Aerial Robotics Kharagpur

Taranisen Naik ,24ME10035

GEBE DICH NIE AUF

In this project, we perform certain image processing operations on different images, and the result of one image is used in the other image for processing, which is then used to unlock a file required for the next task. For this project, we will use different techniques like storing information in an image as the intensity of different pixels, template matching, which is used to check the similarities between 2 images

Applications:

Template matching can be used to compare similarities between 2 images

INTRODUCTION

An image is given to us in which each pixel's intensity corresponds to a digit*10 of pi. There are 4 digits which are distorted by other person we have to first find the distorted digits and then multiply each digit with 10 and pi and take the floor of it and short them in decreasing order and make a 2 x 2 matrics which act as a filter. Then we have to perform AND, XOR and OR operation to each pixel of the artist's painting with this filter. The best image of the three will be used for the template matching in the collage. The starting coordinate of the matched image in the collage is the required x and y for the final password formation, which will be used to get the image of the maze of the third task

PROBLEM STATEMENT

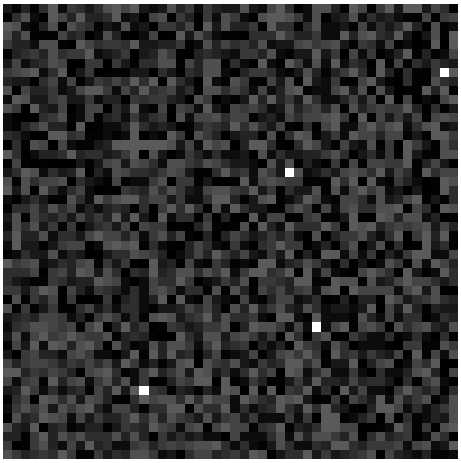
This task entails several steps of image processing, such as filtering, bitwise operations, scaling, and template matching.

Extract the Filter from the π Image

The π image (pi image.png) is distorted. The distortion can be undone using the provided **transformation:**

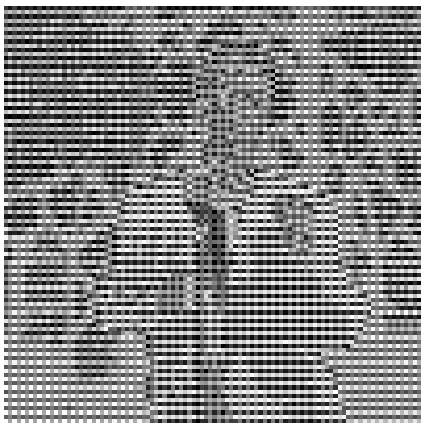
Corrected Digit=⌊Distorted Digit×10π⌋

Place the corrected digits in descending row-major order to create a 2×2 filter.



Pi image

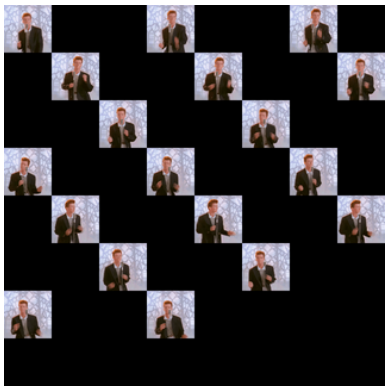
Recover the Portrait:
Apply the 2×2 filter obtained through extraction to the distorted Picasso image (distorted_picasso.png).
Use bitwise operations:
(AND, OR, XOR) between each pixel and pi filter to restore the image.



Distorted image

Perform Template Matching:
Using collage.png, determine the position of the best match by using self made template matching.calculate sum of squared differences (SSD) between the image and the collage and use the starting coordinate of the matched template for retrieving the password.

Calculate the Password:
password is the floor of (x+y) * pi



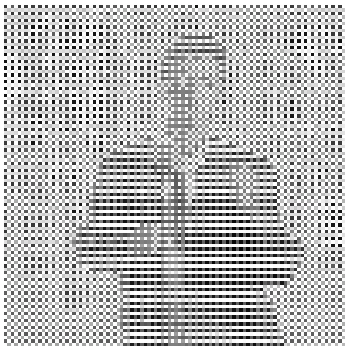
collage.png

APPROACH

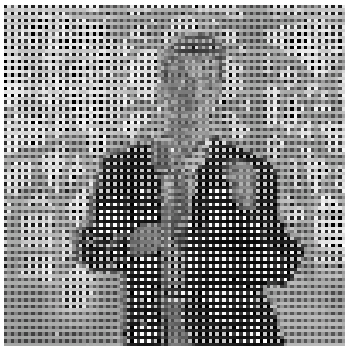
Computing the pi filter
I first convert the image into gray scale and then convert the resulting image into an array whose each element represents a pixel and the value of that element is the intensity, which represents the (digit of pi)* 10. Then I compute the size of the image, which was 50X50=2500 pixels, meaning it contains 2500 digits of pi. Then I use the SciPy library to get 2501 digits of pi. I computed 1 more digit because it always gives the rounded value, and to get the required digits, I removed the last digit and compared each digit from the real pi and the image, and stored the mismatched digits in an array. At last, I computed the required operation Corrected Digit=⌊Distorted Digit×10π⌋, then sorted it in decreasing order and converted it to a 2x2 matrix, which is our required pi filter.

[[282 251]
[94 0]]

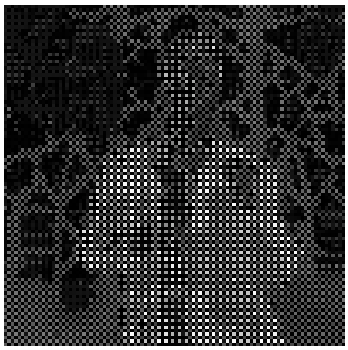
Restoring the image
perform AND, OR, and XOR between each pixel of the image initially, I was not taking a step of 2, for which the operation was done twice for each pixel resolution in either a purely black or white image. Then I introduced a step size of 2 and got the proper restored image. I found the image obtained from xor more accurate for template matching so I used it .



OR output

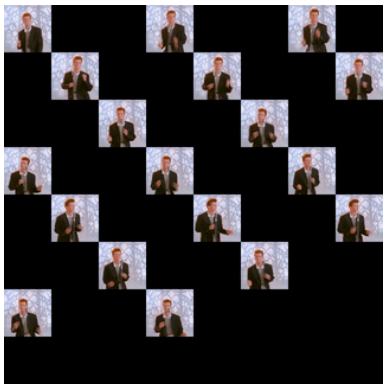


XOR output



AND output

Template matching:
For template matching, we will first convert the collage to a gray-scale image, then we will take the restored image and collage and iterate through each pixel of the collage and compute the sum of square difference, and when the difference is minimum, we can conclude that the image is matched with that template.



Collage

As for template matching, we are using the starting coordinate of the image. So we will get x and y of the matched template very easily, which is (100,100).

matched template

The requied password :
floor((100+100)*pi) = 628

CONCLUSION

After doing each image processing in a sequence, we successfully recovered the hidden portrait and extracted the password.

Filter Extraction
By analyzing pi image.png, we identified the distorted digits of pi.Using the transformation ⌊Distorted×10π⌋, we reconstructed the

original values.
The values were then arranged in descending row-major order to form a 2×2 filter.

Restoration of the Portrait
The filter was applied to distorted image using bitwise operations (AND, OR, XOR) between filter and image. A step of 2 ensured correct transformation to get the correct image.

Template Matching
The portrait wa 100X100 so we didn't resize it. Using a custom template matching approach (SSD), we found the best match template in collage.png. The (x, y) coordinates of the matched template were extracted.

Password Generation
By computing $\lfloor (x+y)\times\pi \rfloor$, we determined the final password for the zip file.

USE CASE IN ARK

This technology of template matching can be used for a searching operation with drones. To spot an object. To follow a predefined path

RESOURCES USED

Mustafa Salam, "Template Matching - Pattern Recognition "
University of Technology
Dept. Computer Science,Oct 20, 2014

https://youtu.be/1_hwFc8PXVE?si=GEGFGnNJzC_0KzL7

https://docs.opencv.org/4.x/de/da9/tutorial_template_matching.html