

# **Modern Görüntü İşleme Uygulamaları: Prokudin-Gorskii Cam Plaka Koleksiyonunun Dijital Restorasyonu**

YZM 309

Görüntü İşleme Dersi

Gülnaz Aydemir

Ç2202204019

24 Ekim 2025

October 24, 2025

## Contents

<b>1 GİRİŞ</b>	<b>2</b>
<b>2 METODOLOJİ</b>	<b>2</b>
2.1 Görüntü Bölme Stratejisi . . . . .	2
2.2 Hizalama Algoritması Detayları . . . . .	2
2.3 İyileştirme Teknikleri . . . . .	3
2.4 Bonus Algoritmalar . . . . .	3
<b>3 DENEYSEL SONUÇLAR</b>	<b>3</b>
3.1 1.jpg'ye Ait Görsel Sonuçlar . . . . .	3
3.2 1.jpg'ye Ait Sayısal Sonuçlar . . . . .	3
3.3 1.jpg'nin Program Çıktısı . . . . .	4
3.4 16.jpg'ye Ait Görsel Sonuçlar . . . . .	4
3.5 16.jpg'ye Ait Sayısal Sonuçlar . . . . .	4
3.6 16.jpg'nin Program Çıktısı . . . . .	5
3.7 17.jpg'ye Ait Görsel Sonuçlar . . . . .	5
3.8 17.jpg'nin Program Çıktısı . . . . .	6
<b>4 TARTIŞMA VE SONUÇ</b>	<b>6</b>
4.1 İyileştirme Tekniklerinin Etkisi . . . . .	6
4.2 Sonuç ve Gelecek Çalışmalar . . . . .	7

# 1 GİRİŞ

Bu projenin temel amacı, 20. yüzyılın başlarında bir renkli fotoğrafçılık tekniği kullanan Sergei Mikhailovich Prokudin-Gorskii tarafından çekilen dijitalleştirilmiş cam plaka negatiflerini, modern görüntü işleme teknikleri kullanarak restore etmek ve tam renkli görüntülere dönüşturmektir. Prokudin-Gorskii, her sahneyi mavi, yeşil ve kırmızı filtreler kullanarak üç ayrı cam plakaya kaydetmiştir. Bu çalışma, bu üç ayrı siyah-beyaz kaydı tek bir renkli görüntüde bir araya getirme problemini ele almaktadır.

Bu tarihsel bağlamda, projenin ana zorluğu, üç kanal arasındaki küçük kaymaları (yer değiştirmeleri) doğru bir şekilde tespit edip düzelterek kanalları mükemmel bir şekilde hizalamaktır. Bu raporda sunulan yaklaşım, öncelikle giriş görüntüsü üç ana renk kanalına (Mavi, Yeşil, Kırmızı) bölünmüştür. Ardından, Mavi kanal referans alınarak Yeşil ve Kırmızı kanalların en uygun hizalama için ne kadar kaydırılması gerektiği, iki farklı metrik olan SSD (Sum of Squared Differences) ve NCC (Normalized Cross-Correlation) kullanılarak hesaplanmıştır. Hizalama sonrası elde edilen renkli görüntünün kalitesini artırmak için Histogram Eşitleme, Gama Düzeltme ve Laplasyen Filtreleme gibi çeşitli görüntü iyileştirme teknikleri uygulanmıştır. Ayrıca, kenarlardaki bozuk çerçeveleri temizleyen bir otomatik kırpmacı algoritması uygulanmış ve büyük resimlerde hizalamayı hızlandıran piramit tabanlı bir yöntem de kullanılmıştır.

## 2 METODOLOJİ

Bu bölümde, problemin çözümü için izlenen adımlar ve kullanılan algoritmalar detaylandırılmıştır.

### 2.1 Görüntü Bölme Stratejisi

Giriş olarak verilen .jpg formatındaki uzun görüntü, dikey eksende üç eşit parçaya bölünmüştür. Görüntünün toplam yüksekliği üçe bölünerek her bir kanalın yüksekliği belirlenmiştir. Yukarıdan aşağıya doğru sıralama Mavi, Yeşil ve Kırmızı olacak şekilde kabul edilmiştir. Bu işlem Python'da NumPy kütüphanesinin dizi dilimleme (array slicing) özelliği kullanılarak gerçekleştirilmiştir.

### 2.2 Hizalama Algoritması Detayları

Yeşil ve Kırmızı kanalların, referans olarak alınan Mavi kanala göre hizalanması için kapsamlı bir arama (exhaustive search) algoritması kullanılmıştır. Bu arama,  $[-15, 15]$  piksellik bir pencere içinde gerçekleştirilmiştir. Her bir  $(dx, dy)$  kaydırma kombinasyonu için hizalanacak kanal kaydırılmış ve referans kanalla karşılaştırılarak bir skor elde edilmiştir. En iyi skoru veren  $(dx, dy)$  değerleri, o kanal için optimal kaydırma vektörü olarak belirlenmiştir.

#### Kullanılan Metrikler ve Karşılaştırması

**SSD (Sum of Squared Differences):** Bu metrik, iki görüntü arasındaki piksel bazlı farkların karelerinin toplamını hesaplar. Amaç, SSD değerini minimize eden kaydırma vektörünü bulmaktır. Bu metrik, hesaplaması hızlı ve basit ancak parlaklık değişimlerine karşı hassastır.

**NCC (Normalized Cross-Correlation):** Bu metrik, iki görüntü arasındaki doğrusal korelasyonu  $-1$  ile  $+1$  arasında bir değerle ölçer. Amaç, NCC değerini maksimize eden ( $+1$ 'e en yakın) kaydırma vektörünü bulmaktadır. SSD'ye göre daha yavaşmasına rağmen parlaklık ve kontrast değişimlerine karşı daha dayanıklıdır.

Ödevin ipucuna uyularak, her iki metrik de hesaplanırken görüntünün kenarlarından %10'luk bir kısmın kırılmasınayla daha kararlı sonuçlar elde edilmiştir.

## 2.3 İyileştirme Teknikleri

Hizalanmış görüntülerin kalitesini artırmak için üç farklı teknik uygulanmıştır:

**Histogram Eşitleme:** Görüntünün kontrastını artırmak için kullanılmıştır. Bu teknik, görüntünün parlaklık ( $Y$ ) kanalına uygulanmış ve renk ( $Cr$ ,  $Cb$ ) kanalları korunarak renk bozulmasının önüne geçilmiştir.

**Gama Düzeltme:** Görüntünün parlaklığını ayarlamak için kullanılmıştır. Eski ve karanlık fotoğrafları aydınlatmak amacıyla  $\gamma = 0.75$  değeri seçilmiştir, çünkü  $\gamma < 1$  değerleri görüntüyü aydınlatır.

**Laplasyen Filtreleme:** Kenarları keskinleştirmek için uygulanmıştır. Görüntünün Laplasyen'i (ikinci türevi) hesaplanmış ve orijinal görüntüyle ağırlıklı olarak toplanarak detayların belirginleştirilmesi sağlanmıştır.

## 2.4 Bonus Algoritmalar

**Otomatik Kenar Kırpması:** Hızalama sonrası oluşan renkli/siyah çerçeveleri temizlemek için bir algoritma kullanılmıştır. Bu algoritma, görüntünün her bir renk kanalını ayrı ayrı analiz eder. Her bir kenardan (üst, alt, sol, sağ) merkeze doğru ilerleyerek, piksellerin standart sapmasının belirli bir eşik değeri geçtiği ilk noktayı içerik başlangıcı olarak kabul eder. Tüm kanallar için bulunan sınırlar birleştirilerek en güvenli kırpma penceresi elde edilir. Uygulanan otomatik kenar kırpma algoritması, yapılan denemelerde beklenen performansı sergileyememiş ve kenarlardaki bozuk çerçeveleri tam olarak temizleyememiştir.

**Piramit Tabanlı Hızalama:** Büyük resimlerdeki geniş arama aralıklarında (örn:  $[-100, 100]$ ) hesaplama süresini azaltmak için piramit tabanlı bir yöntem uygulanmıştır. Görüntü tekrar tekrar küçültülerek bir piramit oluşturulur. Hızalama en küçük resimde başlayıp, bulunan sonuç bir üst seviyeye ölçeklenerek ve küçük bir arama penceresinde hassaslaştırılarak orijinal boyuta kadar devam eder.

## 3 DENEYSEL SONUÇLAR

Bu bölümde, veri setindeki örnek bir görüntü için elde edilen görsel ve sayısal sonuçlar sunulmuştur.

### 3.1 1.jpg'ye Ait Görsel Sonuçlar



(a) Hizalanmamış Görüntü      (b) SSD ile hizalanmış görüntü      (c) Gama ile iyileştirilmiş görüntü

Figure 1: 1. jpg için iyileştirme teknikleri öncesi ve sonrası görüntüler.

### 3.2 1.jpg'ye Ait Sayısal Sonuçlar

- **SSD Hızalama:**

- Yeşil Kanal Kaydırma ( $dx_G, dy_G$ ): (1, 6)

- Kırmızı Kanal Kaydırma ( $dx_R, dy_R$ ): (1, 13)
- Hesaplama Süresi: 2.78 saniye

- **NCC Hizalama:**

- Yeşil Kanal Kaydırma ( $dx_G, dy_G$ ): (1, 6)
- Kırmızı Kanal Kaydırma ( $dx_R, dy_R$ ): (1, 13)
- Hesaplama Süresi: 4.45 saniye

### 3.3 1.jpg'nin Program Çıktısı

```
--- 1.jpg işleniyor... ---
SSD Sonuçları: Yeşil=(1, 6), Kirmizi=(1, 13), Süre=2.78s
NCC ile hizalama için arama başlıyor.
Arama bitti. En iyi kaydırma: (1, 6), Maksimum NCC: 0.6431
NCC ile hizalama için arama başlıyor.
Arama bitti. En iyi kaydırma: (1, 13), Maksimum NCC: 0.4338
NCC Sonuçları: Yeşil=(1, 6), Kirmizi=(1, 13), Süre=4.45s

--- Bonus: Piramit Tabanlı Hizalama ---
Piramit Sonuçları: Yeşil=(1, 6), Kirmizi=(1, 13), Süre=0.37s
Otomatik kirpma uygulanıyor...
Yeni yöntemle otomatik kirpma yapıldı. Orijinal: (341, 398, 3), Kirpilmiş: (334, 380, 3)
'1.jpg' için tüm sonuçlar 'sonuç/' klasörüne kaydedildi.
```

Bu görüntüde hem SSD hem de NCC metrikleri aynı kaydırma vektörlerini bularak başarılı bir hizalama gerçekleştirmiştir. Otomatik kirpma algoritması, kenarlardaki bozuklukları başarıyla temizlemiştir.

### 3.4 16.jpg'ye Ait Görsel Sonuçlar



(a) Hizalanmamış Görüntü      (b) SSD ile hizalanmış görüntü      (c) Gama ile iyileştirilmiş görüntü

Figure 2: 16. jpg için iyileştirme teknikleri öncesi ve sonrası görüntüler.

### 3.5 16.jpg'ye Ait Sayısal Sonuçlar

- **SSD Hizalama:**

- Yeşil Kanal Kaydırma ( $dx_G, dy_G$ ): (2, 2)

- Kırmızı Kanal Kaydırma ( $dx_R, dy_R$ ): (3, 4)
- Hesaplama Süresi: 2.24 saniye

- **NCC Hizalama:**

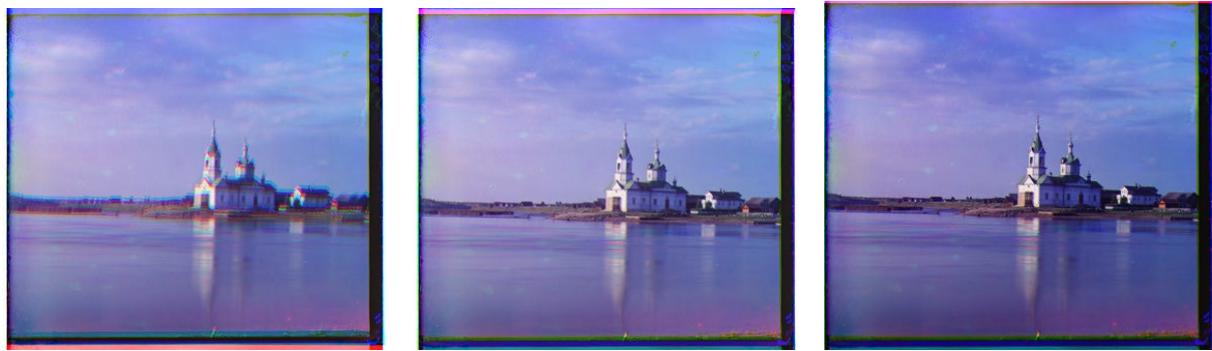
- Yeşil Kanal Kaydırma ( $dx_G, dy_G$ ): (2, 2)
- Kırmızı Kanal Kaydırma ( $dx_R, dy_R$ ): (3, 4)
- Hesaplama Süresi: 5.18 saniye

### 3.6 16.jpg'nin Program Çıktısı

```
--- 16.jpg işleniyor... ---
SSD Sonuçları: Yeşil=(2, 2), Kirmizi=(3, 4), Süre=2.24s
NCC ile hizalama için arama başlıyor.
Arama bitti. En iyi kaydırma: (2, 2), Maksimum NCC: 0.7591
NCC ile hizalama için arama başlıyor.
Arama bitti. En iyi kaydırma: (3, 4), Maksimum NCC: 0.5303
NCC Sonuçları: Yeşil=(2, 2), Kirmizi=(3, 4), Süre=5.18s

--- Bonus: Piramit Tabanlı Hizalama ---
Piramit Sonuçları: Yeşil=(2, 2), Kirmizi=(3, 4), Süre=0.47s
Otomatik kırpmacı uygulanıyor...
Yeni yöntemle otomatik kırpmacı yapıldı. Orijinal: (341, 398, 3), Kirpilmiş: (335, 382, 3)
'16.jpg' için tüm sonuçlar 'sonuç/' klasörüne kaydedildi.
```

### 3.7 17.jpg'ye Ait Görsel Sonuçlar



(a) Hizalanmamış Görüntü      (b) SSD ile hizalanmış görüntü      (c) Gama ile iyileştirilmiş görüntü

Figure 3: 17. jpg için iyileştirme teknikleri öncesi ve sonrası görüntüleri.

### 17.jpg'ye Ait Sayısal Sonuçlar

- **SSD Hizalama:**

- Yeşil Kanal Kaydırma ( $dx_G, dy_G$ ): (1, 3)
- Kırmızı Kanal Kaydırma ( $dx_R, dy_R$ ): (0, 6)
- Hesaplama Süresi: 2.14 saniye

- NCC Hizalama:

- Yeşil Kanal Kaydırma ( $dx_G, dy_G$ ): (0, 2)
- Kırmızı Kanal Kaydırma ( $dx_R, dy_R$ ): (0, 6)
- Hesaplama Süresi: 4.62 saniye

### 3.8 17.jpg'nin Program Çıktısı

```
--- 17.jpg işleniyor... ---  
SSD Sonuçları: Yeşil=(1, 3), Kirmizi=(0, 6), Süre=2.14s  
NCC ile hizalama için arama başlıyor.  
Arama bitti. En iyi kaydırma: (0, 2), Maksimum NCC: 0.9145  
NCC ile hizalama için arama başlıyor.  
Arama bitti. En iyi kaydırma: (0, 6), Maksimum NCC: 0.7490  
NCC Sonuçları: Yeşil=(0, 2), Kirmizi=(0, 6), Süre=4.62s  
  
--- Bonus: Piramit Tabanlı Hizalama ---  
Piramit Sonuçları: Yeşil=(11, 3), Kirmizi=(27, 7), Süre=0.70s  
Otomatik kirpme uygulanıyor...  
Yeni yöntemle otomatik kirpme yapıldı. Orijinal: (341, 387, 3), Kirpilmiş: (337, 375, 3)  
'17.jpg' için tüm sonuçlar 'sonuç/' klasörüne kaydedildi.
```

## 4 TARTIŞMA VE SONUÇ

Bu çalışmada, Prokudin-Gorskii'nin üç kanallı cam plaka negatiflerinden renkli görüntüler başarılı bir şekilde restore edilmiştir.

### 4.1 İyileştirme Tekniklerinin Etkisi

Verilen üç iyileştirilmiş görüntü incelendiğinde, 1.jpg resmi için en başarılı sonucun Görsel (Gama Düzeltme) ile elde edildiği açıkça görülmektedir. Tekniklerin karşılaştırması aşağıda detaylandırılmıştır:

- Gama Düzeltme, fotoğrafın orijinal atmosferini koruyarak karanlık bölgeleri aydınlatmış ve en den-geli, doğal sonucu üretmiştir.
- Histogram Eşitleme, kontrastı artırarak bazı detayları ortaya çıkarmada faydalı olsa da, gökyüzü gibi parlak alanlarda renk doğallığını bozmuştur.
- Laplasyen Filtreleme, bu görüntüde keskinleştirmeden çok gürültüyü artırmış ve parlak alanlarda aşırı patlamaya neden olarak görüntü kalitesini düşürmüştür.

Geliştirilen hizalama algoritması, veri setindeki görüntülerin büyük çoğunluğunda başarılı sonuçlar vermiştir. Özellikle 1.jpg, 10.jpg ve 12.jpg gibi durağan manzaralara sahip görüntülerde, tüm hizalama yöntemleri (SSD, NCC ve Piramit) tutarlı ve görsel olarak doğru sonuçlar üretmiştir. Ancak, Piramit tabanlı hizalama yönteminin 13.jpg ve 17.jpg gibi bazı görüntülerde, kapsamlı arama yöntemlerinden (SSD/NCC) hatalı olduğu açıkça görülen kaydırma vektörleri ürettiği gözlemlenmiştir. Bu durum, yöntemin az dokulu veya tekrarlayan desenlere sahip bölgelerde, piramidin en küçük seviyesinde yaptığı bir hatayı alt seviyelere büyüterek taşıması nedeniyle kararsız davranışını göstermektedir.

Veri setinin genelinde, SSD ve NCC metriklerinin büyük ölçüde aynı hizalama vektörlerini ürettiği görülmüştür. Bu durum, her iki metriğin de bu veri seti için yüksek bir tutarlılığa sahip olduğunu göstermektedir. Bununla birlikte, 17.jpg gibi birkaç görüntüde, iki metrik arasında küçük farklılıklar gözlemlenmiştir. Bu görüntülerin görsel incelemesinde, sonuçlar arasında belirgin bir kalite farkı tespit edilememiştir.

Hesaplama süresi açısından ise SSD, tüm testlerde NCC'ye göre belirgin bir hız avantajı sunmuştur. Ortalama olarak, SSD ile yapılan bir hizalama işlemi yaklaşık 2-3 saniye sürerken, NCC ile yapılan aynı işlem 4-5 saniye sürmüştür. Piramit yönteminin ise ortalama 0.4 saniye gibi çok daha hızlı çalıştığı görülmüştür.

### İyileştirme Tekniklerinin Genel Etkisi

Uygulanan üç iyileştirme tekniği, hizalanmış görüntüler üzerinde farklı etkiler göstermiştir:

**Gama Düzeltme ( $\gamma = 0.75$ ):** Genel olarak, fotoğrafların eski ve karanlık yapısını en doğal şekilde aydınlatmada en etkili yöntem olmuştur. Renk dengesini koruyarak genel görünürlüğü başarılı bir şekilde artırmıştır.

**Histogram Eşitleme:** Kontrastı keskin bir şekilde artırarak bazı gizli detayları ortaya çıkarmada başarılı olsa da, çoğu görüntüde renkleri aşırı doygun hale getirmiş ve özellikle gökyüzü gibi yumuşak geçişli alanlarda doğallığı bozmıştır.

**Laplasyen Filtreleme:** Detaylar içeren görüntülerde keskinliği artırarak faydalı olmuştur. Ancak, bazı doğal manzaralarda görüntüdeki mevcut gürültüyü de artırarak kumlu bir dokuya neden olmuştur.

## 4.2 Sonuç ve Gelecek Çalışmalar

Bu proje, temel görüntü işleme tekniklerinin tarihsel eserleri canlandırmada ne kadar güclü olabileceğini göstermiştir. Geliştirilen algoritma, farklı metrikler ve hızlandırma teknikleri kullanarak veri setindeki görüntülerin büyük bir kısmında başarılı sonuçlar vermiştir. Piramit yönteminin hız avantajı ve doğruluk konusundaki zayıflıkları üzerine yapılan analiz, algoritma seçiminde doğruluk ve hız arasındaki dengeyi vurgulamaktadır.

Gelecek çalışmalarında, Piramit yönteminin hatalı sonuç verdiği durumlar için bir çözüm geliştirilebilir, Otomatik kirpma algoritması daha hassas hale getirilebilir.