

UFPI – CCN – DC

Arquitetura de Computadores 2025.1

Professor: Ivan Saraiva Silva

Grupo: Davi Soares de Macedo & Enrico da Rocha Santos Teixeira

Maio de 2025

Introdução

1. Objetivo

Comparar dois métodos de melhorar o contraste de uma imagem em tons de cinza: um em Python (alto nível) e outro em Assembly RISC-V (baixo nível).

2. Metodologia

- Preparação: Convertida a imagem colorida em QCIF (176×144) para tons de cinza, gerando *Imagem_PB.jpg*.
- **Python:**
 1. Carrega *Imagem_PB.jpg*.
 2. Equaliza a imagem utilizando o script *equalizacao.py*.
 3. Salva *Imagem_PB_equalizada_altonivel.jpg*.
 4. O mesmo script faz a contagem e gera um histograma do resultado no arquivo *histograma_altonivel_equalizada.txt*.
- **Assembly:**
 1. Abrimos via terminal 2 códigos risc-v diferentes com o arquivo *Imagem_PB.raw*, com os 25344 pixels, um vai equalizar e imprimir o histograma em um arquivo .txt (*equaliza_histo_riscvc.asm*) enquanto o outro(*equaliza_raw_riscv.asm*) vai gerar um binário (.raw) a partir da equalização do histograma, gerando os arquivos *Imagem_PB_equalizada_assembly.raw* e *histograma_equalizada_assembly.txt*.

3. Resultados

- A soma de todas as contagens bateu exatamente com o número de pixels (25 344) em ambas as versões, garantindo que nenhum dado se perdeu.
- Nos histogramas originais, Python e Assembly mostraram curvas quase idênticas.
- Após equalizar, ambos passaram a ter maior variedade de tons, evidenciando a eficácia do método.

- A diferença entre os histogramas equalizados (Python vs. Assembly) foi mínima, comprovando que os dois códigos produziram o mesmo efeito.

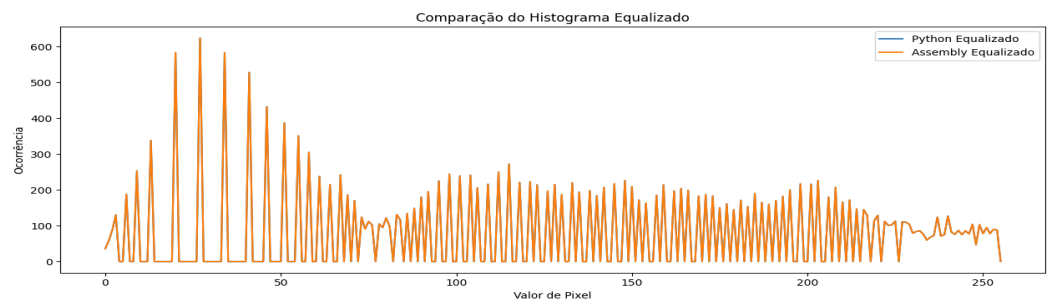
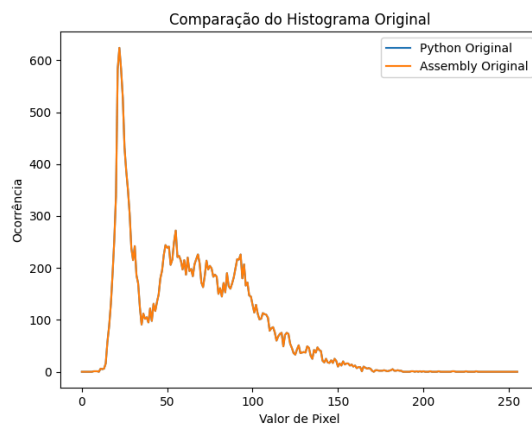


imagem original



imagem_PB convertida em python



imagem equalizada em python



imagem equalizada em risc-v

A proximidade dos dados é tanta que ficou praticamente impossível distinguir nesse gráfico a variação dos dados, apenas um pequeno contraste azul, comprovando a eficiência das duas implementações.

4. Conclusão

A versão em Python foi mais rápida de desenvolver, mas a de Assembly mostrou que é possível reproduzir o mesmo resultado em nível de máquina. Ambos os métodos entregaram imagens com contraste melhorado e histogramas consistentes, utilizamos scripts em python para comparar os resultados que afirmamos acima, provando que a implementação é p