Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi
Afyon Kocatepe University – Journal of Science and Engineering
https://dergipark.org.tr/tr/pub/akufemubid

Araştırma Makalesi / Research Article
e-ISSN: 2149-3367
DOI: https://doi.org/10.35414/akufemubid.1551996
AKÜ FEMÜBİD **25** (2025) 045202 (816-826)
AKU J. Sci. Eng. **25** (2025) 045202 (816-826)

PCB Üretiminde Çok Sınıflı Kusur Tespiti için YOLO Tabanlı Derin Öğrenme Modeli

Makale Bilgisi / Article Info
Alındı/Received: 18.09.2024
Kabul/Accepted: 05.02.2025
Yayımlandı/Published: 04.08.2025

YOLO-Based Deep Learning Model for Multi-Class Defect Detection in the PCB Manufacturing

İzemnur BUDAK[1*], Sezen BAL[2], Hayriye KORKMAZ[1]

[1] *Marmara Üniversitesi, Teknoloji Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, İstanbul, Türkiye*
[2] *Marmara Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, İstanbul, Türkiye*

© 2025 The Authors | Creative Commons Attribution-Noncommercial 4.0 (CC BY-NC) International License

Öz

Bu çalışmada, PCB üretim sürecinde karşılaşılan olası kusurları otomatik olarak tespit ederek insan kaynaklı hataları en aza indirmeyi ve üretim sürecini hızlandırmayı amaçlayan; görüntü işleme tekniklerini ve derin öğrenme algoritmalarını kullanan bir kusur tespit ve sınıflandırma yöntemi önerilmiştir. Özellikle PCB üretim hattında böyle derin öğrenme tabanlı bir sistemin uygulanabilirliği değerlendirilmiş ve bu doğrultuda gerçek zamanlı bir simülasyon gerçekleştirilmiştir. Bu çalışmada farklı kusur türlerini içeren görüntülerden oluşan açık kaynaklı DeepPCB veri seti içinde yer alan 1194 PCB görüntüsü kullanılmıştır. Veri setinde yer alan görüntüler üzerinde toplam 7897 kusur etiketlenmiş ve bunlardan 6385 adedi (%80) eğitim, 1512 (%20) adedi ise test için ayrılmıştır. Eğitim ve test setleri, sınıflar arasında dengeli bir dağılım olacak şekilde rastgele bölünmüştür. Kusurlar, Make Sense yazılımı kullanılarak etiketlenmiş ve bu süreçte altı farklı kusur türü tanımlanmıştır. Etiketleme işlemi, modelin eğitim aşamasında yüksek doğruluk elde edilmesinde kritik bir adım olarak değerlendirilmiştir. YOLOv4 algoritmasıyla eğitilen model, mAP@50 değeri %64,1 olarak hesaplanmıştır. Daha güncel bir algoritma olan YOLOv8 kullanılarak eğitilen modelde ise mAP@50 değeri %93,5'e ulaşmıştır. Bu sonuçlar, YOLOv8 algoritmasının, PCB üretim hatlarında yüksek doğruluk oranı ve daha az hata ile uygulanabilir bir çözüm sunduğunu göstermektedir. Diğer taraftan bu modellerde, YOLOv4 ve v8'in varsayılan standart parametreleri kullanılmıştır. Ancak literatür taraması yapıldığında bu parametrelerin optimize edilmesinin daha yüksek performans elde edilmesinde önemli bir payı olduğu görülmüştür.

Anahtar Kelimeler: *PCB; Kusur Tespiti; Görüntü İşleme; Derin Öğrenme; Gerçek Zamanlı Uygulama; Üretim Hattı Otomasyonu*

Abstract

In this study, a defect detection and classification method using image processing techniques and deep learning algorithms is proposed, which aims to minimize human-induced errors and speed up the production process by automatically detecting possible defects encountered in the PCB production process. In particular, the applicability of such a deep learning-based system in the PCB production line was evaluated and a real-time simulation was performed in this direction. In this study, 1194 PCB images were used in the open-source DeepPCBdataset, which consists of images containing different types of defects. A total of 7897 defects were labeled on the images in the data set, of which 6385 (80%) were reserved for training and 1512 (20%) for testing. The training and test sets were randomly divided to provide a balanced distribution among the classes. Defects were labeled using Make Sense software and six different defect types were identified during this process. The labeling process was considered a critical step in achieving high accuracy during the training phase of the model. The model trained with the YOLOv4 algorithm had a mAP@50 value of 64.1%. In the model trained using YOLOv8, a more up-to-date algorithm, the mAP@50 value reached 93.5%. These results show that the YOLOv8 algorithm offers a viable solution with high accuracy and fewer errors in PCB production lines. On the other hand, the default standard parameters of YOLOv4 and v8 were used in these models. However, when the literature was reviewed, it was seen that optimizing these parameters had an important share in achieving higher performance.

Keywords*: PCB; Defect Detection; Image Processing; Deep Learning; Real-Time Application; Manufacturing Line Automation*

1. Giriş

Baskılı Devre Kartları (PCB'ler), elektronik endüstrisinde onlarca yıldır yaygın olarak kullanılmaktadır. Gelecekte de kullanımı gittikçe artacak olan elektronik cihazların performansı PCB'nin tasarımına ve üretim süreci kalitesine bağlıdır. PCB üretim süreci, tasarımın karmaşıklığına bağlı olarak çeşitli kusurlara açıktır. Bu kusurlar bakır yolların plaka üzerinde aşındırma sürecinde olabileceği gibi, plaka üretimi tamamlandıktan sonra dizgi sürecinde de gözlemlenebilmektedir. Bu sebeple PCB üretim kusurları iki grup altında incelenebilir: 1) Plaka üretimi sırasındaki Kısa Devre, Açık Devre, Aşırı Aşındırma, Yetersiz Aşındırma, Sahte Bakır Parçalar, Delik Hatlar gibi kusurlar ve 2) Dizgi montaj hattında bileşenlerin yanlış yerleştirilmesi, istenmeyen lehim köprülerinin oluşumu gibi kusurlar. Geleneksel yöntemlerde, uzman operatörler

*Sorumlu Yazar/Corresponding Author: İzemnur BUDAK e-posta/e-mail: izemnur.budak@marmara.edu.tr

çıplak gözle veya mikroskop kullanarak bu kusurları tespit ederler; ancak bu yöntemler uzun zaman almasından dolayı seri üretim hatları için uygun değildir. Ayrıca operatörün kusurları gözden kaçırma olasılığı da yüksektir. Bu kusurların tespit edilememesi kalite sorunlarına, sürecin uzunluğu ise maaliyet sorunlarına yol açabilir. Bu nedenle, PCB üretim sürecinde görüntü işleme yöntemlerinin kullanılmasıyla hızlı ve güvenilir kusur tespit uygulamaları büyük önem taşımaktadır. Yüksek çözünürlüklü kameralar, gelişmiş görüntü işleme algoritmaları ve derin öğrenme modellerini kullanan bir otomatik tespit sürecinin PCB plaka üretim hattına eklenmesi, insan kaynaklı hataları azaltırken süreci otonom hale getirerek hızlandırma potansiyeline sahiptir (Zhang ve Liu 2021, Subramanyam vd. 2022, Luo vd. 2021, Du vd. 2024).

PCB plaka üretim sürecinde, plaka üzerinde iç katman tasarımının işlenmesi sonrası, dış katman tasarımının işlenmesi sonrası ve PCB aşındırma işlemi sonrasında olmak üzere her 3 denetim noktasında bu uygulama kullanılabilir. Kusurlar üretim sürecindeki aşamalara göre farklılık gösterebilmektedir. Bu kusurların tespiti için her aşama için ayrı modeller oluşturulabilir ya da tüm aşamaları kapsayacak tek bir model de kullanılabilir. Bu çalışmada oluşturulan model, kullanılan veri setindeki görüntüler tüm aşamaları kapsadığından 3 denetim noktasında da uygulanabilir. PCB plakalarının üretimi sırasında ortaya çıkan kusurların otomatik olarak tespit edilmesi ve sınıflandırılmasına yönelik çalışmada dört aşamalı bir süreç izlenmiştir:

1) Kusur özelliklerinin çıkarılması ve kategorize edilmesi
2) Veri seti hazırlanması
3) Seçilen 2 farklı versiyona sahip YOLO algoritmasıyla derin öğrenme modellerinin oluşturulması
4) Gerçek zamanlı simülasyonun gerçekleştirilmesi

Bu çalışma için literatür taraması, Web of Science veritabanı kullanılarak "PCB" ve "deep learning" anahtar kelimeleri ile yapılmıştır. 2021 ve sonrası için yapılan ilk taramada toplam 174 çalışma incelenmiştir. Daha detaylı bir tarama ise "object detection", "defect detection", "deep learning" ve "PCB" anahtar kelimeleri kullanılarak yapılmış ve 25 adet ilgili çalışma bulunmuştur. PCB yüzey kusurlarını tespit eden Sezer ve Altan (2021), Adibhatla vd. (2021), Lin vd. (2022) çalışmaları incelenmiştir. Xie ve arkadaşları bu çalışmada da kullanılan DeepPCB veri seti ve buna ilave olarak ayrıca NEU-DET veri seti üzerinde YOLOv4 ve FE-YOLO modelleri ile PCB kusur tespit çalışması yapmışlardır. Bu çalışmada DeepPCB veri seti üzerinde YOLOv4 algoritması ile mAP@50 değeri 98,5 olarak elde edilmiştir. (Xie vd. 2023).

Kim ve arkadaşları, atlamalı bağlantılı konvolüsyonel otoenkoder (autoencoder) modeline dayanan bir PCB kusur tespit modeli önermişlerdir. PCB görüntülerini kusurlu veya kusursuz şeklinde sınıflandırmışlar ve %98,08 doğruluk elde etmişlerdir (Kim vd. 2021).

Başka bir çalışmada, PCB yüzeyi tek sınıf tekli kusur tespiti için YOLOv4-MN3 modeli kullanılarak %98,64 mAP değeri ile yüksek bir doğruluk oranı sağlandığı bildirilmiştir (Liao vd. 2021).

Yue Zhang ve arkadaşları PCB üzerinde çok sınıf-çoklu kusur tespiti yapmak için Faster R-CNN, YOLOv3, YOLOv3-Tiny ve kendi oluşturdukları metot ile derin öğrenme modeli oluşturmuştur. Çalıştıkları modellerden en yüksek mAP değerini Faster R-CNN modelinden 49,50 olarak elde etmiştir (Zhang vd. 2021).

Ding ve arkadaşları, baskılı devre kartlarının kalite kontrolünü iyileştirmek amacıyla önerdikleri Tiny Defect Detection (TDD) ağı olan TDD-Net'in, PCB hata veri seti üzerinde yapılan nicel sonuçlarda tekli sınıf çok kusurlu hata tespitinde %98,90 mAP değerine ulaşmışlardır (Ding vd. 2019).

Wang ve arkadaşları, karmaşık arka planlar, küçük kusurlar ve düzensiz şekiller gibi PCB hata görüntülerinde sıkça karşılaşılan zorluklara yanıt olarak, yüzey kusurlarını tespit etme yeteneğini artırmak amacıyla geliştirdikleri YOLO-Biformer modelini tanıtmışlardır (Wang vd. 2024).

Ayrıca WOS veri tabanında “PCB” anahtar kelimesi olmadan filtreleme yapıldığında derin öğrenme tabanlı kusur tespitine yönelik farklı çalışmalar da bulunmaktadır. Zhang ve arkadaşları, deri yüzeyindeki çoklu-sınıf kusurların tespiti için YOLO tabanlı bir model kullanarak %52,3 mAP, %58,2 doğruluk ve %68,7 duyarlılık oranına ulaşmışlardır (Chen vd. 2024).

Liu ve arkadaşları, yüzey kusuru tespiti için YOLOv4 tabanlı Mixed YOLOv4-LITEv1, Mixed YOLOv4-LITEv2 ve Mixed YOLOv4-LITEv3 modellerini kullanarak sırasıyla %79,01, %81,89, %80,17 ve %78,60 mAP değerlerine ulaşmışlardır. Ayrıca bu sonuçlar, Mixed YOLOv4-LITEv3 modelinin sadece tek yüzey kusurlarını değil, aynı zamanda çoklu sınıf kusurlarını da eşzamanlı olarak tespit edebildiğini göstermektedir (Liu vd. 2023).

Li ve arkadaşları, bilgisayar anakart montajının kalite denetiminde otomasyon ve zekanın artırılmasına yönelik olarak, çok ölçekli özellikler ve dikkat mekanizmalarını entegre eden RetinaNet tabanlı geliştirilmiş bir hata tespit yöntemi önermişlerdir. Bu yöntem, deneysel doğrulamalar sonucunda, bilgisayar anakartı montaj hatalarını tespit etmek için %84,5 ortalama tespit

doğruluğu sağlamıştır (Li vd. 2023). Literatürde PCB üzerindeki veya başka nesnelerdeki kusur tespitine yönelik çalışmalar incelendiğinde bir görselde yer alabilecek sınıf sayısı ile bir sınıfa ait kusur miktarını açıklayan veri seti sınıf-kusur matrisi Çizelge 1'de oluşturulmuştur. PCB kusur tespiti uygulamaları, dört ana kategori altında toplanabilir: tek sınıf-tek kusur tespiti, tek sınıf-çok kusur tespiti, çok sınıf-tek kusur tespiti, çok sınıf-çok kusur tespiti. Tek sınıf-tek kusur tespiti (Single-Class, Sinle-Instance Defect Detection) daha basit senaryolar için kullanılır, burada algoritma mevcut görüntüde var olan yalnızca tek bir kusur sınıfına ait bir hatayı tespit etmektedir. Yani görüntüde etiketlenmiş sadece bir tane kusur (kusurlu bölge) bulunmaktadır. Tek sınıf-çok kusurlu tespitte ise (Single-Class, Multi-Instance Defect Detection), bir görüntüde aynı kusur türünden birden fazla kusur (birden fazla işaretlenmiş kusurlu bölge) bulunabilmektedir. Çok sınıf-tekli kusur tespitinde (Multi-Class, Sinle-Instance Defect Detection), model birden fazla kusur tespiti için eğitilmiş olsa bile bir görüntüde sadece tek bir kusurun olmasına karşılık gelmektedir. Çok sınıflı-çoklu kusur tespiti (Multi-Class, Multi-Instance Defect Detection) ise model yine birden fazla kusur sınıfının tespitine göre eğitilmiştir ve bir görüntüde aynı anda birden fazla aynı veya farklı sınıfa ait kusurlar bulunabilmektedir.

Çizelge 1. Veri Setindeki Sınıf-Kusur Matrisi

Sınıf Sayısı	Görseldeki Kusur Miktarı
Tek Sınıf	Tekli Kusur
Tek Sınıf	Çoklu Kusur
Çok Sınıf	Tekli Kusur
Çok Sınıf	Çoklu Kusur

Benzer çalışmalar incelendiğinde, çalışmaların büyük bir çoğunluğunda tek sınıf-tekli kusur veya tek sınıf-çoklu kusurlar tespiti yapıldığı görülmektedir. Bu çalışmanın amacı, YOLO tabanlı bir derin öğrenme modeli kullanarak PCB üretim sürecindeki çok sınıf-çoklu kusur tespitinin gerçekleştirilmesidir. Görüntü özelliği sınıflandırma ve kusurlu bölgeleri tespit etmede çok iyi sonuçlar elde ettiğinden dolayı bu çalışmada da YOLO (You Only Look Once) tabanlı bir derin öğrenme modeli, PCB kusurlarının otomatik tespiti için tercih edilmiştir. Diğer taraftan derin öğrenmeye dayalı nesne dedektörleri iki ana kategoriye ayrılmaktadır: 1) Bir aşamalı dedektörler. 2) İki aşamalı dedektörler. YOLO serisi modeller, nesne tespiti alanında yaygın olarak kullanılan tek aşamalı dedektörlerdir ve sürekli olarak gelişen versiyonları ile bilinirler (Sharma ve Mir 2020). YOLO tabanlı derin öğrenme modelleri nesne tespiti ve sınıflandırma görevlerinde üstün performans göstermektedir ve gerçek zamanlı nesne algılama uygulamaları için çok daha hızlıdır (Bochkovskiy vd. 2020), (Redmon 2016). YOLOv4, giriş görüntüsünü bir ızgaraya bölen ve her ızgara hücresi için birden çok sınırlayıcı kutu ve sınıf olasılığı tahmin eden tek aşamalı bir nesne detektörüdür (Terven vd. 2023). Ayrıca, gerçek sınırlayıcı kutularla yüksek örtüşme puanlarına (IoU) sahip tahmini sınırlayıcı kutuları seçmek için Non-Maximum Suppression (NMS) kullanır.

Bu çalışmada, nesne tespiti performansını kıyaslamak ve en etkili sonuçları belirlemek amacıyla, YOLO serisinin farklı versiyonları arasından YOLOv4 ve YOLOv8 modelleri seçilmiştir. YOLOv8, YOLO serisinin daha yeni bir versiyonu olup, geliştirilmiş mimarisi sayesinde daha hızlı işlem kapasitesi ve düşük gecikme süreleri sunmaktadır. Bu özellik de onu gerçek zamanlı nesne tespiti için daha uygun bir seçenek haline getirmektedir. Makalenin geri kalanı aşağıdaki şekilde düzenlenmiştir: Bölüm 2'de, veri setinin hazırlanması, web tabanlı bir yazılım aracı olan Make Sense (İnt. Kay.2) kullanılarak görüntü veri setinin etiketlenmesi ve Google Colab'da derin öğrenme modelinin oluşturulması anlatılmaktadır. Bölüm 3'te deneysel sonuçlar verilmiştir. Ayrıca kusurları manuel olarak etiketlenen bir görüntü, önerilen model tarafından otomatik olarak tespit edilen aynı görüntü ile karşılaştırılmıştır. Bölüm 4'te ise sonuçlar tartışılmış, benzer çalışmalarla karşılaştırma yapılmış ve gelecek çalışmalar sunulmuştur.

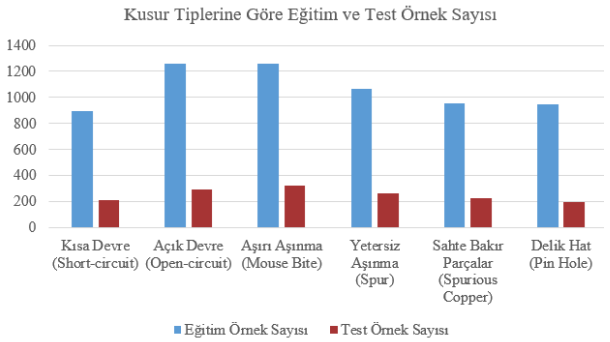
2. Materyal ve Metot

Bu çalışmada farklı kusur türlerini içeren görüntülerden oluşan açık kaynaklı DeepPCB veri seti içinde yer alan, 1194 PCB görüntüsü kullanılmıştır (Tang vd. 2019). Endüstriyel standartlara uygun olarak toplanmış, 16k × 16k piksel boyutundaki yüksek çözünürlüklü ham görüntüler, 640 × 640 piksellik alt görüntülere dönüştürülüp hizalanmış olarak sunulmuştur. Ayrıca bu görüntüler, aydınlatma kaynaklı bozulmaları azaltmak amacıyla ikili hale getirilmiştir. 1194 PCB görüntüsünün 959'u eğitim ve 235'i test için ayrılmıştır. Veri setinde yer alan bu 1194 görüntü üzerinde toplam 7897 adet kusur etiketlenmiştir. Bunlardan 959 eğitim görseli üzerinde etiketlenmiş 6385 kusur (toplam kusur miktarının yaklaşık %80'i) eğitim için, 235 test görseli üzerinde etiketlenmiş 1512 kusur (toplam kusur miktarının yaklaşık %20'si) ise test için kullanılmıştır. Bir görselde birden fazla kusur etiketlenmiş olduğundan, görsel sayısı ile kusur sayısı farklıdır. Bu veri setinin bir başka özelliği de bir görüntünün birden fazla sınıfta kusur içermesi ve aynı sınıf kusurun görüntünün farklı koordinatlarında yer alabilmesidir. Veri kümesi Çizelge 2'den de görüldüğü üzere 6 farklı türde kusur sınıfı içermektedir. Eğitim ve test setleri, sınıflar arasında dengeli bir dağılım olacak şekilde

rastgele bölünmüştür. Şekil 1 farklı PCB kusur sınıflarına ayrılan eğitim ve test örneklerinin sayısal dağılımını göstermektedir.

Çizelge 2. PCB Kusur Tipleri ve Etiket Numaraları

Kusur Tipi	Etiket Numarası
Kısa Devre (Short-circuit)	0
Açık Devre (Open-circuit)	1
Aşırı Aşınma (Mouse Bite)	2
Yetersiz Aşınma (Spur)	3
Sahte Bakır Parçalar (Spurious Copper)	4
Delik Hatlar (Pin Hole)	5



Şekil 1. Eğitim ve Test Setlerinde Sınıfların Dağılımı

Veri seti işaretlemesi için web tabanlı ve masaüstü uygulamalar mevcuttur. Sık tercih edilen web tabanlı araçlar arasında Make Sense (“MakeSense,”), Labelbox ("labelbox,") (İnt.Kay.3), SuperAnnotate ("SuperAnnotate,") (İnt.Kay.7), Roboflow ("Roboflow,") (İnt.Kay.6), Supervisely ("Supervisely,") (İnt.Kay.8) gibi uygulamalar bulunurken; masaüstü araçlar arasında LabelImg ("LabelImg,") (İnt.Kay.4), Labelme ("Labelme,") (İnt.Kay.5) bulunmaktadır. Web tabanlı araçların avantajı kurulum gerektirmeden kullanılması, masaüstü uygulamaların avantajı ise çevrimdışı kullanım sunmasıdır (Mahrishi vd. 2021, Ocepek vd. 2021, Hassan vd. 2024, Ling ve Isa 2023). İşaretleme yapılacak veriye en uygun araç seçilmelidir. Çünkü modelin güven skoru etiketlenen görselin sınırlarına bağlıdır. Ayrıca etiketleme yapmak için gereken süre de tercih edilen araca göre değişmektedir ve bu süreç derin öğrenme modelinin geliştirilmesinde en zahmetli basamak olabilmektedir. Bu çalışmada, kullanım kolaylığı ve ücretsiz erişim olanakları nedeniyle web tabanlı bir uygulama olan Make Sense aracı tercih edilmiştir. Make Sense aracının YOLO için ürettiği çıktı, her bir resim için Şekil 2'de gösterildiği gibi bir txt dosyasıdır.

```
0 0.035714 0.595899 0.044974 0.054233
5 0.595899 0.359788 0.056878 0.055556
2 0.500000 0.398810 0.071429 0.067460
4 0.680556 0.481481 0.070106 0.071429
4 0.792989 0.418651 0.091270 0.091270
3 0.685185 0.615079 0.063492 0.050265
1 0.289683 0.248016 0.063492 0.038360
```

Şekil 2. Make Sense ile Etiketlenmiş Görüntünün Txt Dosyası

Her bir satırda yer alan 5 değer sırasıyla "etiket tipi, x1, y1, x2, y2" formatında belirtilmiştir. YOLO algoritmalarında, etiketlenmiş nesnelerin konumları ve boyutları, görselin boyutlarına göre normalize edilir. Bu normalizasyon işlemi, nesnenin orta noktasının x koordinatı (x_center), y koordinatı (y_center), genişliği (width) ve yüksekliği (height) üzerinden yapılır. x_center ve y_center değerleri, sırasıyla, görselin genişliğine ve yüksekliğine bölünerek hesaplanır. Bu normalizasyon sonucu elde edilen koordinatlar, nesnenin sınırlayıcı kutusunun köşe koordinatlarını hesaplamak için kullanılır. Nesnenin sınırları Denklem (1), (2), (3) ve (4) kullanılarak hesaplanır:

$$x1 = x_center - (width / 2) \quad (1)$$

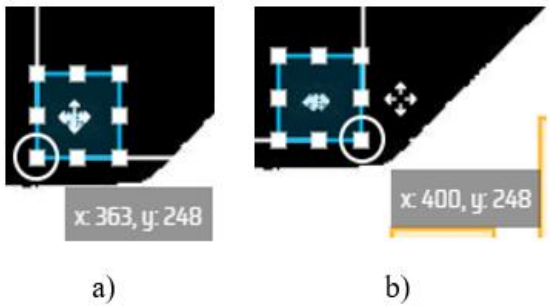
$$y1 = y_center - (height / 2) \quad (2)$$

$$x2 = x_center + (width / 2) \quad (3)$$

$$y2 = y_center + (height / 2) \quad (4)$$

Bu yöntem, YOLO algoritmasının nesne tespiti için kullandığı görüntülerdeki nesnelerin tam olarak yerini ve boyutunu doğru bir şekilde yansıtmak üzere tasarlanmıştır.

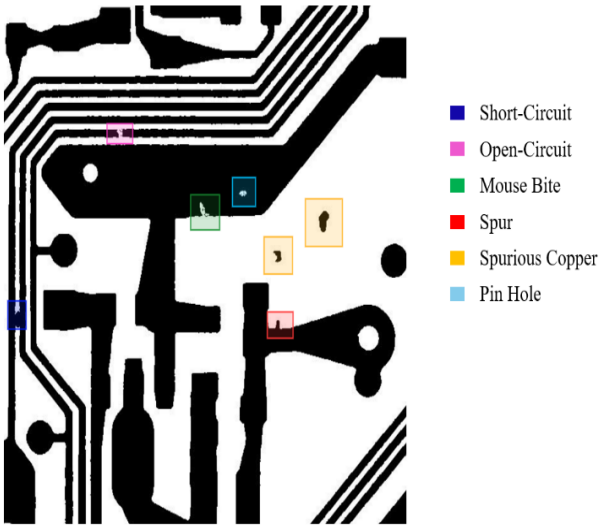
Veri setinde 640 x 640 piksel boyutlarındaki Şekil 4’teki görselde, Şekil 2’de gösterilen etiket sınıfı 5 olan (2.satırdaki) nesnenin x koordinatının başlangıç ve bitiş değerleri Şekil 3.a ve 3.b’ de gösterilmektedir. Bu nesnenin merkez x koordinatı x_center olarak hesaplanır =[[(400-363)/2]+363]/640= 0,6; and width = (400-363)/640= 0,06. Buna göre, x1 yaklaşık olarak x1= 0,6-0,03= 0,57 hesaplanır. Şekil 2, bu değerin daha kesin olarak 0,595899 olduğunu göstermektedir.



Şekil 3. Şekil 2’de 2. Satırda Verilen 5 Nolu Kusura Ait Make Sense Aracında Etiketlenen Bölgeye Ait Koordinatlar a) Nesnenin X Koordinat Başlangıcı b) Nesnenin X Koordinat Sonu

Şekil 4'te, Make Sense aracı kullanılarak etiketlenmiş bir PCB görseli sunulmaktadır. Bu görselde, çeşitli kusurların adları manuel etiketlenmiştir. Etiket isimleri Türkçe verildiğinde derin öğrenme modeli ile test

gerçekleştirilirken Türkçe karakterler tanınmadığından etiketlemede sınıfların İngilizce isimleri kullanılmıştır.



Şekil 4. DeepPCB Veri Kümesindeki Rastgele Seçilmiş Bir Görüntünün Make Sense Aracı ile Manuel Etiketlenmesi

YOLO tabanlı derin öğrenme modelleri nesne tespiti ve sınıflandırma görevlerinde üstün performans göstermektedir ve gerçek zamanlı nesne algılama uygulamaları için çok daha hızlıdır (Bochkovskiy vd. 2020). Ayrıca YOLO, hem CPU hem de GPU'larda çalışacak şekilde optimize edilebilir. Bundan dolayı bu çalışmada PCB kusurlarının otomatik tespiti için YOLO tabanlı derin öğrenme modelleri kullanılmıştır.

Bu çalışmada kullanılan deneysel ortam Google Colab, derin öğrenme aracı da YOLOv4'te Darknet, YOLOv8'de ise PyTorch'tur. Deneysel ortama ait parametreler Çizelge 3'te gösterilmiştir.

Çizelge 3. Google Colab Deneysel Çevre Parametreleri

İsim	Parametre
CUDA Versiyon	12.2.0
cuDNN Versiyon	8.9.6
GPU	Tesla T4

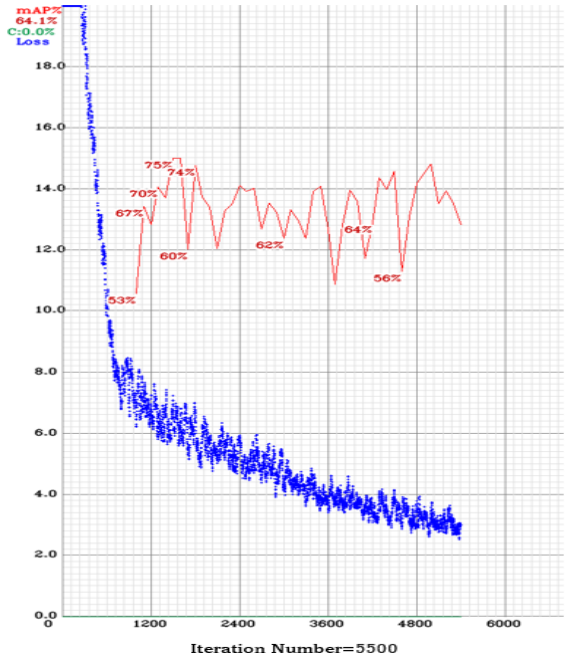
Google Colab, kullanıcıların web tarayıcısı aracılığıyla güçlü GPU'lar ve TPU'lar gibi bulut tabanlı hesaplama kaynaklarına erişim sağlayan ücretsiz bir makine öğrenmesi platformudur. Ancak Google Colab, varsayılan olarak Darknet çerçevesini içermez. Bu nedenle Darknet'in kullanımı için öncelikle kurulması gerekmektedir. Bu süreç, Darknet'in GitHub deposundan gerekli bağımlılıkların yüklenmesini ve kaynak kodunun derlenmesini içermektedir.

Veri kümesini eğitim ve test kümelerine bölünmüştür. Eğitim kümesi, modelin ağırlıklarını ve önyargılarını eğitmek için, test kümesi ise modelin performansının tarafsız bir değerlendirmesini sağlamak için kullanılır. Veri seti ve eğitim aracı dosyaları Google Drive'a yüklenmiştir. Google Colab ortamında Python kullanılarak eğitim işlemi gerçekleştirilmiştir. YOLOv4, YOLOv8 modeline göre daha eski bir model olup, daha düşük donanım gereksinimlerine sahiptir. Bazı eski GPU'larda YOLOv8 modeli ile uyumsuzluklar yaşanabilmektedir, ancak Google Colab'da böyle bir sorun yoktur.

Bu çalışmada da YOLOv4 modelinin performansının yetersiz olduğu tespit edildikten sonra, daha güncel ve etkili olan YOLOv8 ile model eğitimi gerçekleştirilmiştir.

3. Deneyler ve Bulgular

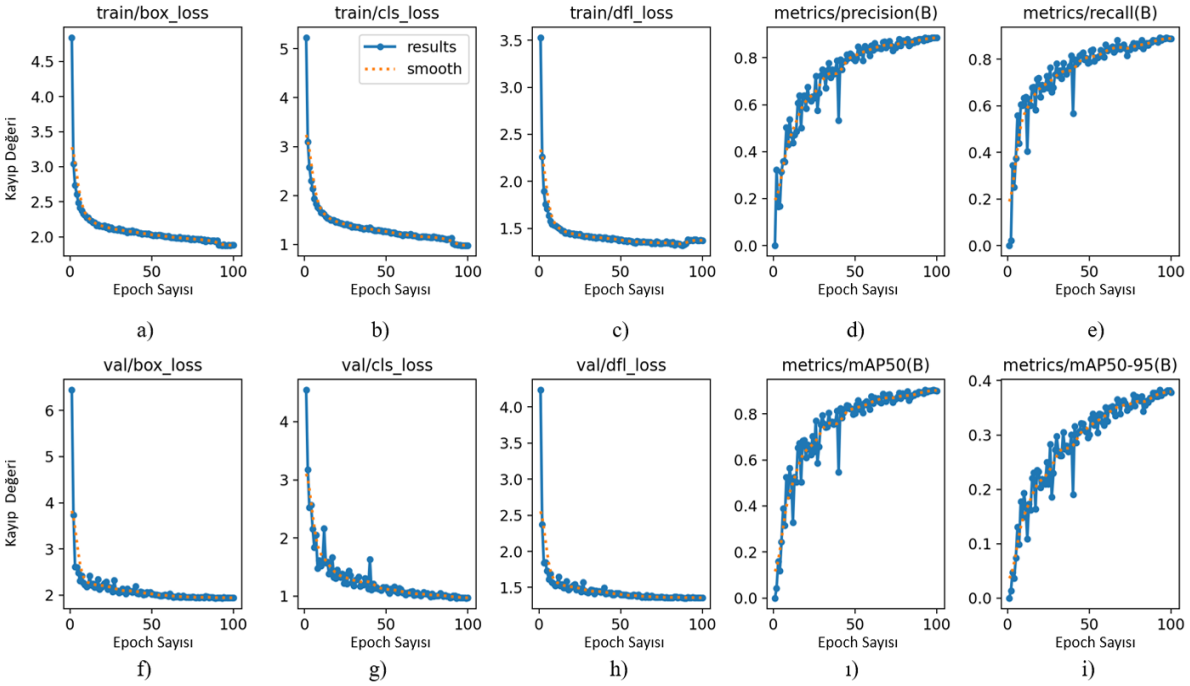
YOLOv4 eğitim sürecini özetleyen grafik Şekil 6'da gösterilmektedir. Bu grafik, modelin performansını eğitim süresince izlemek ve değerlendirmek için kullanılmaktadır. Kırmızı renkli eğri Ortalama Kesinlik (mean Average Precision, mAP) eğrisidir. Şekil 5'te modelin doğruluğunun eğitim miktarıyla artması beklenirken mAP değerinin zamanla dalgalandığı görülmektedir. Bu durumun veri artırma işlemi sonrasında aşırı öğrenmeyi engelleyeceği öngörülmektedir. Mavi renkli kayıp değişim (loss) eğrisi ise modelin ne kadar yanlış tahmin yaptığını gösterir ve bu değerin düşük olması beklenir. Değişim eğrisinin zamanla azaldığı ve 0'a yaklaştığı görülmektedir. Her iterasyon, modelin ağırlıklarının bir kez güncellendiği anlamına gelir. Örneğin, en sonuncu 5500. iterasyonda modelin kayıp değeri 25 olarak kaydedilmiştir ve mAP değeri %64.1 olarak hesaplanmıştır.



Şekil 5. YOLOv4 Ağının Eğitim Süreci

YOLOv8 eğitim sürecini özetleyen grafikler ise Şekil 6'da gösterilmektedir. Sunulan her bir eğri farklı performans ve

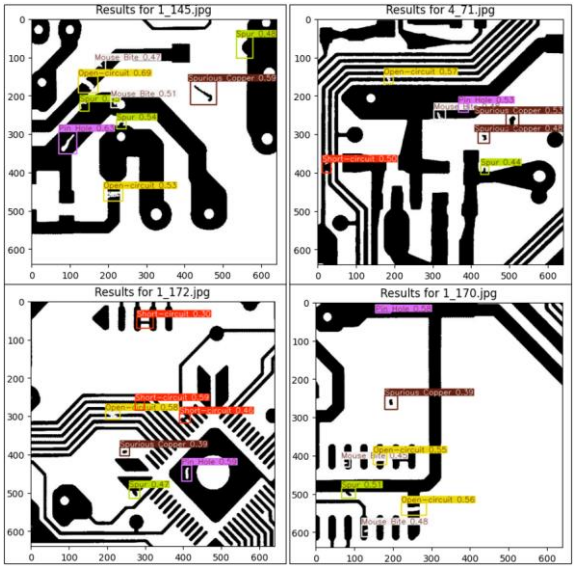
kusur metriklerini temsil eder. Train/box_loss (Şekil6.a) eğrisi, eğitim sırasındaki kutu kaybı olarak ifade edilir. Kutu kaybı nesne tespiti sırasında tahmin edilen kutuların gerçek kutularla olan farkını ölçen bir kayıp fonksiyonudur. Bu kayıp ne kadar düşükse modelin tahminleri o kadar doğru demektir. Train/cls_loss (Şekil6.b) eğrisi eğitim sırasında sınıf kaybını incelemek için kullanılırken train/dfl_loss (Şekil6.c) ve val/dfl_loss (Şekil6.h) eğrisi sınırlayıcı kutuların (bounding boxes) konumlarını daha hassas şekilde optimize etmek için incelenen bir kayıptır. Sınıf kaybı ise modelin tahmin ettiği sınıfın gerçek sınıf ile ne kadar iyi örtüştüğünü ölçen bir kayıp fonksiyonudur. Metrics/precision (Şekil6.d) eğrisi, kesinlik (precision), metrics/recall (Şekil6.e) eğrisi duyarlılık (recall) oranını ifade eder. Val/box_loss (Şekil6.f) eğrisi doğrulama (validation) verisi üzerindeki kutu kaybını, val/cls_loss (Şekil6.g) eğrisi ise doğrulama (validation) verisi üzerindeki sınıf kaybını ifade eder. metrics/map50 (Şekil6.ı) eğrisi mAP@50, ortalama kesinlik (precision) değeridir. mAP@50, modelin IoU 0,50'nin üzerinde olan tahminlerinin kesinliğini ölçer. Bu eğri modelin genel başarı performansını gösterirken, metrics/map50-95 (Şekil6.i) eğrisi mAP@50-95, IoU eşik değerlerinin 0,50'den 0,95'e kadar değiştiği bir aralıkta hesaplanan ortalama kesinlik değeridir. Bu daha genel bir performans ölçümüdür ve modelin farklı hassaslık seviyelerindeki kesinliği gösterir.



Şekil 6. YOLOv8 Ağının Eğitim Süreci

Şekil 7'de YOLOV8 modeline ait test sonuçları sunulmuştur. Test veri setinde 4_71 olarak adlandırılan görselin kusur tahmin sonuçları, manuel olarak Make Sense platformunda oluşturulan Şekil 4'teki etiketlemeyle uyum göstermektedir. Eğitim veri seti Şekil 4'teki 4_71 görselini içermemektedir. Bu görsel sadece test veri setinde yer almaktadır. Şekil 4 sadece etiketlemenin nasıl yapıldığı hakkında bilgi vermek amacıyla kullanılmıştır.

YOLOv4 modelinin sonuçları incelendiğinde, modelin eğitim süresince karşılaştığı zorluklar ve performans dalgalanmaları gözlemlenmiştir. Ancak, daha güncel bir model olan YOLOv8'in performansı, YOLOv4'e kıyasla önemli ölçüde daha iyidir. YOLOv8 modeli, %93,5 gibi daha yüksek bir mAP@50 değeri ile yüksek IoU eşiklerinde bile kusurları daha doğru bir şekilde sınıflandırma ve tespit etme yeteneğine sahiptir.



Şekil 7. Test Sonuçları

Yüzey kusuru tespitinde yaygın olarak kullanılan farklı değerlendirme ölçütleri bulunmaktadır. Her piksel, zemin gerçeği ve tahmin sonuçlarının karşılaştırılmasına dayanarak dört kategori ile sınıflandırılmaktadır: 1) Gerçek pozitif (TP), gerçekte kusurlu olarak sınıflandırılmış kusur örneklerinin kusurlu olarak tahmin edilme sayısını belirtir; 2) Gerçek negatif (TN), gerçekte kusursuz örneklerin, kusursuz olarak tahmin edilme sayısını belirtir; 3) Yanlış pozitif (FP), gerçekte kusurlu örneklerin kusursuz örnekler olarak tahmin edilme sayısını belirtir; 4) Yanlış negatif (FN), gerçekte kusursuz örneklerin kusurlu olarak tahmin edilme sayısını belirtir (Ameri vd. 2024). Çizelge 4'te sunulan karşılaştırmalı veriler, YOLOv8'in her bir kusur türü için daha düşük yanlış pozitif (FP) ve yanlış negatif (FN) değerleri ile daha etkin bir performans sergilediğini göstermektedir. Örneğin, 'Açık Devre' kusur türünde YOLOv4 modeli 218 doğru pozitif (TP), 76 FP ve 74 FN kaydetmişken, YOLOv8 modeli 289 TP, sadece 25 FP ve 3 FN kaydetmiştir.

Duyarlılık (Recall), modelin tespit etmesi gereken tüm nesnelerden kaç tanesini doğru bir şekilde tespit ettiğini gösterir. Kesinlik (Precision), modelin tespit ettiği nesnelerden kaç tanesinin doğru olduğunu gösterir. Doğruluk (ACC) ise doğru sınıflandırılan örnek sayısının toplam örnek sayısına bölünmesiyle elde edilen değerdir. Bu değerler, sırasıyla Denklem (5), (6), (7) ve (8) ile hesaplanabilir.

$$\text{Duyarlılık (Recall)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (5)$$

$$\text{Kesinlik (Precision)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

$$\text{Doğruluk (ACC)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (8)$$

YOLOv4 ve YOLOv8 modelleri ile 235 test görüntüsü üzerindeki toplam 1512 kusur, Çizelge 5'te değerlendirilmiş ve karşılaştırılmıştır. YOLOv4 modelinde mAP@50 değeri tüm sınıflar için %64,1 iken, YOLOv8 için bu değer %93,5 olarak elde edilmiştir. Bu metrikler, modelin doğruluğunu, hassasiyetini ve güvenilirliğini daha kapsamlı bir şekilde değerlendirilmesine olanak tanımaktadır ve modelin çeşitli hata türlerini ne kadar doğru tespit edebildiğini göstermektedir.

Çizelge 4. YOLOv4 ve YOLOv8 Modelinin Sınıflara göre Performans Metrikleri

	YOLOv4			YOLOv8		
Kusur Tipi	**TP**	**FP**	**FN**	**TP**	**FP**	**FN**
Kısa Devre	150	52	59	199	22	8
Açık Devre	218	76	74	289	25	3
Aşırı Aşınma	212	119	110	315	44	7
Yetersiz Aşınma	183	85	80	259	31	4
Sahte Bakır Parçalar	181	58	48	226	37	3
Delik Hatlar	168	30	31	197	22	2

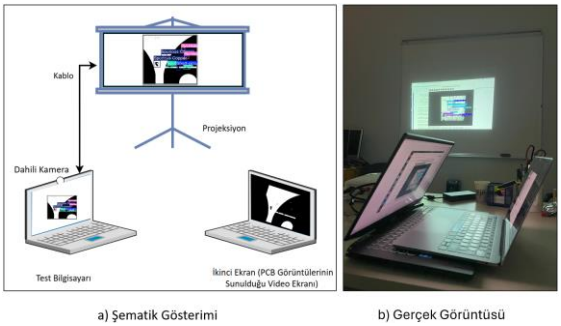
Çizelge 5. YOLOv4 ve YOLOv8 Değerlendirme Parametreleri

	YOLOv4					YOLOv8				
Sınıf	**Doğruluk Oranı (%)**	**mAP@50 (%)**	**Duyarlılık (%)**	**Kesinlik (%)**	**F1 Skor**	**Doğruluk Oranı (%)**	**mAP@50 (%)**	**Duyarlılık (%)**	**Kesinlik (%)**	**F1 Skor**
Kısa Devre	57,4	60,5	71,7	62,8	66,9	96,1	96,3	93,2	92,7	93,0
Açık Devre	59,2	64,2	74,6	65,8	69,9	98,9	90,5	89,7	88,2	89,0
Aşırı Aşınma	48	50,6	65,8	51,1	57,5	97,8	90,7	89,7	88,9	89,3
Yetersiz Aşınma	52,5	57,9	69,5	55,9	62	98,4	94,4	94,3	94,1	94,2
Sahte Bakır Parçalar	63	70,9	79	71,1	74,8	98,6	93,0	88,1	90,6	89,3
Delik Hatlar	73,3	74,7	84,4	78,0	81,1	98,9	96,3	93,5	93,9	93,7

3.1 Gerçek Zamanlı Deney Simülasyonu

Bu çalışmada geliştirilen derin öğrenme modeli, PCB üzerindeki kusurları tespit etmek amacıyla gerçek zamanlı olarak uygulanmıştır. Gerçek zamanlı deney simülasyonunun temel amacı, modelin pratik uygulama potansiyelini göstermektir. Modelin performansını gerçek dünya koşullarına benzer bir ortamda test etmek için, PCB görselleri bir ekranda gösterilmiş ve bu görseller farklı bir bilgisayarın dahili kamerası aracılığıyla algılanmış ve test veri seti model tarafından test edilmiştir. Simülasyon için, Şekil 8'de gösterildiği gibi, YOLOv8 modelinin gerçek zamanlı test kodunun çalıştırıldığı bilgisayar, bu test kodunun yansıtıldığı bir projeksiyon ekranı ve PCB görsellerinin gösterildiği ikinci bir bilgisayar olmak üzere toplam 3 donanım kullanılmıştır. Test bilgisayarında çalışmalar Intel(R) Core(TM) i7-6700K CPU @ 4.00GHz CPU özellikli VirtualBox sanal makinesinde, Ubuntu 22.04 işletim sisteminde, Python 3.10.12 ile Jupyter Notebook ortamında gerçekleştirilmiştir. İkinci ekranda ise görseller, beş saniye aralıklarla değiştirilmiştir. Test bilgisayarının

kamerasının açısı, görüntüleme cihazının ekranına doğru yönlendirilmiş ve ikinci ekranda farklı PCB görselleri sırayla tam ekran modunda video olarak gösterilmiştir. Her bir görsel için, modelin algılama sonuçları gerçek zamanlı olarak gözlemlenmiştir. Bu çalışmada gerçekleştirilen YOLOv8 modelimiz ile farklı kusur türlerini gerçek zamanlı nasıl tespitine yönelik video ilgili linkten izlenebilir. (İnt. Kay.1).



a) Şematik Gösterimi b) Gerçek Görüntüsü

Şekil 8. Gerçek Zamanlı Deney Düzeneği

4. Sonuçlar ve Tartışma

Bu çalışma, geliştirme sürecinin ilk adımında, PCB görüntülerinde çok sınıflı kusur tespiti, tanıma ve sınıflandırma için YOLOv4 modelinin performansını değerlendirmektedir. Analiz sonuçlarına göre, "Delik Hatlar" sınıfı %73,3 doğruluk, %78,0 ortalama kesinlik, %84,4 duyarlılık ve %81,1 F1 puanı ile en başarılı tespit edilmiş kusur türüdür. Bu sonuçlar, modelin öne çıkan kusur özelliklerini etkili bir şekilde tespit edebildiğini göstermektedir. Öte yandan, %48 doğruluk, %51,1 ortalama kesinlik, %65,8 duyarlılık ve %57,5 F1 skoru ile "Aşırı Aşınma" sınıfı modelin zorlandığı bir alan olarak tespit edilmiş ve modelin bu sınıf için geliştirilmesi gerektiği vurgulanmıştır.

Çok sınıflı kusur tespiti için %64,1'lik mAP değeri (Çizelge 5'te verilen her sınıf için ortalama kesinlikler), modelin aynı anda birden fazla kusur türünü tanıma becerisinde bir miktar başarı gösterdiğine işaret etmektedir. Bununla birlikte, bu değer modelin potansiyelini tam olarak yansıtmamakta ve performansında daha fazla iyileştirme için hala yer olduğunu göstermektedir. Bu çalışmada elde edilen YOLOv4 ve YOLOv8 sonuçları, literatürde yer alan diğer çalışmalarla birlikte Çizelge 6'da detaylı olarak sunulmaktadır. Literatürde kullanılan farklı veri setleri, sınıf sayıları ve modellerin performans metrikleri göz önünde bulundurulduğunda, önerilen modelin mAP@50 metriğinde çok sınıf-çoklu kusur kategorisinde rekabetçi bir başarı elde ettiği görülmektedir. Örneğin, bu çalışmadaki YOLOv4 modeli ile kusur sınıflandırması için %64,1 düşük bir mAP oranı gibi görünse de, literatürde çok sınıf-çok kusurlu veri setleri için YOLOv4 modelinde verilen orandan daha yüksektir ve Chen vd. tarafından yapılan çalışmada her kusur için farklı bir puan hesaplanmıştır. Ancak çalışmadaki çoğu modelde ortalama mAP değeri %50'yi aşmamıştır. Xie ve arkadaşlarının çalışmasında aynı veri seti ve YOLOv4 algoritmasıyla mAP@ 98,5 olarak elde edilmiştir. Bu çalışma ile Xie ve arkadaşlarının çalışmasında kullanılan veri seti her ne kadar aynı gözükse de bir miktar farklılık bulunmaktadır. Bu çalışmada veri setinde bulunan 1500 görselden sadece 1194'ü; ancak Xie ve arkadaşlarının çalışmasında ise 1500 görselin tamamı kullanılmıştır (Xie vd. 2023). Ayrıca Xie ve arkadaşlarının YOLOv4 modelinin eğitim sürecinde hiper parametreler genetik algoritmayla optimize edilmiştir. Bizim çalışmamızda ise YOLOv4'ün varsayılan standart değerleri kullanılmıştır.

Bu çalışmanın geliştirme sürecinin ikinci adımında, performansı artırmak amacıyla YOLOv8 modeli varsayılan standart parametrelerle kullanılmış ve gerçek zamanlı test bu model üzerinde çalıştırılmıştır. YOLOv8m modelinin mAP@50 değeri 93,5'tir ve PCB kusurlarını tespit etmede YOLOV4'e göre daha başarılı olmuştur. Modelin her sınıf için mAP değerleri %90'ı aşmış, özellikle kısa devre ve delik hatlar sınıfında maksimum %96,3 değerine ulaşmıştır. Bu sonuçlar, YOLOv8m modelinin, YOLOv4 modeline göre %29,4 daha yüksek mAP@50 değerine sahip olduğunu ve dolayısıyla daha iyi bir tespit performansı sunduğunu göstermektedir.

Bu tür çalışmalarda elde edilen mAP değerleri genellikle yüksektir. Ancak bu çalışmada olduğu gibi çok sınıflı çoklu kusur tespitine yönelik çalışmalar devam etmektedir ve bu alan halen aktif bir araştırma konusudur. Bu tür karmaşık senaryolarda mAP değerlerinin daha düşük olması beklenmektedir. Çünkü modelin aynı anda birden fazla sınıfı ve her sınıftan birden fazla kusuru tespit etmesi gerekmektedir. Bu durum mevcut modellerin sınırlarını zorlamakta ve algoritma geliştiricilerinin yeni iyileştirme yolları aramasını gerektirmektedir. Dolayısıyla bu çalışma, çok sınıflı çoklu kusur tespiti için derin öğrenme modellerinin geliştirilmesi ve iyileştirilmesi ihtiyacını vurgulamaktadır. Bu alanda yapılacak daha fazla araştırma, PCB üretim süreçlerindeki hata tespit sistemlerinin genel performansını artıracak ve endüstriyel otomasyonu daha da ilerletecektir. Simülasyondan da görüldüğü gibi yaklaşık 1 görsellik bir gecikme ile kusur tespiti yapılmaktadır. Modelin tespit doğruluğunu artırırken bir yandan da hızının artırılması önemlidir. Gelecek çalışmalarda modelin FPGA üzerinde donanım hızlandırmasının kusur tespiti üzerindeki etkilerini incelemek hedeflenmektedir.

Çizelge 6. Derin Öğrenme Tabanlı Kusur Tespiti Üzerine Odaklanan Çalışmalar

Referans	Veri Seti	Sınıf Sayısı	Görseldeki Kusur Durumu	Model	mAP @50 (%)
Sezer, A., & Altan, A. (2021)	274 görüntü üzerinde 324 kusur bulunmaktadır.	6	Çoklu Kusur	CNN	96,40
Adibhatla, V. A., Chih, H. C., Hsu, C. C., Cheng, J., Abbod, M. F., & Shieh, J. S. (2021)	23.000 görüntü kullanılmış olup, eğitim amaçlı 20.700 görüntü ve test amaçlı 2300 görüntü olmak üzere ayrılmıştır.	2 (Kusurlu veya kusursuz)	Çoklu Kusur	YOLOv5	99,74
Zhang, Y., Xie, F., Huang, L., Shi, J., Yang, J., & Li, Z. (2021)	1455 eğitim görüntüsü, 624 test görüntüsü içerir.	6	Çoklu Kusur	Feature Pyramid Network (FPN)	44,3
Luo, J., Yang, Z., Li, S., & Wu, Y. (2021)	1124 görüntü üzerinden etiketlenmiş 2556 adet kusur bulunmaktadır.	7	Çoklu Kusur	CNN	94,15
Xuan, W., Jian-She, G., Bo-Jie, H., Zong-Shan, W., Hong-Wei, D., & Jie, W. (2022)	2654 orijinal PCB yüzey kusuru görüntüsü veri artırma işleminden sonra, 18.572 görüntüye çıkarılmıştır. 18.572 görüntü üzerinde 37675 adet kusur bulunmaktadır.	6	Tekli Kusur	YOLOX	99,13
Xie, Y., Hu, W., Xie, S., & He, L. (2023)	DeepPCB veri seti ile 1500 görüntü	6	Çoklu Kusur	FE-YOLO YOLOv4	98,9 98,5
LI, W. (2024)	İnternet kaynakları ve Roboflow'un geniş PCB arıza görüntüleri koleksiyonunun birleşiminden derlenmiştir.	5	Tekli Kusur	YOLOv8x	98,9
Du, M., Chen, M., Cao, X., & Takamasu, K. (2024)	1370 görüntü üzerinden etiketlenmiş 2046 adet kusur bulunmaktadır.	5	Tekli Kusur	RSE-YOLO	93,9
Chen, Z., Zhu, Q., Zhou, X., Deng, J., & Song, W. (2024)	10.000 deri yüzey kusuru görüntüsü bulunmaktadır.	8	Çoklu Kusur	YOLOv8x	52,3
Bu çalışma	DeepPCB veri seti ile 1194 görüntü üzerinden etiketlenmiş 7.899 adet kusur bulunmaktadır.	6	Çoklu Kusur	YOLOv8m YOLOv4	93,5 64,1

Etik Standartlar Bildirgesi

Bu çalışma Prof.Dr.Hayriye KORKMAZ ve Dr.Sezen BAL danışmanlığında İzemnur BUDAK tarafından 17/09/2024 tarihinde tamamlanan "FPGA Tabanlı PCB Hata Tespitinde Derin Öğrenme Uygulaması" başlıklı ve 10678126 tez no'lu yüksek lisans tezinden türetilmiştir.

Yazarlık Katkı Beyanı

Yazar 1: Araştırma, Biçimsel analiz, Doğrulama, Metodoloji, Deney, Yazma, Görselleştirme, Kaynaklar

Yazar 2: Fikir Sahibi, Araştırma, Deney, Biçimsel analiz, Doğrulama, Metodoloji

Yazar 3: Proje Yöneticisi, Araştırma, Deney, Görselleştirme, Yazma, Kaynaklar

Çıkar Çatışması Beyanı

Yazarların bu makalenin içeriğiyle ilgili olarak beyan edecekleri hiçbir çıkar çatışması yoktur.

Verilerin Kullanılabilirliği

Bu çalışma sırasında kullanılan veri seti açık kaynak olup https://github.com/tangsanli5201/DeepPCB linkinden indirilebilir.

5. Kaynaklar

Adibhatla, V. A., Chih, H. C., Hsu, C. C., Cheng, J., Abbod, M. F., & Shieh, J. S., 2021. Applying deep learning to defect detection in printed circuit boards via a newest model of you-only-look-once. *Mathematical Biosciences and Engineering,* **18**,**4**, 4411-4428 https://doi.org/10.3934/mbe.2021223

Ameri, R., Hsu, C. C., & Band, S. S., 2024. A systematic review of deep learning approaches for surface defect detection in industrial applications. *Engineering Applications of Artificial Intelligence*, **130**, 107717. https://doi.org/10.1016/j.engappai.2023.107717

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M., 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*. https://doi.org/10.48550/arXiv.2004.10934

Chen, Z., Zhu, Q., Zhou, X., Deng, J., & Song, W., 2024. Experimental Study on YOLO-Based Leather Surface Defect Detection. *IEEE Access,* **12***,* 32830-32848 https://doi.org/10.1109/ACCESS.2024.3369705

Ding, R., Dai, L., Li, G., & Liu, H., 2019. TDD-net: a tiny defect detection network for printed circuit boards. *CAAI Transactions on Intelligence Technology, **4*****(2)**, 110-116. http://dx.doi.org/10.1049/trit.2019.0019

Du, M., Chen, M., Cao, X., & Takamasu, K., 2024. Deep Learning-Based Multi-Species Appearance Defect Detection Model for MLCC. *Ieee Transactions on Instrumentation and Measurement,***73** https://doi.org/10.1109/TIM.2024.3375957

Hassan, M. H., Reiter, E., & Razzaq, M., 2024. Automatic ovarian follicle detection using object detection models. http://dx.doi.org/10.21203/rs.3.rs-4637709/v1

Kim, J., Ko, J., Choi, H., & Kim, H., 2021. Printed circuit board defect detection using deep learning via a skip-connected convolutional autoencoder. *Sensors,* **21(15)**, 4968. https://doi.org/10.3390/s21154968

Li, W. H., Zhang, H. O., Wang, G. L., Xiong, G., Zhao, M. H., Li, G. K., & Li, R. S., 2023. Deep learning based online metallic surface defect detection method for wire and arc additive manufacturing. *Robotics and Computer-Integrated Manufacturing,* **80**, 12. http://dx.doi.org/10.1016/j.rcim.2022.102470

Li, Z., Chen, J., Huang, H., & Dong, X., 2023. *Defect Detection in Computer Motherboard Assembly through Fusion of Multi-Scale Features and Attention Mechanisms.* Paper presented at the 2023 9th International Conference on Systems and Informatics (ICSAI). http://dx.doi.org/10.1109/ICSAI61474.2023.10423311

Liao, X., Lv, S., Li, D., Luo, Y., Zhu, Z., & Jiang, C., 2021. YOLOv4-MN3 for PCB surface defect detection. *Applied Sciences,* **11(24)**, 11701. https://doi.org/10.3390/app112411701

Lin, Q., Zhou, J., Ma, Q., Ma, Y., Kang, L., & Wang, J., 2022. EMRA-Net: A pixel-wise network fusing local and global features for tiny and low-contrast surface defect detection. IEEE Transactions on Instrumentation and Measurement, **71**, 1-14. https://doi.org/10.1109/TIM.2022.3151926

Ling, Q., & Isa, N. A. M., 2023. Printed circuit board defect detection methods based on image processing, machine learning and deep learning: A survey. *Ieee Access,* **11**, 15921-15944. https://doi.org/10.1109/ACCESS.2023.3245093

Luo, J., Yang, Z., Li, S., & Wu, Y., 2021. FPCB surface defect detection: A decoupled two-stage object detection framework. *Ieee Transactions on Instrumentation and Measurement,* **70**, 1-11. http://dx.doi.org/10.1109/TIM.2021.3092510

Mahrishi, M., Morwal, S., Muzaffar, A. W., Bhatia, S., Dadheech, P., & Rahmani, M. K. I., 2021. Video index point detection and extraction framework using custom YoloV4 Darknet object detection model. *Ieee Access,* **9**, 143378-143391. http://dx.doi.org/10.1109/ACCESS.2021.3118048

Ocepek, M., Žnidar, A., Lavrič, M., Škorjanc, D., & Andersen, I. L., 2021. DigiPig: First developments of an automated monitoring system for body, head and tail detection in intensive pig farming. *Agriculture,* **12(1)**, 2. https://doi.org/10.3390/agriculture12010002

Redmon, J., 2016. You only look once: Unified, real-time object detection. *IEEE conference on computer vision and pattern recognition.* 779-788 https://doi.org/10.1109/CVPR.2016.91

Sharma, V. K., & Mir, R. N., 2020. A comprehensive and systematic look up into deep learning based object detection techniques: A review. *Computer Science Review,* **38**, 100301. http://dx.doi.org/10.1016/j.cosrev.2020.100301

Sezer, A., & Altan, A., 2021. Detection of solder paste defects with an optimization-based deep learning model using image processing techniques. *Soldering & Surface Mount Technology*, **33(5)**, 291-298. https://doi.org/10.1108/SSMT-04-2021-0013

Subramanyam, V., Kumar, J., & Singh, S. N., 2022. Temporal synchronization framework of machine-vision cameras for high-speed steel surface inspection systems. *Journal of Real-Time Image Processing, **19*****(2)**, 445-461. http://dx.doi.org/10.1007/s11554-022-01198-z

Tang, S., He, F., Huang, X., & Yang, J., 2019. Online PCB defect detector on a new PCB defect dataset. arXiv preprint arXiv:1902.06197. https://doi.org/10.48550/arXiv.1902.06197

Terven, J., Córdova-Esparza, D.-M., & Romero-González, J.-A., 2023. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction,* **5(4)**, 1680-1716. https://doi.org/10.3390/make5040083

LI, W., 2024. Detecting defects in PCB manufacturing: an exploration using Yolov8 deep learning. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 1-11. https://doi.org/10.1007/s12008-024-01986-w

Wang, X., Zhang, H., Liu, Q., Gong, W., Bai, S., & You, H., 2024. YOLO's multiple-strategy PCB defect detection model. *IEEE MultiMedia*. https://doi.org/10.1109/mmul.2024.3359267

Xie, Y., Hu, W., Xie, S., & He, L., 2023. Surface defect detection algorithm based on feature-enhanced YOLO. *Cognitive Computation*, **15(2)**, 565-579. https://doi.org/10.1007/s12559-022-10061-z

Xuan, W., Jian-She, G., Bo-Jie, H., Zong-Shan, W., Hong-Wei, D., & Jie, W., 2022. A lightweight modified YOLOX network using coordinate attention mechanism for PCB surface defect detection. IEEE sensors journal, **22(21)**, 20910-20920. https://doi.org/10.1109/JSEN.2022.3208580

Zhang, Q., & Liu, H., 2021. *Multi-scale defect detection of printed circuit board based on feature pyramid network,* **31**, 1, 76-87. http://dx.doi.org/10.1109/MMUL.2024.3359267

Zhang, Y., Xie, F., Huang, L., Shi, J., Yang, J., & Li, Z., 2021. A lightweight one-stage defect detection network for small object based on dual attention mechanism and PAFPN. *Frontiers in Physics,* **9**, 708097. http://dx.doi.org/10.3389/fphy.2021.708097

İnternet kaynakları

1. Budak,İ. *Gerçek Zamanlı PCB Kusur Tespiti* [Video dosyası]. https://drive.google.com/drive/folders/1fS9EMJAtJaci67d6lKK-qT6VJc5-4Vj_?usp=sharing (25.11. 2024)
2. Make Sense, https://www.makesense.ai/, (10.08. 2024)
3. Labelbox, https://labelbox.com/, (10.08. 2024)
4. LabelImg, https://medium.com/@pat.x.guillen/setting-up-labelimg-1ad1887d91c5, (10.08. 2024)
5. Labelme, https://github.com/labelmeai/labelme, (10.08. 2024)
6. Roboflow, https://roboflow.com/, (10.08. 2024)
7. SuperAnnotate, https://superannotate.medium.com/introduction-to-object-detection-with-deep-learning-f4d2b9f4d1b5, (10.08. 2024)
8. Supervisely, https://supervisely.com/labeling-toolbox/images/, (10.08. 2024)