REPORT

COURSE: DIGITAL IMAGE ANALYSIS (COL 783)

ASSIGNMENT -3

Submitted by

Rohit Katariya -2020CSZ8844

Geetha Pavani Pappu -2020EEZ8644

The Assignment is an implementation of the articles “Seam Carving for Content-Aware Image Resizing” and “Fast Seam Carving Using Gaussian Pyramid”.

Contents of Report

INTRODUCTION

IMPLEMENTATION

Part-1

Quantization

Popularity algorithm

Median Cut algorithm

Floyd Steinberg dithering

EXPERIMENTS & RESULTS

Varying the parameters

Comparison of Quantization methods

Part- 2

Color Transfer

Global Image Matching

Swatches

EXPERIMENTS & RESULTS

Varying the parameters

Comparison of Color Transfer Methods

# INTRODUCTION:

This Assignment we are using seam carving method to resize the image. Since scaling is not considering the image content, it applies uniformly all over the image. This makes scaling an inefficient method. Cropping on the other end considers the image content but it can only be applied on the periphery of the image which makes it difficult to apply where redundant part is not at the center of the image. We also apply fast seam carving method using gaussian pyramids. We also explore the object removal from the image which also works very well. Different energy functions were tried and entropy was working the best.

Orignal image -> Scaling image -> cropped image -> Seam carving

Object remoal



Original Image



Cropped Image



Rescaled Image



Seam Carving Resize

# IMPLEMENTATION:

The assignment is implemented in python, Open CV, numpy and scipy.

## Seam Carving Image scale down by m rows, n columns

Steps followed:

1. Input m, n
2. Converting to grayscale
3. Apply energy function
4. Find seams with dynamic programming in both horizontal and vertical direction
5. Select optimum minimum energy seams from both directions
6. Remove minimum energy seams

## Fast Seam Carving

1. Create different layers of Gaussian pyramids from original imput image
2. Estimate the number of seams to be removed from each layer of the pyramid
3. Consider the lowest resolution layer of Gaussian pyramid
4. Apply energy function and seam carving
5. Transform the seams got in the layer *i+1*and transfrom them to the layer *i* and remove the transformed seams from layer *i.*
6. Apply seam carving for the remaining seams which were calculated in step 2.
7. Repeat the process to all the layers

Eg:

Lets say the image size was 100\*100 and number of layers to be 3. 50 seams to be removed.

So from step 2 we need to remove [ 0 1 12] seams from every layer.

## Energy Functions

We looked at 4 energy functions:

1. Gradient energy e1
2. Sobel energy :
3. Entropy:
4. HOG:

# Parts left:

# Seam insertion

In this part, we apply seam insertion i.e. increasing the width and height of the image.

1. Suppose we are to add *m* rows and *n* columns
2. Here we start with the original image and find a sequence of seams to be removed for removing m rows and n columns.
3. Once we know the sequence of removal, we can insert seams to the original imaga at the same places where we found the lowese energy seams removed.
4. We also tried with using 50% removal twice.

# Object removal

Here the aim is to remove object using seam removal.

1. To achieve this, first step is to take input from user the parts of image to be removed.
2. We set the energy of this part as minimum energy
3. Then we apply seam removal algorithm, which will choose the best horizontal or vertical seam to be removed.

# Object Preservation

Some parts of image are too important to be modified eg. face of a person. So this part saves this part of the image from being modified in seam removal.

1. To achieve this, first step is to take input from user the parts of image to be preserved.
2. We set the energy of this part as maximum energy
3. Then we apply seam removal algorithm, which will choose the best horizontal or vertical seam to be removed.

# Experiments and Observations

# Seam removal:

|  |  |  |
| --- | --- | --- |
| Original Image | Seams image | Result |
| 1 | 1_20210513_1937_seams_image20_c0 | 1_20210513_1937_sizedown_r20_c0 |
| 6 | 6_20210513_1937_seams_image0_c30 | 6_20210513_1937_sizedown_r0_c30 |
| 3 | 3_20210513_1933_seams_image0_c30 | 3_20210513_1933_sizedown_r0_c30 |
| 4 | 4_20210513_1939_seams_image0_c30 | 4_20210513_1939_sizedown_r0_c30 |

|  |  |
| --- | --- |
| Input | Result |
| 2 | 2_20210513_1942_sizedown_r10_c30 |
| 3 | 3_20210513_1927_sizedown_r30_c30 |
| 5 | 5_20210513_1942_sizedown_r20_c10 |
| 14 | 14_20210513_1946_sizedown_r30_c50 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Original Image | Gradient | Sobel | Entropy | HOG |
| 2 | 2.jpggen_energy | 2.jpgsobel_energy | 2.jpgentropy_energy | 2.jpghog_energy |
| 6 | 6.jpggen_energy | 6.jpgsobel_energy | 6.jpgentropy_energy | 6.jpghog_energy |
| 12 | 12.jpggen_energy | 12.jpgsobel_energy | 12.jpgentropy_energy | 12.jpghog_energy |
| 14 | 14.jpggen_energy | 14.jpgsobel_energy | 14.jpgentropy_energy | 14.jpghog_energy |

# Seam insertion

|  |  |
| --- | --- |
| Input image  3 | Insert 30 vertical seams  3_20210503_1202_finalinsert_r0_c-10 |
| Insert 10 rows  3_20210503_1215_finalinsert_r-10_c0 | Insert 30 rows remove 20 columns  3_20210503_1229_finalinsert_r-30_c20 |
| Insert 40 rows 30 columns  3_20210503_1243_finalinsert_r-40_c-30 | Remove 30 rows 30 columns  3_20210513_1927_sizedown_r30_c30 |

# Compare enrgy function outputs-

|  |  |  |
| --- | --- | --- |
| Object | Seam Removal using different functions (85, 89) | |
|  |  |
| Gradient Energy |  |  |
| Sobel |  |  |
| Entropy |  |  |
| HOG |  |  |

Object Preservation

Saving the interviewee in the following image

|  |  |
| --- | --- |
| Without Mask | With mask |
| 80 | Screenshot from 2021-05-13 01-36-04 |
| Screenshot from 2021-05-13 01-29-10 | Screenshot from 2021-05-13 01-31-36 |
|  | Screenshot from 2021-05-13 01-49-01 |

Object Removal

|  |  |
| --- | --- |
| Original image  9 |  |
| Mask without object preservation  masked_image_9.jpg_20210513_1354_0 | Mask with object preservation  masked_image_9.jpg_20210513_1319_0 |
| Result  9.jpg_20210513_1354_0 | Result  9.jpg_20210513_1319_0 |

|  |  |
| --- | --- |
| Original Image  12 | Object image  object_mask_remove12.jpg_20210513_1444_0 |
| Mask image  masked_image_12.jpg_20210513_1444_0 | Result  12.jpg_20210513_1444_0 |

Observations:

We see that the global matching algo prefrorms well when there are not too many colors in the image. Whereas the swatch based matching is able to do much better because it is gets user’s input telling from what part of the image it should pick the colours.

With swatches, we are able to cover images with wide range of colours but we could find that image is getting distinct swatch boundaries which are not looking that good.