

**Project ID:** 31

**Project Title:** Poisson Image Editing

**Github Link:** <https://github.com/jyotishp/Poisson-Image-Editor>

**Team members:**

Roll Number	Name
201530228	Sai Manish Rao
20161217	Jyotish

**Main Goals of the Project:**

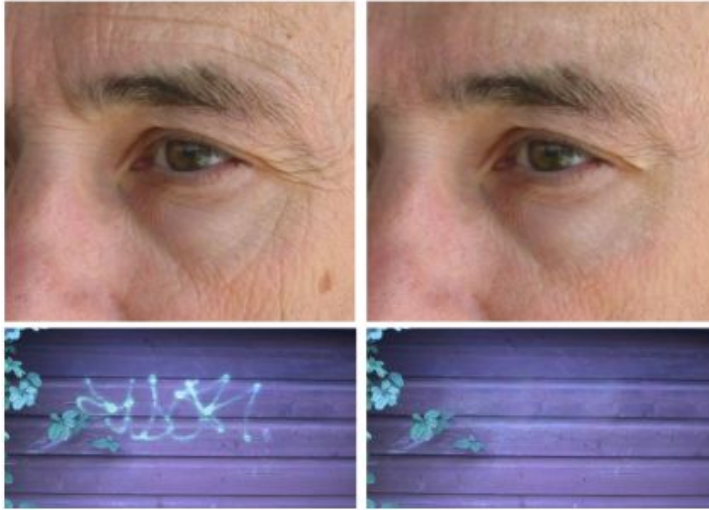
Seamless image editing and image cloning with the help of Poisson partial differential equation with Dirichlet boundary conditions.

**Problem Definition:**

- Image editing tasks can be local changes or global changes.
- The classical tools for image editing or image cloning usually involve a local selection (done manually) and applying filters (in case of editing) or copy-paste (in case of cloning) of this selected region.
- Such methods result in visible patches (seams) that disturb the naturality of the images.
- These patches are usually blended by feathering (transparency gradient along the borders of the patch) which works to some extent.
- Moreover, the classical methods require a precise delineation of object boundaries.
- In this project, we try to solve the image cloning and editing as a minimization problem where we compute the function whose L2 norm gradient is closest to the desired vector field called guidance vector field.
- This is equivalent to solving for the Laplacian of an unknown function over a domain with certain boundary conditions and filling the domain values. The domain filling is done by solving the Poisson equation.

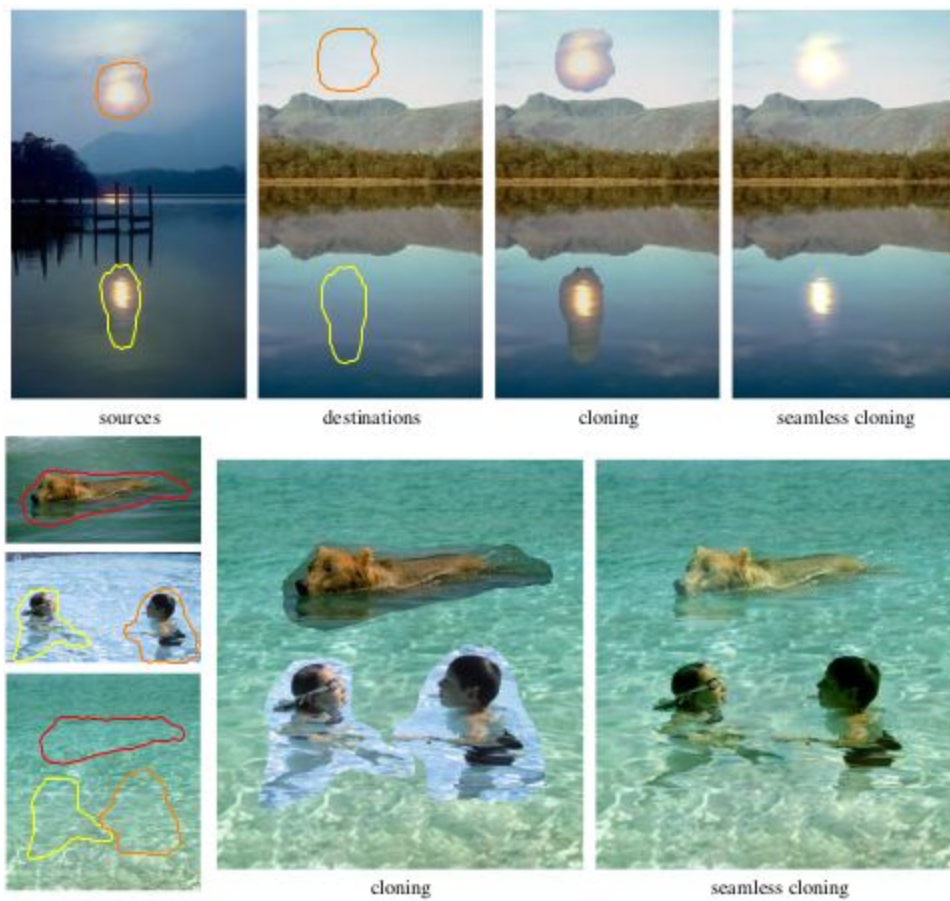
## Results of the Project:

- Concealment



- Cloning

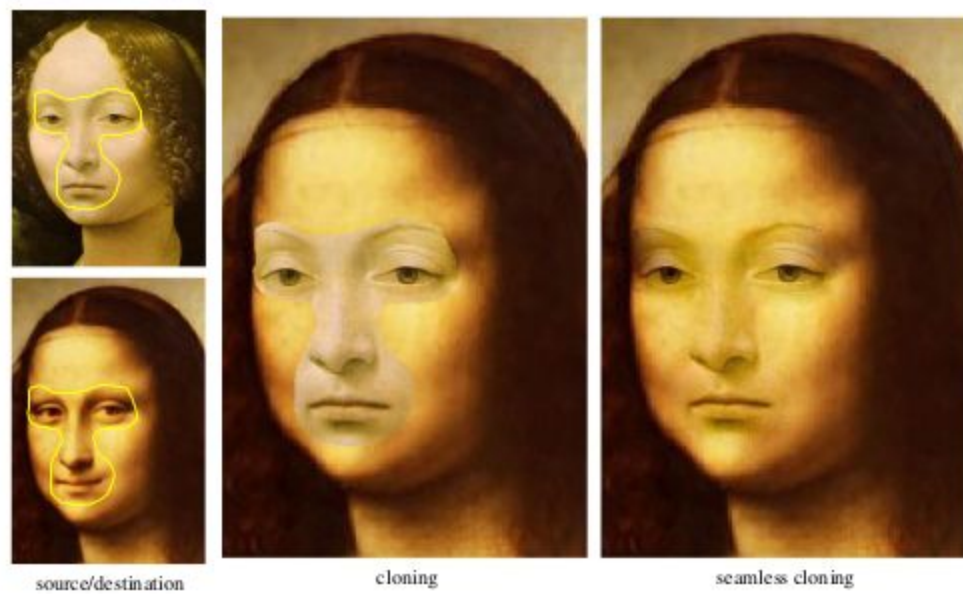
- Insertion



- Color transfer/exchange



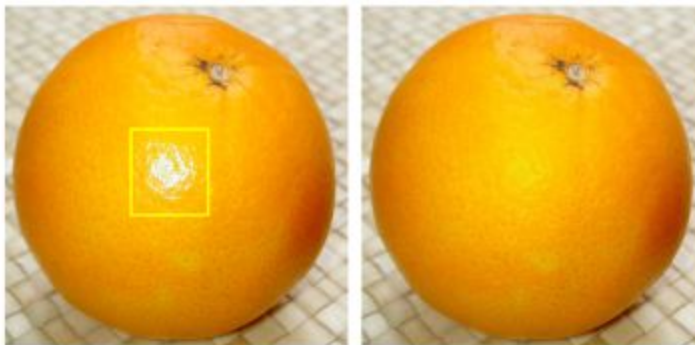
- Feature transfer/exchange



- Texture Flattening



- Illumination changes



- Local color changes without precise delineation





- Tiling



### Task Allocation:

Member	Task
Manish	<ul style="list-style-type: none"> <li>• Handle input images</li> <li>• Process the input images (Laplacian)</li> <li>• Implement concealment, cloning, texture flattening</li> <li>• Analyze the outputs from the implemented code</li> </ul>
Jyotish	<ul style="list-style-type: none"> <li>• Implement the Poisson sparse matrix</li> <li>• Implement cloning, local color changes, tiling</li> <li>• Sanitize code for user-friendly CLI</li> <li>• Analyze the outputs with 3rd party softwares</li> </ul>

### Project Milestones and Timeline:

Date	Task
Milestone 1	
09-10-2018	Study and understand the methods proposed in the paper
24-10-2018	Basic implementation of Poisson Image editing
Milestone 2	
02-11-2018	Sanitize the code for a user-friendly CLI interface
09-11-2018	Compare the results with the ones obtained from 3rd party softwares