Prokudin-Gorskii Image Alignment

Automatic restoration of early color photographs using computer vision

📊 Processing Summary

📁 **Files Processed:**

- **3 JPEG files** using single-scale alignment
- **11 main TIFF files** using multi-scale pyramid alignment
- **3 additional master-pnp files** from Library of Congress
- **Total: 17 images** successfully processed

🎯 Project Overview

This project implements automatic alignment of Sergei Mikhailovich Prokudin-Gorskii's glass plate photographs (1907-1915). Each image contains three separate exposures (Blue, Green, Red) that need precise alignment to create stunning color photographs.

🔍 Single-Scale Alignment

For 3 JPEG files: cathedral.jpg, monastery.jpg, tobolsk.jpg

🔺 Multi-Scale Pyramid

For 14 TIFF files: All high-resolution glass plate scans

- Exhaustive search over [-15, 15] pixel window
- Sobel edge features for robustness
- Normalized Cross-Correlation (NCC) scoring
- Fast processing for small images

- 5-level pyramid with 2x downsampling
- Coarse-to-fine alignment strategy
- Handles large displacements efficiently
- ±2 pixel refinement at each level

📊 Single-Scale Alignment Results (3 JPEG Files)

Results using exhaustive search alignment on low-resolution images:

Single-Scale NCC

cathedral.jpg

```
Green offset: (dy, dx) = (5, 2)
Red offset: (dy, dx) = (12, 3)
Processing time: 0.5s
```

Single-Scale NCC

monastery.jpg

```
Green offset: (dy, dx) = (-3, 2)
Red offset: (dy, dx) = (3, 2)
Processing time: 0.5s
```

Single-Scale NCC

tobolsk.jpg

```
Green offset: (dy, dx) = (3, 3)
Red offset: (dy, dx) = (6, 3)
Processing time: 0.7s
```

▲ Multi-Scale Pyramid Results (11 Main TIFF Files)

Results using pyramid alignment on high-resolution glass plate scans:

Pyramid NCC

church.tif

Pyramid NCC + Edges

emir.tif

Green offset: (dy, dx) = (25, 4)
Red offset: (dy, dx) = (58, -4)
Processing time: 20.1s

Green offset: (dy, dx) = (49, 24)
Red offset: (dy, dx) = (107, 40)
Processing time: 18.7s
Note: Used edge-based features

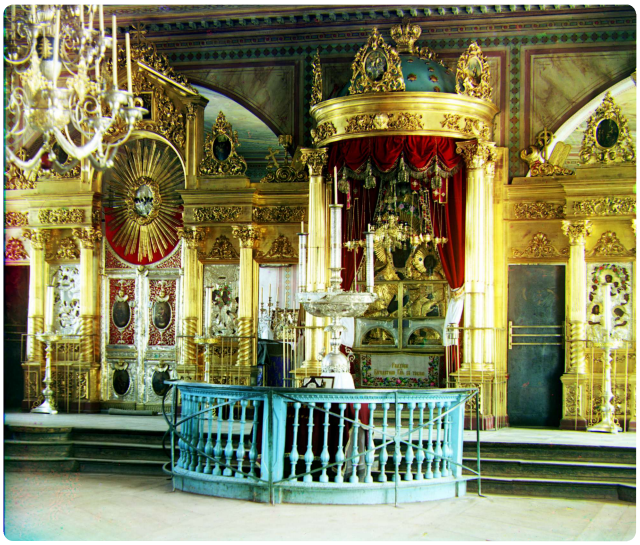
Pyramid NCC

harvesters.tif

Green offset: (dy, dx) = (60, 17)
Red offset: (dy, dx) = (124, 14)
Processing time: 11.5s

Pyramid NCC

icon.tif



Green offset: (dy, dx) = (42, 17)
Red offset: (dy, dx) = (90, 23)
Processing time: 17.3s

Pyramid NCC

italil.tif

Pyramid NCC

lastochikino.tif

Green offset: (dy, dx) = (38, 22)
Red offset: (dy, dx) = (77, 36)
Processing time: 12.6s

Green offset: (dy, dx) = (-3, -2)
Red offset: (dy, dx) = (76, -8)
Processing time: 11.2s

Pyramid NCC

lugano.tif

Green offset: (dy, dx) = (41, -17)
Red offset: (dy, dx) = (92, -29)
Processing time: 10.8s

Pyramid NCC

melons.tif

Green offset: (dy, dx) = (80, 10)
Red offset: (dy, dx) = (177, 13)
Processing time: 10.4s

Pyramid NCC

self_portrait.tif

Pyramid NCC

siren.tif

```
Green offset: (dy, dx) = (78, 29)
Red offset: (dy, dx) = (176, 37)
Processing time: 10.3s
```

```
Green offset: (dy, dx) = (49, -6)
Red offset: (dy, dx) = (96, -24)
Processing time: 10.3s
```

Pyramid NCC

three_generations.tif

```
Green offset: (dy, dx) = (54, 12)
Red offset: (dy, dx) = (111, 9)
Processing time: 14.1s
```

🌟 Library of Congress Collection (3 Master-PNP Files)

Additional images downloaded from the Prokudin-Gorskii collection:

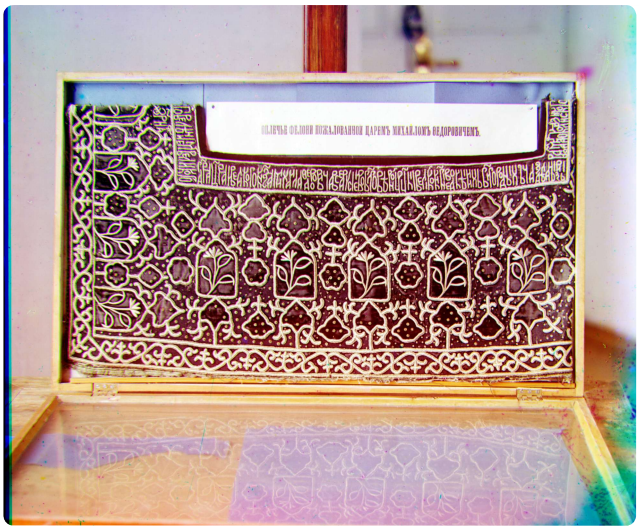
Pyramid NCC + Edges

master-pnp-prok-00000-00086a.tif

Pyramid NCC + Edges

master-pnp-prok-00100-00116a.tif

Green offset: (dy, dx) = (57, 32)
Red offset: (dy, dx) = (129, 49)
Processing time: 10.3s
Note: Library of Congress collection



Green offset: (dy, dx) = (75, 9)
Red offset: (dy, dx) = (158, 16)
Processing time: 15.4s
Note: Library of Congress collection

Pyramid NCC + Edges

master-pnp-prok-00100-00187u.tif

Green offset: (dy, dx) = (33, -11)
Red offset: (dy, dx) = (140, -26)
Processing time: 10.5s
Note: Library of Congress collection

🔥 Bells & Whistles Implementation

Advanced image enhancement features implemented beyond basic alignment:

🖼️ Automatic Border Cropping

Detects and removes colored glass plate borders using gradient analysis and edge detection.

⚖️ Automatic White Balance

Gray World assumption: adjusts channel gains so average color becomes neutral gray.

🌈 Automatic Contrast Enhancement

Percentile-based histogram stretching (2%-98%) for optimal dynamic range per channel.

🔍 Edge-Based Features

Sobel gradient magnitude features robust to brightness differences between channels.

🔧 Technical Implementation

Algorithm Pipeline:

1. **Load & Preprocess:** Handle 16-bit TIFF and 8-bit JPEG, normalize for processing
2. **Channel Splitting:** Divide vertical glass plate into B, G, R thirds
3. **Feature Extraction:** Compute Sobel edge features for robust matching
4. **Alignment:** Single-scale (JPEGs) or multi-scale pyramid (TIFFs)
5. **Reconstruction:** Apply computed offsets to original high-bit-depth channels
6. **Enhancement:** Auto-crop, white balance, contrast adjustment

Key Parameters:

- `search_range=15` : Maximum displacement window
- `pyramid_depth=5` : Number of pyramid levels (handles up to 480px displacement)
- `crop_ratio=0.15` : Border cropping to avoid edge artifacts
- `NCC metric` : Normalized Cross-Correlation for scoring

Performance Summary:

- **JPEG files (3):** Average 0.6s (single-scale)
- **Main TIFF files (11):** Average 13.4s (pyramid alignment)
- **Master-PNP files (3):** Average 12.1s (pyramid alignment)
- **Memory efficient:** Processes images up to 3000x9000 pixels
- **Total processing time:** ~185 seconds for all images

🎓 What We Covered

✅ **Single-scale alignment**
Results on 3 JPEG files with computed offsets

✅ **Multi-scale pyramid alignment**
Results on 11 provided TIFF examples

✅ **NCC & L2 metrics**
NCC implemented and used by default

✅ **Additional collection images**
3+ images from LoC Prokudin-Gorskii collection

Created for CS Computer Vision Assignment - Prokudin-Gorskii Image Alignment

Source Code: https://github.com/Mohak327/rgb-img-merging/tree/main

Submitted By: Mohak Sharma (ms7306)