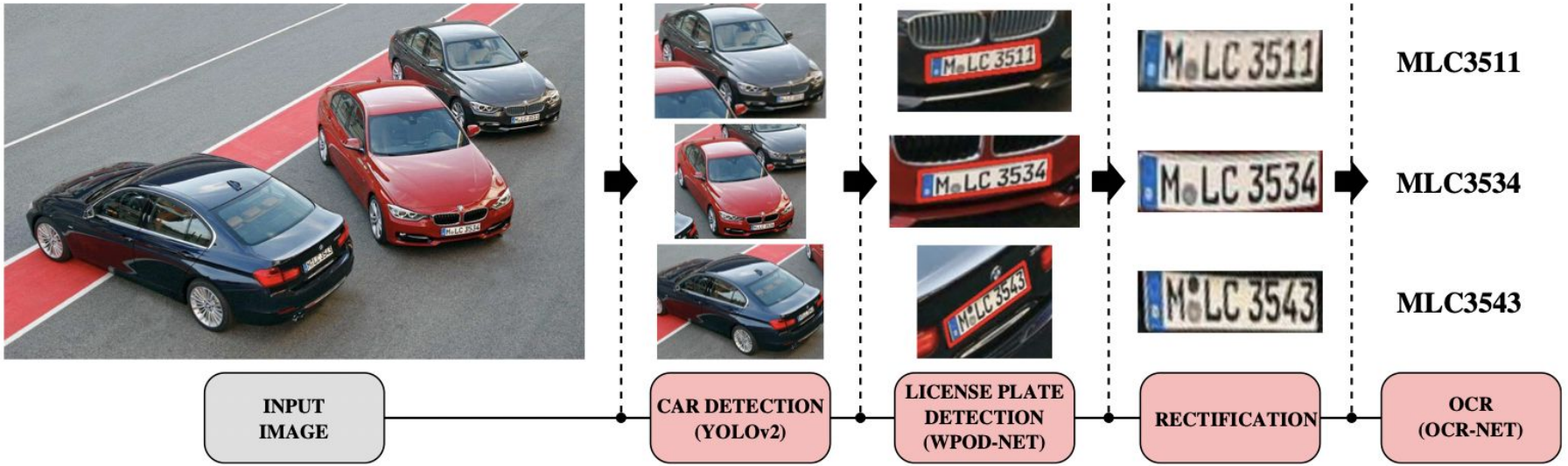
Tracklets

ID Pessoa: 7
Vezes Detectado: 2
Strikes: 0
Última Coord.: 200,30,80,30

ID Pessoa: 21
Vezes Detectado: 25
Strikes: 5
Última Coord.: 200,10,80,30

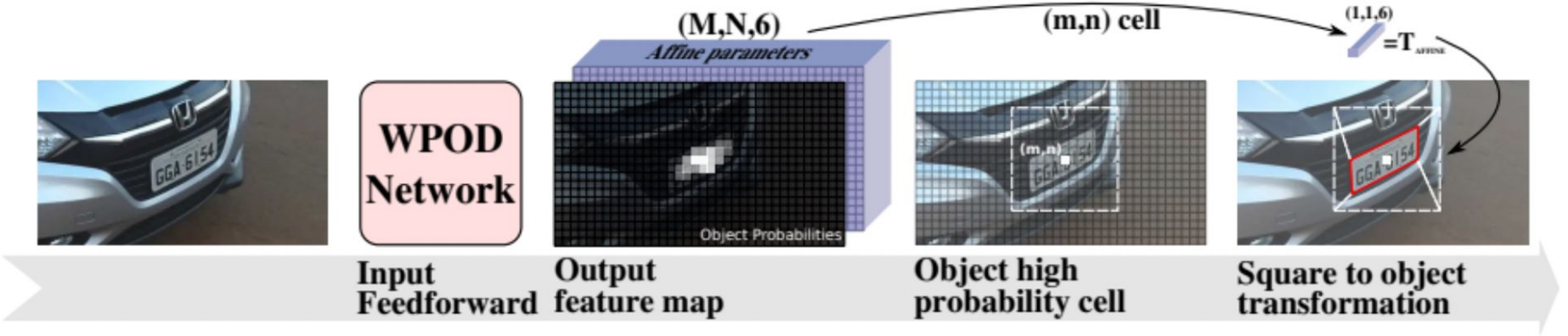


Leitura de Placas – ALPR



Leitura de Placas – ALPR

Detecção de Placa



Leitura de Placas – ALPR

Desafios Práticos – Câmera

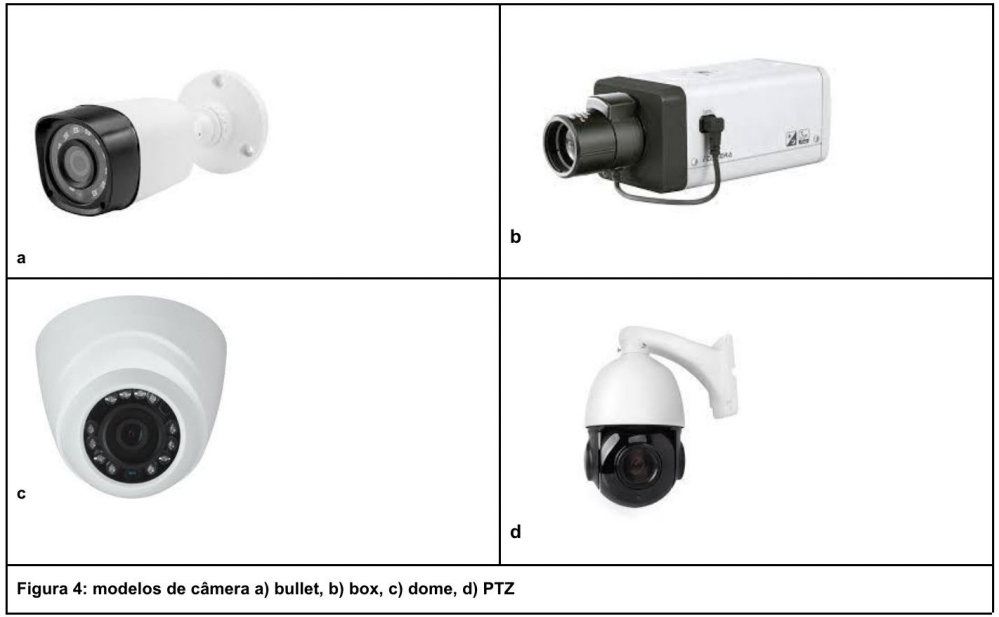
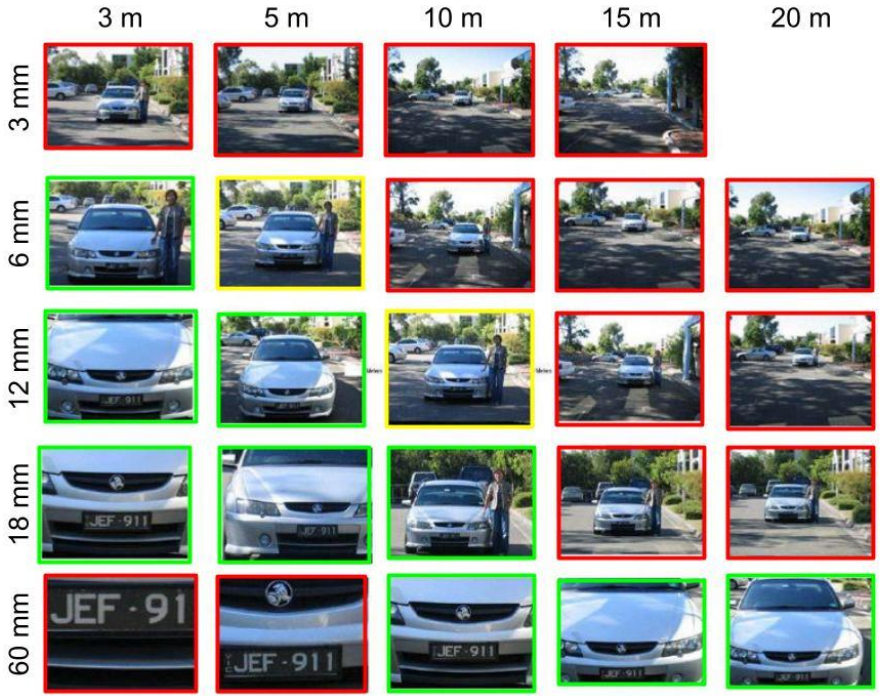


Figura 4: modelos de câmera a) bullet, b) box, c) dome, d) PTZ

Leitura de Placas – ALPR

Desafios Práticos – Distância entre Câmera e Veículo



Leitura de Placas – ALPR

Desafios Práticos – Iluminação

Camera without BLC

Camera with BLC

Leitura de Placas – ALPR

Desafios Práticos – Iluminação

HLC OFF

HLC ON

Leitura de Placas – ALPR

Desafios Práticos – Velocidade de Obturador

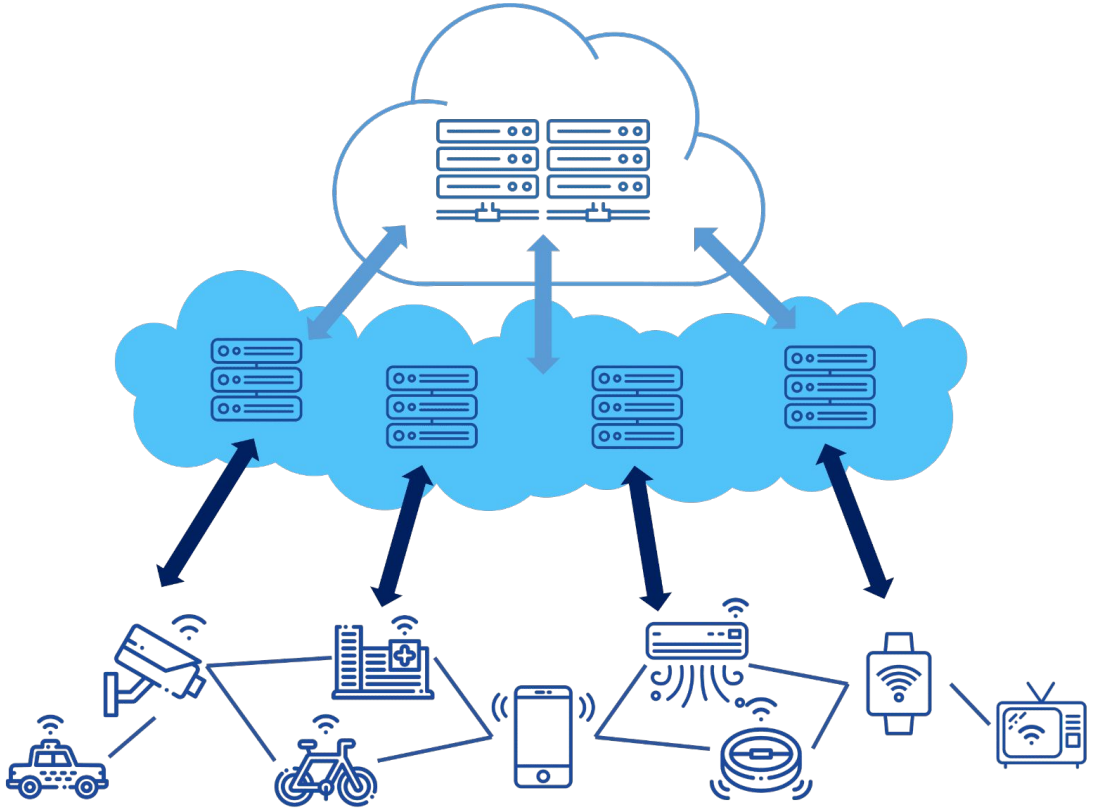
License Plate Recognition Camera Shot

1/1000

Regular Camera -- no-shutter speed

1/32

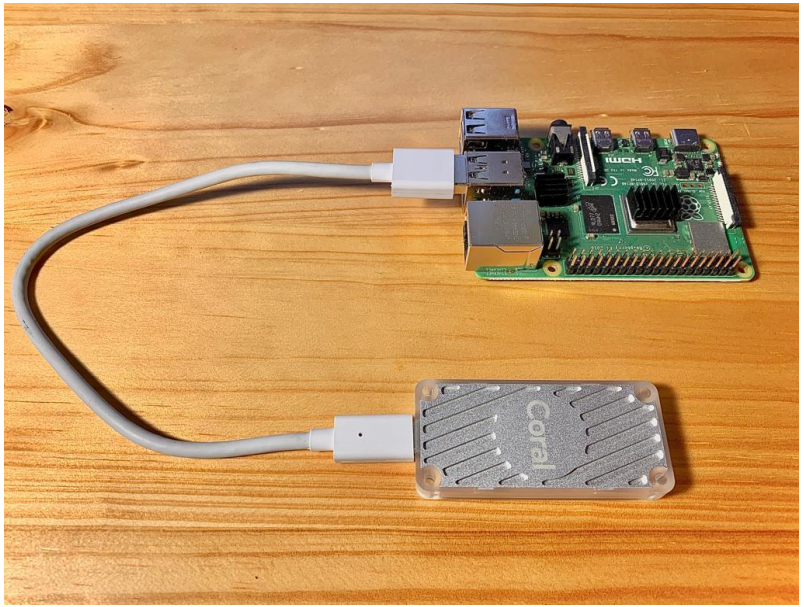
Computação em Borda (Edge) vs. Cloud



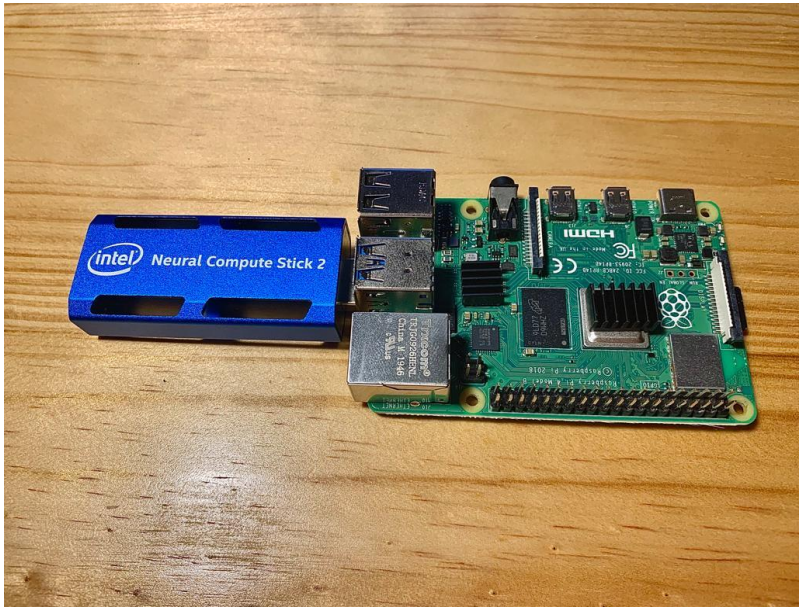
Borda vs. Cloud

Borda	Cloud
Computação ocorre em dispositivos IoT	Computação ocorre em máquinas mais potentes
Menor custo de processamento	Maior custo de processamento
Custo de manutenção e substituição de equipamentos	Custo previsível e flexível
Processamento limitado	Processamento "ilimitado"
Solução boa o bastante	Solução ótima (mas custosa)

O que são VPUs/NPUs/TPUs?



Google Coral



Intel NCS2

O que são VPUs/NPUs/TPUs?

Vision Processing Unit, Neural Processing Unit, Tensor Processing Unit.

GPUs podem ser utilizadas para treino, inferência, processamento em geral.

VPUs, NPUs, TPUs muitas vezes realizam apenas inferência.

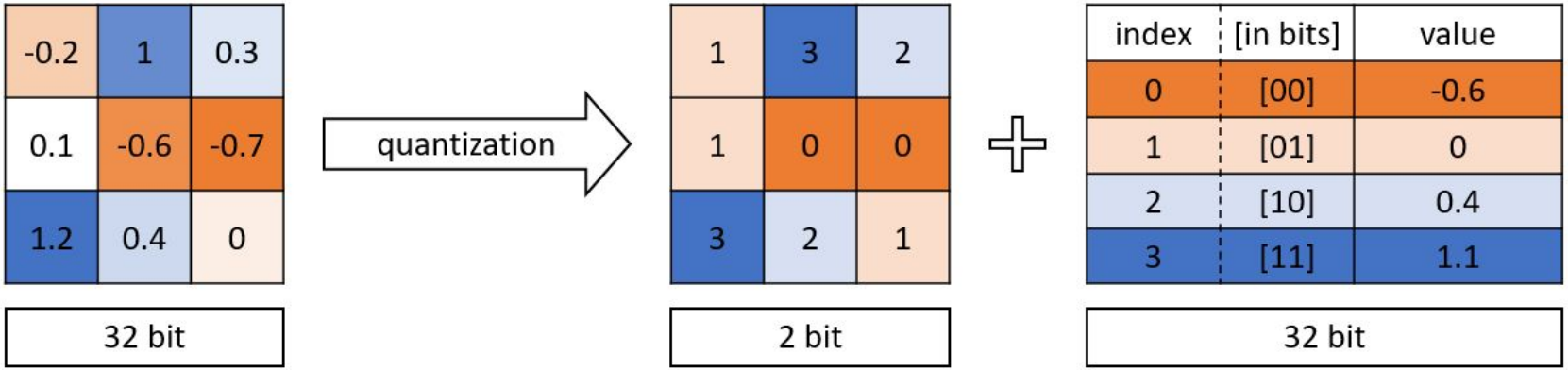
Desempenho de Modelos em VPUs/NPUs

Modelo	Framework	Raspberry Pi (TF-Lite)	Raspberry Pi Intel Neural Stick 2	Raspberry Pi Google Coral USB
EfficientNet-B0 (224x224)	TensorFlow	14.6 FPS (Pi 3) 25.8 FPS (Pi 4)	95 FPS (Pi 3) 180 FPS (Pi 4)	105 FPS (Pi 3) 200 FPS (Pi 4)
ResNet-50 (244x244)	TensorFlow	2.4 FPS (Pi 3) 4.3 FPS (Pi 4)	16 FPS (Pi 3) 60 FPS (Pi 4)	10 FPS (Pi 3) 18.8 FPS (Pi 4)
MobileNet-v2 (300x300)	TensorFlow	8.5 FPS (Pi 3) 15.3 FPS (Pi 4)	30 FPS (Pi 3)	46 FPS (Pi 3)
SSD Mobilenet-V2 (300-300)	TensorFlow	7.3 FPS (Pi 3) 13 FPS (Pi 4)	11 FPS (Pi 3) 41 FPS (Pi 4)	17 FPS (Pi 3) 55 FPS (Pi 4)
Tiny YOLO V3 (416x416)	Darknet	0.5 FPS (Pi 3) 1 FPS (Pi 4)	-	-
VGG-19 (224x224)	MXNet	0.5 FPS (Pi 3) 1 FPS (Pi 4)	5 FPS	-

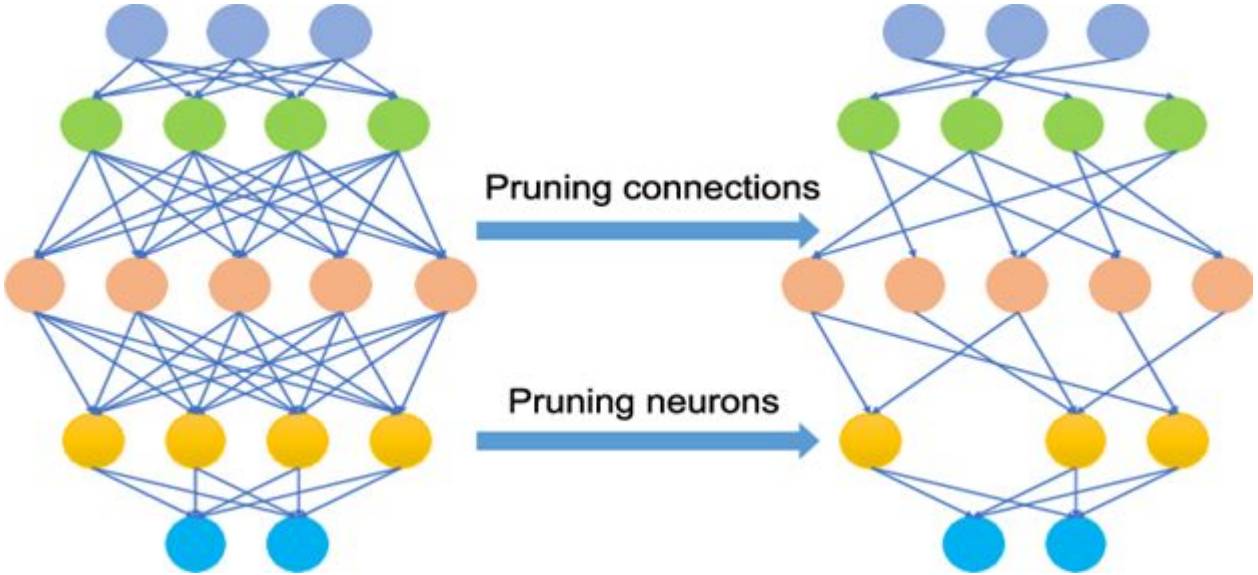
Como aumentar a performance?

- Quantizar os modelos (32bits -> 8bits)
- Pruning dos Modelos (Poda)
- Utilizar modelos otimizados para borda (EfficientNet, MobileNet)
- Utilizar frameworks e bibliotecas otimizados para inferência (ONNX, OpenVINO)
- Aceitar resultados sub-ótimos (modelos menores)

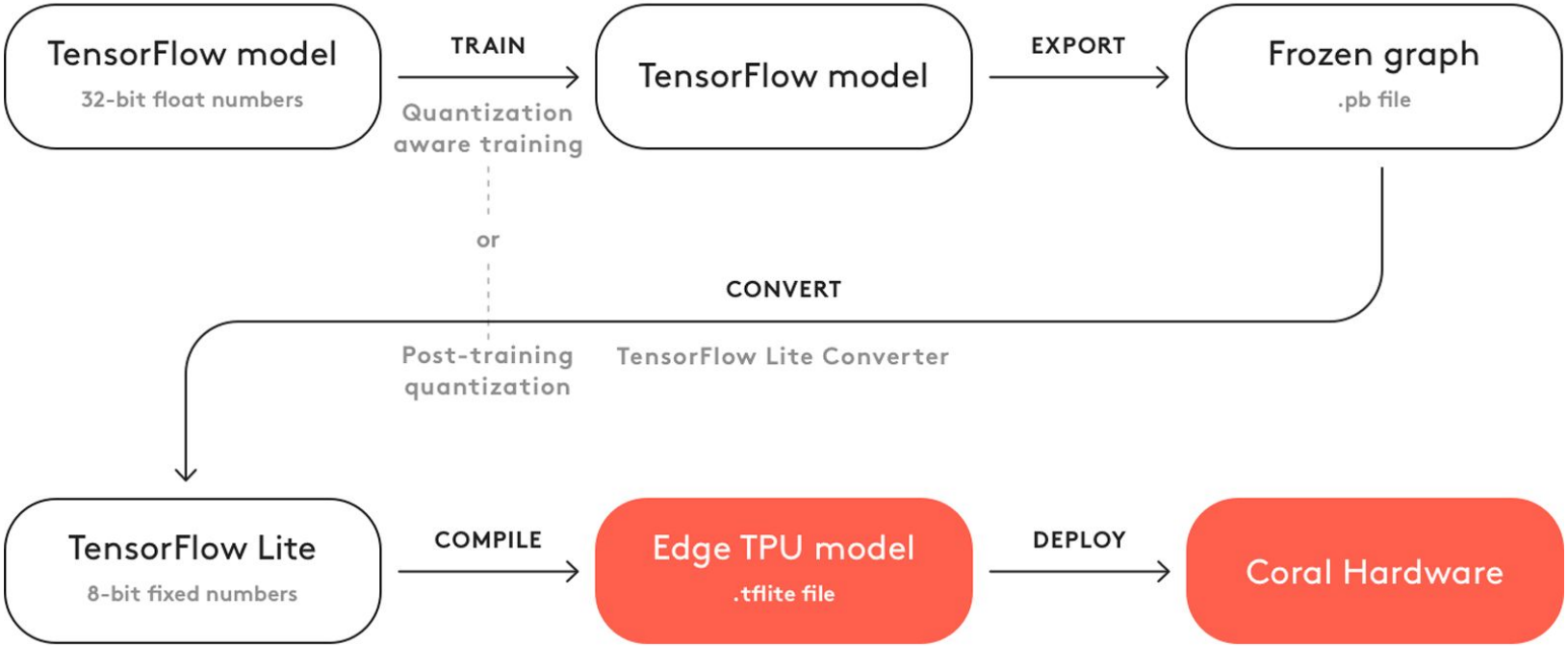
Quantização



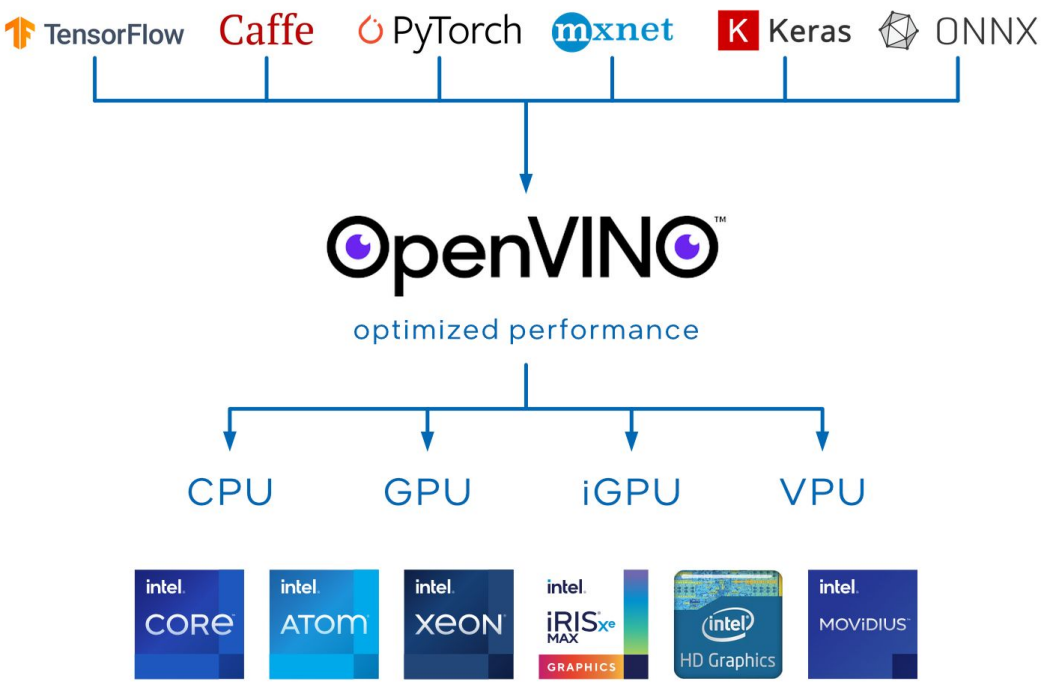
Pruning (Poda)



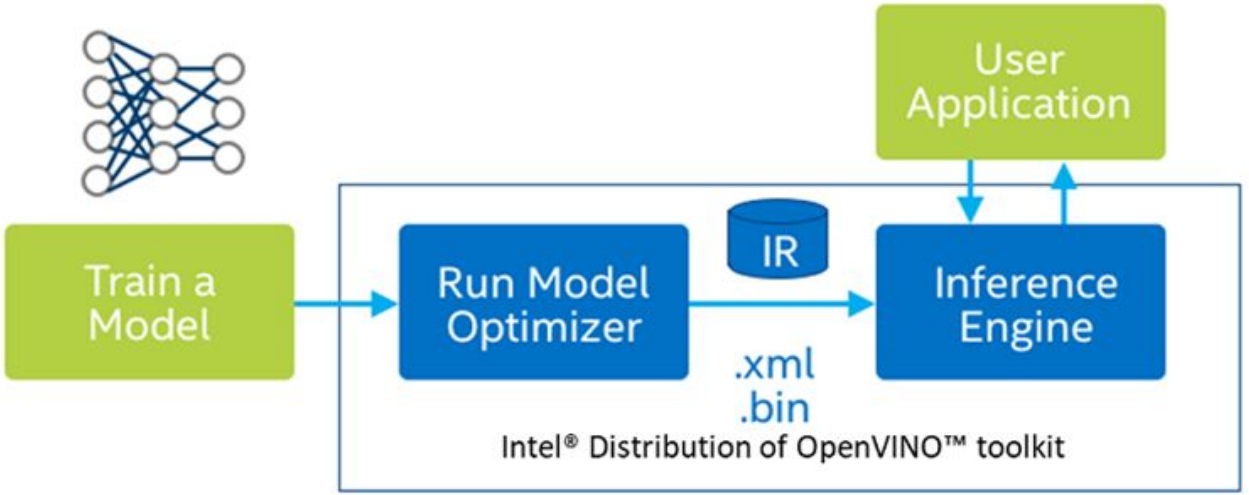
Deploy em TPU (Google Coral)



Deploy em NPU (Intel NCS2)



Deploy em NPU (Intel NCS2)



Problemas com VPUs/NPUs/TPUs

- Quantização reduz acurácia
- Memória limitada
- Limitações para múltiplos modelos paralelos
- Incompatibilidade com funções de ativação, modelos, bibliotecas
- Exigem re-treino para alguns casos

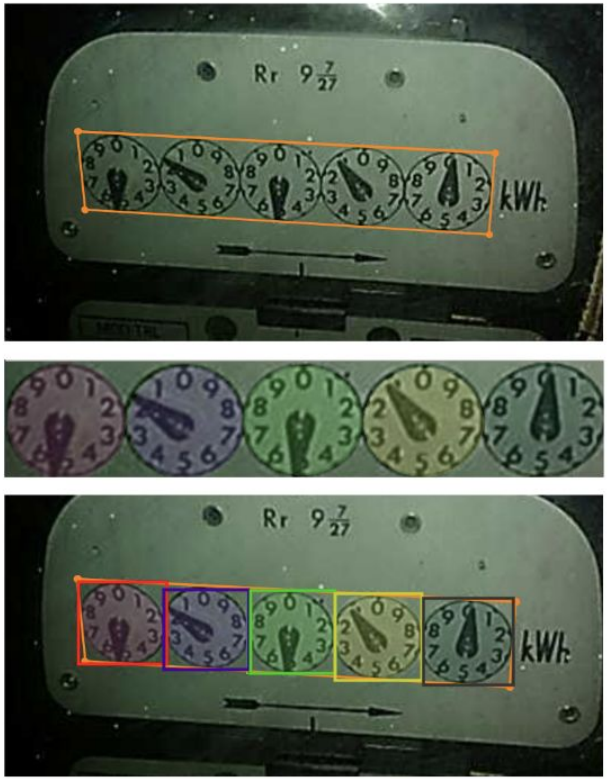
Problemas com VPUs/NPUs/TPUs

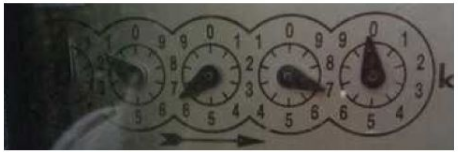
```
Number of operations that will run on Edge TPU: 133
Number of operations that will run on CPU: 72

Operator                       Count      Status

CONV_2D                        70         Mapped to Edge TPU
QUANTIZE                       3          Operation is otherwise supported, but not mapped due to some unspecified limitation
QUANTIZE                       15         Mapped to Edge TPU
RESIZE_NEAREST_NEIGHBOR        2          Operation version not supported
STRIDED_SLICE                  4          Only Strided-Slice with unitary strides supported
PAD                            6          Mapped to Edge TPU
LEAKY_RELU                     8          Operation not supported
TRANSPOSE                      3          Operation not supported
RESHAPE                        2          Mapped to Edge TPU
RESHAPE                        1          Operation is otherwise supported, but not mapped due to some unspecified limitation
MAX_POOL_2D                    3          Mapped to Edge TPU
ADD                            15         Mapped to Edge TPU
MUL                            8          Mapped to Edge TPU
HARD_SWISH                     51         Operation not supported
CONCATENATION                  14         Mapped to Edge TPU
```

Aplicações

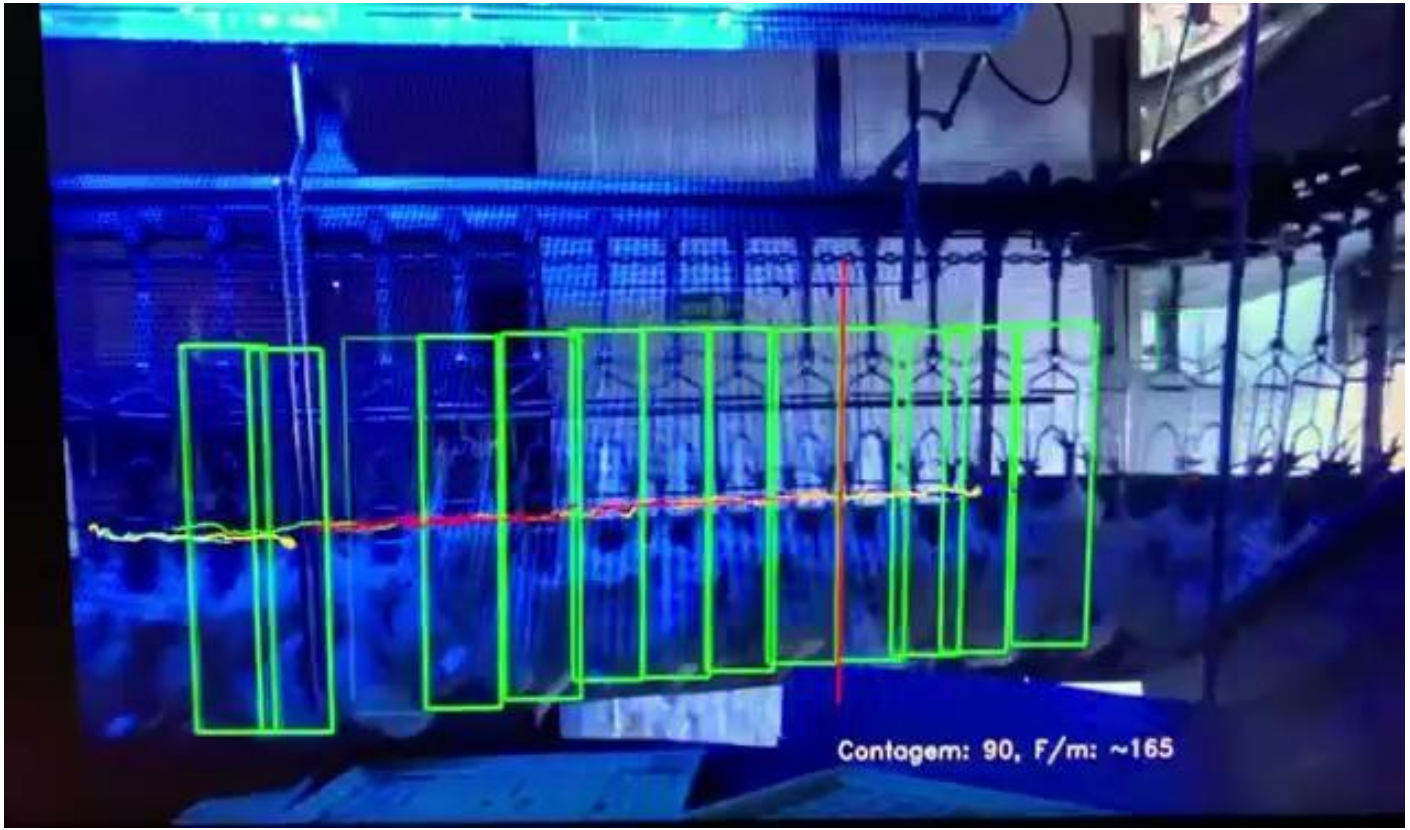




01669
01669

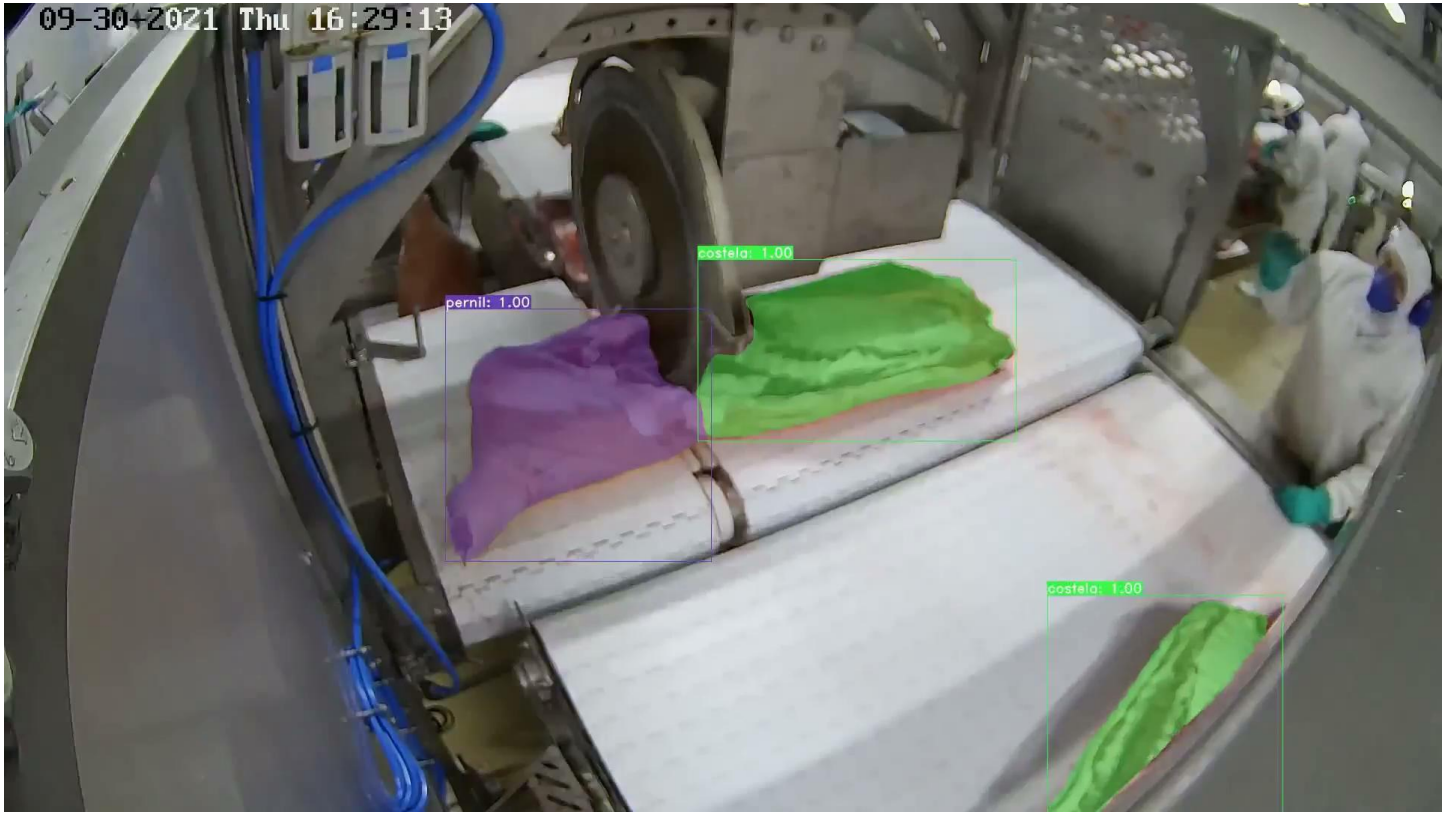
5545
5545

Aplicações

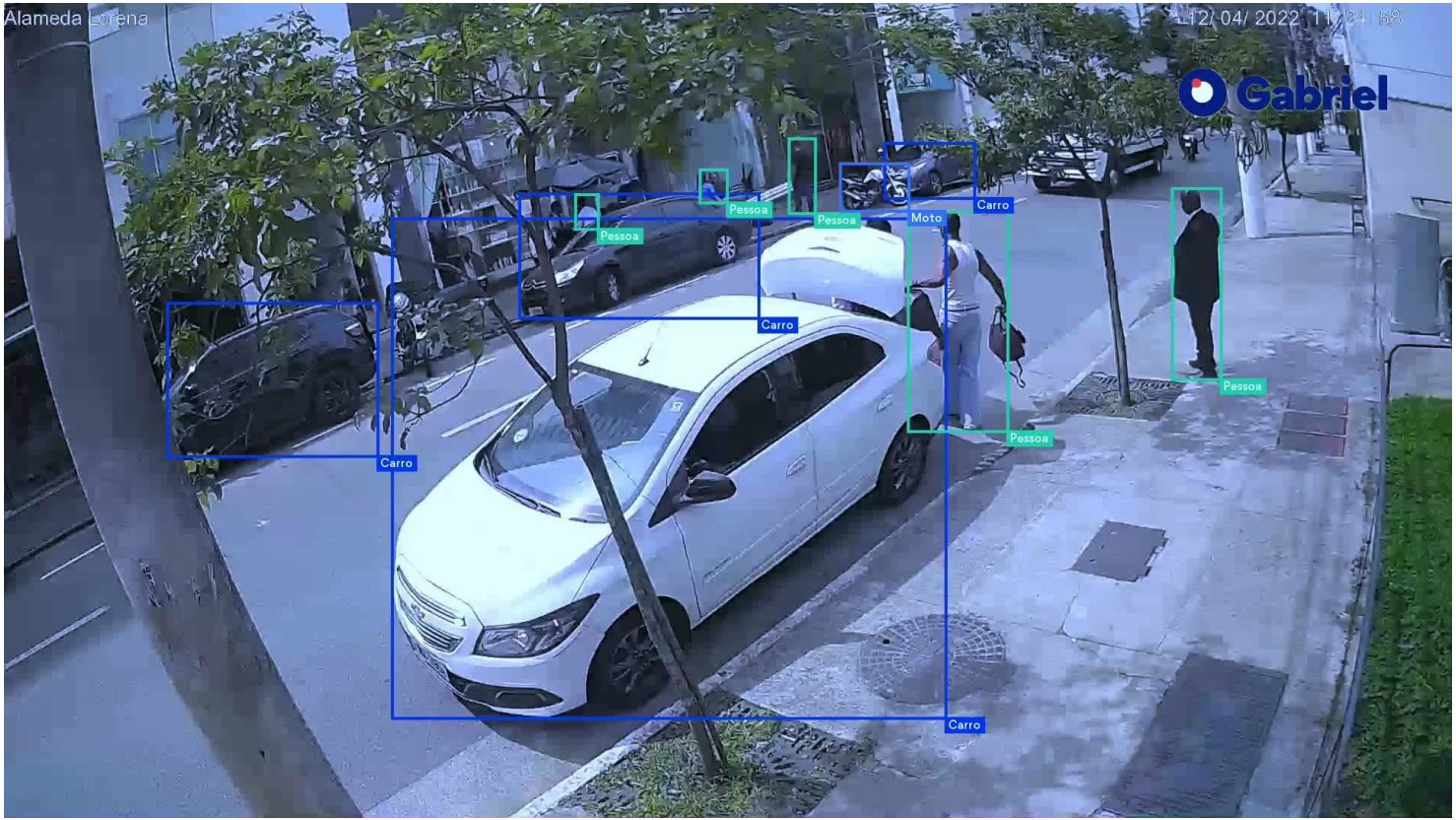


Aplicações

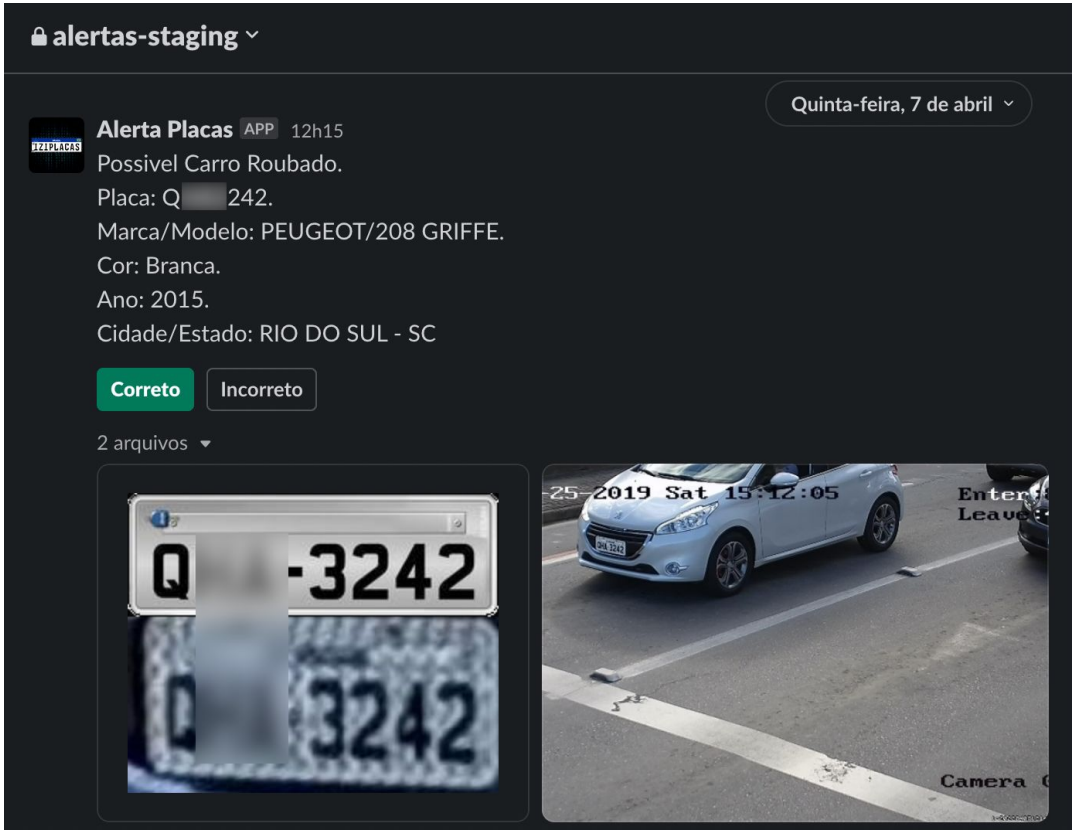
Aplicações



Aplicações


Alameda Lorena
12/ 04/ 2022 11: 24: 58
Gabriel
Pessoa
Pessoa
Pessoa
Carro
Moto
Carro
Pessoa
Carro
Pessoa
Carro

Aplicações


alertas-staging
Quinta-feira, 7 de abril
Alerta Placas APP 12h15
Possivel Carro Roubado.
Placa: Q 242.
Marca/Modelo: PEUGEOT/208 GRIFFE.
Cor: Branca.
Ano: 2015.
Cidade/Estado: RIO DO SUL - SC
Correto
Incorreto
2 arquivos
Q -3242
3242
25-2019 Sat 15:12:05
Camera

Perguntas?

Gabriel Salomon
gabrielsalomon@gmail.com
https://www.linkedin.com/in/gabriel-salomon