Georgia Tech's Computational Photography Portfolio

Soham Ghormade

soham.ghormade@gatech.edu

Assignment #1: A Photograph is a Photograph

- **Purpose**
  - The goals of this assignment were as follows:
    - know the technical specifications and settings of your camera,
    - identify a Computational Photography process which can be used to improve a photograph of your choice.
    - Get used to the process of submitting assignments on Canvas for this course.
- **Experience**
  - I intentionally chose a poorly shot photo so that it could be improved.
  - Specifically,the difference in the contrast between the background and my face in the foreground can be improved.
  - I looked up tone mapping[1]which could be used to improve the the photograph I chose.

Assignment #1:Results

Hells Hollow waterfall, Portersville, PA, USA

Exposure Time (seconds)	1/60
Aperture (f-stop, i.e. f4.0)	*f*/2
ISO value	100

Table 1:Settings of the smartphone camera used to take the photo

Assignment #2: Epsilon Photography

- **Background**
  - In a layman’s language,epsilon photography is a form of computational photography in which only *one* parameter changes throughout the image sequence.
  - The camera parameter could be aperture, exposure, focus, film speed or viewpoint.
  - The parameter changes by a small amount ε (hence the name epsilon) in every image.
- **Purpose**
  - The purpose of this assignment ,thus,was to generate a single image artifact which demonstrated the change in epsilon seen in an image sequence.
- **Experience**
  - Ensuring that other parameters remain the same was challenging for this assignment.
  - This challenge was overcome by :
    - trying out different epsilons,and
    - Using a tripod stand to ensure that the camera is held steady while capturing the images
  - For generating the final image artifact,alpha blending using OpenCV proved useful.
  - Essentially,all the images in the image sequence were repeatedly blended to create the final artifact.

Assignment #2:Results

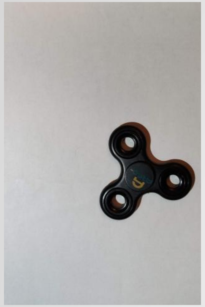


Image 1

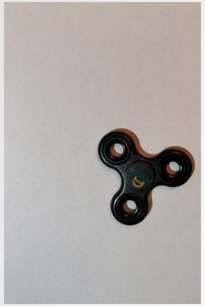


Image 2

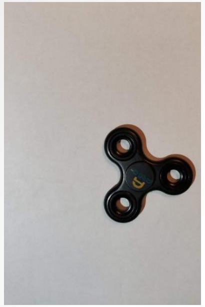


Image 3

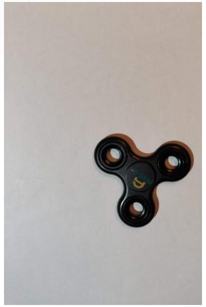


Image 4

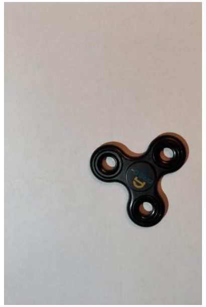


Image 5

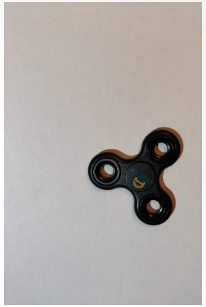


Image 6

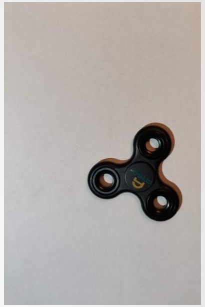


Image 7

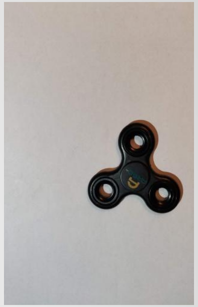


Image 8

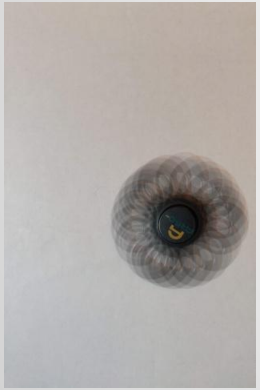

Final Artifact

Figure 1: The image sequence above demonstrates a fidget spinner in motion.The final artifact captures the motion of the fidget spinner in a single image.

Epsilon :Stop motion with a single moving object

Assignment #2: Above & Beyond

Image 1

Image 2

Image 3

Image 4

Image 5

Image 6

Image 7

Image 8

Epsilon :Exposure Value

For the A&B section,I experimented with changing the exposure value for a photograph.The final artifact(GIF) captures the variation of exposure value in a single image artifact.

Final artifact https://drive.google.com/file/d/11o5Ihpa8DVPW5ZotK5F5IC-vK-7Csj3E/view?usp=sharing

Assignment #3: Camera Obscura

- **Background**
  - The camera obscura (commonly known as the pinhole camera)is the forerunner of modern cameras.
- **Purpose**
  - The goal of this assignment was to attempt to understand how the pinhole camera serves as a basic model of all cameras.
  - This goal was achieved by building a room-sized pinhole camera from scratch.
- **Experience**
  - I liked this assignment because it was more hands-on compared to the other assignments in the course.
  - Specifically,I got to study the following things:
    - The effect various of refractory media and ,
    - effect of a moving screen.
  - **Challenges**
    - Capturing the pinhole camera image on a smart phone was a challenge for this assignment.
    - It was a challenge because the photo was to be taken in a room with very little light.
    - This challenge,however,was overcome by experimenting with the exposure time camera setting.
    - Specifically,a longer exposure time helped me capture the image in the dimly lit room sized camera.

Assignment #3: Results

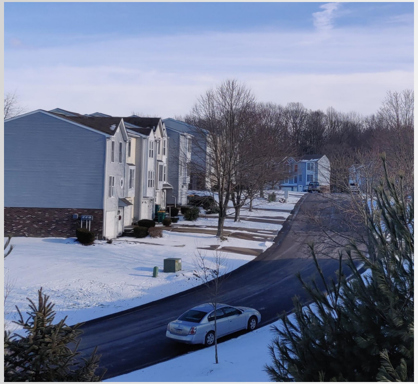


Figure 2:This photo captures the scenery outside my apartment window.

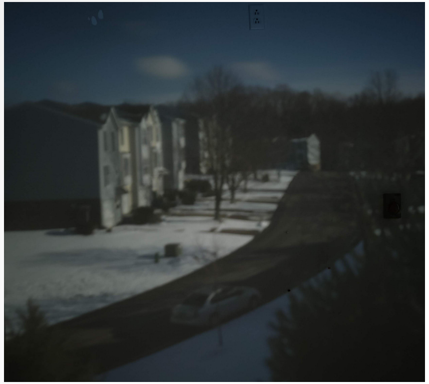


Figure 3:This image is a photo of Figure 2 captured by the room-sized pinhole camera on the wall which which served as a screen.This image was captured on a smartphone camera with 30 seconds exposure time.

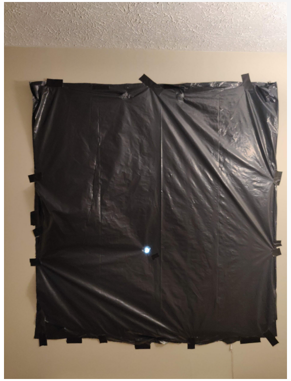


Figure 4:My apartment window was blocked using a opaque sheet to construct the camera.A pinhole was made in the sheet so that the light generates an image on the opposite wall(Figure 3)

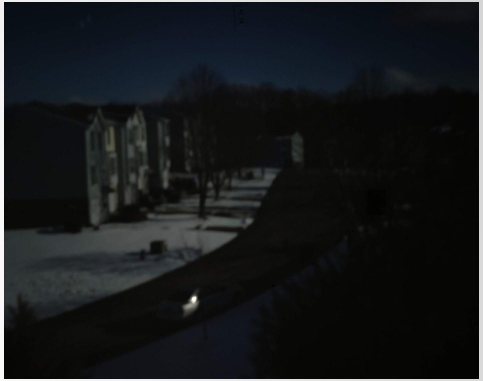


Figure 5:Another image generated by the pinhole camera But this image was captured with 15 seconds exposure time.Longer exposure time (Figure 3)works well for capturing images in low lighting conditions.

Assignment #3: Above & Beyond

- For the A&B part of this assignment ,I studied the following two effects:
- **Effect of using a moving screen(whiteboard)**
  - On reducing the distance between the pinhole and the screen,
    - the image becomes brighter and
    - its area becomes smaller.
  - Refer to pinhole optics light ray diagrams below respective figures for reference.

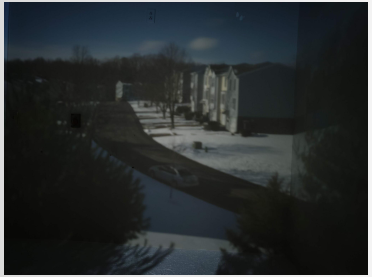
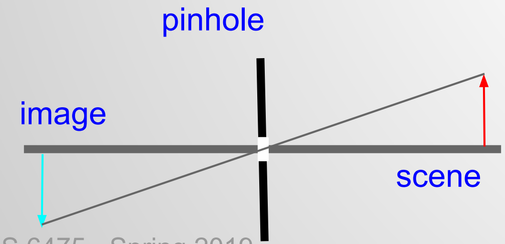


Figure 6: Original image



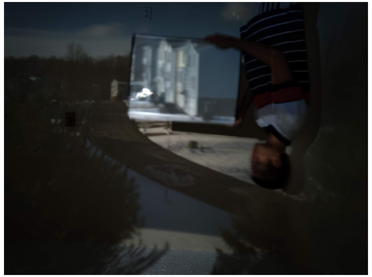
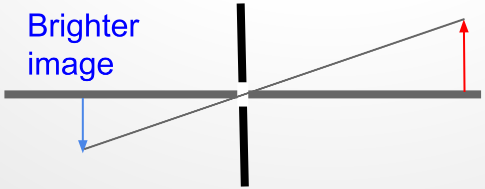


Figure 7: Whiteboard used as a screen at a distance of 60” from the smartphone camera



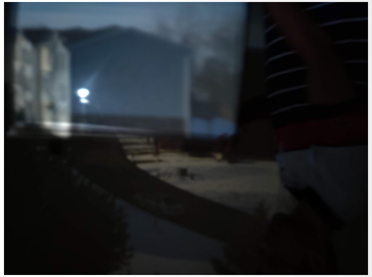
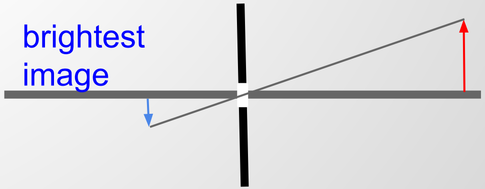


Figure 8 : Whiteboard used as a screen at a distance of 30” from the smartphone camera



Assignment #4:Blending

- **Purpose**
  - The goal of this assignment is to put together a pyramid blending pipeline.
  - This pipeline allows you to combine separate images into a seamlessly blended image.
  - The technique is based on this paper[4]
- **Experience**
  - This assignment was the first coding assignment for this course.
  - I got to learn and experiment with different types of blending in this assignment.
  - Manual mask creation process using GIMP was interesting.
  - Going through the manual process,enabled me to appreciate the importance of automating the mask generation.
  - After learning about edge detection,I noted that Sobel operator could be applied to detect edges and then finally draw a mask based on the edges.

Assignment #4: Results

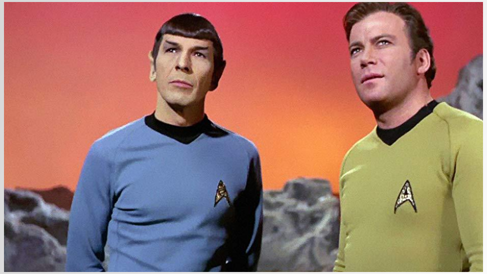


Figure 8:Black Input image.Mr.Spock and Captain Kirk on a foreign planet.

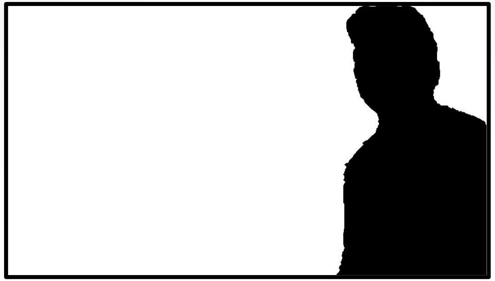


Figure 9:Mask capturing the profile of object of interest(Captain Kirk)

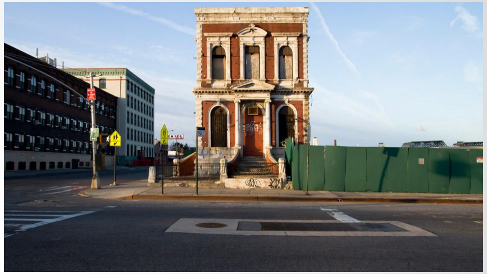


Figure 9:White Input Image

Captain Kirk (Figure 8)was "transported" to Earth using blending(Figure 10).
The mask(Figure 9) was manually generated using GIMP.

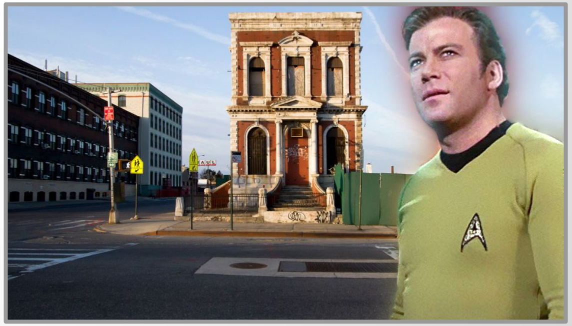


Figure 10:Final Blended Image

Assignment #4:Above & Beyond

- For the A&B part,I experimented with various types of blending operators available.
- Seamless cloning (Poisson image blending)using OpenCV in normal mode(Figure 12)
- worked better compared to mixed mode(Figure 13) and linear blending(Figure 11).

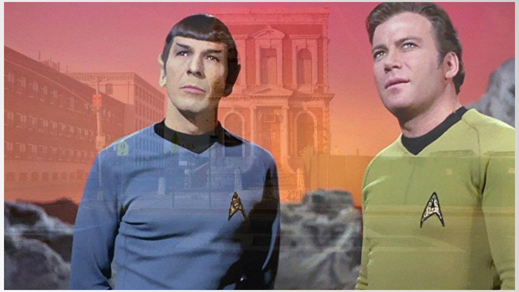
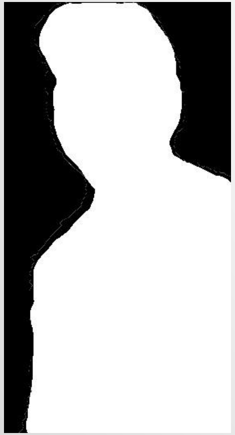


Figure 11: linear blending using OpenCV function addWeighted.

Mask Image.



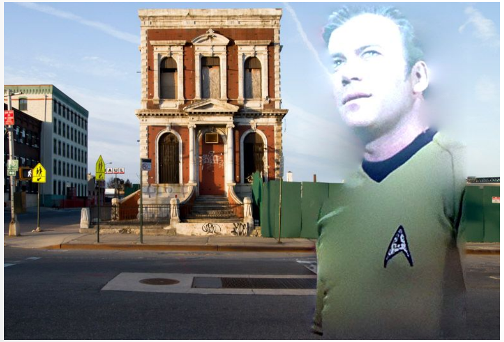


Figure 12:Normal mode seamless cloning using OpenCV

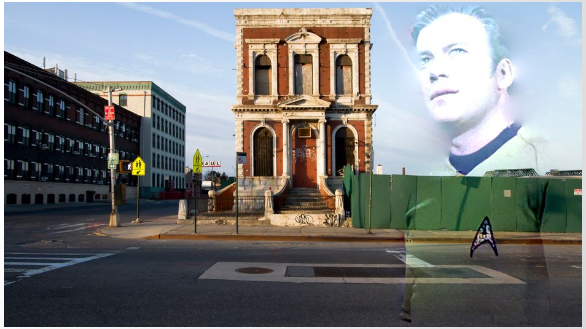


Figure 13:Mixed cloning.This method picks the dominant texture out of the two images.

Assignment #5: Panoramas

Purpose

- This assignment creates the function blocks for building a pipeline which stitch multiple
- images to create a panorama.
- The functions help find features, calculate the homography, warp images from left to right, align
- and then blend the images to create the final artifact.

Experience

- Nowadays,most smartphone cameras have basic functionality to generate a panorama from a set of images.
- However,until before completing this assignment,I was not aware of the details of how panoramas are generated.
- In short,after doing this assignment,I was able to understand the basics of generating a panorama from images.
- In addition,while doing this assignment,I learnt about the following items :
  - Feature detection,and warping for image stitching,
  - Extending alpha blending to warped images,and
  - Automating the mask generation process for image stitching using distance transforms

Assignment #5: Results

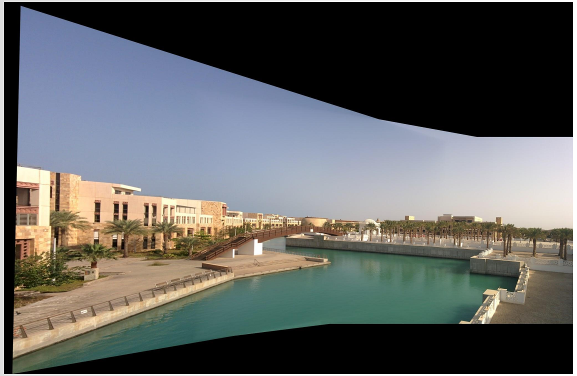


Figure 14: the panorama image generated for the sample images.

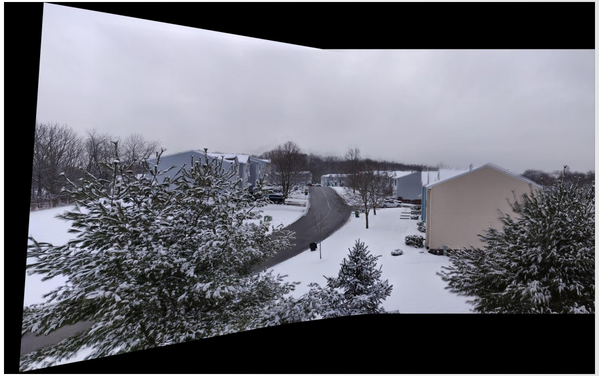


Figure 15 : the panorama view from my balcony window.It was created by combining five images.

Assignment #6: HDR

Purpose

- The goal of this assignment was to combine images of varying exposures to build a composite image.
- This final image captures a wider dynamic range of irradiance.
- The building block functions compute the response curve and radiance map which then are used to generate the final image by using pixels from all the different exposure images.

Experience

- **Linear Algebra**
  - This assignment involved use of linear algebra and hence reviewing numpy functions was useful.
  - Specifically,working through the provided example matrix and reviewing pseudo inverse numpy method documentation helped for this assignment.

- The final output image(Figure 16) was dull because no tone mapping was applied.

- **Tone Mapping**
  - Applying tone mapping via one of the algorithms such as Durand,Drago or Reinhard helps the image get displayed better
  - Comparing my tone mapping implementation results with the results from OpenCV calls to tone methods(Figure 19) was an interesting experiment.
  - The Results and A&B sections of this assignment both discuss this topic.

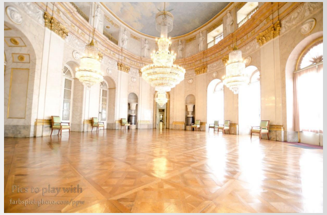
Assignment#6 :Results

“1/2500”

“1/640”

“1/160”

“1/40”



“1/10”

The above image sequence was shot with following camera parameters f/3.5 and ISO 800

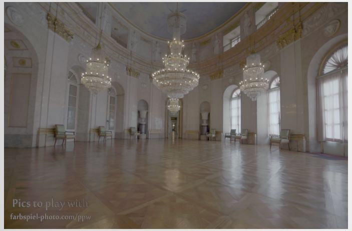


Figure 16:Original output image without tone mapping.Observe the dullness of this image.

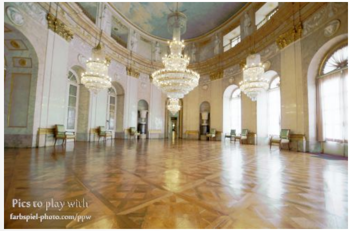


Figure 17: Global Reinhard Tone Mapping operator implemented from scratch applied to Figure 16.

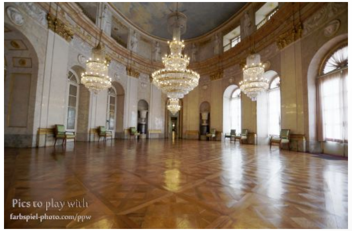


Figure 18:Linear tone mapping applied to Figure 16.Note the richness in tone in Figures 17 and 18 compared to 16.

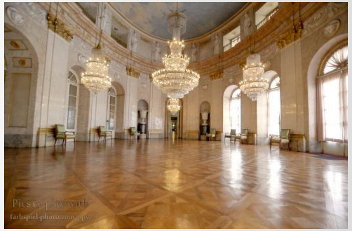


Figure 19: Reinhard Tone Mapping from OpenCV applied to Figure 16.
The image contrast is more richer than Figure 17 and 18.

Assignment#6 :Above & Beyond

- For the A&B part,I experimented with Reinhard tone mapping operator on my own input images.
- For this specific image set,however,the details from the bright outdoors are not retained with the tone mapping operator written from scratch(Figure 21).
- However,the OpenCV function for Reinhard appears to handle the bright outdoors well.(Figure 22)

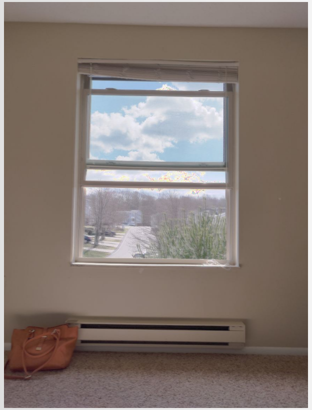


Figure 20: HDR without tone mapping.

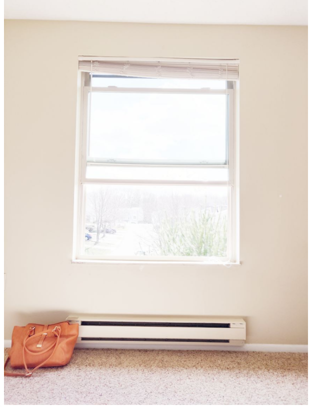


Figure 21 :HDR Reinhard tone mapping(a = 0.09).This is an improvement over Figure 20.Specifically,the color of the bag is retained.However the outdoor scene detail is lost.

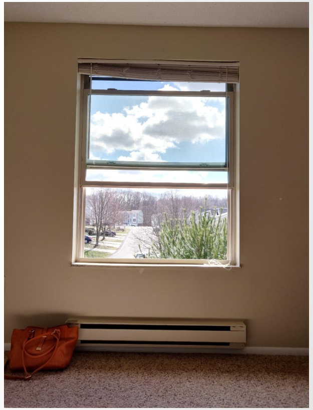


Figure 22 HDR with Reinhard tone mapping using OpenCV function.This method does a better job of retaining the details from the outdoor scene from the window.

Assignment #7: Video Textures

Purpose

- The purpose of this assignment is to identify a part of a video which can generate an infinite loop
- video with minimal or no visible break.
- Specifically, we build a pipeline to compute the similarity matrix between frames of a video and use them to identify and create a video texture.
- The assignment focuses on finding the biggest loop possible from the similarity matrix and generating the infinite looping video.

Experience

- Being the first assignment using a video instead of images, this assignment helped understand how multiple image frames are stacked together to form a video volume.
- Finding the right value of alpha was necessary to create a smooth video texture.
- This part proved particularly challenging for my own input video.
- To overcome this challenge,two things helped:
    - experimenting with different values of alpha
    - Referring to the scoring image (Figure 25 on the next slide)

Assignment #7: Results(1)

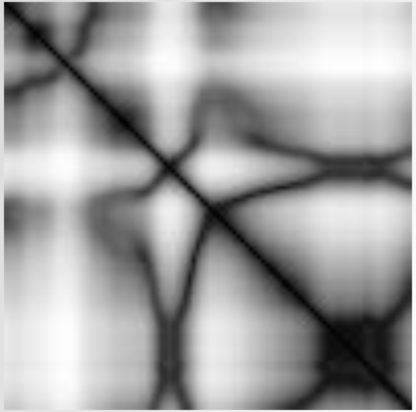


Figure 23: Similarity

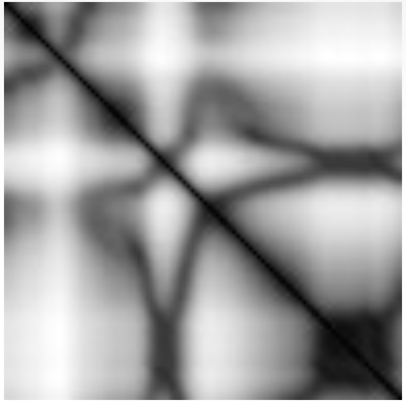


Figure 24: Transition

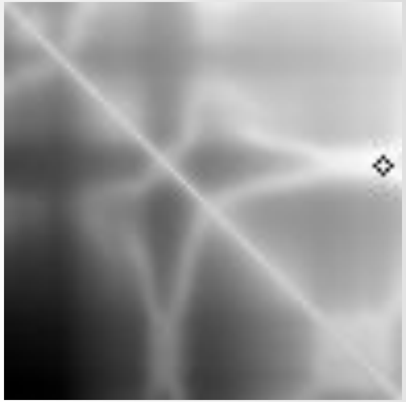


Figure 25: Scoring

α	(start, end)	Comment
2	(2, 97)	The default alpha value leads to a large loop size.Specifically, almost all the 100 frames are selected.
0.02	(39, 91)	The “Goldilocks” alpha value.This results in a smooth video but with a reasonable length.
0.01	(2, 2)	Only one frame is selected.i.e very small loop size.

Table 2::alpha values with corresponding start and end values for sample input frames.

Assignment #7: Results(2)

Candle flame

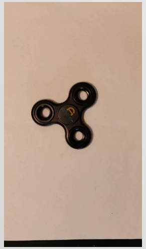
Alpha value for the best loop - 0.02

Link to thecandle video texture gif - https://drive.google.com/drive/u/0/folders/1ah-GIVHPqqOUPhxJ3AZ_Y6e7RHPzS5b3

Ballet dancer

Alpha value for the best loop - 0.0149

Link to the video texture gif - https://drive.google.com/drive/u/0/folders/1ah-GIVHPqqOUPhxJ3AZ_Y6e7RHPzS5b3



Fidget spinner

Alpha value for the best loop - 0.007

Link to the video texture gif(own input video) - https://drive.google.com/drive/u/0/folders/1ah-GIVHPqqOUPhxJ3AZ_Y6e7RHPzS5b3

Midterm Project

Purpose

- The goal of this project was to replicate results of the seminal paper[6] on seam carving.

Experience

- For me,this project was the hardest one as compared to other assignments and projects in this class.
- Learning how to make assumptions when the implementation details are left out was one of the challenges for this project.
- Overcoming this challenge ,prepared me for the final project.
- This is because ,research papers in general,leave out the implementation details and hence the onus is on the reader to figure out these details.
- I overcame this challenge by:
  - using different values and techniques
  - working with smaller input images and writing unit tests

Midterm Project:Results

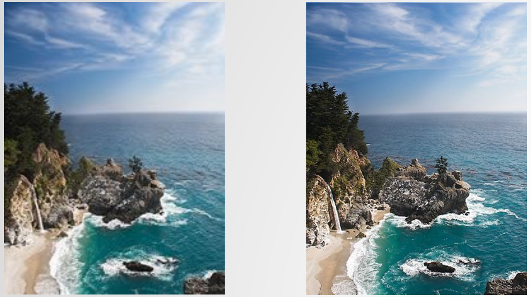


Figure 26 :Seam removal for the beach scene.Authors results provided on the left for comparison

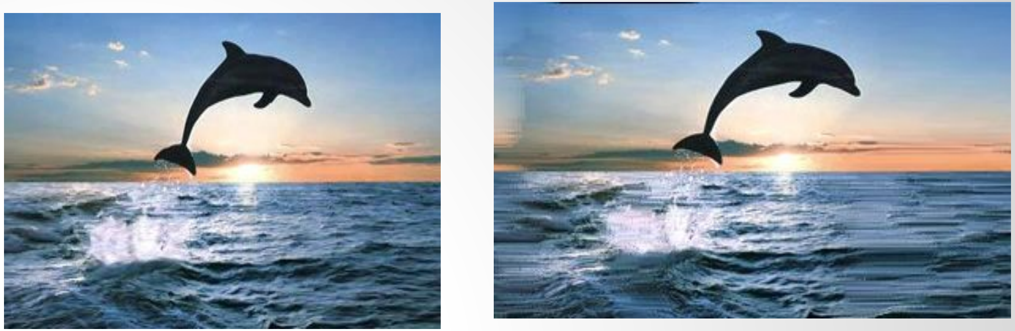


Figure 27: Seam insertion for 50% enlargement on the dolphin image.Authors results provided on the left for comparison

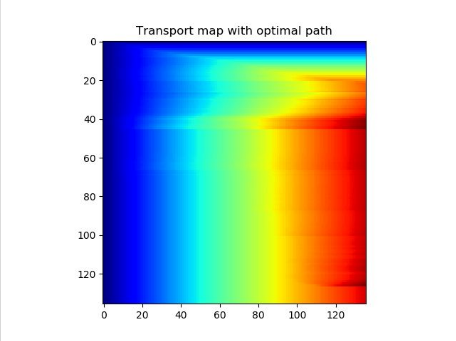


Figure 28 :Transport map for optimal retargeting of flower image[6].

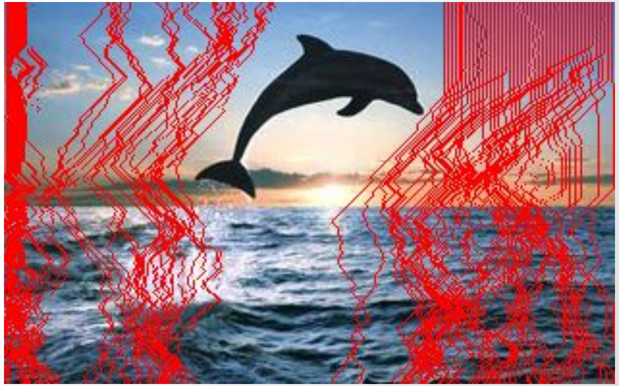


Figure 29 :Seam insertion on the dolphin image with seams shown in red.

Final Project

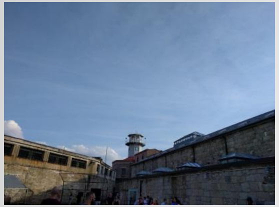
Purpose

- Given a colored image ,convert it into a painting of particular style of your choice.
- Specifically,explore different style parameters and brush stroke types while generating paintings.

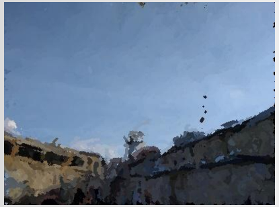
Experience

- I found this project to be fun and interesting.
- I was able to replicate results from a research paper [7] by implementing it in Python.
- Getting the painterly rendering algorithm to work for my own input images was a challenge by itself.
- For this challenge,trying out different styles to find the which worked best,helped in overcoming roadblocks.
- In order to make the project even more challenging,painterly rendering was applied to video as well.

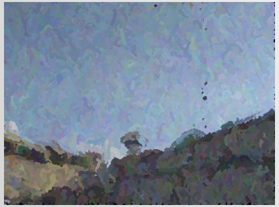
Final Project:Results



Input:Penitentiary in Philadelphia ,USA



Impressionist

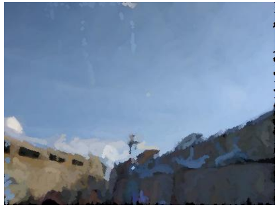


ColoristWash

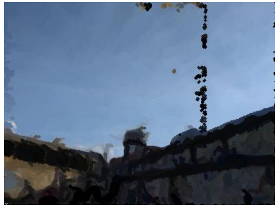
Styles (Radii = 8,4,2,)

Output[11]

Stroke Ordering



Light to Dark

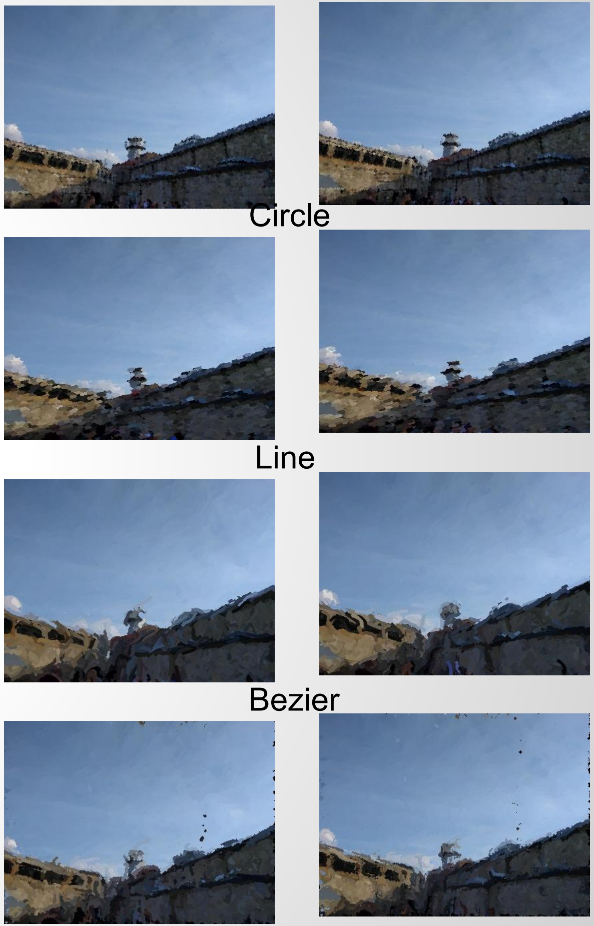


Dark to light

GIMP Filter>Artistic>Oilify output provided for comparison

Stroke Types(Radii = 8.4,2)

Constant Radii Tapering Radii



Circle

Line

Bezier

B-Spline

Project Pipeline for painterly images

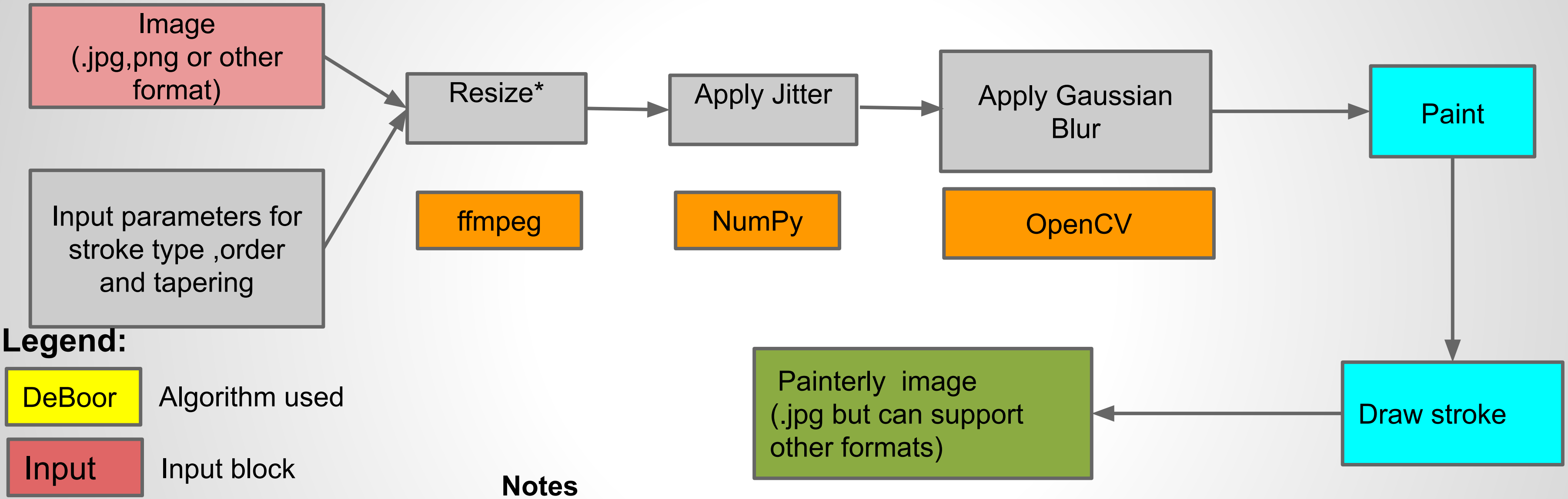


Image (.jpg,png or other format)

Input parameters for stroke type ,order and tapering

Resize*

ffmpeg

Apply Jitter

NumPy

Apply Gaussian Blur

OpenCV

Paint

Draw stroke

Painterly image (.jpg but can support other formats)

Legend:

DeBoor	Algorithm used
Input	Input block
NumPy	Third party code
Draw	Code written from scratch
🟩	Output block

Notes

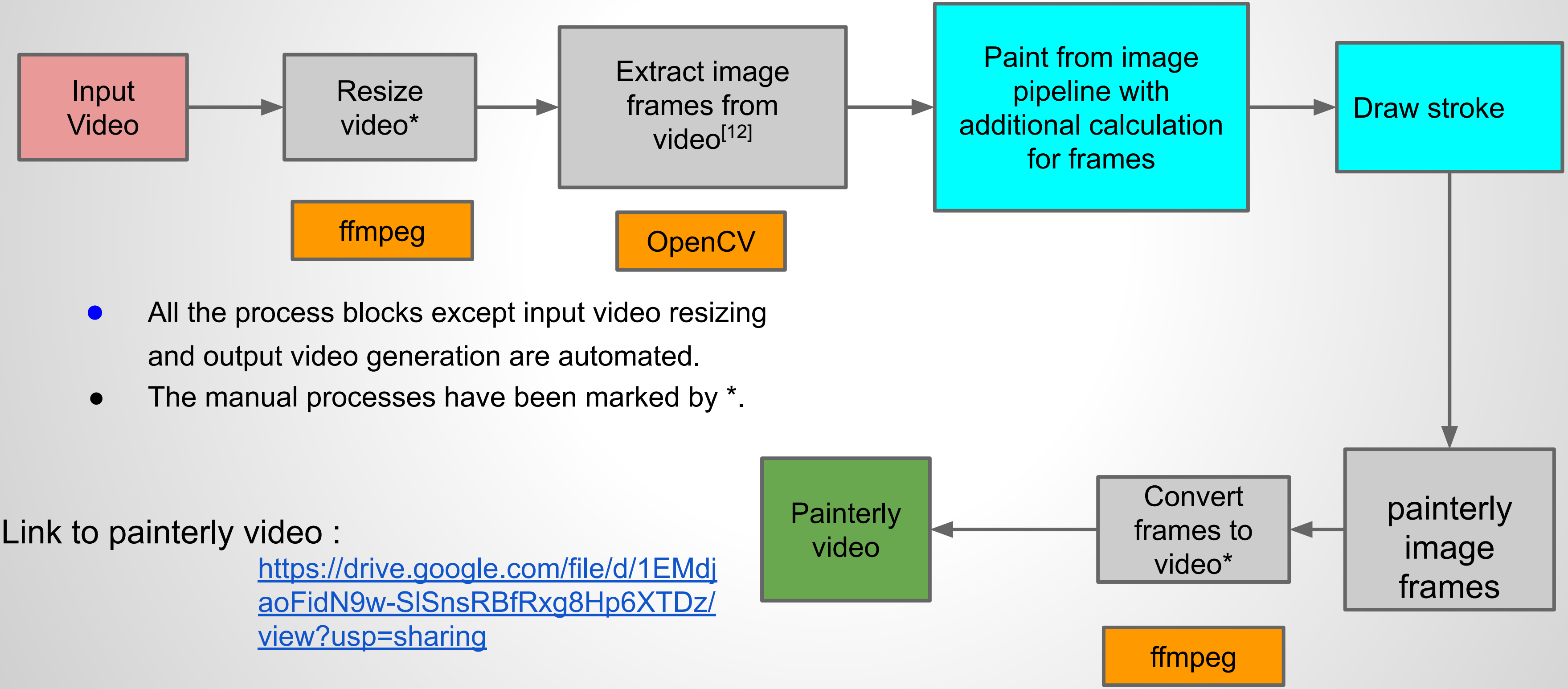
- OpenCV was used for blurring and drawing.
- Numpy was used for array arithmetic and random numbers.
- ffmpeg was used to scale down images for speed up.
- Each of the processes in this pipeline is automated except resizing.
- Manual processes have been marked by *.
- Algorithms were also written from scratch.

DeBoor

Casteleu

OpenCV

Project Pipeline for painterly videos



- All the process blocks except input video resizing and output video generation are automated.
- The manual processes have been marked by *.

Link to painterly video : https://drive.google.com/file/d/1EMdjaoFidN9w-SISnsRBfRxg8Hp6XTDz/view?usp=sharing

References

- [1]Qi Shan, Jiaya Jia, and Michael S. Brown Globally Optimized Linear Windowed Tone-Mapping.TVCG 2010.http://grail.cs.washington.edu/projects/sq_tonemapping/
- [2]CS6475 github repo :https://github.gatech.edu/omscs6475/assignments
- [3] Wikipedia entry for Epsilon Photography :https://en.wikipedia.org/wiki/Epsilon_photography
- [4]Burt and Adelson; ACM 1983 http://persci.mit.edu/pub_pdfs/spline83.pdf
- [5]Seamless cloning using OpenCV :https://docs.opencv.org/3.0-alpha/modules/photo/doc/cloning.html
- [6] Avidan and Shamir (2007), “Seam carving for content-aware image resizing”, SIGGRAPH 2007.
- [7] Painterly Rendering project website (includes test input images) https://www.mrl.nyu.edu/publications/painterly98/
- [8]OpenCV Python documentation :https://docs.opencv.org/3.0-alpha/doc/py_tutorials/py_tutorials.html
- [9] NumPy documentation :https://docs.scipy.org/doc/numpy-1.16.1/
- [10]Numba for performance optimization:https://numba.pydata.org
- [11] Final project output:https://drive.google.com/drive/folders/1Mm61tc7_JTEIEKbLAu-UAuohTICH4Irc?usp=sharing