DIP Project Proposal

Project ID: Big Dipper - 11

Project Title:

[Reflection Removal by Ghost Effect from A Single Image]

GitHub:

https://github.com/Digital-Image-Processing-IIITH/dip-project-big-dipper

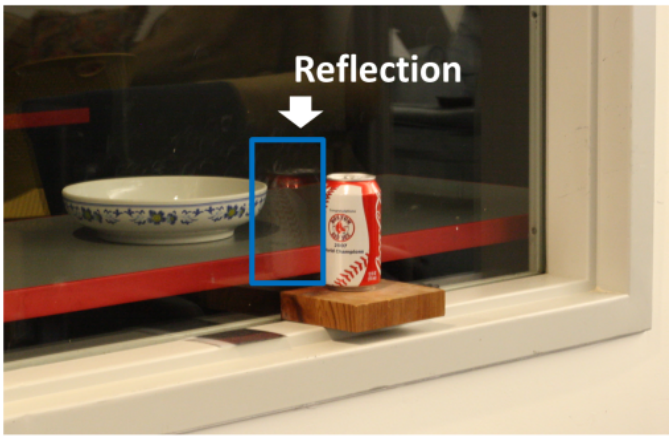
Team Members:

- Subodh Sondkar (2018101064)
- Isha Gupta (2019101111)
- Pranjal Jain (2019101103)
- Ashish Gupta (2019101061)

Main Goal

When we take an image through a window, the glass reflects the light rays from the photographer's side. Hence, the obtained photo is a linear superposition of two images: a scene image beyond the window plus a reflection image before the window.

If there is a non-zero angle between the camera optical axis and the glass normal, the reflection image appears a ”ghost effect” due to the thickness of the glass, and hence **our main goal is to propose a reflection removal method using this ghost effect**. We will do this by separating the gradient of two images and then reconstructing the scene image from its gradients.



This is an Input Image. In this image, the coke can is the reflection image, while the plate and desk are desired images. Our goal is to remove that reflection image while maintaining the plate and desk image.

Problem Definition

What is the problem?

When each side of the glass generates a reflection image and the difference between
the two duplicates are only up to a scale in image intensity, this phenomenon is known as the ghost effect.

It can happen when:

1. the glass is thick and is round or more than 0.5 cm
2. angle between camera optical axis and glass normal is non-zero.
3. the reflected object is close to the glass.

The obtained image can be represented by

$$I(x) = Is(x) + Ir(x) + \beta Ir(x-d)$$

Is = scene image

Ir = reflection image

Beta = intensity attenuation by glass

d = shift caused by ghost effect

Problem: We want to solve for Ir and Is, but the reflection image and scene image are mixed together, therefore we cannot separate the two images by assigning each pixel to one of the images. Hence we focus on the gradient domain.

How will things be done?

We will solve our problem using **Gradient Domain Image Editing**. In this technique, we collect the gradient of the scene image by what we call the gradient separation technique and then reconstruct the scene image accordingly.

STEPS:

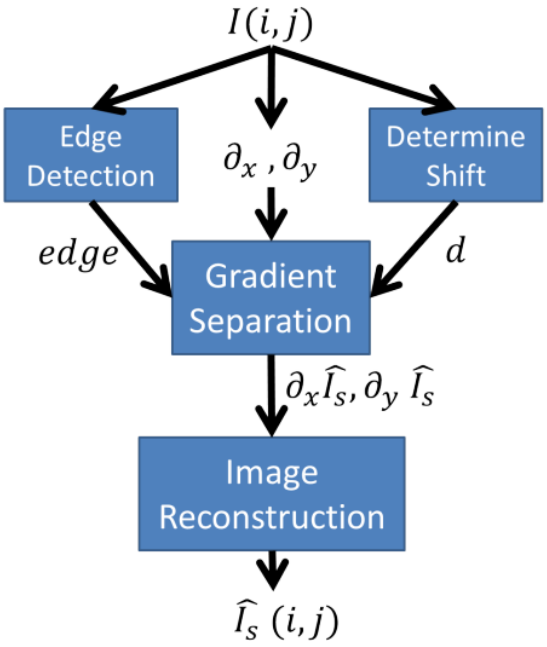
1. **Shift Amount Determination**
   Before separating gradient, we need to determine the shift amount d between the two reflection duplicates which we do by designing a user-assisted system.
2. **Gradient Separation**
   The task of gradient separation is to determine if each gradient belongs to either the reflection image or the scene image.

- Firstly use edge detection to label strong gradients.
- Then separate these important gradients or in other words, we discard edges.

  A user can set up a lower threshold for edge detection to preserve more gradients.

3. **Image Reconstruction**

   In this task, we discard the gradient of the reflection image and reconstruct the scene image by scene gradients using the devolution technique.

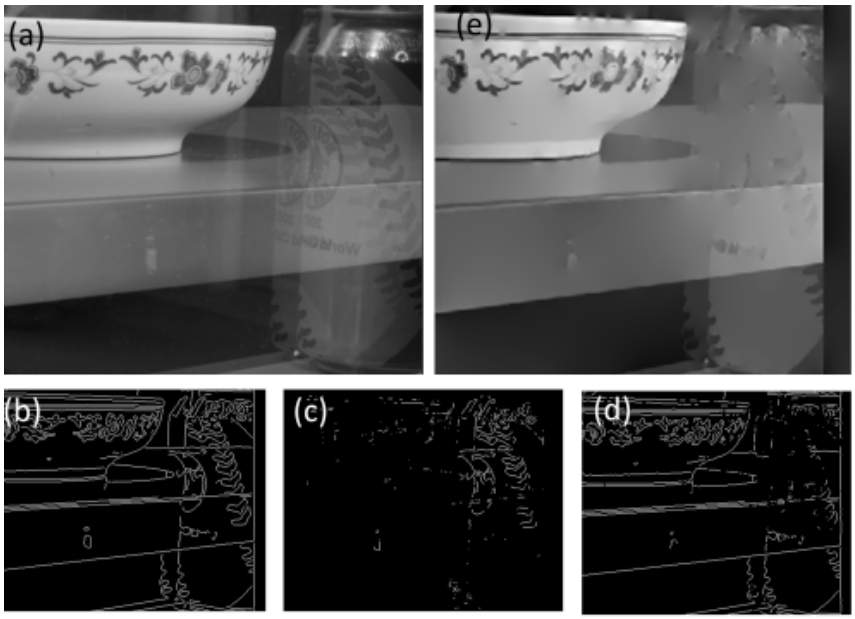


Algorithm overview of our proposed reflection removal.

Results

What will be done? What is the expected result?

The result of the project will be an algorithm that will demonstrate how a physics-based approach helps in the removal of reflection from a single image.



Different images at different steps of our algorithm. (a) Input Image. (e) Reconstructed desired image.

What are the project milestones and expected timeline?

10 Nov - 16 Nov:

Using edge detection algorithm to determine edge map. We use different thresholds for choosing the computational task amount and to preserve relevant details.

16 Nov - 22 Nov:

Defining and implementing all the steps of Gradient Domain Image Editing.

22 Nov - 28 Nov:

Final Testing: Use different datasets to test our algorithm on different machines and work on optimization of the algorithm.

Is there a data set that you require? How do you plan to get it?

Dataset is required to test the code on different images.

We can use the given data set also:

https://github.com/thongnguyendev/single_image/tree/master/release/synthetic_data