

Project 2 - Image Manipulation

Checkpoint Deadline: End of tutor hours Thursday 11/16

Project Soft Deadline: 10pm Tuesday 11/21

Project Hard Deadline: 10pm Monday 11/27

Overview

In this project, you will work individually to create a photo collage or image of personal meaning to you using the Python PIL library. You'll be expected to use your own pictures, potential in combination with other open-access pictures (with attribution), to create something that you'll be proud of. As with the last project, the goal is for this activity to showcase your creativity and be valuable to you personally. It's a success to us if you share these with your family or friends and say "Look how cool this is! I made this by myself using Python!"

As with Project 1, you'll have a checkpoint due on Thursday 11/16 where you will talk through your plans with a tutor who will help you scope the project well. For the project deadline, you'll submit your code, a diagram of the functions you wrote for the project, and a video explaining the project.

Your Tasks

Image Selection and Project Vision

The first step is figuring out what you want to create. Do you want to do a green-screen substitution where you superimpose yourself in your favorite movie in some way? Do you want to create a collage of pictures with your best friend adding artistic touches? Are you an artist and you want to blend your physical art with digital components? The first step is to figure out what images you want to use and what you want to create.

For images, be sure you either own the image (you took it yourself) or you have permissions to use it. There are tons of open-access images on Google Images and other websites, so we suspect you won't have any problems. But in your project code, be sure to say where you got the images.

Project Requirements/Expectations

Although creativity is our primary goal here, we do have some expectations for you in terms of your learning goals for the class:

- The project should require at least **4 picture functions** to complete (see Problem Decomposition)
- Your project should use **at least 3 different source images**
- You should have at least one function that copies or blends one image onto another image
- You should have at least one function that performs some kind of image filter (negative, edge detection, etc.)
- You should be using the PIL library and doing the image changes yourself (i.e., if you find a Python library that does green-screen substitution, you can't use that)
 - Similarly, you should restrict the functions/methods that you use from the PIL library to those discussed in class. For example, the PIL library has a grayscale function, merge functions, paste, etc. that you should not be using. If you want a function like that, you'll need to make it yourself. 😊
 - To clarify, you can use the following:
 - size
 - getpixel
 - putpixel
 - copy
 - save
 - show
 - Image.open
 - Image.new
 - rotate (only by 90/180/270 degrees to make an image that is stored incorrectly the right direction)
- You **cannot** use Photoshop or any similar tool. You must use the original source images and we will be checking that the original images are authentic and that your code only uses those original images. Yes, some of the things you'll be doing here are available in image processing software, but the goal here is for you to write those tools! (Really, how cool is that, right?!)
 - Examples of things not allowed: using a tool to make the background of an image uniformly one color to make it easier for green screen substitution, applying filters to an image (e.g., blurring or blemish removal), etc.
 - The *only* exception to using image processing software is that you can use other tools to resize your images if you want (e.g., resizing a 10MB image into something smaller)
 - Similarly, no use of AI image generation tools.
 - You *can* use a 3rd party tool to convert a PNG image (or other format) to a JPG image as we're learning how to work with JPG files in this class. Just include the JPG image in your original sources.

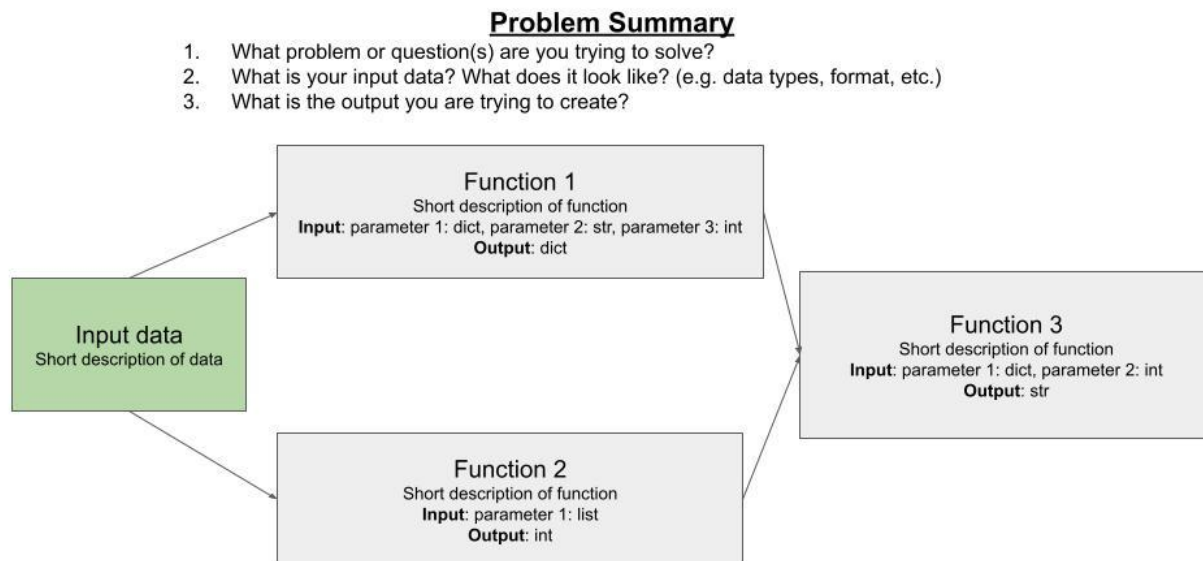
- The grader should be able to reproduce the result images you submitted by simply running your Python code on the original images. Any discrepancy between the images generated by the tutor and the ones you submitted will be considered use of an outside image editing tool/software and result in you losing points on the project.

Problem Decomposition

Just like Project 1, we expect you to break apart your problem into functions. Perhaps one function does green-screen substitution, another copies one image onto another, and another does edge detection. Try to make the functions you write as versatile as possible (i.e., not just useful for your project this quarter). You'll ultimately need to break up the overall problem into small enough tasks that Copilot can help you write the code for those steps. Once you've completed the problem decomposition, we'll expect you to create a figure like Figure 7.3 in the textbook. Please expand that figure to show the full function signatures and a brief description of each function.

Below you can find an example of what your problem decomposition may look like.

IMPORTANT: This image is only a **suggestion** and your problem decomposition can have a different number of functions that interact with each other differently (or not at all).



Coding and Testing your Solution

In conjunction with the task of problem decomposition, you'll be authoring the functions (optionally with help from Copilot) to complete the tasks. Be sure to have a plan on how you'll

test each function, likely using a smaller image or single color images to help you understand if it is working or not. One nice thing about images is you can see the result of your code and can visually inspect if it's working or not.

Checkpoint - Due at end of tutor hours on Thursday 11/16 (5%)

Your meeting with the tutor will last roughly 5-10 minutes. This is an opportunity for you to get feedback on your plans for the project and for us to make sure you are on track to complete the project on time. The checkpoint is worth 5% of your project grade. Completing it before Tuesday 11/14 will yield an additional 1% bonus toward the project. A calendar with all the tutor hours can be found on the Course Calendar [here](#).

What to have done before your meeting with the tutor:

Before meeting with the tutor, have the following done:

1. Have your images selected
2. Have your idea for what you want to create already in mind (preferably as a basic sketch)
3. Have code that can open the images you wish to use and display them.

If you are missing any one of these **three** things when meeting with a tutor, you will **NOT** be checked off and will have to resubmit an Autograder ticket to meet with the tutor again.

What to expect from your meeting with the tutor:

You will request your meeting with the tutor either online or in-person using [Autograder](#) based on the Tutor Lab Hours posted on the [Class Calendar](#). (Professor and TA Office Hours will not be used for checkpoints and will be reserved for other questions you may have about the class. In your meeting, the tutor will ask you about your images, the image you plan on making, and ask you to show them your code that opens your images. If you have each item done, you'll receive full credit and the tutor will check you off as completing your checkpoint. The tutor will also offer you feedback about the difficulty of the task and you can ask them for advice on how to modify images.

Note that tutor hours are limited. **Waiting until Thursday to ask for your checkpoint risks not getting your checkpoint completed in time.** There are no late checkpoints, so not completing your checkpoint by the end of the last tutor hours on Thursday will result in a 0 for your checkpoint. We are happy to still discuss your project with you, etc., after the checkpoint deadline but no credit will be given.

Project Submission - due 10pm Thursday 11/21

You will be turning in 3 parts to Canvas for your project submission: your code, the diagram of your function hierarchy, and a link to a video of you explaining your project. See details below:

Code submission (20%)

You will upload all the code and your dataset to Canvas for grading. **The tutor grading your submission needs to be able to download your code and run it and have the code work as expected.** You need to include all the source images so they can run the files and you should hard-code the images (i.e. the code should know the image for grayscale is "image.jpg" not ask the tutor for the image name). Once the tutor runs the code, the file result file should be displayed and put into the same directory (e.g., named "result.jpg").

Problem Decomposition (20%)

You will upload an image summarizing your problem decomposition to Canvas. Specifically, how did you take your large problem and break it into functions that helped to solve the larger problems. Your image should be an expanded version of the example in Figure 7.3. Each function should have its inputs and outputs included and a brief description of the function.

Explanation Video (55%)

You will upload a link to a video of you explaining the project on Canvas. Your video should include the following:

Video quality/Details:

- Record the video with both your face and your presentation/code.
- Please either give us a link to the video in a **google drive folder** (that is public to anyone with the link) or to a private **Youtube** video.

The 5 minute video should contain:

- (1 min) You briefly describing your images and what your original vision for the project
- (1 min) You talking through why you decomposed the problem into the functions you did
- (3 min) A detailed walkthrough of one of your functions where you explain how it works.

We will only grade the first 5 minutes of your video, so if you post a video longer than 5 minutes, we will only grade the content in the first 5 minutes.

Submitting Files

You will be submitting 3 things for the project

1. Python code and your source and result images
2. A PNG, PDF, or JPEG image of your project decomposition
3. A text file that contains the link to your video

All of these files can be compressed into a **zip folder** which you can upload to **Canvas**.

Academic Integrity

You can ask for help from the instructional staff and Copilot. You should not discuss your project with other students in the class. We expect the questions you ask of your data and the code you write to be your own (the code can be aided by Copilot). Use of any existing analysis online is forbidden (as is using the project of a classmate) and turning in that code will result in your work being submitted for an Academic Integrity Violation.

Please see the class Academic Integrity Agreement for more details.