CSC 481 – Intro. To Image Processing
2024-25 – Winter Term
Instructor: Kenny Davila Castellanos

Weekly Assignment 2: Image Transformations

Objective

The main objective of this assignment is to help student get familiar with multiple image transformations.

Data

To complete this assignment, please pick 3 images of your liking. You should use:

1) A picture from any interesting dish from your hometown. This can be a picture taken by you or from the internet. Please clearly indicate the source on your report.
2) A picture from any interesting dish from Chicago. This must be a picture taken by you. If you are from Chicago, please pic a different picture from the first one.
3) A random picture of your liking different from the first two.

You will have to run your code with each of these images, and then you must display and briefly discuss the results in your report.

Part 1 – Image Scaling by Pixel Replication

Write your own code capable of **shrinking** and **zooming** an image by **pixel replication and decimation**. Apart from the image, your code should take as input the desired zoom or shrink factor using integer values which will be used as follows:

- A **positive value** will expand the image. For example, a value of 3 will make the input image 3 times its original size, by replicating each value 3x3 times:
  - $\begin{bmatrix} 0 & 10 \\ 20 & 30 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 20 & 20 & 20 & 30 & 30 & 30 \\ 20 & 20 & 20 & 30 & 30 & 30 \\ 20 & 20 & 20 & 30 & 30 & 30 \end{bmatrix}$
- A **negative value** will shrink the image. For example, a value of -2 will make the input image 1/3 of its original size, by decimating (deleting) every 2 rows and columns, and then keeping the next one:

$$\begin{bmatrix} 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 0 & 0 & 0 & 10 & 10 & 10 \\ 20 & 20 & 20 & 30 & 30 & 30 \\ 20 & 20 & 20 & 30 & 30 & 30 \\ 20 & 20 & 20 & 30 & 30 & 30 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 10 \\ 20 & 30 \end{bmatrix}$$

- The **value zero** will create a copy of the original image, without changing its size.

Do not use preexisting language libraries for decimation and replication. Specifically, you must deal with pixels in your code.

(a) Use your code to shrink an image by a factor of 8 in each dimension (e.g. from 800x800 to 100x100 pixels). Show the shrunk image.

(b) Use your code to zoom the image back to its original size (e.g., from 100x100 to 800x800). Show the zoomed image and explain how and why the original image and the shrunk/zoomed images are different. In addition to your commentary on the visual differences, perform image subtraction between the original image and the restored image and show the subtraction results.

For **every input image** (you have 3 inputs, see **Data**), you must show: the original image, the shrunk image, the restored image, and the difference image.

Part 2 – Basic Gray Level Transformations

Write code that will do the following:

1) Load your input image.
2) If the image is not grayscale already, your code must convert it to grayscale.
3) If your image is stored as an array of bytes or integer values, convert it to floating point values.
4) If range of values in your image is from 0 to 255, convert them to values between 0.0 to 1.0 by dividing them by 255.0 (this will not work well if you skip the previous step!)
5) Create an **inverted image**, where for every pixel: $[New\ Intensity] = 1.0 - [Old\ Intensity]$
6) Use the power function to play with the **contrast** of the image.
    a. Pick a total of 4 values: 2 values between 0.25 to 0.99, and 2 values between 1.1 to 4.0.
    b. For each of these values, create a copy of the image where:
        i. $[New\ Intensity] = power([Old\ Intensity], [your\ value])$

For **every input image** (you have 3 inputs, see **Data**), you must show: the original image, the inverted image, 2 images with decreased contrast, 2 images with increased contrast. Keep in mind that to display these images, you might need to revert step 3 and 4 (multiply by 255.0, and convert to byte or integer values).

Part 3 – Image Transformations

Image scaling is just one kind of transformation that we could use to modify an image. It belongs to a more general class of transformations called **affine transformations**. Do some research on affine transformations (Wikipedia is typically a good place to start). In this part, you will be required to write code to rotate the image around a **given point**. Note that results for any rotation vary significantly depending on the point used as the axis. You are allowed to use existing library functions for this. Your code should do the following:

1) Load the input image.
2) Use a library function to rotate the image **around it's top-left corner** +45 degrees.
3) Use a library function to rotate the image **around it's top-left corner** -45 degrees.
4) Use a library function to rotate the image **around it's center** +45 degrees.
5) Use a library function to rotate the image **around it's center** -45 degrees.

For **every input image** (you have 3 inputs, see **Data**), you must show: the original image, 2 images rotated around the top-left corner, and 2 images rotated around their center.

The Program

In this assignment, you will write a program that handles multiple operations. You can use any programming language of your preference, but Python is highly recommended. Your code should be properly organized and well-commented to clearly identify the correspondence between portions of the code and each part of this assignment. The code should compile and run on any properly configured programming environment without the need of making significant changes to it. In this sense, it is highly recommended to avoid the usage of absolute paths.

The report

You are asked to submit a written report showing the results for each part of the assignment. Use proper section headings and descriptions to clearly identify the results of each part. If I cannot easily find a match in the report for a required result, I might assume that this portion was not completed, and a score of 0 will be given by default.

For students working with Python, you are allowed to use Jupyter Notebooks. These allow you to combine code with the report in a single deliverable, which is perfectly acceptable for this course.

Delivery Instructions

You are given the freedom to use any programming language and IDE of your preference. You are also required to provide your own images, and to produce a written report. When done, **you should submit everything using a single Zip file**.

File names. The zip file that you submit should use "[Last Name(s)], [Given Name(s)].zip" as it appears in D2L. For example, "Kenny Davila Castellanos" (Davila Castellanos is two last names), would have to

submit the homework with the name “Davila Castellanos, Kenny.zip”. Another student named “Kenny Mauricio Davila” (Mauricio is a middle name), would have to submit the homework as “Davila, Kenny Mauricio.zip”. Not following these instructions might lead to a penalty.

Policies

1. All general policies about Plagiarism and Cheating apply to this homework. If you plagiarize or receive code from other people, you will be caught and you will receive a score of 0, and a report of the academic integrity violation will be filed.
2. Please limit the usage of Chat-GPT and other code generators to ethical usage only. Submitting code that was directly generated by these tools is considered a form of plagiarism.
3. Do not post your solutions online and do not share them with anyone. It is your responsibility to safeguard your private data.
4. **Code that does not compile due to syntax and/or semantic errors will automatically receive a score of 0.** It is hard to assign partial credit when I cannot even run your code.
5. You can use any programming language of your choosing.
6. You must follow the delivery instructions.
7. Very late homework’s without justification will receive a score of 0.
8. The homework description outlines very specific requirements. You are welcome to try other things and report your results. However, no amount of extra work can be used as a substitution of the actual requirements.
9. Do ask for help if anything is unclear, but do it in a timely manner (e.g., by e-mail, Discord or during the Office Hours).