Project 2

Smaller Images

Cathedral
R displacement : (3, 12)
G displacement : (2, 5)

Monastery
R displacement : (2, 3)
G displacement : (2, -3)

Tobolsk
R displacement : (3, 6)
G displacement : (3, 3)

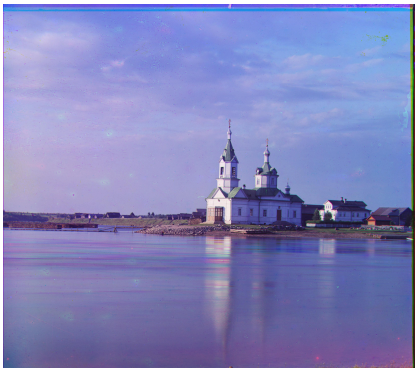
1. Smaller Images

My goal for this part was to simply create a score metric which accurately assessed how "similar" two channels were and use that to determine the alignment of these images. To create my score metric, I tried both the L2 Norm between the two images as well as the NCC, and decided to go with the former for these smaller images. Then, I looped between

all possible combinations from -15 and +15 for the height and width displacements. I displaced the red and green panels over the blue panel and calcuated the scores for both these pairs, settling on the two displacements which yielded the lowest scores. In order to calculate the scores for these displacements, I would temporarily crop the images so that they lined up with the proper displacement and then calculate the score between the new image matrices.

One issue which I ran into initially was innacurate scores. I noticed the displacements which were yielding the lowest scores where not the ones that seemed visually the most aligned. To fix this, I realize I would first have to crop the borders off of my images, since the borders are pure white, that would lead to big shifts in the scores which doesn't have to do with the alignment of the image itself. After doing this, the accuracy of my alignment improved greatly.

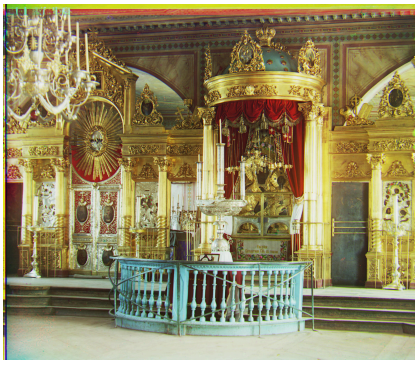
Larger Images



Church
R displacement : (-4, 58)
G displacement : (4, 26)

Emir
R displacement : (56, 104)
G displacement : (24, 50)

Harvesters
R displacement : (14, 124)
G displacement : (16, 60)



Icon
R displacement : (22, 90)
G displacement : (16, 40)

Lady
R displacement : (12, 118)
G displacement : (8, 54)

Melons
R displacement : (14, 178)
G displacement : (10, 82)

Onion Church
R displacement : (36, 108)
G displacement : (26, 52)

Sculpture
R displacement : (-26, 140)
G displacement : (-10, 34)

Self Portrait
R displacement : (36, 174)
G displacement : (30, 80)

Three Generations
R displacement : (10, 112)
G displacement : (14, 54)

Three Generations
R displacement : (32, 88)
G displacement : (6, 42)

By the Gundukush Dam
R displacement : (-60, 132)
G displacement : (-30, 56)

Lugano
R displacement : (38, 886)
G displacement : (12, 36)

Irrigation Canal
R displacement : (48, 46)
G displacement : (24, 28)

2. Image Pyramids for Larger Images

On larger images, I found my original algorithm taking much longer, especially since I needed to calculate larger offsets for these images as well. Therefore, I attempted to speed up the alignment process by using image pyramids. To do this, I started by scaling down my image by a factor of 16. Then, I ran the same process as before, calculating the score

for different offsets, this time from -10 to +10. Once i found the optimal offset on the coarse images, I rolled the original images by that amount of displacement and scaled it down by a factor of 8 next. I then repeated the same process again and again, scaling my image down by a smaller factor each time, until I came back up to the original image. I also searched a smaller and smaller window of displacement each time, since I didn't have to look as far each time.

For the score calculation, I found NCC working slightly better for this part, so I opted to using that instead. One other issue i ran into was that some of my images still weren't aligning that well, particularly the Emir image. To combat this, I thought that perhaps rolling the image each time before calculating the score again on the next level up the pyramid might be altering the score, since the edges which eand up on the opposite side may skew the score. Therefore, when calculating the score, I cropped off 5% from each edge of the image, just to confirm that we are only calcuating the score off of the parts of image we are focused on aligning.