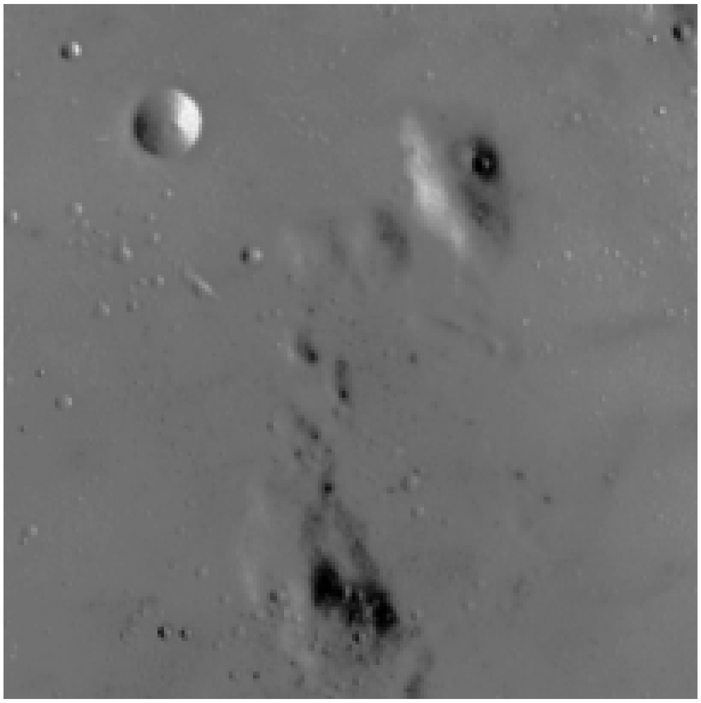
Image Processing: Assignment #2

Problem 1 – Point operations

a)

Image 1:



This image appears to be a grayscale representation.
The contrast between image parts is moderate, but the details could be enhanced.

<u>Brightness and contrast stretching:</u>

This could be a good option for this image.
It will increase the contrast between the different features, making the details of the surface more discernible.
It will make the bright areas brighter and the dark areas darker, which could help in emphasizing the texture and depth of the holes.

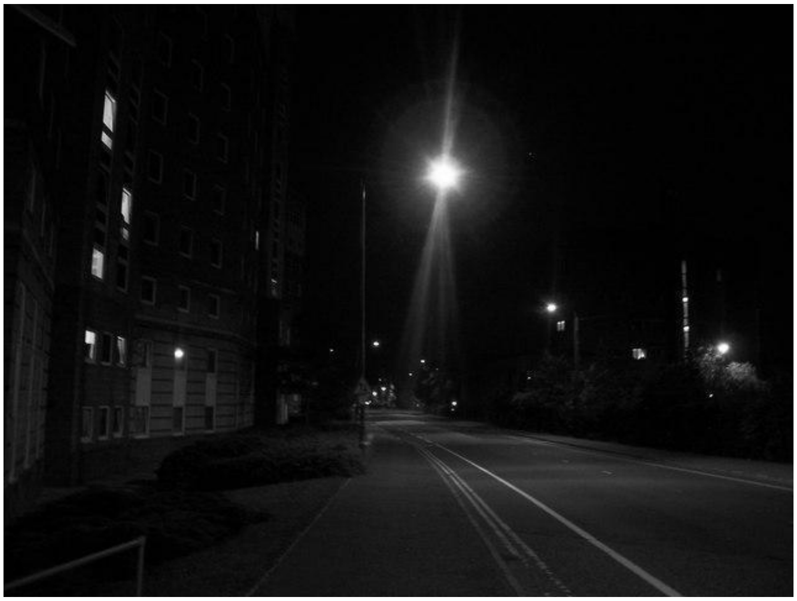
<u>Gamma correction:</u>

It may not be as effective as contrast stretching for this particular image.
Could either wash out details or make the image too dark.

<u>Histogram equalization:</u>

Could spread out the most frequent intensity values, which might be useful in this case.
Would enhance the contrast and could bring out details, but it may also introduce unwanted noise or make some areas look unnatural.

Image 2:



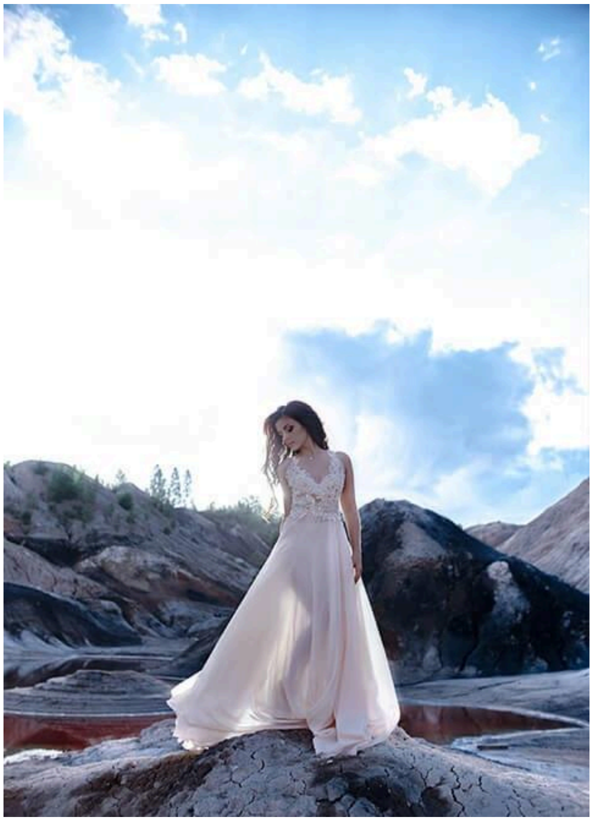
This image is a night scene on a street with a light source directly in view, causing a glare. The surrounding areas are underexposed due to the strong contrast in lighting.

Brightness and contrast stretching:
This may not be the best choice as it would make the glare from the light source more intense and could further underexpose the darker areas.
could lead to loss of details in the light due to overexposure.

Gamma correction:
Could be a good option, as it can help lighten up the darker areas without overexposing the already bright areas too much.
Might help to reveal details in the darker areas without affecting the brighter areas too much.

Histogram equalization:
Might not be suitable because it could make the glare from the light source too hard and still fail to reveal details in the darkest areas.
Could potentially make the image look unnatural because of the extreme range of intensities from the light source to the shadowed areas.

Image 3:



This image features a person in a white dress against a bright sky and dark ground. The sky is slightly overexposed.

Brightness and contrast stretching:

This could be beneficial to balance the lighting between the sky and the darker ground, potentially enhancing the overall detail.

Would enhance the details in the dress and ground without affecting the sky too much.

Gamma correction:

The lighting doesn't appear to be too imbalanced.

Histogram equalization:

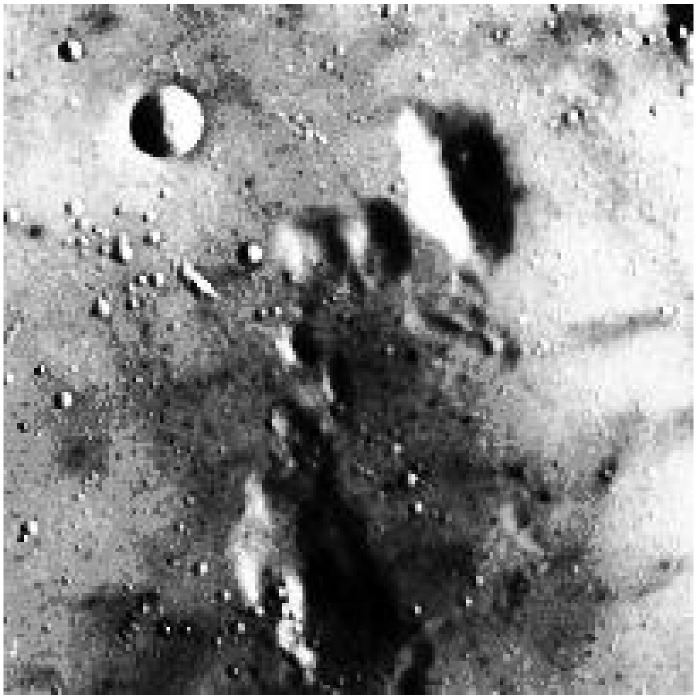
This might help in bringing out the details in the dress and the clouds, but it could also overemphasize the already bright areas of the sky.

Could lead to an overexposed sky and loss of cloud details.

Could improve the texture of the dress.

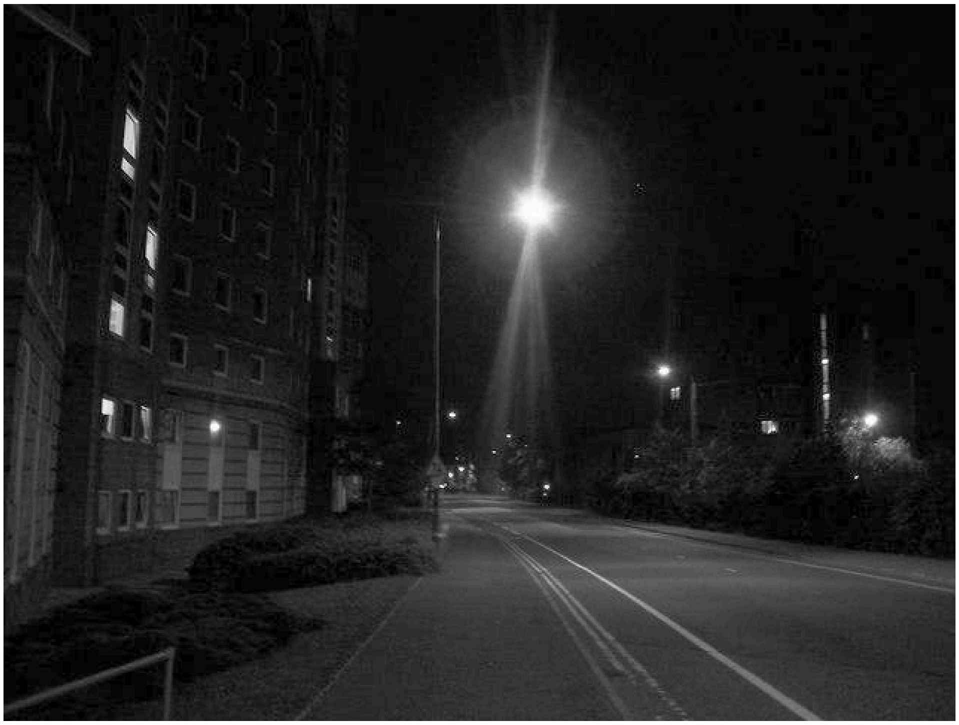
b)

Performing Histogram equalization on **image 1**:



Parameters:
None.

Performing Gamma correction: on **image 2**:



Paramaetrs:

```
gamma = 1.5
inv_gamma = 1.0 / gamma
```

image 3:
We would apply any change.
The image has a high dynamic range, with a bright sky and a darker foreground.

Brightness and Contrast Stretching:
While it could help balance the lighting, it might make the sky too bright, losing the subtle details in the clouds.

Gamma Correction:
Applying gamma correction might not yield a significant improvement since the lighting does not seem imbalanced in the mid-tones. It could potentially make the image look unnatural.

Histogram Equalization:
This technique could over-enhance the brightness of the sky, leading to a loss of detail in the clouds.
Since the dress appears to be well-exposed, histogram equalization might make it too bright, risking the loss of texture and detail.

Problem 2 – Puzzle solving with geometric operations:

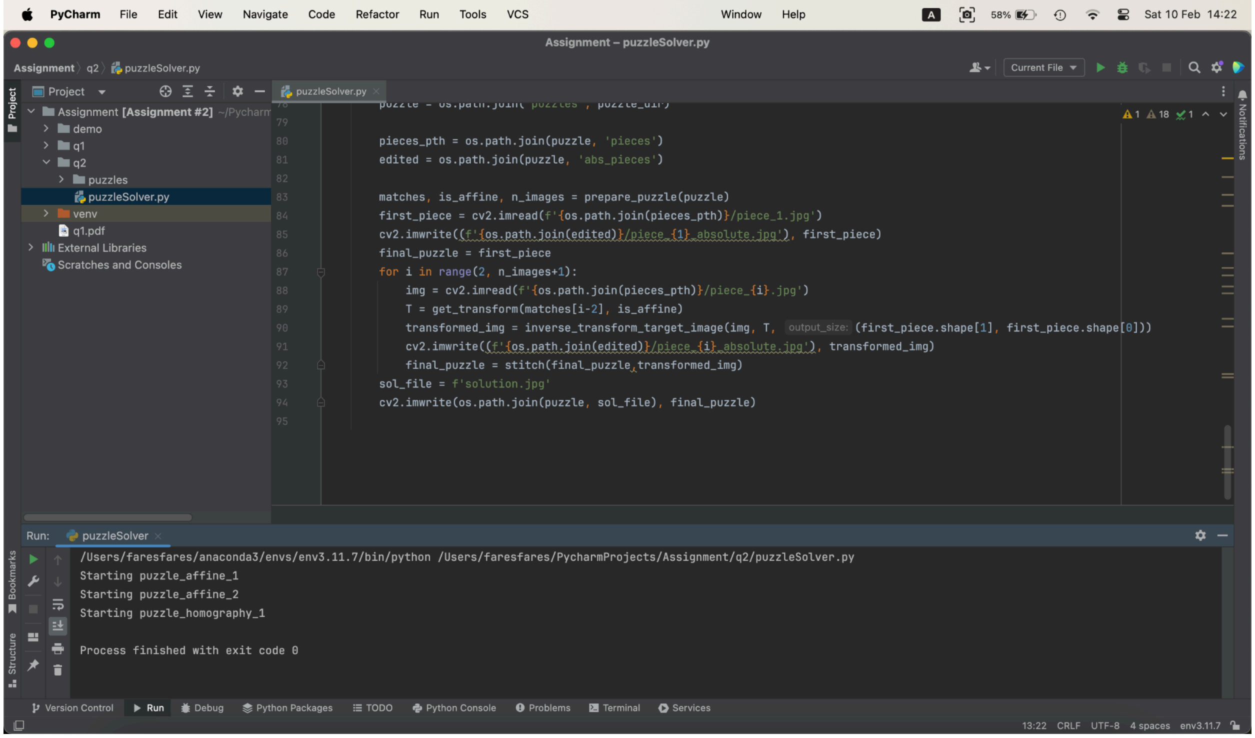
get_transform: This function calculates the transformation matrix needed to stitch two puzzle pieces together. It can compute either an affine or projective (homography) transformation based on the matches provided and a boolean flag is_affine.

stitch: This function takes two images and stitches them together using the maximum value of each pixel.

inverse_transform_target_image: This function applies the inverse transformation to an image to revert it to its original position. It handles both affine and homography transformations.

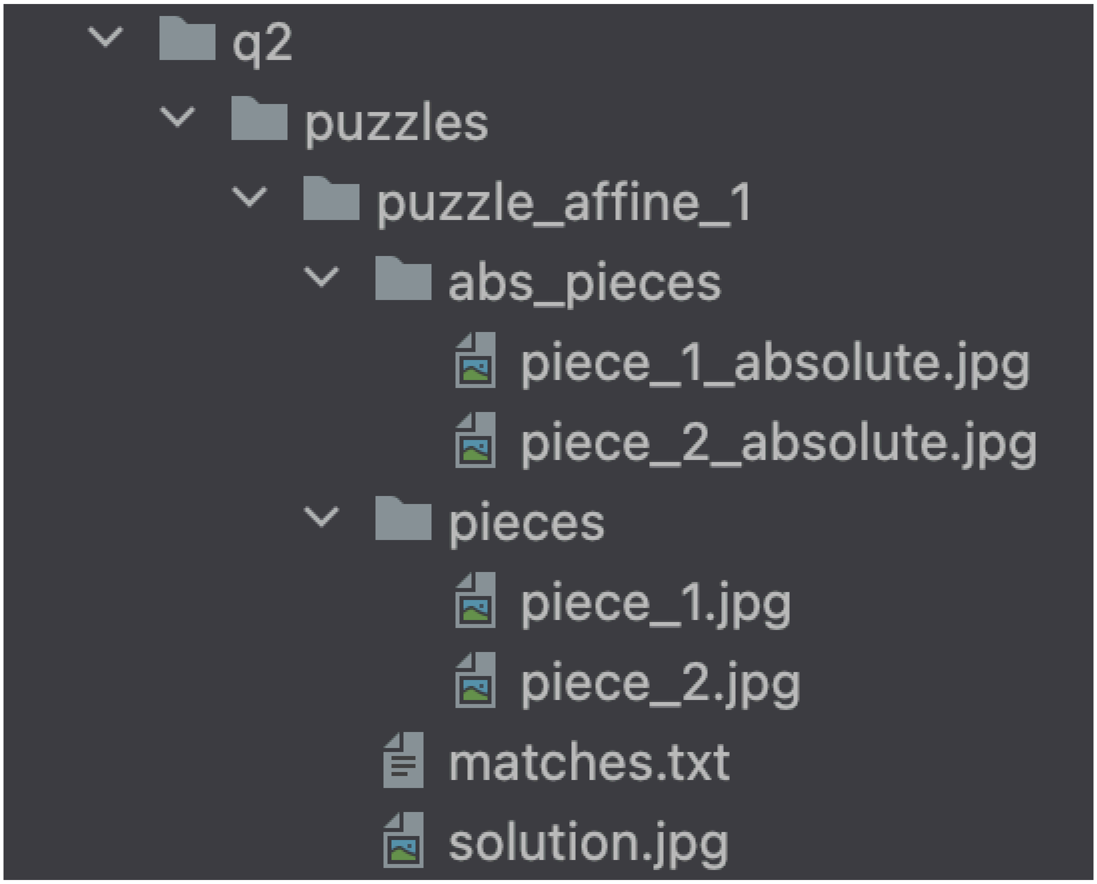
prepare_puzzle: This function prepares the puzzle by reading a directory of puzzle pieces and a file containing matches data, which is used to later stitch the puzzle together.

The main iterates over a list of puzzle directories, loads the puzzle pieces, calculates the necessary transformations, and stitches the pieces together to form the final puzzle image.

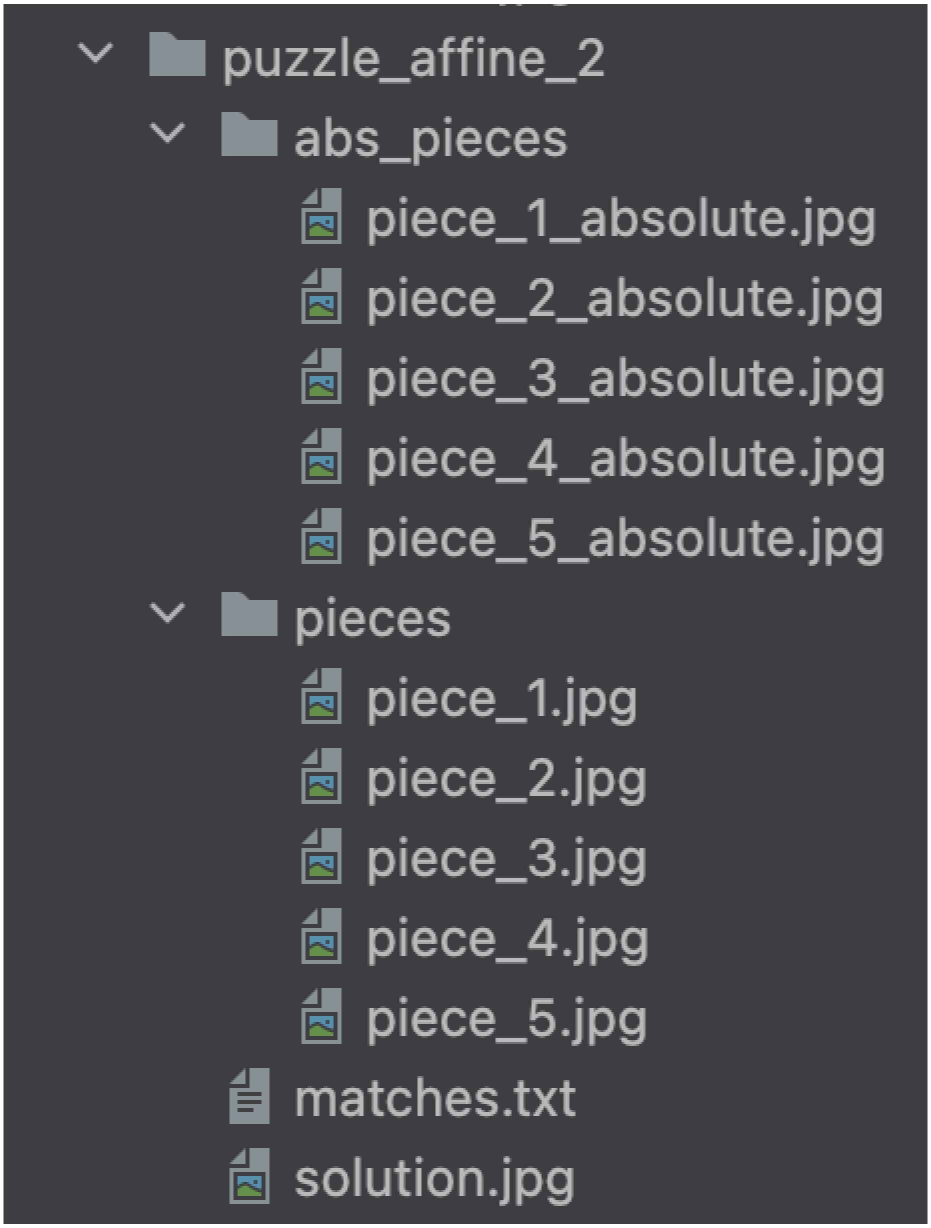


```python
    puzzle = os.path.join('puzzles', puzzle_dir)

    pieces_pth = os.path.join(puzzle, 'pieces')
    edited = os.path.join(puzzle, 'abs_pieces')

    matches, is_affine, n_images = prepare_puzzle(puzzle)
    first_piece = cv2.imread(f'{os.path.join(pieces_pth)}/piece_1.jpg')
    cv2.imwrite((f'{os.path.join(edited)}/piece_{1}_absolute.jpg'), first_piece)
    final_puzzle = first_piece
    for i in range(2, n_images+1):
        img = cv2.imread(f'{os.path.join(pieces_pth)}/piece_{i}.jpg')
        T = get_transform(matches[i-2], is_affine)
        transformed_img = inverse_transform_target_image(img, T, output_size: (first_piece.shape[1], first_piece.shape[0]))
        cv2.imwrite((f'{os.path.join(edited)}/piece_{i}_absolute.jpg'), transformed_img)
        final_puzzle = stitch(final_puzzle,transformed_img)
    sol_file = f'solution.jpg'
    cv2.imwrite(os.path.join(puzzle, sol_file), final_puzzle)
```

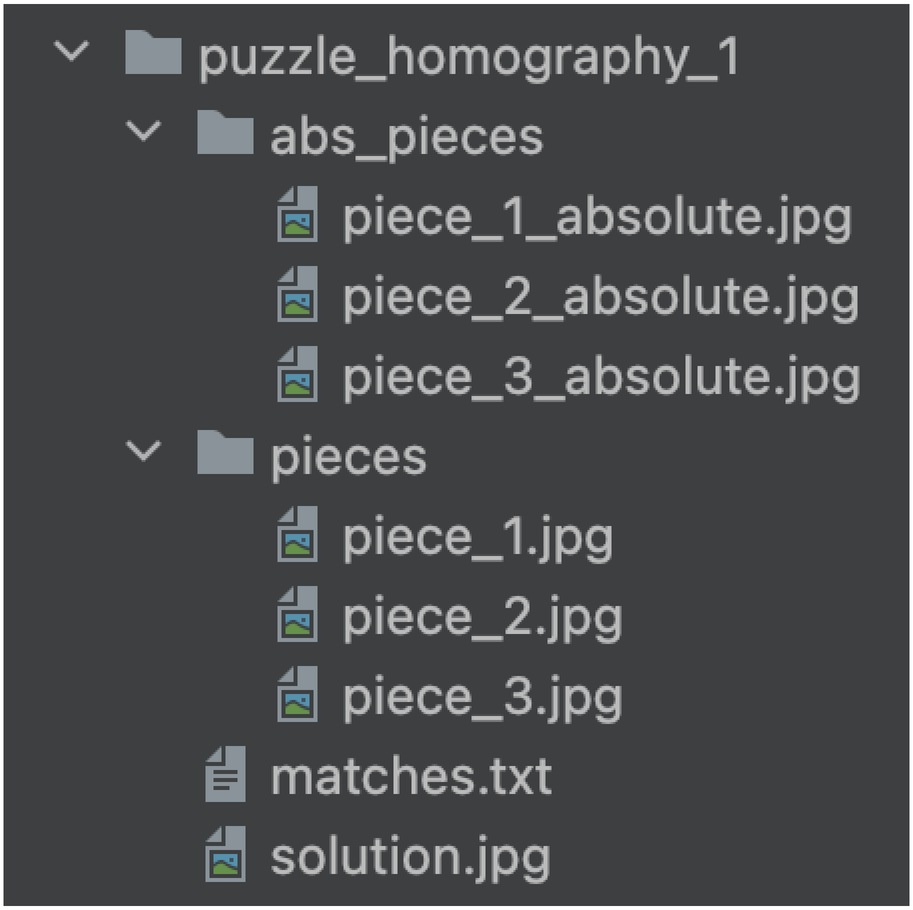
```
/Users/faresfares/anaconda3/envs/env3.11.7/bin/python /Users/faresfares/PycharmProjects/Assignment/q2/puzzleSolver.py
Starting puzzle_affine_1
Starting puzzle_affine_2
Starting puzzle_homography_1

Process finished with exit code 0
```



```
q2
  puzzles
    puzzle_affine_1
      abs_pieces
        piece_1_absolute.jpg
        piece_2_absolute.jpg
      pieces
        piece_1.jpg
        piece_2.jpg
      matches.txt
      solution.jpg
```



```
puzzle_affine_2
    abs_pieces
        piece_1_absolute.jpg
        piece_2_absolute.jpg
        piece_3_absolute.jpg
        piece_4_absolute.jpg
        piece_5_absolute.jpg
    pieces
        piece_1.jpg
        piece_2.jpg
        piece_3.jpg
        piece_4.jpg
        piece_5.jpg
    matches.txt
    solution.jpg
```



```
puzzle_homography_1
  abs_pieces
    piece_1_absolute.jpg
    piece_2_absolute.jpg
    piece_3_absolute.jpg
  pieces
    piece_1.jpg
    piece_2.jpg
    piece_3.jpg
  matches.txt
  solution.jpg
```