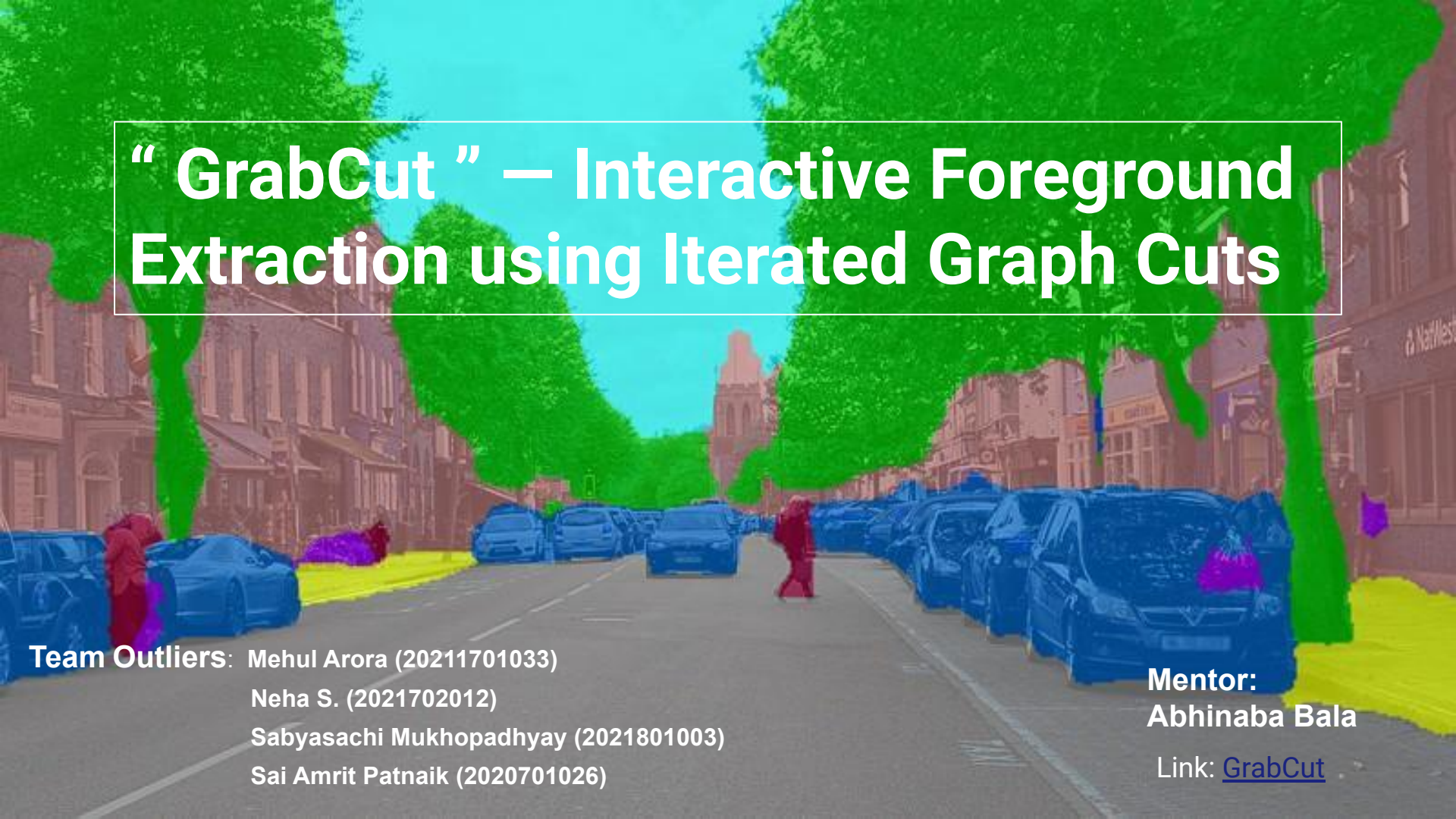
“ GrabCut ” — Interactive Foreground Extraction using Iterated Graph Cuts



Team Outliers: Mehul Arora (20211701033)
Neha S. (2021702012)
Sabyasachi Mukhopadhyay (2021801003)
Sai Amrit Patnaik (2020701026)

Mentor:
Abhinaba Bala

Link: GrabCut

1. Graph Cut
2. Grab Cut
3. Unary potentials
4. Pairwise Potentials
5. Grabcut Algorithm
6. Results
7. Failure analysis
8. References

Project Details

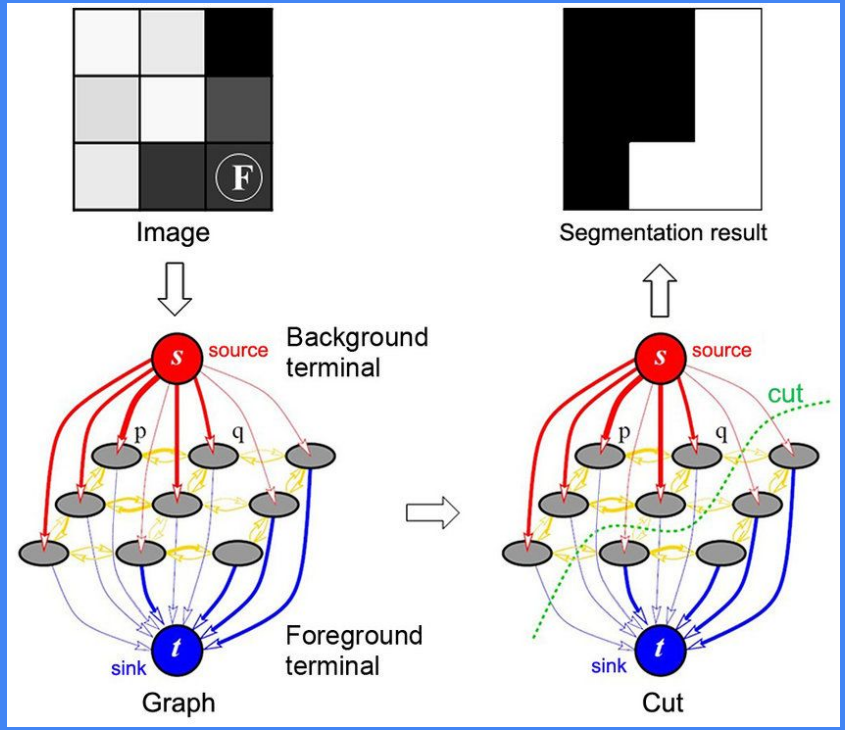
Graph Cut

- The framework of Graph cut consists of 2 parts: First, a network of flow graphs is built based on the input image. After that a max-flow algorithm is run on the graph in order to find the min-cut, which produces the optimal segmentation.
- The problem can be transferred to energy minimization. Its minimum should correspond to a good segmentation, as it is guided by both observed foreground and background grey-level histograms.
- The cost function to optimize is:

$$\arg\min_{w_{1\dots N}} \sum_{n=1}^{N} U_n(w_n) + \sum_{(m,n)\in\mathcal{C}} P_{mn}(w_m, w_n),$$

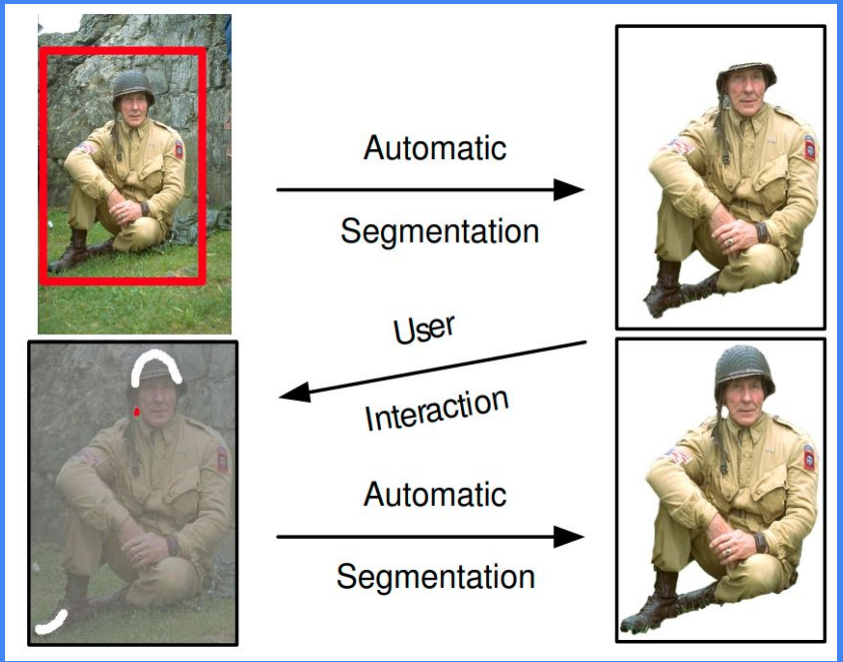
Unary terms (compatability of data with label y)

Pairwise terms (compatability of neighboring labels)



GrabCut

- Grabcut is a more powerful, iterative version of the optimization of Graph-cut. This substantially simplifies the user interaction needed for a given quality of result.
- In Grabcut, the monochrome image model is replaced for colour by a Gaussian Mixture Model (GMM) in place of histograms
- The one-shot minimum cut estimation algorithm is replaced by an iterative procedure that alternates between estimation and parameter learning and the demands on the interactive user are relaxed by allowing incomplete labelling.



Unary Potentials

- The Unary potentials are learnt using GMMs based on color intensities.
- Learnt as likelihood of the points belonging to Kth component of fg/ bg GMM.

$$U(\alpha, k, \theta, z) = \sum_n D(\alpha_n, k_n, \theta, z_n)$$

$$D(\alpha_n, k_n, \theta, z_n) = -\log \pi(\alpha_n, k_n) + \frac{1}{2} \log \det \Sigma(\alpha_n, k_n) + \frac{1}{2} [z_n - \mu(\alpha_n, k_n)]^T \cdot \Sigma(\alpha_n, k_n)^{-1} [z_n - \mu(\alpha_n, k_n)]$$

Pairwise Potentials

- The pairwise potentials are computed as a pixel smoothening computation.
- It has pixel wise intensity difference terms, weighted by a beta parameter which controls amount of smoothening to offer.
- The gamma parameter controls the overall weight given to this computation

$$\beta = \left(2\left\langle (z_m - z_n)^2 \right\rangle\right)^{-1}$$

$$V(\alpha, z) = \gamma \sum_{m,n \in C} [\alpha_n \neq \alpha_m] \cdot e^{-\beta \|z_m - z_n\|^2}$$

GrabCut Algorithm

Initialisation

- User initialises trimap T by supplying only T_B. The foreground is set to $T_F = \emptyset$; $T_U = \overline{T}_B$, complement of the background.
- Initialise $\alpha_n = 0$ for $n \in T_B$ and $\alpha_n = 1$ for $n \in T_U$.
- Background and foreground GMMs initialised from sets $\alpha_n = 0$ and $\alpha_n = 1$ respectively.

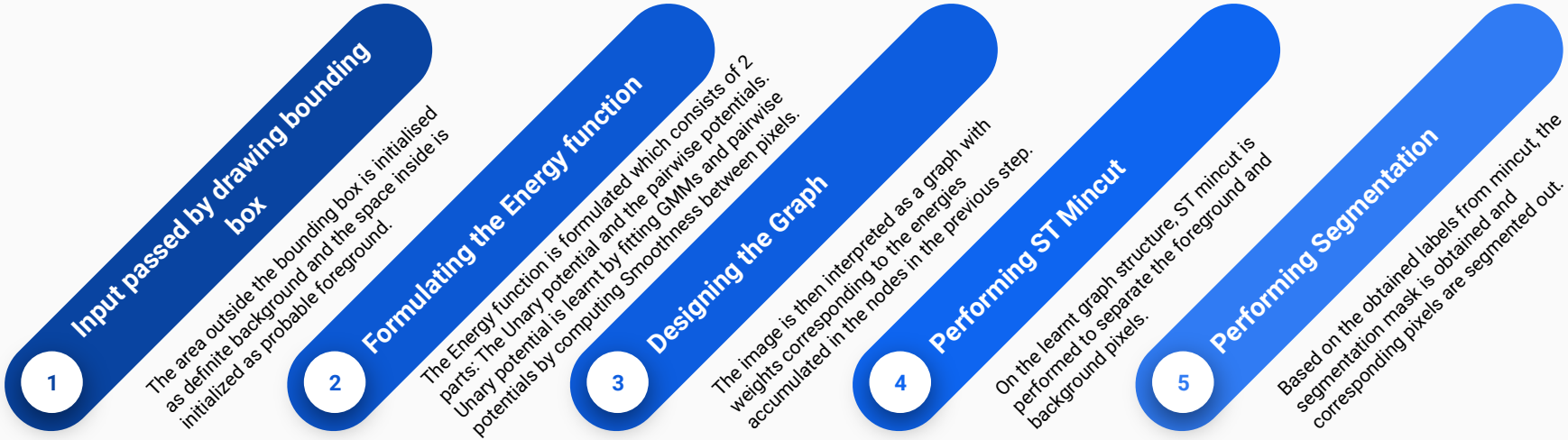
Iterative minimisation

1. *Assign GMM components to pixels:* for each n in T_U,
$$k_n := \arg\min_{k_n} D_n(\alpha_n, k_n, \theta, z_n).$$
2. *Learn GMM parameters* from data $\mathbf{z}$:
$$\underline{\theta} := \arg\min_{\underline{\theta}} U(\underline{\alpha}, \mathbf{k}, \underline{\theta}, \mathbf{z})$$
3. *Estimate segmentation:* use min cut to solve:
$$\min_{\{\alpha_n:\, n \in T_U\}} \min_{\mathbf{k}} \mathbf{E}(\underline{\alpha}, \mathbf{k}, \underline{\theta}, \mathbf{z}).$$
4. Repeat from step 1, until convergence.
5. Apply border matting (section 4).

User editing

- *Edit:* fix some pixels either to $\alpha_n = 0$ (background brush) or $\alpha_n = 1$ (foreground brush); update trimap T accordingly. Perform step 3 above, just once.
- *Refine operation:* [optional] perform entire iterative minimisation algorithm.

Flow Chart


1
Input passed by drawing bounding box
The area outside the bounding box is initialised as definite background and the space inside is initialized as probable foreground.
2
Formulating the Energy function
The Energy function is formulated which consists of 2 parts: The Unary potential and the pairwise potentials. Unary potential is learnt by fitting GMMs and pairwise potentials by computing Smoothness between pixels.
3
Designing the Graph
The image is then interpreted as a graph with weights corresponding to the energies accumulated in the nodes in the previous step.
4
Performing ST Mincut
On the learnt graph structure, ST mincut is performed to separate the foreground and background pixels.
5
Performing Segmentation
Based on the obtained labels from mincut, the segmentation mask is obtained and corresponding pixels are segmented out.

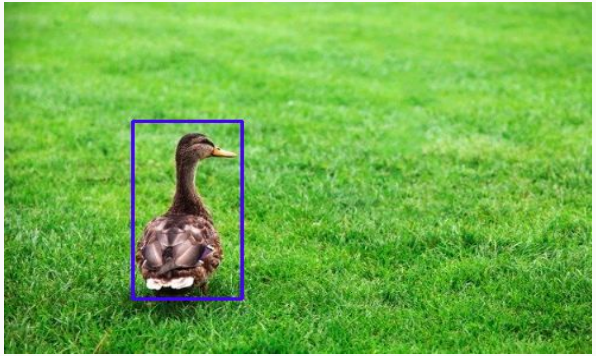
Deliverables

- A fully working implementation of the iterative GrabCut Algorithm implemented from scratch.
- Implementation for the components of the energy function and optimization procedure.
- An interactive user interface that lets the user provide initial input by drawing bounding box.
- Generate and display the segmented output.
- Implement the interactive user editing feature to allow the user to improve segmentation.
- A detailed ablation study on various parameters of the algorithm and their effect on the output.
- Release the code open sourced and Presentation and description of the project along with well documented README instructions on run the demo.

Results



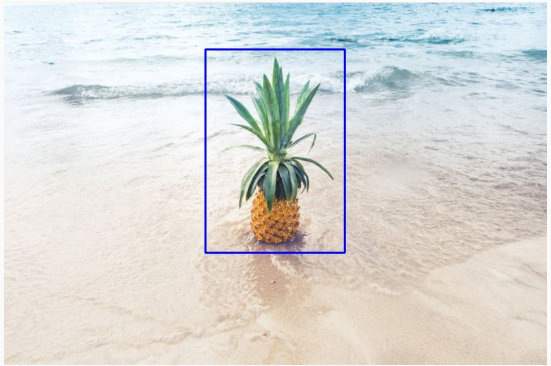





Results



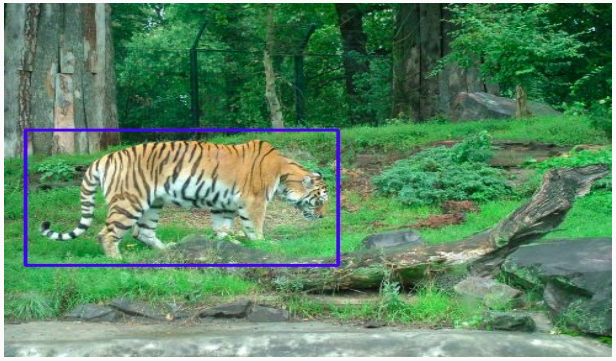




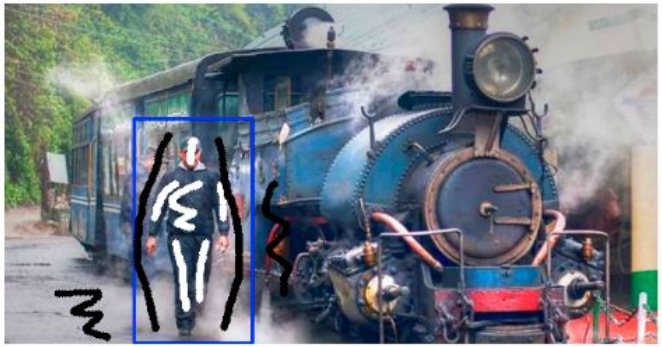

Failed Results



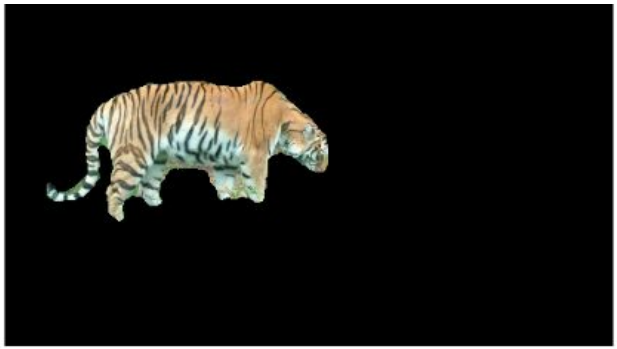





Improving Failed Results by Interaction

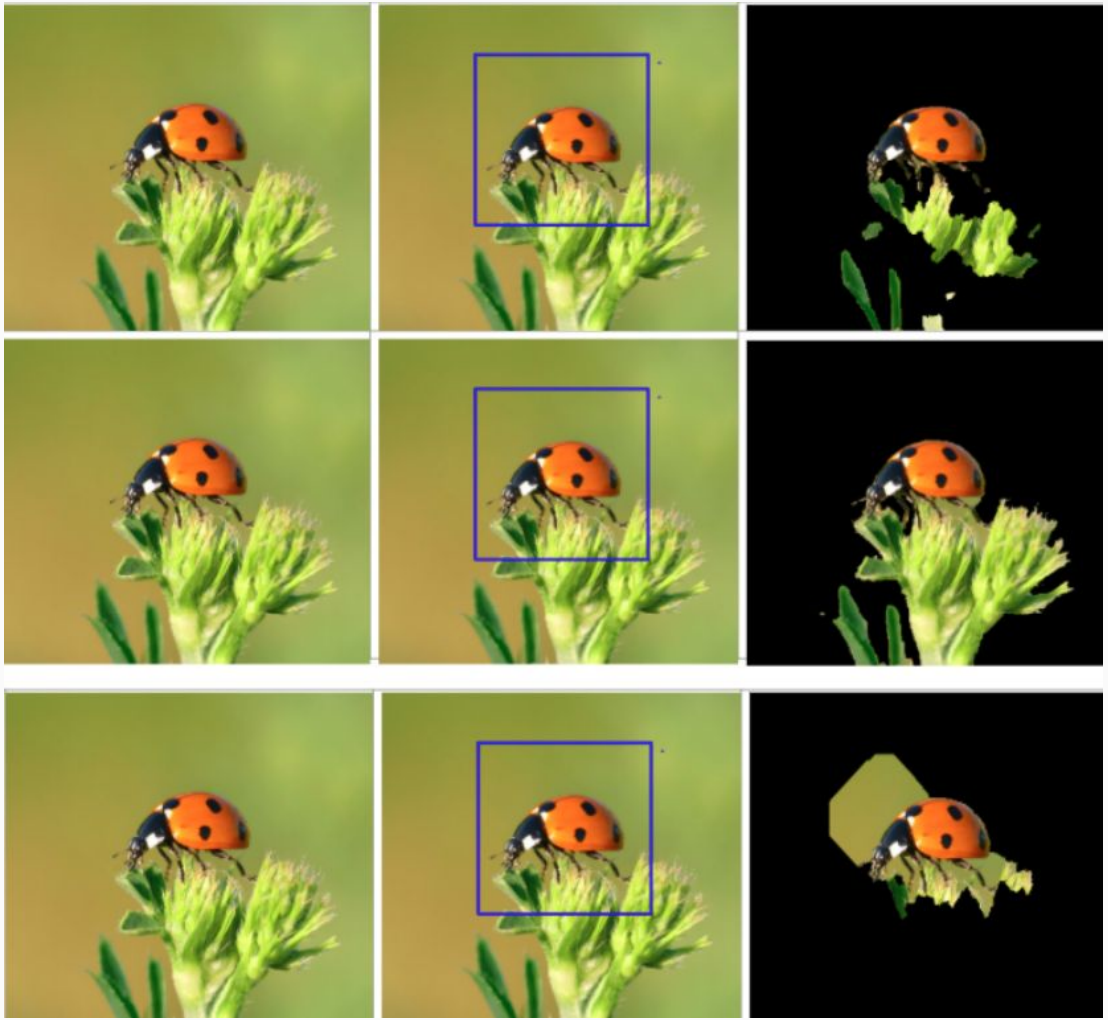




Ablation Studies

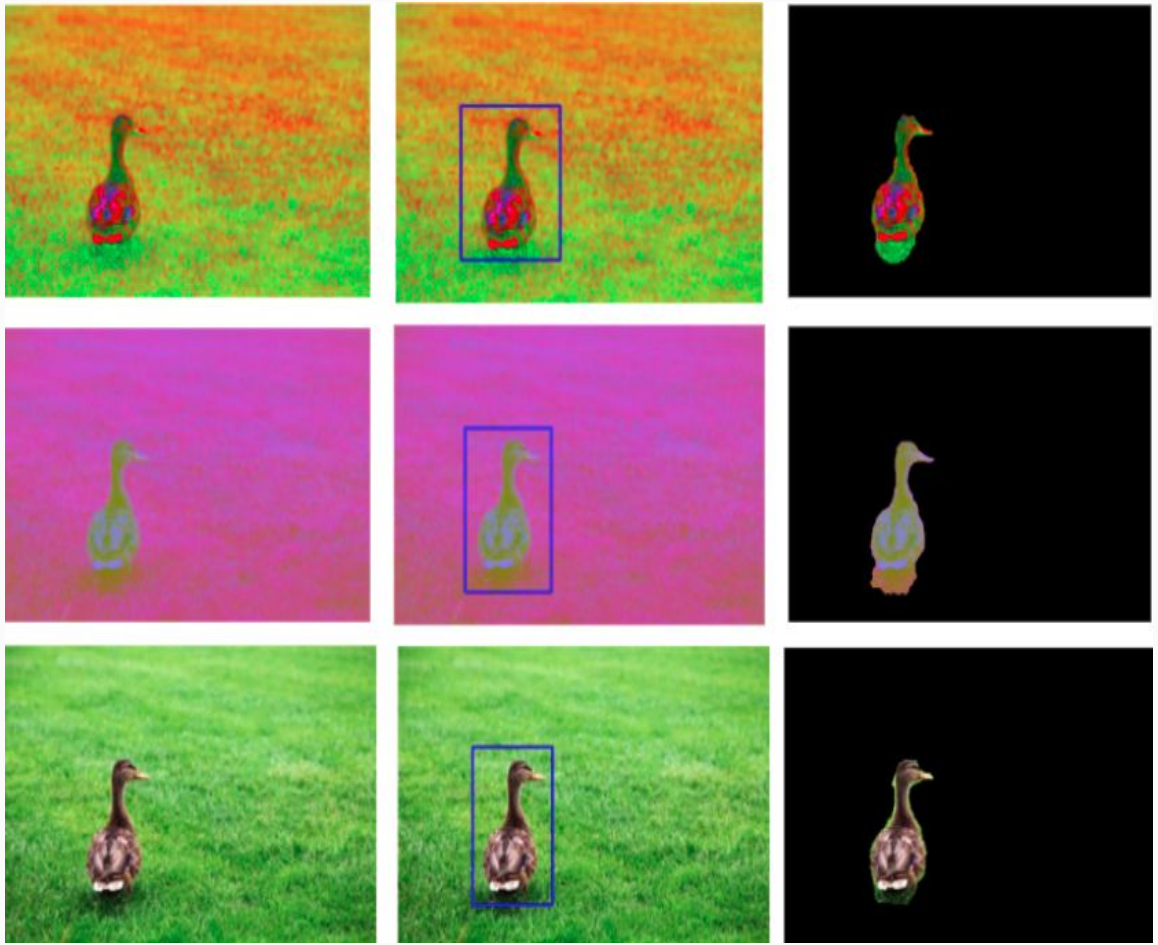
Varying Gamma Parameter

Ablation Studies



Varying No of GMM Components

Ablation Studies



In different Color Spaces

Conclusion

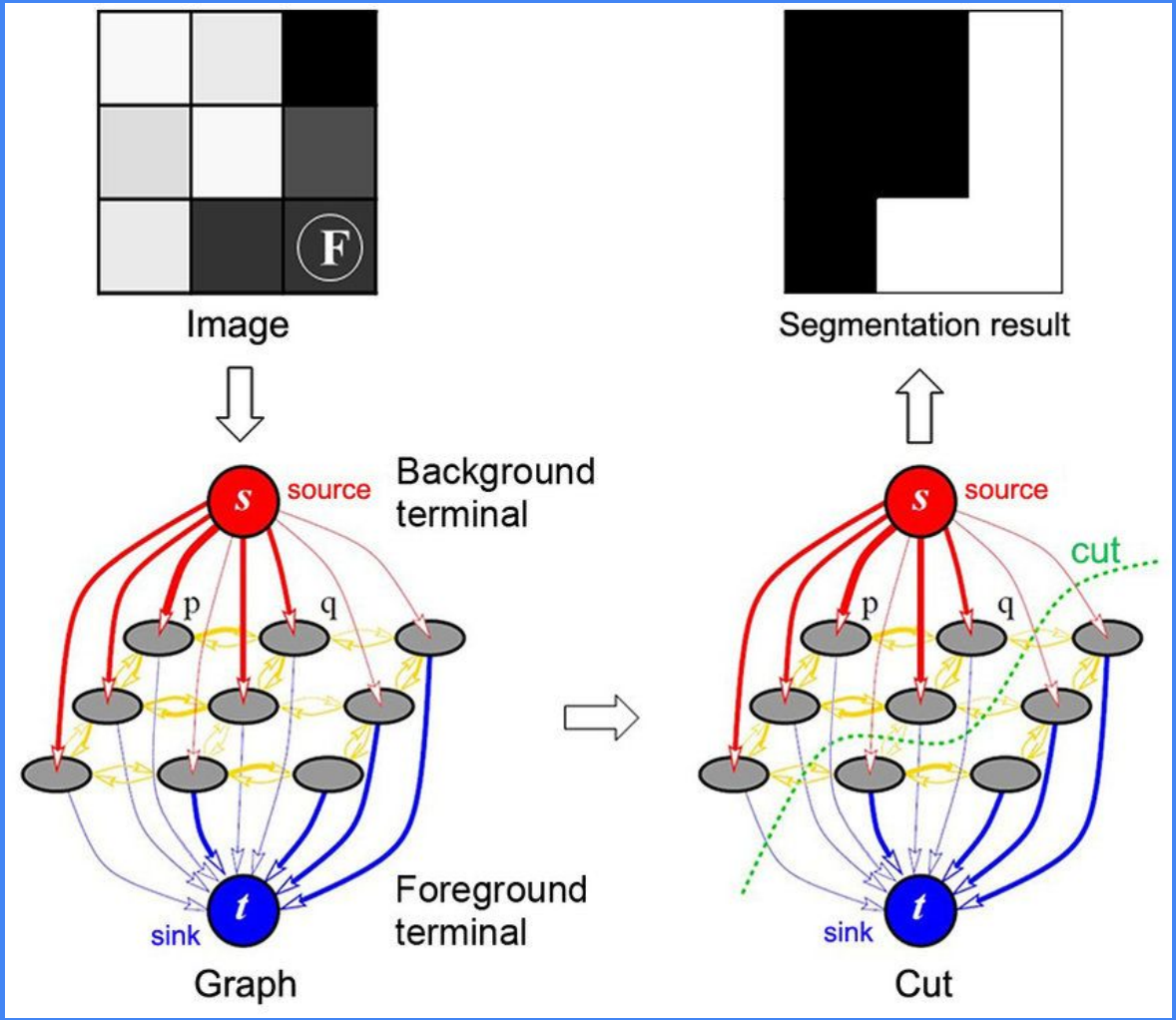
References

1. Graph Cut
   a. http://www.eecs.berkeley.edu/~efros/courses/AP06/Papers/boykoviccv01.pdf

2. GrabCut Interactive Foreground Extraction using Iterated Graph Cuts
   a. http://research.microsoft.com/apps/pubs/default.aspx?id=67890

3. Implementing Grabcut
   a. http://read.pudn.com/downloads94/doc/374106/Grabcut.pdf
   b. http://www.justintalbot.com/research/coursework/

Thanks!


F
Image
Segmentation result
s
source
Background
terminal
p
q
t
sink
Foreground
terminal
Graph
cut
Cut