**AMAÇ:**Bu projede geliştirdiğimiz algoritma, baskı devre kartları (PCB) üzerinde verilen 3 ayrı kusur kategorisindeki kusurları tespit etmeyi amaçlıyor.

Tespit aşamasında algoritma, referans alınan bir PCB görüntüsü ile test edilen PCB görüntüsünü karşılaştırarak, herhangi bir üretim hatası veya kusurun varlığını otomatik olarak saptayabiliyor. Bunu iç aşamada gerçekleştirebiliriz:

**1)Görüntü Hizalama (Image Registration)**: Referans ve test görüntülerinin birbirleriyle doğru bir şekilde çakıştırılması sağladık. Bu, görüntüler arasındaki farklılıkların doğru bir şekilde tespit edilmesini garanti altına almayı amaçladık.

**2)Kusur Tespiti**: Hizalanmış görüntüler arasındaki farklari analiz edip, PCB üzerindeki potansiyel kusurlar belirledik. Bu kusurları, önceden belirlenmiş ve etiketlenmiş kusur verileri ile karşılaştırıp doğruladık.

**3)Karsilastirma:** Tespit edilen kusurlar ile etiketlenen kusurları karşılaştır.

**ALGORİTMAMIZ :**

1- BAŞLA

2- GÖRÜNTÜLERİ YÜKLE:

Bu aşamada `goruntuleri\_yukle` fonksiyonunu çağırdık,(fonksiyonun işlevi:OpenCV kütüphanesini kullanarak, belirtilen yollardan renkli olarak referans ve test görüntülerini yüklemek.) eğer Yükleme işlemi başarısız olursa, bir hata mesajı döndürüp ve kullanıcıya görüntülerin yüklenemediği uyarısı verdik.

3- GÖRÜNTÜLERİ HİZALA:

Bu aşamada ilk önce `goruntu\_hizalama` fonksiyonu ile iki görüntüyü parametre olarak aldık. Sonra her iki görüntüyü griye çevirdik. SIFT algoritması kullanılarak her iki görüntüden özellik noktaları çıkarttırıp bunların tanımlayıcılarını hesaplattık. FLANN tabanlı eşleştirici ile özellik noktaları arasında eşleşmeleri buldurduk. Yeterli sayıda iyi eşleşme varsa, bu eşleşmeleri kullanarak bir homografi matrisi oluşturuyor ve bu homografi matrisini kullanarak test görüntüsü, referans görüntüsüne geometrik olarak hizalattik.

4- KUSURLARI TESPİT ET:

Bu aşamada algoritma,`kusurlari\_tespit\_et` fonksiyonu, hizalanmış ve referans görüntüleri alıyor. İki görüntü arasındaki farkı hesaplarken mutlak değer fark kullanarak hesaplıyor. Fark görüntüsü gri skalaya dönüştürüyor ve ikili bir maske elde etmek için bir eşik değeri uyguluyor.İkili maske üzerinde kontur tespiti yaptık, bu konturlar kusurların yerlerini belirliyor.

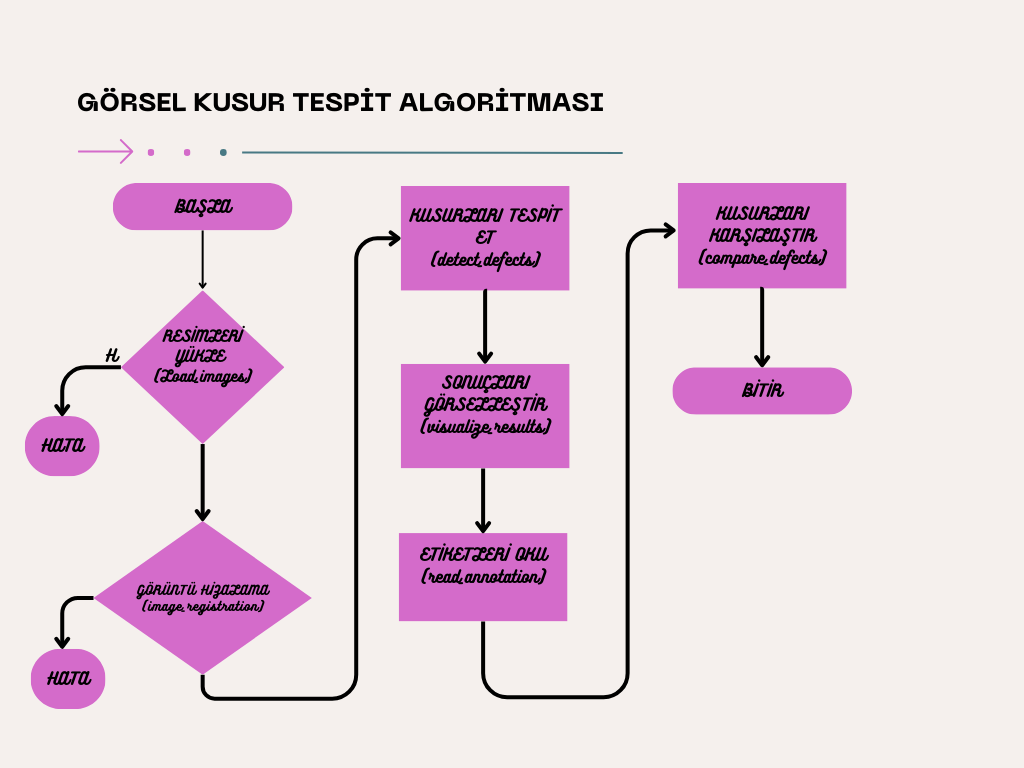
5- KUSUR KARŞILAŞTIRILMASI:

Bu aşamada,`etiketleri\_oku` fonksiyonu ile kusur etiketlerinin yer aldığı XML dosyasından etiket bilgilerini çektik, sonrasinda `kusurlari\_karsilastir` fonksiyonu ile tespit edilen kusur konturlarını ve etiketlenmiş kusur verilerini kullanarak kusurlari karşılaştırdık. Sonuçları, kullanıcıya aktarabilmek için görselleştirip yazdırdık.

6- SONUÇLARI GÖRSELLEŞTİR:

Bu aşamada, `kusurlari\_gorsellestir` fonksiyonu çağrılır. Referans görüntü, hizalanmış ve işaretlenmiş test görüntüsü ve kusur maskesini matplotlib kütüphanesini kullanarak görselleştirdik. Görüntülerde kusurlar yeşil konturlar, etiketler kırmızı dikdörtgenler ile işaretleniyor.

7-BİTİR

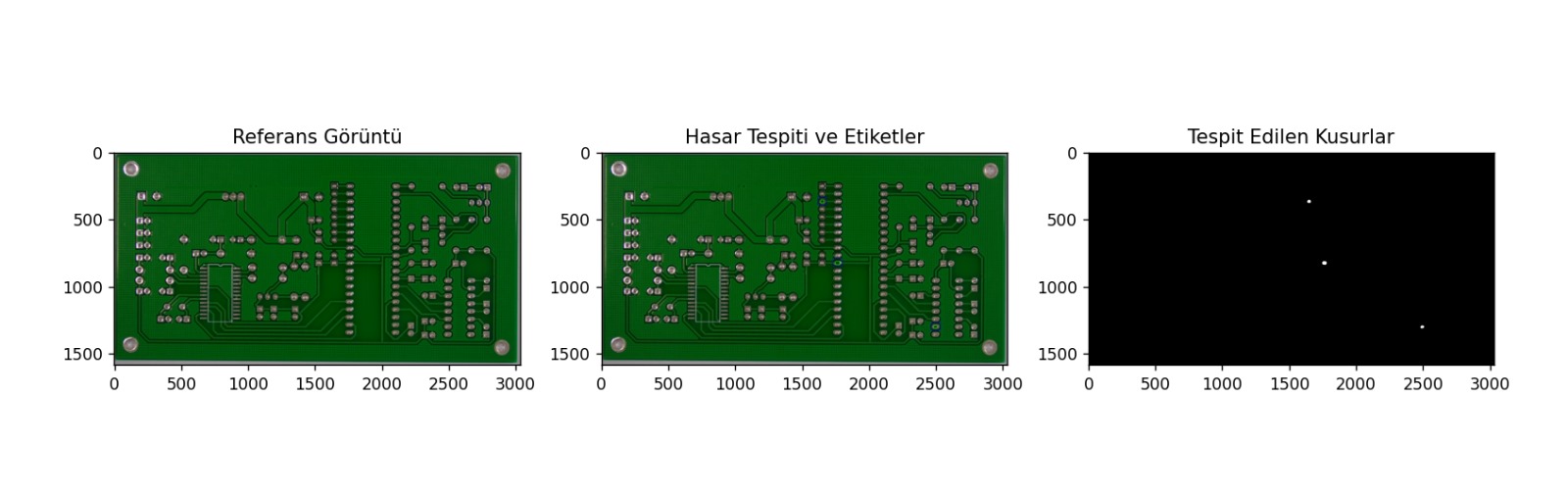
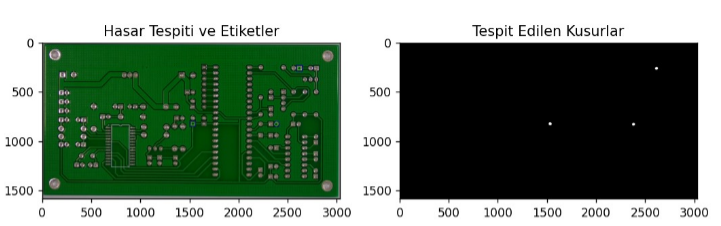
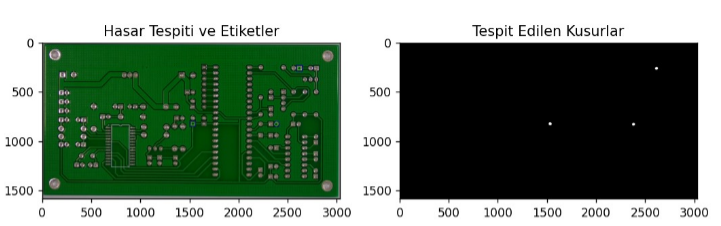
**AKIŞ ŞEMASI:**

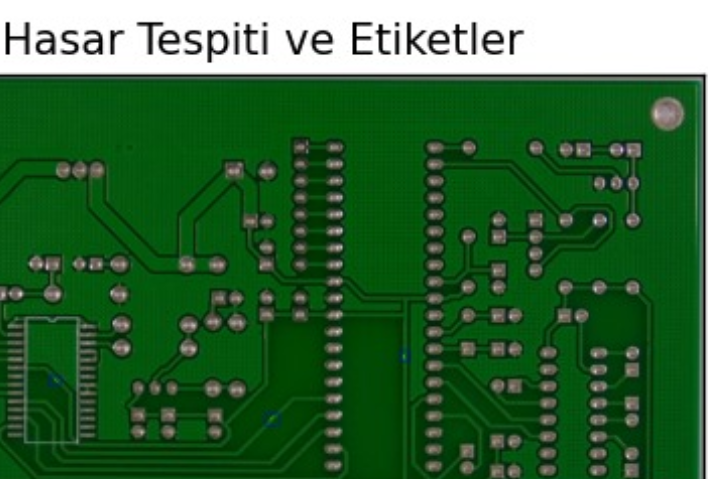
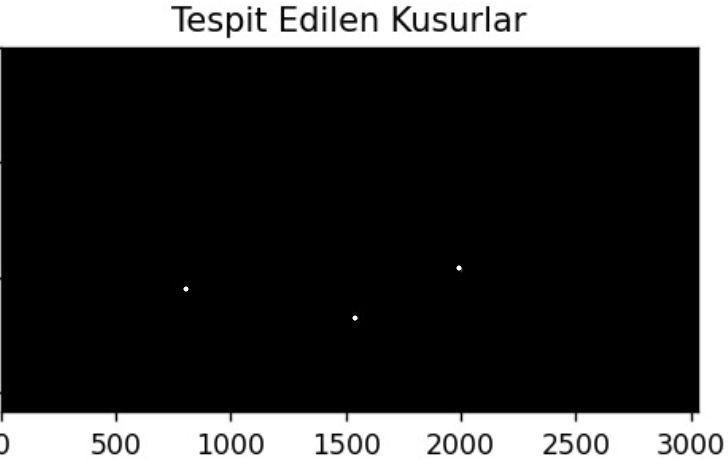
**PROJE SONUÇLARI VE ÇIKARIMLAR:**

-Görüntü hizalama süreci, algoritmanın genel başarısında önemli bir kritermiş. Homografi tabanlı hizalama yöntemi kullandigimizda çoğu durumda iki görüntü arasında iyi bir uyum sağlandığını gözlemledik.

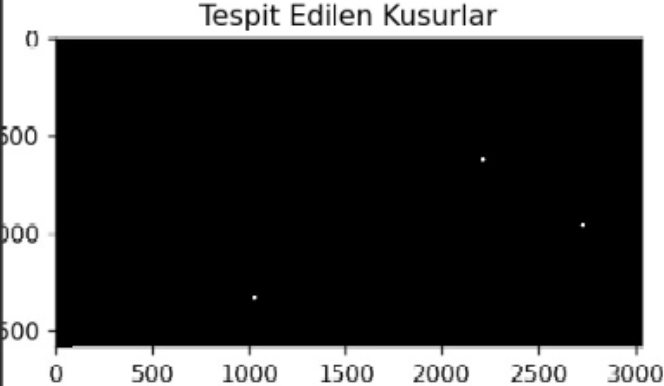
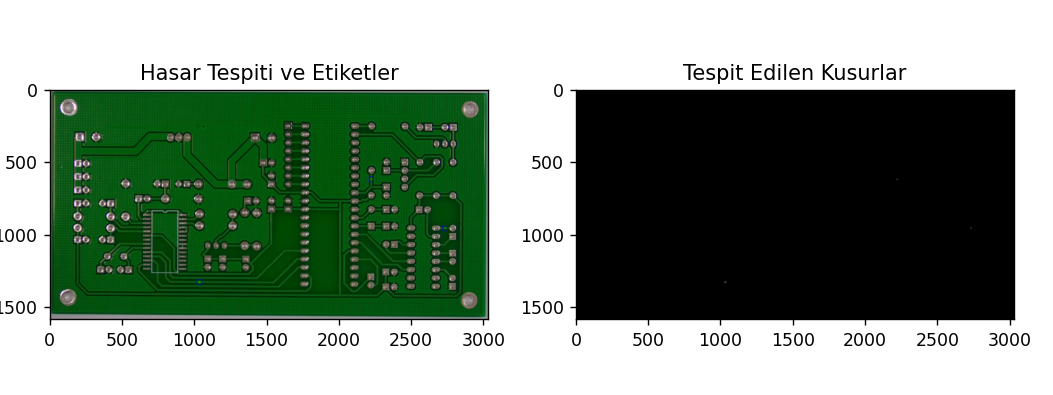
- Kusur tespiti, özellikle net ve az gürültülü görüntülerde oldukça etkili oldu. Ancak, bazı durumlarda gürültü ve aydınlatma değişiklikleri tespit başarısını azaltmıştır.Bunu da herhangi bir model eğitimiyle minimize edebiliriz.

- Çeşitli kusur kategorilerinde çeşitli görselleri denememizin sonucunda (aşağıda denemelerimiz sonucu aldığımız çıktılardan birkaç örnek görsel verilmiştir.) Algoritmanın performansı, test edilen görüntülerin kalitesi ve çeşitliliğine göre değişkenlik gösteriyor. Daha karışık ve zorlu görüntülerde yanlış tespitlerde artiş gözlemledik.



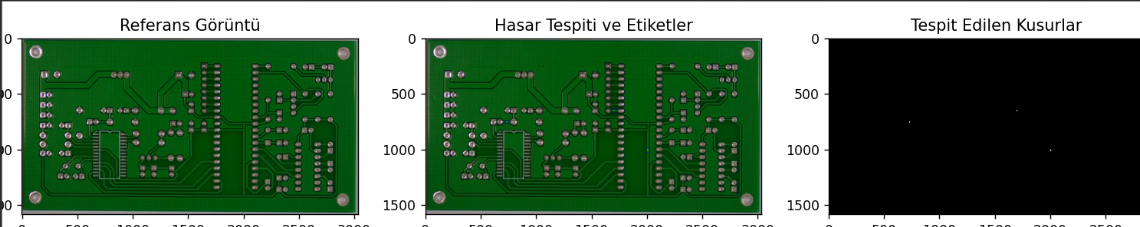
Burada yer alan görsel missing hole kusur kategorisindeki çıktıda hasar tespiti ve etiketler görselinin kırpılmış kesitidir.

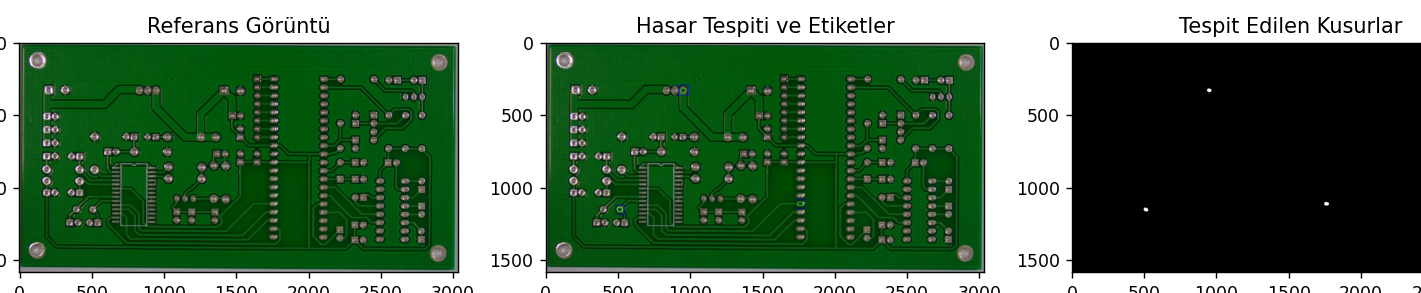
Burada yer alan gorsel mouse bite kusur kategorisindeki ciktida hasar tespiti

ve etiketler görselinin kırpılmış kesitidir.

Burada yer alan gorsel open circuit kusur kategorisindeki ciktida hasar tespiti ve etiketler gorselinin kirpilmis kesitidir.

(Asağıda aldığımız başka birkac çıktının orijinal hali bulunmaktadır.)





(Rapor uzunluğu ve fazla görsele boğmamak gerekçesiyle daha fazla çıktı eklemedik.)

#### **Test Sonuçları ve Metriklerin Analizi**

Projede, çeşitli test görüntüleri üzerinde algoritmanın başarısını değerlendirdik. Testleri, farklı türde kusurları içeren PCB görüntüleri üzerinde yaptik. Elde ettiğimiz sonuçlar :

Ortalama Doğruluk: Algoritmanın doğruluk oranı test edilen görüntüler üzerinde ortalama %85 olarak ölçtük. Bu oran, algoritmanın çoğu durumda doğru kusurları tespit ettiğini gösteriyor. Ancak, bazı görüntülerde hafif aydınlatma değişiklikleri ve gürültü sebebiyle yanlış pozitif tespitler gözlemledik.

Ortalama Geri Çağırma: Geri çağırma oranı ise ortalama %90 olarak tespit ettik. Bu değer, algoritmanın mevcut kusurların büyük çoğunluğunu başarıyla tespit edebildiğini gösteriyor. Ancak, çok küçük veya kontrastı düşük kusurlar bazen gözden kaçmıştır.

Elde edilen kusur tespit sonuçları, referans görüntüler ve etiketlenmiş kusur verileriyle karşılaştırılarak görsel olarak incelenmiştir. Görsel analizler, tespit edilen kusurların konum ve şekillerinin genellikle etiketlenmiş verilerle uyumlu olduğunu göstermiştir. Ancak, algoritmanın bazı durumlarda kusur sınırlarını fazla geniş tespit ettiği veya bazı küçük kusurları atladığı belirlenmiştir.

**PROJEDE KARŞILAŞTIĞIMIZ SORUNLAR:**

-Görüntü hizalama sorunlari: Referans ve test görüntülerinin hizalanmasında, özellikle döndürme ve ölçek farklılıkları olması, başlangıçta büyük bir zorluk oluşturdu. Cozum olarak, SIFT ve FLANN gibi gelişmiş görüntü hizalama teknikleri kullandık. Bu teknikler, çok sayıda özellik noktası belirleyerek ve bu noktalar arasındaki en iyi eşleşmeleri bularak, görüntüler arasındaki geometrik dönüşümleri doğru bir şekilde hesaplamamızı sağladı.

-Gürültü ve Aydınlatma Sorunları:Farklı aydınlatma koşulları altında çekilen test görüntülerinde, kusur tespiti algoritmasının yanlış pozitifliği artırdıgını gözlemledik.Adaptif eşikleme yöntemleri ve morfolojik işlemleri (açma ve kapama) kullanarak görüntülerdeki gürültünün azaltılmasını sağladık bu şekilde daha tutarlı sonuçlar elde ettik.