

Detecção Automática de Postes Elétricos e seus Componentes em Imagens de Nível de Rua via Métodos de Detecção de Objetos

Iuri Wladimir Molina Junior¹, Jonathan de Andrade Silva²

¹Universidade Federal do Mato Grosso do Sul 79070-900 – Campo Grande – MS – Brazil
Universidade Federal do Mato Grosso do Sul (UFMS) 79070-900 – Campo Grande – MS – Brazil

iuri.molina@ufms.br, jonathan.andrade@ufms.br

Abstract. *This article presents an approach to the inspection of the electrical network using the city of Campo Grande as a scenario. Advanced deep learning and computer vision techniques were used with the aid of the MMDetection library. The article performs a comparative analysis of several models in different versions of the dataset, evaluating the impact of data augmentation techniques on the robustness of the model. The results highlight the importance of these practices to generalize the model, evidencing the efficacy of each technique implemented.*

Resumo. *Esse artigo apresenta uma abordagem para inspeção da rede elétrica utilizando a cidade de Campo Grande como cenário. Foram utilizadas técnicas avançadas de aprendizado profundo e visão computacional com o auxílio da biblioteca MMDetection. O artigo realiza uma análise comparativa de vários modelos em diversas versões do dataset, avaliando o impacto das técnicas de aumento de dados na robustez do modelo. Os resultados destacam a importância dessas práticas para generalizar o modelo, evidenciando a eficácia de cada técnica implementada.*

1. Introdução

A infraestrutura energética urbana desempenha um papel central no funcionamento e desenvolvimento das cidades de todo o país, e os postes desempenham um papel fundamental nessa estrutura. Danos causados por eventos climáticos como tempestades ou por atividades humanas podem levar ao mal ou não funcionamento da energia para casas, empresas e outros estabelecimentos. Com isso, distribuidoras de energia vem buscando formas de identificar e monitorar os postes, a fim de mitigar o maior número de problemas possíveis para a rede de distribuição elétrica.

A abordagem utilizada pelas empresas é o monitoramento manual de um técnico da distribuidora indo realizar inspeções periódicas, ou da detecção reativa, dependendo da população relatar um problema no poste, dando a localização do poste com possível defeito. Apesar de ser uma abordagem simples, não é eficaz na prevenção de acidentes, e nem muito rápido. Uma abordagem eficiente seria a automação do processo de monitoramento utilizando detecção de objetos com o uso de inteligência artificial.

A cidade de Campo Grande, Mato Grosso do Sul, Brasil, é a que mais sofre apagões, devido a problemas nos postes, defeitos na rede elétrica, entre outros fatores. Nesse caso, a área de Visão Computacional surge como uma possível solução para a detecção de postes em centros urbanos. Trabalhos como [Fambrini et al. 2020] e [Ramlal et al. 2020], tem explorado técnicas, com uso de aprendizado profundo e aprendizado não-supervisionado, bastante comuns na área de visão computacional.

Tendo em vista esses fatos, o objetivo deste artigo é explorar modelos de visão computacional e avaliar a eficiência deles na tarefa de Detecção de Postes e seus componentes na cidade de Campo Grande, utilizando a biblioteca do Open MM-Lab chamada MMDetection.

2. Revisão de Literatura

Nesta seção, iremos nos aprofundar nas pesquisas já existentes sobre detecção de postes e seus componentes, veremos que aplicações os autores usaram e as lacunas que os trabalhos buscam preencher.

2.1. Artigo "Deep learning based system to electric distribution network inspection"

O artigo usa aprendizado profundo para reconhecer pontos quentes na rede elétrica. Para chegar nesse resultado, um veículo foi equipado com oito câmeras em todas as direções, permitindo inspecionar centenas de quilômetros sem necessidade de parar o veículo ou do uso de um operador humano analisando poste por poste. O estudo mostra que o sistema de detecção baseado em redes neurais convolucionais com uma precisão de 95% foi capaz de identificar postes e pontos quentes na rede elétrica com precisão, uma melhora em relação ao método tradicional que geralmente varia entre 70% e 80%. Esse sistema oferece uma vantagem por ser mais rápido, mais preciso além de ter mais segurança para suas operações.

2.2. Artigo "Toward Automated Utility Pole Condition Monitoring: A Deep Learning Approach"

Aborda o desenvolvimento de um sistema que possa identificar automaticamente através de detecção, danos (como rachaduras) em postes. Diversas imagens foram capturadas de postes com e sem danos. E com o uso de redes neurais treinaram um modelo capaz de detectar danos com até 95% de precisão. Devemos destacar nesse artigo que o conjunto de dados (Dataset) utilizado para treinar o modelo foi relativamente pequeno, podendo ter sua precisão diminuída em dados de outros ambientes. Ainda assim, o estudo representa um avanço na parte de detecção de defeitos em postes de utilidade pública.

2.3. A Framework of Power Pylon Detection for UAV-based Power Line Inspection

Os integrantes do artigo apresentam um novo framework para detecção através de veículos aéreos não tripulados (VANTs) de linhas de alta tensão. O framework combina fatores como calibração da câmera, projeção da torre de energia em um modelo 3d e a extração de características de torres com defeito, além da localização das mesmas. Os experimentos mostraram que o framework é capaz de detectar torres de energia defeitos com até 96% de precisão.

2.4. Mask R-CNN Powerline Detector: A Deep Learning approach with applications to a UAV

Os autores utilizam um método baseado em deep learning para detecção de linhas de transmissão por meio de imagens aéreas, para reduzir o tempo e o custo das inspeções que são feitas de maneira manual, além de melhorar a vida dos trabalhadores. O modelo convolucional Mask R-CNN detecta as linhas e gera máscaras para delimitar a região dos objetos detectados, o método teve a excelente precisão de 98,2% de precisão.

2.5. End-to-End Object Detection with Transformers

Artigo publicado por pesquisadores de Stanford e do Google AI, apresenta um novo método de detecção end-to-end¹ baseado no uso de transformadores, o transformador gera uma representação global da imagem usando o processo de self-attention, essa representação é usada para gerar uma detecção de regiões, essas regiões são então classificadas usando um classificador de regiões.

3. Metodologia

Nesta seção será apresentada o método adotado para detecção de postes, iremos destacar diferentes características dos modelos abordados, tais como: one-stage, two-stages e transformes. Apesar de distintas, todas elas são eficientes no problema de detecção de objetos.

3.1. Método One-Stage

Os modelos de detecção YOLO (You Only Look Once) e RetinaNet são os maiores exemplos de modelos one-stage, esses e outros métodos one-stage realizam a detecção de objetos e a regressão da caixa delimitadora em uma única passagem pela arquitetura da rede sem o uso de propostas de região pré-geradas, fazendo com que o tempo de treino e inferência sejam mais rápidos.

3.2. Método Two-Stage

Contrastando com a abordagem anterior, a detecção de dois estágios divide o processo de detecção em duas etapas: propostas de geração de região e a classificação de objetos para cada proposta de região. Apesar de ser mais lento que a etapa anterior, os detectores de dois estágios atingem uma precisão melhor. O modelo Faster R-CNN é um modelo exemplo de detector de dois estágios.

3.3. Métodos baseados em Transformers

Inicialmente desenvolvidos para tarefas de Natural Language Procession (NLP), os detectores baseados em transformers têm se mostrado bastante eficientes nas tarefas de classificação e detecção de objetos. Esse método elimina a necessidade de caixas delimitadoras (bounding boxes), além de serem capazes de aprender representações globais de cada imagem ao invés de representações locais feitas pelos métodos anteriores. O DETR é o modelo mais famoso de detecção de objetos baseado em transformers, sendo especialmente útil quando a precisão é prioridade em relação ao tempo de treino e inferência.

¹Realiza todas as etapas do processo de detecção de objetos em uma única etapa.

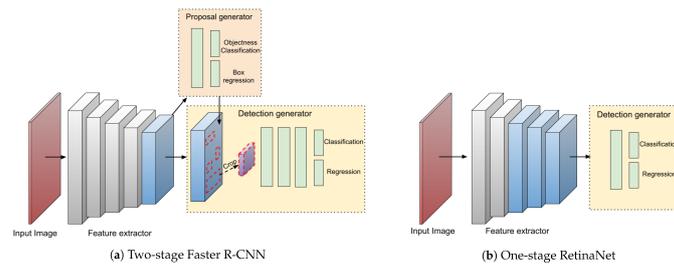


Figure 1. One-Stage e Two-Stages.
 Fonte: [Carranza-García et al. 2021].

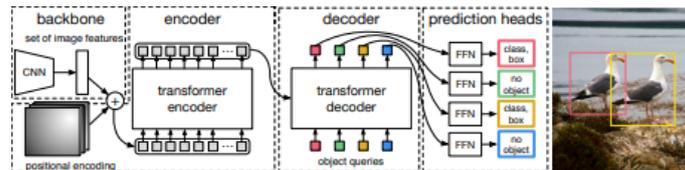


Figure 2. Arquitetura Transformers.
 Fonte: [Carion et al. 2020].

4. Experimentos

Nesta seção, será detalhado como foram feitos os experimentos, desde a aquisição dos dados, configuração da câmera, processo de anotação de imagens, modelos que foram treinados para detecção.

4.1. Dataset

O dataset é composto por um conjunto de dez vídeos com média de 2 minutos de duração por vídeo, todos eles se passam no período matutino/vespertino em ambiente urbano da cidade de Campo Grande, Mato Grosso do Sul. Ao todo, cerca de 8 mil frames foram recortados dos vídeos utilizando a ferramenta Ffmpeg, todas as imagens com o tamanho de 1920x1080, por questão de armazenamento e processamento nem todos os frames foram utilizados no trabalho.

O processo de anotação foi feito utilizando a ferramenta Roboflow, quatro classes foram anotadas no dataset em formato de Bounding Box, são elas: Postes, Cruzeta, Transformador e Liso (referente a não presença da cruzeta no Poste). Cinco versões de dataset foram geradas, onde todas possuem a divisão de 80% das imagens para o conjunto de treinamento, 10% para o conjunto de validação e 10% para o conjunto de teste, totalizando 2209 imagens ao todo. Em duas versões do dataset, foi utilizada a técnica de aumento de dados (Data Augmentation), os processos utilizados foram: Flip na Horizontal em um terço do dataset, imagens em escala de cinza em 25% do dataset, processo de mosaico em 15% do dataset, onde até quatro imagens ficam em um mosaico, e por último o processo de Cut-Out, onde adiciona um ruído em forma de bounding box preta em alguns pontos da imagem em 15%, totalizando 3387 imagens nos datasets com augmentation. do dataset. Todos esses processos de Augmentation podem ajudar o modelo a generalizar melhor para dados de teste deixando o modelo mais robusto.

4.2. Configuração da Câmera

A câmera de modelo GoPro é montada na parte superior de um carro com visão lateral do veículo, uma lente monocular Olho-de-Peixe (Fisheye) com angulo de 180 graus é usada na captura dos vídeos.



Figure 3. Exemplo de anotação do Poste, Cruzeta e Transformador.

4.3. Treinamento

O MMDetection é um framework de código aberto desenvolvido pela Open MM-Lab projetado especificamente para tarefa de detecção de objetos, desde a lida com modelos pré-treinados até o processo de treinamento e inferência. Quatro modelos foram escolhidos da benchmark do mmdetection e treinados em todas as versões do dataset, os modelos estão divididos entre One-Stage, Two-Stage e Transformers. Cada um com suas características e abordagens. Abaixo estão alguns detalhes sobre os modelos:

4.3.1. Faster R-CNN

É um modelo de dois estágios onde utiliza uma rede neural convolucional (CNN) para gerar propostas de região, áreas na imagem onde podem conter objetos. Após isso, ele usa outra rede para classificar cada proposta de região e prever sua localização. Para detecção de postes, vamos usar uma arquitetura com o backbone ResNet-101, otimizador SGD, batch de tamanho 16 e o learning rate inicial de 0,01.

4.3.2. RetinaNet

Ao contrário do modelo anterior, a arquitetura do RetinaNet utiliza somente um estágio, que usa um mecanismo chamado focal loss para lidar com o desbalanceamento de classes. Ele funciona da seguinte forma: A rede neural gera uma pirâmide de características, em seguida, classifica os objetos em cada nível da pirâmide. Para detecção de postes, vamos

usar uma arquitetura com o backbone ResNet-101, otimizador SGD, batch de tamanho 16 e o learning rate inicial de 0,01.

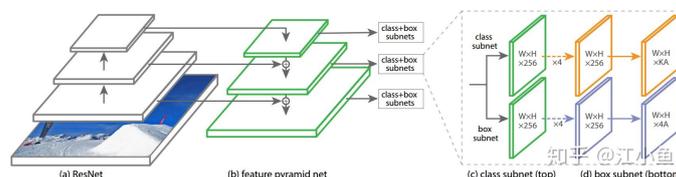


Figure 4. Retinanet.

Fonte: MMDetection Repository

(<https://github.com/open-mmlab/mmdetection>).

4.4. Deformable DETR

Modelo proposto pela equipe do Google [Carion et al. 2020], é baseado em transformador de 12 camadas que usa uma deformação baseada em cross-attention permitindo que o modelo se ajuste ao tamanho de diferentes objetos. Para detecção de postes, vamos usar uma arquitetura com o backbone Swin-L ², otimizador SGD, batch de tamanho 16 e o learning rate inicial de 0,01.

4.5. Dino

É um modelo de detecção end-to-end baseado no DETR [Zhang et al. 2022]. O DINO usa uma seleção de consulta mista para iniciar a ancoragem, ajudando na detecção de objetos de diferentes tamanhos, o DINO também usa o método look-forward twice fazendo duas previsões de caixa, uma para o primeiro nível da rede e outra para o segundo. As previsões são combinadas na previsão final. Para detecção de postes, vamos usar uma arquitetura com o backbone Swin-L, otimizador SGD, batch de tamanho 16 e o learning rate inicial de 0,01.

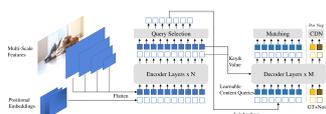


Figure 5. Dino DETR.

Fonte: MMDetection Repository

(<https://github.com/open-mmlab/mmdetection>).

5. Experimentos

Todos os treinamentos foram realizados no ambiente do Google Colab, esse ambiente foi escolhido devido a sua hospedagem de GPUs de forma gratuita. Cada modelo realizou cinco treinamentos em diferentes versões do dataset, porém todos com as mesma imagens, mesmo número de épocas e mesmo learning rate. Após esse treinamento, os modelos terão que fazer inferência em um conjunto de teste com imagens transformadas de

²Ainda não foi lançada ResNet-101 para esse modelo até a presente data.

fish-eye para equirretangular [Ohashi et al. 2016], que não foram introduzidas no conjunto de validação e treinamento, isso serve para avaliar a capacidade dos modelos treinados em qualquer configuração de câmera. A planificação foi feita utilizando o repositório Defisheye <https://github.com/duducosmos/defisheye>.

Nosso objetivo é verificar se os modelos de detecção aqui apresentados são uma abordagem eficaz na detecção de postes e seus componentes, para auxiliar as distribuidoras de energia a detectar e conter possíveis erros.

Para isso iremos testar diferentes modelos em diferentes tipos de versões de datasets, iremos avaliar como os modelos desempenham em imagens com resolução original 1920x1080, em imagens de tamanho reduzido 640x640 e em imagens planificadas. Após isso avaliaremos se o data augmentation realmente melhora a performance de aprendizado dos modelos. Segue abaixo uma tabela para ilustrar como foram feitos os experimentos:

Dataset	AUG	EPOCHS	LR
1920x1080	X	12	0.01
1920x1080		12	0.01
640x640		12	0.01
640x640	X	12	0.01
Teste PLANIFICADO			0.01
			0.01

Table 1. Configurações de Treinamento

A tabela fornecida detalha as configurações de treinamento e serve como referência para organizar e entender os experimentos realizados. Cada entrada na tabela indicam as variações nos conjuntos de dados, a presença ou ausência de data augmentation, o número de épocas (EPOCHS) e a taxa de aprendizado (LR). A escolha de avaliar o desempenho em diferentes resoluções e a aplicação do data augmentation oferece insights sobre robustez e adaptabilidade de cada modelo.

6. Resultados

Nesta seção, iremos avaliar os resultados obtidos dos treinamentos para cada comparação da tabela [Table 1]. Para a avaliação dos experimentos, a métrica mAP@50 será utilizada para mensurar a precisão dos modelos de detecção dos postes. A sigla mAP representa a Precisão Média de uma série de pontos de operação, e o sufixo @50 indica um overlap de 50 entre a bounding box predita e de referência. A métrica é calculada pela fórmula:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$$

Onde N é o número total de classes e AP_i é a Precisão Média para uma classe específica i .

Modelo	Dataset	AUG	ÉPOCAS	LR	mAP@50 (%)
Faster RCNN	1920x1080	X	12	0.01	98.8
	1920x1080		12	0.01	98.0
	640x640		12	0.01	86.0
	640x640	X	12	0.01	96.5
	Teste PLANIFICADO				88.0
Retinanet	1920x1080	X	12	0.01	94.0
	1920x1080		12	0.01	92.0
	640x640		12	0.01	92.0
	640x640	X	12	0.01	95.0
	Teste PLANIFICADO				85.3
DETR	1920x1080	X	12	0.01	97.6
	1920x1080		12	0.01	92.0
	640x640		12	0.01	90.1
	640x640	X	12	0.01	93.0
	Teste PLANIFICADO				88.4
DINO	1920x1080	X	12	0.01	98.5
	1920x1080		12	0.01	98.1
	640x640		12	0.01	95.8
	640x640	X	12	0.01	98.4
	Teste PLANIFICADO				91.3

Table 2. Comparação de Modelos com mAP@50 no conjunto de teste.

Segue abaixo os gráficos de comparação das métricas de treino e imagens de referência:

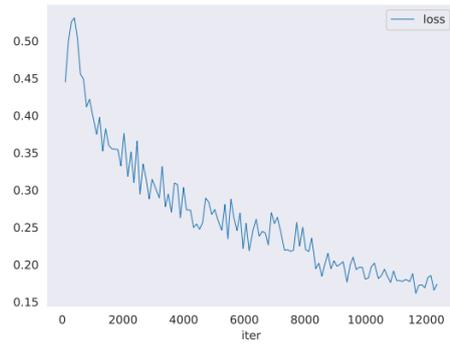


Figure 6. Treinamento Tamanho Original com Augmentation Faster R-CNN.

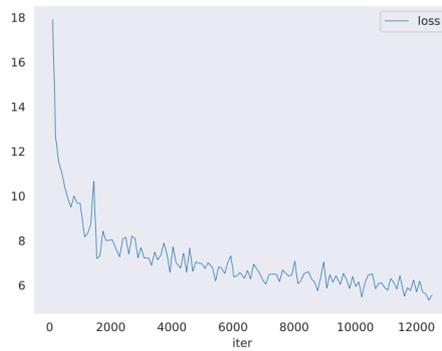


Figure 7. Treinamento Tamanho Original com Augmentation RetinaNet.

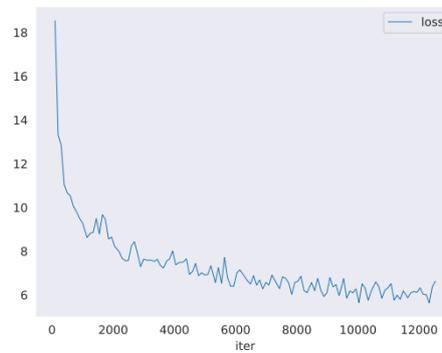


Figure 8. Treinamento Tamanho Original com Augmentation DETR.

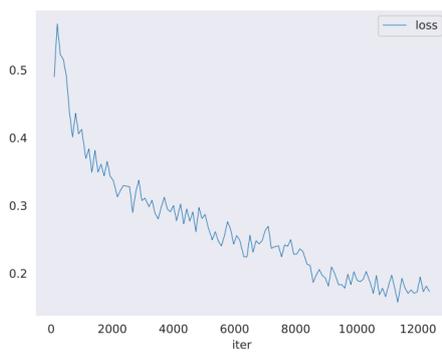


Figure 9. Treinamento Tamanho Original com Augmentation DINO.



Figure 10. Exemplo de Inferência 1.

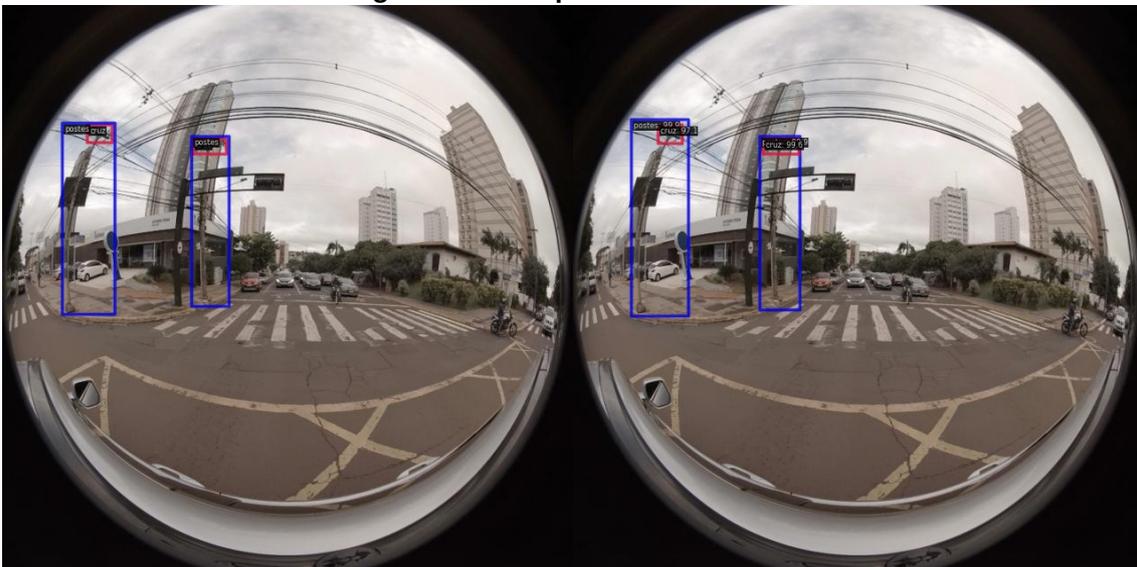


Figure 11. Exemplo de Inferência 2.

Figure 12. A esquerda o que foi anotado, a direita o que foi predito.



Figure 13. Inferência em imagem planejada 1.



Figure 14. Inferência em imagem planejada 2.

Figure 15. Exemplos de Inferência em imagens planejadas.

Ao analisar os gráficos de loss durante o treinamento, notamos a ausência de overfitting. Todos os modelos demonstram uma convergência estável no decorrer das épocas, isso é fundamental pois indica que o modelo não está simplesmente "memorizando" os exemplos de treinamento, mas sim aprendendo padrões relevantes. Essa convergência contribui para a confiabilidade dos modelos ao serem aplicados ao mundo real.

Pode-se afirmar que todos os modelos tiveram bons resultados em todas as versões do dataset. Independente da distorção causada ao poste pelas imagens fisheye, os modelos conseguiram manter uma precisão boa até mesmo nas imagens planificadas, isso sugere que o modelo consegue generalizar bem para imagens com diferentes perspectivas. Podemos concluir que o modelo DINO obteve consistentemente os melhores resultados em quatro das cinco configurações avaliadas apresentando as maiores taxas de mAP@50, isso se deve por ele ser capaz de aprender características mais complexas e abstratas das imagens, lidando muito bem com o problema da distorção nas imagens fisheye. O modelo Faster R-CNN apesar de mais antigo, conseguiu a melhor métrica no Dataset com o tamanho original da imagem com aumento de dados.

7. Conclusão

A conclusão deste estudo evidencia a robustez e eficácia de todos os modelos avaliados na detecção de postes em diferentes cenários, destacando a capacidade de adaptação desses algoritmos a variações como imagens fisheye e planificadas. Os resultados indicam que, mesmo diante das distorções causadas pelas imagens fisheye, todos os modelos mantiveram níveis de precisão satisfatórios, sugerindo uma notável capacidade de generalização para diferentes perspectivas.

Entre os modelos avaliados, o DINO se destacou como o mais eficiente, alcançando os melhores resultados de precisão em diversas configurações de dataset. Essa performance superior pode ser atribuída à abordagem inovadora do DINO na utilização de redes neurais para a detecção de objetos, evidenciando sua capacidade de aprendizado e adaptação em contextos desafiadores.

Este artigo reforça a viabilidade da visão computacional em conjunto com a inteligência artificial como uma abordagem promissora para a detecção de postes e seus componentes. A capacidade desses modelos de lidar com diferentes condições e perspectivas ressalta seu potencial para aplicações práticas, incluindo a detecção de problemas na rede elétrica. Essa abordagem mostra-se crucial para aprimorar a eficiência e confiabilidade dos sistemas de distribuição de energia, destacando a importância crescente da visão computacional no contexto da gestão inteligente de infraestruturas elétricas.

Além disso, destaca-se que essa abordagem se mostra especialmente eficiente para a realidade da cidade de Campo Grande. Considerando as características urbanas e as demandas específicas dessa localidade, a aplicação de modelos de detecção de postes baseados em visão computacional e inteligência artificial oferece uma solução promissora. A capacidade de lidar com diferentes condições climáticas e perspectivas, como as proporcionadas pelas imagens fisheye e planificadas, torna esses modelos adaptáveis a ambientes urbanos diversificados. Todos os modelos avaliados com a ajuda da biblioteca MMDetection se provaram uma escolha eficaz para aplicações de detecção de postes e componentes e mais para frente de toda malha elétrica.

8. Referências

References

- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J.-M., editors, *Computer Vision – ECCV 2020*, pages 213–229, Cham. Springer International Publishing.
- Carranza-García, M., Torres-Mateo, J., Lara-Benítez, P., and García-Gutiérrez, J. (2021). On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data. *Remote Sensing*, 13(1).
- Fambrini, F., Iano, Y., Dueñas, A., Ambrosio, R. R. A., Caetano, D. G., and Rangel, A. (2020). Deep learning based system to electric distribution network inspection. In *International Conference on Digital Image Processing*.
- Ohashi, A., Tanaka, Y., Masuyama, G., Umeda, K., Fukuda, D., Ogata, T., Narita, T., Kaneko, S., Uchida, Y., and Irie, K. (2016). Fisheye stereo camera using equirectangular images. In *2016 11th France-Japan 9th Europe-Asia Congress on Mechatronics (MECATRONICS) /17th International Conference on Research and Education in Mechatronics (REM)*, pages 284–289.
- Ramlal, C. J., Singh, A., Rocke, S., Manninen, H., Kilter, J., and Landsberg, M. (2020). Toward automated utility pole condition monitoring: A deep learning approach. In *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, pages 255–259.
- Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L. M., and Shum, H.-Y. (2022). Dino: Detr with improved denoising anchor boxes for end-to-end object detection.