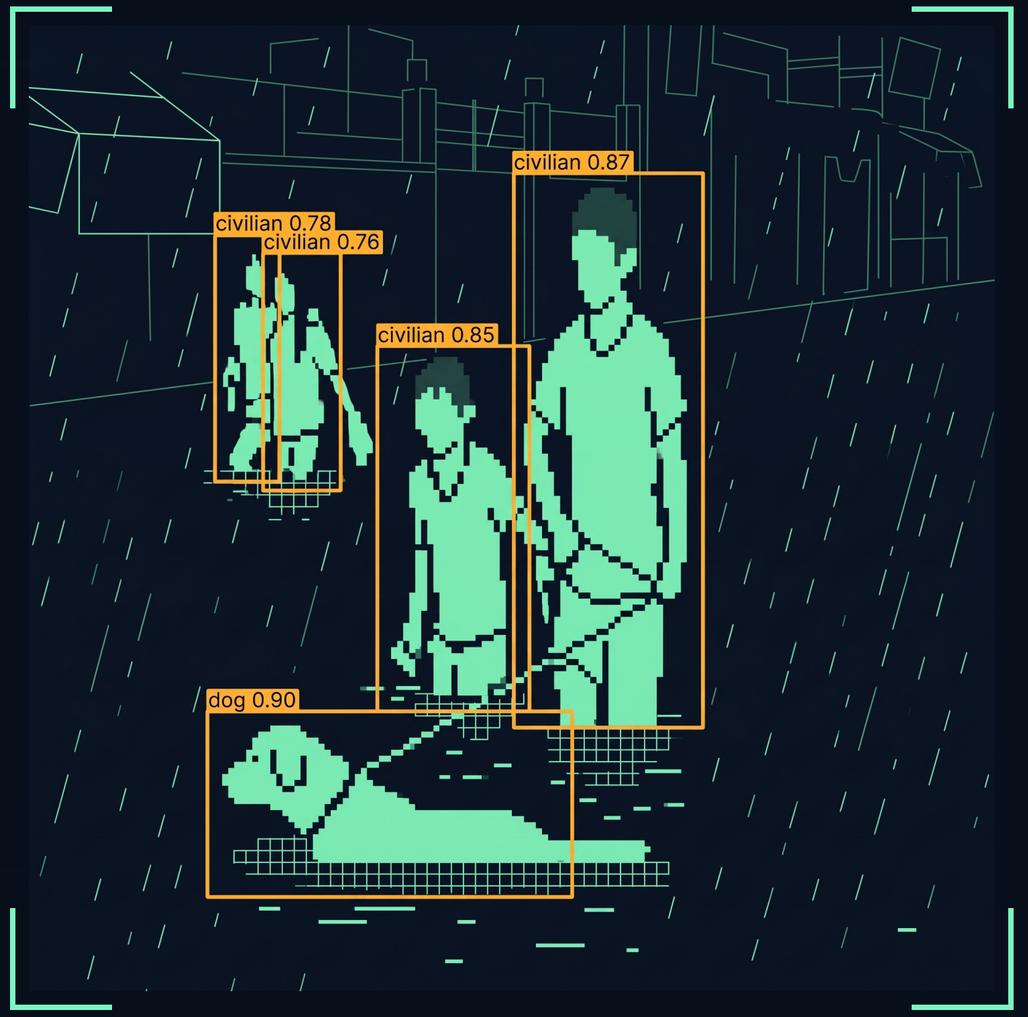
PYTHON BRASIL 2026

IA no Caos

Pipelines multimodais em desastres urbanos

Carolina Soares



QUEM FALA

Carolina Soares

Natal · RN

ALURA

Desenvolvedora de software

UFRN

Graduanda em Ciências e Tecnologia

meu site
carolinasoares.dev



ALÉM DESTE PROJETO

Também atuo com dados abertos, inovação cívica e software livre

Organizei o Open Data Day e atuei como Embaixadora de Inovação Cívica na Open Knowledge Brasil.



COMUNIDADE REUNIDA



COMPARTILHANDO CONHECIMENTO

PERGUNTA DE ABERTURA

Se você tivesse poucos segundos, o que procuraria primeiro?



Enchente no Rio Grande do Sul · Fast Company Brasil

01

O problema

decisões rápidas com informação parcial.

CAOS

Decidir no caos visual

cena

água, oclusão, ângulos

operação

segundos, informação parcial, prioridade

dados

escassos, desbalanceados, sensíveis





DE ONDE VEM O PROJETO

A trajetória do projeto

01

DISCIPLINA

Visão Computacional

ECT · UFRN

02

BOLSA

Smart Metropolis

cidade inteligente

03

VOLUNTARIADO

SPICI

segurança pública

04

HOJE

TCC

Computação Aplicada

QUEM CONSTRÓI

Este projeto foi construído em colaboração



Carolina Soares
autora e desenvolvedora



Helton Maia
orientador

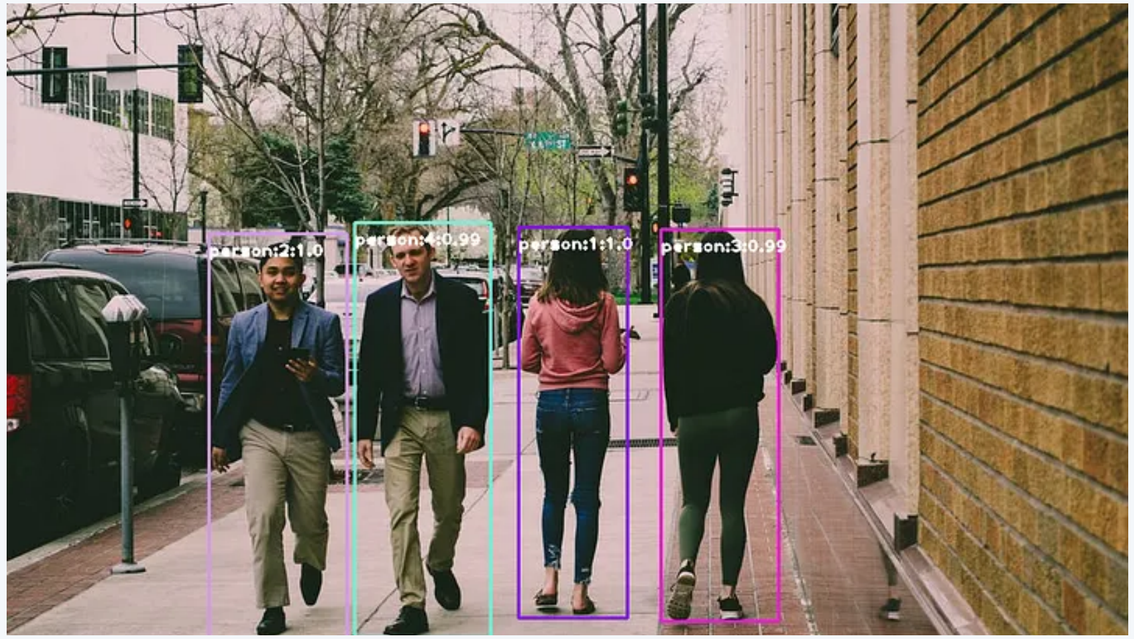


João Galdino
colaborador



EQUIPE E CONTEXTO INSTITUCIONAL, SPICI

Civil e socorrista não são a mesma classe





O QUE O COCO VÊ

person

Uma categoria para qualquer pessoa.

Fonte: Chinmay Bhalerao · Medium

O QUE A OPERAÇÃO PRECISA

civil · socorrista · cão

Cada classe pede uma resposta diferente.

Fonte: Projeto Colabora

TRADUÇÃO DA OPERAÇÃO

Da necessidade operacional à tarefa computacional

O que aparece?	`classe`
Onde está?	`caixa`
É civil ou socorrista?	`classe operacional`
Quão segura é a identificação?	`confiança`



ENTRADA VISUAL · CENA DE RESGATE

VISÃO COMPUTACIONAL

O que é visão computacional

Aqui, imagens de desastres viram respostas sobre o que aparece e onde está.

01 `imagem`
Imagens e vídeos entram como matrizes de valores.

02 `visão computacional`
O sistema aprende o que aparece e onde está.

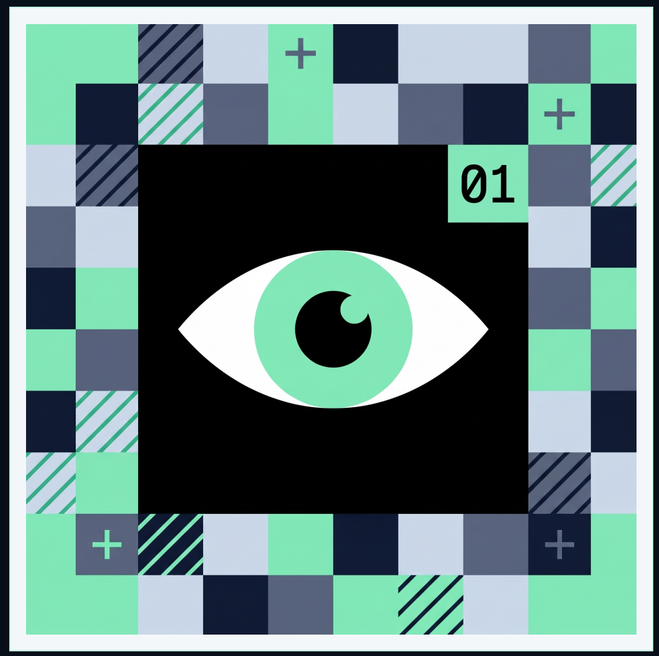


Imagem: Zeljko Loncar, Dribbble

VISÃO COMPUTACIONAL

Para o computador, a imagem é uma matriz de números

amostrar	Divide a cena em uma grade.
quantizar	Mede cada intensidade de 0 a 255.
a rede lê	Convolução, mapas de características e previsão, camada a camada.

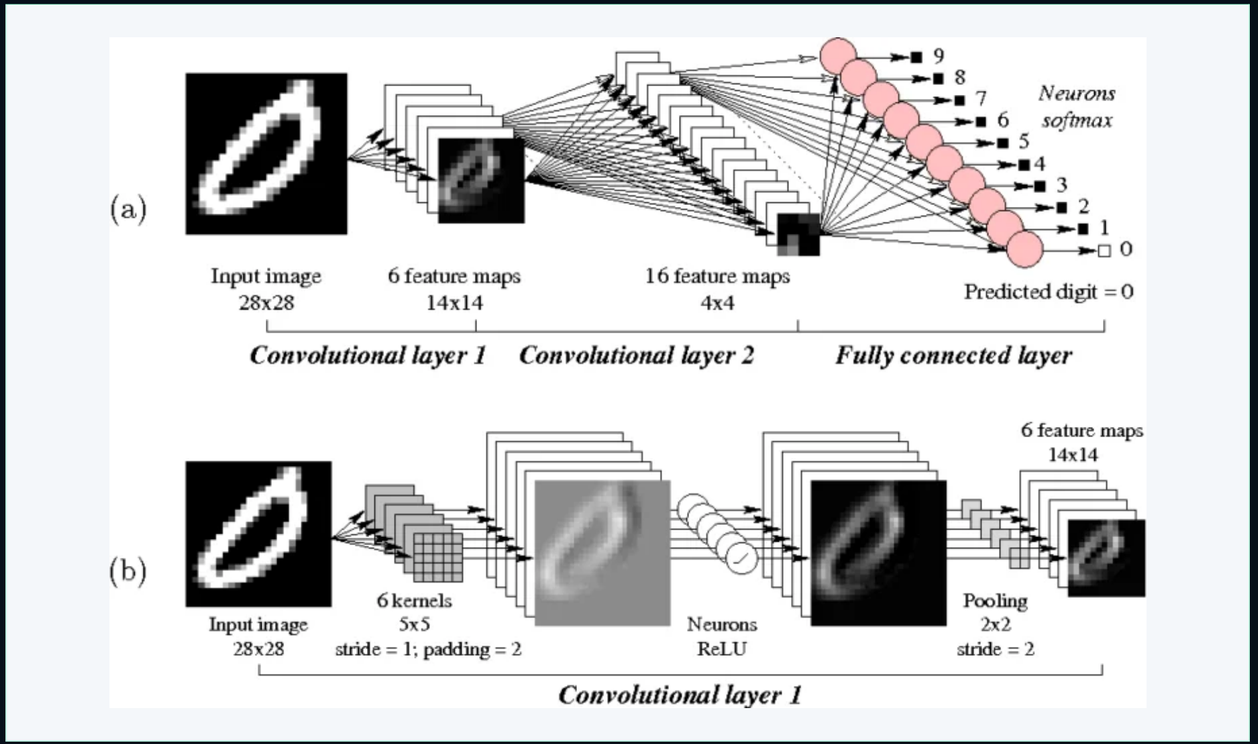


Figura: Iguazio

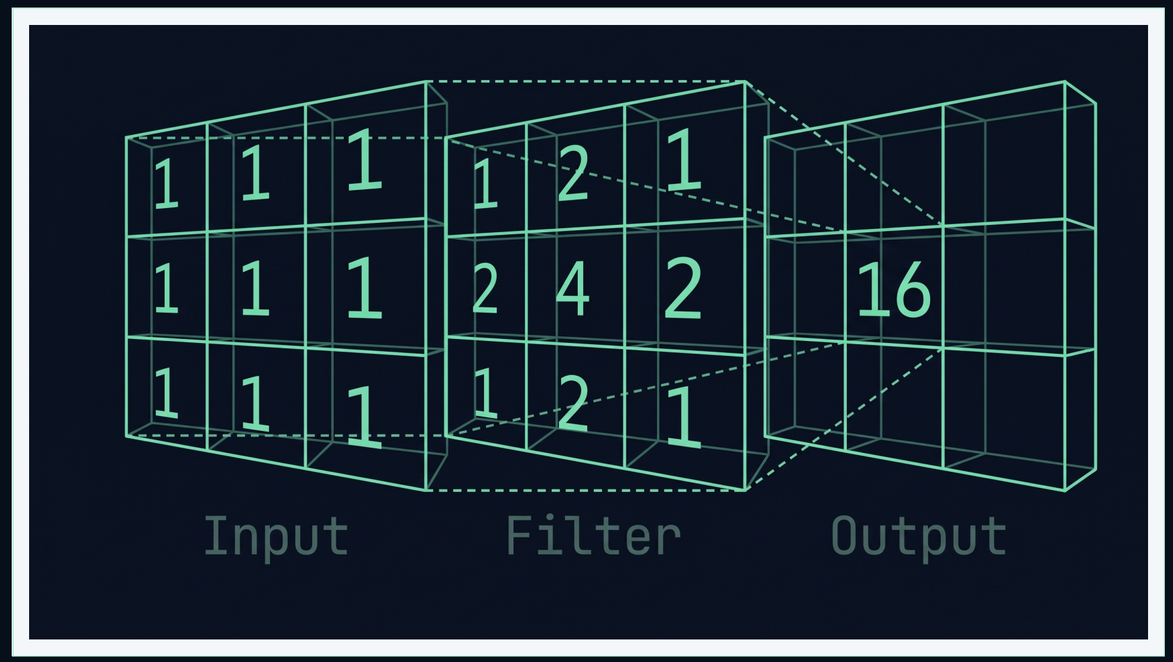
DEMONSTRAÇÃO

O filtro desliza e soma

Em cada posição, entrada vezes filtro vira um pixel da saída. A figura congela esse instante.

DEMONSTRAÇÃO AO VIVO

adamharley.com/nn_vis/cnn/3d.html

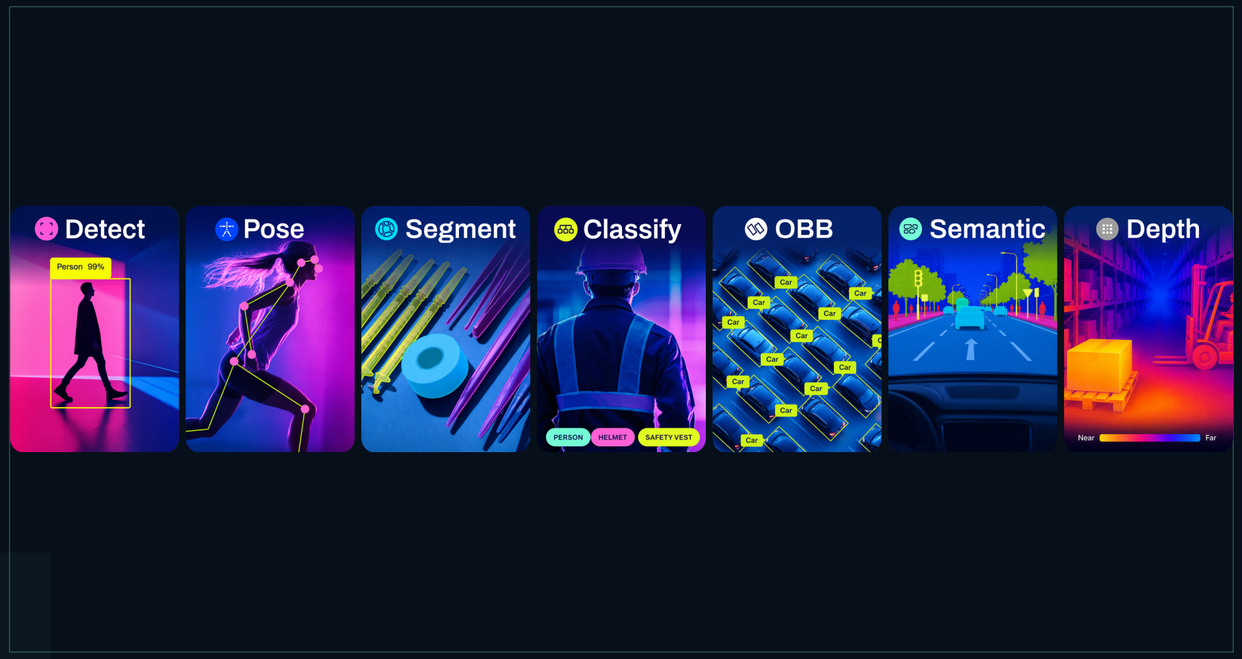


VISÃO COMPUTACIONAL

Detectar é uma técnica entre várias

A faixa mostra o cardápio: classificar, detectar, segmentar, pose e mais.

> A tarefa escolhida define a resposta do modelo.



Famílias de tarefa apresentadas pelo YOLO26.

O DOMÍNIO ALTERA OS PADRÕES

O domínio visual do desastre

água

Corpos parcialmente submersos.

reflexos

Superfícies que duplicam formas.

oclusão

Detritos, veículos e outras pessoas.

baixa luz

Chuva, noite e fumaça reduzem contraste.



Simulado do Corpo de Bombeiros Militar de Santa Catarina · CBMSC

02

Os dados

Coleta, geração sintética, curadoria e anotação, onde o projeto passou a maior parte do tempo.

DADOS DE DOMÍNIO

O que o domínio exige

SEIS CLASSES

Civilians · Rescuers · Cows · Horses · Dogs · Cat

CENÁRIOS

Água, lama e submersão em inundações, deslizamentos e colapsos.

SINAIS DE RESGATE

Capacete, EPI e roupa em destaque, como vermelho ou amarelo.



Coleta em repositórios públicos

Wikimedia Commons	Flickr	Google Images	Roboflow Universe

1.903

imagens no conjunto inicial, todas com licença aberta ou compartilhável.

CUIDADO

São pessoas reais em situação de vulnerabilidade. Licença e uso responsável fazem parte do método.

Algumas das classes mais relevantes tinham os menores números de exemplos.

Sintéticas para complementar o conjunto

- Gerada com `Gemini 2.5 Flash Image`, revisada manualmente.
- Complementa as reais em `dog, cat, cow e horse`.
- Reduz o `desbalanceamento` entre animais.

832
imagens sintéticas





GERAÇÃO MULTIMODAL

O prompt também é método

UM EXEMPLO

"Foto documental em baixa resolução de dois gatos molhados presos em um telhado cercado pela enchente, proporções realistas, rostos visíveis, evento urbano de enchente autêntico"

conteúdo

espécie e ambiente

condição

baixa resolução, céu nublado, luz natural

linguagem

documental, fotojornalística, plano aberto

Variações ampliam o que o modelo observa

GERAÇÃO SINTÉTICA

Cria cenas novas para ampliar as classes escassas.

AUMENTO DE DADOS

Transforma imagens existentes em novas variações.

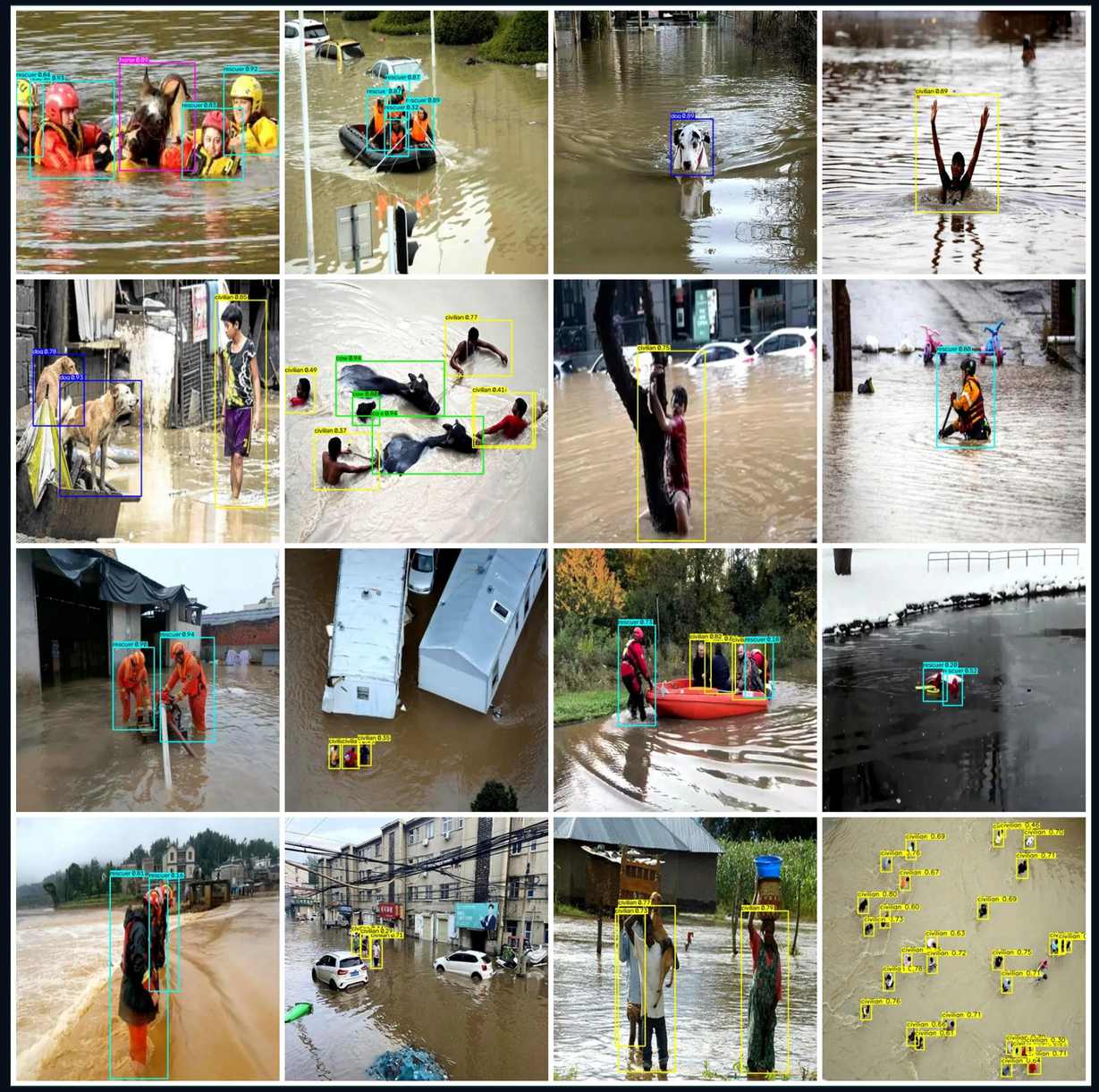


640×640	rotação ±15°	shear ±10°	saturação ±25%	ruído 0,89%

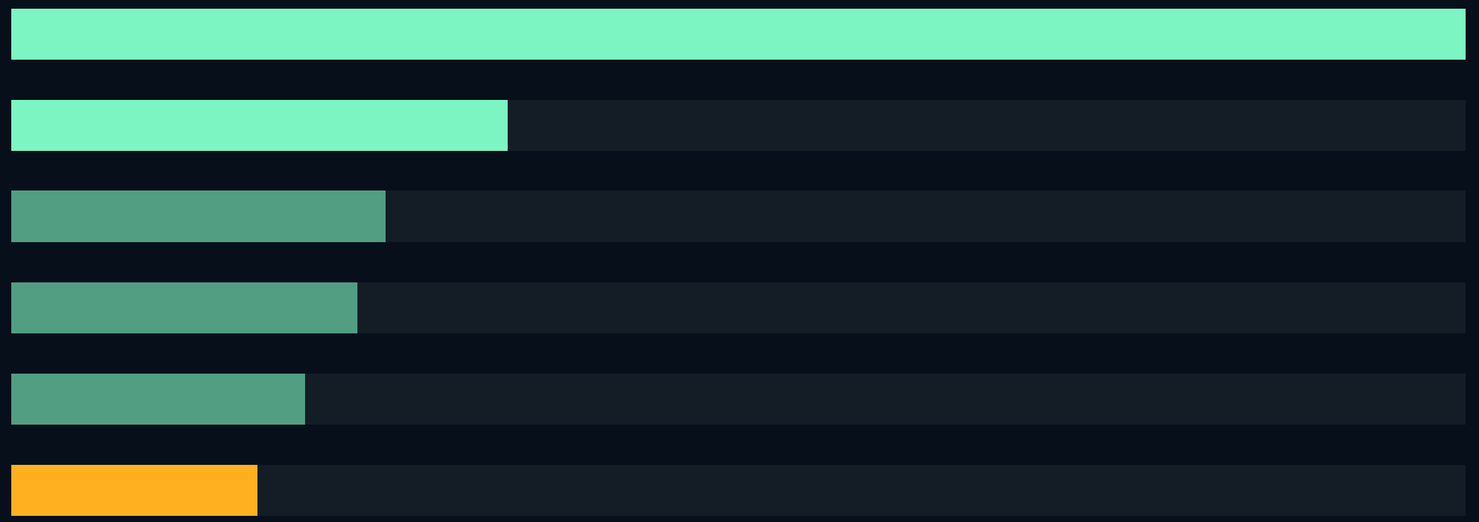
Anotação transforma exemplos em supervisão

Cada caixa associa uma localização a uma das seis classes. Sem isso o treino não tem referência.

> CASO AMBÍGUO
>
> Pessoa ajudando no resgate sem EPI visível: civil ou socorrista? Decisões assim foram revisadas uma a uma.



Seis classes, 6.952 objetos anotados



	OBJETOS	IMAGENS
civilian	3.150	1.060
rescuer	1.074	397
horse	811	279
cow	749	180
cat	637	207
dog	531	373

distribuição multi-label · uma imagem pode conter várias classes e vários objetos

3.240 imagens no conjunto final



treinamento · 82,5%

validação · 11,8%

teste · 5,7%

REGRA DO EXPERIMENTO

Validação usada nas comparações.
Teste mantido separado.

03

O experimento

Seis variantes treinadas, limites de infraestrutura e um protocolo comum de comparação.

LINHA DO TEMPO DO EXPERIMENTO

YOLO11, a base de comparação

PRIMEIRO TESTE

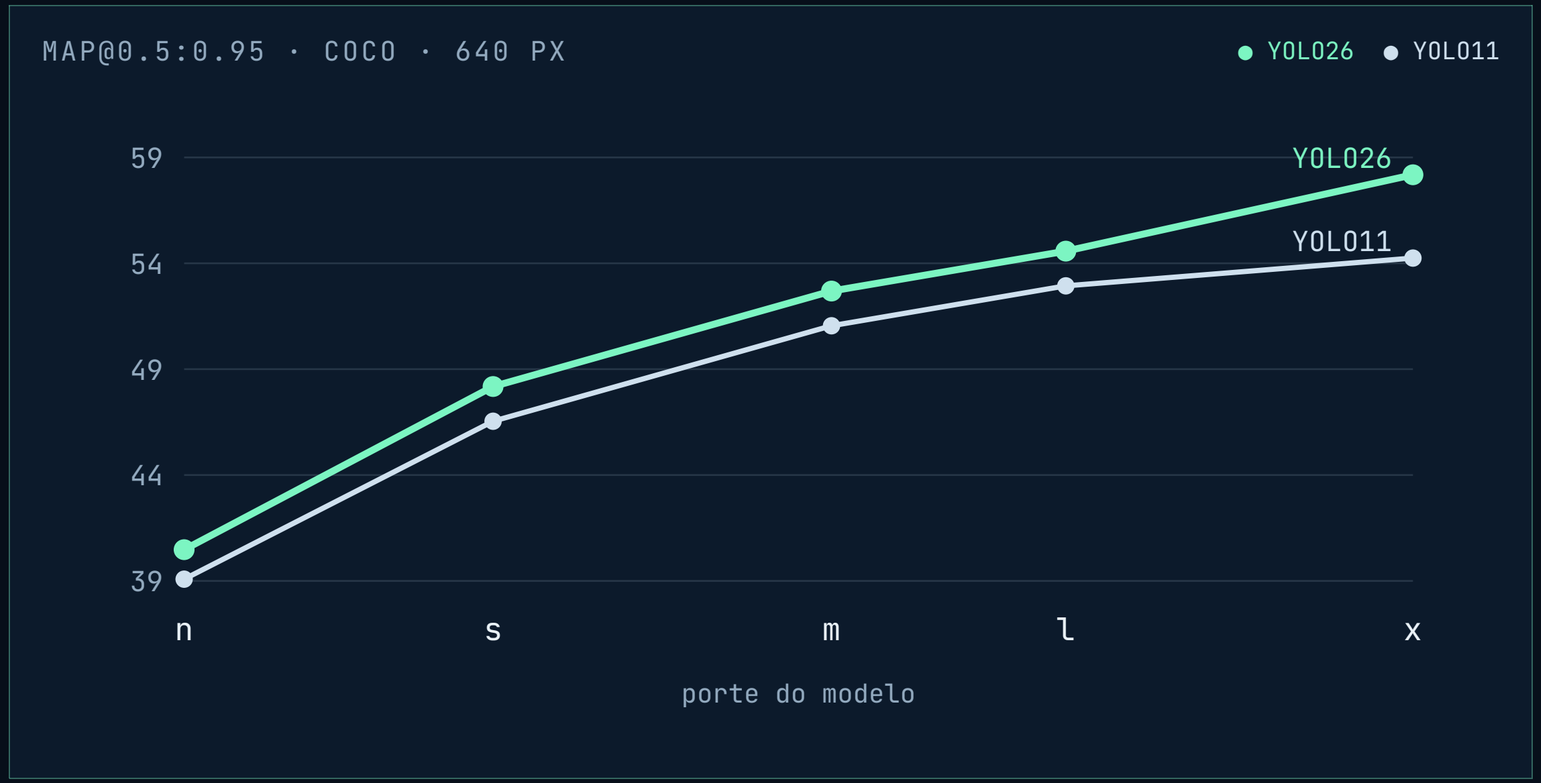
YOLO11

A geração disponível quando o projeto começou.

COMPARAÇÃO POSTERIOR

YOLO26

Lançado depois, entrou como referência adicional.



Referência oficial da Ultralytics, COCO. Não é resultado deste projeto. fonte

DETECTOR PRINCIPAL

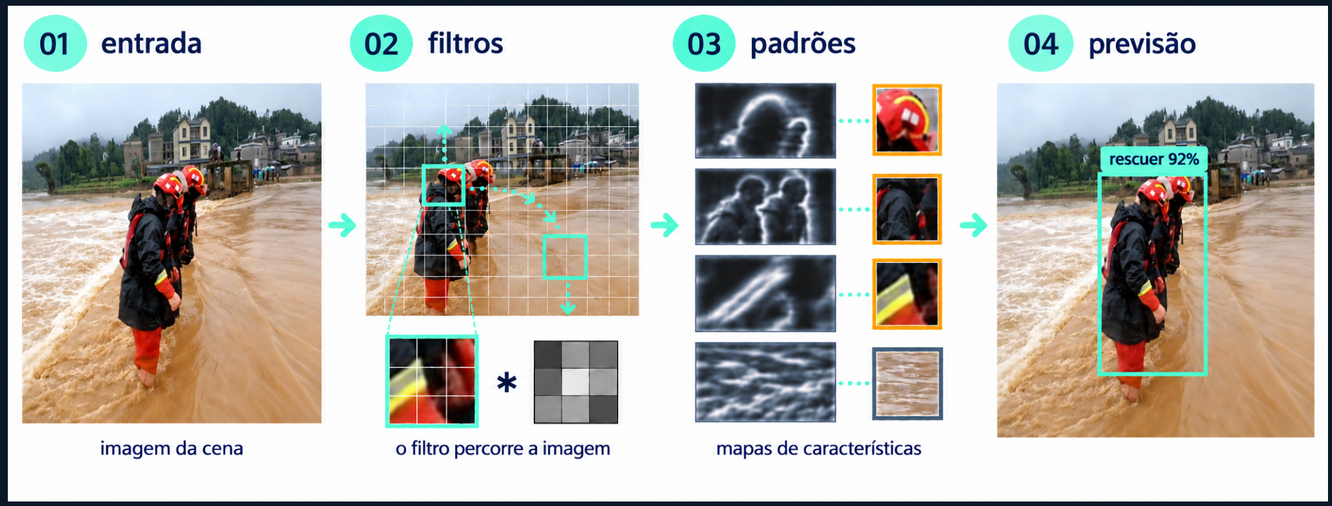
Como o YOLO26 transforma uma imagem em previsões

01 **Olha a imagem**
recebe a cena completa como entrada

02 **Encontra padrões**
reconhece formas, texturas e partes

03 **Combina escalas**
relaciona detalhes pequenos e estruturas maiores

04 **Faz a previsão**
indica posição, classe e confiança



DA IMAGEM AOS PADRÕES E À PREVISÃO

EVOLUÇÃO EXPERIMENTAL

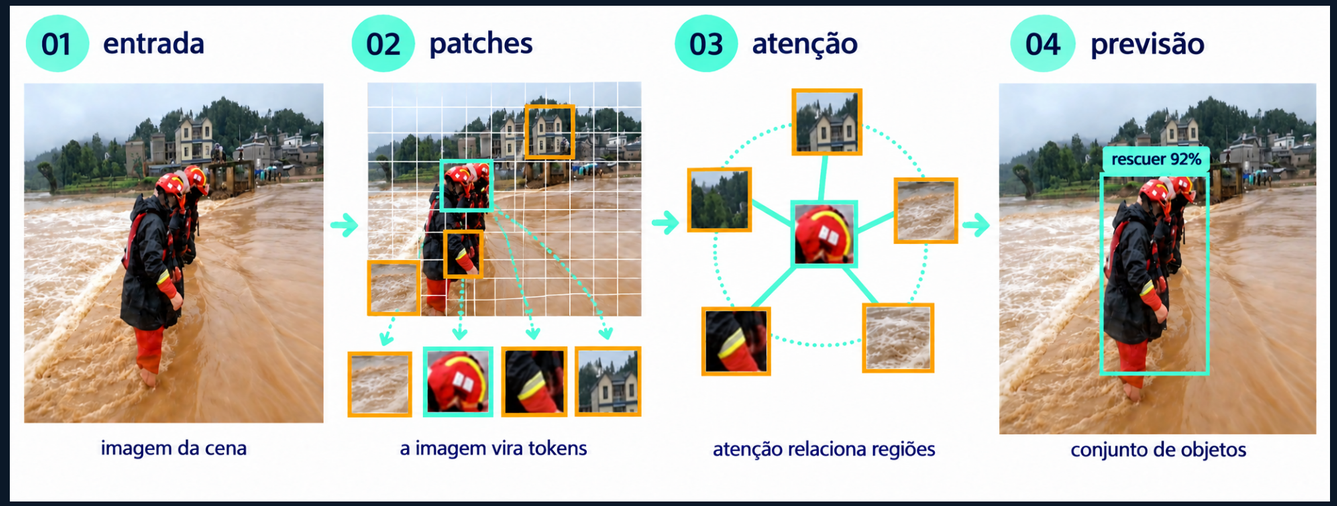
Como o RF-DETR relaciona regiões para detectar objetos

01 **Divide a imagem**
transforma a cena em trechos

02 **Relaciona regiões**
a atenção conecta trechos distantes

03 **Consulta objetos**
queries procuram respostas na cena

04 **Entrega o conjunto**
produz caixas e classes diretamente



DOS PATCHES À ATENÇÃO E À PREVISÃO

O experimento começa com poucas linhas

treino_yolo26.py

```python
from ultralytics import YOLO

model = YOLO("yolo26m.pt")
model.train(
    data="urban-disaster.yaml",
    epochs=50,
    imgsz=640,
    batch=16,
)
```

Pesos pré-treinados, seis classes e o mesmo particionamento.

VARIANTES AVALIADAS

YOLO26	n · s · m · l

50 épocas · 640×640 · batch 16 · Ultralytics

SEGUNDA FAMÍLIA

O RF-DETR foi treinado em duas escalas

RF-DETR NANO

384 px

Variante mais compacta para testar a detecção com menor resolução.

RF-DETR MEDIUM

576 px

Variante maior, com mais detalhe espacial para a mesma tarefa.

configuração comum

batch efetivo 16 · mesmo dataset · mesmo particionamento

CONDIÇÕES DO EXPERIMENTO

O treino também depende da infraestrutura

T4

GPU T4 no Colab para treinar o YOLO26.

100

100 unidades computacionais no Colab para concluir o RF-DETR.

checkpoint

Retomadas do RF-DETR após interrupções da sessão.

Essas condições acompanham a leitura dos resultados.

Um protocolo comum para ler os registros

01

Mesmo conjunto

Seis classes e a mesma partição.

02

Melhor época

Maior mAP@0.5 na validação.

03

Mesma avaliação

Precisão, recall e F1 da mesma linha.

os números valem para estas configurações, com arquiteturas e resoluções diferentes

04

Os resultados

O que as métricas dizem, e o que elas não dizem.

COMO LER UMA DETECÇÃO

Precisão, recall e IoU

Três perguntas sobre cada previsão.

01 · precisão

Quando aponta, quantas vezes acerta?



3 acertos · 1 alarme falso

02 · recall

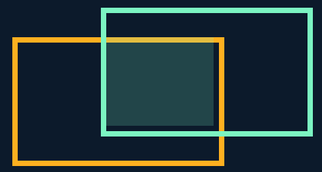
Do que existia, quanto encontrou?



3 encontrados · 1 ignorado

03 · IoU

Quanto as caixas se sobrepõem?

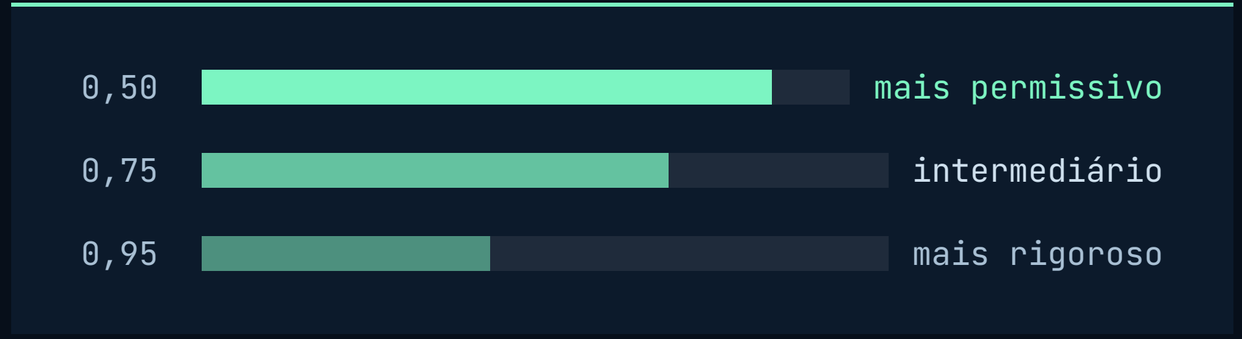


interseção ÷ união

DE UMA PREVISÃO AO RESULTADO

mAP resume muitas avaliações

Quanto melhor a localização exigida, mais difícil a correspondência.



mAP@0.5

Aceita IoU a partir de 0,5.

mAP@0.5:0.95

Faz a média entre vários limiares.

uma métrica única para comparar o desempenho geral

YOLO26 · VALIDAÇÃO · MAP@0.5

O efeito do porte no YOLO26

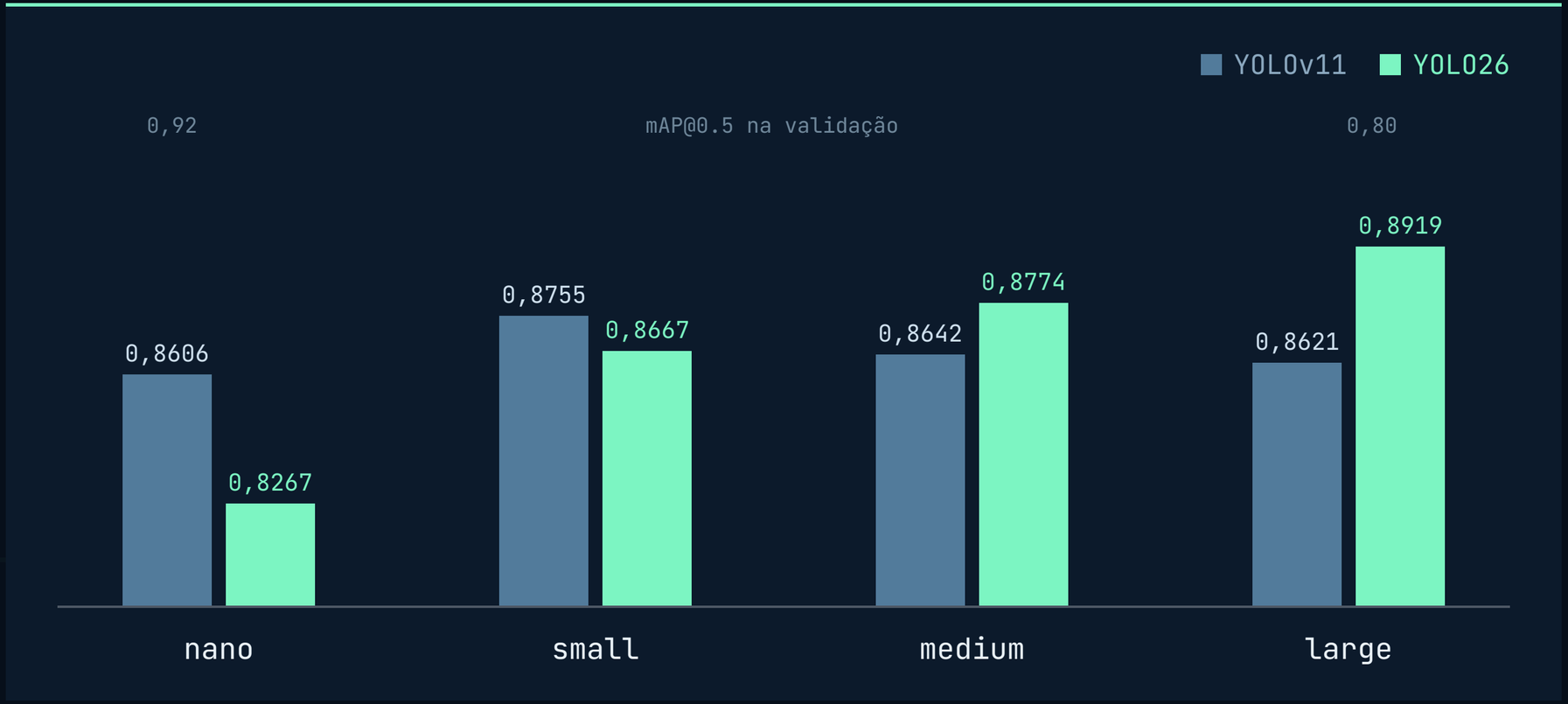
Nesta família, o mAP aumentou com o tamanho do modelo.



COMPARAÇÃO POR PORTE

YOLOv11 e YOLO26

A vantagem muda conforme o tamanho do modelo.



PEQUENO PORTE

nano e small

MAIOR PORTE

medium e large

mAP@0.5

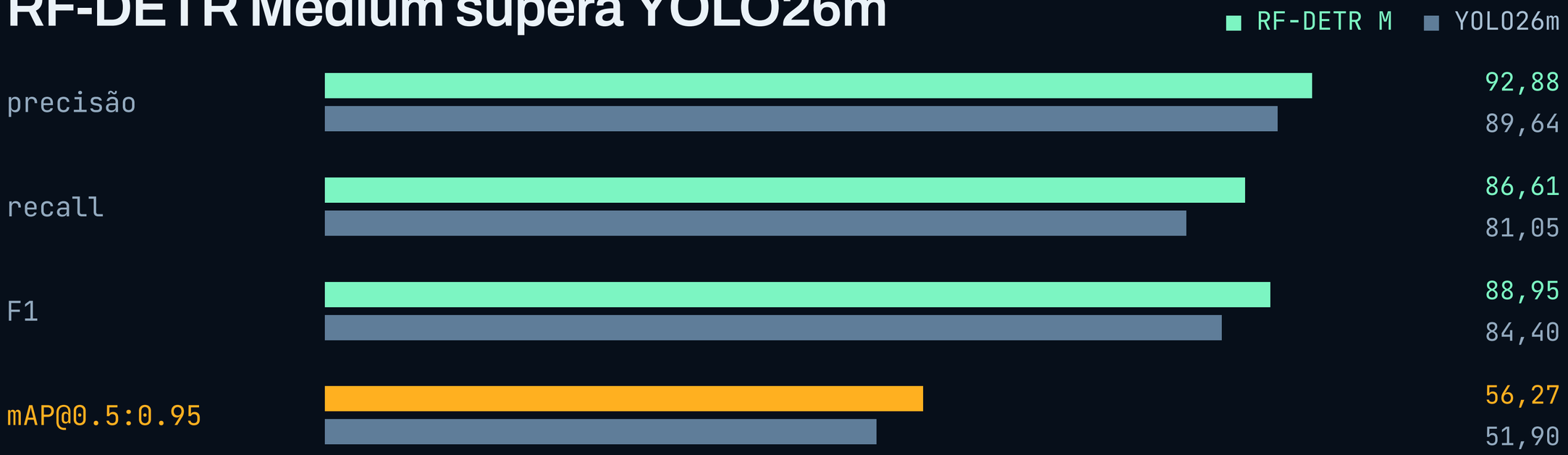
Desempenho na validação, em uma comparação descritiva entre arquiteturas.

RF-DETR: Medium à frente em todas as métricas

VARIANTE	PRECISÃO	RECALL	F1	MAP@0.5
Nano	91,15%	81,95%	84,16%	86,85%
Medium	92,88%	86,61%	88,95%	91,97%

Por classe, `horse` e `rescuer` tiveram melhor desempenho. `civilian`, `cat` e `dog` indicam onde ampliar os dados.

RF-DETR Medium supera YOLO26m



comparação válida para estas configurações, com resoluções e treinos diferentes

TREINO NO DOMÍNIO · COMPARAÇÃO

Treino no domínio melhora recall e mAP

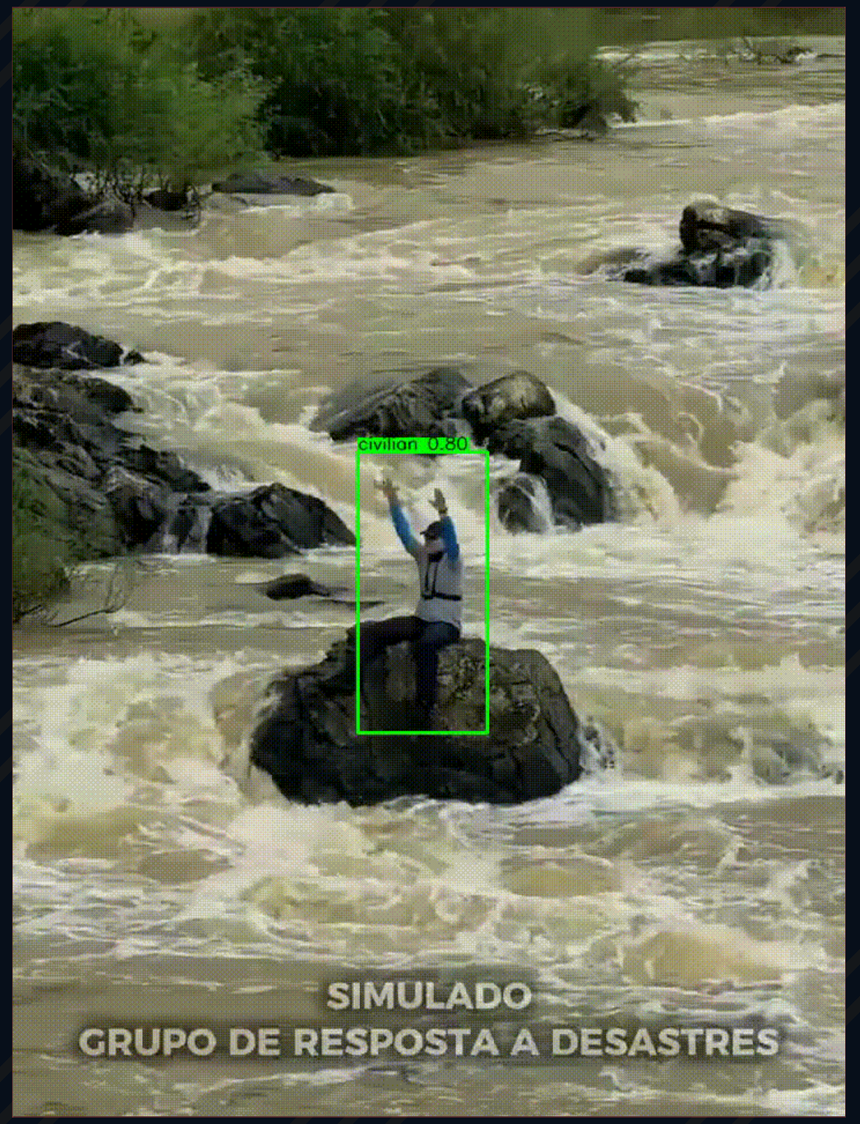
MÉTRICA	YOLO26M TREINADO NO DOMÍNIO	YOLO26M PRÉ-TREINADO EM COCO
precisão	0,861	0,881
recall	0,800	0,701
mAP@0.5	0,865	0,798
mAP@0.5:0.95	0,520	0,402

Comparação no mesmo conjunto de validação, com as classes do Urban Disaster Monitor.

TESTE QUALITATIVO

Que limites aparecem no teste qualitativo?

- Baixa luz e oclusão ainda causam perdas de detecção.
- Não houve benchmark controlado de latência nem de memória.
- Ainda faltam testes para medir o efeito das imagens sintéticas.
- Em vídeo, ainda faltam testes de consistência entre quadros, estabilidade e latência.



Possíveis cenários de uso

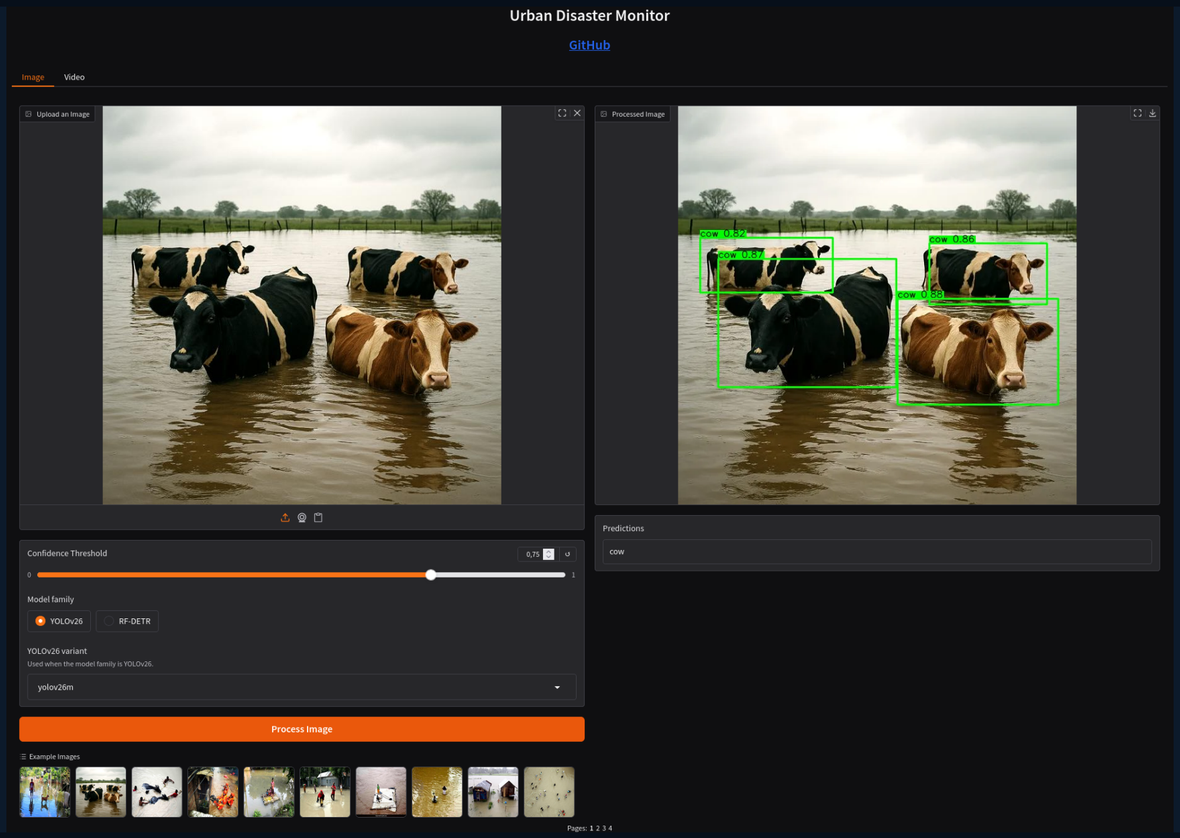
MODELO	CENÁRIO RECOMENDADO
YOLO26n	Drone e edge, com pouca energia.
YOLO26s	Câmera urbana em tempo real.
YOLO26m	Aplicação crítica, com mais capacidade de processamento.
YOLO26l	Ambiente com infraestrutura robusta.
RF-DETR Medium	Prioridade em qualidade de detecção.

Base: documentação Ultralytics e documentação RF-DETR. Os cenários são inferências de uso, não benchmark de hardware.

DEMONSTRAÇÃO

Vamos testar a detecção

imagem · modelo · limiar de confiança



A mesma cena, dois modelos e diferentes limiares de confiança.

Abrir o Urban Disaster Monitor ↗

BALANÇO

Considerações finais

entrega	Dataset do domínio, comparação de seis variantes e interface testável.
alcance	Os números valem para a validação, nestas configurações.
aberto	Generalização, latência e efeito das imagens sintéticas.

O que vem depois desta palestra

`efeito dos dados`	Medir o efeito isolado das 832 imagens sintéticas.
`desempenho`	Medir latência, taxa de quadros e memória.
`vídeo`	Usar contexto temporal para estabilizar detecções.
`dados`	Adicionar cenas noturnas, fumaça, chuva e novas classes.

DADOS ABERTOS

Dados e links para continuar

Kaggle	Conjunto completo para download e reprodução.
Roboflow Universe	Versão anotada para explorar e ampliar.
COCO dataset	Referência de pré-treinamento dos modelos.
Material de apoio	Fundamentos de visão computacional.
licença	CC BY 4.0, com crédito à autora.

CÓDIGO ABERTO

Continue no GitHub

Código, treinamento, avaliação e interface do projeto.

github.com/MariaCarolinass/urban-disaster-monitor

Obrigada. Perguntas?

Carolina Soares

carolinasoares.dev

ACESSE MATERIAIS E CONTATOS