Análise e Síntese de Algoritmos

Relatório do 2º Projeto – Grupo 54

I. Introdução

Pretende-se desenvolver um programa que permita, dada uma imagem representada por um retângulo de pixéis (matriz m x n em que cada posição (i, j) com $0 \le i < m$ e $0 \le j < n$ representa um pixel), classificar cada pixel como sendo de primeiro plano ou de cenário, de forma a minimizar o peso total. É importante referir que cada pixel tem um valor de peso referente ao primeiro plano, um valor de peso referente ao cenário, bem como um peso para cada arco que liga um pixel aos seus adjacentes.

II. Descrição da solução

Para a implementação da solução representa-se o problema como uma rede de fluxos $Gf = (V,E)$ em que cada $v \in V$ representa um pixel da imagem e cada par de arcos $\{(u,v),(v,u)\}$, $u,v \in V$ uma ligação entre pixéis, cujas capacidade são iguais e corresponde ao peso da ligação entre os dois pixéis representados por u e v. Gf tem a particularidade de, estando a "source" representada por $s \in V$ e a "sink" por $t \in V$, existirem os arcos $\{(s,v),(v,t)\}$, $s,t,v \in V$, para todo o $v \in V$, correspondendo a capacidade do arco (s,v) ao peso referente ao primeiro plano e a capacidade do arco (v,t) ao peso referente ao cenário do pixel representado por v.

A solução do problema consiste em encontrar o fluxo máximo e o corte mínimo de Gf, representando o fluxo máximo o menor peso total possível e o corte mínimo a respetiva segmentação de pixéis, na qual os pixéis que correspondem a vértices pertencentes ao mesmo subconjunto de corte mínimo do vértice s são classificados como sendo de cenário, e os pixéis que correspondem a vértices pertencentes ao mesmo subconjunto do vértice t são classificados como sendo de primeiro plano.

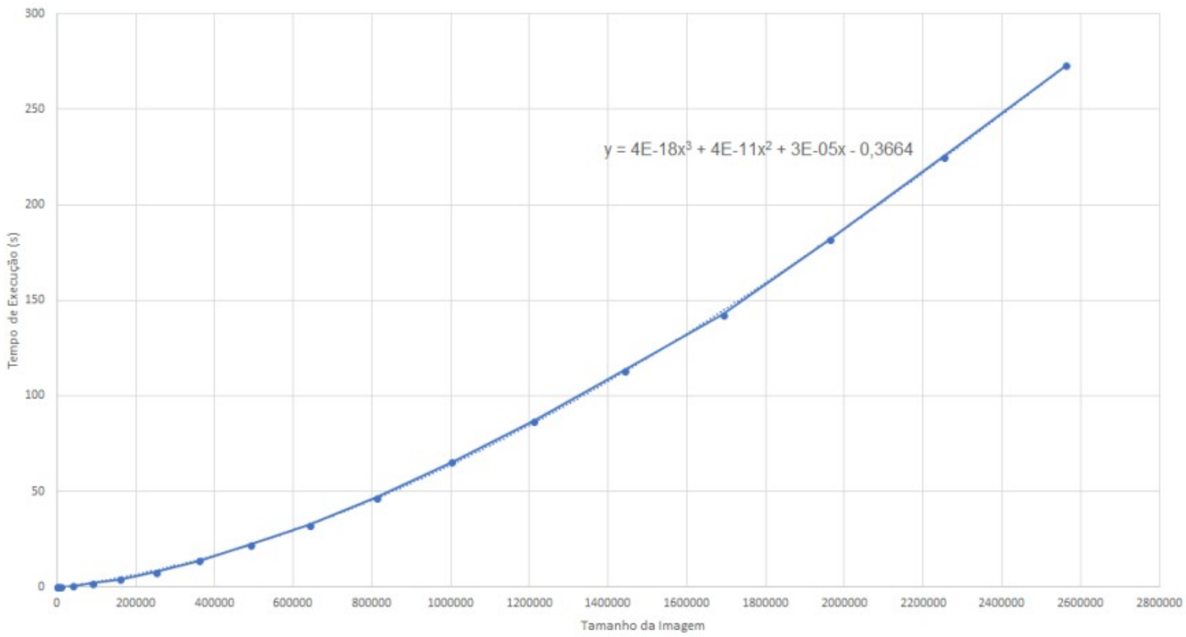
III. Análise Teórica

A representação de Gf é composta por uma matriz, com V linhas e 4 colunas, que contém a informação relativa à capacidade dos arcos adjacentes, nas 4 direções (cima, baixo, esquerda e direita), de cada vértice do grafo (excluindo os vértices s e t). A não existência de um arco é representada por um arco com capacidade -1, situação esta que acontece nos vértices que representam pixéis das extremidades da imagem. Em adição á matriz, existem também dois vectores, com tamanho V cada, que representam as ligações de cada vértice ao vértice s e ao vértice t. Concluímos assim que a representação utilizada para Gf tem complexidade espacial $O(4V + 2V) = O(V)$.

Para o cálculo do fluxo máximo e do respetivo corte mínimo de Gf, é utilizado o algoritmo de Edmonds-Karp. Este algoritmo é uma implementação do método de Ford-Fulkerson e consiste na utilização de várias BFS (Breadth-First Search) para a sucessiva busca de caminhos mais curtos entre o vértice s e o vértice t, e a inserção do máximo fluxo suportado pelo caminho encontrado nos arcos que o integram. Sendo que existem $O(VE)$ caminhos de aumento possíveis em qualquer rede de fluxos, e que, para cada um é preciso aplicar uma BFS, de complexidade $O(V+E)$, para o encontrar, podemos concluir que o algoritmo utilizado tem complexidade $O(VE(V+E))$. No entanto, para a rede de fluxos em questão, podemos concluir que cada vértice tem, no máximo, 6 arcos, levando assim a que, nesta rede de fluxos em específico, $O(E) = O(6V) = O(V)$. Podemos então finalmente concluir que a aplicação do algoritmo de Edmonds-Karp em Gf tem complexidade temporal $O(V^3)$.

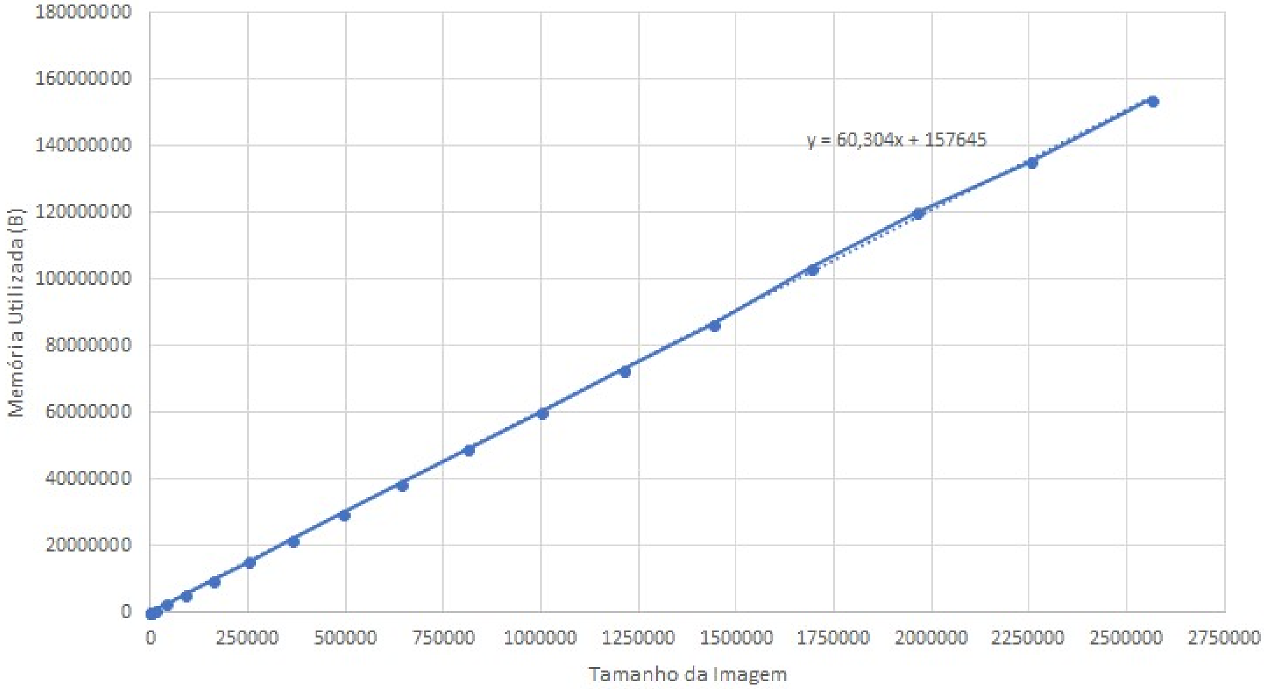
IV. Análise dos Resultados Experimentais

O gráfico ilustrado em seguida, representa os resultados experimentais obtidos após a execução do programa com inputs de tamanhos variados, que relaciona o tamanho do input com o tempo de execução, numa máquina não isolada e com outros processos em execução, resultando na não garantia da sua estabilidade, e por consequência, não conferindo uma precisão perfeita aos resultados obtidos.



Através da sua observação, verificamos que este suporta a hipótese teórica proposta na Análise Teórica, pois a equação aproximada da função por si representada consiste num polinómio de 3º grau, o que indica que o tempo de execução escala cubicamente com o tamanho do input.

Do mesmo modo, o próximo gráfico ilustrado, representa os resultados experimentais obtidos após a execução do programa com inputs de tamanhos variados, que relaciona o tamanho do input com a quantidade de memória utilizada.



Através da sua observação, verificamos que este suporta a hipótese teórica proposta na Análise Teórica, pois a equação aproximada da função por si representada consiste num polinómio de 1º grau, o que indica que a quantidade de memória utilizada escala linearmente com o tamanho do input.