Abordagens para minimizar o custo de rotulagem em CNNs

Áquila Oliveira Souza – 2021019327
Felippe Veloso Marinho – 2021072260
Universidade Federal de Minas Gerais (UFMG)
Professores: Antônio de Pádua Braga e Frederico G. Coelho

---

1. Introdução

As Redes Neurais Convolucionais (CNNs) representam o estado da arte em tarefas de visão computacional, alcançando desempenho superior em classificação, detecção e segmentação de imagens. No entanto, esses métodos dependem de grandes volumes de exemplos rotulados, o que torna o processo de anotação um gargalo crítico. Em domínios especializados, como medicina, inspeção industrial ou satélites, a anotação requer especialistas, tornando-se cara e demorada.

Nesse contexto, surgem três linhas de pesquisa destinadas a reduzir a dependência de rótulos: (i) o Aprendizado Semi-Supervisionado (SSL), que explora dados não rotulados; (ii) o Aprendizado Ativo (AL), que maximiza o uso dos rótulos solicitados; e (iii) tarefas de autoaprendizado (pretext tasks), que permitem aprender representações úteis sem qualquer rótulo humano.

O objetivo central deste trabalho é investigar, de forma integrada, como SSL, AL e pretext tasks podem reduzir o custo total de rotulagem mantendo desempenho competitivo em CNNs.

2. Revisão da Literatura

Dada a limitação no processo de rotulagem para treinar CNNs, a literatura recente indica que estratégias eficientes combinam Aprendizado Semi-Supervisionado (SSL) e Aprendizado Ativo (Active Learning — AL), explorando a abundância de dados não rotulados e usando de forma moderada os rótulos humanos.

2.1. Aprendizado Semi-Supervisionado (SSL)

O SSL busca aproveitar dados não rotulados impondo que o modelo produza predições consistentes sob pequenas perturbações. Um dos métodos mais influentes nesse contexto é o *FixMatch* (Sohn et al., 2020), que combina:

- **Pseudo-rotulagem de alta confiança**: apenas pseudo-rótulos cuja confiança excede um limiar τ são utilizados no treinamento;

- **Regularização por consistência**: o pseudo-rótulo gerado para uma versão com fraca alteração (*weak augmentation*) é imposto como alvo para uma versão com forte alteração (*strong augmentation*) da mesma amostra.

O FixMatch destaca-se por sua simplicidade e eficácia, alcançando desempenho competitivo mesmo quando apenas uma pequena fração do conjunto de treinamento possui rótulos.

2.2. Aprendizado Ativo Semi-Supervisionado (AS2L)

Embora o SSL maximize o uso dos dados não rotulados, ele não garante que os rótulos obtidos sejam os mais informativos. O Aprendizado Ativo (AL) resolve essa limitação selecionando amostras que o modelo considera ambíguas. A integração entre AL e SSL é conhecida como *Active Semi-Supervised Learning* (AS2L).

O trabalho *Active Semi-Supervised Learning by Exploring Per-Sample Uncertainty and Consistency* (Luo et al., 2020) propõe um método que combina:

- **Incerteza por amostra**: prioriza amostras com maior entropia na distribuição de predição;
- **Consistência como critério**: seleciona amostras que apresentam baixa consistência sob perturbações, indicando regiões críticas do espaço de decisão.

Essa combinação busca maximizar o ganho informacional a cada anotação humana.

Ao integrar SSL e AL, diversos trabalhos mostram ganhos de eficiência na redução de rótulos necessários para atingir determinada acurácia — justificando a abordagem adotada neste relatório.

2.3. Aprendizagem Auto-Supervisionada (SSL)

A Aprendizagem Auto-Supervisionada (*Self-Supervised Learning* — SSL) surgiu como uma alternativa para reduzir a dependência de grandes conjuntos de dados rotulados, um dos principais gargalos no treinamento de redes neurais modernas. Diferentemente do aprendizado supervisionado tradicional, o SSL cria seus próprios rótulos a partir da estrutura interna dos dados, permitindo que o modelo aprenda representações úteis sem intervenção humana.

Os métodos de SSL geralmente formulam tarefas artificiais, chamadas *tarefas pretexto*, que induzem o modelo a extrair padrões semânticos relevantes. Entre os pretextos mais estudados estão: (1) **rotação de imagens** — prever o ângulo aplicado à entrada (Gidaris et al., 2018); (2) **jigsaw puzzles** — reordenar permutações de *patches* da imagem (Noroozi & Favaro, 2016); (3) **colorização** — reconstruir canais de cor a partir da versão em escala de cinza (Zhang et al., 2016).

Em visão computacional, o SSL demonstrou ganhos significativos por aprender representações gerais transferíveis para tarefas posteriores (*downstream tasks*), mesmo com arquiteturas pequenas ou poucas amostras rotuladas. Revisões recentes destacam que métodos modernos, como SimCLR, MoCo e BYOL, evoluíram para técnicas baseadas em contraste, mas pretextos clássicos permanecem eficazes em cenários computacionalmente restritos.

No contexto deste trabalho, utilizamos três tarefas pretexto (rotação, jigsaw e colo-

rização) com o objetivo de avaliar como diferentes formas de supervisão automática afetam a qualidade das representações e sua interação com estratégias de Aprendizagem Semi-Supervisionada e Aprendizagem Ativa.

3. Metodologia

Nesta seção descrevemos a arquitetura e os componentes do método proposto, que integra um esquema FixMatch-like, aprendizagem ativa e tarefas de pré-treino (pretext tasks) em um fluxo multitarefa de Treinamento Semi-Supervisionado Ativo (ASSL).

3.1. FixMatch

O FixMatch combina pseudo-rotulagem e consistência entre *weak* e *strong augmentation*. A predição a partir de uma imagem fracamente aumentada só é usada como pseudo-rótulo se o modelo estiver suficientemente confiante.

A confiança é definida como a maior probabilidade prevista dentre todas as classes:

$$\mathrm{conf}(x_u) = \max_i p_i^{(w)}, \tag{1}$$

onde $p_i^{(w)}$ são as probabilidades produzidas pelo modelo (após softmax) sobre a versão fracamente aumentada $x_u^{(w)}$. Mantemos o pseudo-rótulo $\hat{y}$ quando

$$\mathrm{conf}(x_u) \geq \tau, \tag{2}$$

com

$$\hat{y} = \arg\max_i p_i^{(w)}. \tag{3}$$

Tipicamente usa-se $\tau = 0.95$ no FixMatch original.

3.1.1 Weak Augmentation

Na etapa de *weak augmentation* aplicamos transformações leves:

$$x_u^{(w)} = \mathrm{weak\_aug}(x_u),$$

e calculamos logits e probabilidades:

$$z^{(w)} = f_\theta(x_u^{(w)}), \qquad p^{(w)} = \mathrm{softmax}(z^{(w)}).$$

3.1.2 Strong Augmentation

Na etapa de *strong augmentation* aplicamos transformações fortes:

$$x_u^{(s)} = \mathrm{strong\_aug}(x_u),$$

e treinamos o modelo para prever o pseudo-rótulo $\hat{y}$ a partir da versão fortemente aumentada:

$$z^{(s)} = f_\theta(x_u^{(s)}), \qquad p^{(s)} = \mathrm{softmax}(z^{(s)}).$$

A perda não supervisionada é então:

$$\mathcal{L}_{uns} = \mathbb{1}[\mathrm{conf}(x_u) \geq \tau] \cdot \mathrm{CE}\left(p^{(s)}, \hat{y}\right), \tag{4}$$

onde $\mathbb{1}[\cdot]$ é a máscara que descarta amostras de baixa confiança.

3.2. Threshold Adaptativo

Substituímos o limiar fixo τ por um limiar adaptativo:

$$\tau_{\text{adapt}} = \min\big(0.95,\ \max(\tau_{\text{base}},\ \mathrm{P90}(\mathrm{conf}))\big), \tag{5}$$

onde P90 é o percentil 90 da distribuição de confianças no batch. Assim, o threshold se ajusta dinamicamente à qualidade das representações.

3.3. Perda Total

A perda total do sistema é a soma ponderada de três componentes:

$$\mathcal{L}_{total} = \mathcal{L}_{sup} + \lambda_u \mathcal{L}_{uns} + \lambda_p \mathcal{L}_{pre}, \qquad (6)$$

onde $\mathcal{L}_{sup}$ é a perda supervisionada (entropia cruzada), $\mathcal{L}_{uns}$ é a perda FixMatch-like, $\mathcal{L}_{pre}$ é a perda da(s) tarefa(s) pretexto e λ_u, λ_p são pesos escalares.

3.4. Aprendizagem Ativa via Entropia

Para seleção de amostras utilizamos a entropia como medida de incerteza:

$$H(p) = -\sum_i p_i \log p_i. \qquad (7)$$

Selecionam-se as k amostras de maior entropia:

$$U_{\text{select}} = \text{TopK}_{x_u \in U}\ H(p(x_u)). \qquad (8)$$

O oráculo anota essas amostras e elas são movidas para o conjunto rotulado L:

$$L \leftarrow L \cup \mathcal{A}(U_{\text{select}}), \qquad U \leftarrow U \setminus U_{\text{select}}.$$

Esse ciclo se repete a cada rodada de aprendizagem ativa.

3.5. Arquitetura e Configuração

O encoder base é o *SmallConvEncoder*: três camadas convolucionais seguidas de batch norm, ReLU e pooling (110k parâmetros). As tarefas pretexto foram:

- **Rotação**: prever ângulo entre {0°,90°,180°,270°};
- **Jigsaw 2×2**: reconstruir a posição correta de 4 quadrantes (24 permutações);
- **Colorização**: reconstruir RGB a partir de escala de cinza com um decodificador adicional (540k parâmetros).

Após pré-treino, o encoder foi congelado e um classificador linear (*linear probe*) foi treinado para avaliar a separabilidade das representações.

3.6. Configurações Experimentais

Os experimentos foram executados em CPU com:

- **MNIST**: 20.000 amostras, 32×32 pixels;
- **STL-10**: 5.000 amostras, 48×48 pixels;
- **Épocas**: 5 (pré-texto) + 5 (linear probe);
- **Lotes (batches)**: 128 (pré-texto), 256 (linear probe);
- **Seed**: 42.

4. Testes e Resultados

Esta seção apresenta a avaliação empírica do método proposto. Os experimentos foram conduzidos com três seeds distintas ($\{13, 42, 123\}$) para avaliar robustez e estabilidade.

4.1. Configuração Experimental

O treinamento segue a função de perda total definida anteriormente. A seleção ativa usa entropia conforme descrito na metodologia.

4.2. Acurácia vs. Tamanho do Conjunto Rotulado (L_{size})

Avalia-se como a acurácia evolui conforme mais amostras são adquiridas via active learning. Apresentamos quatro gráficos:

- Seed 13 — Figura 1
- Seed 42 — Figura 2
- Seed 123 — Figura 3
- Média entre as três seeds — Figura 4

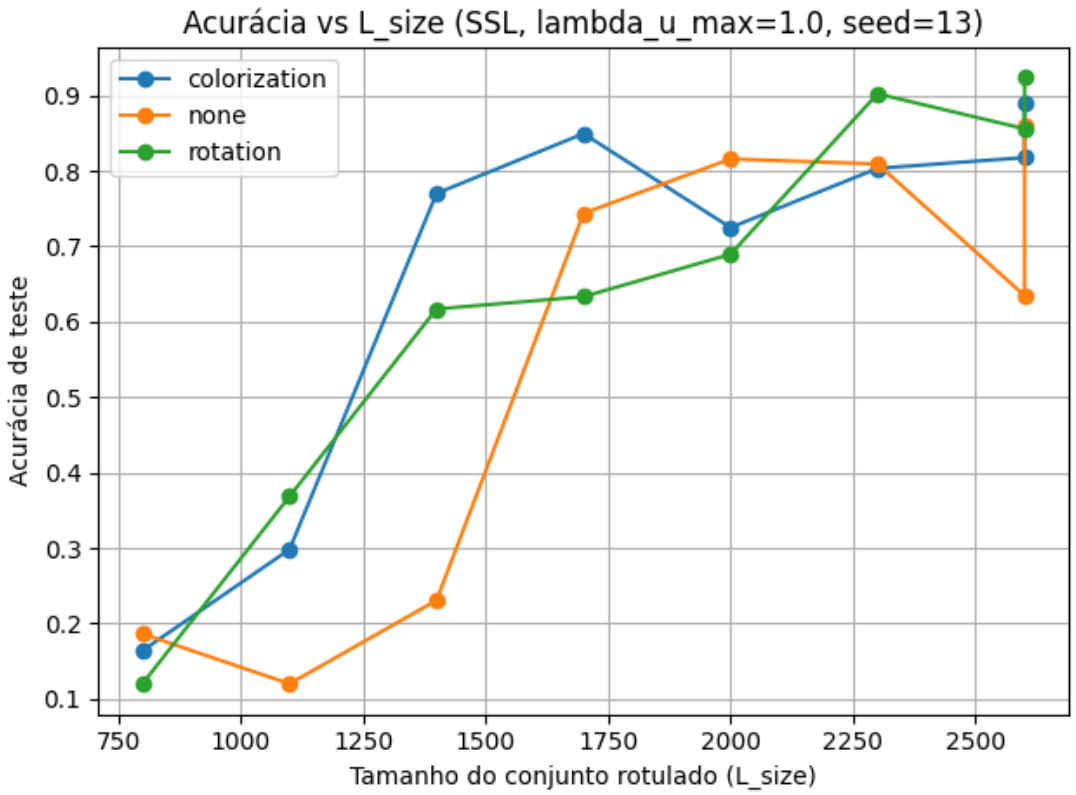

Figura 1: Acurácia vs. L_{size} (seed 13).

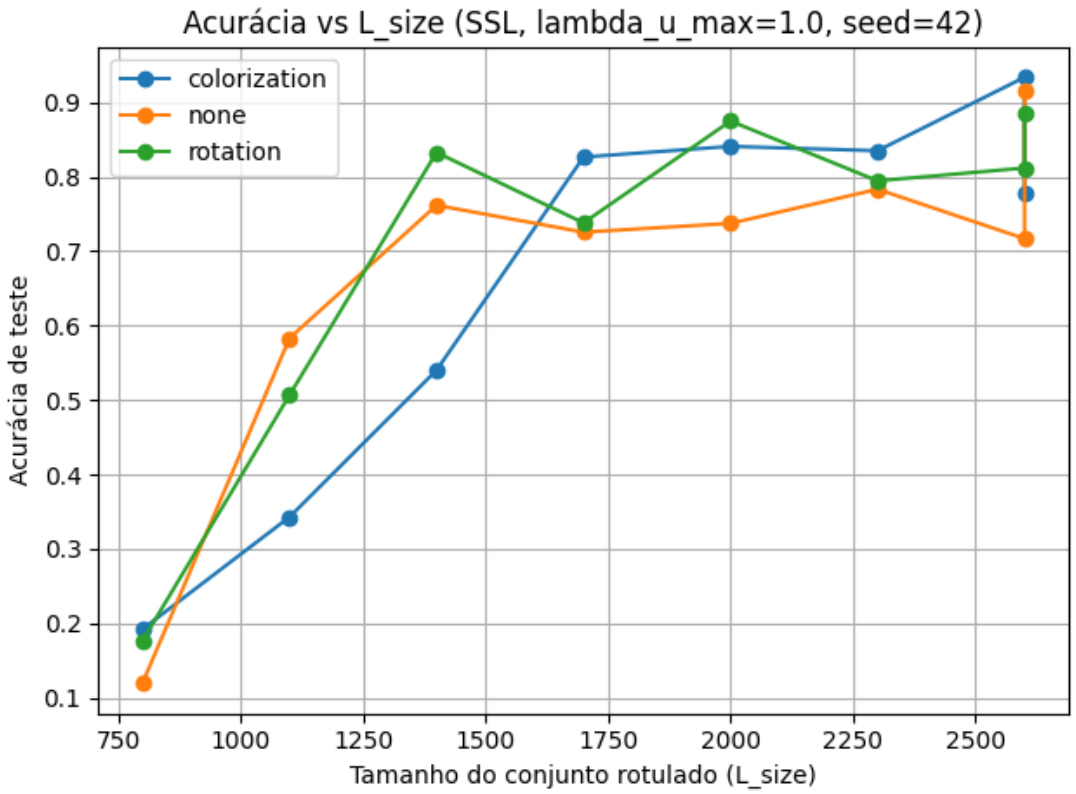

Figura 2: Acurácia vs. L_{size} (seed 42).

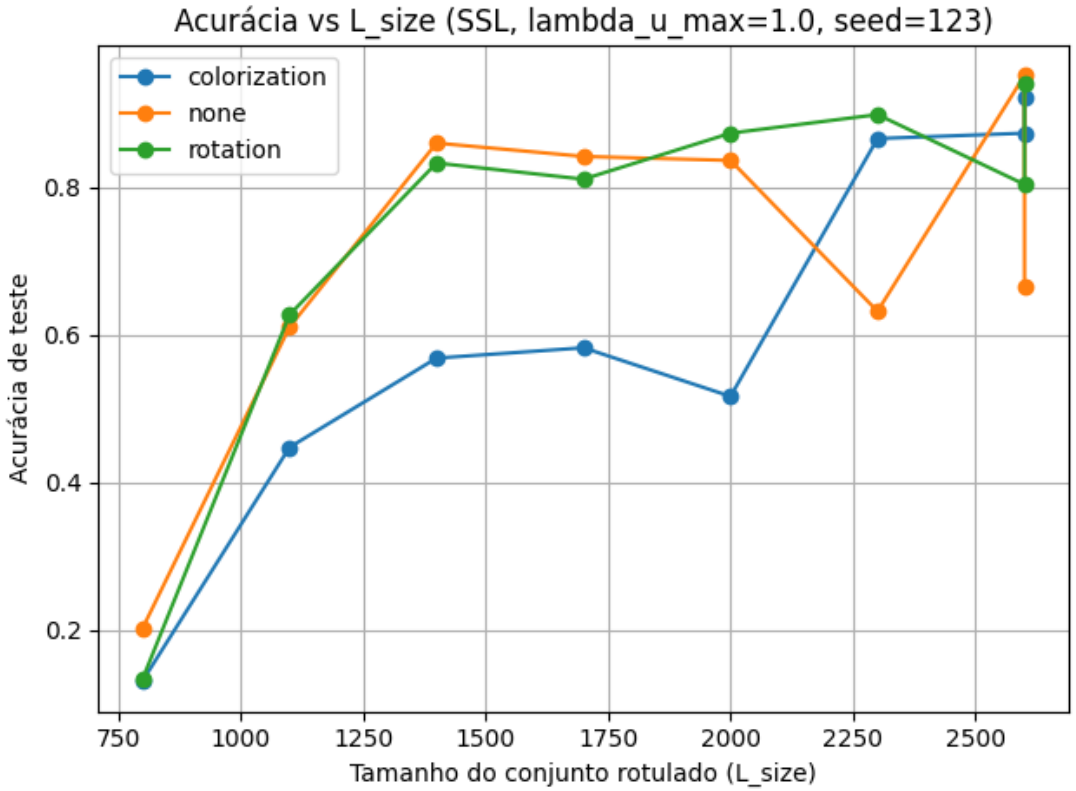

Figura 3: Acurácia vs. L_{size} (seed 123).

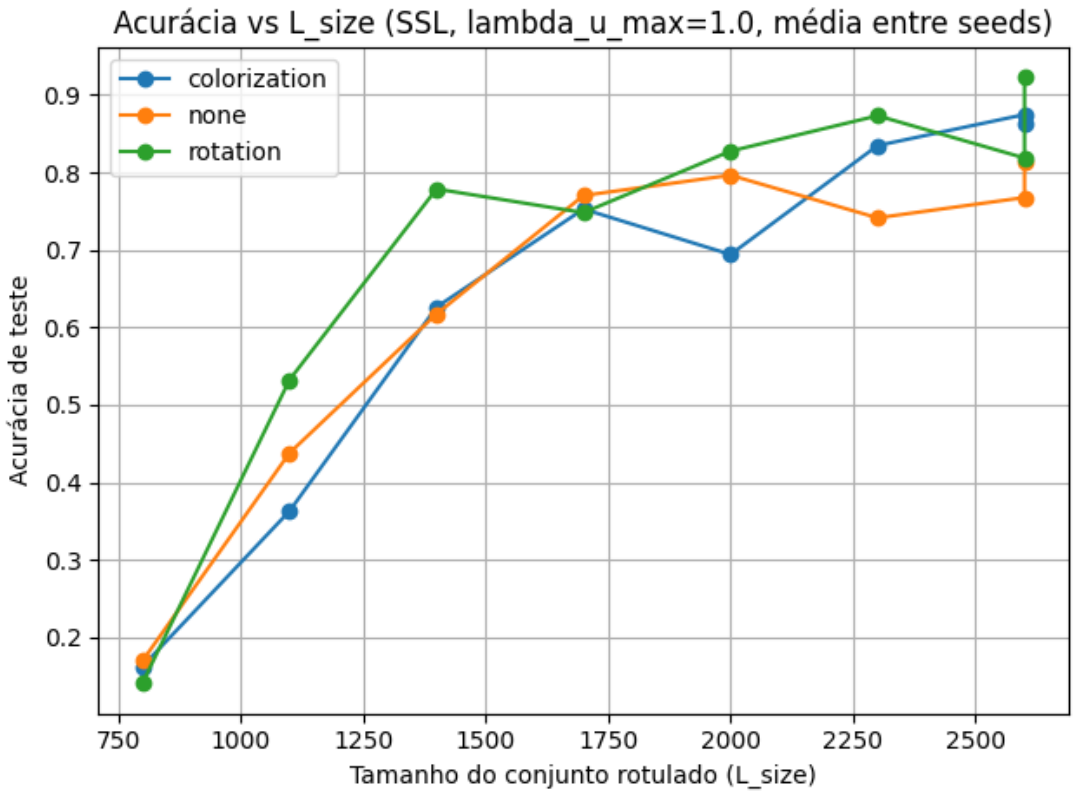

Figura 4: Acurácia média vs. L_{size} (média entre seeds).

A partir desses resultados observamos que métodos com tarefas pretexto (Rotation e Colorization) superam consistentemente o SSL puro; Rotation apresenta curvas sistematicamente superiores; o baseline supervisionado ($\lambda_u = 0$) cresce mais lentamente.

4.3. Acurácia vs. Rodada de Aquisição

Analisamos a acurácia ao longo das rodadas de aquisição para cada seed.

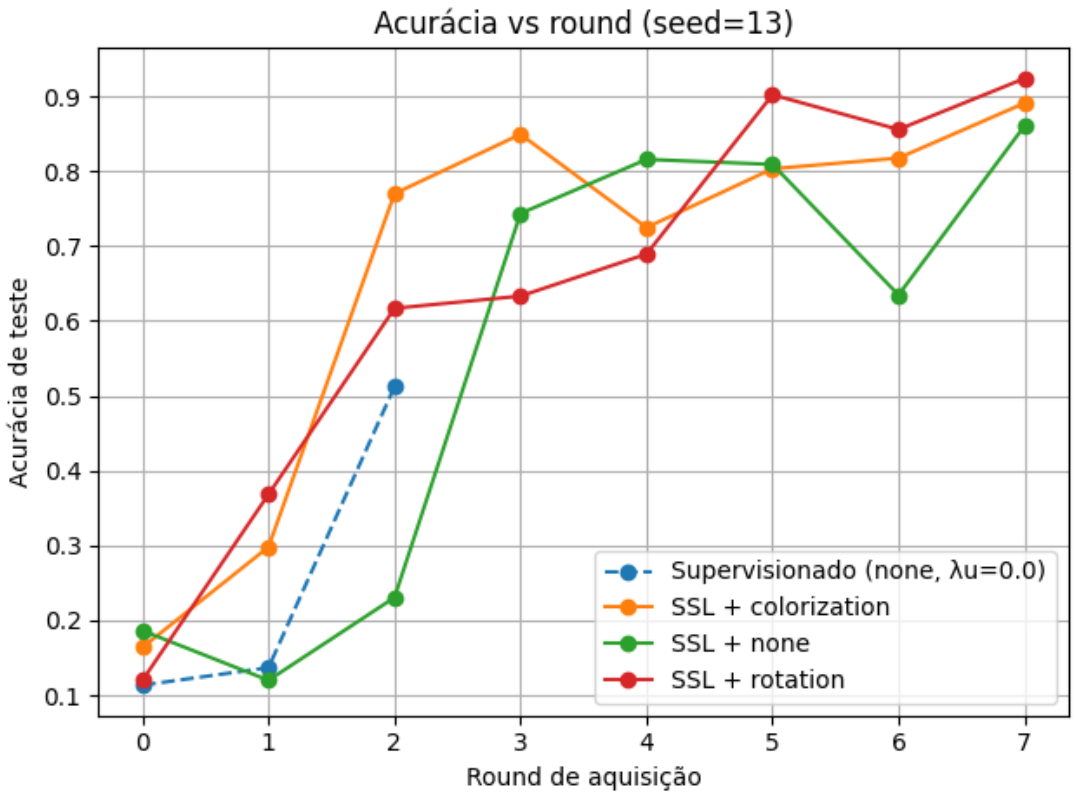

Figura 5: Acurácia vs. rodada (seed 13).

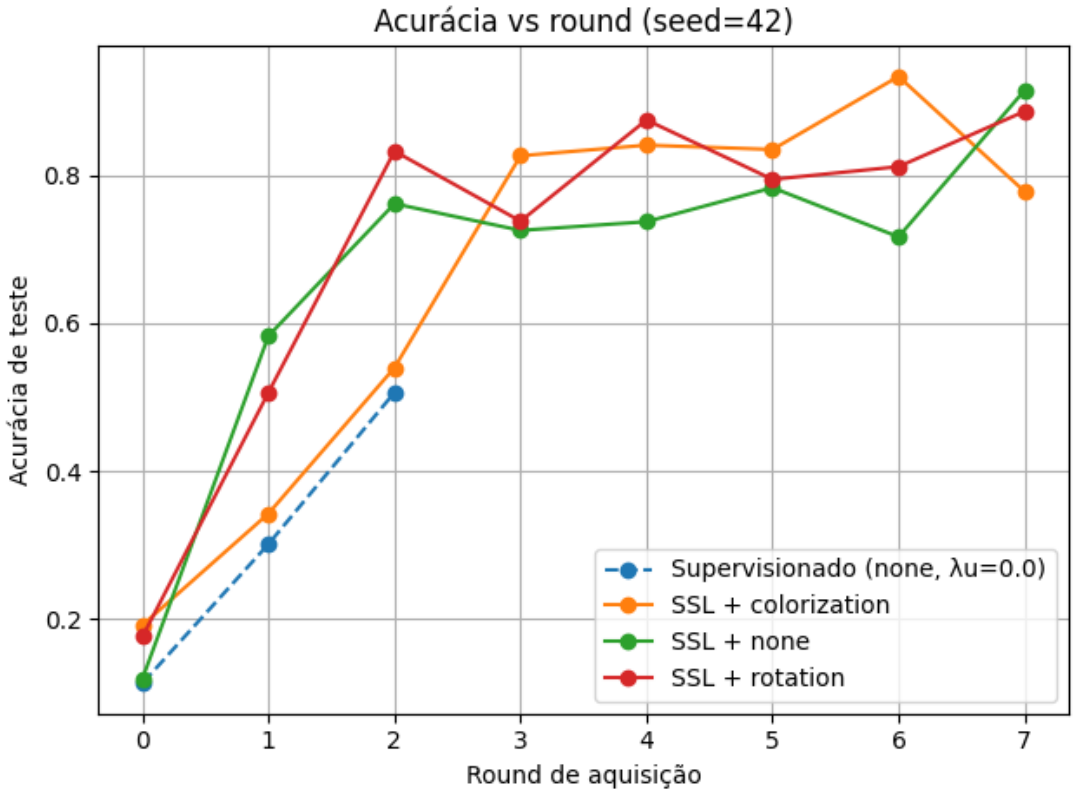

Figura 6: Acurácia vs. rodada (seed 42).

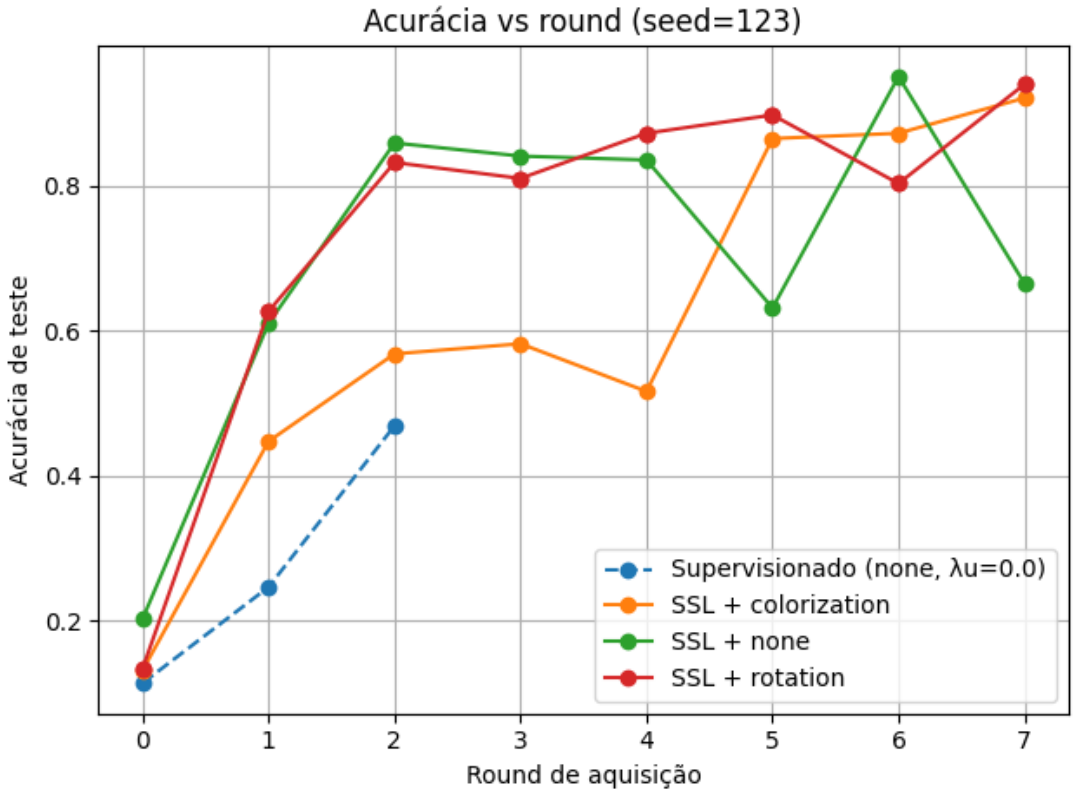

Figura 7: Acurácia vs. rodada (seed 123).

Principais conclusões: variantes com $\lambda_u = 1.0$ exibem crescimento consistente; Rotation é a mais estável; o SSL puro apresenta alta volatilidade entre seeds.

4.4. Trade-off: Acurácia Final vs. Tempo Total

Analisamos desempenho final e tempo total de execução.

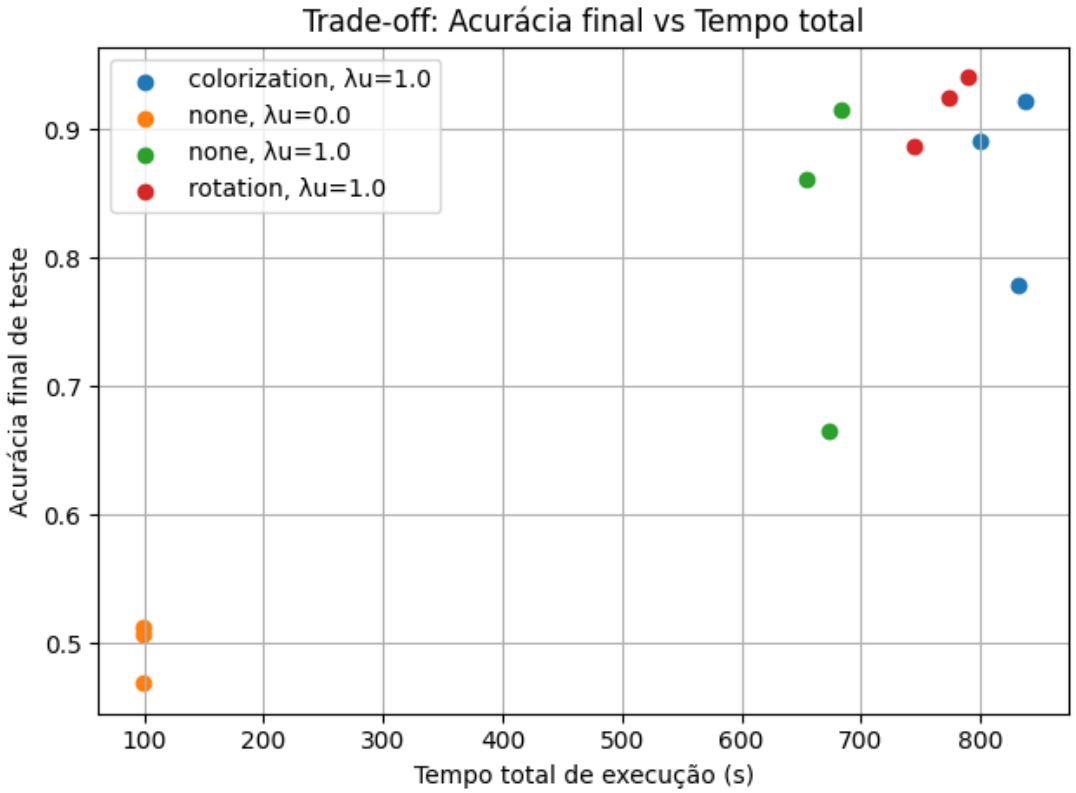

Figura 8: Trade-off entre acurácia final e tempo total de execução.

Observações: variantes SSL com $\lambda_u = 1.0$ exigem mais tempo, mas alcançam acurácia superior; Rotation apresenta melhor relação acurácia/tempo; SSL puro é mais rápido porém instável.

4.5. Conclusões Experimentais

Com base nos testes:

- SSL é essencial em regimes com poucos rótulos — ganho superior ao baseline supervisionado;
- Tarefas pretexto melhoram estabilidade e qualidade das representações;
- Rotation foi a melhor tarefa pretexto em nosso setup (maior acurácia média e menor variância);
- SSL puro é competitivo, porém inconsistente entre execuções.

5. Análise dos Resultados

MNIST. No MNIST, a colorização apresentou maior acurácia no *linear probe* (94,3%), seguida de jigsaw (68,2%) e rotação (55,3%). A perda MSE da colorização foi estável, indicando aprendizado efetivo das estruturas visuais; o custo adicional do decodificador foi compensado pela melhora de desempenho.

STL-10. No STL-10, mais complexo, os resultados foram inferiores: rotação atingiu 37,2% no melhor caso. A limitação da capacidade do encoder, a baixa resolução (48×48) e o reduzido número de épocas explicam parte do desempenho reduzido.

Eficiência Computacional. O tempo de treinamento foi moderado (250–300 s por execução em MNIST). A colorização demandou mais tempo, mas trouxe ganho proporcional;

jigsaw mostrou boa relação custo-benefício.

6. Observações

- **Colorização** gera gradientes mais informativos, acelerando o aprendizado mesmo em encoders rasos.
- **Rotação** é um sinal simples e robusto, útil quando a orientação espacial é relevante.
- **Jigsaw 2×2** captura menos estrutura semântica em imagens naturais; recomenda-se Jigsaw 3×3 para futuros testes.
- O desempenho no STL-10 sugere necessidade de mais amostras não rotuladas ou de arquitetura mais profunda.

7. Limitações

Este estudo tem limitações importantes: execução em CPU (limita exploração de arquiteturas maiores), número reduzido de épocas e resoluções baixas para STL-10. Além disso, os resultados podem variar com as augmentations escolhidas e a política de aquisição em AL.

8. Conclusão

Os experimentos mostram que a combinação de SSL com tarefas pretexto e aprendizagem ativa reduz de forma eficaz a necessidade de rótulos, mantendo desempenho competitivo. A colorização destacou-se no MNIST; a rotação demonstrou robustez e bom custo-benefício. Em domínios mais complexos (STL-10), recomenda-se ampliar dados, aumentar resolução e testar arquiteturas mais profundas.

9. Próximos Passos

- Ampliar experimentos ao split "train+unlabeled" do STL-10;
- Aumentar épocas de pré-texto (ex.: 20) mantendo probe curto (5);
- Implementar Jigsaw 3×3 com 30 permutações;
- Incluir augmentações adicionais (RandomResizedCrop, ColorJitter, GaussianBlur);
- Avaliar consumo energético (CPU/GPU) e custo por acurácia;
- Comparar com rede supervisionada equivalente para quantificar ganho de eficiência.

Referências

[1] Amitness. *FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence*, 2020. Disponível em: `https://amitness.com/posts/fixmatch`. Acesso em nov. 2025.

[2] Tsang, Sh. *Review: FixMatch – Simplifying Semi-Supervised Learning with Consistency and Confidence*, 2020. Disponível em: `https://sh-tsang.medium.com/review-fixmatch-simplifying-semi-supervised-` Acesso em: 23 nov. 2025.

[3] Marinho, F. V. *Active Semi-supervised Learning CNNs*. Disponível em: `https://github.com/FelippeVelosoMarinho/ActiveSemisupervisedLearningCNN-s`. Acesso em nov. 2025.

[4] Autor Desconhecido. *Survey on Semi-Supervised Learning*. Disponível em: https://www.molgen.mpg.de/3659531/MITPress--SemiSupervised-Learning.pdf. Acesso em nov. 2025.

[5] GeeksforGeeks. *Self-Supervised Learning (SSL) — An Overview*. Disponível em: /mnt/data/RP.pdf. Acesso em: nov. 2025.