REFLECTION REMOVAL BY GHOST EFFECT FROM SINGLE IMAGE

TEAM : BIG DIPPER

SUBODH SONDKAR : 2018101064 (CSE)

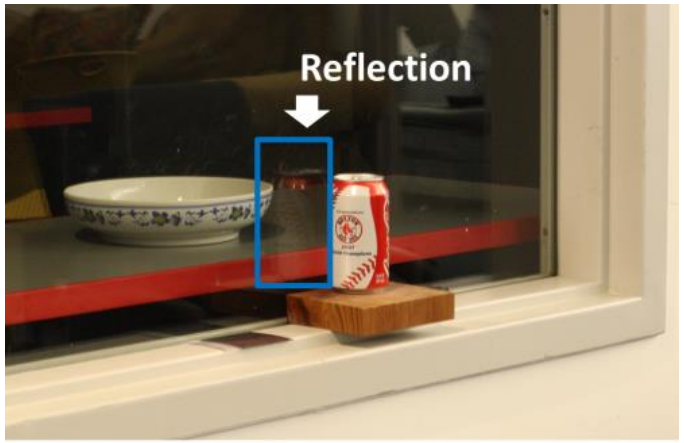
PRANJAL JAIN : 2019101103 (CSE)

ASHISH GUPTA : 2019101061 (CSE)

ISHA GUPTA : 2019101111 (CSE)

TA Mentor : FIZA HUSSAIN

PROJECT REPOSTORY URL

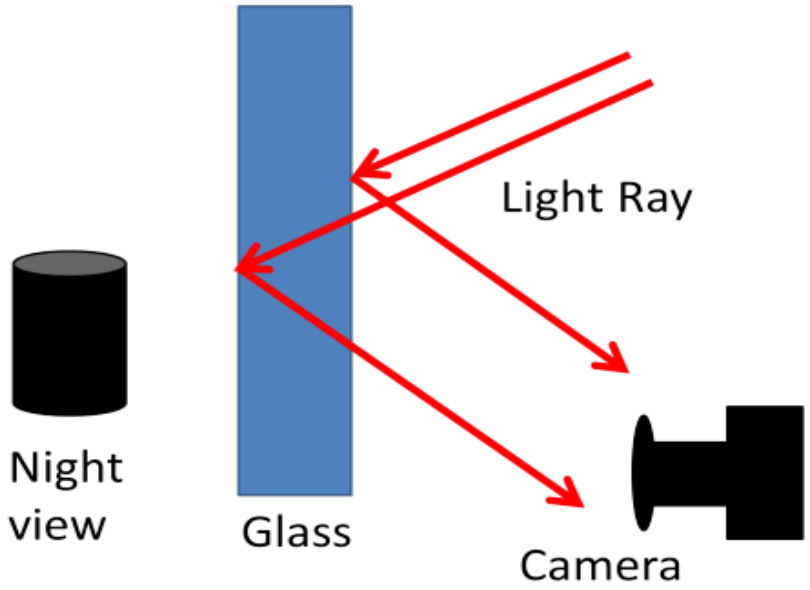


MAIN GOAL:

- The main goal of our project is to process the images using a proposed algorithm to remove the reflection artifacts caused while capturing image through the glass window.
- The objective is achieved using the ghost effect as a cue to distinguish the reflection image from scene image.

GHOST EFFECT:

Due to glass thickness, when each side generates reflection image and the difference between two duplicates are only upto a scale in image intensity, this phenomenon is called ghost effect.



SITUATIONS FOR GHOSTING

- The thickness of glass is more than 0.5cm.
- Angle between camera optical axis and glass normal is around 45 degrees.
- The reflected object is close to the glass (typically around 20 cm)

IMAGE MODEL

The image obtained is given by:

$$I(\vec{x}) = I_s(\vec{x}) + I_r(\vec{x}) + \beta I_r(\vec{x} - \vec{d})$$

- $I_s(x)$ -> scene image (desired image)
- $I_r(x)$ -> reflection image
- β -> intensity attenuation by glass
- d-> distance between two duplicate reflection images

OBJECTIVE :

- For reflection removal we want to solve for $I_r(x)$ and $I_s(x)$
- Solving for both yields us the scene image I.e. $I_s(x)$

IDEAS :

Treat $I_s(x)$ as noise and deconvolve using kernel. This removes ghost effect but reflection and desired scene image are still mixed up.

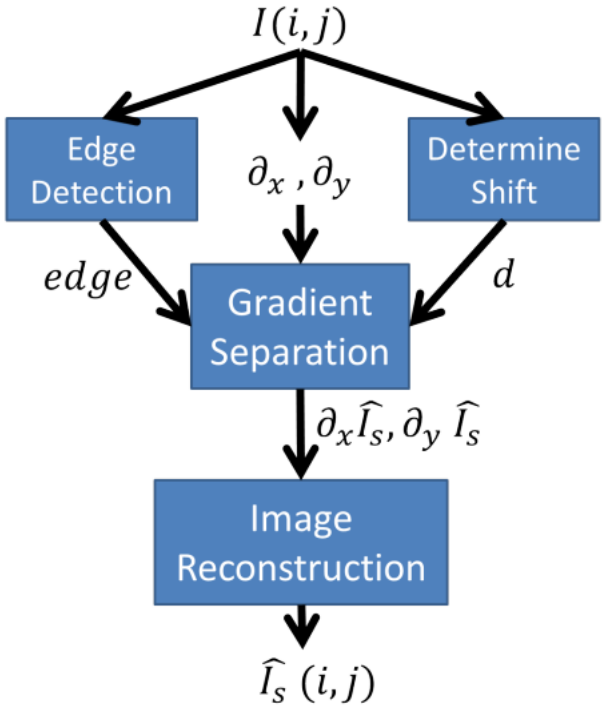
The reflection image and the scene **image** are mixed together, so we cannot separate the two **images** by assigning each pixel to one of the two images.

As edges of reflection and desired image do not overlap hence we assign gradient of each pixel to one of two images.

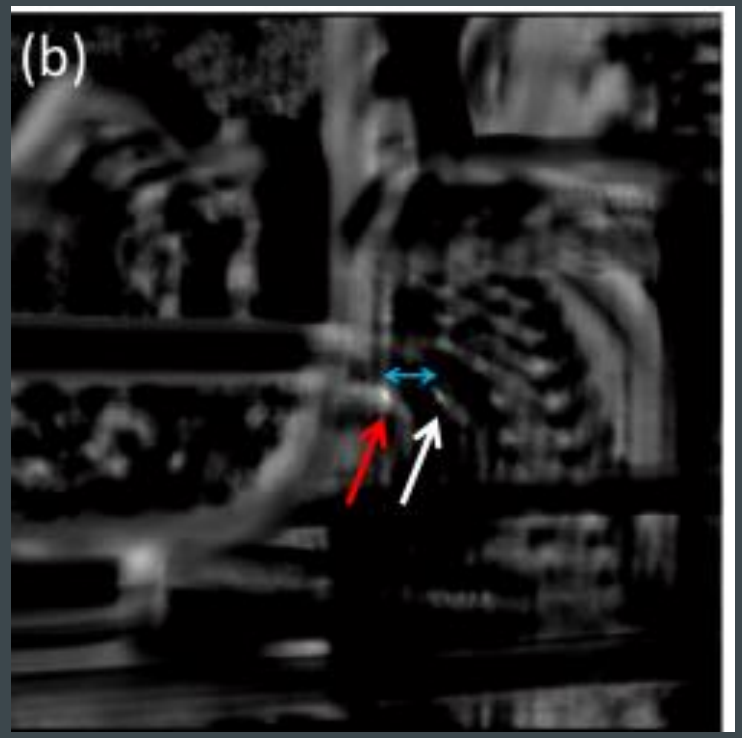
So, In our implementation we find gradient of scene image using gradient separation technique and then reconstruct the desired image.

ALGORITHM OVERVIEW

1. Determine the shift amount of ghost effect.
2. Determine gradient by edge detection.
3. According to ghost effect separate gradient of reflection image from gradient of desired image.
4. Reconstruct desired image by scene gradient.



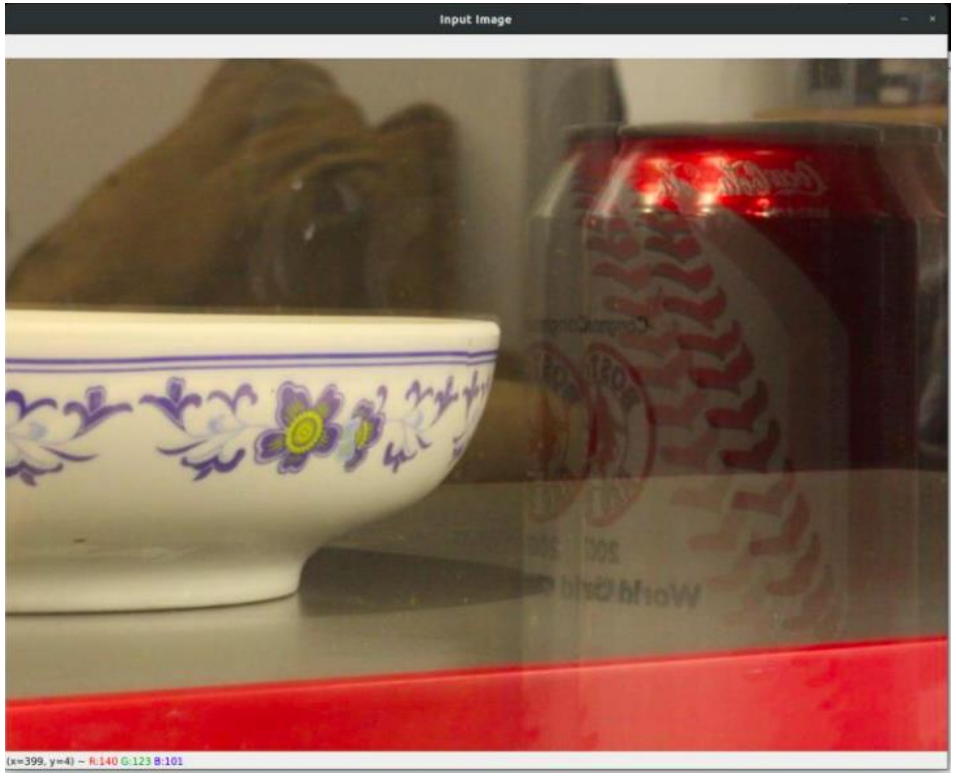
STEP 1 :
SHIFT DETERMINATION



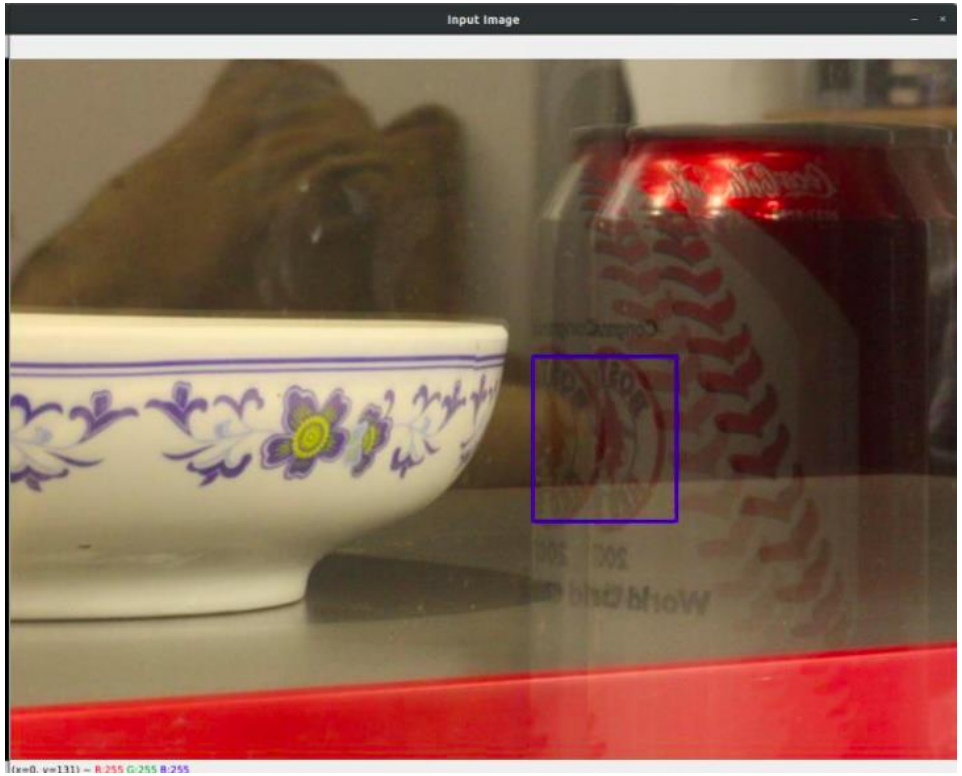
SHIFT DETERMINATION

- We need to determine the shift d between the two reflection duplicates.
- The shift due to the ghost effect helps us to separate gradient (calculated in later stage) of reflection image from gradient of desired image.
- To determine this shift amount, we devise a user assisted system.
- The user crops a patch of reflection image as a template, which is used to perform template matching.

PATCH SELECTION

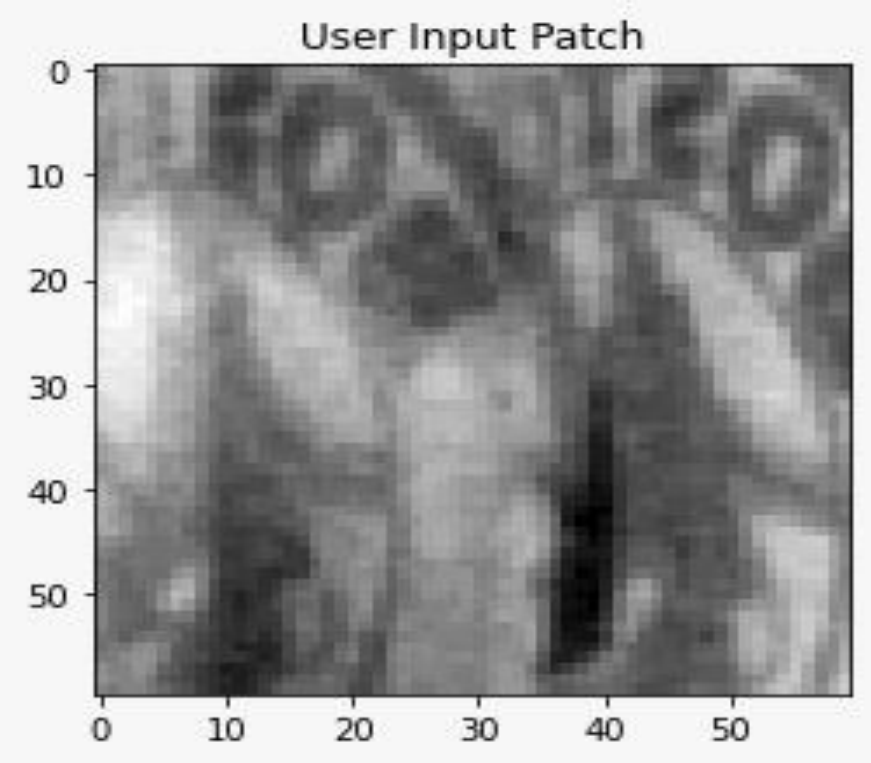


Original Image

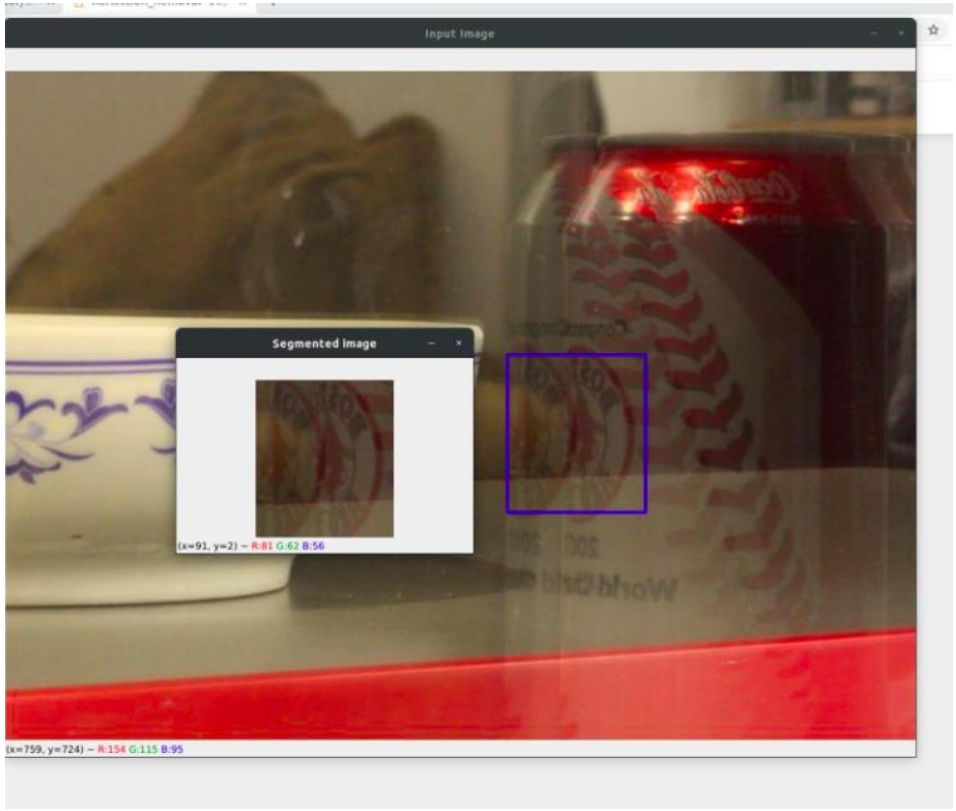


Patch selection by user (drag and drop)

PATCH SELECTION



User Input Patch



Screen shown to user after patch selection

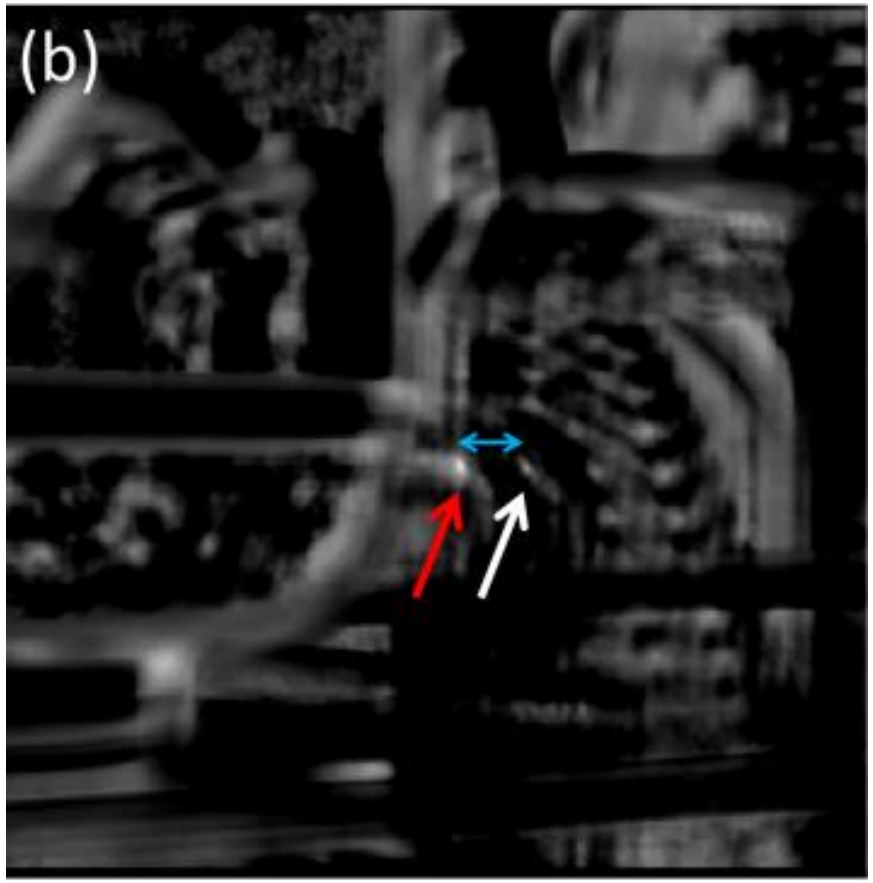
TEMPLATE MATCHING

- In template matching we find similar templates in a source image by giving a base template to compared on.
- This is done by comparing each of the pixel values of the source image one at a time to the template image.
- The output would be an array of similarity values when compared to the template image.

FLOW OF SHIFT DETERMINATION

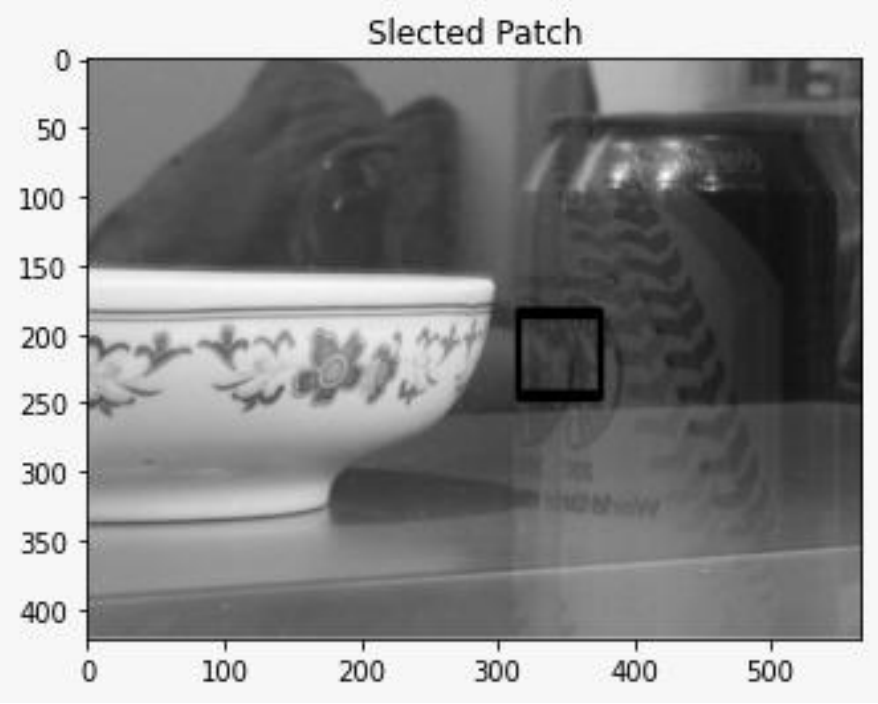
- Using template matching we get matrix of values corresponding to each image pixel which tells us how similar our image pixel is to the selected patch.
- The value 1 in matrix signifies the portion corresponding to selected patch which acts as the **global maximum** and the original part of reflection image.
- The second maxima value in this matrix, gives us the **local maxima** and it corresponds to the duplicate of reflection image.

d = pos(global maxima) – pos(local maxima)

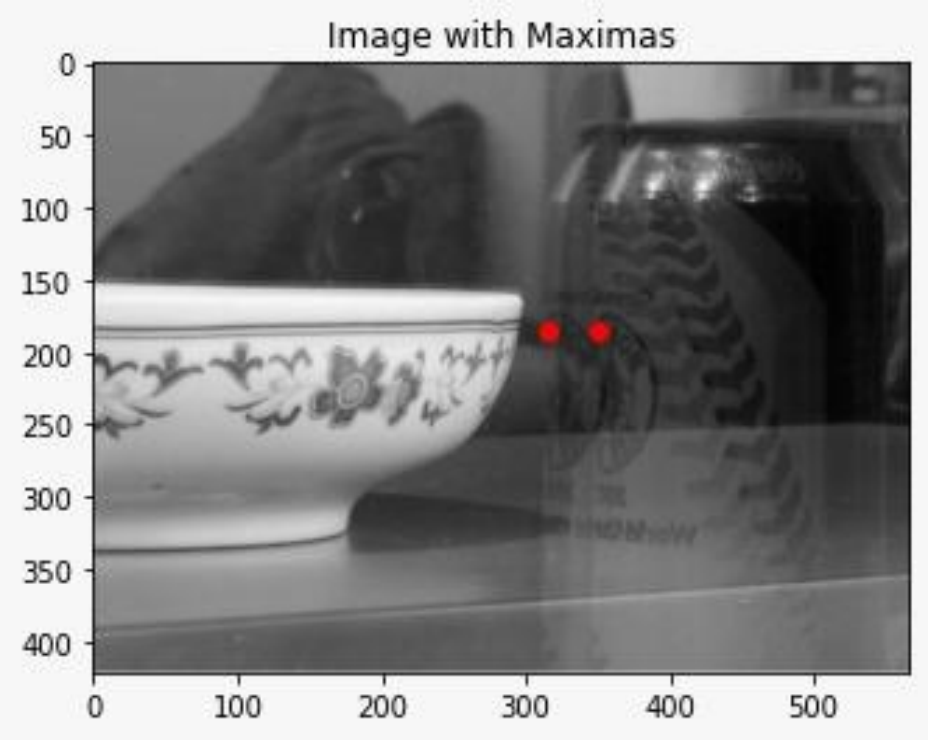


Template matching result.

SHIFT DETERMINATION

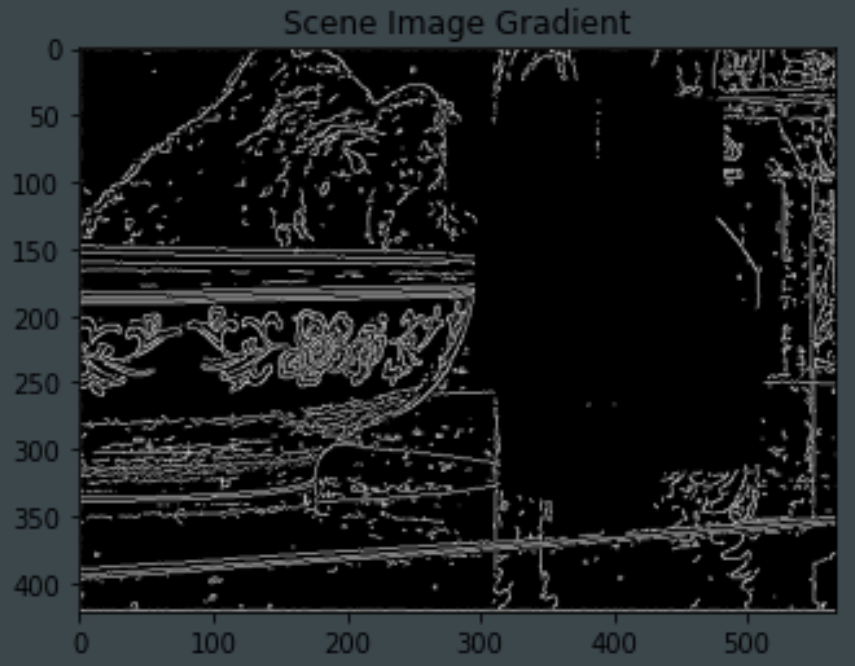


Selected patch for processing



Two red dots signifies global and local maxima respectively.
Distance b/w these points is called shift.
Shift(d) = (35,1)

STEP 2 :
GRADIENT SEPARATION



GRADIENT SEPARATION

- The task of gradient separation is to find if each gradient belongs to either reflection image or scene image.
- This is accomplished with the help of ghost effect i.e. shift calculated in previous step.

ALGORITHM

Considering each pixel in the image as a patch, we perform template matching between patch and original image.

If there is duplicate at (i,j) + d or (i,j) - d and so local maximum on template matching result:

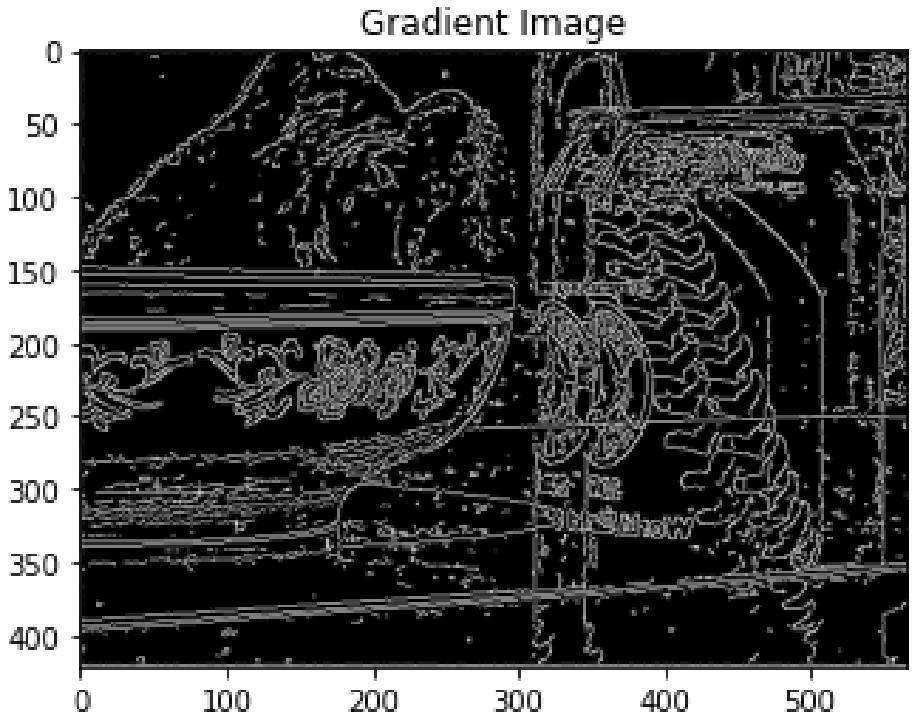
Then (i,j) corresponds to **reflection image.**

If there is no duplicate at (i,j) + d or (i,j) - d

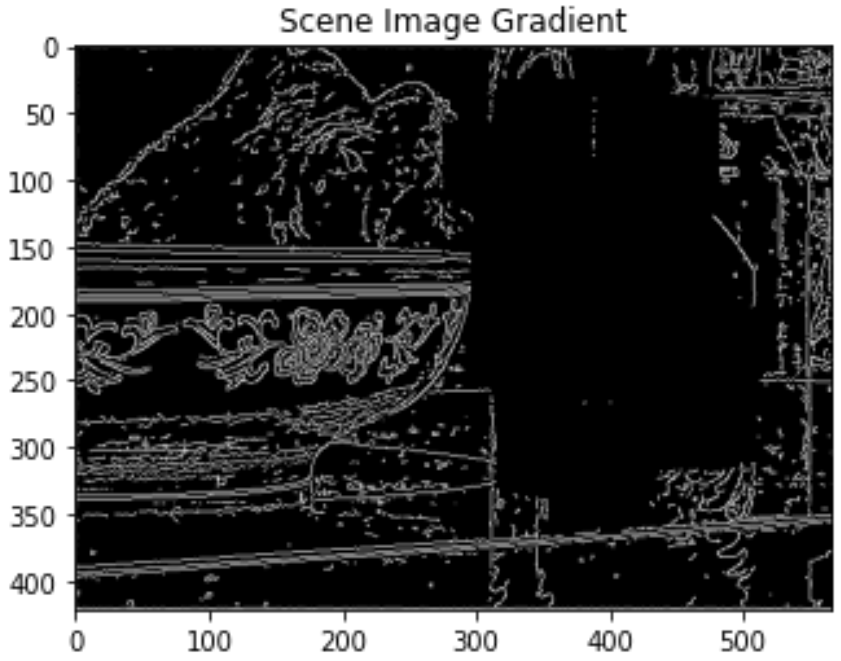
Then patch P and hence (i,j) corresponds to **scene image.**

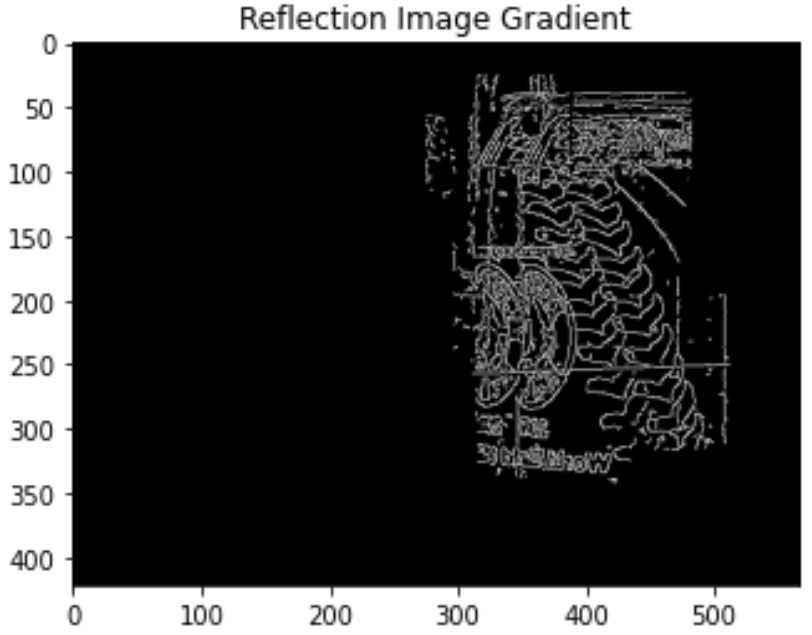
EDGE DETERMINATION

- Running template matching on each pixel costs a huge amount of calculation, and hence unnecessary to run on every pixel.
- Therefore, we only run template matching on the pixels of the important gradient.
- Therefore, we use an edge detection to label strong gradient, and then only separate these important gradients.



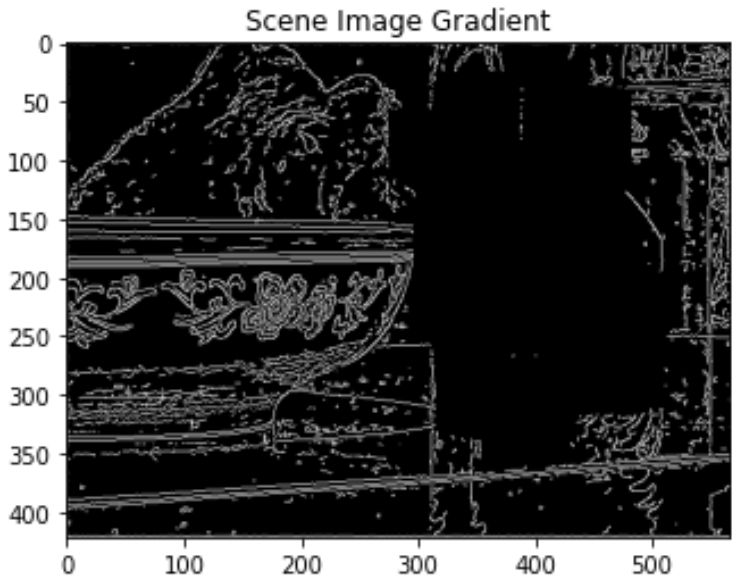
GRADIENT SEPARATION RESULTS

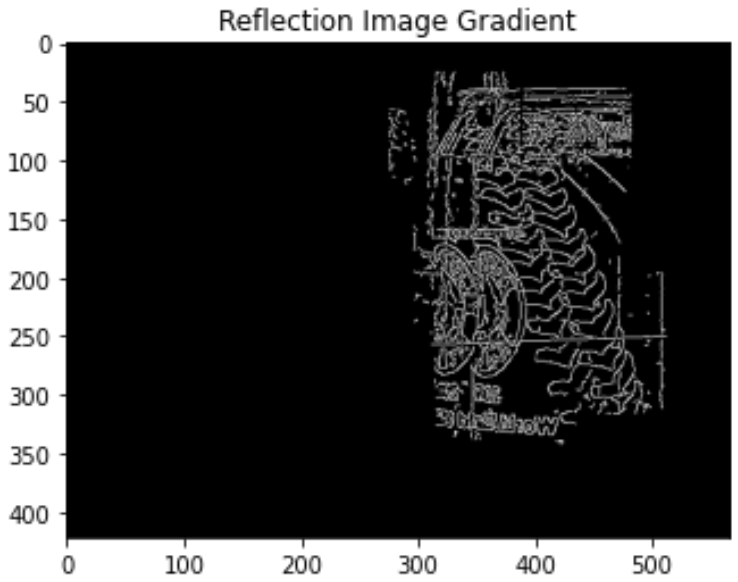


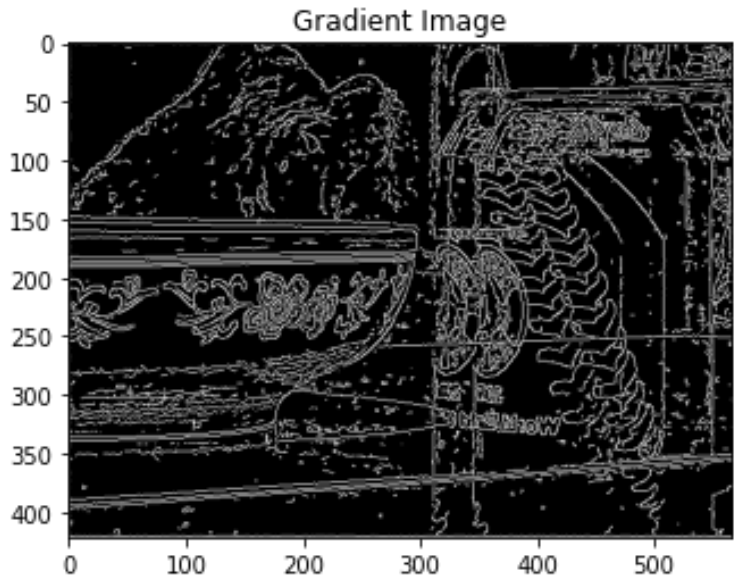


EDGE DETERMINATION

- In the paper, **Canny edge detection algorithm** is used for edge determination using different thresholds.
- We also tried **Morphological Operations** for edge detection because there was too much noise in canny algorithm results.
- So therefore edge determination can be performed using both methods.



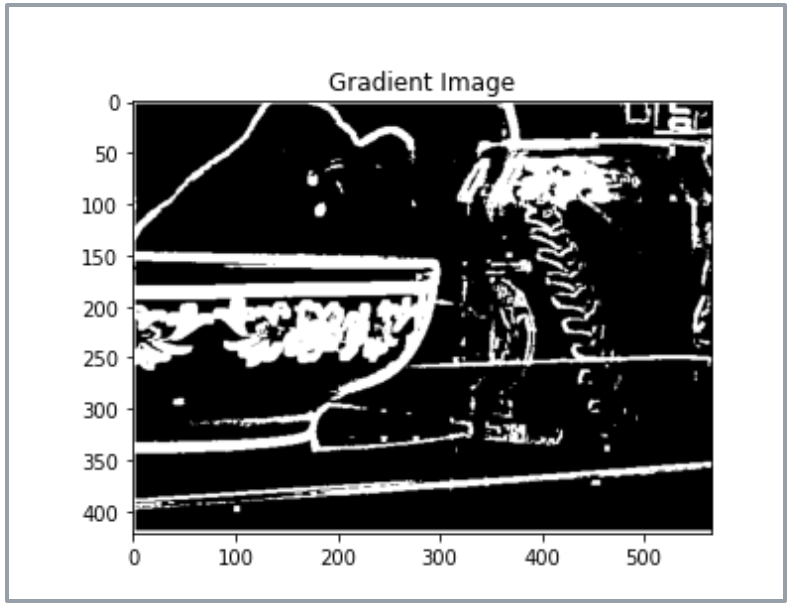


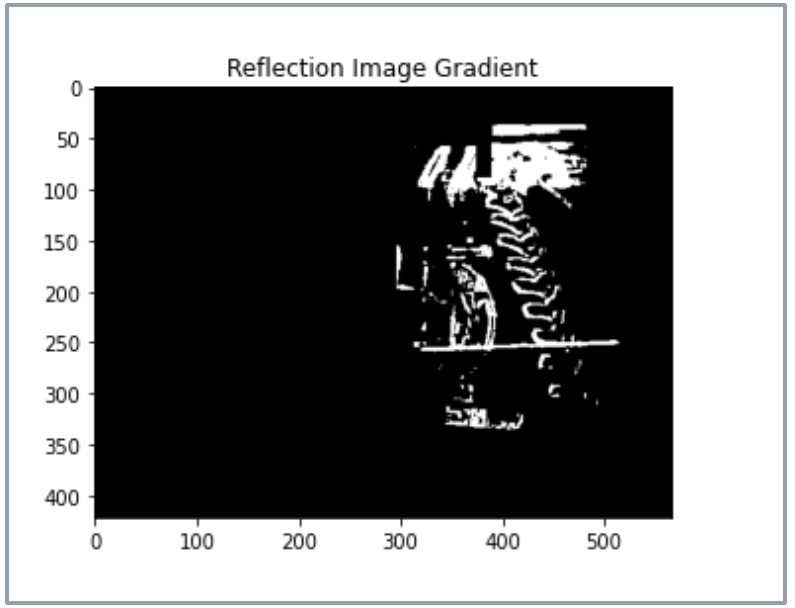


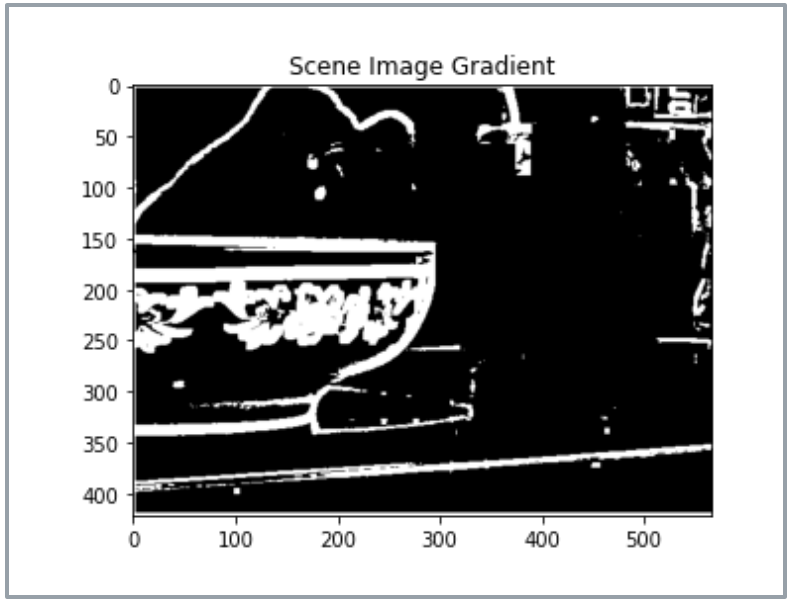
CANNY EDGE DETECTION OUTPUTS

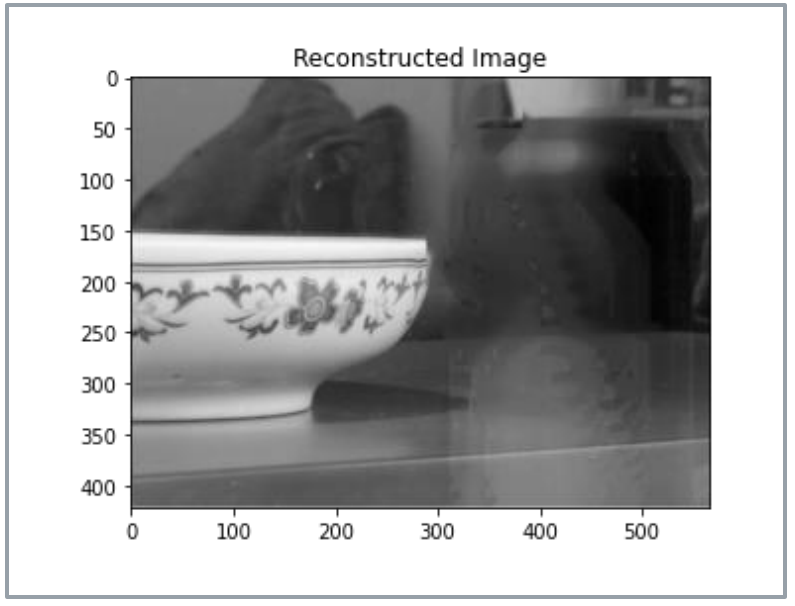
These are the outputs obtained using canny edge detection algorithm.

MORPHOLOGICAL OPERATION OUTPUT









STEP 3 :
IMAGE RECONSTRUCTION

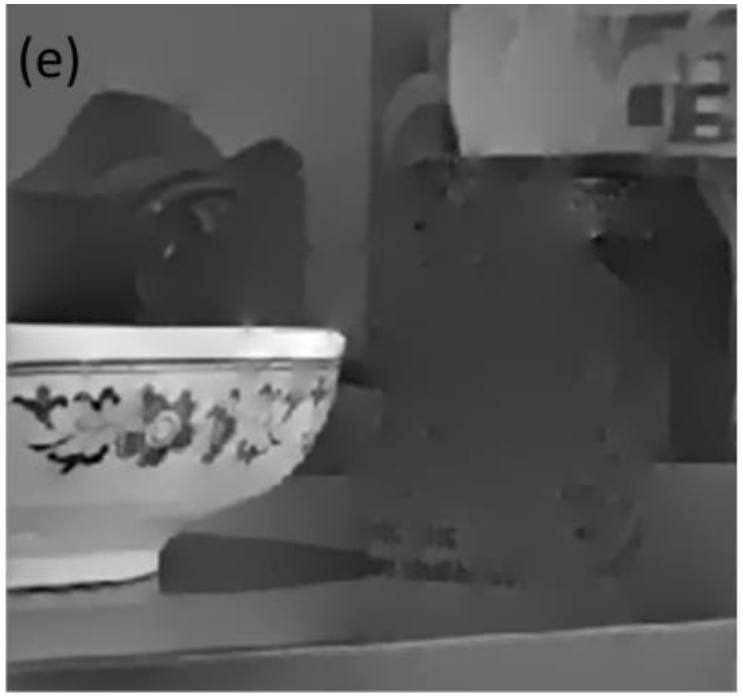
ALGORITHM

Dilation is performed on reflection image gradient to incorporate the entire reflection part.

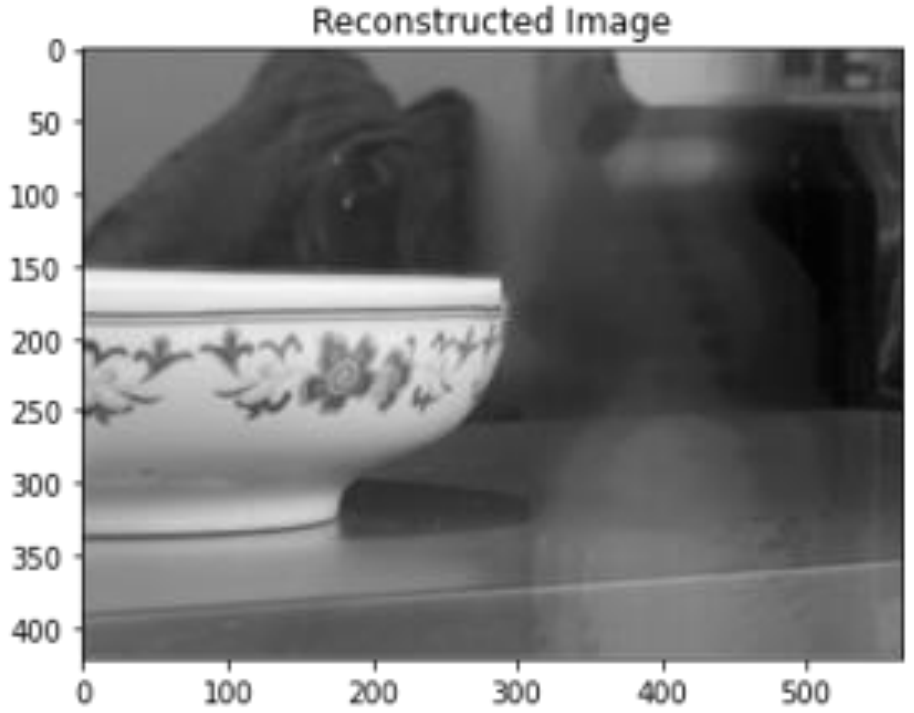
After that blurring is performed on the area of reflection image to remove the reflected part.

This blurred reflection image is combined with the scene image to result in final image.

RECONSTRUCTED IMAGE



Expected output in paper

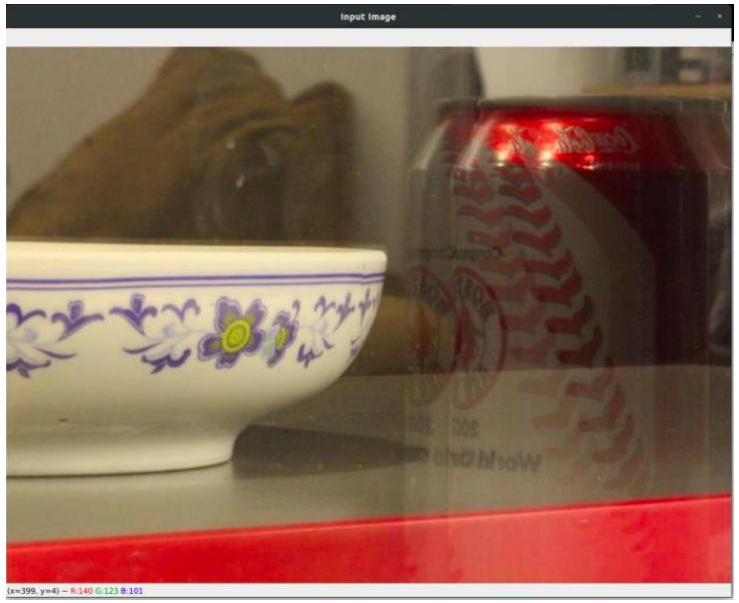


Output from our code.

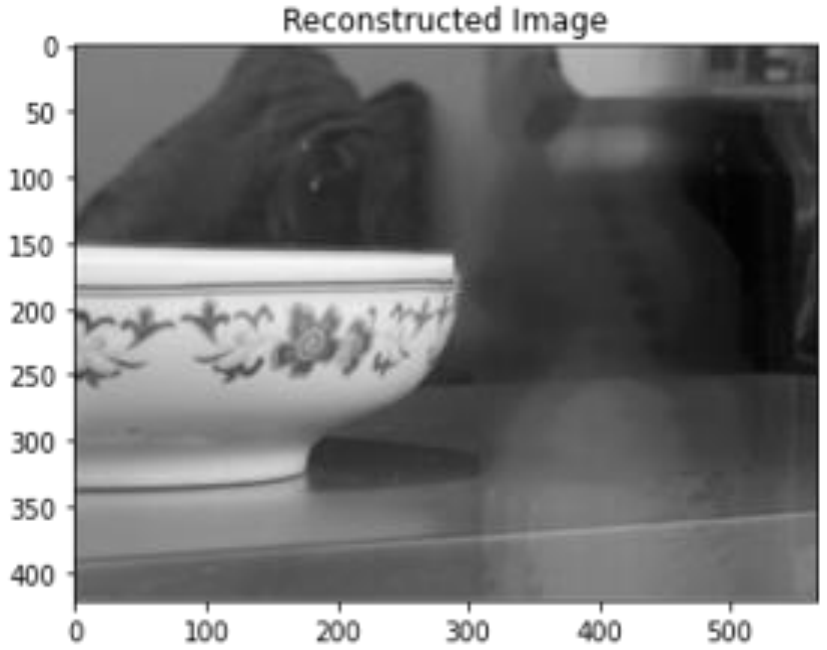
- The edges of the table are preserved in our output.
- In reconstruction results reflection image portion is removed.
- Desired scene image is produced.

RESULTS ON IMAGES

RESULTS OF IMAGE (A)

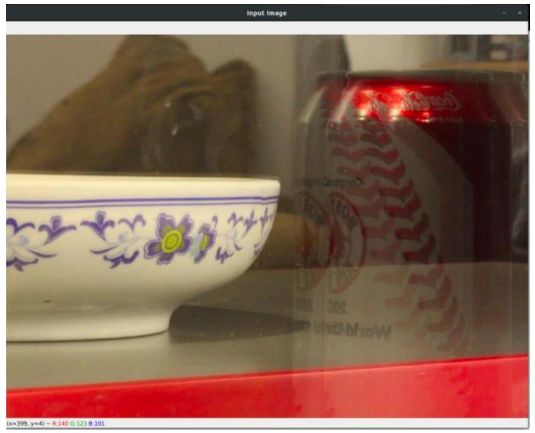


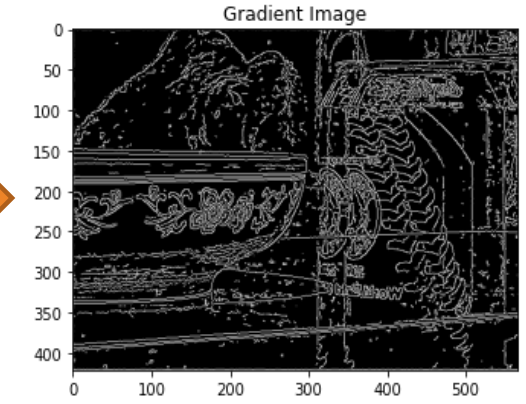
INPUT IMAGE

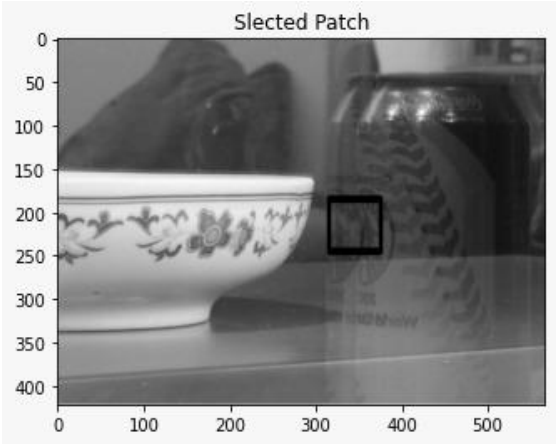


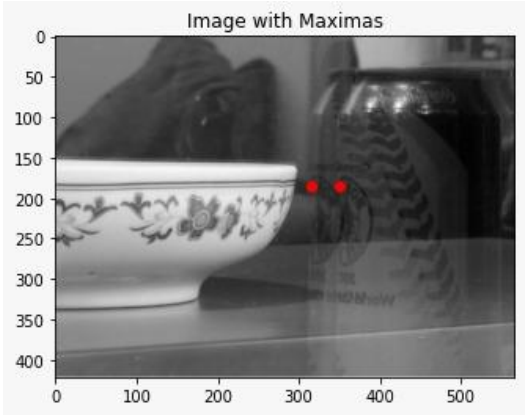
FINAT OUPUT

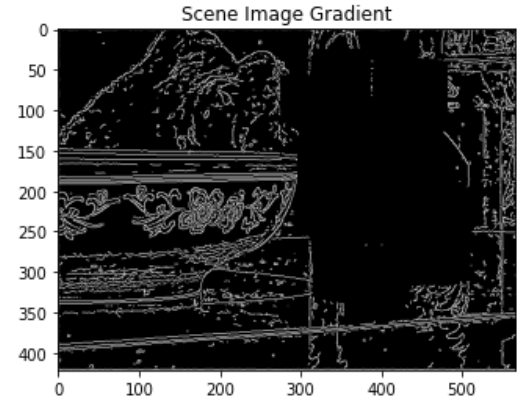
COMPLETE PROCESS OF IMAGE (A)

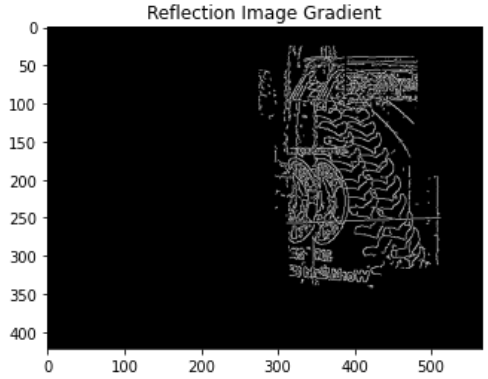


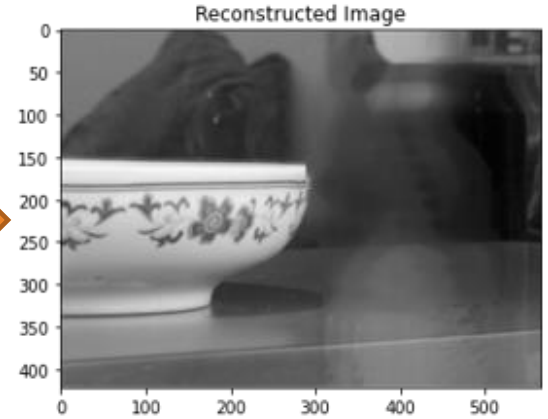




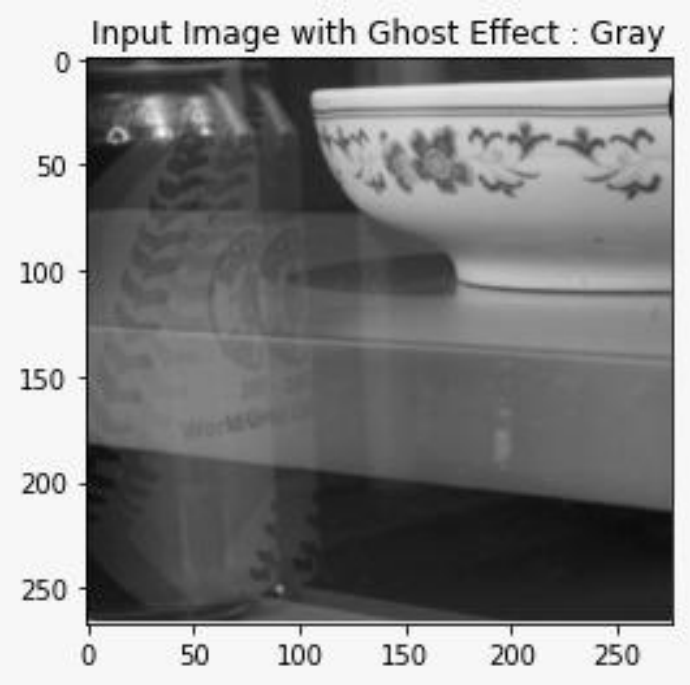




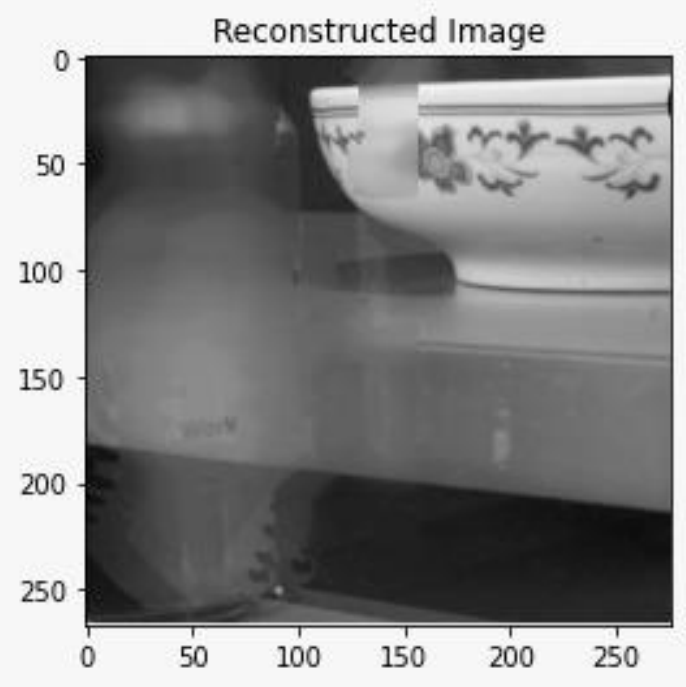




RESULTS OF IMAGE (B)



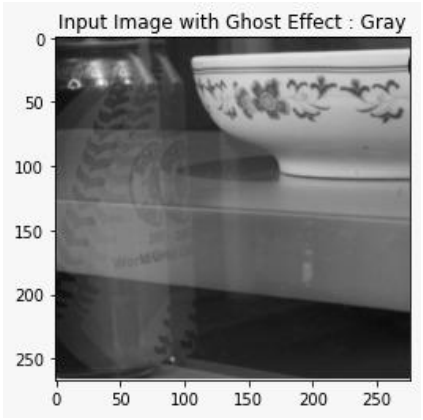
INPUT IMAGE

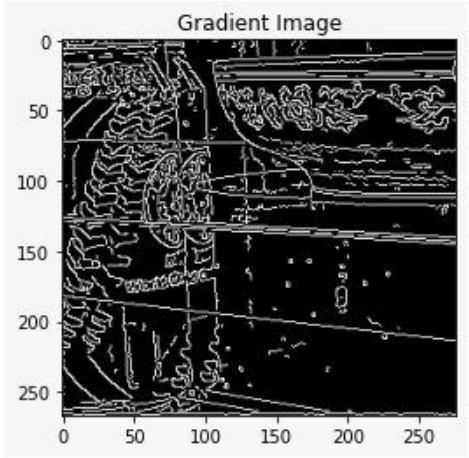


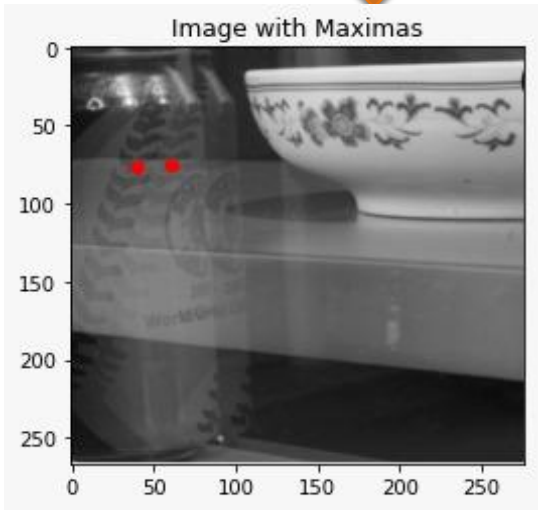
FINAT OUPUT

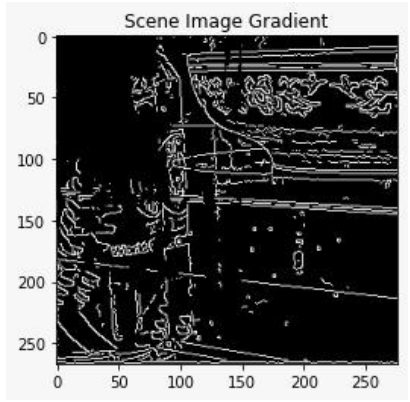
Note: This image is taken through different angle, and flipped from previous one.

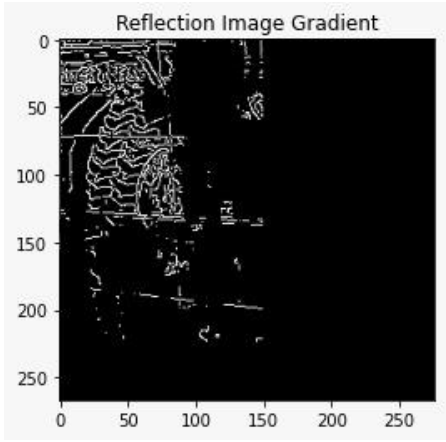
COMPLETE PROCESS OF IMAGE (B)

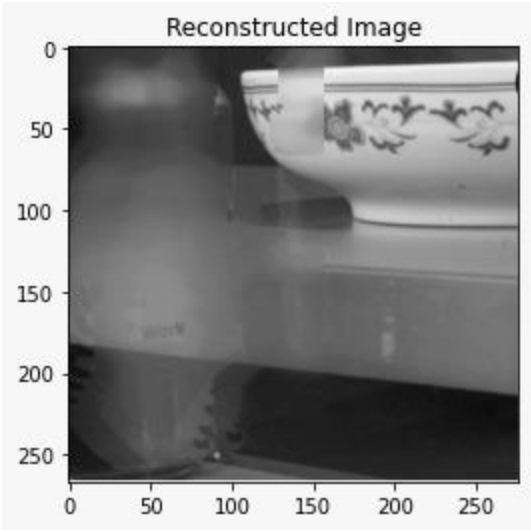




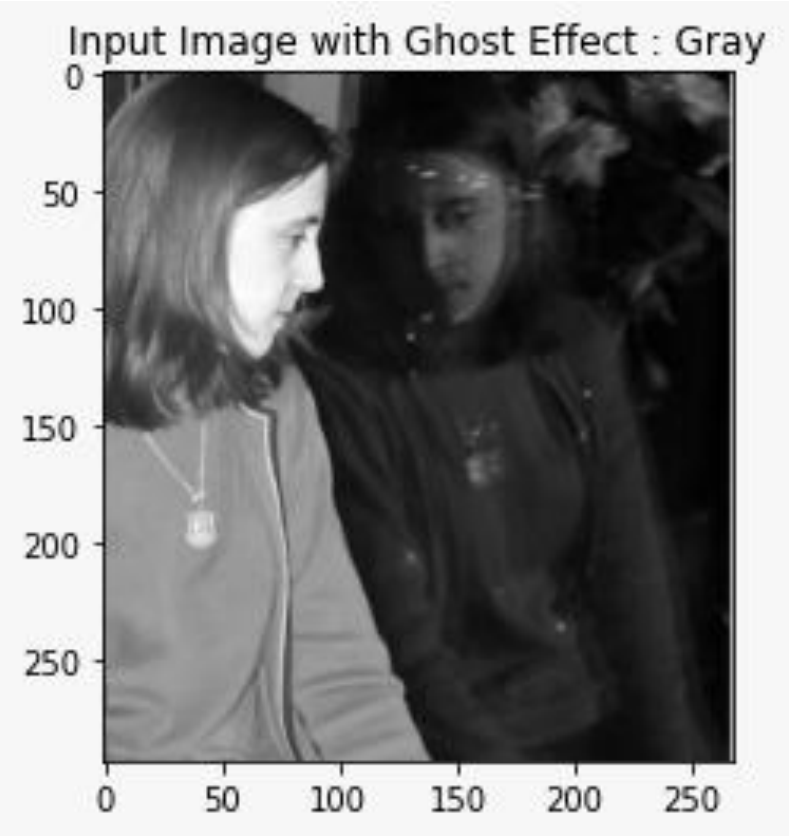








FAILED CASES



LIMITATIONS

Obtained image must have strong ghost effect.

When user is perpendicular to glass, i.e. d = 0; hence no ghost effect and algorithm fails.

When reflection object is too far from glass, ghost effect is less.

When glass is too thin (β is small), ghost effect is less.

When reflection image is very complex and if more overlap, then algorithm doesn't work well.

If scene image has duplicates with period d, then **it treats** scene gradient as reflection gradient, and so fails.

WORK DISTRIBUTION

Pranjal Jain

- Gradient Separation.
- Shift Determination
- Paper understanding + Internet exploration.

Ashish Gupta

- Template matching
- Shift Determination.
- Exploring various papers.

Subodh Sondkar

- GUI for patch selection.
- Morphological edge detection.
- Reconstruction of image.

Isha Gupta

- Canny Algorithm edge detection.
- Documentation -> proposal + presentation + README.
- Paper understanding + Internet exploration.

THANK YOU