Single Image Layer Separation Using Relative Smoothness

Sudarshan Biswas, Rahul Kumar, Rajat Roy

I. INTRODUCTION:

The goal of this project is to solve the problem of layer separation from a single-image with application to

- Intrinsic Image Decomposition
- Single Image Reflection Interference removal using focus

In both of the cases, the problems take the form

$$I = L_1 + L_2 \quad (1)$$

II. INTRINSIC IMAGE MODEL:

This model assumes that an image scene is a product of a scene's reflectance and illumination at each pixel, expressed as:

$$I = RL \quad (2)$$

$$\log(I) = \log(R) + \log(L) \quad (3)$$

where, R = Reflective property of the object

L = Illumination falling onto the object

The aim of Intrinsic image decomposition is to separate R and L from the observed intensity I.

Note: Equation (1) and (3) take the same form.

III. REFLECTION INTERFERENCE:

Reflection interference arises when the image is taken behind a glass window. This can be expressed as a linear combination of a reflection layer and desired background scene:

$$I = L_B + L_R \quad (4)$$

where, L_B = Desired background scene

L_R = Reflection layer

The aim of this is to separate these two layers as well.

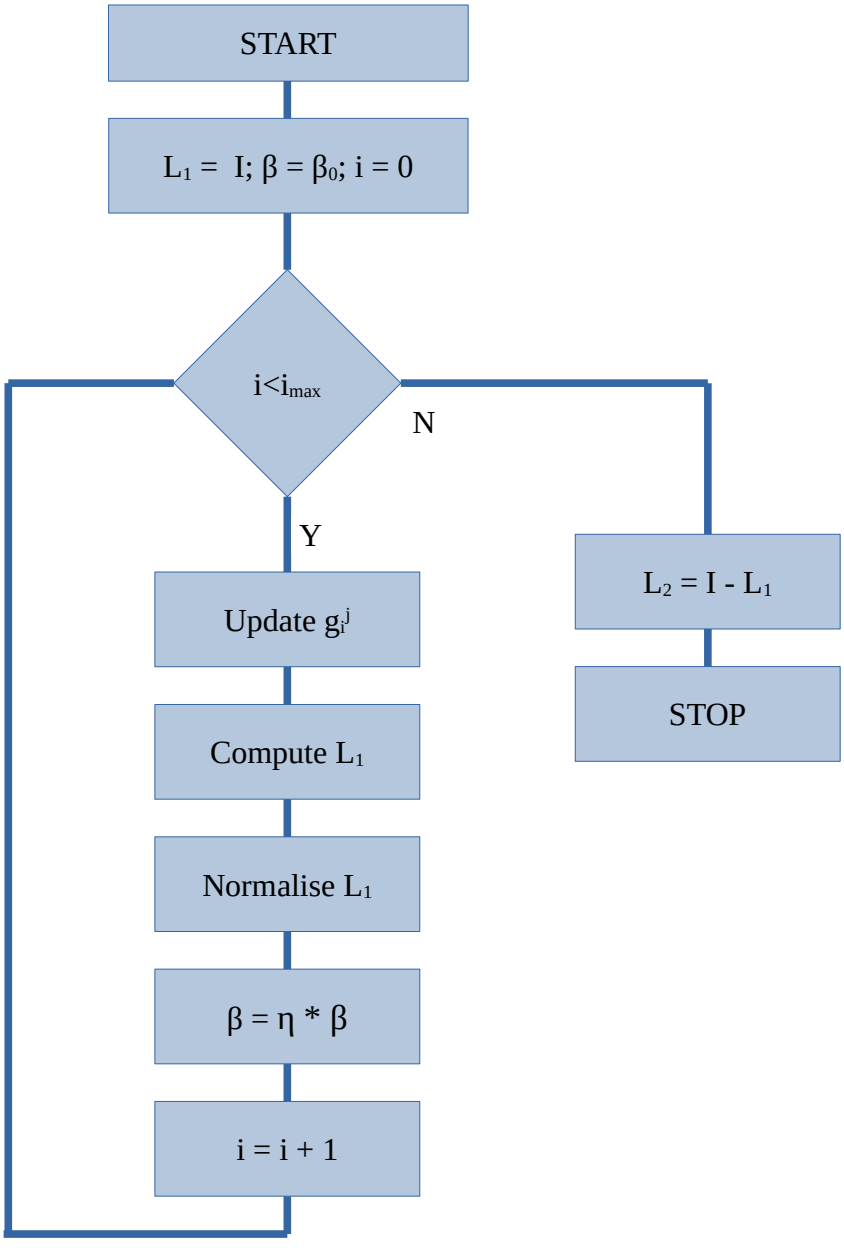
IV. DISTINGUISHING BETWEEN THE OPERATIONS:

To separate the two layers for intrinsic decomposition/reflection removal, the algorithm (explained in the next section) stays the same but the boundary conditions are changed.

For intrinsic image model, the original images are normalised to [1/256,1], therefore after log, the boundary of the reflectance layer becomes **[log(I),0]**.

For reflection removal, the estimated background value should fall in the range **[0,I]**.

V. ALGORITHM:



The above flowchart illustrates quite well how the algorithm works. The following section shows how to carry out the above algorithmic processes.

VI. FORMULAE:

I = input image; λ = Smoothness Weight; β_0 = Initial weight ;i_{max} = Iterations Number; η = Increasing rate of the weight β;

- Update g_i^j:

$$g_i^j = \begin{cases} F_i^j & (F_i^j L_1)^2 > \frac{1}{\beta} \\ 0 & otherwise \end{cases}$$

$$F_i^j = (L * f_j)_i$$

$$f_1 = \begin{bmatrix} -1 & 1 \end{bmatrix}$$

$$f_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$f_3 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

- Compute L_1:

$$L_1 = \mathcal{F}^{-1}(A)$$

$$A = \frac{\beta \sum_j (\mathcal{F}(F^j)^* \mathcal{F}(g^j)) + \lambda \mathcal{F}(F^3)^* \mathcal{F}(I)}{\beta \sum_j (\mathcal{F}(F^j)^* \mathcal{F}(F^j)) + \lambda \mathcal{F}(F^3)^* \mathcal{F}(F^3) + \tau}$$

- Normalise L_1:

$$min_t \sum_i m_i ((L_1)_i + t - lb_i)^2 + \sum_i n_i ((L_1)_i + t - ub_i)^2$$

In the above formulae, $\tau = 10^{-16}$ required to increase the stability of the algorithm. $[lb_i, ub_i]$ are boundaries which satisfy the condition mentioned in section IV.

For intrinsic image decomposition, **$[lb_i, ub_i] = [I_i`, 0]$** (Where $I_i` = \log(I_i)$)
For reflection removal, **$[lb_i, ub_i] = [0, I_i]$**

VII. RESULTS:

The above algorithm was implemented in MATLAB[3] (which has been uploaded with this report at the time of submission). For intrinsic image decomposition, the image has been taken from the MIT Dataset[2] . For the reflection removal, the image was generated by a MATLAB code written by us.

Intrinsic Image Decomposition:

Figure 1: Original Image

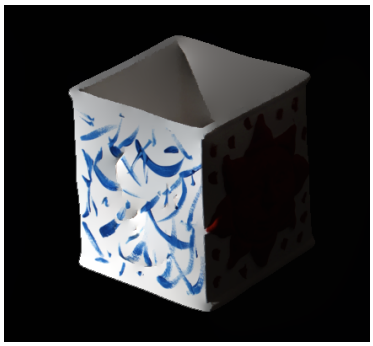


Figure 2: Reflectance Layer

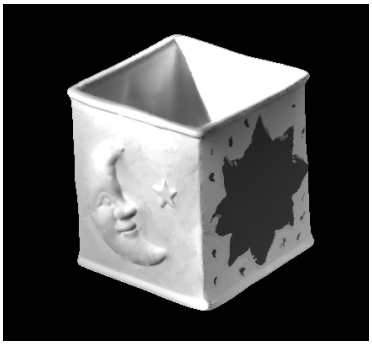


Figure 3: Illumination Layer

Refection Removal:

Figure 4: Original Image

Figure 5: Background Image

Figure 6: Reflection

Figure 7: Tampered Original Image

Figure 8: Background Image

Figure 9: Reflection

VIII. Discussion:

We have implemented the paper Single image layer separation using relative smoothness[1] to verify the sanctity of the paper. The algorithm is quite efficient because it generally takes just about 6-7 iterations to achieve the desired result.

"Relative Smoothness" is a very big part of the entire approach given that we have only 1 known and need to find 2 unknowns, we have to set a prior. And the prior thus set is that the one of the image layers has a smoother gradient distribution when compared to the other layer and thus, two layers can be separated.

For the intrinsic image model, the MIT Dataset helped out in getting hold of a suitable test picture and the program run very smoothly and gave excellent results. The test can be expanded upon the entire dataset and still hold up quite well.

For the reflection removal problem, an extra bit of code was written to mix two different images with one of them being "defocused" using gaussian filter. The image thus obtained was used in the test (as evident in figure 4, where Lena and another lady was mixed to generate the new image)

One of the problems which we discovered with this approach was when the reflection in the image was less defocussed. When the FOV blur becomes less i.e. when the reflection causing object is closer to the object, the program fails to separate the two layers properly. The main reason this happens is because our assumption of one of the layers having smoother gradients breaks down and hence, the entire approach fails. In such cases, a little bit of human guidance might be required. For example, the user can draw something on the layer which is part of the background can help separate the layers(as shown in Fig.7,8,9).

The entirety of the development assets can be found on GitHub[3]

IX. REFERENCE:

[1]Y. Li and M. S. Brown, "Single Image Layer Separation Using Relative Smoothness," 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2752-2759
[2]http://people.csail.mit.edu/rgrosse/intrinsic
[3]https://github.com/jaybee-117/Image_Processing/tree/main/IP_Mini_Project