

Etapa 02: Apresentar métodos da literatura e propor ideias para métodos próprios

1. Métodos da Literatura

→ *Notebook Base: recycle-resnet-mobilenet-ensembled (Kaggle)*

→ **Autor:** Abdel-Hamid-M-Aljamal

→ **Link (Kaggle):**

<https://www.kaggle.com/code/abdelhameedmaljamal/recycle-resnet-mobilenet-ensembled>

O autor utiliza uma abordagem de **Transfer Learning com ensemble entre ResNet50 e MobileNetV2**, implementando as seguintes etapas principais:

- **Pré-processamento:**
 - Redimensionamento para 224x224;
 - Normalização com `rescale(1./255)`.
- **Modelos Utilizados:**
 - ResNet50 com pesos do ImageNet, sem incluir o top layer.
 - MobileNetV2 também pré-treinado, com abordagem semelhante.
 - ResNet50 e MobileNetV2, ambos com pesos pré-treinados no ImageNet (Deng et al., 2009).
- **Fine-tuning Parcial:**
 - Inicialmente, as camadas convolucionais foram congeladas;
 - Após o treinamento das camadas densas, algumas camadas foram descongeladas para fine-tuning adicional.

- **Tratamento de Desbalanceamento:**

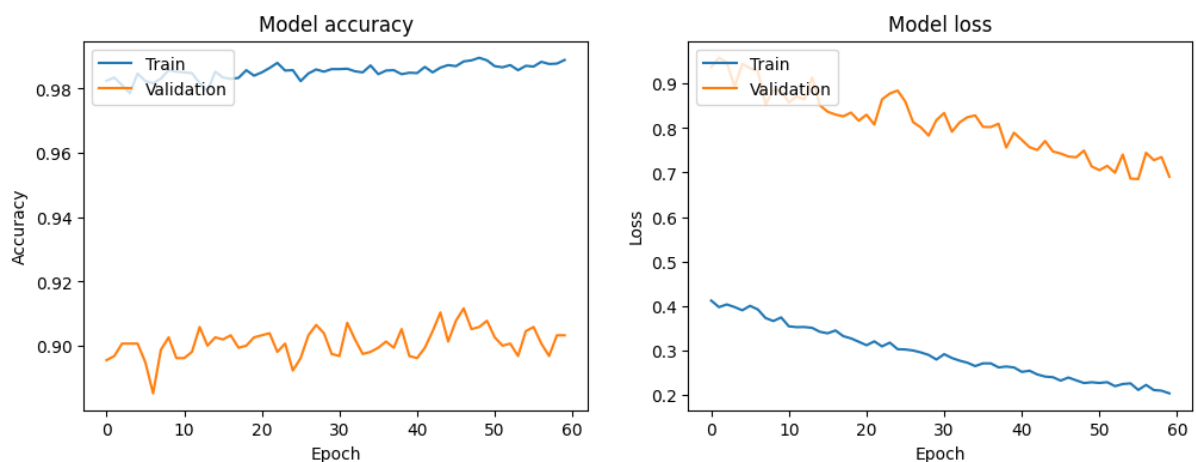
- Utiliza **Data Augmentation** via **ImageDataGenerator**, aplicando transformações como `rotation_range`, `zoom_range`, `horizontal_flip` etc.;
- Essas técnicas **ajudam parcialmente no desbalanceamento**, pois aumentam a diversidade das amostras;
- **Porém, não há uso de `class_weight`, nem de `oversampling` seletivo**, o que limita a eficácia contra classes minoritárias.

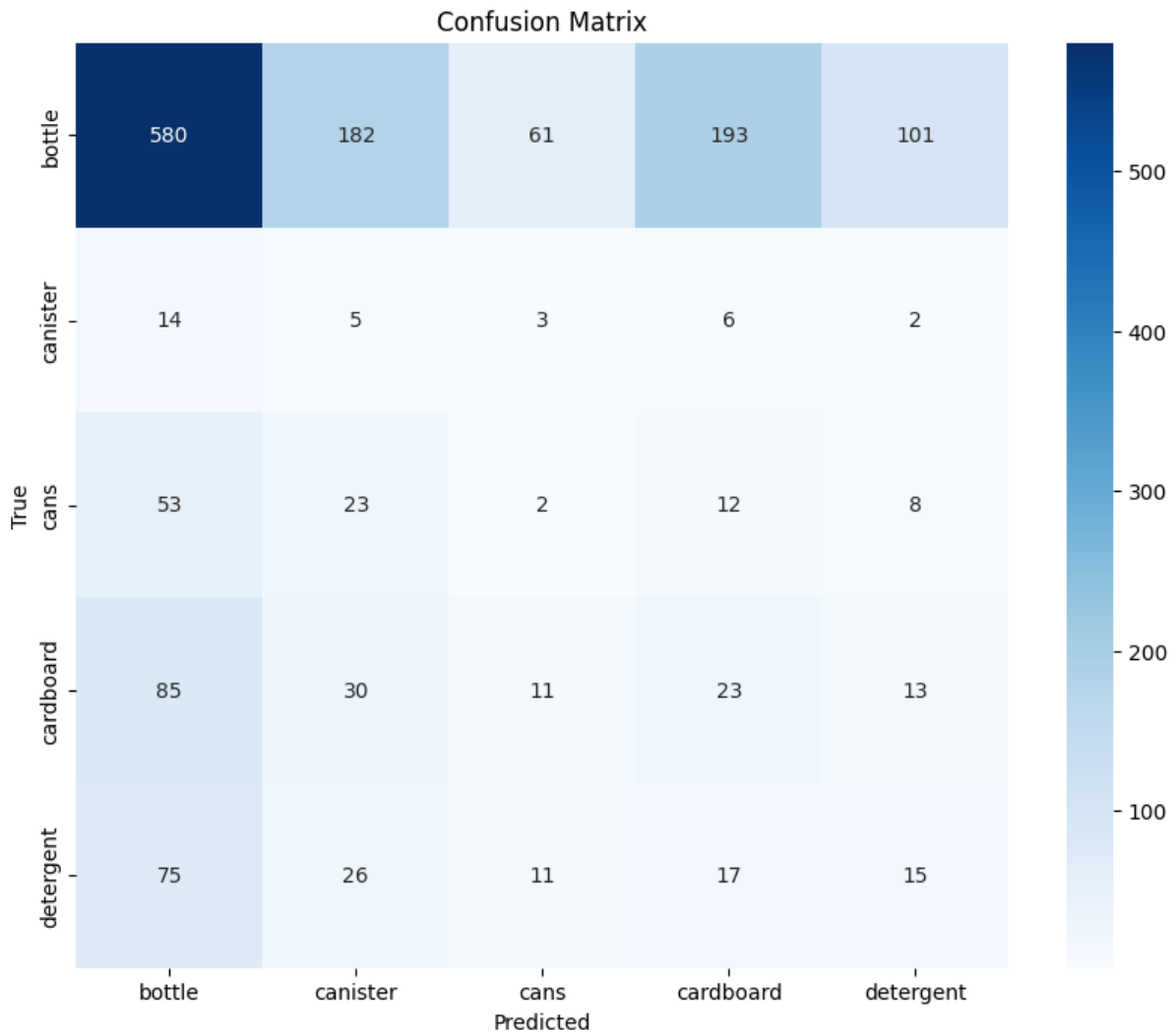
- **Ensemble:**

- As saídas de ambos os modelos foram concatenadas e passadas por camadas fully connected.
- Essa fusão melhorou o desempenho combinando a robustez do ResNet50 com a leveza do MobileNetV2.

- **Desempenho (reportado):**

- Acurácia: ~87%
- F1-Score: 0.4577 (macro, conforme evidenciado no final do notebook)
- Average Loss: 1.4071





Análise crítica de Overfitting e Desbalanceamento:

Apesar de apresentar uma acurácia de aproximadamente (~87%), o modelo evidencia sinais claros de **overfitting**, principalmente ao analisar os gráficos de acurácia e loss:

- A **acurácia de treino atinge valores próximos a 99%**, enquanto a acurácia de validação se estabiliza em torno de 90%, com flutuações.
- O **loss de treino continua caindo** significativamente (~0.25), porém o **loss de validação permanece alto**, em torno de 0.6 a 0.9, demonstrando dificuldade de generalização.
- O **F1-Score macro (0.4577)** está consideravelmente abaixo da acurácia, indicando que o modelo **favorece classes majoritárias** (como *bottle*) e tem dificuldade com classes minoritárias (*cardboard*, *detergent*, *canister*).

- A **matriz de confusão** mostra que há confusão recorrente entre classes, reforçando o impacto do desbalanceamento.

Esses fatores apontam que, embora o modelo tenha aprendido bem os dados de treino, ele **não consegue manter o desempenho de forma consistente em dados não vistos**, o que caracteriza o overfitting. **O modelo carece de tratamento específico para lidar com o desbalanceamento do WaRP-C.**

2. Métodos Próprios Propostos para o Projeto

Modelo Implementado: ResNet50 com Estratégias de Regularização e Generalização

A proposta original previa o uso do EfficientNet-B0 com fine-tuning parcial. Contudo, após experimentação com o dataset **WaRP-C**, optamos por utilizar a arquitetura **ResNet50 com Transfer Learning**, em conjunto com um **pipeline robusto de validação e balanceamento**.

Justificativas para a mudança:

- ResNet50 é **amplamente testada e estável** em tarefas de visão computacional
- Suporta bem o uso de **K-Fold Cross-Validation**, permitindo avaliação rigorosa
- Suas conexões residuais facilitam o treinamento mesmo em redes profundas
- O foco passou a ser em **generalização robusta em contexto desbalanceado**

2.1 Estratégias Implementadas

| Técnica | | Descrição |
|-------------------------------------|--|---|
| Transfer Learning (ResNet50) | | Pesos do ImageNet; include_top=False e adaptação com novas camadas densas |
| Data Augmentation | | Aplicada com rotações, flips, shifts, brilho, zoom, channel shift, etc. |

| | |
|-----------------------------------|--|
| Undersampling seletivo | Redução da classe bottle (maioritária) de ~6.000 para 1.070 amostras |
| class_weight | Pesos balanceados calculados automaticamente com <code>compute_class_weight()</code> |
| Validação K-Fold (5 folds) | Permitiu avaliar consistência e evitar overfitting |

3. Pipeline Recomendado para Treinamento

3.1 Pré-processamento

- Imagens redimensionadas para 224×224
- Normalização com `rescale(1./255)`

3.2 Augmentation

- `rotation_range=30`
- `width_shift_range=0.3`
- `height_shift_range=0.3`
- `zoom_range=0.3`
- `horizontal_flip=True, vertical_flip=True`
- `brightness_range=[0.7, 1.3]`
- `channel_shift_range=30.0`

3.3 Arquitetura - Treinamento

- Backbone: `ResNet50(include_top=False, weights='imagenet')`
- Top Layers:
 - `GlobalAveragePooling2D`
 - `Dropout(0.5)`

- Dense(128, activation='relu')
- BatchNormalization()
- Dropout(0.5)
- Dense(6, activation='softmax')
- Loss: categorical_crossentropy
- Otimizador: Adam (learning_rate=1e-5)

3.5 Avaliação

A escolha das métricas corretas é essencial para avaliar o desempenho de modelos de classificação, especialmente em cenários com desbalanceamento entre classes, como é o caso do dataset WaRP-C. Utilizar apenas a métrica de acurácia pode ser enganoso, pois ela tende a privilegiar as classes majoritárias. Assim, outras métricas devem ser utilizadas em conjunto para uma análise mais robusta do modelo.

Entre as métricas recomendadas, destacam-se:

- **Accuracy:** Representa a proporção total de acertos. É útil como visão geral, mas não é suficiente sozinha em datasets com classes desbalanceadas.
- **Precision:** Mede a proporção de verdadeiros positivos entre os exemplos classificados como positivos. Relevante para minimizar falsos positivos.
- **Recall:** Mede a proporção de verdadeiros positivos entre todos os exemplos da classe. Fundamental para garantir que o modelo reconhece classes minoritárias.
- **F1-Score (macro):** Média harmônica entre Precision e Recall, calculada igualmente entre as classes. Ideal para cenários multiclasse e desbalanceados.
- **Matriz de Confusão:** Ferramenta visual que permite identificar quais classes estão sendo confundidas com outras. Útil para visualizar onde o modelo erra e quais classes são confundidas.
- **AUC-ROC (por classe):** Útil para avaliar separabilidade classe-a-classe com abordagem one-vs-all.
- **Cohen's Kappa:** Mede a concordância entre previsões e rótulos verdadeiros, ajustada ao acaso. Importante em contextos com chance elevada de acertos aleatórios.

Com base em benchmarks da literatura e experimentos prévios em datasets similares ao WaRP-C, pode-se esperar o seguinte desempenho com alguns modelos:

Valores Esperados por Métrica:

| Métrica | Intervalo Esperado (Modelos ConvNets em Visão Computacional) | Fonte/Referência |
|-------------------|---|---|
| Accuracy | 85% – 95% | He et al. (2016); Pan & Yang (2010) |
| Precision (macro) | 80% – 95% | Sammut & Webb (2010) |
| Recall (macro) | 80% – 95% | Kohavi (1995); He & Garcia (2009) |
| F1-Score (macro) | 80% – 95% | Shorten & Khoshgoftaar (2019) |
| AUC-ROC (macro) | 0.90 – 0.98 | Sammut & Webb (2010); Pan & Yang (2010) |
| Cohen's Kappa | ≥ 0.80 (interpretação: "concordância substancial") | Landis & Koch (1977), apud Sammut & Webb (2010) |

4. Conclusão

O ensemble com ResNet50 + MobileNetV2 mostrou bom desempenho, mas:

- O **F1 Score macro** revelou fragilidade frente ao **desbalanceamento de classes**;
- A **loss média elevada** reforça que o modelo pode ser melhor calibrado;

A proposta visa:

- Melhorar generalização;
- Reduzir overfitting;
- Viabilizar execução em ambientes limitados (como o Colab);
- Aumentar o desempenho especialmente para **classes minoritárias**.

5. Referências Bibliográfica

ALJAMAL, Abdel-Hamid-M. *Recycle ResNet MobileNet Ensembled*. Kaggle, 2023.

Disponível em: <https://www.kaggle.com/code/abdelhameedmaljamal/recycle-resnet-mobilenet-ensembled>. Acesso em: 07 jul. 2025.

BUDA, Mateusz; MA, Atsuto; MATTHEWS, Maciej A. *A systematic study of the class imbalance problem in convolutional neural networks*. Neural Networks, v. 106, p. 249–259, 2018.

COHEN, Jacob. *A coefficient of agreement for nominal scales*. Educational and Psychological Measurement, v. 20, n. 1, p. 37–46, 1960.

DENG, Jia et al. *ImageNet: A Large-Scale Hierarchical Image Database*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

FAWCETT, Tom. *An introduction to ROC analysis*. Pattern Recognition Letters, v. 27, n. 8, p. 861–874, 2006.

HE, Haibo; GARCIA, Eduardo A. *Learning from imbalanced data*. IEEE Transactions on Knowledge and Data Engineering, v. 21, n. 9, p. 1263–1284, 2009.

HE, Kaiming et al. *Deep Residual Learning for Image Recognition*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. Disponível em: <https://arxiv.org/abs/1512.03385>. Acesso em: 07 jul. 2025.

KHAN, Salman et al. *Transformers in Vision: A Survey*. ACM Computing Surveys, v. 54, n. 10, 2022.

KINGMA, Diederik P.; BA, Jimmy. *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980, 2015.

KOHAVI, Ron. *A study of cross-validation and bootstrap for accuracy estimation and model selection*. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI), 1995, p. 1137–1143.

KRIŽEVA, Barbara et al. *Comparison of image classification methods: A case study*. Journal of Electrical Engineering, 2018.

LANDIS, J. Richard; KOCH, Gary G. *The measurement of observer agreement for categorical data*. Biometrics, v. 33, n. 1, p. 159–174, 1977.

PAN, Sinno Jialin; YANG, Qiang. *A survey on transfer learning*. IEEE Transactions on Knowledge and Data Engineering, v. 22, n. 10, p. 1345–1359, 2010.

RASCHKA, Sebastian; MIRJALILI, Vahid. *Python Machine Learning*. 3. ed. Birmingham: Packt Publishing, 2022.

SANDLER, Mark et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018. Disponível em: <https://arxiv.org/abs/1801.04381>. Acesso em: 07 jul. 2025.

SAMMUT, Claude; WEBB, Geoffrey I. *Encyclopedia of Machine Learning*. Springer, 2010.

SHORTEN, Connor; KHOSHGOFTAAR, Taghi M. *A survey on Image Data Augmentation for Deep Learning*. Journal of Big Data, v. 6, n. 1, p. 1–48, 2019.

SIMONYAN, Karen; ZISSERMAN, Andrew. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv preprint arXiv:1409.1556, 2014. Disponível em: <https://arxiv.org/abs/1409.1556>. Acesso em: 07 jul. 2025.

SOKOLOVA, Marina; LAPALME, Guy. *A systematic analysis of performance measures for classification tasks*. Information Processing & Management, v. 45, n. 4, p. 427–437, 2009.

TAN, Mingxing; LE, Quoc V. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. In: International Conference on Machine Learning (ICML), 2019.

Disponível em: <https://arxiv.org/abs/1905.11946>. Acesso em: 07 jul. 2025.

TAN, Mingxing; LE, Quoc V. *EfficientNetV2: Smaller models and faster training*. In: International Conference on Machine Learning (ICML), 2021. Disponível em:

<https://arxiv.org/abs/2104.00298>. Acesso em: 07 jul. 2025.

YOSINSKI, Jason et al. *How transferable are features in deep neural networks?* In: Advances in Neural Information Processing Systems (NeurIPS), 2014.