



DATASET WARP-C:

1. **Tarefa:** Classificação de resíduos recicláveis
2. **Dados:** 8.823 imagens, 28 categorias
3. **Desafios:**
 - Classes desbalanceadas (algumas com 24 imagens)
 - Dimensões variadas (35×40 a 668×703 px)
 - Poucos dados por classe



ESCOLHA DA ESTRATÉGIA DE APRENDIZADO

1. **Tipo:** Aprendizado Supervisionado
2. **Justificativa:** Possui rótulos → adequado para classificação
3. **Técnicas avaliadas:**
 - Modelos clássicos: MLP, k-NN, SVM, Random Forest, Gradient Boosting
 - Modelos baseados em Redes Neurais

FROM SCRATCH (TREINAMENTO DO ZERO)

1. A rede é treinada com **pesos aleatórios**.
2. Requer **grande volume de dados e muito tempo de treinamento**.
3. Pouco recomendado para datasets pequenos e desbalanceados.

TRANSFER LEARNING

1. Reutiliza conhecimento prévio de redes treinadas no ImageNet;
2. Ideal para datasets **pequenos e desbalanceados**.
3. Evita overfitting e acelera o treinamento.
4. Melhora desempenho mesmo com **poucas amostras por classe**.

FINE-TUNING PARCIAL

1. Congela as camadas iniciais e ajusta apenas as finais.
2. **EfficientNet-B0 ou B2** com fine-tuning parcial é a melhor escolha para o WaRP-C.
3. ResNet50 serve como benchmark secundário.

PIPELINE RECOMENDADO

1. **Pré-processamento:** resize 224×224 + padding.
2. **Data Augmentation:** flip, rotação, brilho, zoom com foco nas classes minoritárias.
3. **Treinamento:** utilizar EfficientNet-B0 congelando as camadas iniciais e treinando apenas as camadas finais (classifier + FC head).
4. **Ajustes:** oversampling seletivo ou ajuste de pesos (Class Weights no Loss) conforme necessário, análise do F1 por classe.

MÉTRICAS DE AVALIAÇÃO

Como o dataset é **desbalanceado**, usar apenas **acurácia (accuracy)** pode mascarar o desempenho real. Por isso, recomenda-se: **Accuracy, Precision, Recall, F1-Score (Macro) e Matriz de Confusão**.



MODELOS CLÁSSICOS

1. Exigem extração manual de atributos (features), como cor, textura, forma: não trabalham diretamente com imagens brutas.
2. Baixo desempenho com alta dimensionalidade: podendo sofrer com overfitting ou lentidão
3. Pouca capacidade de capturar padrões espaciais complexos, como contornos, formas, partes do objeto



MODELOS BASEADOS EM REDES NEURAIS

1. Extração automática de padrões visuais
2. Robustez a variações de tamanho, posição e ruído:
3. Transfer Learning + Fine-Tuning: Permite aproveitar redes pré-treinadas e ajustar apenas as últimas camadas — útil para **datasets pequenos ou desbalanceados**

EFFICIENTNET (B0–B2 OU V2)

1. **Escalabilidade eficiente:** usa *compound scaling* para balancear profundidade, largura e resolução, otimizando desempenho com baixo custo computacional.
2. **Robustez em transferência:** apresenta alta acurácia mesmo em conjuntos de dados pequenos e desbalanceados.
3. **Custo-benefício:** treinável com GPUs comuns (224×224), ideal para ambientes como Google Colab.
4. **Acurácia 88–92% | F1-Score: 70–80%.**

RESNET50

1. **Confiabilidade e estabilidade:** é uma arquitetura amplamente adotada, com excelente desempenho em tarefas de classificação com *Transfer Learning*.
2. **Ampla documentação e suporte,** Menor F1 em classes minoritárias e apesar de mais pesada, ainda é um excelente benchmark
3. **Acurácia 86–90% | F1-Score: 68–78%**

- **InceptionV3:** Boa acurácia, uso de múltiplas escalas, **Complexidade elevada e custo computacional**.
- **VGG16/VGG19:** Arquitetura simples e fácil de implementar, **Muito pesada, propensa a overfitting**.
- **MobileNetV2:** Leve, ideal para **dispositivos móveis (edge)**, **Baixa acurácia em tarefas mais complexas**.
- **DenseNet:** Excelente fluxo de gradiente, **Elevado consumo de memória**.
- **ViT/Swin:** Alto potencial em grandes bases de dados, **Alta demanda computacional, requer muitos dados, Setup sofisticado, risco de underfitting**.