Prokudin-Gorskii Image Colorization

Automated alignment and colorization of Sergei Prokudin-Gorskii's glass plate photographs using image processing techniques

Background Information

Sergei Mikhailovich Prokudin-Gorskii (1863-1944) was a Russian photographer who pioneered color photography. Between 1909-1915, he traveled across the Russian Empire and captured thousands of color photographs using a specialized camera that exposed three glass plates in rapid succession through red, green, and blue filters.

This project focuses on digitally reconstructing Prokudin-Gorskii's images by aligning the three color channels and producing a single color image with minimal visual artifacts. The process involves sophisticated image processing techniques including pyramid-based alignment, edge detection, and contrast enhancement.

Single-Scale Alignment

For smaller images, a single-scale alignment approach is sufficient. This method exhaustively searches for the optimal displacement between color channels within a predefined window (typically ±15 pixels). The algorithm calculates alignment scores using Normalized Cross-Correlation (NCC).

Normalized Cross-Correlation (NCC)

The NCC between two image patches is calculated as:

$$NCC = \frac{\sum_{x,y}(I_1(x,y) - \mu_1)(I_2(x,y) - \mu_2)}{\sqrt{\sum_{x,y}(I_1(x,y) - \mu_1)^2 \sum_{x,y}(I_2(x,y) - \mu_2)^2}}$$

Where I_1 and I_2 are the image patches, and μ_1, μ_2 are their mean intensities.

Algorithm Steps

1. Split the input image into three equal parts (B, G, R channels)
2. Crop borders to avoid edge artifacts (typically 10-15%)
3. For each possible displacement in the search window:

- Shift the target channel (G or R)
- Calculate NCC with reference channel (B)

4. Select the displacement with the highest NCC score
5. Apply the optimal displacements to align all channels
6. Combine into a final RGB image

Results

Cathedral

Green offset: `(x=2, y=5)`

Red offset: `(x=3, y=12)`

Computation time: `0.39s`

Monastery

Green offset: `(x=2, y=-3)`

Red offset: `(x=2, y=3)`

Computation time: `0.27s`

Tobolsk

Green offset:	`(x=3, y=3)`
Red offset:	`(x=3, y=6)`
Computation time:	`0.30s`

Multi-Scale Pyramid Alignment

For larger images, an exhaustive search becomes computationally expensive. The multi-scale pyramid approach addresses this by performing alignment in a coarse-to-fine manner:

Pyramid Alignment Steps

1. Create image pyramids for each channel by repeatedly downsampling by a factor of 2
2. Start alignment at the coarsest level (smallest image) with NCC
3. Use the result from each level to initialize search at the next finer level
4. Refine the displacement estimates through the pyramid levels
5. Apply the final displacements to the original full-resolution images

This approach significantly reduces computation time while maintaining alignment accuracy. For a typical 3-5-level pyramid, the computation time is reduced to below a minute.

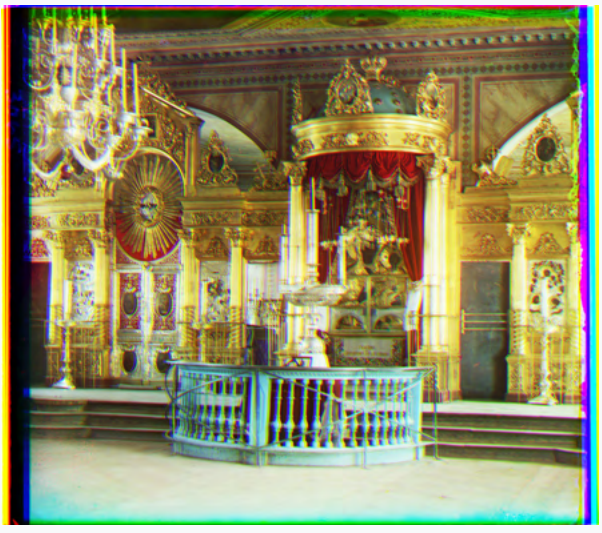
Results

Harvesters

Green offset: `(x=7, y=32)`

Red offset: `(x=24, y=27)`

Computation time: `8.14s`



Icon

Green offset: `(x=-2, y=32)`

Red offset: `(x=11, y=44)`

Computation time: `8.83s`

Church

Green offset: `(x=-24, y=27)`

Red offset: `(x=-36, y=22)`

Computation time: `8.12s`

Emir of Bukhara

Green offset: `(x=-33, y=47)`

Red offset: `(x=-56, y=78)`

Computation time: `56.33s`

Aligned with SSIM instead of NCC and cropped

Lastochikino

Green offset: `(x=-70, y=-3)`

Red offset: `(x=14, y=-17)`

Computation time: `9.88s`

Italil

Green offset: `(x=-26, y=40)`

Red offset: `(x=-49, y=68)`

Computation time: `8.64s`

Lugano

Green offset: `(x=-16, y=-31)`

Red offset: `(x=-10, y=-56)`

Computation time: `10.01s`

Self Portrait

Green offset: `(x=-10, y=56)`

Red offset: `(x=15, y=70)`

Computation time: `9.45s`

Vendor with Melons

Green offset: `(x=-2, y=21)`

Red offset: `(x=25, y=27)`

Computation time: `8.25s`

Siren

Green offset: `(x=-32, y=-11)`

Red offset: `(x=-68, y=-48)`

Computation time: `8.01s`

Three Generations

Green offset:	`(x=-24, y=27)`
Red offset:	`(x=-36, y=22)`
Computation time:	`7.66s`

Bells & Whistles

To improve results, especially for challenging images like the Emir of Bukhara, several advanced techniques were implemented:

1. Structural Similarity (SSIM) for Emir Alignment

The Emir image presented challenges due to different brightness distributions across channels. Using Structural Similarity Index (SSIM) rather than NCC improved alignment:

Structural Similarity Index (SSIM)

SSIM between two image patches is calculated as:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Where μ represents mean intensity, σ represents variance, and c_1, c_2 are stabilization constants.

Before (NCC-based)

After (SSIM-based)

2. Automatic Border Cropping

To reduce complexity, I manually cropped out about 10% of the photos to remove the non-image border colors:

Before Cropping

After Cropping

Before Cropping

After Cropping

Before Cropping

After Cropping

3. Contrast Enhancement

Applied contrast stretching to improve visual quality and reveal details:

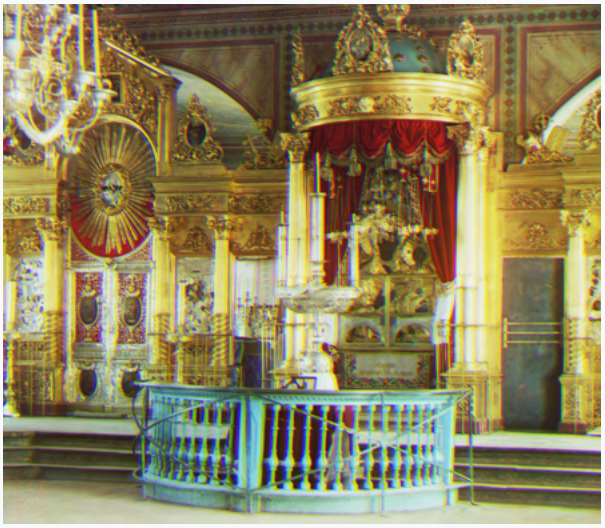
No Contrast

With Contrast

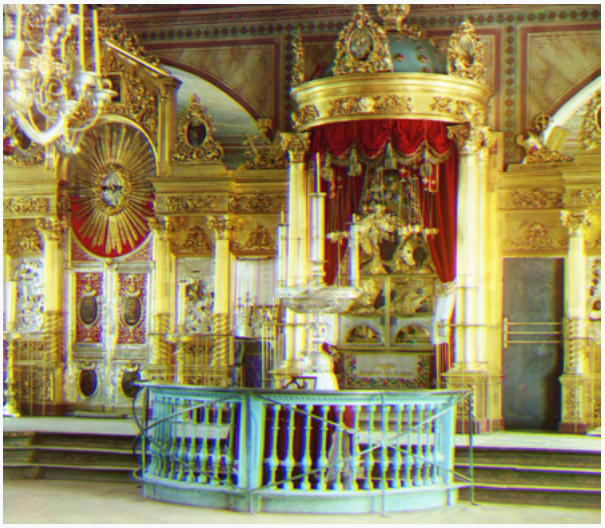
No Contrast

With Contrast

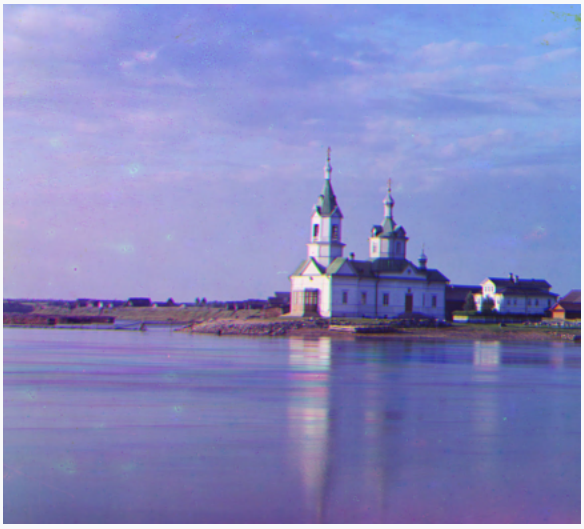
No Contrast



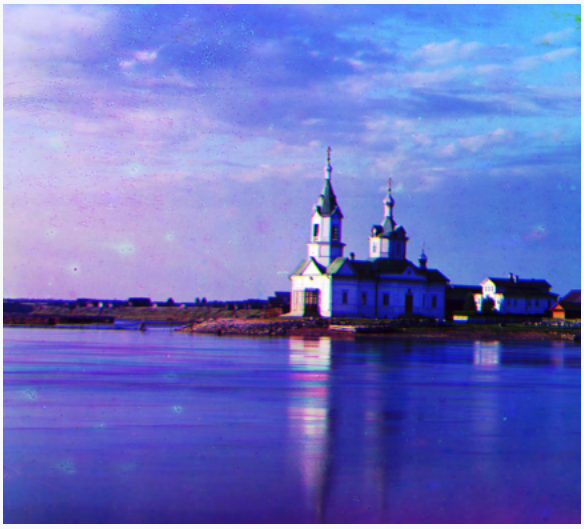
With Contrast



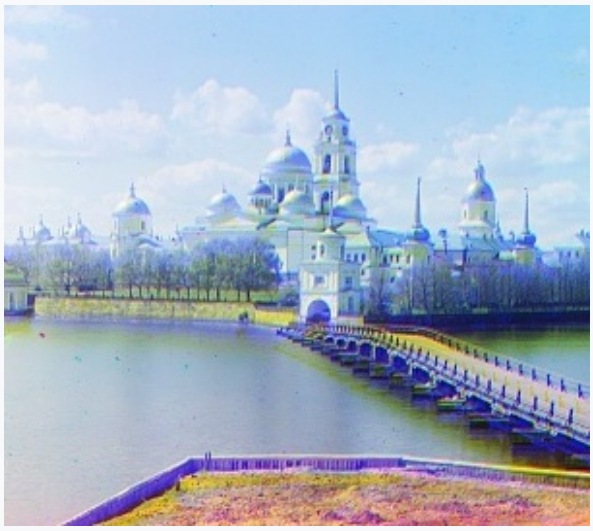
Before Enhancement



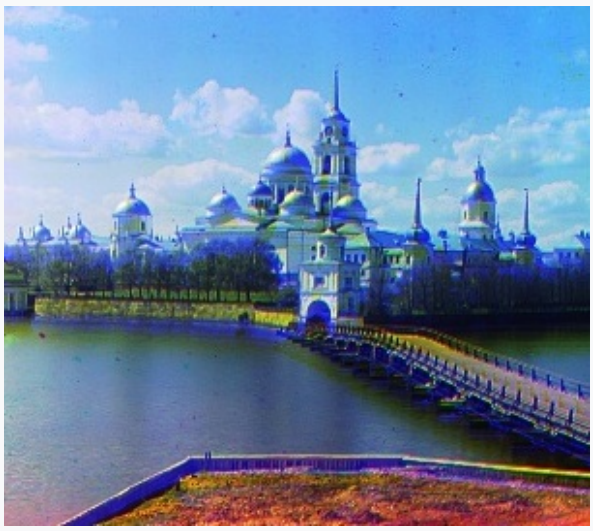
After Enhancement



No Contrast



With Contrast



No Contrast

With Contrast

Additional Results

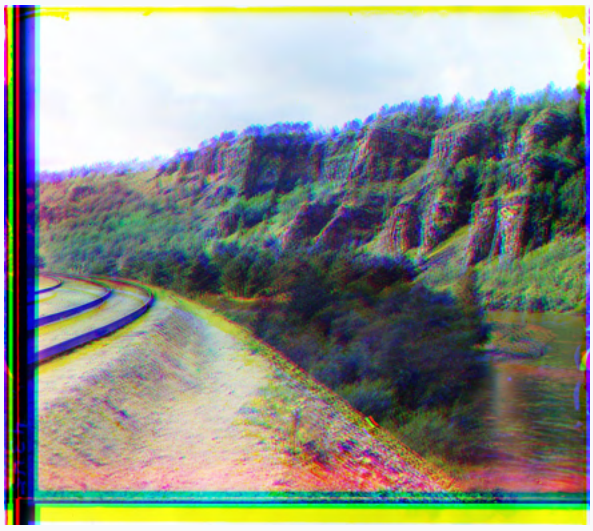
Results on other images from the Prokudin-Gorskii collection:

Napoleon

Green offset: `(x=0, y=9)`

Red offset: `(x=14, y=-3)`

Computation time: `9.93s`

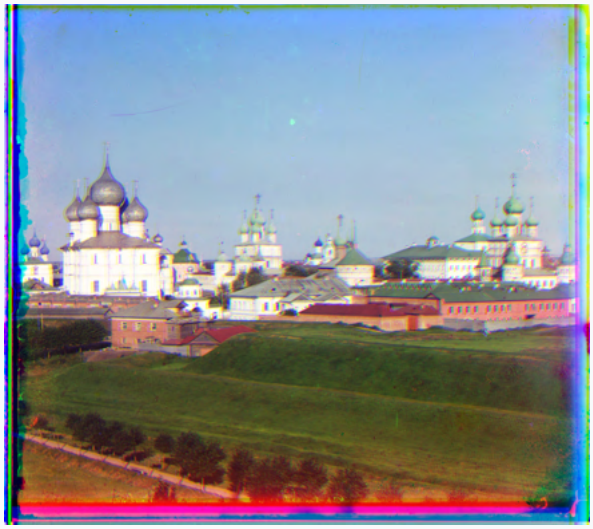


Skaly Na R

Green offset: `(x=-132, y=43)`

Red offset: `(x=-160, y=84)`

Computation time: `9.29s`



Kremlin View

Green offset: `(x=-18, y=32)`

Red offset: `(x=-20, y=49)`

Computation time: `9.21s`

Portrait of Marfy

Green offset: `(x=-13, y=32)`

Red offset: `(x=-19, y=48)`

Computation time: `10.16s`

Fresco

Green offset: `(x=-17, y=4)`

Red offset: `(x=-14, y=-26)`

Computation time: `9.05s`

CS 180/280A Project - Prokudin-Gorskii Image Colorization

Image source: Library of Congress Prokudin-Gorskii Collection