

Gömülü Sistem Destekli Yapay Sinir Ağları ile Baskı Devre Kartlarındaki Hataların Otomatik Tespiti

Şennur İBRAHİMOĞLU¹, Erdal BÜYÜKBIÇAKÇI^{2*}

¹*Alplas Industrial Investments Incorporated Company, Embedded Software Department, İstanbul, Türkiye*
²*Bilgisayar Teknolojileri Bölümü, Bilişim Teknolojileri MYO, Güç Elektroniği Teknolojileri Uygulama ve Araştırma Merkezi, Sakarya Uygulamalı Bilimler Üniversitesi, Sakarya, Türkiye*

Geliş: 5.10.2024, Kabul: 3.12.2024, Yayınlanma: 25.12.2024

ÖZ

Bu çalışmada, yapay sinir ağları ve görüntü işleme teknikleri kullanılarak baskı devre kartlarının kalite testlerinde insan gözüyle yapılan denetimlerden oluşacak zaman, iş gücü kaybı ve maliyet risklerini ortadan kaldırarak otomatik hata ayıklamayı sağlayan bir gömülü sistemin geliştirilmesi amaçlanmıştır. Teknolojik gelişmeler neticesinde geleneksel kalite kontrol yöntemleri, insan gözetimine dayalı denetimler düşük doğrulama gibi önemli sorunlar oluşturmakta ve bu sorunlar sürecin üretim safhasında devamlılığının sağlanabilmesi adına hızlı otomasyon sistemlerinin kullanımı gerekliliğini doğurmaktadır. Yapay zekâ ve görüntü işleme teknikleri akıllı otomasyon sistemlerinde etkili bir şekilde tercih edilir hale gelmiştir. Çalışmanın yapısal niteliğine özgünlük katmak açısından tasarlanan sistemde mikrodenetleyici olarak Raspberry Pi, 5 MP çözünürlüğe sahip bir kamera modülü ayrıntıların düzenlenmesi ve sınıflandırılmasında temel işlemci olarak tercih edilmiş, böylece sistemin taşınabilirliği ve maliyet durumu geliştirilmiştir. Elde edilen görüntüler bir veri tabanı aracılığı ile depolanmaktadır. Sistemdeki görüntü işleme sürecinde sayısal hale dönüştürülen veriler filtreleme, histogram, threshold, gri tonlama ve eşleme gibi yöntemler içererek üzerinde geliştirmeler yapılmıştır. Bu tekniklerle işlenen görüntüler, dijital hale getirilip iyileştirilerek hatalı ve hatasız nesne görüntüleri özellik çıkarımı yöntemiyle oluşturulmuştur. Bu ayırt edici özellik değerleri, ileri beslemeli yapay sinir ağı modeli olan çok katmanlı algılayıcı ile eğitme işlemi gerçekleştirilmiştir. Eğitme işlemlerinin ardından bulunan ağırlık katsayıları gerçek zamanlı ileri beslemeli görme algoritmasında işlenerek hatalı, hatasız ürün kontrolü yapılarak sonuçlar bulunmuştur.

Anahtar Kelimeler: Hata Tanıma Sistemi; Raspberry Pi; Görüntü İşleme; Yapay Sinir Ağı

Automated Detection of Faults in Printed Circuit Boards Using Embedded System Supported Artificial Neural Network

ABSTRACT

In this study, using artificial neural networks and image processing techniques, it is aimed to develop an embedded system that provides automatic debugging by eliminating the time, labour loss and cost risks that will arise from human visual inspections in the quality tests of printed circuit boards. As a result of technological developments, traditional quality control methods, inspections based on human supervision create significant problems such as low verification, and these problems necessitate the use of fast automation systems in order to ensure the continuity of the process in the production phase. Artificial

intelligence and image processing techniques have become effectively preferred in intelligent automation systems. In order to add originality to the structural nature of the study, Raspberry Pi was used as the microcontroller in the designed system, and a camera module with a resolution of 5 MP was preferred as the basic processor for organising and classifying the details, thus improving the portability and cost status of the system. The images obtained are stored in a database. In the image processing process in the system, the data converted into digital form are enhanced with methods such as filtering, histogram, threshold, grey scale and mapping. The images processed with these techniques are digitised and enhanced, and object images with and without errors are created by feature extraction method. These distinctive feature values were trained with a feed-forward artificial neural network model, a multilayer perceptron. After the training process, the weight coefficients were processed in the real-time feed-forward vision algorithm and the results were found by checking the defective and error-free products.

Keywords: Error Recognition System; Raspberry Pi; Image Processing; Neural Network System...

1. GİRİŞ

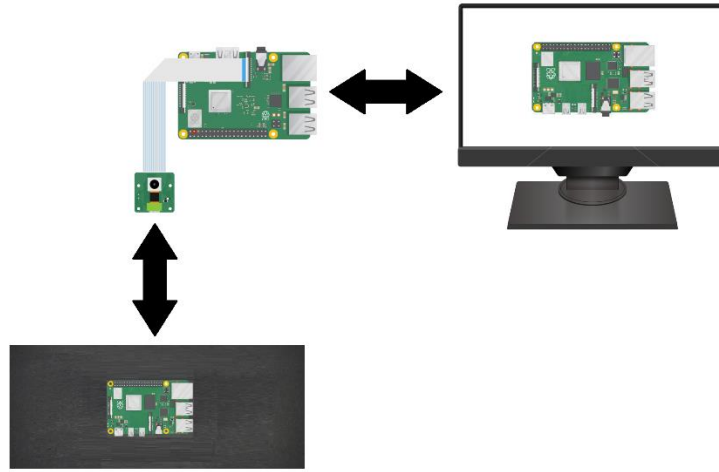
Üretim süreçlerinde kalite kontrolü ve iyileştirme için yapay zeka (YZ), özellikle yapay sinir ağları (YSA) tekniklerini kullanılabilir. Teknolojinin gelişmesiyle üretim sektörleri hızla büyüdükçe, geleneksel insana dayalı kalite kontrol sistemleri yetersiz kalmıştır. Bu nedenle, yenilikçi yaklaşımlara her zaman ihtiyaç vardır. Uzman sistemler, genetik algoritmalar ve bulanık mantıkla birlikte YSA' lar, çeşitli endüstriyel kalite problemlerini çözmek için başarıyla uygulanmıştır (Kaya, 2011). YSA tabanlı sistemler için görsel arayüzler, gerçek zamanlı uygulamalarda öğrenmeyi ve uygulamayı kolaylaştırmak için tasarlanmıştır (Bayındır, 2008). Ek olarak, YSA' lar, Box-Jenkins (ARIMA) metodolojisi gibi geleneksel yöntemlere kıyasla tahminleme de daha iyi performans göstererek onları çeşitli sektörlerde tahmini modelleme için uygun hale getirmektedir (Ataseven, 2013).

Derin öğrenme ve yapay zeka, işgücü piyasalarını ve iş operasyonlarını hızla değiştiriyor. Bu teknolojiler, kalite kontrolü ve insan kaynakları yönetimi gibi görevlerde verimliliği, doğruluğu ve hızı önemli ölçüde artırabilir. Bilgisayarlı görme uygulamalarında derin öğrenme algoritmaları, yorulmadan sürekli kalite kontrolleri gerçekleştirebilir ve potansiyel olarak üretim hatlarındaki insan uzmanlarının yerini alabilir (Korkmaz, 2020). Derin öğrenme uygulamaları doğal dil işleme, tıbbi uygulamalar, görüntü işleme de dahil olmak üzere çeşitli teknolojik ve bilimsel alanları kapsamaktadır (Seker, 2017). Görüntü işleme ve yapay zeka teknolojilerinin çok çeşitli alanlarda kullanımı giderek artıyor. Nesnel tanıma ve iyileştirilmiş görüntüler elde etmek için sayısal görüntü işleme, tasarım, imalat, güvenlik ve sağlık gibi alanlarda kullanılmaktadır (Samtaş, 2012). Örneğin, elektronik devre kartlarının üretiminde hata tespiti ve kalite kontrol süreçlerinde bu teknolojiler önemli rol oynamaktadır. Bu teknolojilerin ilerlemesi, karar alma süreçlerinde yeni yöntemlerin benimsenmesini sağlamıştır.

Yapay sinir ağları (YSA), bilgisayarlı görme ve makine öğrenimi gibi teknolojilerin yaygın kullanımıyla karar alma sistemleri için önemli bir yapı haline geldi (Bayındır, 2008). Yazılım geliştirme, görüntü işleme teknikleri, öğrenme algoritmaları ve ışık koşulları bu sistemlerin nesne tespitindeki başarısını etkiler (Gültekin, 2019). Son çalışmalar, görüntü işleme yöntemlerinin ve yapay zekanın (YZ) kalite kontrol ve hata tespit sistemlerinde giderek önemli bir yere sahip olduğunu göstermektedir. Geleneksel el yordamıyla kontrol edilenlere göre bu yöntemler daha yüksek doğrulama, daha hızlı sonuç süresi ve daha kapsamlı değerlendirmeler sunar (Çelik, 2020). Şirketlerin verimliliği, kalite kontrolünü ve üretim planlamasını kolaylaştırmak için yapay zeka teknolojisinin kullanımı gün geçtikçe daha yaygın kullanılır hale geliyor (Ever, 2022).

Bu çalışmanın amacı, özellikle endüstride kullanılması amacıyla bir ürünün, malzemenin veya nesnenin hatalı yada hatasız olduğu durumları belirleyerek ayrıca hatalı ürünler için uyarı verecek bir sistem oluşturmaktır.

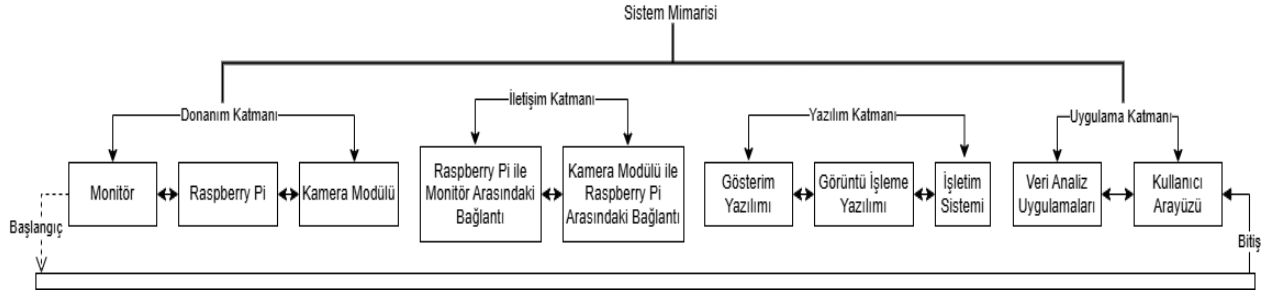
Çalışmada kullanılan yazılımlar, hedef alınan ürünün özellik çıkartımı ile elde edilen veri tabanını güncelleyebilir ve çeşitli malzemelerin kontrolünü sağlayabilir. Veri tabanı, bu sistem için tasarlanmış ve değiştirilebilir bir yapıya sahiptir. Şekil 1’ de görülen donanım düzeni, bu kontrol sisteminin temel bileşenlerini göstermekte olup, kamera, mikrodenetleyici ve ekran arasında veri akışını sağlayarak nesne analizi sürecini desteklemektedir.



Şekil 1: Hata tespit sisteminin genel şeması

2. MATERYAL VE YÖNTEM

Hata tespiti yapılması hedeflenen malzemenin görüntüsünü oluşturabilmek için sensör olarak 5 MP, 2592x1944 piksel değerinde çözünürlük içeren Raspberry Pi 3 kamera modülü kullanılmıştır. Mikrobilgisayar olarak Raspberry Pi 3 seçilme nedeni ise kamera modülünden elde edilen görüntü üzerinde gerekli algoritmaların çalıştırılması, verilerin kaydedilerek yeni veriler ile karşılaştırılması amaçlanmıştır. Hata tespiti gerçekleştirilirken gerçek zamanlı olarak verilerin Raspberry Pi 3 programı üzerinde kurgulanmış yapay sinir ağları çıktı verileriyle beraber çalıştırılan algoritmalarla karşılaştırılarak, değerlendirme sonuçlarının Raspberry Pi HDMI çıkışına bağlanan monitöre aktarılıp takibi sağlanmaktadır.



Şekil 2: Sistemin kurulum mimarisi

Sistemin genel olarak kurulum mimarisinin işleyişi Şekil 2’ de verilmiştir. Bu blok mimaride, donanım katmanı fiziksel bileşenleri içererek temel işlevleri tanımlamaktadır. Bu katmanda bulunan; Kamera Modülü, görüntüleri yakalayan bileşendir, çekilen görüntüler, veri akışı yoluyla Raspberry Pi’ye iletilir. Raspberry Pi, sistem merkezi olarak çalışmakta ve verileri alarak, işler ve monitöre gönderir. Ayrıca, işlemci, hafıza ve diğer yardımcı bileşenleri de içermektedir. Monitör, işlenen görüntüleri veya çıktıları kullanıcının görmesini sağlar.

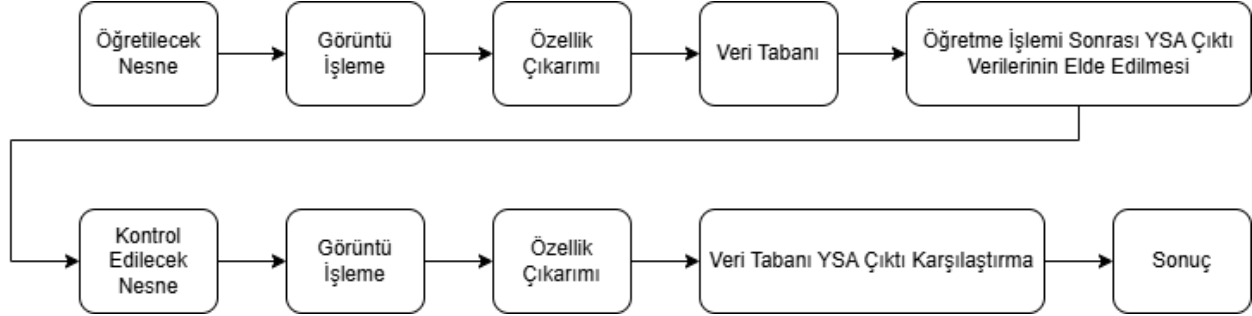
İletişim Katmanı, donanımlar arasındaki veri akışını ve kullanılan protokolleri tanımlamaktadır. Bu çalışmadaki iletişim, veri aktarımı yoluyla sağlanmaktadır. Bu katmanda, Kamera Modülü ile Raspberry Pi Arasındaki Bağlantı, kamera modülünden Raspberry Pi’ye veri akışını sağlamaktadır. Bu bağlantı genellikle CSI (Camera Serial Interface) protokolü üzerinden yapılmıştır. Raspberry Pi ile Monitör Arasındaki Bağlantı, Raspberry Pi’den monitöre veri aktarımı HDMI bağlantı kablosu üzerinden gerçekleştirilmiştir.

Yazılım Katmanı, donanımı kontrol eden ve veri işleyen yazılımları içermektedir. İşletim Sistemi, Raspberry Pi üzerinde sürülen Linux tabanlı bir işletim sistemi (örneğin, Raspberry Pi OS). Bu işletim sistemi donanım bileşenlerini yönetmekte ve uygulamaların çalışmasını sağlamaktadır. Görüntü İşleme Yazılımı, Raspberry Pi üzerinde çalıştırılan görüntü işleme veya nesne tanıma yazılımları olarak ifade edilmektedir. Bu yazılım, kameradan gelen ham veriyi işlemekte ve monitöre gönderilecek hazır hale getirmektedir. Gösterim Yazılımı ise işlenen görüntüleri monitörde gösteren yazılım bileşenidir. Bu, genellikle Grafik Kullanıcı Arayüzü (GUI) içeren bir uygulamadır.

Uygulama Katmanı, kullanıcının sistemle etkileşime geçtiği ve spesifik görevleri yerine getirdiği uygulamaları içermektedir. Kullanıcı Arayüzü, kamera ile çekilen görüntülerin monitörde görüntülenmesi, işlenmiş verilerin kullanıcıya sunulması gibi işlevleri içerir. Veri Analiz Uygulamaları ise görüntü işleme sonrası elde edilen bilgilerin analiz edilmesi, yorumlanması veya kaydetmesi gibi görevleri içermektedir.

Raspberry Pi’de görüntü işlemek için görüntü işleme kütüphanesi olarak Python OpenCV (Open Source Computer Vision), Matlab yazılımdan ise Yapay sinir ağlarının (YSA) kullanımında yararlanılmıştır. Hata

tespit sistemlerinde sistem yazılımları bütünü en önemli parçasını temsil etmektedir. Geliştirilen sistemin hata tespitini sorunsuz olarak yerine getirebilmesi için yazılımın çevre girdilerinin iyi olması gerekmektedir. Şekil 3’de genel olarak geliştirilen hata tespit sistemin algoritması oluşturulmuştur.



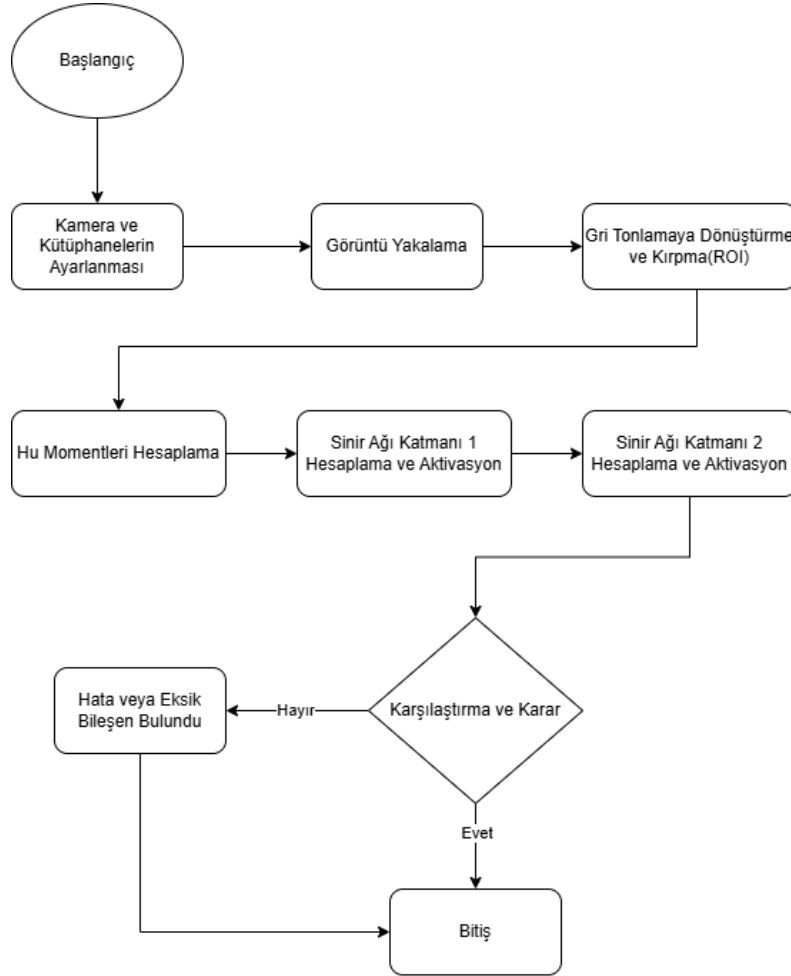
Şekil 3: Hata tespit genel algoritması

Şekil 3’ deki çalışma adımları açıklanacak olursa; örnek görüntü eğitilecek nesne algoritmasını kullanarak RPi kamera modülü ile renkli dijital olarak elde edilir. Python dilinde görüntü işleme teknikleri ve OpenCV açık kaynak kod kütüphanesi, elde edilen verileri kullanarak istenilen iyileştirmeleri yaparak görüntülemeyi geliştirir. Görüntü üzerinde sırasıyla yapılan dönüşümler; gri düzeye dönüştürme, histogram eşitleme, medyan filtreleme fonksiyonları ve eşikleme işlemleri kullanılarak görüntü üzerinde istenilen düzenlemeler gerçekleştirilir. Görüntü işleme işlemleri ile oluşturulan veriler belli bir dosyada saklanmak üzere kaydedilir. Bu kaydı sağlanan görüntüler için ayırt edici olacak, şekil ve piksel özellikleri de bu aşamada ortaya çıkartılır. Sistemin eğitime işlemi Çok Katmanlı Algılayıcı (ÇKA) YSA’ na bu veriler girdi olarak tanıtılarak gerçekleştirilir. Aynı zamanda hata tespiti algoritmasında da aynı işlemler uygulanır. Görüntü kaydedilerek görüntü özellikleri, özelliklerin aktarımı, görüntü işleme teknikleri de kullanılarak istenilen işlemler ve değişiklikler yapılır. Son olarak ise önceden öğretilen nesnelerin ÇKA ağırlık sonuçları, aynı ÇKA algoritması kullanılarak öğretilen nesne ve kontrol edilecek nesne ile önceden öğretilen nesneler arasındaki karşılaştırmalar yapılarak sonuç elde edilir.

2.1. Algoritmanın Pseudocode ve Akış Diyagramı ile Tanımlanması

Bu bölümde, PCB üzerindeki bileşenlerin doğruluğunu kontrol etmek için geliştirilen algoritmanın işleyişi hem pseudocode hem de akış diyagramı ile tanımlanmaktadır. İlk işlem olarak, görüntü ve kamera işleme ayarları yapılandırılarak PCB görüntüsü alınır ve gri tonlamaya çevrilir. Daha sonra ilgi bölgesi (ROI) kırılarak, Hu momentleri hesaplanır ve bu özellikler sinir ağına giriş verisi olarak kullanılır. İki katmanlı sinir ağı yapısında, ilk katmanda Hu momentleri ağırlıklarla çarpılarak aktivasyon fonksiyonuna tabi tutulur ve ara çıktılar elde edilir. İkinci katmanda bu çıktılar yeni ağırlıklarla çarpılarak nihai sonuç üretilir. Karar

aşamasında, PCB üzerindeki bileşenlerin doğruluğu değerlendirilir ve tüm bileşenlerin doğru yerleştirildiği veya hata durumu hakkında bilgi sağlanır. Bu süreç pseudocode ve akış diyagramı aracılığıyla Şekil 4 ve Şekil 5’ de verilmiştir.



Şekil 4: Hata tespit pseudocode genel algoritması

BEGIN

Kamerayı başlat ve gerekli ayarları yapılandır

Görüntüyü yakala ve kaydet

Yakalanan görüntüyü yükle ve gri tonlamaya çevir

İlgi bölgesini (ROI) kırp

Hu momentlerini hesapla

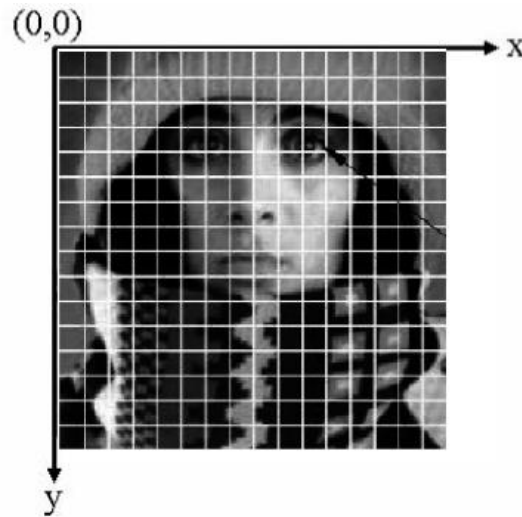
FOR i FROM 0 TO 24 DO

```
L[i] = SUM(hu[j] * iw1[i][j] FOR j FROM 0 TO 6) + b1[i]
L[i] = (2 / (1 + exp(-2 * L[i]))) - 1 # Aktivasyon
END FOR
FOR i FROM 0 TO 1 DO
  L1[i] = SUM(L[j] * iw2[i][j] FOR j FROM 0 TO 24) + b2[i]
  L1[i] = (2 / (1 + exp(-2 * L1[i]))) - 1
END FOR
IF L1[0] > L1[1] THEN
  Print "PCB kartında tüm bileşenler doğru yerleştirilmiş"
ELSE
  Print "PCB kartında bileşen hatası var"
END IF
END
```

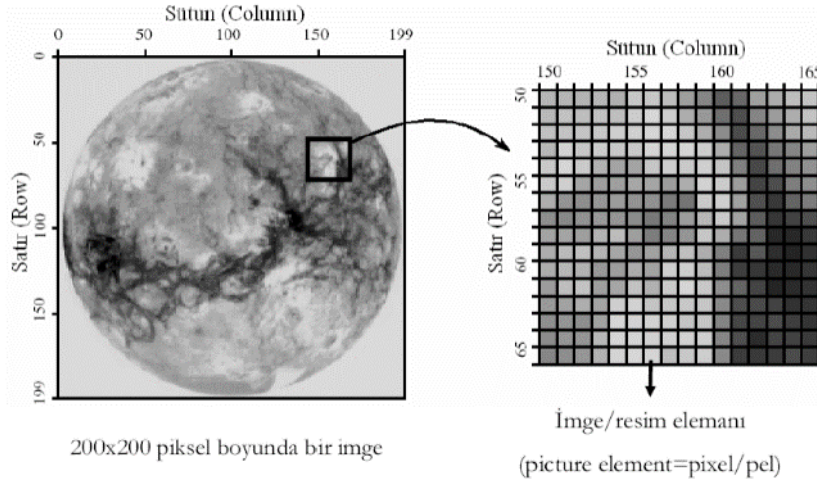
Şekil 5: Hata tespit pseudocode

2.2. Görüntü İşleme

Görüntü işleme işlemi görüntüyü dijital formata dönüştürerek belirli modelleri oluşturarak, bunlardan faydalı bilgiler çıkarma sonucunda bazı istenilen işlemleri gerçekleştirmek amacıyla geliştirilen bir tekniktir. Şekil 6' da görüntüsü iki boyutlu olarak görülen nesnenin şık miktarını sıklık fonksiyonu olarak da tanımlayabiliriz. Ayrıca matematiksel olarak da $f(x, y)$ fonksiyonu olarak gösterilebilir. Oluşturulan bu görüntüler piksel değerleri olarak ifade edilir. Herhangi bir (x, y) değeri için keşiştiği pikselin ışık enerjisi ise görüntü sinyalinin değeridir (Gonzalez, 1992).



Şekil 6: Görüntünün iki boyutlu ifadesi



Şekil 7: Görüntünün piksel ifadesi

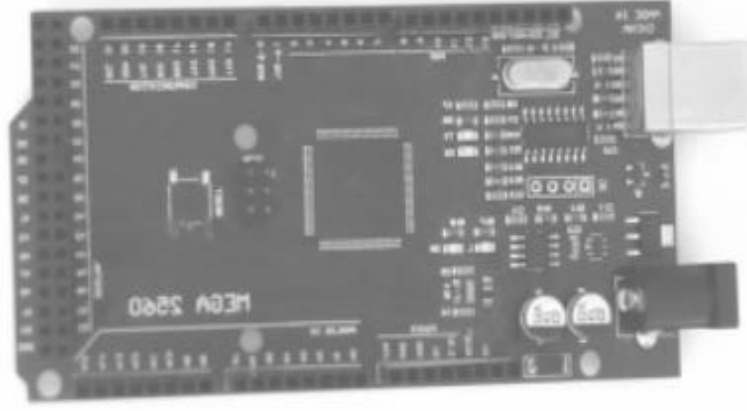
Piksel değerleri üzerindeki değişimler Görüntü üzerinde yapılan bütün işlemler sonucunda elde edilir. Görüntüde yapılan düzenlemeler piksel değerlerindeki değişimlerin miktarı ile elde edilir (Çetin, 2024). Görüntü üzerinde yapılan önemli düzenlemeler sonrasında, ilgili resim üzerinde yer alan nesnelere daha belirgin, net ve kolay işlenebilir hale gelmektedir.

Bu çalışmada görüntü iyileştirme teknikleri olarak gri seviyeye dönüştürme, medyan filtreleme, eşikleme (threshold) yöntemleri kullanılmıştır. Birçok alanda (astronomi, robotik, tıp ve imalat gibi) renkli bölümlerin grileştirilmesi, dönüştürülmesi popüler görüntü işleme tekniklerinden biridir. Bir piksel, renkli bir görüntüde Red (R), Green (G) ve Blue (B) renk paketlerini içerir. Bu renklerin değerlerinin ağırlıklı ortalaması alınarak renkli görüntüyü gri seviyeye çevirmek işleminde kullanılmaktadır. Aşağıdaki Denklem 1’ de görülen eşitlikte gri seviye ağırlıklı ortalamasının alınması ifade edilmiştir. Denklem 1’ de r, g ve b sembolleri deęer olarak bir görüntüden elde edilen 0-255 arası renk deęer aralığını göstermektedir.

$$x=(0.299\times r)+(0.587\times g)+(0.114\times b) \quad (1)$$

Denklem 1’ de renk deęer katsayılarının aynı olamadığı ifade edilmektedir. Bu durumun sebebi görüntülerde renk tonlarının farklı olmasıdır. Mavi rengi daha soęuk renk tonuna, Yeşil rengi ise daha sıcak açık renk tonuna sahiptir. Bu nedenledir ki mavi rengin katsayısı en düşük deęere sahip olmaktadır (Saravanan, 2010).

İki renk, siyah – beyaz ve gri seviye görüntülerde yer alır. Bu durumda her piksel için deęer 8 bit olarak kodlanmaktadır. Gri rengin tam sayı olarak deęer aralığı 0-255 arasında olmaktadır. Renk deęerleri kodlamaları gri seviye renk deęerleri için, beyaz (255,255,255), siyah (0, 0, 0) ve orta seviye gri (127,127,127) şeklinde ifade edilmektedir.



Şekil 8: Gri seviyeye indirgenmiş Elektronik kart görüntüsü

Bir resim üzerinde, görüntüdeki bir piksel üzerinde işlem gerçekleştirildiğinde bu pikselin ve ona komşu pikselinin parlaklık değerleri işleme medyan filtreleme yöntemi yapılarak alınmaktadır. Orta parlaklık değeri, değerlendirilmesi yapılan bu pikseller içerisindeki değeri yeni parlaklık değeri olarak atanmaktadır. İşlem yapılan piksel sonuçlarında orta değerlerin seçilmesinde hedef bu yöntem kullanılarak, minimum ve maksimum değerlerin etkinliğinin düşürülmesi amaçlamaktır. Orta değeri dizinin orta sayısına karşılık gelen değer olarak belirleyecek piksel değeri küçükten en büyüğe doğru dizilmiş olmalıdır (Horozoglu, 2013).

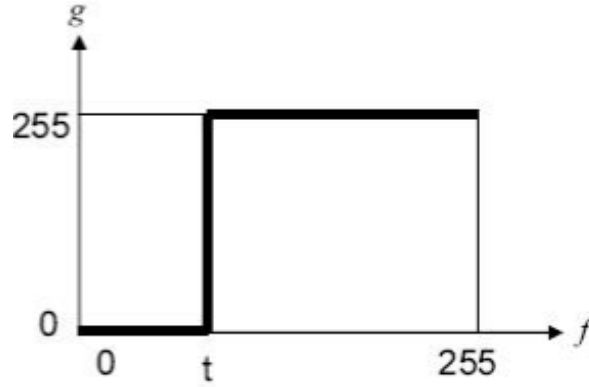
Medyan filtreleme, görüntü işleme alanında dürtüsel veya tuzbiber gürültüsünü azaltmada oldukça etkili ve doğrusal olmayan alçak geçiren bir filtre tipi olarak kullanılmaktadır (Akar, 2006; Yu, 1990). Medyan filtresi kullanarak görüntüdeki tespit edilen gürültüler ortadan kaldırılarak ayrıştırıldıktan sonra eşikleme yoluyla görüntüdeki ayırt edici bölümler arka plandan ayrılmıştır. Böylece görüntü 8 bit gri değerden 2 bit (0 veya 255) indirgenmiştir. Bu fonksiyon, cismi görüntünün arka planından ve birbirinden ayırmak için görüntü işleme teknikleri içerisinde oldukça fazla kullanılmaktadır (Saravanan, 2010). Eşikleme işleminin dijital görüntüye uygulanmasındaki amaç detayların ön plana çıkarılarak cisim özelliklerinin elde edilerek ayırt edici bölümlerin, noktaların belirlenmesinde kolaylık oluşturmaktır. Eşleştirme yöntemi, görüntünün iki renkle gösterilebilecek şekilde formata dönüştürülmesidir.

Eşik değeri görüntüye eşikleme işlemi uygulanmadan önce belirlenir. Bu eşik düşük olan piksellere 0, fazla olanlara ise 1 değeri atanır. Bu değer atamaları neticesinde görüntü siyah ve beyaza dönüşüm sağlar. İşlemlerin sonucunda, hatalı olan bölge arka plandan tamamen ayrı olarak görülebilir hale getirilmiş olur.

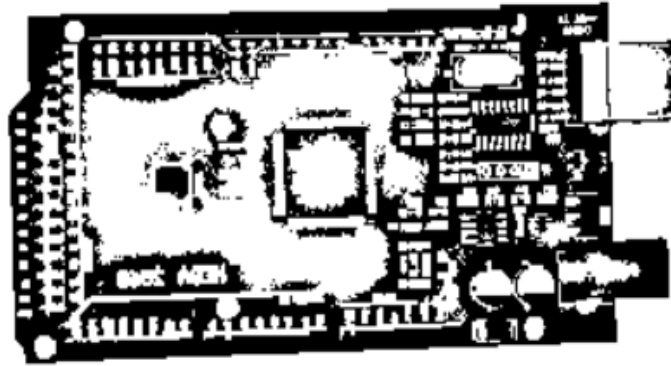
Görüntü işleme tekniklerinde en yaygın yöntemlerden biri eşiklemedir. Genellikle görüntüdeki arka plan ile nesneyi birbirinden ayırmak için tercih edilir. Dijital görüntüye uygulanan eşikleme işlemi, nesnenin ayırt edilebilir özelliklerini ortaya çıkarır. Bu yöntemle görüntü iki renkli bir formata dönüştürülür. Bu, önceden belirlenen bir eşik değeri kullanılarak pikselleri gri seviye değerlerine göre ayırt etmek için kullanılır.

Grnt siyah-beyaz hale gelir nk eřik deđerinin zerindeki piksellere 1 deđer, eřik deđerinin altındaki piksellere ise 0 deđer verilir. Bu prosedr, arka plandaki hataları ayrıca grebilir. Herhangi bir grnt piksel deđerleri, $f(i,j) > T$ durumunda nesneye ait kabul edilir; $f(i,j) \leq T$ durumunda ise arka plana ait kabul edilir. Bu fonksiyon, grnt ieriđi belirlenen eřik deđerine gre Denklem 2' deki fonksiyonla ifade edilir. Bu eřikleme yntemi, grntnn nesne ve arka plan olarak ayrılmasını sađlayan etkili bir ayırım mekanizmasıdır.

$$g(x,y) = \begin{cases} 1 & \text{eđer } f(x,y) > T \text{ ise} \\ 0 & \text{eđer } f(x,y) \leq T \text{ ise} \end{cases} \quad (2)$$



řekil 9: Eřiklemenin grafiksel ifadesi



řekil 10: Eřikleme iřlemi uygulanmıř elektronik kart grnts

2.3. zellik ıkarımı İřlemi

zellik ıkarımı, bir grntnn tanınmasında ve dijital hale getirilmesinde nem tařımaktadır. Grnt zelliklerinin elde edilmesi iin grntnn sınıfları arasında ayrımı gerekleřtirmek gerekmektedir. Bunun iin temel iřlev grntnn en belli olan, ayırt edici zelliklerinin ıkarımının yapılarak seilen farklı zelliklerin yapay sinir ađının eđitme iřleminde kullanılması gerekmektedir. Grntnn nemli ayırt edici

özellikleri olan moment ve piksel değerleri Görüntü işleme teknikleri uygulanan görüntü üzerinden elde edilmektedir. Gerçekleştirilmesi amaçlanan sistem için dönme, taşıma ve ölçeklendirme gibi fiziksel etkilerden ve dış etkenlerin bozucu etkisinden etkilenmeden, nesnelerin tanınmasını sağlayacak nitelikte olmalıdır. İki boyutlu örüntü tanıma ve uygulamalarında momentler ve momentlerin polinom fonksiyon değerleri oldukça önemlidir. Momentleri, nesnelerin şekillerinin belirlenmesinde, doku (texture) analizinde, hata belirlemede ve sınıflandırma için özellik çıkarmada kullanılmaktadır. Bu amaçla, gerçekleştirilen çalışmada da, 1960'ların başında geliştirilen Hu' nun sabit olan yedi değişmez matematiksel moment teorisinden geliştirdiği moment modelleri hata tespitinde yararlanılmıştır (Yu, 1990).

Hu, normalize merkez momentlerini kullanarak yedi adet değişmez moment geliştirmiştir (Yu, 1990). Hu'nun geliştirdiği momentler şöyledir;

$$\phi_1 = \eta_{20} + \eta_{02} \quad (3)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11} \quad (4)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (5)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2 \quad (6)$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (7)$$

$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (8)$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{03} + \eta_{12})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (9)$$



Şekil 11: Moment değerleri hesaplanan Türkçe karakterlerin görüntüsü

Çizelge 1.'de, Şekil 11.'de görülen her bir karakter için hesaplanan moment değerleri ayrı ayrı verilmiştir. Ön işlemlerden geçirilen görüntüler, Yapay Sinir Ağı (YSA) sınıflandırıcısının girişlerine gönderilmeden önce kendilerini karakterize edecek momentler hesaplanmıştır. Burada YSA girişine uygulanan moment sonuçları oldukça küçük olduğundan logaritmaları alınmıştır.

Çizelge 1: Moment değerlerinin sonuçları

Giriş 1	Giriş 2	Giriş 3	Giriş 4	Giriş 5	Giriş 6	Giriş 7
1.1275	3.5427	8.0562	10.3677	19.6363	12.2448	21.9169
1.1275	3.5427	8.0562	10.3677	19.6363	12.2448	21.9165
1.3729	5.5033	7.9705	12.1408	23.0868	15.8025	22.8975

Çizelge 2: Piksel değerlerinin sonuçları

Giriş 1	Giriş 2	Giriş 3	Giriş 4	Giriş 5
86	88	85	95	82
84	90	91	86	81
86	84	89	87	80

İki farklı özellik çıkarımı yapılarak hata tespit sistemi geliştirilmiştir. Çizelge 1 bir görüntüden elde edilen 7 adet ayırt edici moment değerlerini içermektedir. Çizelge 2 ise bir görüntüden elde edilen 528 adet piksel değerlerini içermektedir. 324 adet görüntünün moment ve piksel değerleri alınarak Çizelge 1 ve 2 tabloları oluşturulmuştur. Bu tablolardaki değerler YSA eğitime işleminde kullanılarak, ÇKA fonksiyonunun ağırlık ve bias değerleri elde edilmektedir. Bu değerler sistemin gerçek zamanlı karar sürecinde kullanılmaktadır.

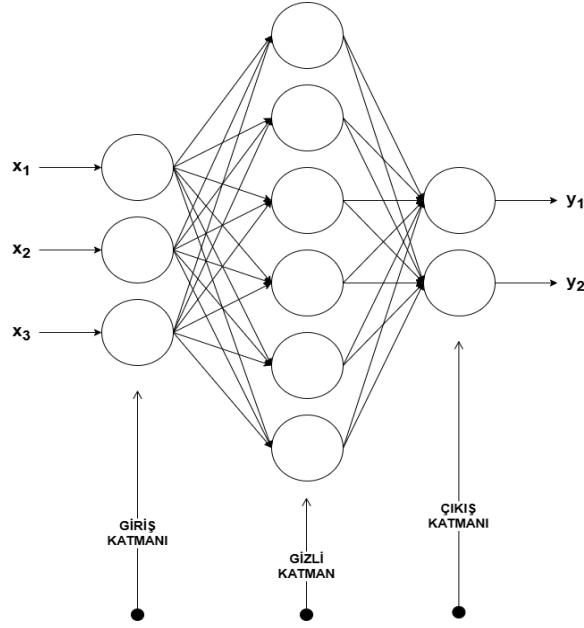
2.4. Yapay Sinir Ağları

Yapay sinir ağlarının çalışma yöntemi, işlevsel olarak insan beyin hücrelerinin çalışması yapısına benzer yapıdadır. Yapay öğrenme sistemi, örnek verileri ve olayları kullanarak öğrenebilen, bu bilgileri hatırlayıp yeni olaylarla ilişkilendirerek karar verebilen aynı zamanda çevreden gelen girişlere de tepki oluşturabilen sistemdir (Öztemel, 2020). Çalışma yapısına göre YSA sınıflandırma, ilişkilendirme, genelleme, öğrenme, özellik belirleme, en uygun şekle sokma (optimizasyon) sayesinde ilişkilendirme gibi işlevlerde başarılı şekilde uygulanmaktadır (Öztemel, 2020; AI Deport, 2024). Ayrıca YSA birçok problemin çözümünü geliştirebilir yapıdadır. YSA da ilk işlem olarak ağ eğitilmesi gerekmektedir.

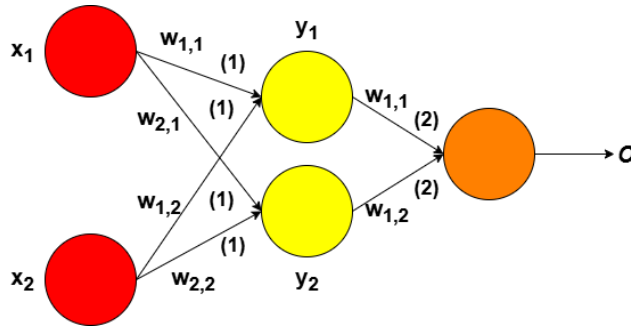
YSA eğitime işleminde örnek verilerin tanıtılması ağıın girdi ve çıktıları arasında bağıntı kurması ile sağlanmaktadır. Ağıın eğitime işlemi gerçekleştirildiğinde doğruluğun elde edilmesi, kullanılan veriler haricinde tutulan veriler ile yapılan test işlemi sonucunda gerçekleştirilir. Test sonucunda hedeflenen çıktılar üretiliyorsa YSA'nın performansının yeterli olduğu ifade edilebilir. YSA teknolojinin birçok farklı alanında ve uygulamalarda kullanılmaktadır (Öztemel, 2020; AI Deport, 2024). Geleceğin teknolojisinin altyapı niteliği taşımaktadır. Çalışmamızda eğitime ve karar işleminde çok katmanlı algılayıcı ağ yapısı kullanılmıştır.

İleri beslemeli YSA modeli, ağıın eğitilme aşamasında girdi değerlerine göre çıktı değerleri üreterek, girdi çıktı arası bağıntıyı kuran Çok Katmanlı Algılayıcı (ÇKA) modelidir (Haykin, 1994). Ağ modeli üç katmanlı bir yapıdan oluşmaktadır. Bu katmanlar girdi katmanı (giriş değerlerinin işlendiği), ara/gizli ve çıktı

katmanıdır. Burada gizli katman miktarını ve katmanlarda yer alan hücre sayısını geliştirici belirlemektedir. Bu sayılar deneme yanılma yoluyla ya da başarıyla sonuçlanmış ideal projeler referans alınarak belirlenir. Ara katman girdi katmanından alınan verilerin işlendiđi bölümdür. Çıktı katmanında ise gizli katmandan alınan verilerin durumuna göre çıktı deđerleri üretilmektedir. Yapıda bulunan tüm katmanlar ve katmanlara ait hücreler birbirleriyle bağlantılıdır. Üç giriş ve iki çıkışa sahip çok katmanlı olarak ifade edilen ileri beslemeli ağ yapısının modeli Şekil 12’ de görölmektedir.



Şekil 12: 3-girişli 2-çıkışlı çok katmanlı ileri beslemeli ağ yapısı modeli



Şekil 13: ÇKA çalışma fonksiyonu

ÇKA çalışma fonksiyonu ise Şekil 13’de görölmekte olup, $F(x)$ aktivasyon fonksiyonuna karşılık gelmektedir. Q deđerleri olarak ifade edilenler ise sistemin eşik deđerleridir. Gizli katmanın NET deđerleri, giriş ile ağırlık deđerlerinin çarpılıp toplanmasıyla elde edilir. Gizli katmanın net çıktı deđerleri bulunurken,

NET gizli katman değerine eşik değerlerinin eklenmesi ve aktivasyon fonksiyonunun da uygulanması sonucunda elde edilir (Tsoukalas, 1997; Öztemel, 2020).

Katman 1: y_1 ve y_2 iki nörondur. Her nöron, x_1 ve x_2 girişlerini kendi ağırlıkları ve bias terimleri ile birleştirerek hesaplanmaktadır.

Denklemler:

$$\bullet y_1 = f(w_{(1,1)}^{(1)}x_1 + w_{(1,2)}^{(1)}x_2 + \theta_1^{(1)}) \quad (10)$$

$$\bullet y_2 = f(w_{(2,1)}^{(1)}x_1 + w_{(2,2)}^{(1)}x_2 + \theta_2^{(1)}) \quad (11)$$

Burada:

- $w_{(i,j)}^{(1)}$: Birinci katmandaki ağırlıkları temsil eder.
- $\theta_{i_j}^{(1)}$: Birinci katmandaki bias (sapma) terimlerini temsil eder.
- f : Aktivasyon fonksiyonunu temsil eder.

Katman 2: Bu katmanda yalnızca bir çıktı vardır. Bu çıktı, y_1 ve y_2 nöronlarının birinci katmandaki ağırlıklar ve bias ile birleştirilmesinden oluşur.

Denklem:

$$\bullet o = f(w_{(1,1)}^{(2)}y_1 + w_{(1,2)}^{(2)}y_2 + \theta_1^{(2)}) \quad (12)$$

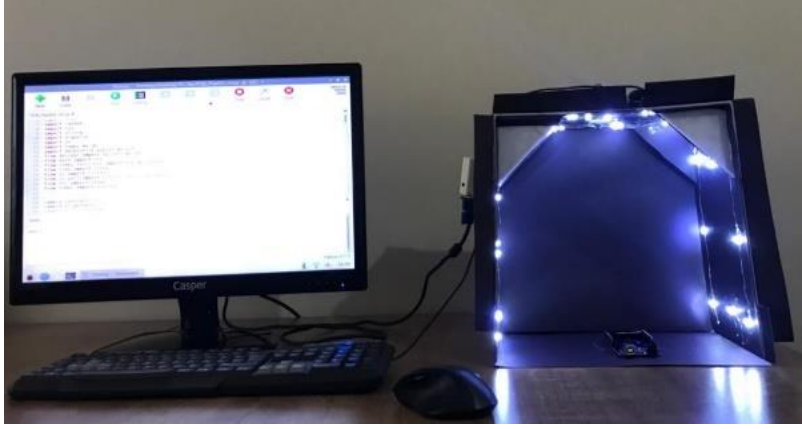
Burada:

- $w_{(i,j)}^{(2)}$: İkinci katmandaki ağırlıkları temsil eder.
- $\theta_1^{(2)}$: İkinci katmandaki bias terimi.

Bu denklemler 10., 11., ve 12., yapay sinir ağının her katmanındaki hesaplamalarının nasıl yapıldığını açıklar. Aktivasyon fonksiyonu f , ağın lineer olmayan öğrenme yeteneğini sağlar. Bu sinir ağı yapısı, temel iki katmanlı bir model olup, daha karmaşık yapılara göre oldukça basittir ancak temel prensipleri anlamak için etkilidir.

3. DENEYSEL ÇALIŞMA

Hata tespit sisteminin deneysel tasarımı Şekil 14'de görüldüğü gibi gerçekleştirilmiştir. Uygulaması gerçekleştirilen deneysel sistemde Raspberry Pi 3 kamera modülü komponent kartın yerleştirildiği stant üzerinden 30 cm aralık olacak şekilde yukarıya konumlandırılmıştır. Uygulamalara başlanılmadan önce ışığın seviyesi ve çevresel etkileri azaltacak bir ortam oluşturulmalıdır.



Şekil 14: Hata tespit sistem tasarımı

Görüntüleri önceden yapılan çalışma ile elde edilen, komponent baskı devre kart görüntülerinin işlenmiş sonuçları gösterilerek sonrasında YSA yöntemi ile hata tespit işlemi gerçekleştirilmiştir. Kamera vasıtasıyla görüntülenen ham hatalı/hatasız görüntüler Şekil 15 ve Şekil 16' de verildiği gibi 8 bit 256 gri seviyeye dönüştürülmüştür. Alınan görüntü ilk olarak gri seviyeye indirgenmiştir. Kontrastı sınırlandırılarak gri seviyeye çevrilen görüntü, sonrasında 5x3 pencere genişliğinde medyan filtreden geçirilmiştir.

Medyan filtre yapılarak görüntü üzerindeki tespit edilen gürültüler kısmen ortadan kaldırılmıştır. Sonra aşamada ise eşikleme işlemi yapılarak görüntüde bulunan ayırt edici farklı bölgeler arka plandan ayrıştırılmıştır. Görüntü 8 bit gri seviyeden, 2 bite (0 veya 255) indirgenmiştir duruma getirilmiştir.



Şekil 15: Hatasız filtrelenmiş görüntüler

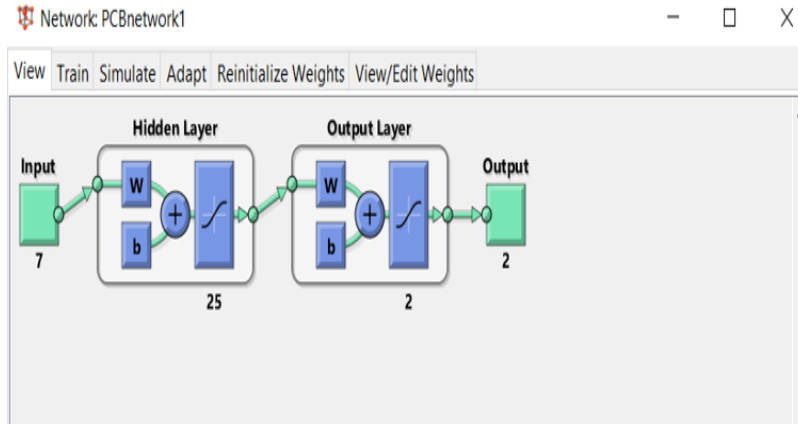


Şekil 16: Hatalı filtrelenmiş görüntüler

Yapay sinir ağı eğitme işlemi için matlab programında nntool arayüzünde ÇKA kullanılarak yapılmıştır.. ÇKA modeli tasarımı öncesinde Raspberry Pi 3 mikrobilgisayarında Python programı ile 2 farklı özellik çıkarımı yapılmıştır. Printed Circuit Board (PCB) ayırt edici özellikleri olarak 7 adet hu vektör bileşeni ve 528 adet piksel değerleri elde edilmiştir. Ayırt edici özellikler (girdiler) ve bu değerleri oluşturan çıktı değerleri, eğitme işlemi sürecinde de test amaçlı olarak kullanılan 24 adet test girdi verilerinin excel dosyaları oluşturularak matlab programında eğitme işleminde kullanılmıştır.

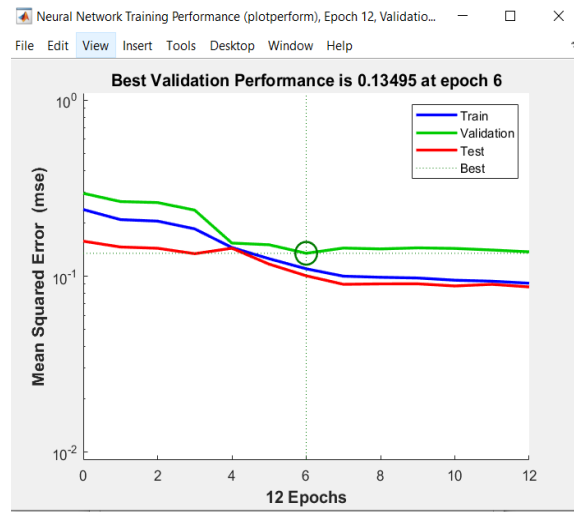
Bu girdi değerleri aynı ÇKA model mantığıyla farklı eğitme işlemleri için de gerçekleştirilmiştir. Eğitme için kullanılan algoritmada geri yayılım algoritması (trainlm), aktivasyon fonksiyonu olarak da tansig ve logsig fonksiyonları kullanılmıştır. Farklı fonksiyonların kullanım amacı hangi konfigürasyonda sistemin minimum hata maksimum performans ile çalışacağını karşılaştırılmalıdır. Geliştirilen ağda Performans ölçütü olarak hata karelerinin ortalaması olan MSE kullanılmıştır.

Grafik çıktıları ve ÇKA modeli hu vektörleri üzerinden anlatılmıştır. Gizli katmandaki nöron adedi 25 olarak belirlenmiş olup girdi ve çıktıların toplamının iki katına yaklaşık bir değer elde edilecek katsayıları en iyi şekilde sonuçlandırmaktır.



Şekil 17: Matlab ÇKA modeli

Girdi katmanında ise 7 adet baskı devre kartının (Printed Circuit Board-PCB) ayırt edici farklı özellikleri olan hu değerleri atanmıştır. Hatasız (1), hatalı (0) olarak 2 farklı duruma ait çıktı katmanı 2 olarak belirlendiği durum Şekil 17’ da görüldüğü üzere ÇKA modeli tasarlanmıştır.



Şekil 18: Eğitim işleminin çalışma performansı

Şekil 18’deki grafikte eğitim işleminin sonucunda elde edilen çalışma performansı grafiksel değer olarak gösterilmiştir. Eğitim işleminin performansı yaklaşık MSE 0.13495 olarak bulunmuştur.

	1	2	3	4	5	6	7	8	9	10
1	0.9080	1	1	0.9871	0.9979	0.9984	1	0.9731	0.9149	0.9783
2	0.0179	0	0	3.1833e-04	1.4142e-05	1.0209e-05	0	0.0300	0.0907	0.1234
3										
4										
5										
6										
7										

Şekil 19: Eğitim işleminin sonuçları

Hu moment değerleri referans alınarak kullanılan test yapısına göre yapay sinir ağı eğitim işlemi sonrasında doğru çalışıp çalışmadığının test edilmesi için eğitim işleminde kullanılan hatalı ve hatasız kart verileri dışında 12 adet hatasız, 12 adet hatalı kart verisi test edilmiştir. Test işlemini sonucunda hatasız kart verilerin karşılığı 1 değerinde hatalı kart verilerinin karşılığı 0 değerinde olmalıdır. Test işleminde kullanılan hatasız 12 adet test verisinde; 9 adet veri karşılığı 1'e yakın değerde, 3 adet veri karşılığı ise 0.7 olarak gözlemlenmiştir. 3 adet hatasız kartın karşılığı 1 değerine yakın olmadığı için başarısız olmuştur. Hatalı kart test verilerinde ise 6 adet verinin karşılığının 0'a yakın değerde doğru çıktı ürettiği, 6 adedin ise 0'a yakınsamadığı için hatalı çıktı ürettiği gözlemlenmiştir. Sonuçlar bölümünde çıktılar detaylandırılmıştır (Tekin, 2021).

4. SONUÇ

Gerçekleştirilen çalışmada hata tespit sistemi tasarımı gömülü yapıda mikrobilgisayar Raspberry Pi 3 kullanılmıştır. Kameradan elde edilen görüntüler ilk olarak gri seviyeye çevrilerek daha sonra da gürültü azaltama teknikleri uygulanarak medyan filtre ile gürültülerden uzaklaştırılmış ve eşikleme yöntemiyle de karanlık – aydınlık bölgeler birbirinden ayrıştırılmıştır. Moment değerleri ise görüntünün bu ayrıştırılmış/arındırılmış son durum hali kullanılarak nesnelere için geometrik ağırlık merkezi noktası ve tamamlayıcı vektör seti kullanılarak bulunmuştur.

Çizelge 3: Eğitim ve test işlemi veri sayısı bilgileri.

Özellik Çıkarımı	Hatasız Kart Eğitim Veri Sayısı	Hatalı Kart Eğitim Veri Sayısı	Hatasız Kart Test Veri Sayısı	Hatalı Kart Test Veri Sayısı
Hu Vektör Değerleri	150	150	12	12
Piksel Değerleri	150	150	12	12

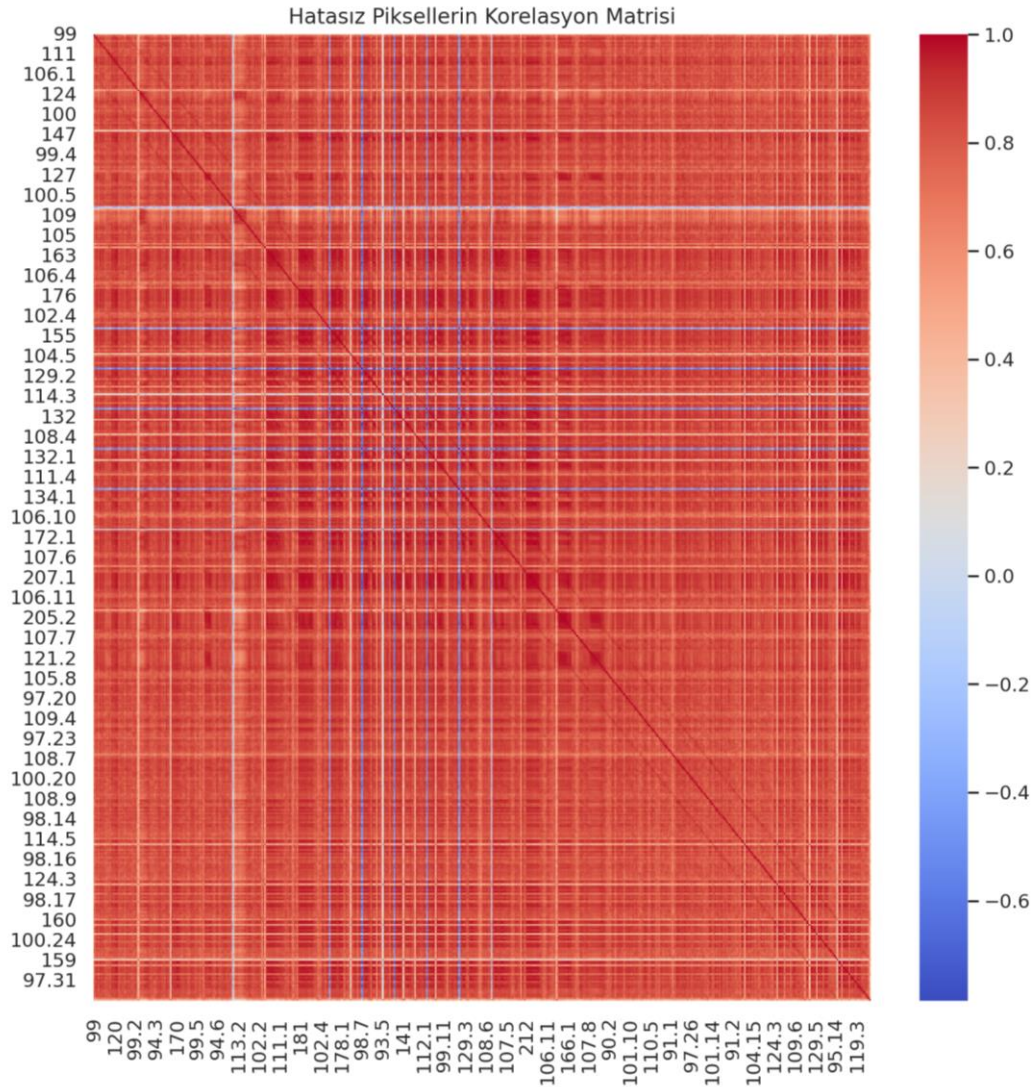
Çizelge 4: Matlab eğitime ve test işlemi sonuçlarının karşılaştırılması.

Özellik Çıkarımı-ÇKA Yapısı	Hatasız Kart (1) Test Sonucu	Hatalı Kart (0) Test Sonucu
Hu Vektör-Tansig Fonksiyon	3 Adet 0.7 /	6 adet 0, 6 /
2 Gizli Katman	9 adet 0, 99	6 adet 0, 1
Piksel-Tansig Fonksiyon	12 adet 1	12 adet yaklaşık
2 Gizli Katman		0, 00000998
Piksel-Logsig Fonksiyon	12 adet	12 adet yaklaşık
3 Gizli Katman	0, 999	0, 000000095

Hu vektör özellikleri kullanılarak YSA eğitime işlemi Çizelge 3' de verilmiştir. Burada 300 adet veri girdileri tanıtılmış olup, bunların dışında 24 adet de test veri girdileri ile doğruluk testi yapılmıştır. Yapılan test işlemleri sonucunda; 24 veri girdisinden 9 tanesinin hedeflenen 0 ve 1 değerine yakınsamadığı ölçülmüş, başarı oranı %62,5 olarak elde edilmiştir. Hu vektörlerinin özellik çıkarımında istenilen başarıyı yakalayamamasının nedeni olarak farklı çıktıların aynı veri olarak algılanması yani hata vektörlerinin benzer cisimler için ayırt etme özelliğinin yetersiz kalması durumu söylenebilir.

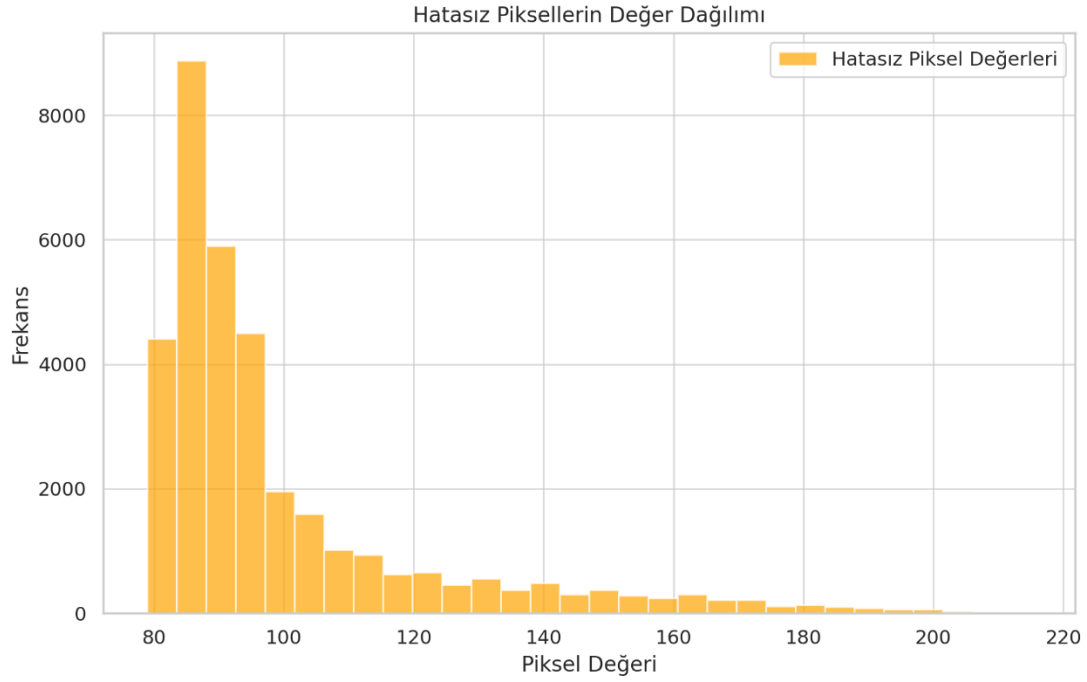
Aynı şekilde 300 adet veri girdisi ve bunların dışındaki 24 adet test girdisi kullanılarak doğruluk testi için görüntülerin piksel özelliklerine eğitime işlemi yapılmıştır. Bu işlemde transfer fonksiyonu ve katman sayı değişimleri yapılarak farklı eğitime işlemlerinin gerçekleştirildiği durumun sonuçları Çizelge 4' De verilmiştir. Eğitime işlemi sonuçlarına göre; hu ışığın bozucu etkisi ve nesnelerin şekillerinin benzer olması gibi durumlar göz önüne alındığında hu vektörlerin Off-line ve On-line çalışmada performansı düşük bulunmuştur. Performans durumu en yüksek başarıyı Off-line ve On-line hata tespit işleminde piksel özelliklerinin girdi olarak kullanıldığı, transfer fonksiyonu olarak logsig ve katman sayısı 3 seçildiğinde elde edilmektedir.

PCB hata tespit sisteminin YSA modelinde logsig transfer fonksiyonu referans olarak alındığında; Off-line % 98 performans, On-line kontrolde ise 75 adet kontrolde tamamını doğru tahmin yaparak % 100 başarılı performans sağlamıştır. Yapılan eğitime işlemlerinde aydınlatma seviyesinin doğru olarak belirlenmesi, sistemin doğru çalışmasında önemli bir faktör olduğu görülmüştür. Görüntülerin kamera modülünden başarılı, net bir şekilde alınması sonraki adımlardaki işlemlerin performansını olumlu yönde etkilemektedir. YSA eğitime işleminde piksel değerlerine ve vektör setine karşılık gelen çıktıların binary olarak kodlanması eğitime işleminin sürecini oldukça hızlandırdığı da ifade edilmelidir. Ayrıca benzer olacağı düşünülen nesnelerin hata tespitinde eğitime veri seti sayısının artırılması gerekmektedir. Bu bağlamda çalışmadaki piksel değerleri gerçek zamanlı sistemler için başarılı bulunmasına rağmen aydınlatma ve gürültüye karşı hassasiyet gösterdikleri için çalışmanın başarısını da direkt olarak etkiledikleri görülmüştür.



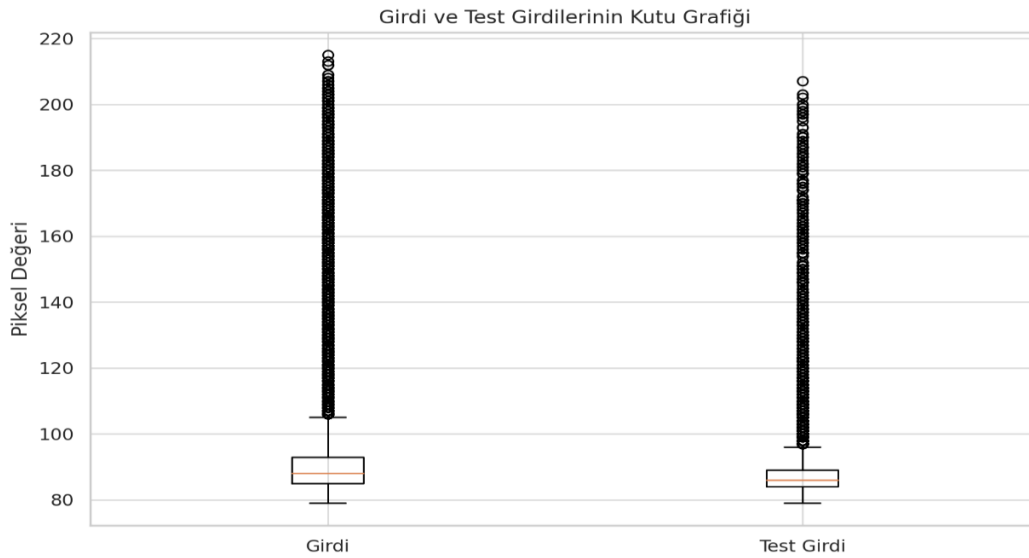
Şekil 20: Hatasız Piksellerin Korelasyon Matrisi

Şekil 20 Hatasız Piksellerin Korelasyon Matrisi; bu grafik, hatasız pikseller arasındaki korelasyonları görselleştirmektedir. Grafikte her bir piksel çifti arasındaki korelasyon katsayıları renk yoğunluğuna göre değişmektedir. Kırmızıya yakın renkler yüksek pozitif korelasyonu, maviye yakın renkler ise negatif korelasyonu gösterir. Bu grafik, hatasız piksellerin birbirleriyle olan ilişkilerini analiz etmek için kullanılır. Korelasyon katsayısının yüksek olduğu alanlar, piksellerin birbirine daha bağımlı olduğunu gösterir.



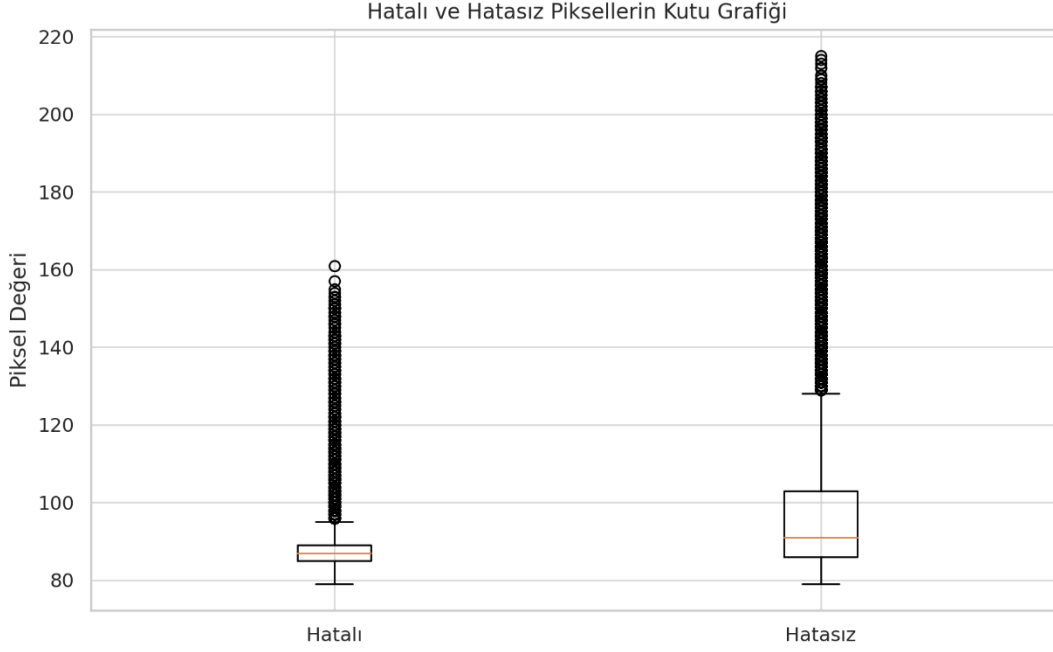
Şekil 21: Hatasız Piksellerin Değer Dağılımı

Şekil 21 Hatasız Piksellerin Değer Dağılımı; bu histogram, hatasız piksellerin değer dağılımını göstermektedir. X eksenini piksel değerlerini, Y eksenini ise bu değerlere sahip piksel sayısını temsil eder. Bu grafik, hatasız piksellerin yoğunlaştığı değer aralıklarını ortaya koyar. Piksel değerlerinin çoğunlukla 80-100 aralığında toplandığı gözlemlenebilir, bu da hatasız piksellerin büyük ölçüde düşük değerlere sahip olduğunu gösterir.



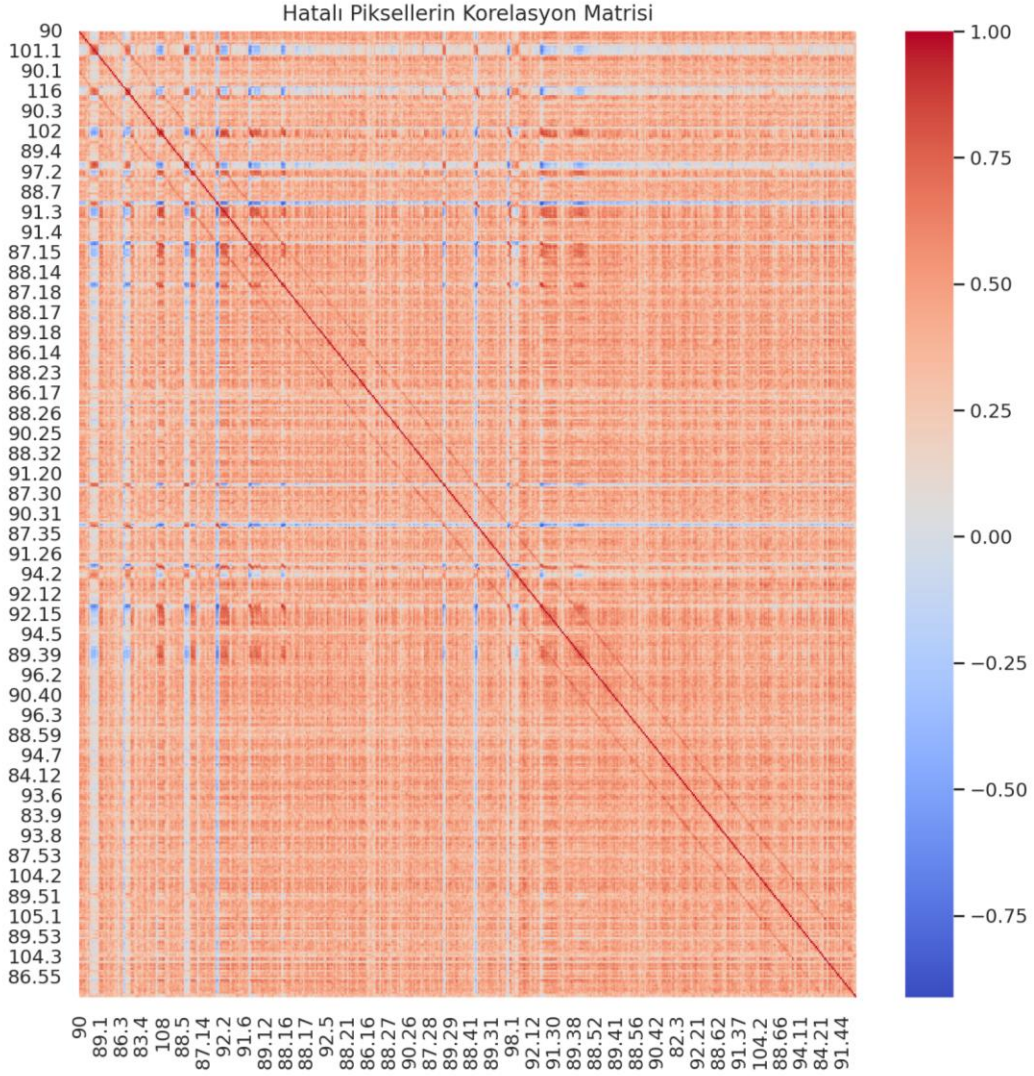
Şekil 22: Girdi ve Test Girdilerinin Kutu Grafiği

Şekil 22 Girdi ve Test Girdilerinin Kutu Grafiği; bu kutu grafiği, orijinal girdi verileri ile test girdilerinin piksel değeri dağılımlarını karşılaştırır. Girdi ve Test Girdi olarak iki farklı grup halinde görselleştirilen bu grafik, her iki grup için minimum, birinci çeyrek, medyan, üçüncü çeyrek ve maksimum değerleri içerir. Girdi ve test verileri arasında benzer bir dağılım olması, veri tutarlılığını gösterir.



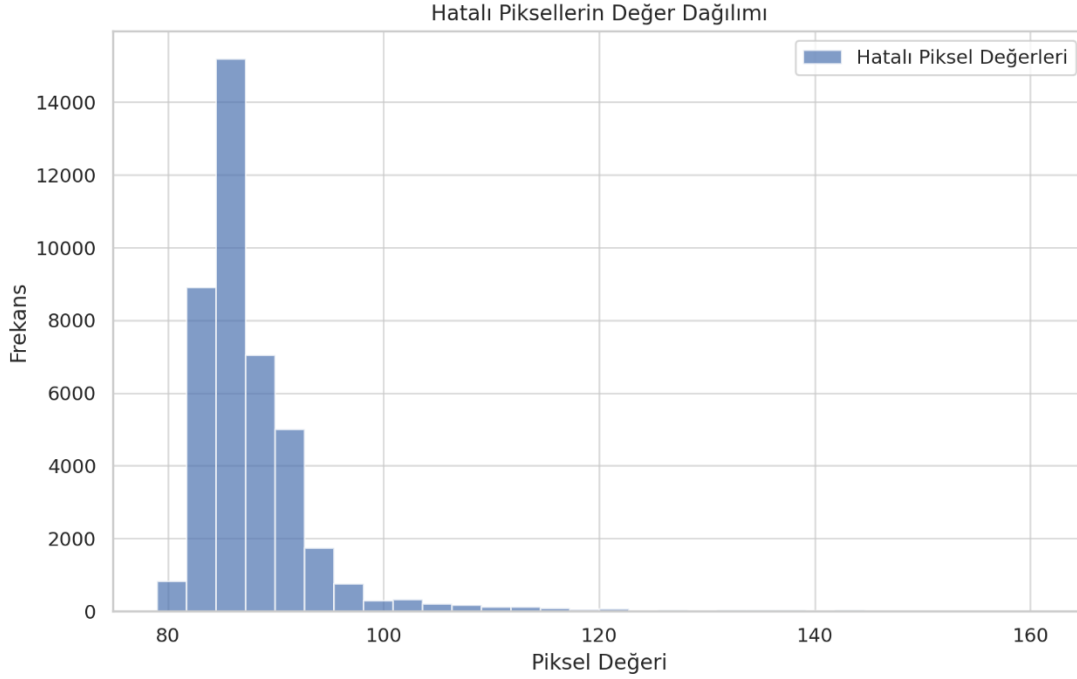
Şekil 23: Hatalı ve Hatasız Piksellerin Kutu Grafiği

Şekil 23 Hatalı ve Hatasız Piksellerin Kutu Grafiği; bu kutu grafiği, hatalı ve hatasız piksellerin dağılımlarını karşılaştırır. Hatalı ve Hatasız olarak iki ayrı grup halinde görselleştirilen bu grafik, her iki grup için de minimum, birinci çeyrek, medyan, üçüncü çeyrek ve maksimum değerleri içerir. Hatalı piksellerin değerleri daha düşük aralıklarda yoğunlaşırken, hatasız piksellerde daha geniş bir dağılım görülmektedir.



Şekil 24: Hatalı Piksellerin Korelasyon Matrisi

Şekil 24 Hatalı Piksellerin Korelasyon Matrisi; bu grafik, hatalı pikseller arasındaki korelasyonları görselleştirmektedir. Her bir piksel çifti arasındaki korelasyon katsayıları renk yoğunluğuna göre değişmektedir. Kırmızıya yakın renkler yüksek pozitif korelasyonu, maviye yakın renkler ise negatif korelasyonu gösterir. Hatalı pikseller arasındaki korelasyonun incelenmesi, hataların dağılımında belirli bir desen olup olmadığını anlamak için faydalıdır.



Şekil 25: Hatalı Piksellerin Değer Dağılımı

Şekil 25 Hatalı Piksellerin Değer Dağılımı; bu histogram, hatalı piksellerin değer dağılımını göstermektedir. X eksenini piksel değerlerini, Y eksenini ise bu değerlere sahip piksel sayısını gösterir. Hatalı piksellerin değerlerinin çoğunlukla 80-90 aralığında toplandığı gözlenmektedir. Bu, hatalı piksellerin daha düşük değerlere sahip olduğunu ve belirli bir aralıkta yoğunlaştığını gösterir.

Benzer geliştirilen uygulamalarda taşınabilirlik, maliyet yüksekliği, elde edilen görüntülerdeki çözünürlük değerlerinin düşük olması gibi sistemin optimum çalışmasına engel olacak eksiklikleri arasında sayılabilir durumlardır. Bu eksiklikleri en aza indirmek amacıyla bu sistem kontrol işlemlerinin daha hızlı ve hatasız gerçekleştirilebilmesi amacıyla tasarlanarak uygulanmıştır.

Aynı zamanda gömülü sistemde kullanılan yazılımların açık kaynak kodlu kütüphane ve görüntü işleme algoritmaları modelleri olduğu düşünülürse sistemin devamlı olarak güncellenebilir olması da çalışmanın sonuçları arasına eklenebilir. Son olarak bu sistemde kullanılan yazılım, ürün özellik çıkartımı ile oluşturulan veri tabanının da güncellenerek farklı nesnelerin kontrol edilebilmesi amacıyla değişik sektör/işletmelerde kullanılabilir olması çalışmanın güncelliğine ve geleceğine katkı vermektedir.

ÇIKAR ÇATIŞMASI BEYANI

Yazarlar arasında çıkar çatışması bulunmamaktadır.

YAZARLARIN KATKILARI

B.Y.: Kavramsallaştırma, yöntem, yazılım, doğrulama, formel analiz, araştırma, kaynaklar, yazı yazma.

İ.Y.: Yöntem, yazılım, kaynaklar, yazı yazma - gözden geçirme, orijinal taslak hazırlama ve düzenleme.

KAYNAKLAR

- AI Deport.* (2024, 08 04). AI Deport: <https://ai-depot.net> adresinden alındı
- Akar, E. O. (2006). *Tarihi görüntülerin kalitesinin görüntü işleme teknikleri kullanılarak artırılması*. Konya: Selçuk Üniversitesi / Fen Bilimleri Enstitüsü.
- Ataseven, B. (2013). Yapay Sinir Ağları İle Öngörü Modellemesi. *Öneri Dergisi*, 101-115.
- Bayindir, R., Sesveren, Ö. (2008). YSA Tabanlı Sistemler İçin Görsel Bir Arayüz Tasarımı. *Pamukkale University Journal of Engineering Sciences*, 101-109.
- Çelik, A., Tekin, E. (2020). Tekstil Baskı Kalite Kontrolünün Görüntü İşleme Teknikleri ile Gerçekleştirilmesi. *Avrupa Bilim ve Teknoloji Dergisi*, 268-276.
- Çetin, Ö. (2024, 08 04). Ömer Çetin: <http://www.omercetin.com.tr> adresinden alındı
- Ever, D., Demircioğlu, E.N. (2022). Yapay Zekâ Teknolojilerinin Kalite Maliyetleri Üzerine Etkisi. *Çukurova Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 59-72.
- Gonzalez, R.C., Wood, R.E. (1992). *Digital Image Processing*. Massachusetts: Pearson.
- Gültekin, E. Çelik, H.İ., Dülger, L.C., Sünbül, H.İ., Kani, H. (2019). Image Processing Applications on Yarn Characteristics and Fault Inspection. *Journal of Textile Engineering*, 340-345.
- Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*. Pearson.
- Horozoğlu, E. (2013). *Görüntü işleme ile yüzey pürüzlülüğü ölçümü ve analizi /*. Konya: Selçuk Üniversitesi / Fen Bilimleri Enstitüsü .
- Kaya, İ., Engin, O. (2011). Kalite İyileştirme Sürecinde Yapay Zekâ Tekniklerinin Kullanımı. *Pamukkale University Journal of Engineering Sciences*, 103-114.
- Korkmaz, M. Barstuğan, M. (2020). A Deep Learning-Based Quality Control Application. *European Journal of Science and Technology*, 332 - 336.
- Öztemel, E. (2020). *Yapay Sinir Ağları*. İstanbul: Papatya Yayıncılık.
- Santaş, G., Gülesin, M. (2012). Sayısal Görüntü İşleme ve Farklı Alanlardaki Uygulamaları. *Electronic Journal of Vocational Colleges*, 85-97.
- Saravanan, C. (2010). Color Image to Grayscale Image Conversion. *2010 Second International Conference on Computer Engineering and Applications* s. 196-199. Island: IEEE.
- Seker, A., Diri, B., Balık, H.H. (2017). Derin Öğrenme Yöntemleri Ve Uygulamaları Hakkında Bir İnceleme. *Gazi Mühendislik Bilimleri Dergisi*, 47-64.
- Tekin, Ş. (2021). *Yapay Sinir Ağları İle Gömülü Sistem Tabanlı Hatalı Baskı Devre Kartı Tespiti*. Sakarya: Sakarya Uygulamalı Bilimler Üniversitesi Lisanüstü Eğitim Enstitüsü.
- Tsoukalas, L.A., Uhrig, R.E. (1997). *Fuzzy and Neural Approaches in Engineering*. John Wiley & Sons, Inc.
- Yu, F.T.S., Yajun. L. (1990). Applications of moment invariants to neural computing for pattern recognition. *Hybrid Image and Signal Processing II*, (s. 1297-1307). Orlando.