

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO CEARÁ
CAMPUS FORTALEZA**

ENGENHARIA DA COMPUTAÇÃO

**JOSAFÁ DE ALENCAR SANTIAGO
JOYCE KELLE DA SILVA**

**RELATÓRIO RASTERIZAÇÃO DE RETAS E POLÍGONOS
COMPUTAÇÃO GRÁFICA**

FORTALEZA – CE

2021

1. Objetivo

A rasterização é o processo de conversão da representação vetorial para a matricial. O objetivo do trabalho é implementar o algoritmo de rasterização de um segmento de reta no R^2 e polígonos convexos, para uma imagem e observar os resultados obtidos em diferentes situações.

2. Desenvolvimento

Nesse trabalho abordaremos o algoritmo de rasterização de retas, que foi implementado em Python, utilizando a biblioteca “Numpy” e “Matplotlib”. Utilizamos o Algoritmo DDA (Digital Differential Analyser) que se baseia na aplicação direta das fórmulas que descrevem uma reta no plano.

O DDA é um típico algoritmo incremental. Algoritmos incrementais são aqueles em que uma das variáveis é obtida apenas incrementando o seu valor, por exemplo $X = X + 1$, e a outra é calculada por alguma regra a partir da primeira.

No algoritmo de rasterização, para construir uma reta definimos a função rasterização que tem como entrada os pontos da reta, a resolução desejada e matriz de zeros do tamanho da resolução.

Exemplo:

rasterizacao(ponto_1_x, ponto_1_y, ponto_2_x, ponto_2_y, resolucao_x, resolucao_y, matriz):

Ela modifica a matriz de entrada, na sequência plotamos o gráfico utilizando a função plot_rasterização que tem entrada os pontos a resolução e a matriz.

Exemplo:

plot_rasterizacao_reta(matriz, ponto_1_x, ponto_1_y, ponto_2_x, ponto_2_y, resolucao_x, resolucao_y):

Para os polígonos utilizamos uma função rasterizacao ,específica para cada polígono, para fazer todas as retas do polígono e na sequência pegamos os pixels do polígono através da função pixel_poligono que tem como entrada a matriz das retas que formam o polígono e a matrix onde vai ser colocado os pixels do polígono

Exemplo:

pixels_poligono(matriz, matriz_poligono):

Então plotamos o gráfico utilizando a função plot_rasterizacao_poligono que tem como entrada a matriz resolução e nome do polígono

Exemplo:

plot_rasterizacao_poligono(matriz,resolucao_x,resolucao_y,nome_poligono):

Para plotar o gráfico das retas foram obtidas imagens com 3 resoluções diferentes, 40x40, 320x280 e 580x420 para as seguintes situações:

- Reta entre os pontos (0.2,0.2) e (0.9,0.5), ou seja, $|\Delta x| > |\Delta y|$;
- Reta entre os pontos (0.2,0.2) e (0.6,0.9), ou seja, $|\Delta y| > |\Delta x|$;
- Reta vertical, entre os pontos (0.4,0.2) e (0.4,0.8);
- Reta horizontal, entre os pontos (0.2,0.4) e (0.8,0.4).

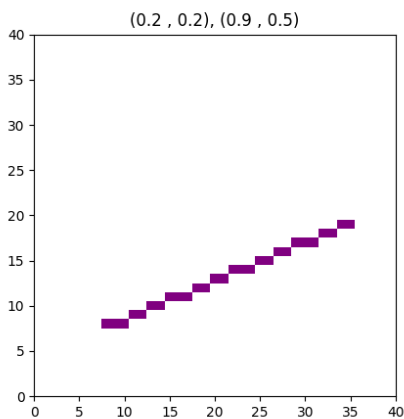
Para plotar o gráfico dos polígonos foram obtidas imagens com 3 resoluções diferentes, 50x50, 300x300 e 600x600 para os seguintes pontos:

- 1° Triângulo equilátero : **(0.1, 0.1), (0.7, 0.1), (0.4, 0.6)**
- 2° Triângulo equilátero: **(0.2, 0.2), (0.8, 0.2), (0.5, 0.7)**
- 1° Quadrado: **(0.1, 0.1), (0.1, 0.3), (0.3, 0.3), (0.3, 0.1)**
- 2° Quadrado: **(0.2, 0.2), (0.2, 0.8), (0.8, 0.8), (0.8, 0.2)**
- 1° Hexágono: **(0.2, 0.6), (0.4, 0.6), (0.5, 0.4), (0.4, 0.2), (0.2, 0.2), (0.1, 0.4)**
- 2° Hexágono: **(0.1, 0.2), (0.2, 0.1), (0.3, 0.1), (0.4, 0.2), (0.3, 0.3), (0.2, 0.3)**

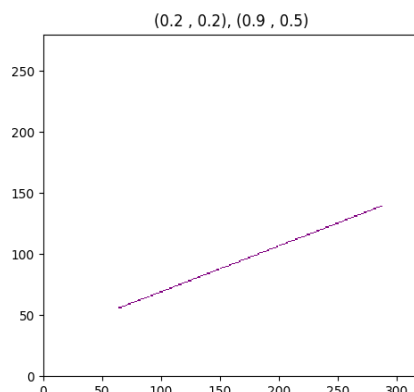
3. Resultados

- **Reta $|\Delta x| > |\Delta y|$**

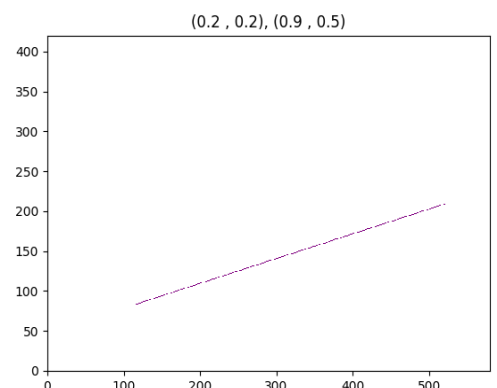
As imagens foram obtidas para a reta que passa pelos pontos (0.2,0.2) e (0.9,0.5). Na figura abaixo estão representadas as imagens rasterizadas para (a) uma matriz 40x40, (b) uma matriz 320x280 e (c) uma matriz 580x420.



(A)



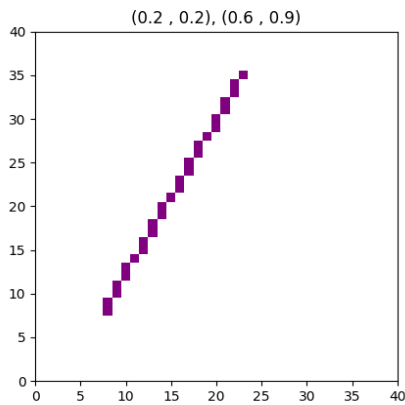
(B)



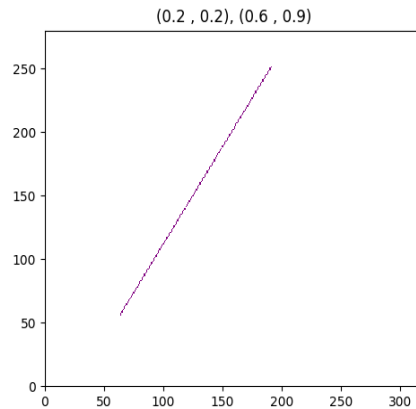
(C)

- **Reta $|\Delta y| > |\Delta x|$**

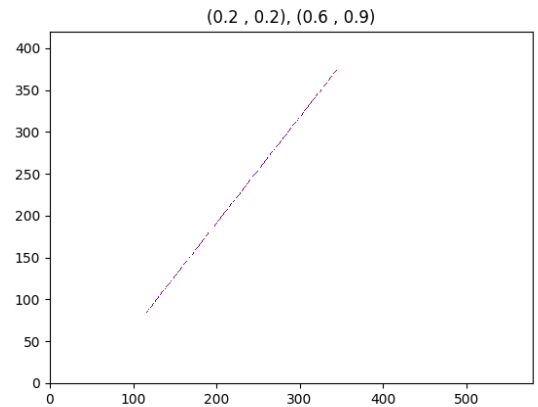
As imagens foram obtidas para a reta que passa pelos pontos (0.2,0.2) e (0.6,0.9). Na figura abaixo estão representadas as imagens rasterizadas para (a) uma matriz 40x40, (b) uma matriz 320x280 e (c) uma matriz 580x420.



(A)



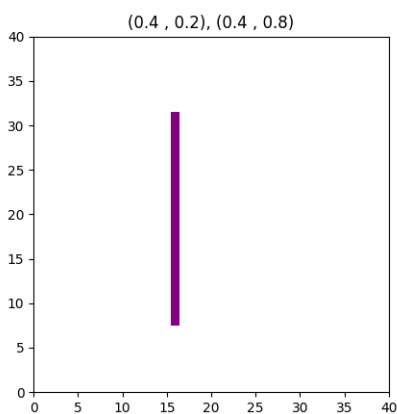
(B)



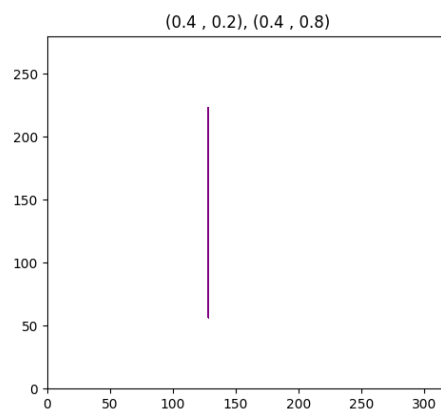
(C)

- **Reta vertical**

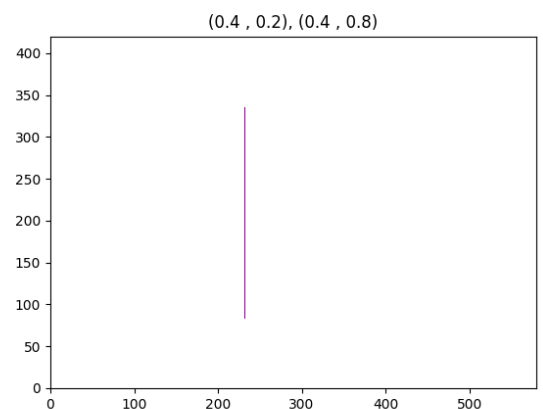
As imagens foram obtidas para a reta que passa pelos pontos (0.4,0.2) e (0.4,0.8). Na figura abaixo estão representadas as imagens rasterizadas para (a) uma matriz 40x40, (b) uma matriz 320x280 e (c) uma matriz 580x420.



(A)



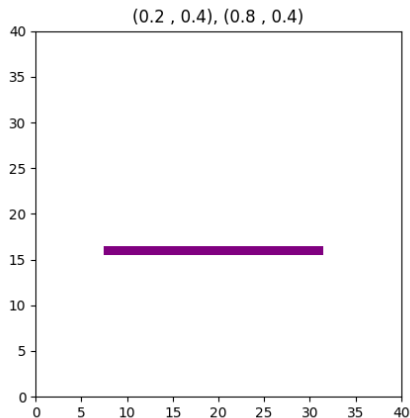
(B)



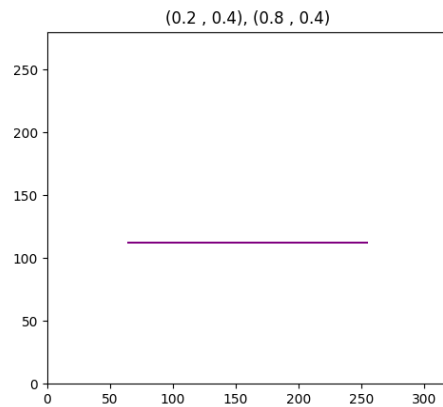
(C)

- **Reta horizontal**

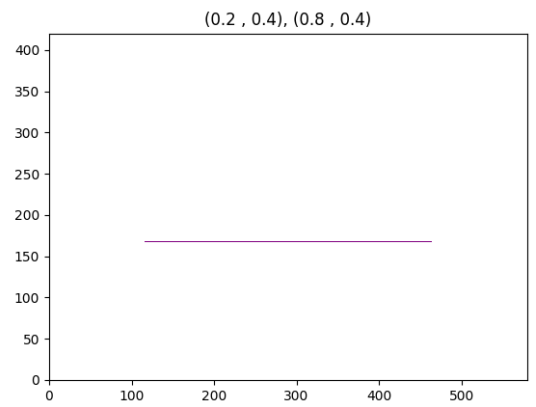
As imagens foram obtidas para a reta que passa pelos pontos (0.2,0.4) e (0.8,0.4). Na figura abaixo estão representadas as imagens rasterizadas para (a) uma matriz 40x40, (b) uma matriz 320x280 e (c) uma matriz 580x420.



(A)



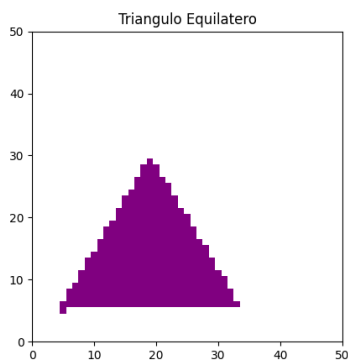
(B)



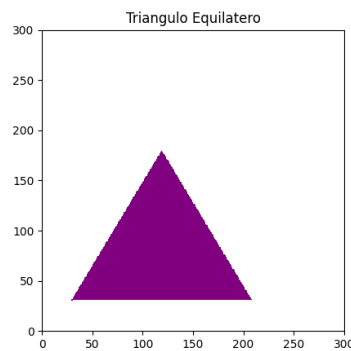
(C)

- **1° Triângulo Equilátero**

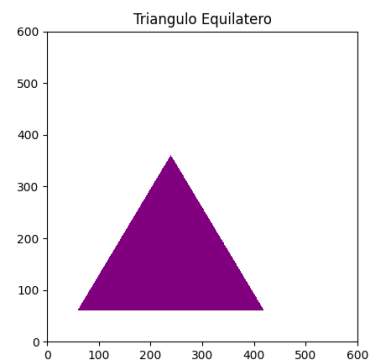
As imagens foram obtidas com resoluções diferentes 50x50, 300x300 e 600x600 a partir dos pontos (0.1, 0.1), (0.7, 0.1), (0.4, 0.6)



50x50



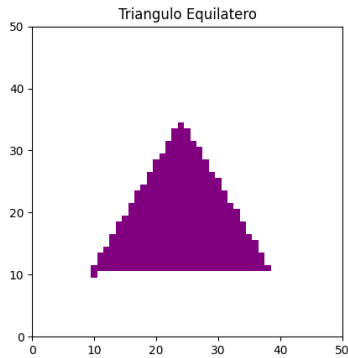
300x300



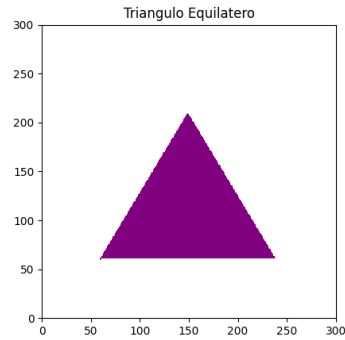
600x600

- **2° Triângulo Equilátero**

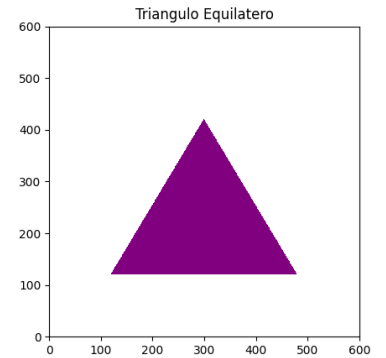
As imagens foram obtidas com resoluções diferentes 50x50, 300x300 e 600x600 a partir dos pontos **(0.2, 0.2), (0.8, 0.2), (0.5, 0.7)**



50x50



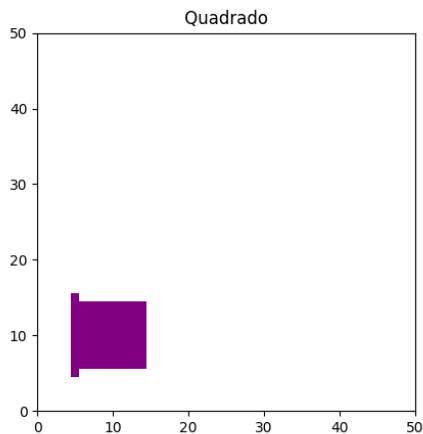
300x300



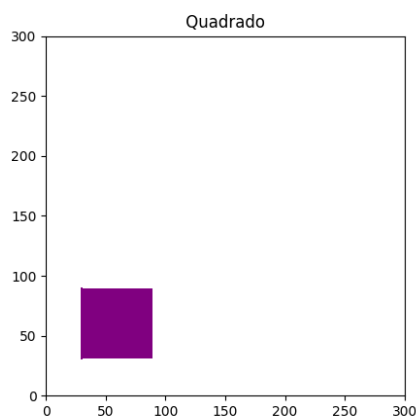
600x600

- **1° Quadrado**

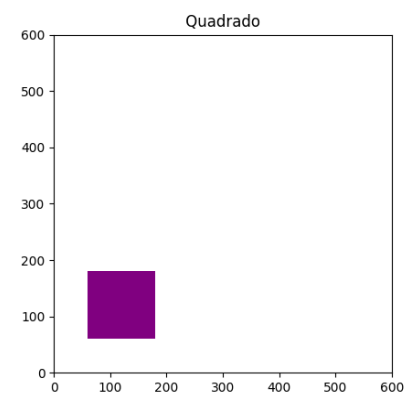
As imagens foram obtidas com resoluções diferentes 50x50, 300x300 e 600x600 a partir dos pontos **(0.1, 0.1), (0.1, 0.3), (0.3, 0.3), (0.3, 0.1)**



50x50



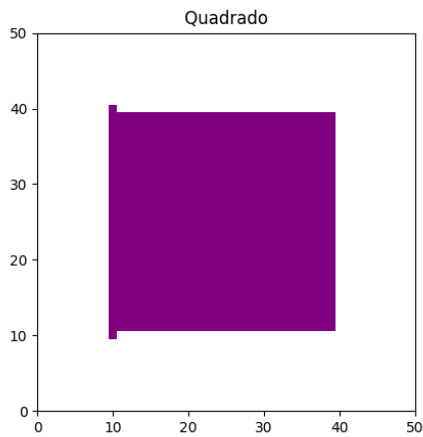
300x300



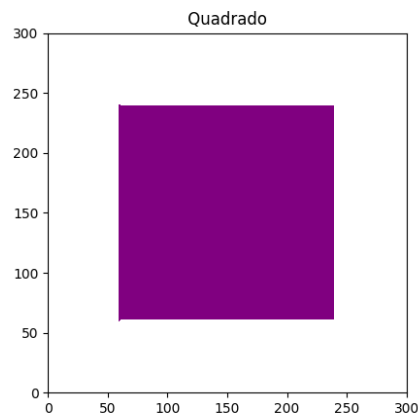
600x600

- **2° Quadrado**

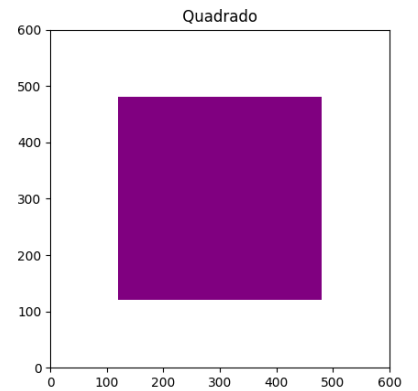
As imagens foram obtidas com resoluções diferentes 50x50, 300x300 e 600x600 a partir dos pontos **(0.2, 0.2), (0.2, 0.8), (0.8, 0.8), (0.8, 0.2)**



50x50



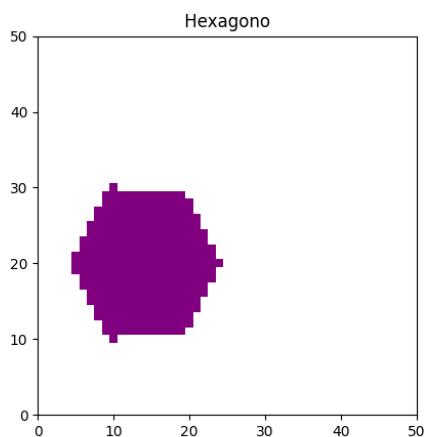
300x300



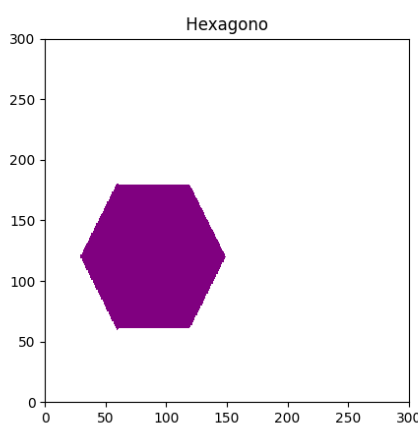
600x600

- **1° Hexágono**

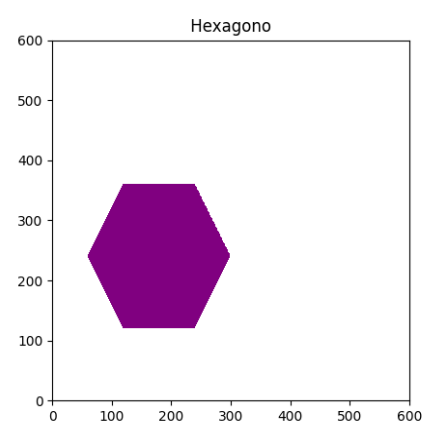
As imagens foram obtidas com resoluções diferentes 50x50, 300x300 e 600x600 a partir dos pontos **(0.2, 0.6), (0.4, 0.6), (0.5, 0.4), (0.4, 0.2), (0.2, 0.2), (0.1, 0.4)**



50x50



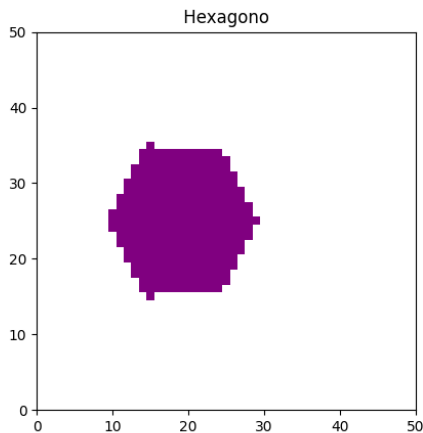
300x300



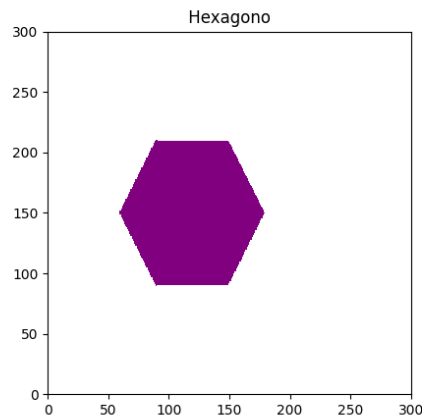
600x600

- **2º Hexágono**

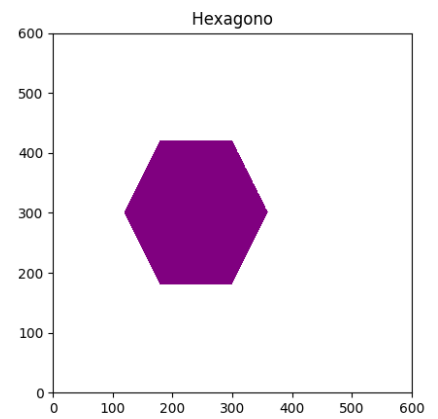
As imagens foram obtidas com resoluções diferentes 50x50, 300x300 e 600x600 a partir dos pontos (0.1, 0.2), (0.2, 0.1), (0.3, 0.1), (0.4, 0.2), (0.3, 0.3), (0.2, 0.3)



50x50



300x300



600x600

4. Conclusões

Podemos observar que para as duas primeiras retas ao aumentar as resoluções, a percepção visual de uma reta aumenta. Com isso constatamos que qualidade da representação é diretamente proporcional à resolução utilizada.

Para as retas horizontal e vertical, o aumento da resolução não influenciou na qualidade e percepção, como já era esperado.

Para os polígonos tivemos os mesmos resultados, que é possível observar que com o aumento da resolução a percepção de retas nos polígonos também aumenta. Com exceção do quadrado que utiliza somente retas verticais e horizontais.