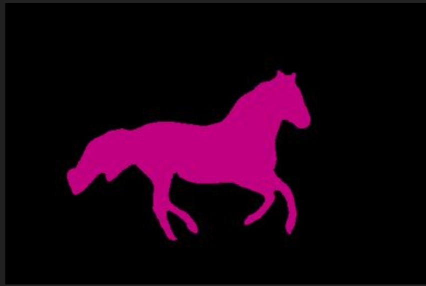
Image Foreground/Background Segmentation using Max-Flow

Raj Vipani
Chathura Abeyrathne
Sofia Briquet

Problem: Segment objects in Images

Detecting the foreground in an image is one of the biggest tasks in computer vision and image processing, commonly used to detect change in image sequences.

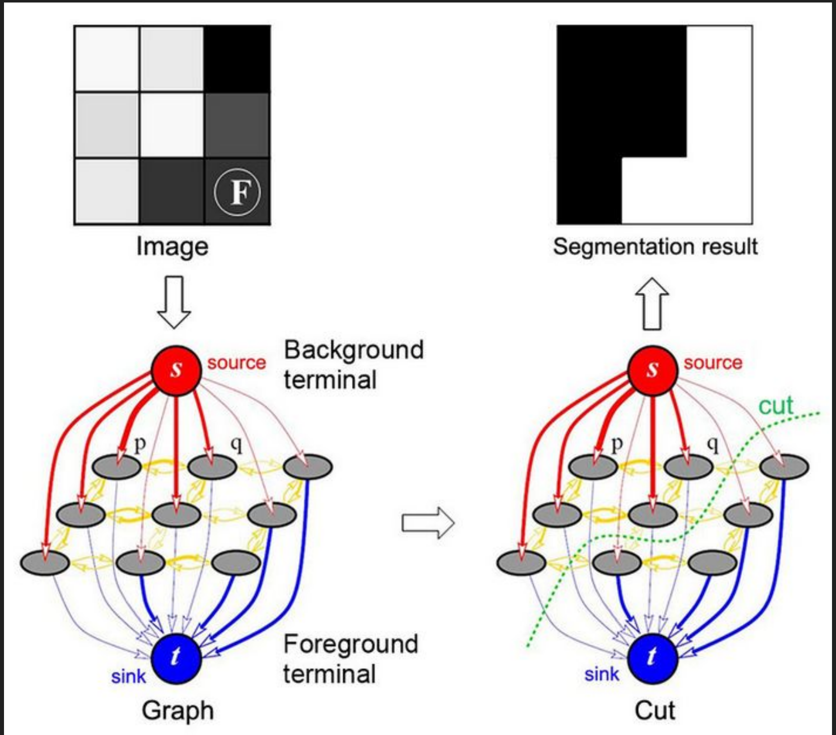
Foreground and background segmentation allows for the image's foreground to be extracted for further analysis (e.g. object recognition)



Approach: based on Min-Cut Max-Flow Theorem

Use graph cuts to segment the foreground and background of an image.

1. Built flow graph from input image
2. Apply max-flow algorithm to graph to find minimum cut which results in the optimal image segmentation



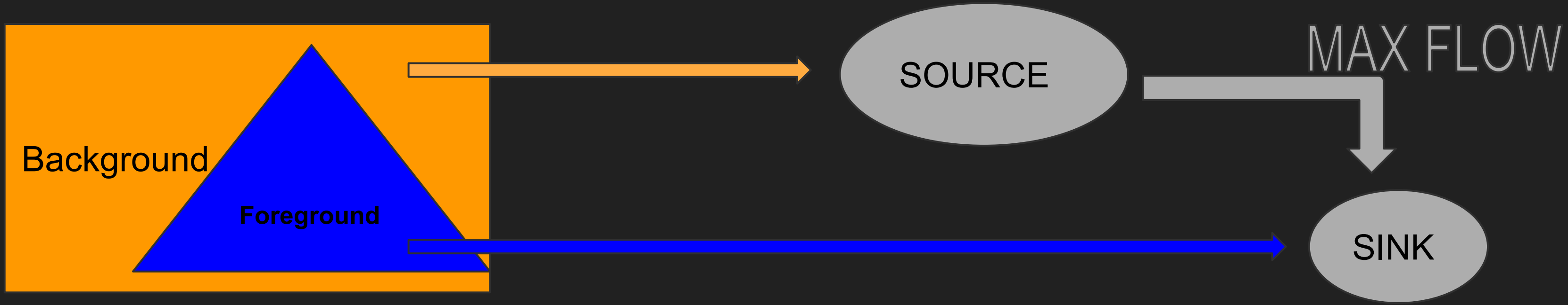
Transform image into Graph

Input: Grayscale image → **Output:** network flow graph

Each pixel in the image is considered as a node and the arc weights are calculated using a function of pixel similarity.

Two more vertices added: one is **SOURCE** and the other is the **SINK**

<u>**USER INPUT:**</u> One **background** point & one **foreground** point (seeds)



Edge Weight Calculation

From the paper “Graph cuts and efficient N-D image segmentation” by Boykov and Funka-Lea, we found that for two pixels p and q, the weight for the edge connecting the two points in the flow graph can be calculated as:

$$B(I_p, I_q) = 100 \cdot \exp\left(\frac{-(I_p - I_q)^2}{2\sigma^2}\right)$$

Here, I_p and I_q are the intensities of pixels p and q respectively and σ from empirical results was chosen to be 30.

Max-Flow Algorithms

1. **Edmonds-Karp**

$O(VE^2)$

Path from s to t in residual network:

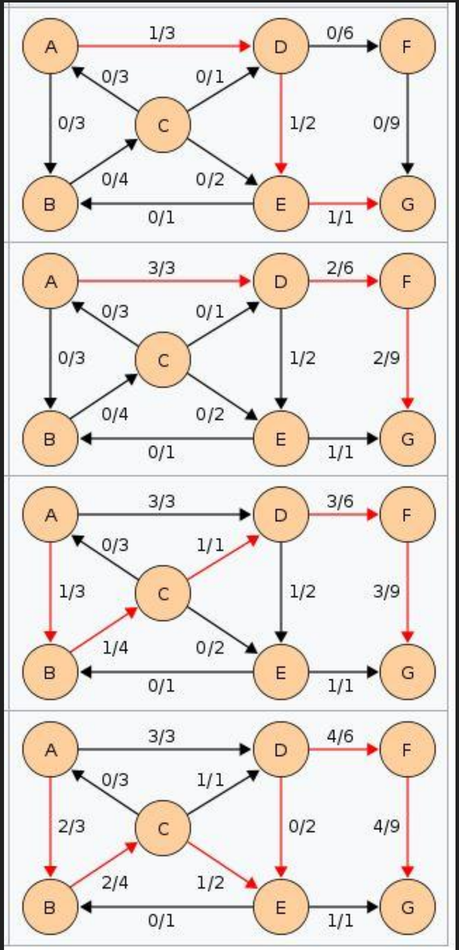
Flow = minimum of residual capacity of every arc in the path

Residual capacity (u,v) -= flow

Residual Capacity (v,u) += flow

When no other augmenting path can be found → Current flow = MAX FLOW

↳ (Breadth-First Search)


A
B
C
D
E
F
G
1/3
0/6
0/3
0/1
0/3
1/2
0/9
0/4
0/2
0/1
1/1
3/3
2/6
0/3
0/1
0/3
1/2
2/9
0/4
0/2
0/1
1/1
3/3
3/6
0/3
1/1
1/3
1/2
3/9
1/4
0/2
0/1
1/1
3/3
4/6
0/3
1/1
2/3
0/2
4/9
2/4
1/2
0/1
1/1

Max-Flow Algorithms

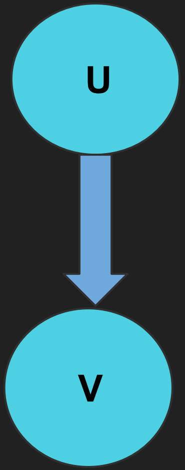
2. Push Relabel

$O(V^2E)$

Preflow: flow through network, doesn't need to satisfy flow conservation

Overflowing vertex: difference between flow entering and flow exiting the node

Initialize height of s = # of nodes, all other heights to zero



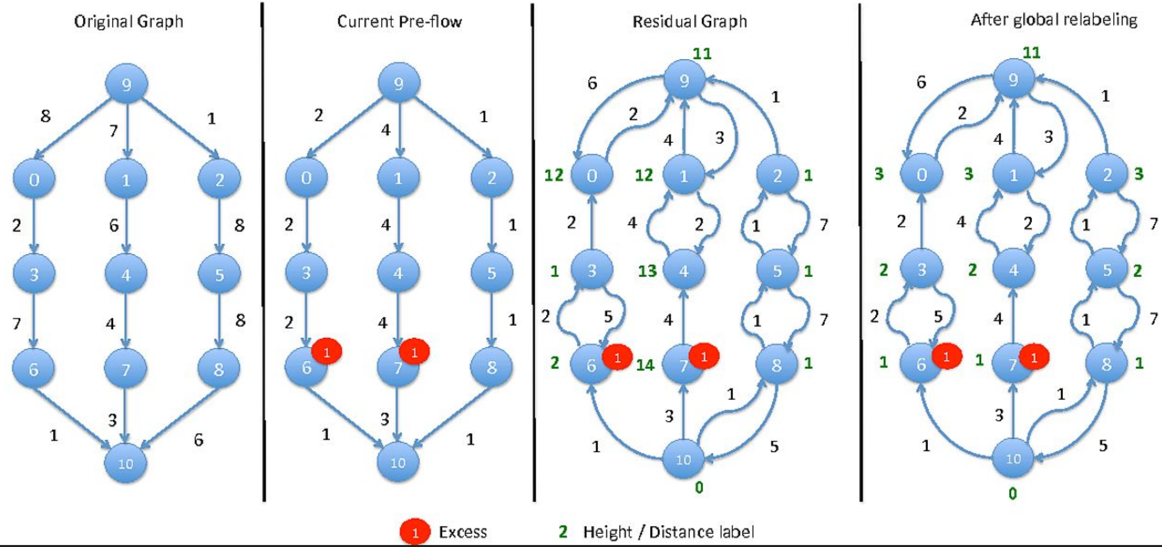
PUSH

Push = min(residual capacity of the arc, excess flow)
U must be higher than V

OR

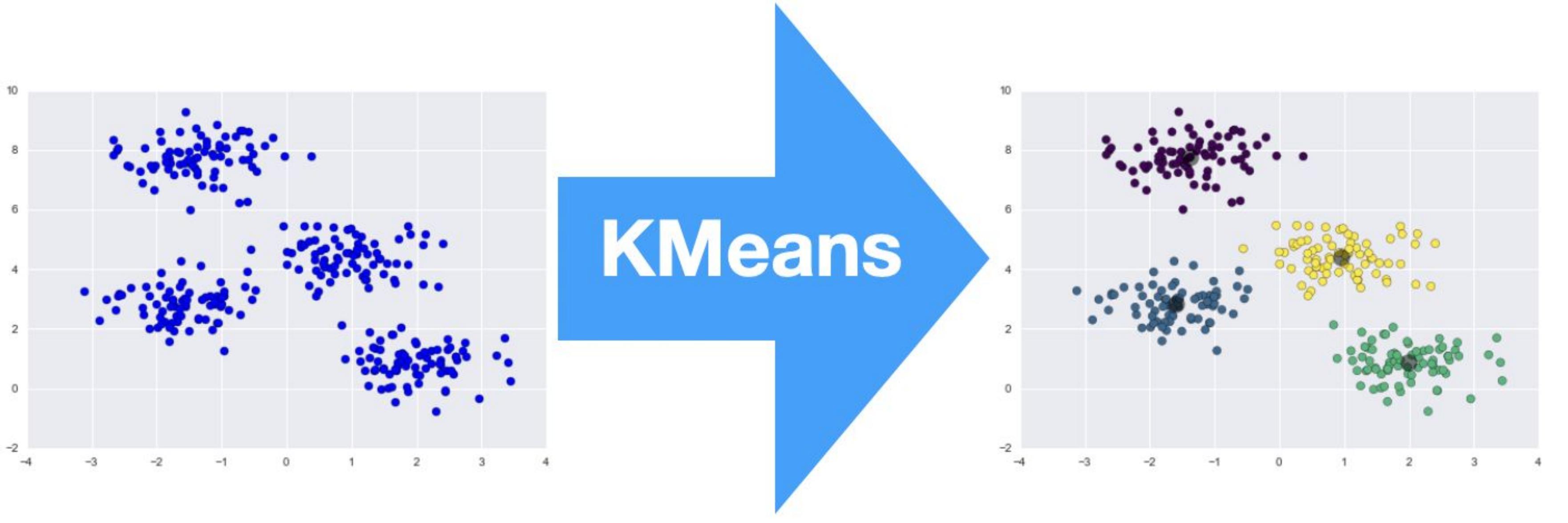
RELABEL

Set h(u) = 1 + min(h(u),h(v))


Original Graph
Current Pre-flow
Residual Graph
After global relabeling
1 Excess
2 Height / Distance label

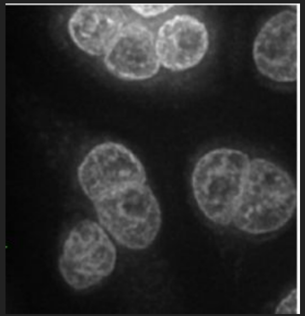
Comparison to Clustering Algorithms

1. K- Means : O(V^2)



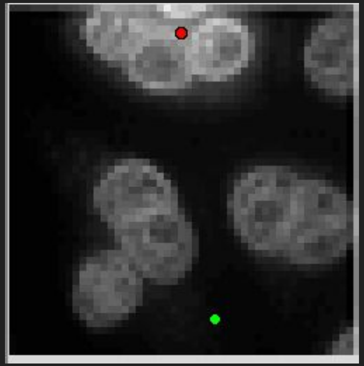
GUI Design

User selects image



Display of chosen image

User prompt to select 1 foreground and 1 background point



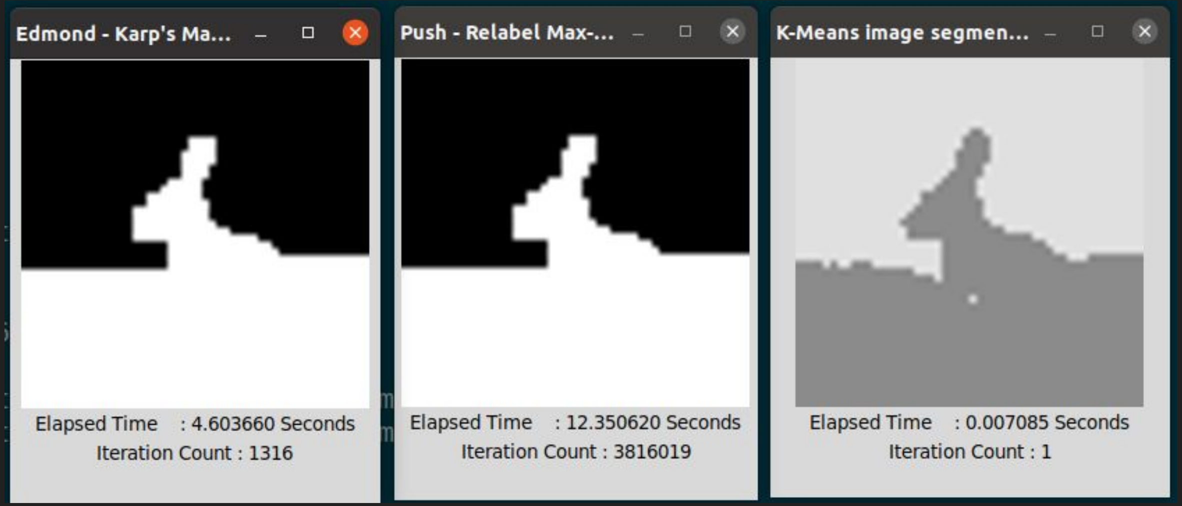
Max- Flow & Clustering methods called

Display of output segmentation by the different algorithms

Results

INPUT:

OUTPUT:



DEMO

QUESTIONS?