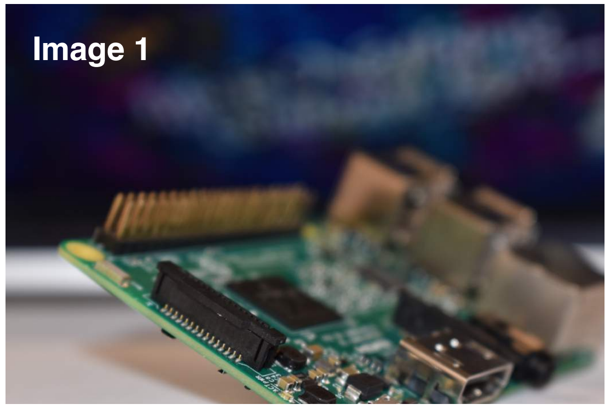
Computational Photography

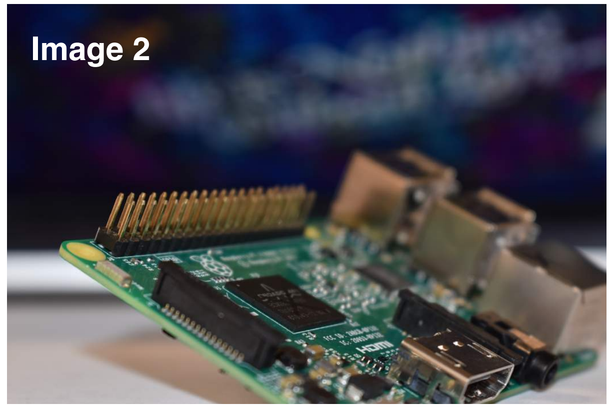
Assignment #2: Epsilon Photography

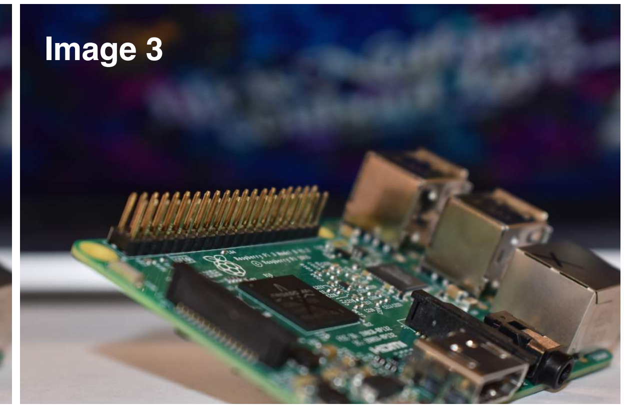
Nitish Sanghi

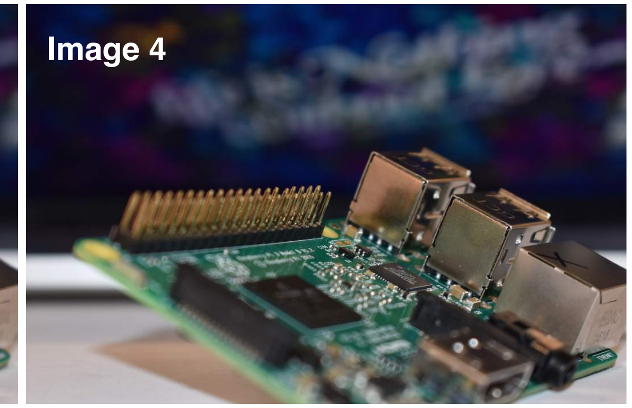
Spring 2019

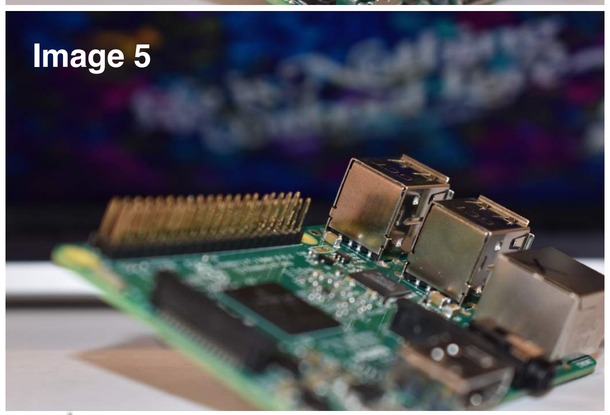
Epsilon Project Overview

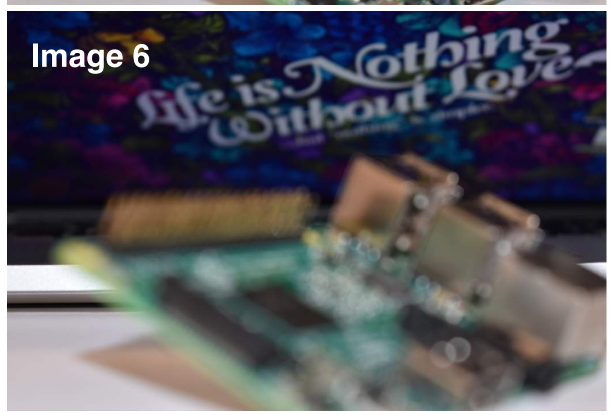


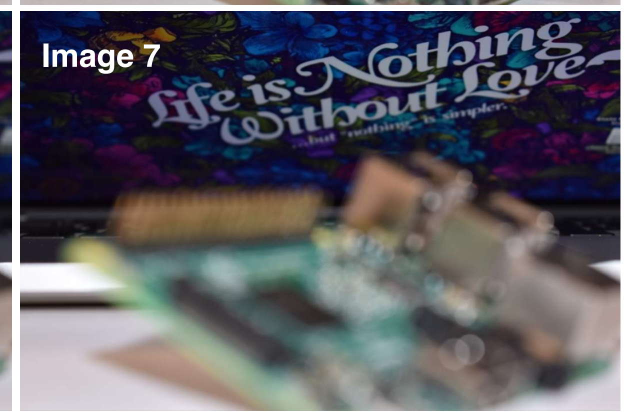












Description:
Include a short description of your project.

The sequence of images capture different features of a Raspberry Pie (RasPi) set up against a background on a MacBook Pro. The RasPi is staged at an angle with a small depth of field. The images capture the blurred/bokeh effect as the focal distance moves. The laptop screen is positioned at a n angle of about 45 degrees from the table surface thus the text is angle into the plane of the slide. Features are captured at different focal distances and the images are digitally processed and combined for focal stacking, resulting in a greater depth of field [3]. The Epsilon parameter is the location of the focal distance which is incrementally changed to capture features at a multitude of locations.

Project Discussion

What is your epsilon parameter?

Focal distance is the epsilon parameter. It is changed to capture information which can be used to make a composite image with high depth of field.

Location of pictures?

Pictures were clicked in at Intuitive Surgical, Sunnyvale California.

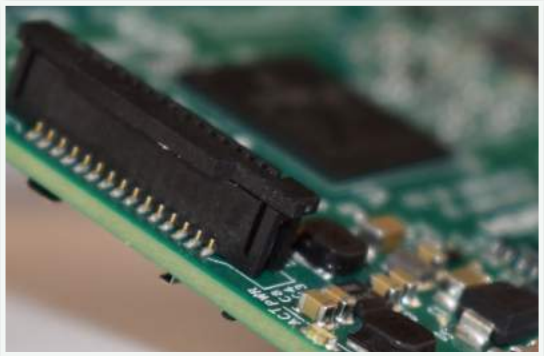
Date and time?

The images were captured on 01/21/2019, between 3:50 pm - 4:00 pm.

How did you control the settings, the environment, and the camera to meet your epsilon requirement?

My primary impetus to pursue this project was to get hands on experience with focal stacking technique with the intention to apply it in the future on the images I capture using digital optical microscopy. I use optical microscopes quite often and a lot of the times I struggle with capturing tiny features in a single frame due to focal distance differences. Digital image post processing to increase depth of field and hence we able to see more features in a single composite frame will be a useful tool for analysis.

Controlling the epsilon parameter for this project required focusing at different features from nearest to the farthest in a static scene and ensuring no movement or jitter in the camera setup. The scene was setup and the camera was fixed at one location. The scene was illuminated by disperse ceiling light. Based on literature review which suggest a low ISO for focal stacking, I fixed the ISO to 400 and the exposure time to 1/40 seconds.I attempted to use a tripod but was not able to keep the camera stable as I manually changed the focus hence I taped the camera to the desk and very gently adjusted focus for each shot. I also used a 5 second time as a mitigation for any vibrations/jitters I might cause as I press and release the shutter button. The thought behind this was that the inertia would most likely damp the jitter within 5 seconds.



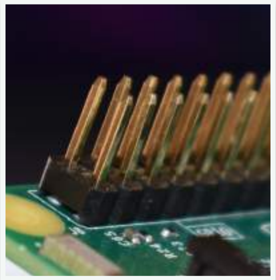
Feature of Focus

Image 1

Description: The camera is focused on the black DSI display connection which is parallel to the edge of the RasPi closest to the camera. This is the first feature captured. As can be see the depth of field is quite small, with only a few resistors on the lower right corner in sharp focus and the rest blurred.

Camera settings:

- Exposure (seconds): 1/40
- Aperture: f5.6
- ISO: 400
- Focus Distance: 0.316



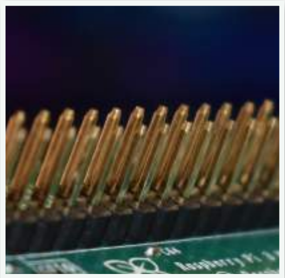
Features of Focus

Image 2

Description: In image 2 the feature of focus has changed from the DSI display connector to the pins of the 40 pin GPIO header. Note only a few pins closest to the camera are in focus in this image. The resistors and DSI connector from the previous image appear to be blurred but not as blurred as the features further back. The large black square is the microprocessor which is almost fully in focus with the edge away from the camera slightly out of focus. Also, on the lower right the HDMI connector is also in focus.

Camera settings:

Exposure (seconds): 1/40
Aperture: f5.6
ISO: 400
Focus Distance: 0.316



Features of Focus

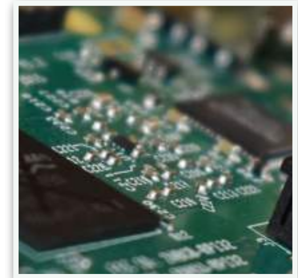
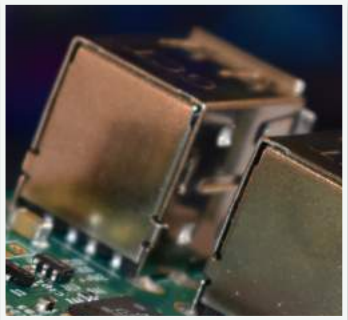


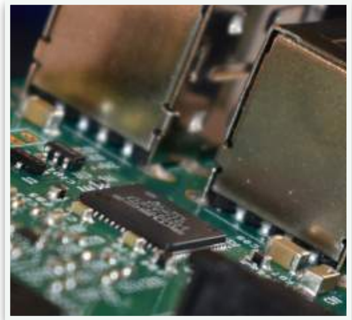
Image 3

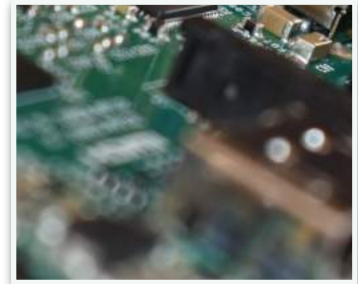
Description: The small measure of the depth of field becomes very apparent in this image. The pins in the center portion of the GPIO header are in focus but pins on either end are blurred and out of focus. Half of the CPU has also become blurred and the tiny capacitors, inductors, and resistors are coming into focus but due to the metallic connections the flash light form the camera seems to be reflecting and gives the appearance of tiny white LEDs. These specular like reflections might cause issues during the computation processing and create unwanted artifacts.

Camera settings:

- Exposure (seconds): 1/40
- Aperture: f5.6
- ISO: 400
- Focus Distance: 0.335







Features of Focus

Bokeh Effect

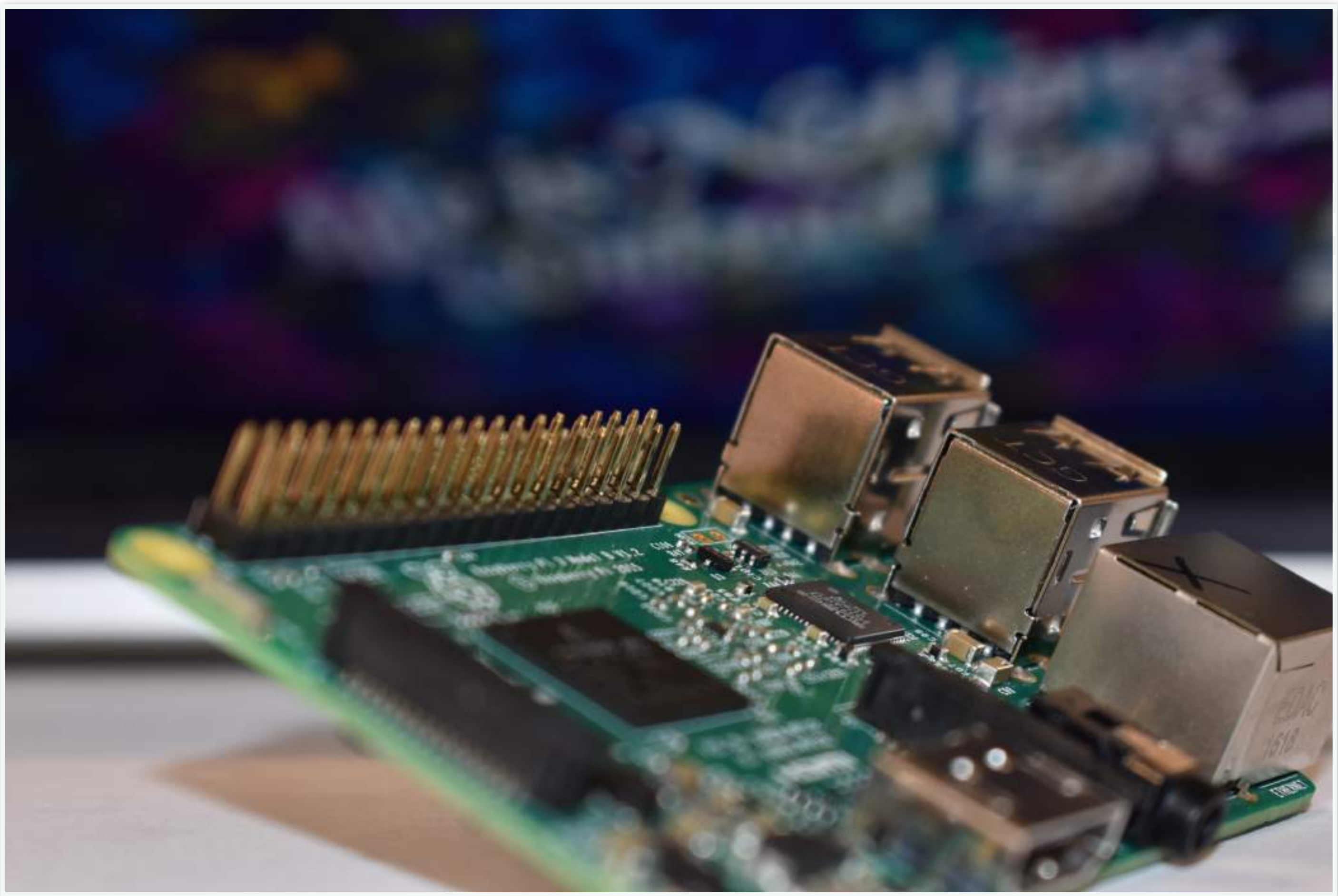


Image 4

Description: Electrical features on the edge away from the camera have come into focus. The large metallic cuboidal structures are the USB ports. It seems due to the surface texture they are not reflecting much light and have yellow - orangish tinge. A small square is not in focus. It is the ethernet controller whose edges are quite crisp in the image. On the lower right corner the green edge of the PCB makes a sharp image. Note the **Bokeh effect** is perceivable from all the specular reflections.

Camera settings:

- Exposure (seconds): 1/40
- Aperture: f5.6
- ISO: 400
- Focus Distance: 0.335

Last RasPi Features of Focus

Description: Image 5 looks identical to image 4, but on further inspection it is observed that the USB port lips on the far extreme make a sharp line. Most of the RasPi is out of focus with the very last features of the RasPi captured in this image.

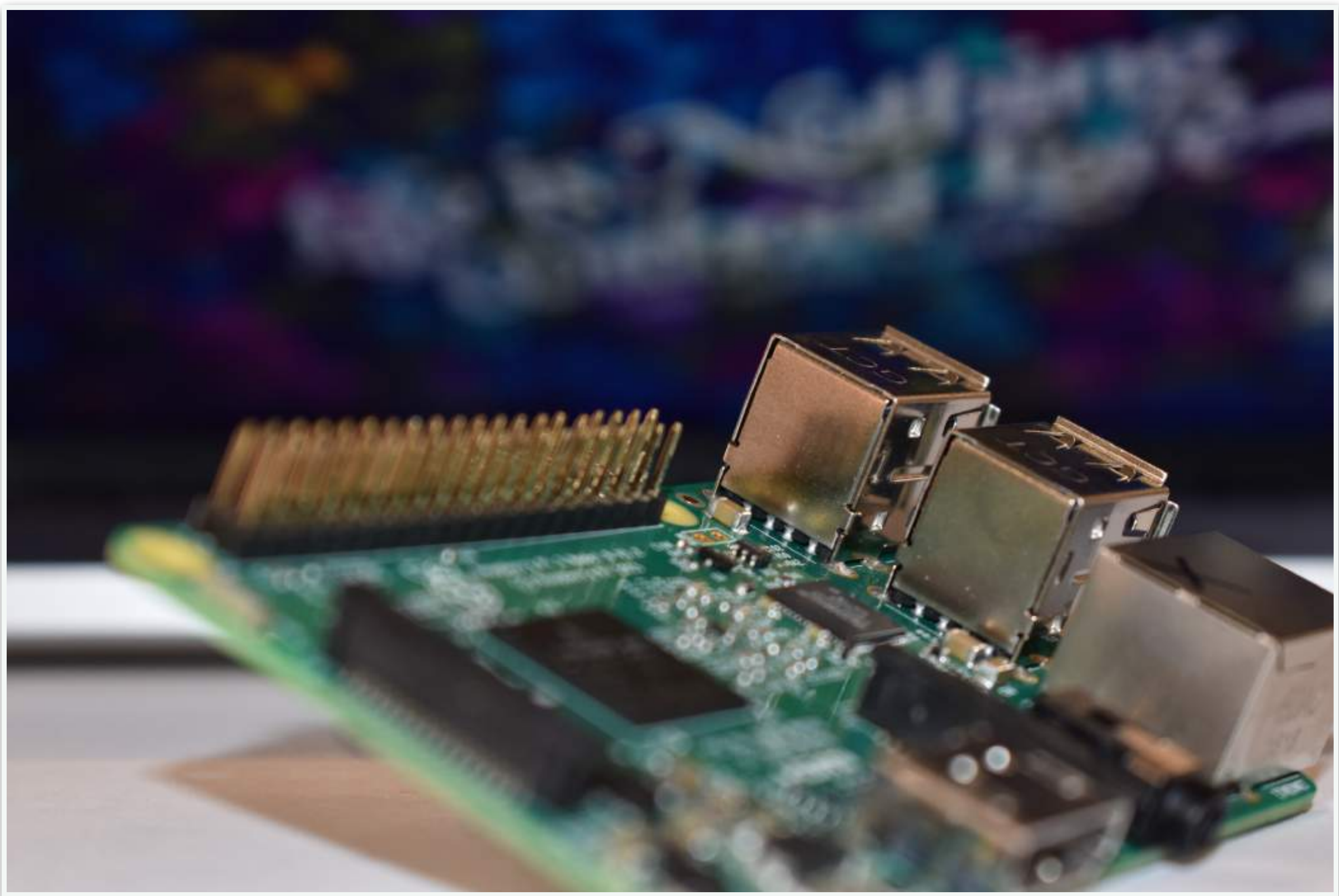


Image 5

Camera settings:

- Exposure (seconds): 1/40
- Aperture: f5.6
- ISO: 400
- Focus Distance: 0.376

Edge of Laptop in Focus

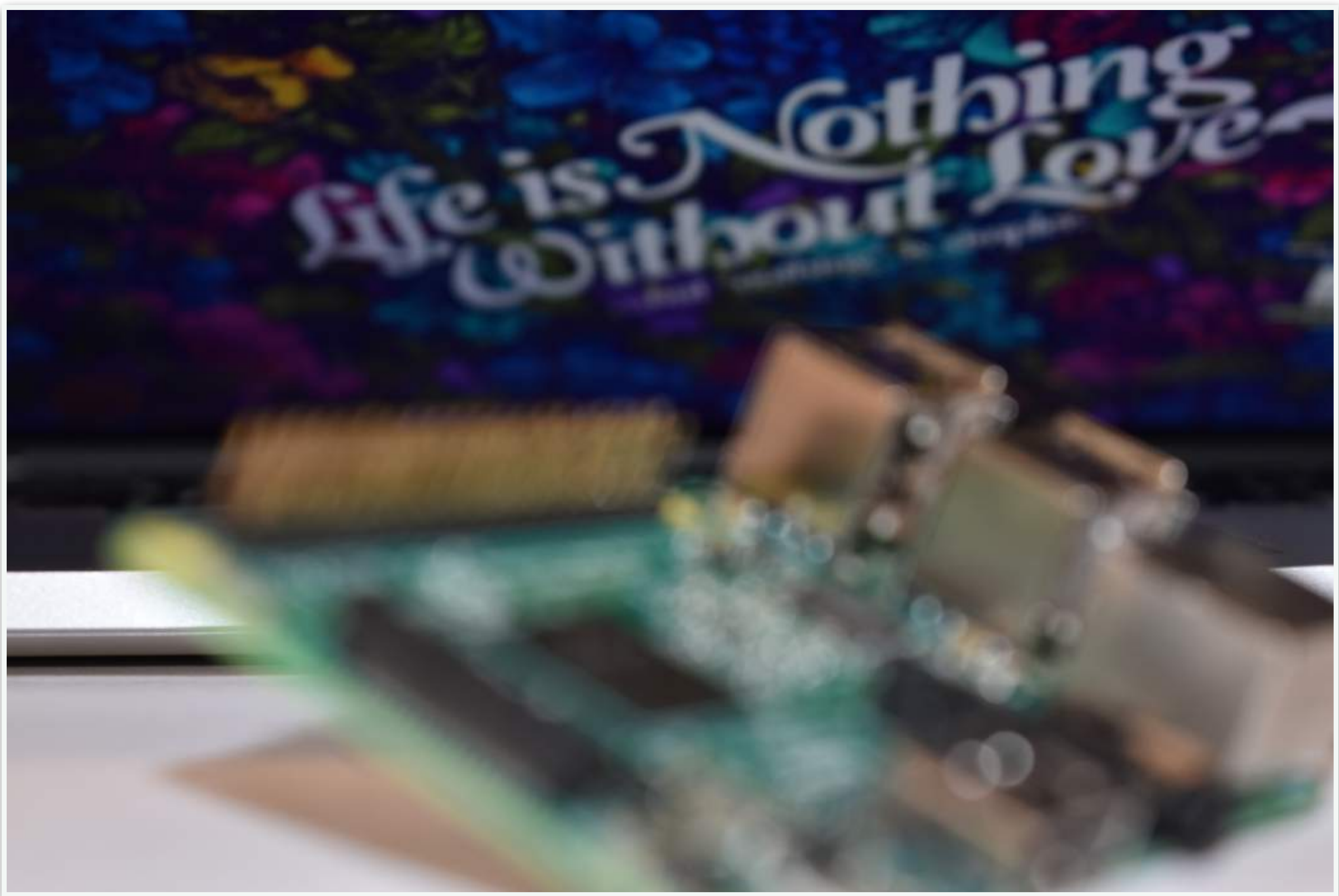


Image 6

Description: This is an important transition image. The RasPi is completely out of focus. On the lower left side of the image, there is a sharp white feature. It is the body of the laptop. The focus has shifted to the laptop and the text on the screen is clearly legible with a slight blur. The edge of the laptop body and the text almost seem to be in the same plane. That is because the laptop screen is bent about 45 degree forward. Due to the angle of the screen the text appears to be squeezed. Note the depth of field appears to be very small in this image compared to other. That is primarily because in the region of the depth of field there aren't many features to focus upon.

Camera settings:

Exposure (seconds): 1/40
Aperture: f5.6
ISO: 400
Focus Distance: 0.473

Text in Focus

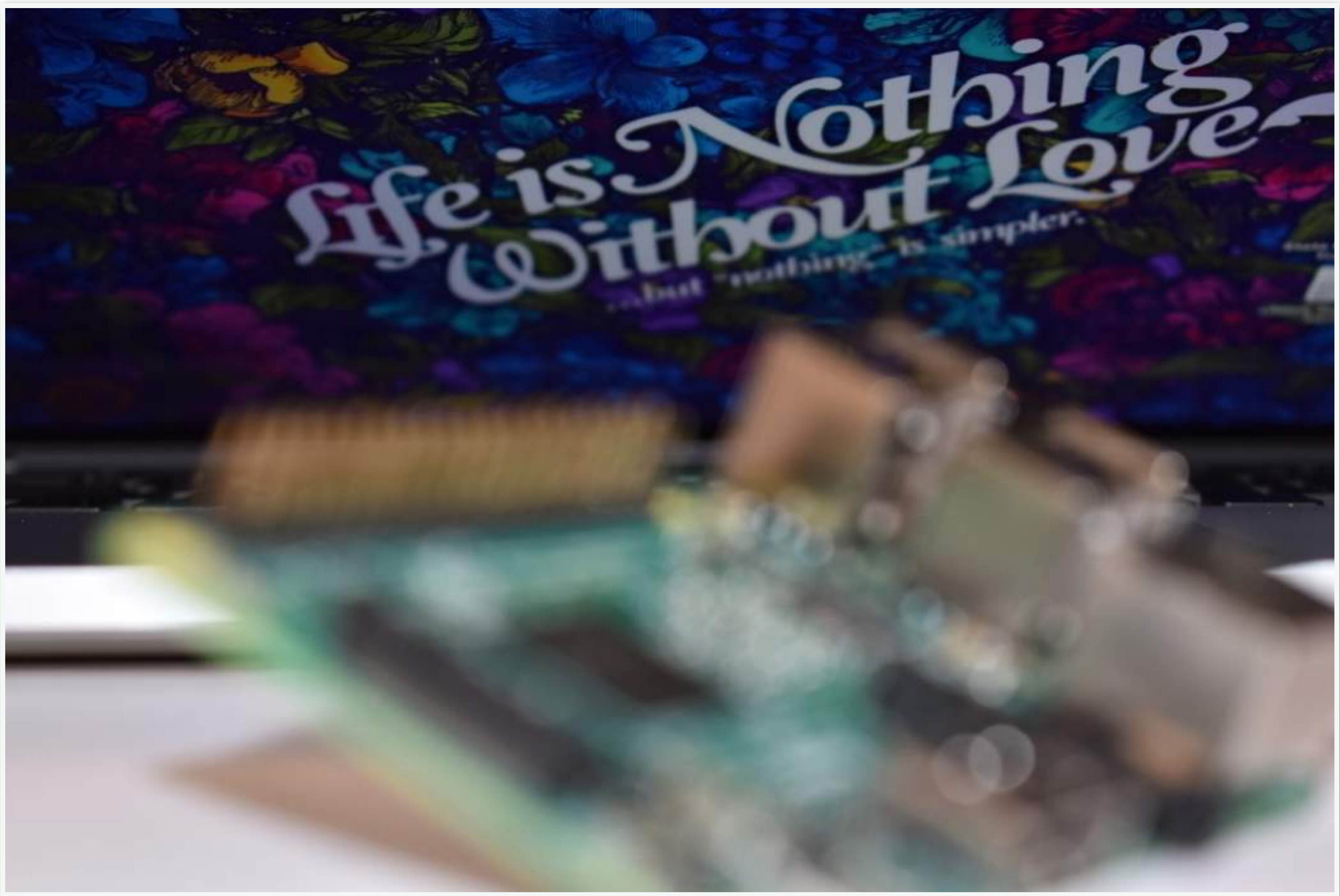


Image 7

Description: This is the final image with the text almost completely in focus. The lower left portion of the text is slightly out of focus. The laptop horizontal edge and the Raspi are highly blurred. This the final feature. The flowers in the background seem to be slightly in focus.

Camera settings:

Exposure (seconds): 1/40
Aperture: f5.6
ISO: 400
Focus Distance: 0.562

FINAL ARTIFACT: EXTENDED DEPTH OF FIELD

Final Artifact Details

Discuss how your final artifact demonstrates your epsilon.

The final artifact is a composite of multiple images being used to increase the depth of field, with each image focusing on feature/features between the nearest and farthest point from the camera in the scene. The epsilon was the focal distance which was incrementally varied while other camera parameters were fixed. As the focal distance varied, pictures of different portions/locations of the static scene were captured with the smaller depth of field. When those images were combined the depth of fields added up and the composite image which is the final artifact had larger depth and field and more details visible with minimal blurring. Epsilon photography requires change in one and only one parameter to get an enhanced image and that was accomplished by varying the focal distance.

Did you do anything to prepare your N images? (e.g. cropping, alignment) If yes, describe your process and mention if you used any image editing software (e.g. Photoshop). If you did not do any processing to your N images beforehand, say so.

Yes, I had to align the images to reduce secondary artifacts. On the first attempt, I found that the composite image had jitter/noise around the features that were in focus. After evaluating raw images and side by side with the processed images and based on literature review [6] I concluded alignment might help improve the final result. I used algorithm for alignment explained in OpenCV tutorial [4]. It uses motion model to estimate 2D or 3D motion/movement and adjust the images with the estimated motion model.

Final Artifact Details

Walk through the code you wrote to create the final artifact. You may provide code snippets to help in your discussion. Do NOT just copy and paste your code with in-line comments. Explain the purpose of the code that you're presenting.

The code is composed of helper functions which breakdown the pipeline into simpler building blocks. The images were held in a list containers for ease of passing as arguments to functions.

Step 1: Transform/align images to account for distortions/misalignment due to focus distance changing.

<u>def transformingimages()</u>

Step 2: Convert images to gray scale and smoothen to reduce noise. GaussianBlur function used from OpenCV.

Step 3: Identify sharp features in each image which will be used to form the composite image. Sharp features will have well defined edges. Image gradients can potentially indicate sharp features. Testing two OpenCV transformations: Laplacian() and Sobel()

<u>def depthoffield()</u> and ***<u>def depthoffieldsobel()</u>***

Step 4: Using Gradient create mask for each image to isolate the sharp regions.

<u>def makingmask()</u>

Step 5: Use mask to create partial images from the color images.

<u>def colorstacks()</u>

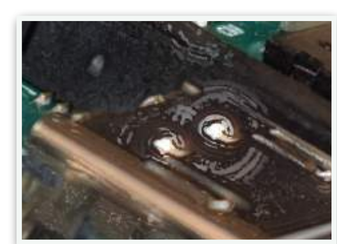
Step 6: Combine masked color images.

<u>def colorfocusstack()</u>

Project Retrospective

- In what ways was your project successful?
- Yes, the project was successful as the desired result of increasing depth of field with focal stacking was achieved. Image processing pipeline is based of simple and very basic ideas for accomplishing the task of focal stacking. The core idea around which the pipeline is built is to identify sharp features/locations on different issues and use those to stitch together a composite image with higher depth of field and details in a single frame. There is scope for improving the pipeline to further improve the composite and get rid of secondary unwanted artifacts.
- If you were to repeat the project, is there anything you do differently knowing what you do now?
- Yes, even though the intended goal was achieved, there are improvements to be made. A number of secondary unwanted artifacts appeared which did not change the end result but do take away the realism from the image. Better control of illumination and camera flash should help in avoiding the specular reflections. The positioning of the camera seems to have led to perceived motion between images as the focus is changed even though the camera was firmly fixed. Changing and trying out different positions to get a good field of view and frame should help.






Secondary Artifacts Left to right: Translucency, mixing pixels, surface texture change, halo effect, specular reflection appearing as LED

Resources

[1] https://stackoverflow.com/questions/15911783/what-are-some-common-focus-stacking-algorithms
[2] https://stackoverflow.com/questions/37765188/focus-stacking-using-opencv-emgu-cv
[3] https://en.wikipedia.org/wiki/Focus_stacking
[4] https://www.learnopencv.com/image-alignment-ecc-in-opencv-c-python/
[5] https://affinity.help/photo/en-US.lproj/index.html?page=pages/FocusMerging/focusmerging.html?title=Focus%20merging%20images
[6] https://photo.stackexchange.com/questions/90885/macro-photography-stacking-focus-shift-vs-camera-movement
[7] https://www.cambridgeincolour.com/tutorials/focus-stacking.htm