COL783: Digital Image Processing

Assignment 3: Seam Carving

Introduction

In this assignment, you will design a seam carving algorithm. Seam carving is an algorithm for content-aware image resizing. It functions by establishing a number of seams (paths of least importance) in an image and automatically removes seams to reduce image size or inserts seams to extend it. Seam carving also allows manually defining areas in which pixels may not be modified, and features the ability to remove whole objects from photographs.

Part 1: Implementing a classical Seam carving algorithm

Refer to the course webpage for part 1 of the assignment.

Part 2: State of the Art Seam Carving Algorithm

In this part, you can use any state of the art Deep learning based Seam carving algorithm. You should fill the following google form in order to indicate the choice of algorithm, along with its base paper.
You are informed to use the following link to access the google form: `https://docs.google.com/forms/d/e/1FAIpQLSfqwc_XhyLsgJ-wDWDpWwDeQNbHgoc8rbZmRi6YmB2Z80cjgw/viewform?usp=sf_link`
No two teams shall implement the same algorithm. Please note that filling out the google form does not guarantee the algorithm. You can consider that the algorithm is blocked in your name only after it gets reflected in the following excel sheet.
`https://docs.google.com/spreadsheets/d/1l0oPCNcFg_hvZEt5T1RkjPyDO9q0P1pGsjyFqGG67tU/edit?usp=sharing`.
Also, please note that any algorithm published before 2019 can not be considered a state-of-the-art algorithm. You should first reproduce the numbers as seen in the base paper. A clear justification on choice of algorithm is required.

Dataset Description

Use the link: `https://csciitd-my.sharepoint.com/:u:/g/personal/anz197518_iitd_ac_in/Efvl0H-hlWxBqJyMZGPHb4EBJnRVfXs-6RcIOSZ2URm-WA?e=SDfS7q` to access the dataset required to test your algorithms of this assignment.
It contains a folder called "Seam_carving_input". Run the part 1 of the assignment (remove 25% of the seams from both the directions.) on each of the 300 images present in this folder. You now have 300 image to image correspondances. In this part of the assignment, we wish to learn this correspondance.

You can use transfer learning (freezing weights of some layer) on pretrained weights. Perform the following experimentation by using Adam as an optimizer and by using RMS-prop as an optimizer

You should now test each of the algorithms of part 2 on the dataset provided and report the similarity between the Ground truth seam carved images (as generated by the classical image processing based algorithm) and the predicted seam carved image (as generated by the deep neural network) using the quantitative parameters such as Root Mean Squared Error (RMSE error), Normalized Mutual Information (NMI), Structural Similarity Index Metric (SSIM). You can use inbuilt libraries to compute these quantitative analysis. In addition to that, you are supposed to report the following

1. A brief description of the training strategy, training dataset, including the architecture, loss function, data preprocessing methods, and data augmentation techniques.
2. Learning rate
3. Number of epochs used for training
4. Batch size used for training
5. The optimizer used for training and optimization hyper-parameters
6. Plot indicating the value of loss functions over different epochs

Note

- The assignment can be accomplished in a team of two or fewer.
- You are informed to fill the google form indicating the team details and choice of algorithm on or before **12th November 2023**. Only one person shall fill the google form and will be considered the team leader. He/she shall be responsible for the consent of the other team member to be a part of that team. Note that the team once formed, can not be dissolved or changed.
- You are supposed to submit the final code-base, trained model, final report consisting of all the details of the proposed solution for this assignment.
- The training code and testing codes are to be maintained separately. This means you will have to save the trained model after training and call it in the testing code. The testing code should take path to the image folder as input and save the inpainted images in another folder as output.
- You will have to submit a separate code to evaluate the performance of the system. This code should take the paths to the folders containing ground truth images and inpainted images as input and output the results (NMI, SSIM, PSNR).
- We would like to create an executable for both "testing" and "evaluation" scripts in all the parts. You are expected to use pyinstaller (`http://www.pyinstaller.org/`) to generate an executable.

  A tutorial for the same can be found at: `https://datatofish.com/executable-pyinstaller/`

  Additionally we expect to compile the code ourselves and generate the executable. Hence, you are supposed to create a makefile for each of them. makefile must also install all the necessary packages.

  i.e. We expect a separate "makefile" for each of the parts. We shall run the command "make" in your folder. It should install all the necessary packages and create two executables, namely "test" and "evaluate". The executable "test" must take path to "test images" folder as input and generate another folder named "inpainted images" with all the inpainted images as output. Similarly, the executable "evaluate" must take the path to both the folders (ground truth images and inpainted images) as input and save a text file contaning all the required evaluations (NMI etc) as output.
- Any sort of plagiarism will lead strict punishments depending on the case.