

Avaliação Prática - AV3: Sistema de Detecção de Buracos com API Roboflow

Descrição do Projeto

Este projeto foi reorientado para focar especificamente na **detecção de buracos em estradas**, utilizando uma abordagem baseada em aprendizado profundo e a API da plataforma Roboflow. O objetivo principal é desenvolver um sistema em Python capaz de identificar buracos em imagens, empregando um modelo YOLOv8 especializado hospedado no Roboflow.

Inicialmente, o projeto explorou modelos locais (YOLOv8 e YOLOv5), mas enfrentou desafios na obtenção de pesos pré-treinados específicos para buracos. Com o fornecimento de uma chave de API pelo usuário, a abordagem foi direcionada para o uso do modelo `pothole-detection-yolov8/1` disponível no Roboflow Universe, acessado via `inference-sdk`.

Solução Adotada: API Roboflow

Após tentativas com modelos locais que se mostraram inadequados (eram modelos padrão COCO), a solução definitiva foi integrar o sistema com a API do Roboflow:

1. **Chave de API:** Utilização da chave `d3nDPMwWE1xqV74ma2MN` fornecida pelo usuário.
2. **Modelo Roboflow:** Emprego do modelo `pothole-detection-yolov8/1` hospedado na plataforma Roboflow.
3. **SDK Roboflow:** Uso da biblioteca `inference-sdk` (especificamente `InferenceHTTPClient`) para realizar as chamadas de inferência.
4. **Correções no SDK:** Ajustes no código para compatibilidade com a versão instalada do SDK, incluindo a importação correta de classes (`InferenceHTTPClient`) e a implementação manual da visualização das caixas delimitadoras com OpenCV, já que `render_boxes` não estava disponível na versão do SDK utilizada.

Técnicas de Processamento Digital de Imagens Aplicadas

O desenvolvimento manteve o uso de Python com OpenCV, NumPy e Matplotlib. As técnicas aplicadas incluem:

1. **Conversão para Escala de Cinza:** Pré-processamento opcional.
2. **Filtragem Gaussiana:** Suavização opcional para redução de ruído.
3. **Detecção de Bordas Canny:** Identificação opcional de contornos.
4. **Detecção de Objetos com API Roboflow:** Utilização do modelo `pothole-detection-yolov8/1` via API para identificar buracos na imagem original.
5. **Visualização Manual:** Desenho das caixas delimitadoras e rótulos sobre a imagem original utilizando OpenCV.

Resultados Obtidos (com Imagem `buracos.jpg` via API Roboflow)

A seguir são apresentados os plots gerados com Matplotlib, exibindo os resultados das etapas de processamento aplicadas à imagem `buracos.jpg` fornecida, utilizando a API Roboflow.

Figura 1: Imagem Original (`buracos.jpg`)

Plot da Imagem Original Descrição: Plot da imagem de entrada colorida, mostrando uma rua com buracos visíveis e um caminho.

Figura 2: Imagem em Escala de Cinza

Plot da Imagem em Escala de Cinza Descrição: Plot do resultado da conversão da imagem original para escala de cinza.

Figura 3: Imagem com Filtro Gaussiano Aplicado

Plot da Imagem com Filtro Gaussiano Descrição: Plot da imagem original após a aplicação de um filtro Gaussiano.

Figura 4: Detecção de Bordas com Canny

Plot da Detecção de Bordas Canny Descrição: Plot do resultado da aplicação do detector de bordas Canny.

Figura 5: Detecção de Buracos com API Roboflow na Imagem `buracos.jpg`

Plot da Detecção Roboflow Descrição: Plot do resultado da aplicação do modelo `pothole-detection-yolov8/1` via API Roboflow na imagem `buracos.jpg`. O modelo detectou 4 buracos (classe "Potholes") com diferentes níveis de confiança (0.82, 0.80, 0.60, 0.42), marcados com caixas verdes.

Análise e Comentários sobre os Resultados

As técnicas clássicas de PDI foram aplicadas opcionalmente (Figuras 2, 3, 4).

A aplicação do modelo especializado em buracos via API Roboflow (Figura 5) na imagem `buracos.jpg` foi **bem-sucedida**. O modelo identificou corretamente múltiplos buracos presentes na via, demonstrando a eficácia de usar um modelo treinado especificamente para a tarefa e acessível via API.

Os resultados foram salvos em um arquivo CSV (`roboflow_pothole_detections.csv`) contendo as coordenadas das caixas delimitadoras, a classe detectada ("Potholes") e a confiança de cada detecção.

Conclusão

Este trabalho implementou com sucesso um sistema de detecção de buracos utilizando a API do Roboflow e um modelo YOLOv8 especializado. Após superar desafios com modelos locais e SDKs, a integração com a API, utilizando a chave fornecida, permitiu a detecção precisa dos buracos na imagem de teste.

O projeto demonstra a aplicação de técnicas de PDI e a integração com APIs de modelos de visão computacional, destacando a vantagem de usar modelos especializados hospedados em plataformas como o Roboflow para tarefas específicas. A documentação reflete a jornada técnica, as correções implementadas e os resultados positivos obtidos com a abordagem final.