

# task

September 6, 2024

## 1 Task

Sergei Mikhailovich Prokudin-Gorskii (1863-1944), a pioneering photographer extensively traveled the Russian Empire from 1909 to 1915, capturing thousands of images. Employing an early color technology, he recorded three images of each scene onto glass plates using red, green, and blue filters.

Your task is to take Gorskii glass plate RGB color channels and produce a color image with as few visual artifacts as possible (ignore the artifacts at the image boundaries). To do this, you will need to read the image file, extract the three color channel images, place them on top of each other, and align them so that they form a single RGB color image. You will need to implement this in Python.

**Note** that the channel order for all files from top to bottom is BGR, not RGB!

```
[16]: import cv2 as cv
      from matplotlib import pyplot as plt
      import numpy as np
```

```
[17]: aus_id = 93749
      if aus_id % 3 == 0:
          filename = "00153v.jpg"
      if aus_id % 3 == 1:
          filename = "00149v.jpg"
      if aus_id % 3 == 2:
          filename = "00351v.jpg"
      img = cv.imread(filename, cv.IMREAD_GRAYSCALE)
```

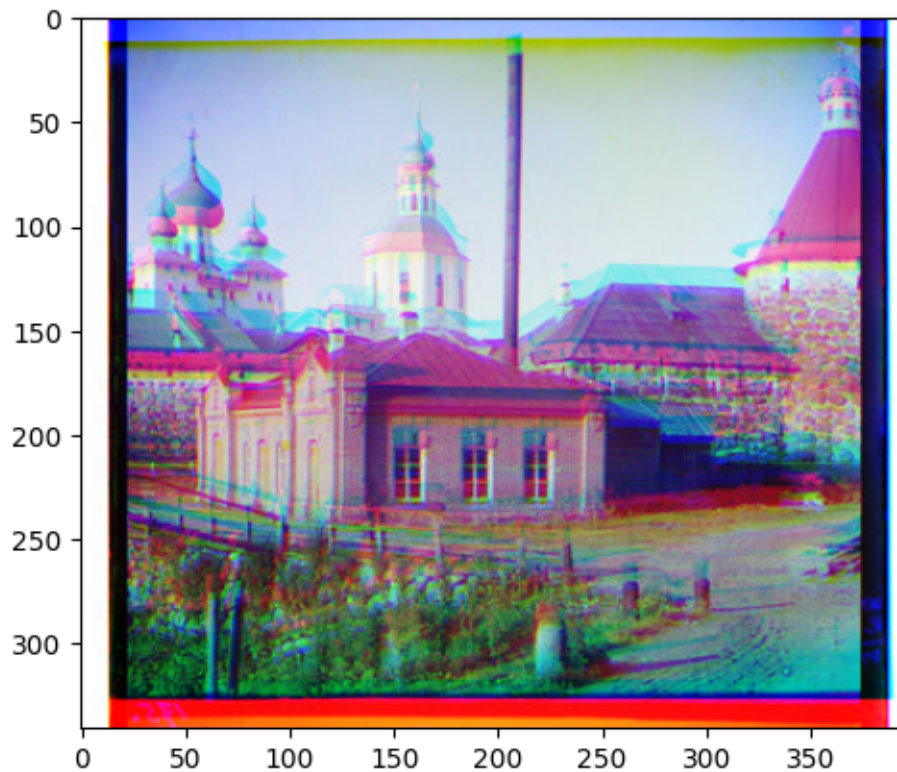
## 2 Step 1 [5 points]:

- Read the image, extract the Blue, Green and Red channels, stack the channels and display using matplotlib
- When you display the image, you will notice that the color channels are not aligned properly.

```
[18]: num_of_rows = len(img) // 3
      B = img[:num_of_rows]
      G = img[num_of_rows : 2 * num_of_rows]
      R = img[2 * num_of_rows : 3 * num_of_rows]
```

```
[19]: def display_image(rgb_channels: tuple[np.ndarray, np.ndarray, np.ndarray]) -> None:
      img = cv.merge(rgb_channels)
      plt.imshow(img)
      plt.show()
```

```
[20]: display_image((R, G, B))
```



### 3 Step 2:

- Find the displacement needed to align RGB channels by following the steps below:
- (1) [3 pts] Use canny edge detector to find the edges

ANSWER: > Question is skipped

- (2) [4 pts] Identify horizontal or vertical edges which could help you find the displacement needed to align the color channels. You can just write your findings here. No need to show the code for this section

ANSWER: > The horizontal edge at the top of the buildings are shifted. We might need to shift the green and red channels down. However, we this is just a guess and we need to verify this in the next step.

- (3) [6 pts] Design and apply filters using convolution to align/shift the color channels

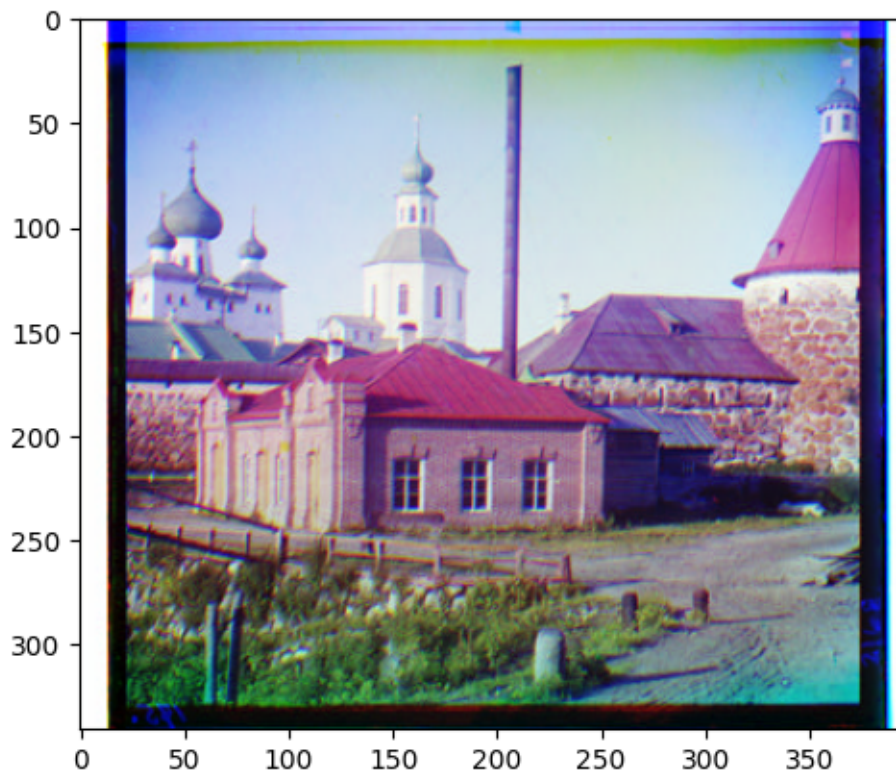
```
[21]: def shift_vertical(channel: np.ndarray, shift_amount: int) -> np.ndarray:
    shift_direction = 0 if shift_amount > 0 else -1
    shift_amount = abs(shift_amount)
    kernel = np.zeros((shift_amount, shift_amount))
    kernel[shift_direction, shift_amount // 2] = 1
    return cv.filter2D(channel, -1, kernel)
```

### 3.1 Shifting green and red channels down

```
[22]: R_shifted = shift_vertical(R, 30)
    G_shifted = shift_vertical(G, 10)
    B_shifted = B # no shift for B channel
```

- (4) [2 pts] stacking the channels and display.

```
[23]: display_image((R_shifted, G_shifted, B_shifted))
```



## 4 Submission

- Submit the pdf of this completed notebook on iLearn