



İki Boyutlu (2D) Arkeolojik Parçalanmış Nesnelerin Görüntü İşleme ve Geometrik Yöntemleri Kullanarak Birleştirilmesi

Reassembly of Two-Dimensional(2D) Archaeologically Fragmented Objects Using Image Processing and Geometric Methods

Gürel YILDIZ

Kocaeli Üniversitesi
Bilgisayar Mühendisliği Bölümü
Kocaeli, Türkiye
gurel.yildiz@kocaeli.edu.tr
ORCID: 0000-0002-5350-6764

Nevcihan DURU

Kocaeli Sağlık ve Teknoloji Üniversitesi
Bilgisayar Mühendisliği Bölümü
Kocaeli, Türkiye
nevcihan.duru@kocaelisaglik.edu.tr
ORCID:0000-0003-2154-7067

Öz

Arkeologlar ya da restoratörler tarafından arkeolojik kalıntılar ile yapılan tekrar birleştirme süreçlerinde; ilk olarak kalıntı parçalarının analizi, eşleşen varsayımları tasarlama, kalıntı gerçek parçaları üzerinde bu varsayımları prova yapma ve çapraz karşılaştırma ve son olarak da öngörülen yeniden yapılandırma varsayımına göre bitişik parçaların birleştirilmesi adımları el ile yapılmaktadır. Bununla birlikte parça sayısının fazla olması, parçaların aşınması, parçaların üzerindeki desen ve renklerin silinmesi nedeniyle tekrar birleştirme adımları zaman alıcı ve zor bir süreçtir [1],[2],[3]. Bu çalışmada arkeolojik kazılar sonucunda ortaya çıkan iki boyutlu parçalanmış duvar resimleri ve mozaikler gibi kalıntı nesnelerinin dijital görüntüleri üzerinden görüntü işleme ve geometrik yöntemler kullanarak tekrar birleştirerek gerçek nesnenin görüntüsünü oluşturan bir uygulama geliştirilmiştir. Geliştirilen uygulama üzerinde farklı iki boyutlu parça görüntülerinin belirlenen kısıtlar kapsamında gerçek nesne görüntüsünü yeniden oluşturmada başarılı sonuçlar vermiştir.

Anahtar sözcükler: Arkeolojik Kalıntı Parça Görüntüleri, Görüntü İşleme, Geometrik Yöntemler, Gerçek Görüntü Oluşturma Algoritması

Abstract

In the reassembly processes made with archaeological remains by archaeologists or restorers; the analysis of fragments of the remains, designing the matching assumptions, rehearsing and cross-comparison of these assumptions on the real fragments of the remains, and finally assembling the adjacent fragments according to the proposed reconstruction assumptions are done manually. However, reassembly steps are time-consuming and exhausting because of the large number of parts, wear of the parts and deletion of the patterns and colors on the parts. In this study, an application has been developed that creates the image of the real object by combining digital images of relic objects such as two-dimensional fragmented wall paintings and mosaics, which emerged as a result of archaeological excavations, with the help of using image processing and geometric methods. On the developed application, it has given successful results in recreating the real object image within the specified constraints.

Keywords: Archaeologically Fragmented Remains, Image Processing, Geometric Methods, Real Image Generation Algorithm

1. Giriş

Arkeoloji alanında resimli kazı buluntuları hemen her zaman iki boyutlu resimsel parçalar şeklindedir. Örneğin, Şekil 1'deki

gibi orijinal nesneyi oluşturmak için bir araya getirilmesi gereken çanak çömlek, duvar resimleri veya mozaiklerdir [1].



Şekil-1: Fotoğraf çekme ortamı Apollonia-Arsuf arkeolojik bölgesinde çıkartılan seramik parçaları [4].

Antik kalıntılar kil, seramik, çini, metal, ahşap, tekstil olabilir. Örneğin kırık bir çini nesnesini geri oluşturmak için bir arkeolog her parçanın şeklini ve sınırını kontrol etmeli, orijinal görünüşünü tahmin etmeli ve her parçanın tutarlılığını test etmelidir [5]. Aynı zamanda, uzun süre ortaya çıkarılmayan bir kalıntı, yüzeyindeki boyalı orijinal pigmentini kaybedebilir. Arkeologlar için, solgun veya düşmüş eski renkleri yeniden oluşturabilmeleri de diğer çalışmalarından ve problemlerinden biridir [6]. Geçtiğimiz yıllarda, görüntüler gibi iki boyutlu veriler de dahil olmak üzere çeşitli parça türlerinin yeniden birleştirilmesinde araştırmalarda ilerleme kaydedilmiştir [7]. Mevcut yeniden birleştirme algoritmaları genel olarak renk tabanlı yaklaşımlar ve geometri tabanlı yaklaşımlar olarak iki kategoriye ayrılmaktadır [8]. Arkeolojik kalıntı görüntülerinden özellik çıkarmada elde edilen kenar ve renk bilgilerine göre geometrik olarak iki görüntüden elde edilen bütün parçaların karşılaştırılması büyük zaman karmaşıklığı çıkmaktadır. Bu zamana kadar yapılan diğer çalışmalarda karşılaştırma algoritmalarındaki hız sorunu karşılaşılan zorluklar arasındadır [9]. Bu çalışmada arkeologlar ya da restoratörler tarafından yapılan kazılarda çıkarılan kırık parçaların el ile yeniden birleştirilmesi ve oluşturulması işlemlerindeki; parçacık sayısındaki fazlalık, parçacıkların çeşitliliği, eksik parçacık, deneme yanılma ile parçacıkların eşleştirilmesi işlemleri, yanlış birleştirme işlemlerinde parçaların geri ayrılamaması problemlerine karşılık parça görüntüleri dijital ortama taşınarak görüntü işleme ve geometrik yöntemler yardımıyla doğru eşleşme ve yeniden gerçek nesne görüntüsünü oluşturan bir uygulama geliştirilmesi hedeflenmiştir.

2. Literatür Taraması

Paumard, Picard and Tabia dijital görüntü üzerinde kırma işlemi uygulayarak eşit ve düz kenarlı örnekler ile çalışmalarını gerçekleştirmişler. Bir parçanın diğerine göre göreceli konumunu tahmin edebilen bir sınıflandırıcı geliştirmeyi önermişlerdir. Sırasız bir parça listesinden yeniden birleştirme problemini çözmenin, dikkatlice tasarlanmış bir grafikte en kısa yol problemi olarak ifade edilebileceğini göstermişler.

Parçaların konumlarını belirlemek için her bir parçanın sınırlarını göz ardı ederek içeriğe odaklanmışlardır [10]. Gur ve Ben-Shahar çalışmalarında tuğla duvar bulmacası çözmek için farklı boyutlara sahip olabilen ve bitişik kenarları boyunca isteğe bağlı olarak yan yana yerleştirilebilen dikdörtgen parçalar kullanmışlardır. Çalışmalarında tuğla duvar bulmacalarında parçalar dikdörtgendir, sabit genişliğe sahiptirler ancak aynı zamanda farklı yüksekliklere sahip olabildiğinden ve bu tarz parçaların geometrik kuruluşu köşelerde birleşen kare parçalardan daha karmaşık olduğu ve yine de yeniden yapılandırma için daha az kısıtlama sağladığı belirtilmektedir [11]. Zhang, Chen, Shu, Xin, Zhao, Jin, Zhang, ve Beyerer çalışmalarında parça görüntülerinde en iyi kısmi eşleşmeyi bulmak için parça görüntüsünün dış kontur özelliklerini veya görsel ipuçlarını kullanılması gerektiğinden bahsetmişlerdir. Önceki algoritmalarında iki komşu parçanın en iyi eşleşen konfigürasyonunu tahmin etmede zayıf olduğunu göz önünde bulundurarak, uzunluk ve özellik sonuçlarına dayalı olarak birinci kontur üzerindeki bir noktayı ikinci kontur üzerindeki arzu edilen bir hedef nokta ile eşleştiren benzerliği ölçerek yaklaşımda bulunmuşlardır [12]. Amigoni, Gazzani ve Podico çalışmalarında iki parçanın yeniden birleştirilmesi, nihai görüntünün herhangi bir bilgisine dayanmadan, parçaların ana hatlarından ve renk içeriklerinden çıkarılan bilgilere dayandığını belirterek çalışmalarında parçaların ana hat piksel katarlarının biri sabit tutularak diğerinin doğrusal ötelenmesiyle karşılaştırma yapmışlardır [8]. Canyu ve Xin ise geliştirdikleri uygulamada giriş görüntü parçalarından ikili uyum adaylarını hesapladıktan sonra ikili eşleştirme yaparak çıkan sonuçlardan doğru ve yanlış eşleşmeleri yapay zeka algoritması konvolüsyonel sinir ağı (CNN) kullanarak sınıflandırmışlardır [17].

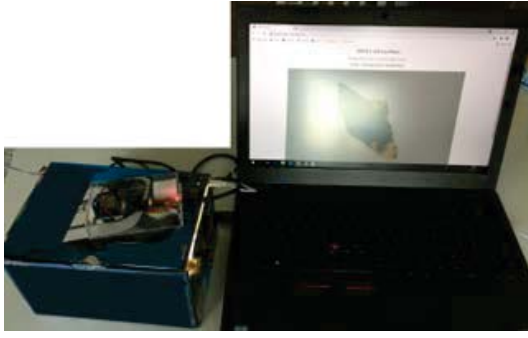
3. Yöntem

Bu çalışmada önerilen görüntü yeniden birleştirme uygulaması; görüntü edinme, görüntü kenar bölümlenme, özellik çıkarma, ikili eşleme ve görüntü genel bileşimi bölümlerinden oluşmaktadır.

3.1 Görüntü Edinme

Uygulamada kullanılacak dijital görüntülerin hazırlanması için iki görüntü edinme yöntemi kullanılmıştır.

Birinci yöntem olarak; arkeolojik kalıntılardan dijital görüntü edinme işlemi için Şekil 2'deki bütünleşik fotoğraf çekme ortamı hazırlanmıştır. Fotoğraf çekme ortamı; iki boyutlu parçaların içine konulduğu boyutları sabit ve ışıklandırma içeren kapalı kutu, kutunun üstünde belirli sabit pozisyona yerleştirilmiş ESP32-CAM Board ve FTDI Programmer modülleri, kamera ile bağlı olan bir dizüstü bilgisayardan oluşmaktadır.



Şekil-2: Fotoğraf çekme ortamı

Bilgisayara aktarılan dijital görüntüler Python programlama dilinde geliştirilen uygulama üzerinde OpenCV görüntü işleme kütüphanelerinin kırpma, yeniden boyutlandırma, bulanıklaştırma, parlaklık, arka planı kaldırma yöntemleri uygulanarak bir sonraki bölüme hazır hale getirilmiştir.

İkinci yöntem olarak; Şekil 3'deki gibi orijinal arkeolojik kalıntı resimlerinden dijital ortamda parçalama ve döndürme işlemleri uygulanarak Şekil 4'deki gibi sentetik görüntü örnekleri elde edilmiştir. Çalışma içerisinde oluşturulan parça görüntüleri img_1 , img_2 ve img_3 olarak adlandırılmıştır.



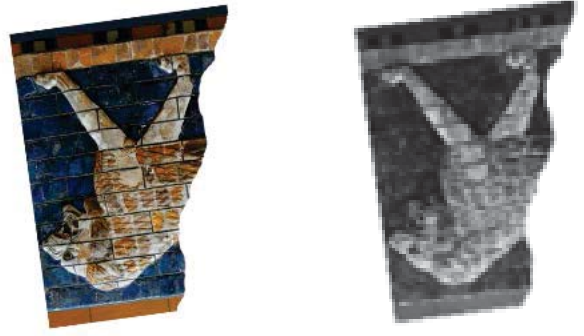
Şekil-3: Gerçek dijital görüntü [13]



Şekil-4: Dijital ortamda parçalanmış ve döndürülmüş görüntü örnekleri

3.2 Görüntü Kenar Bölümleme

Görüntü edinme bölümünde hazırlanan dijital parça görüntüleri üzerinde ilk olarak gri ölçeğe dönüştürme işlemi uygulanarak görüntü Şekil 5'deki gibi RGB renk modundan gri ölçeğe dönüştürülmüştür.



Şekil 5: Gri ölçeğe dönüştürülmüş görüntü

İkinci olarak görüntünün dış konturunu bulma işlemi uygulanarak görüntü dış sınır noktaları Şekil 6'deki gibi tespit edilmiştir. Bu işlem için OpenCV görüntü işleme kütüphanesindeki findContours fonksiyonu kullanılmıştır. Elde edilen kontur katarlarının oluşturduğu en büyük kapalı döngülü alan seçilerek görüntü dış kenar konturları elde edilmiştir



Şekil 6: Görüntü üstünde mavi renkli dış kontur

Üçüncü olarak tespit edilen dış kontur üzerinde çokgen kenar noktaları belirleme işlemi uygulanmıştır. Çokgen kenar noktaları belirleme işlemi için Douglas-Peucker algoritması kullanan OpenCV görüntü işleme kütüphanesindeki approxPolyDP fonksiyonu kullanılmıştır [14]. Bu algoritma bir eğriye veya çokgene aralarındaki mesafenin belirtilen hassasiyet değerinin eşit veya daha az olması için daha az köşeli başka bir eğri veya çokgen ile yaklaşır [15]. Fonksiyonda hassasiyet değeri görüntü kenarlarında Şekil 7'deki gibi dört nokta belirleyecek şekilde seçilmiştir.



Şekil 7: Görüntü üzerinde kırmızı renkli çokgen kenar noktaları

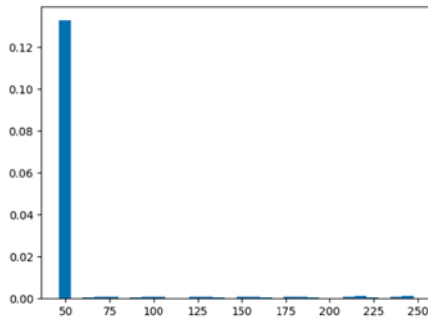
Son olarak belirlenen çokgen kenar noktalarından dış kontur eğrisi geliştirilen parçalara ayırma algoritması ile görüntü kenar parçalarına Şekil 8'deki gibi ayrılmıştır. Her görüntüden dört tane dış kontur kenar parçası elde edilmiştir. Çalışma içinde görüntü parçaları p_1, p_2, p_3 ve p_4 olarak adlandırılmıştır.



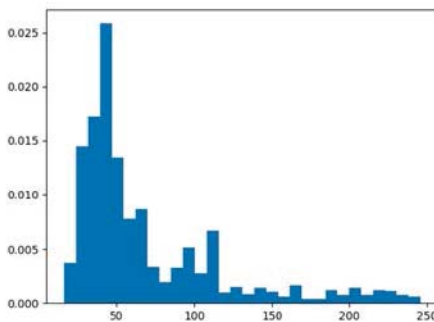
Şekil 8: Görüntü kenar parçası

3.3 Özellik Çıkarma

İki nesne görüntüsünün kenar parça piksel renk farklılık değeri histogram dağılımları kullanılarak farklılık formülü ile hesaplanmıştır. Histogram dağılımı x ekseninde gri ölçekte görüntü parça eğrisi etrafındaki beneklerin $[0,255]$ renk değerleri, y ekseninde de her değer için tekrarlanma sayısını göstermektedir.



Şekil 9: Birinci görüntünün birinci kenar parçasının etrafındaki renk histogramı



Şekil 10: İkinci görüntünün birinci kenar parçasının etrafındaki renk histogramı

Denklem (1)'de gösterildiği üzere iki görüntü kenar parçalarının renk benzerliğini bulmak için birinci görüntünün kenar parçası etrafında bulunan her renk kodunun sıklık sayısının toplamının parça uzunluğuna oranından, ikinci görüntünün kenar parçası etrafında bulunan her renk kodunun sıklık sayısının toplamının parça uzunluğuna oranın farklılık değeri hesaplanmaktadır.

$$farklilik = \sum_{i=0}^{255} \left| \frac{renk_sayısı1_i}{parça_uzunluk_1} - \frac{renk_sayısı2_i}{parça_uzunluk_2} \right| \quad (1)$$

3.4 İkili Eşleme

Şekil 11'da görüntülerin belirlenmiş kenar parçaları gösterilmektedir. İkili eşleme bölümünde karşılıklı olarak görüntü parçaları eşleştirilerek en iyi eşleşme değerleri hesaplanmıştır. Birinci görüntüden elde edilen M kenar parçası ile ikinci görüntüden elde edilen N kenar parçası birbirleriyle MN işlem ile sırayla eğrilik özelliğine göre karşılaştırılmışlardır. Karşılaştırma işlemine hazırlık olarak; önce birinci ve ikinci kenar parça görüntülerinin orta noktaları belirlenmiştir. Bu işlemden sonra birinci parça görüntüsü sabit tutularak ikinci parça görüntüsünün orta noktası birinci parça noktasının orta noktasına kaydırılmıştır.



Şekil 11: İkinci görüntünün kenar parçaları

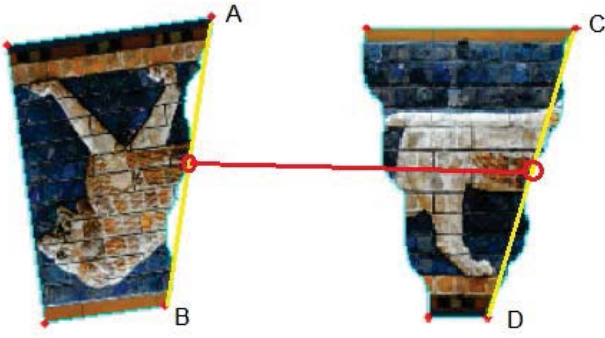
İkili eşleşme işleminde önce Şekil 12'deki gibi karşılaştırılan parçaların uç noktaları olan $A(x_2, y_2)$ ve $B(x_1, y_1)$ arasında bir doğru parçası çizilerek eğim değerleri m_1, m_2, \dots olarak Denklem (2)'deki gibi hesaplanmaktadır.



Şekil 12: Parça uç noktaları arasında çizilen doğru parçası

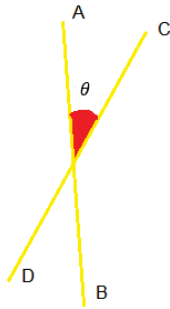
$$m_1 = \frac{y_2 - y_1}{x_2 - x_1} \quad (2)$$

İki görüntü parçaları üzerindeki doğru parçaları için m_1 ve m_2 eğim değerleri hesaplandıktan sonra Şekil 13'deki gibi iki doğru parçası orta noktalarından birbirlerinin üzerine kaydırılır.



Şekil 13: Parça uç noktaları arasında çizilen doğru parçası

Şekil 14'deki gibi orta noktalarından üst üste getirilen doğru parçalarının arasındaki açı Denklem (3)'deki gibi doğru parçalarının eğim değerleri kullanılarak hesaplanır.



Şekil 14: Doğru parçaları arasındaki açı

Açı değeri Denklem (3)'de elde edilen sonucun ters tanjant değeridir [16].

$$\tan \theta = \left| \frac{m_2 - m_1}{1 + m_1 m_2} \right| \quad (3)$$

Elde edilen açı değerine göre iki görüntü parçası birinci görüntü parçası üzerinde Şekil 15'deki gibi döndürülür.



Şekil 15: Hesaplanan açı kadar döndürülmüş kenar parçaları

İki görüntünün kenar parça uzunluklarının aynı noktalarda kopma olduğu için benzer ve yakın olduğu varsayılarak döndürme işlemi sonucunda aynı kenara ait parçaların üst üste gelerek en iyi eşleşmeyi oluşturması beklenmektedir. En iyi eşleşme hesaplaması için önce görüntü parçaları Gri ölçekte 128 ve 127 renk kodu ile çizilmiştir. Dış alanları Siyah 0 renk kodundadır. Parçalar üst üste geldiği koordinatlara göre parça noktalarının renk kodları toplanmış ve [250,255] eşik değeri ile filtrelenmiştir. Sonuçta elde edilen değerler parça uzunluklarıyla normalize edilip toplanarak eşleşme değeri oluşturulmuştur.

En iyi eşleşme değeri Denklem (4) 'deki gibi hesaplanan eşleşme değerinin farklılık değerine oranı ile hesaplanmıştır.

$$\text{En iyi eşleşme değeri} = \frac{\text{eşleşme değeri}}{\text{farklılık değeri}} \quad (4)$$

En iyi eşleşme değeri en büyük olan parçalar doğru eşleşme olarak kabul edilmiştir.

3.5 Görüntü Genel Bileşimi

Görüntülerin kenar parçalarının en iyi eşleşme değerlerinden eşleşme tablosu oluşturulmuştur. Eşleşme tablosunda en büyük en iyi eşleşme değerinden başlanarak doğru eşleşen kenar parçaları seçilmiştir. Her satırda ve sütunda seçilen eşleşme seçimine göre diğer eşleşen sonuçlar yanlış olarak kabul edilmiştir. Bu işlem sonucunda elde edilen doğru eşleşen parçalar tablosuna göre görüntüler açı ve öteleme değerlerine göre birleştirilmiştir.

4. Bulgular

Uygulama görüntü işleme ve geometrik yöntemler kullanılarak Python programlama dili ile yazılmıştır. Şekil 3'deki görüntüler uygulamaya girdi olarak kullanılmıştır. Uygulama çıktı olarak Çizelge 1'deki sonuçları vermiştir. Çizelgede img₁ ve img₂ sütunları karşılaştırılan parça görüntülerinin numaralarını, img₁ p ve img₂ q parça görüntülerinin kenar parça numaralarını ve en iyi eşleşme değeri de karşılaştırma sonucunda Denklem 4'den elde edilen sonuçları göstermektedir. Örnek birleştirilmiş görüntüler olarak Şekil 16 ve Şekil 18 en iyi eşleşme değerine göre doğru eşleşmeyi göstermektedir. Şekil 17 ise en iyi eşleşme değerine göre yanlış eşleşmeyi göstermektedir.

Çizelge-1: En iyi eşleşme sonuçları

img ₁	img ₂	img ₁ p	img ₂ p	En iyi Eşleşme Değeri
0	1	0	3	2057
0	1	3	1	1472
0	1	1	2	651
0	1	2	3	412
0	2	2	2	2974
0	2	0	0	712
0	2	3	3	584
1	2	3	0	3216



Şekil 16: En iyi eşleşme değeri: 2974



Şekil 17: En iyi eşleşme değeri: 412



Şekil 18: En iyi eşleşme değeri: 3216

Bir sonraki adımda Şekil 19'daki algoritma kullanılarak en iyi eşleşme sonuçlarından gerçek görüntü için doğru eşleşen görüntü parça listesini oluşturulmuştur.

Girdiler:

Görüntü parçalarının en iyi eşleşme sonuç listesi

Görüntü parça sayısı

Çıktılar:

Doğru eşleşen görüntü parça listesi

Prosedür:

En iyi eşleşme değerlerini büyükten küçüğe sırala

for i=0 to en iyi eşleşme liste uzunluğu do

En iyi eşleşme değerini "Doğru" ve "kontrol edilmiş olarak" işaretle

for j=0 to kontrol edilmemiş liste uzunluğu do

Doğru olan görüntü ve parça numaralarını diğer sonuçlarda ara ve o sonuçları "Yanlış" olarak işaretle

Görüntü parça sayısını bir azalt

Eğer görüntü parça sayısı sıfırsa

Durdur ve Doğru eşleşen parça listesini oluştur

Şekil 19: Sonuçlardan gerçek görüntü için doğru eşleşen görüntü parça listesini oluşturma algoritması

Algoritma adımlarında ilk olarak en iyi eşleşme değeri en büyük olan eşlemeden başlanır. Çizelge 2'de gösterildiği gibi görüntü img_1 'in parçası p_3 ile görüntü img_2 'in parçası p_0 doğru eşleşme olarak işaretlenir ve görüntü parça eşleşme listesinde başka görüntü img_1 'in parçası p_3 varsa onlar

"Yanlış" olarak işaretlenir ya da görüntü img_2 'in parçası p_0 varsa onlar da "Yanlış" olarak işaretlenir.

Çizelge-2: Eşleşme durumları (Görüntü sayısı 2)

img ₁	img ₂	img ₁ p	img ₂ p	En iyi Eşleşme Değeri	Eşleşme Sonucu
0	1	0	3	2057	Yanlış
0	1	3	1	1472	Kontrol edilmemiş sonuç
0	1	1	2	651	Kontrol edilmemiş sonuç
0	1	2	3	412	Yanlış
0	2	2	2	2974	Kontrol edilmemiş sonuç
0	2	0	0	712	Yanlış
0	2	3	3	584	Kontrol edilmemiş sonuç
1	2	3	0	3216	Doğru

İkinci adımda en iyi eşleşme değeri en büyük olan eşleme ile aynı işlem uygulanır.

Çizelge-3: Çizelge açıklaması(Görüntü sayısı 3)

img ₁	img ₂	img ₁ p	img ₂ p	En iyi Eşleşme Değeri	Eşleşme Sonucu
0	1	0	3	2057	Yanlış
0	1	3	1	1472	Kontrol edilmemiş sonuç
0	1	1	2	651	Kontrol edilmemiş sonuç
0	1	2	3	412	Yanlış
0	2	2	2	2974	Doğru
0	2	0	0	712	Yanlış
0	2	3	3	584	Yanlış
1	2	3	0	3216	Doğru

Aynı adımlar doğru bulunan görüntü sayısı girdi görüntü sayısına ulaşana kadar tekrar edilir. Uygulamanın eşleştirme sonuçlarına göre çizilmiş birleştirilmiş görüntülerin Şekil 16 ve Şekil 18'de doğru olduğu görülmektedir.

Farklı yöndeki görüntülerin Şekil 20'deki gibi üst üste birleştiği durumlarda en iyi eşleşme değerini yüksek tutarak yanlış sonuçlar çıkardığı gözlemlenmiştir. Bu durumu engellemek için görüntülerin maskeleri 49 renk kodunda oluşturulmuştur. İki görüntü üst üste geldiği zaman aynı benek rengi değeri 98 olmaktadır. Bu değer esas alınarak belirlenen eşik değere göre kontrol yapılmaktadır. Bu yaklaşım ile üst üste gelme durumundan ortaya çıkan yanlış sonuçlar engellenmiştir.



Şekil 20: Üst üste eşleşme durumu

4. Sonuç ve öneriler

Bu çalışmada, görüntü işleme ve geometrik yöntemler kullanılarak farklı yönde ve düzgün olmayan kenara sahip iki boyutlu parça görüntülerinden doğru eşleşmeler tespit edilmiştir. İki boyutlu nesne görüntüsünün kenar özelliği çıkarımı sırasında kenar parçalarını orta noktalarına göre ön açılı hesaplanarak aynı merkez etrafında döndürme işlemi başarımlarını arttırmıştır. İki boyutlu nesne görüntüsünün renk özelliği çıkarımı sırasında kenar parça üzerindeki piksellerin renk histogramının çıkarılarak kontur uzunluğuna göre normalize edilmiş değerlerinin karşılaştırılması başarılı sonuç vermiştir. Eşleşme değeri arttıkça ve farklılık değeri küçüldükçe en iyi eşleşme değeri artmaktadır. Eşleşme değeri ve farklılık değerlerinden elde edilen maksimum en iyi eşleşme değeri doğru eşleştirmeye karar vermede başarılı olmuştur. İleriki çalışmalar değerlendirildiğinde, farklı yapay zeka algoritmaları kullanılarak özellikle eksik parça tespiti ve üç boyutlu parçaların eşleştirilmesi için çalışmalar yapılabilir.

Kaynakça

- [1] Tsamoura E., Pitas I., "Automatic Color Based Reassembly of Fragmented Images and Paintings," IEEE Transactions On Image Processing, vol. 19, no. 3, 2010
- [2] Papaodysseus C., Arabadjis D., Panagopoulos M., Rousopoulos P., Exarhos M., Papazoglou E. "Automated reconstruction of fragmented objects using their 3D representation - application to important archaeological finds" 9th International Conference on Signal Processing (2008): 769-772
- [3] Papaodysseus C., Exarhos M., Panagopoulos M., Rousopoulos P., Triantafillon C., Panagopoulos T. , "Image and Pattern Analysis of 1650 B.C. Wall Paintings and Reconstruction" IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans (2008) : 958 – 965
- [4] Willis A.R., Cooper D. B., "Computational reconstruction of ancient artifacts" IEEE Signal Processing Magazine (2008): 65-83
- [5] Palmas G., Pietroni N., PCignoni P., Roberto Scopigno "A computer-assisted constraint-based system for assembling fragmented objects" Digital Heritage International Congress (DigitalHeritage) (2013, Volume: 1): 529-536.
- [6] Zheng J. Y., Zhang Z. L., Abe N. , "Virtual recovery of excavated archaeological finds" IEEE International Conference on Multimedia Computing and Systems (1998): 348-357
- [7] Liu H., Cao S., ve Yan S., "Automated assembly of shredded pieces from multiple photos," IEEE Transactions on Multimedia, vol. 13, no. 5, pp. 1154–1162, 2011.
- [8] Amigoni F., Gazzani S., Podico S., "A method for reassembling fragments in image reconstruction", in: Proceedings of ICIP, vols. 3,2, 2003, pp. III-581–4.

- [9] Leitao H. da G. ve Stolfi J., "A multiscale method for the reassembly of twodimensional fragmented objects," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, no. 9, pp. 1239–1251, Sep 2002
- [10] MM P., Tabia H P. D, (2018) "Image reassembly combining deep learning and shortest path problem". In: Proceedings of the European conference on computer vision (ECCV), pp 153–167
- [11] Gur, S., Ben-Shahar, O.: "From square pieces to brick walls: The next challenge in solving jigsaw puzzles". In: ICCV. (2017)
- [12] Zhang M., Chen S., Shu Z., Xin S., Zhao J., Jin G., Zhang R., ve Beyerer J., 2016. "Fast algorithm for 2D fragment assembly based on partial EMD". The Visual Computer 32 (2016), 1–12.
- [13] <https://muze.gov.tr/>
- [14] Wu, S.T.; Marquez, M.R.G. "A non-self-intersection Douglas-Peucker algorithm." In Proceedings of the 16th Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI), Sao Carlos, Brazil, 12–15 October 2003.
- [15] Douglas, D. H., & Peucker, T. K. "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature" The International Journal for Geographic Information and Geovisualization, 10(2), 112–122. doi:10.3138/fm57-6770-u75u-7727
- [16] Machisi, E.. 2014. "Students' Use of Multiple Solution Strategies to Find the Angle Between Two Intersecting and Non-perpendicular Lines". Mediterranean Journal of Social Sciences MCSERPublishing, Rome-Italy, 5 (6) : 309-317
- [17] Le C. ve Li X., "JigsawNet: Shredded Image Reassembly using Convolutional Neural Network and Loop-based Composition," IEEE Transactions on Image Processing, 28(8), 4000-4015, August 2019