

ÜRETİM SÜREÇLERİNDE KUSUR ORANLARININ SINIFLANDIRILMASI: YENİLİKÇİ KAREKOD DÖNÜŞÜMÜ İLE DERİN ÖĞRENME TABANLI BİR YAKLAŞIM

Yunus Emre GÜR¹, Mesut TOĞAÇAR², Bilal SOLAK³, Cem AYDEN⁴

Öz

Bu çalışma, üretim süreçlerindeki kusur oranlarının doğru sınıflandırılması ve kalite kontrol süreçlerinin optimize edilmesi için yenilikçi bir yöntem sunmaktadır. Çalışmada, sayısal veriler iki boyutlu QR kod görüntülerine dönüştürülerek AlexNet modeli ile analiz edilmiştir. Bu yöntem, derin öğrenme modellerinin güçlü desen tanıma yeteneklerinden yararlanarak kusur oranlarını yüksek doğrulukla sınıflandırmayı amaçlamaktadır. Veri seti, düşük ve yüksek kusur oranları olarak etiketlenmiş ve %80 eğitim, %20 test olarak bölünmüştür. Karar Ağacı, Gradient Boosting, K-En Yakın Komşu, Lojistik Regresyon, Saf Bayes, Rastgele Orman ve Destek Vektör Makinesi gibi çeşitli makine öğrenmesi modelleri ile karşılaştırılmıştır. Sonuçlar, AlexNet modelinin kusur oranlarını %100 doğrulukla sınıflandırdığını göstermektedir. Bu bulgular, derin öğrenme algoritmalarının üretim süreçlerindeki kalite kontrol ve kusur tespiti için son derece etkili olabileceğini vurgulamaktadır. Ayrıca, çalışmanın kısıtlılıkları ve gelecekteki araştırmalar için öneriler sunulmuştur. Bu yenilikçi metodoloji, diğer endüstriyel süreçlerde ve farklı veri setlerinde de geniş bir kullanım potansiyeline sahip olup, üretim verimliliğinin artırılmasına katkı sağlayacaktır.

Anahtar Kelimeler: Kusur Oranı Sınıflandırma, Makine Öğrenmesi, Derin Öğrenme, Karekod Dönüşümü, AlexNet

JEL Kodları: C38, C45, O14

CLASSIFICATION OF DEFECT RATES IN PRODUCTION PROCESSES: AN APPROACH BASED ON DEEP LEARNING WITH INNOVATIVE QR CODE CONVERSION

Abstract

This study presents an innovative method for accurately classifying defect rates in manufacturing processes and optimizing quality control processes. In the study, numerical data are transformed into two-dimensional QR code images and analyzed with the AlexNet model. This method aims to classify defect rates with high accuracy by utilizing the powerful pattern recognition capabilities of deep learning models. The dataset was labeled as low and high defect rates and divided into 80% training and 20% testing. It is compared with various machine learning models such as Decision Tree, Gradient Boosting, K-Nearest Neighbor, Logistic Regression, Naive Bayes, Random Forest and Support Vector Machine. The results show that the AlexNet model classifies defect rates with 100% accuracy. These findings emphasize that deep learning algorithms can be highly effective for quality control and defect detection in manufacturing processes. Furthermore, limitations of the study and suggestions for future research are presented. This innovative methodology has the potential to be widely used in other industrial processes and different data sets, contributing to the improvement of production efficiency.

Keywords: Defect Rate Classification, Machine Learning, Deep Learning, QR Code Conversion, AlexNet

JEL Codes: C38, C45, O14

¹ Dr. Öğr. Üyesi, Fırat Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, yegur@firat.edu.tr, <https://orcid.org/0000-0001-6530-0598>

² Doç. Dr., Fırat Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, mtogacar@firat.edu.tr, <https://orcid.org/0000-0002-8264-3899>

³ Doç. Dr., Fırat Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, bilalsolak23@gmail.com, <https://orcid.org/0000-0002-7804-2038>

⁴ Doç. Dr., Fırat Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, cayden@firat.edu.tr, <https://orcid.org/0000-0002-7648-7973>

GİRİŞ

Üretim süreçlerindeki kusur oranlarının minimize edilmesi, işletmelerin kalite kontrol ve operasyonel verimliliği açısından kritik bir öneme sahiptir. Modern üretim tesislerinde, üretim hattında meydana gelen kusurlar, doğrudan maliyetlerin artmasına, üretim verimliliğinin azalmasına ve müşteri memnuniyetsizliğine neden olabilir. Kusurların tespiti ve sınıflandırılması, üretim süreçlerinin optimize edilmesi ve üretim kalitesinin artırılması için temel adımlardan biridir. Bu nedenle, üretim hattında meydana gelen kusurların hızlı ve doğru bir şekilde tespit edilmesi ve sınıflandırılması, rekabet avantajı sağlamak adına büyük bir önem taşımaktadır (Chen vd., 2021).

Üretim süreçlerindeki hata oranları, malzeme bilimi, bilgisayar bilimi, fizik ve mühendislik gibi çeşitli alanlardan etkilenir. Kompozit malzemeler, üretim sürecinde kusurlara karşı hassastır ve titiz kalite kontrol gerektirir (Belnoue vd., 2018). CNN gibi teknolojiler, kusur tespit süreçlerini otomatikleştirerek kalite kontrolü iyileştirir (Wang ve Chen, 2020; Breitenbach vd., 2022). Hata modu ve etkileri analizi, potansiyel hataların belirlenmesinde stratejik bir yaklaşımdır (Borkovskaya ve Passmore, 2018). Makine öğrenmesi, sıfır hatalı üretime ulaşmada değerli bir araçtır (Yoshida, Iwasawa, Sano, Tanaka, Suzuki, 2017; Papageorgiou vd., 2022). İleri teknolojiler ve kalite kontrol metodolojileri ile üreticiler kusurları tespit edip azaltarak ürün kalitesini ve operasyonel verimliliği artırabilir. Endüstri 4.0 ve dijitalleşme, bu gelişmelerin önemini artırmaktadır. Endüstri 4.0 ve dijitalleşme ile üretim süreçlerindeki veri miktarı ve çeşitliliği artmaktadır (Burt vd., 2018). Bu verilerdeki karmaşık desenler, geleneksel yöntemlerle zor analiz edilir. Makine ve derin öğrenme teknikleri, bu büyük veri setlerindeki desenleri tanımlayarak daha doğru ve hızlı karar destek sistemleri geliştirir. Makine öğrenmesi, geçmiş verilerle gelecekteki olayları tahmin ederek potansiyel kusurları öngörebilir. Üretim sektöründe, büyük veri analizleri, verimliliği artırma ve kusur oranlarını azaltmada büyük önem taşır. Bu teknikler, potansiyel kusurları öngörerek önceden müdahale edilmesini sağlar.

Bu çalışmanın amacı, üretim süreçlerindeki kusur oranlarını doğru bir şekilde sınıflandırmak ve bu sayede üretim verimliliğini artırmaktır. Bu bağlamda, çalışmada, makine öğrenmesi modellerinin performansları ile derin öğrenme algoritmalarının performansları kıyaslanmıştır. Derin öğrenme algoritmalarının kullanımını sağlamak amacıyla, csv uzantılı veri setindeki sayısal veriler, karekollara (QR kod) dönüştürülmüştür. Bu dönüşüm süreci, verilerin iki boyutlu görsel temsillerini oluşturarak, derin öğrenme modellerinin güçlü desen tanıma yeteneklerinden faydalanmayı amaçlamaktadır. Böylece, makine öğrenmesi ve derin öğrenme algoritmalarının kusur oranlarını sınıflandırma performansları karşılaştırılarak, üretim süreçlerinde hangi yöntemin daha etkili olduğu belirlenmeye çalışılacaktır. Bu yenilikçi yaklaşım, makine öğrenmesi ve derin öğrenme yöntemlerinin sınıflandırma performanslarının kıyaslanmasına olanak

tanımaktadır. Çalışmanın bulguları, üretim yönetimi ve kalite kontrol süreçlerine önemli katkılar sunmaktadır. Bu bulgular, endüstriyel uygulamalarda kalite kontrol süreçlerinin iyileştirilmesi ve üretim verimliliğinin artırılması için önemli bir yol gösterici olacaktır. Ayrıca, bu çalışmanın sunduğu yenilikçi metodolojinin, diğer endüstriyel süreçlerde ve farklı veri setlerinde de uygulanarak, geniş bir yelpazede kullanım potansiyeline sahip olacağı düşünülmektedir.

Üretim süreçlerindeki kusurların doğru bir şekilde sınıflandırılması, maliyetlerin azaltılması ve müşteri memnuniyetinin artırılması açısından kritik bir öneme sahiptir. Geleneksel yöntemler, büyük ve karmaşık veri setlerinde sınırlı performans göstermektedir. Derin öğrenme tabanlı algoritmalar, özellikle karmaşık desenleri tanımada önemli avantajlar sunmasına rağmen, mevcut çalışmalarda sayısal verilerin görsel formatlara dönüştürülerek derin öğrenme modellerine entegrasyonu yeterince araştırılmamıştır. Bu bağlamda, araştırma aşağıdaki boşlukları doldurmayı amaçlamaktadır:

- Makine öğrenmesi ve derin öğrenme yöntemlerinin üretim süreçlerindeki kusur oranlarını sınıflandırmada doğrudan karşılaştırılması üzerine literatürde sınırlı çalışma bulunmaktadır.
- Sayısal verilerin QR kod formatına dönüştürülmesiyle derin öğrenme modellerinin performansını artırma potansiyeli henüz kapsamlı bir şekilde araştırılmamıştır.

Bu çalışmada aşağıdaki araştırma sorularına yanıt aranmaktadır:

1. Sayısal verilerin QR kod formatına dönüştürülmesi, derin öğrenme modellerinin sınıflandırma performansını nasıl etkilemektedir?
2. AlexNet modeli, üretim süreçlerindeki kusur oranlarını sınıflandırmada mevcut makine öğrenmesi modellerine kıyasla daha iyi bir performans sergilemekte midir?
3. Önerilen metodoloji, endüstriyel kalite kontrol süreçlerine nasıl katkı sağlayabilir?

Çalışmanın ilerleyen bölümlerinde, literatür taraması yapılarak kusur oranlarının tespitinde kullanılan mevcut yöntemler ve bu çalışmanın literatüre katkıları incelenmiştir. Ardından, kullanılan veri seti, veri ön işleme süreçleri ve önerilen metodoloji detaylandırılmış, verilerin QR kodlara dönüştürülme adımları açıklanmıştır. Bir sonraki bölümde, çalışmada kullanılan makine öğrenmesi ve derin öğrenme modellerinin performansları değerlendirilmiş ve elde edilen sonuçlar karşılaştırılmıştır. Son bölümde ise, bulgular tartışılmış, çalışma sonuçları özetlenmiş ve gelecekte yapılabilecek araştırmalar için öneriler sunulmuştur. Bu bağlamda, çalışma hem metodolojik yenilik sunmakta hem de endüstriyel kalite kontrol uygulamalarına katkı sağlamaktadır.

LİTERATÜR İNCELEMESİ

Üretim süreçlerindeki kusur oranlarının minimize edilmesi ve kalite kontrolünün optimize edilmesi, uzun yıllardır hem akademik araştırmaların hem de endüstriyel uygulamaların odak noktası olmuştur. Gelişen teknoloji ile birlikte, üretim süreçlerinde toplanan verilerin analiz edilmesi ve bu verilerden anlamlı sonuçlar çıkarılması daha da önemli hale gelmiştir. Üretim süreçlerinde kusur tespiti ve sınıflandırılması konusundaki mevcut literatür incelendiğinde, makine öğrenmesi ve derin öğrenme tekniklerinin kullanıldığı çeşitli çalışmalar değerli bilgiler sunmaktadır.

Ademujimi, Brundage, Prabhu (2017), üretim sistemlerindeki hata teşhisi için makine öğrenmesi algoritmalarını inceledi. Bayesian Ağları (BN), Yapay Sinir Ağları (ANN), Destek Vektör Makineleri (SVM) ve Gizli Markov Modelleri (HMM) gibi teknikler değerlendirildi. BN ve HMM, hiyerarşik modellemede başarılı; ANN doğrusal olmayan problemleri çözmeye etkili ancak aşırı öğrenmeye eğilimli; SVM küçük veri setlerinde bile iyi genelleme yapabiliyor; HMM, gözlemlenemeyen durumları modellemede başarılı fakat hesaplama yoğun. Demetgül (2013) çalışmasında, SVM ve karar ağaçları ile %100 sınıflandırma oranı elde edilmiştir. Lee, Cheon, Kim (2017), yarı iletken üretimde arıza tespiti için FDC-CNN modelini geliştirdi. Bu model, multivariate sensör sinyallerinden %97.9 doğrulukla hata özelliklerini çıkararak diğer modellerden daha iyi performans gösterdi. FDC-CNN, hızlı eğitim süresi ve önemli sensör değişkenlerini belirleyebilme yeteneği sunmaktadır. Bu sonuçlar, makine öğrenmesi ve derin öğrenme tekniklerinin üretim hatalarını teşhis etmede etkili olduğunu göstermektedir.

Essid, Laga, Samir (2018), metal kutu üretiminde kusurların tespiti için otoenkoder derin sinir ağı kullanarak %99.80 doğruluk elde etmiştir. Yanlış pozitif ve yanlış negatif oranları sırasıyla %10.6 ve %5.41 olarak belirlenmiştir. Bu yöntem, geleneksel teknikleri geride bırakmıştır. Escobar ve Morales-Menendez (2018), yüksek uyumlu üretim ortamlarında nadir kalite olaylarının tespiti için 11-düzenlemeli lojistik regresyon kullanarak %100 hata tespit oranına ulaşmıştır. Kim, Oh, Jung, Kim (2018), dengesiz veri kümeleriyle üretim kalitesini sınıflandırmak için maliyet duyarlı karar ağacı toplulukları kullanmış, özellikle AdaC2 yöntemi F-measure, G-means ve AUC gibi ölçütlerde en iyi sonuçları vermiştir. Bu çalışmalar, derin öğrenme ve maliyet duyarlı topluluk yöntemlerinin üretim süreçlerinde kusur tespitinde etkinliğini göstermektedir.

Yang vd. (2020), üretim süreçlerinde derin öğrenme kullanarak %99 doğruluk oranı ile ürün kusurlarını tespit etmişlerdir. Ruiz vd. (2020), uçak üretiminde Konvolüsyonel Sinir Ağları (CNN) kullanarak sabitleme elemanlarını %98.3 doğrulukla belirlemişlerdir. Kotsiopoulos, Sarigiannidis, Ioannidis, Tzovaras (2021), düşük kaliteli kaynakları L1-regularize Lojistik Regresyon yöntemiyle %99.80

doğruluk oranı ile tespit etmişlerdir. Chien, Hung, Liao (2022), yarı iletken üretiminde VGG-16 tabanlı bir CNN modeli ile %98.3 doğruluk elde etmişlerdir. Blondheim (2022), elektronik bileşenlerde CNN, otoenkoder ve derin sinir ağları kullanarak kusurları tespit etmiş ve model performansını Grad-CAM ve transfer öğrenme teknikleri ile değerlendirmiştir. Abdallah vd. (2023), sensör verileri ile anomali tespiti yaparak Random Forest ve AutoEncoder yöntemleri ile en iyi performansı sırasıyla %83.33 ve %78.33 doğruluk oranlarıyla sağlamış, transfer öğrenimi ile MEMS sensöründe %71.71 doğruluk elde etmişlerdir. Singh ve Desai (2023), ResNet-101 ve SVM kullanarak otomatik yüzey kusuru tespitinde %100 doğruluk oranına ulaşmış, EfficientNet-b0 modeli ise %99.91 doğruluk ile en iyi yüzey kusuru tespitini gerçekleştirmiştir. Nuhu, Zeeshan, Safaei, Shahzad (2023), makine öğrenmesi ile anomali tespitinde yüksek performans elde etmiş, transfer öğrenimi doğruluğu artırmıştır. Herzog, Brandt, Trinchi, Sola, Molotnikov (2024), lazer bazlı metal imalatta CNN ve SVM kullanarak %95 doğruluk oranı elde etmiş, KNN ve Karar Ağaçları %85 ile %90 arasında doğruluk oranları sağlamış ve SVM için ortalama AUC değeri 0.93 olarak rapor edilmiştir. CNN'lerin özellikle yüksek çözünürlüklü görüntü analizinde etkili olduğu vurgulanmıştır. Mevcut literatürde, üretim süreçlerinde kusur tespiti için makine öğrenmesi ve derin öğrenme teknikleri kullanılmış olmasına rağmen, bu tekniklerin doğrudan karşılaştırılması üzerine yapılan çalışmalar sınırlıdır. Bu çalışma, sayısal verilerin QR kodlara dönüştürülmesi yoluyla derin öğrenme modellerinin desen tanıma yeteneklerinden yararlanarak yenilikçi bir yaklaşım sunmaktadır. Makine öğrenmesi ve derin öğrenme tekniklerinin kusur sınıflandırma performanslarını doğrudan karşılaştırarak hangi yöntemin daha etkili olduğunu belirlemeyi amaçlamaktadır. Literatürde bu tür karşılaştırmaların azlığı göz önünde bulundurulduğunda, bu çalışma bu eksikliği gidermeyi hedeflemektedir. Sonuç olarak, QR kodlar ve derin öğrenme modelleri kullanılarak kusur oranlarının yüksek doğrulukla sınıflandırılması amaçlanmakta olup, bu metodolojinin diğer endüstriyel süreçlerde ve veri setlerinde de uygulanabilirliği ve kalite kontrol süreçlerinin iyileştirilmesine katkı sağlayabileceği öne sürülmektedir.

METODOLOJİ

Bu bölümde, çalışmada kullanılan veri seti, veri ön işleme adımları, model seçimi ve eğitimi, karekod dönüşüm süreci, kullanılan modeller ve model değerlendirme kriterleri detaylı olarak açıklanmaktadır.

Veri Seti

Bu çalışmada, 2024 yılında “Üretim Kusurlarını Tahmin Etme” başlığı altında Kaggle platformunda yayınlanan veri seti kullanılmıştır (Kharoua, 2024). Veri seti, üretim operasyonlarındaki kusur oranlarının sınıflandırılması için kapsamlı bir ölçüm koleksiyonunu içermektedir. Veri seti, çeşitli üretim değişkenleri ile hedef değişken olarak kusur durumunu içermektedir. Hedef değişken ikili bir yapıya sahiptir ve düşük

kusurlar (0) ve yüksek kusurlar (1) olarak sınıflandırılmıştır (Kharoua, 2024). Veri setinin detaylı yapısı ve içeriği Tablo 1’de gösterilmektedir.

Tablo 1: Veri setinde kullanılan tüm değişkenler ve açıklamaları

Değişkenler	Açıklama
Üretim hacmi	Günlük üretilen birim sayısı (100 ile 1000 ünite/gün).
Üretim maliyeti	Günlük üretim için katlanılan maliyet (5000 Dolar ile 20000 Dolar arası).
Tedarikçi kalitesi	Tedarikçilerin kalite derecelendirmeleri (%80 ile %100 arası).
Teslimat gecikmesi	Teslimattaki ortalama gecikme (0 ile 5 gün).
Kusur oranı	Üretilen bin birim başına düşen kusur sayısı (0,5 ile 5,0 kusur).
Kalite puanı	Genel kalite değerlendirmesi (%60 ile %100 arası).
Bakım saatleri	Haftada bakıma harcanan saat (0 ile 24 saat).
Kesinti yüzdesi	Üretimdeki duruş süresinin yüzdesi (%0 ile %5 arası).
Stok devir hızı	Stok devir hızı oranı (2 ile 10 arası).
Stok tükenme oranı	Stokların tükenme oranları (%0 ile %10 arası).
Çalışan verimliliği	İşgücünün üretkenlik düzeyi (%80 ile %100 arası).
Güvenlik olayları	Aylık güvenlik olayı sayısı (0 ile 10 olay).
Enerji tüketimi	kWh cinsinden tüketilen enerji (1000 ile 5000 kWh).
Enerji verimliliği	Enerji kullanımının verimlilik faktörü (0,1 ile 0,5 arası).
Katkı işlem süresi	Katmanlı üretim için gereken süre (1 ile 10 saat arası).
Katkı Malzeme Maliyeti	Katkı maddelerinin birim başına maliyeti (100 ile 500 Dolar arası).
Kusur Durumu (Hedef Değişken)	Tahmini kusur durumu (0: Düşük kusurlar, 1: Yüksek kusurlar).

Kaynak: Kharoua, 2024

Veri seti toplam 17 kolondan oluşmaktadır. Bu kolonlardan 4 tanesi tam sayı (int64) veri tipindeyken, 13 tanesi kayan nokta (float64) veri tipindedir. Her kolon için 3240 veri girişi bulunmaktadır. Hedef değişken olarak, düşük kusur sınıf sayısı 517, yüksek kusur sınıf sayısı ise 2723’tür. Veri seti, üretim süreçlerindeki farklı değişkenleri kapsayarak, kusur oranlarının doğru bir şekilde sınıflandırılması için gerekli olan çeşitli bilgileri sağlamaktadır.

Veri Ön İşleme

Veri ön işleme, modelleme sürecinin önemli bir adımıdır ve verilerin analiz için uygun hale getirilmesini sağlar. Bu çalışmada eksik veri bulunmadığından eksik veri doldurma uygulanmamıştır. İlk

adımında, tüm değişkenler Python'un pandas kütüphanesi ile işlenmiş ve sklearn'in LabelEncoder fonksiyonu kullanılarak dönüştürülmüştür. Makine öğrenmesi modelleri ile sınıflandırma yapılırken veriler Min-Max normalizasyon yöntemi ile 0-1 aralığına ölçeklendirilmiştir. Bu işlem, farklı ölçeklerdeki verilerin uyumlu hale getirilmesi ve model performansının artırılması açısından önemlidir (Hidayaturrohman, Clarke, Taflan, Sancaktar, 2023). Normalizasyon işlemi Denklem 1'de gösterilmiştir.

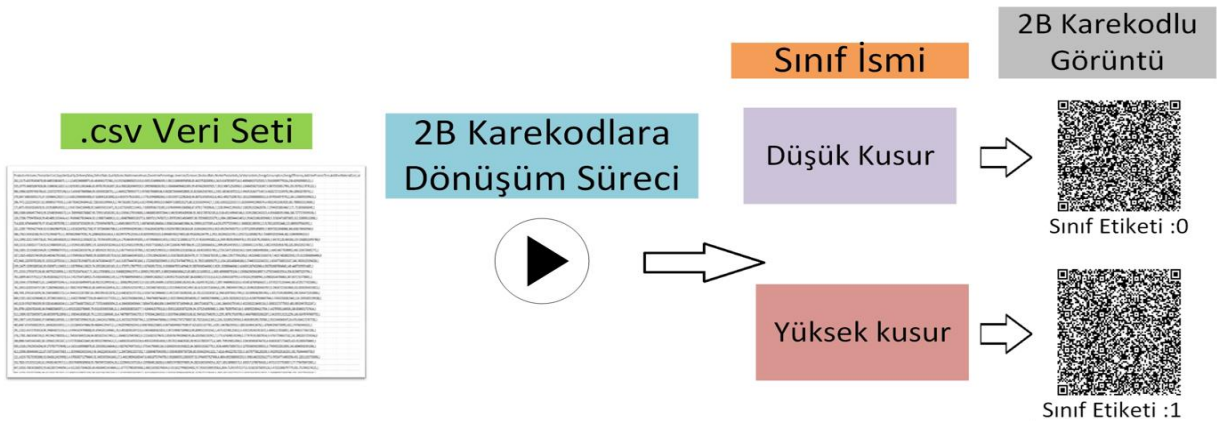
$$x_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Bu işlemler sonrasında, özellikler (X) ve hedef değişken (y) olarak iki ayrı veri seti oluşturulmuştur. Model eğitimi ve test işlemleri için scikit-learn kütüphanesinden faydalanılmıştır. Veri seti, "train_test_split" fonksiyonu ile eğitim ve test setlerine ayrılmıştır. Eğitim seti (X), modelin öğrenmesi için kullanılırken, test seti (y) modelin genelleştirme performansını değerlendirmek için kullanılmıştır. Oluşturulan X ve y veri setleri, eğitim ve test setleri olmak üzere ikiye ayrılmıştır. Bu ayrım, veri setinin %80'i eğitim, %20'si test olacak şekilde ayarlanmıştır. Eğitim seti, modelin öğrenmesi için; test seti ise modelin performansının değerlendirilmesi için kullanılmıştır.

Veri Dönüşümü

Bu çalışmanın yenilikçi yönlerinden biri, sayısal verilerin QR kodlara dönüştürülmesidir. Bu işlem, derin öğrenme modellerinin kullanılabilirliği için Python kullanılarak yapılmıştır. qrcode, numpy ve PIL kütüphanelerinden yararlanılarak, veri noktaları önce string formatına çevrilmiş, sonra QR kodlara dönüştürülmüştür. Oluşturulan QR kodlar, PIL kullanılarak 224x224 piksel boyutunda yeniden boyutlandırılmış ve RGB formatına dönüştürülmüştür. numpy ile array formatında saklanan bu veriler, AlexNet modeli ile sınıflandırma için hazırlanmıştır. Veri dönüşüm süreci, Şekil 1'de gösterilmektedir.

Şekil 1: Sayısal verilerin karekollara (qr kod) dönüşüm süreci



Yazılım ve Donanım

Bu çalışmada, veri işleme, model eğitimi ve değerlendirme süreçlerinde Python 3.8 kullanılmıştır. Veri manipülasyonu için Pandas, sayısal işlemler için NumPy, makine öğrenmesi modelleri için scikit-learn, QR kod oluşturma için qrcode, görüntü işleme için Pillow, derin öğrenme modelleri (özellikle AlexNet) için TensorFlow ve Keras kütüphaneleri kullanılmıştır. Model eğitim ve değerlendirmesi, Intel Core i7-9700K işlemci, NVIDIA GeForce RTX 2080 Ti GPU, 32 GB RAM ve 1 TB NVMe SSD donanımına sahip bir bilgisayarda gerçekleştirilmiştir. Bu donanım, büyük veri setlerini işlemek ve karmaşık modelleri eğitmek için yeterli performansı sağlamış, GPU hızlandırması ise eğitim süresini önemli ölçüde azaltmıştır.

Makine Öğrenmesi ve Derin Öğrenme Modelleri

Bu çalışmada, üretim süreçlerindeki kusur oranlarını sınıflandırmak için çeşitli makine öğrenmesi modelleri kullanılmıştır. Kullanılan modeller aşağıda tanıtılmaktadır.

Karar ağacı (Decision tree)

Karar ağaçları, sınıflandırma için yaygın olarak kullanılan denetimli makine öğrenimi algoritmalarıdır (Dudryev ve Kuznetsov, 2021). Bu algoritmalar, her iç düğümün bir özelliği temsil ettiği, her dalın bir kararı belirlediği ve her yaprak düğümünün sınıflandırmayı gösterdiği akış şeması benzeri bir yapı oluşturur (Karthikeyan ve Thangaraju, 2013). Karar ağaçları, yukarıdan aşağıya özyinelemeli olarak böl ve yönet yöntemiyle oluşturulur, bu da onları yorumlanabilir ve görsel olarak anlaşılabilir kılar (Safavian ve Landgrebe, 1991). Temel bileşenleri düğümler ve dallardır; adımları bölme, durdurma ve budamayı içerir (Mai, Chen, Yi, Ding, 2021). Karar ağaçları, veri kümesini belirli özelliklere göre bölme yetenekleri sayesinde hızlı tahminlerde verimlidir (Cao ve Chun-hai, 2012). Ağaçlar, bilgi kazancı veya Gini indeksini kullanarak dallanır, bunlar sırasıyla Denklem 2 ve 3'te gösterilmektedir.

Bilgi Kazancı (Information Gain):

$$IG(T, A) = H(T) - \sum_{v \in \text{values}(A)} \frac{|T_v|}{|T|} H(T_v) \quad (2)$$

Burada, $H(T)$ veri setinin entropisi, T_v ise özellik A 'nın v değerine sahip olan alt kümesidir.

Gini İndeksi (Gini Index):

$$Gini(T) = 1 - \sum_{i=1}^C p_i^2 \quad (3)$$

Burada C sınıf sayısı ve p_i sınıf i 'nin oranıdır.

Gradyan artırma (gradient boosting)

Gradient Boosting, modelin doğruluğunu artırmak için zayıf öğrenicileri, genellikle karar ağaçlarını yinelemeli olarak birleştiren bir tekniktir (Natekin ve Knoll, 2013). Bu süreç, önceki modellerin hatalarını düzelterek tahmin performansını her adımda iyileştirir (Wen, Wang, Huo, 2020). Nihai model, bu zayıf tahmincilerin topluluğundan oluşur (Wang, Li, Tu, 2019). Sınıflandırma için genellikle log-loss (lojistik kayıp) kullanılır. Bu, doğru sınıflandırma olasılığını artırmayı hedefler. Başlangıç modeli Denklem 4'te gösterilmektedir.

$$F_0(x) = \log\left(\frac{p}{1-p}\right) \quad (4)$$

Burada, p sınıfın olasılığıdır. Bununla birlikte iteratif model güncellemesi ise Denklem 5'teki gibidir.

$$r_{im} = - \left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x) = F_{m-1}(x)} \quad (5)$$

Burada, r_{im} , i 'inci örneğin m 'inci iterasyondaki rezidüelidir. $F_{m-1}(x)$ modelin i 'inci örnek için tahmin ettiği olasılıktır. Bu aşamadan sonra, rezidüeller üzerine yeni bir zayıf tahmin edici $h_m(x)$ eğitilmektedir. Daha sonra model güncellenmektedir. Bu işlem Denklem 6'da gösterilmektedir.

$$F_m(x) = F_{m-1}(x) + \eta h_m(x) \quad (6)$$

Burada, η öğrenme oranıdır ve genellikle 0 ile 1 arasında bir değere sahiptir. Modelin çıktısı, sigmoid fonksiyonu kullanılarak Denklem 7'de gösterildiği gibi sınıf olasılığına dönüştürülür.

$$P(y = 1|x) = \frac{1}{1 + e^{-F_m(x)}} \quad (7)$$

Burada M , toplam iterasyon sayısıdır.

K-En yakın komşu (K-Nearest neighbors-KNN)

K-En Yakın Komşular (KNN), veri noktalarını en yakın komşularının sınıfına göre sınıflandıran denetimli bir öğrenme algoritmasıdır (Bullejos, Cabezas, Martín-Martín, Alcalá, 2022). Bu algoritma, örnekleri doğrudan depolar ve sınıflandırma sırasında hesaplamalar yapar, bu da onu düşük boyutlu veri kümeleri için verimli kılar (Rizal, Purba, Siregar, Sinaga, Azizah, 2020). KNN, parametrik olmayan bir sınıflandırıcı olup, tahminler veya sınıflandırmalar için yakınlık ve benzerliği kullanır (Papernot, 2018).

Gürültülü verilerde sağlamdır ve büyük veri kümelerinde etkilidir (Wermelinger, 2023). Test veri noktası x ile her bir eğitim veri noktası x_i arasındaki Öklid uzaklığı Denklem 8’de gösterilmektedir.

$$d(x, x_i) = \sqrt{\sum_{j=1}^n (x_j - x_{ij})^2} \quad (8)$$

Burada, x_j test noktasının j ’inci özelliği ve x_{ij} eğitim noktasının j ’inci özelliğidir. Bu işlemin ardından, Hesaplanan uzaklıklar sıralanır ve en küçük k uzaklık seçilir. Bu uzaklıklara karşılık gelen veri noktaları, test noktasının komşuları olarak belirlenir. Ardından, k komşunun sınıf etiketleri alınır ve en sık görülen sınıf etiketi (mode) test noktası için Denlem 9’da ifade edildiği gibi tahmin edilir.

$$\hat{y} = mode(y_1, y_2, \dots, y_k) \quad (9)$$

Komşuların sınıf etiketleri arasında çoğunluk oyu kullanılarak test veri noktasının sınıf etiketi belirlenir. Örneğin, $k=5$ ise ve komşu etiketleri $[1, 0, 1, 1, 0]$ ise, sınıf etiketi 1 olarak tahmin edilir çünkü 1 sınıfı en sık görülen etikettir.

Lojistik regresyon (logistic regression)

Lojistik regresyon, ikili sınıflandırma görevlerinde verilerin belirli bir sınıfa ait olma olasılığını tahmin etmek için yaygın olarak kullanılır (Olorunlambe, Hua, Shepherd, Dearn, 2021). Bu tahmin, çıktıyı 0 ile 1 arasında bir değere eşleyen sigmoid fonksiyonuyla yapılır. Lojistik regresyon, verilerin doğrusal olarak ayrılabilir olduğunu varsaymaz; sınıfları ayırmak için doğrusal bir karar sınırı bulur. Bu yöntem, olasılıkları 0-1 aralığında tutarak sınıflandırma sonuçlarının net yorumlanmasını sağlar (Liu, Cui, Chen, 2023). Bağımsız değişkenlerin doğrusal kombinasyonunu kullanarak bir lineer model oluşturur, bu Denklem 10’da gösterilmiştir.

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n \quad (10)$$

Burada z , bağımsız değişkenlerin ağırlıklı toplamı, β_0 sabit terim ve β_i , x_i bağımsız değişkeninin katsayısıdır. Lojistik regresyon, lineer modelin çıktısını sigmoid fonksiyonuna uygulayarak olasılıkları Denklem 11’de gösterildiği gibi tahmin eder.

$$P(y = 1|x) = \sigma(z) = \frac{1}{1+e^{-z}} \quad (11)$$

Burada $\sigma(z)$, sigmoid fonksiyonudur ve z bağımsız değişkenlerin doğrusal kombinasyonudur. Model eğitildikten sonra, bağımsız değişkenler x için olasılıklar Denklem 12'deki gibi hesaplanmaktadır.

$$P(y = 1|x) = \frac{1}{1+e^{-(\beta_0+\beta_1x_1+\beta_2x_2+\dots+\beta_nx_n)}} \quad (12)$$

Bu formül, her bir veri noktasının pozitif sınıfa ($y = 1$) ait olma olasılığını vermektedir. Tahmin edilen olasılık, belirli bir eşik değeri ile karşılaştırılarak sınıf etiketi belirlenir. Genellikle bu eşik değer Denklem 13'te gösterildiği gibi 0.5 olarak alınmaktadır.

$$\hat{y} = \begin{cases} 1 & \text{eğer } P(y = 1|x) \geq 0.5 \\ 0 & \text{eğer } P(y = 1|x) < 0.5 \end{cases} \quad (13)$$

Saf bayes (Naive bayes)

Naive Bayes, özelliklerin bağımsız olduğunu varsayarak olasılık teorisini kullanan bir sınıflandırma algoritmasıdır (Wang, Wang, Li, 2007). Bu varsayıma rağmen, özelliklerin güçlü bir şekilde ilişkili olmadığı durumlarda etkilidir. Algoritma, bir veri noktasının belirli bir sınıfa ait olma olasılığını, her özelliğin o sınıfta bağımsız olarak ortaya çıkma olasılıklarına dayanarak hesaplar (Rafdi, Mawengkang, Efendi, 2021). Naive Bayes sınıflandırması, Bayes teoremi üzerine kuruludur.

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{P(x)} \quad (14)$$

Burada, $P(C_k|x)$, x veri noktası verildiğinde C_k sınıfının olasılığını, $P(x|C_k)$, C_k sınıfı verildiğinde x veri noktasının olasılığını, $P(C_k)$, C_k sınıfının önsel olasılığını ve $P(x)$, x veri noktasının genel olasılığını temsil etmektedir. Naive varsayımında, özelliklerin birbirinden bağımsız olduğu varsayılmaktadır.

$$P(x|C_k) = \prod_{i=1}^n P(x_i|C_k) \quad (15)$$

Burada x_i , x veri noktasının i 'inci özelliğidir. Naive Bayes sınıflandırıcısı, her sınıf için olasılığı Denklem 16'daki gibi hesaplamaktadır.

$$P(C_k|x) = \frac{P(C_k) \prod_{i=1}^n P(x_i|C_k)}{P(x)} \quad (16)$$

Ancak, $P(x)$ tüm sınıflar için sabit olduğundan, hesaplama sırasında genellikle göz ardı edilmektedir:

$$P(C_k|x) \propto P(C_k) \prod_{i=1}^n P(x_i|C_k) \quad (17)$$

Daha sonra, en yüksek olasılığa sahip sınıf, Denklem 18’de gösterildiği gibi tahmin edilen sınıf olarak seçilmektedir.

$$\hat{y} = \arg \max_{C_k} P(C_k) \prod_{i=1}^n P(x_i | C_k) \quad (18)$$

Rastgele orman (Random forest)

Rastgele Orman, birden fazla karar ağacını birleştirerek nihai bir sınıflandırma sonucu sağlayan bir topluluk öğrenme yöntemidir (Goel ve Abhilasha, 2017). Bu yöntem, bireysel ağaçların çıktılarının birleşimiyle sınıflandırma doğruluğunu ve sağlamlığını artırır. En önemli avantajlarından biri, ağaç çeşitliliği sayesinde aşırı uyum riskini azaltmasıdır. Bu topluluk yaklaşımı, karmaşık ilişkileri yakalayarak yüksek doğruluk sağlar (Wakayama vd., 2015). Rastgele Orman, B et karar ağacını rastgele örneklenmiş veri alt kümeleri üzerinde eğitir. Her ağaç için eğitim veri kümesi D_b Denklem 19’daki gibi oluşturulmaktadır.

$$D_b = \{(x_i, y_i)\}_{i=1}^{n_b} \quad (19)$$

Burada D_b , b ’inci karar ağacının eğitim veri kümesidir ve n_b bu alt kümedeki örnek sayısıdır.

Her karar ağacı için, her düğümde rastgele bir özellik alt kümesi seçilir. Bu alt küme, m sayıda özellik içerir ve bu özellikler arasından en iyi bölünme noktası seçilir. Her karar ağacı $h_b(x)$, test veri noktası x için Denklem 20’deki gibi sınıf tahmini yapmaktadır.

$$h_b(x) = \text{Karar Ağacının Sınıf Tahmini} \quad (20)$$

Rastgele Orman, her bir karar ağacının sınıf tahminlerini birleştirerek nihai sınıf tahminini yapar. Bu birleştirme işlemi, çoğunluk oylaması ile Denklem 21’deki gibi yapılır:

$$\hat{y} = \arg \max_{c \in C} \sum_{b=1}^B I(h_b(x) = c) \quad (21)$$

Burada I , gösterge fonksiyonudur ve $h_b(x) = c$ ifadesinin doğru olduğu durumlarda 1, aksi halde 0 değerini alır. C , olası sınıfların kümesidir ve \hat{y} , test veri noktası x için nihai sınıf tahminidir.

Destek vektör makinesi (Support vector machine-SVM)

Destek Vektör Makinesi (SVM), bir veri kümesindeki iki sınıfı etkili bir şekilde ayıran en uygun hiper düzlemi bulmayı amaçlayan bir sınıflandırma algoritmasıdır. Bu hiper düzlem, iki sınıf arasındaki

marjı en üst düzeye çıkarmak için stratejik olarak konumlandırılır ve modelin görünmeyen verilere iyi genelleme yeteneğini geliştirir (Xue-jun ve Yang, 2006). SVM, iki sınıfı ayıran hiper düzlemi Denklem 22’de gösterildiği gibi tanımlamaktadır:

$$w \cdot x + b = 0 \quad (22)$$

Burada w , hiper düzlemin normal vektörü ve b , bias terimidir.

SVM, iki sınıf arasındaki marjini maksimize etmeye çalışır. Bu, Denklem 23’te gösterilen optimizasyon problemini çözmek anlamına gelmektedir:

$$\min_{w,b} \frac{1}{2} \|w\|^2 \text{ subject to } y_i(w \cdot x_i + b) \geq 1, \forall_i \quad (23)$$

Burada $y_i \in \{-1,1\}$ sınıf etiketleri ve x_i veri noktalarıdır. Bununla birlikte, optimizasyon problemini çözmek için Lagrange çarpanları (α_i) kullanılmaktadır.

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i [y_i(w \cdot x + b) - 1] \quad (24)$$

Dual formda çözüm:

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i \cdot x_j \text{ subject to } \sum_{i=1}^n \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0 \quad (25)$$

Eğitim tamamlandıktan sonra, karar fonksiyonu Denklem 26’da gösterildiği gibi tanımlanmaktadır.

$$f(x) = \sum_{i=1}^n \alpha_i y_i (x_i \cdot x) + b \quad (26)$$

Test veri noktası x için sınıf etiketi, karar fonksiyonunun işaretine göre belirlenir:

$$\hat{y} = \text{sign}(f(x)) = \text{sign}(\sum_{i=1}^n \alpha_i y_i (x_i \cdot x) + b) \quad (27)$$

AlexNet

AlexNet, derin öğrenme alanında çığır açan bir Convolutional Neural Network (CNN) modelidir. 2012 yılında ImageNet Large Scale Visual Recognition Challenge (ILSVRC) yarışmasında büyük bir başarı elde ederek popülerlik kazanmıştır. AlexNet modeli, beş evrişim (convolutional) katmanı ve üç tam bağlı (fully connected) katmandan oluşur (Lv vd., 2020). Modelde, ReLU aktivasyon fonksiyonları, max-pooling katmanları, dropout ve veri artırma (data augmentation) teknikleri kullanılır (Oktaria, Prakasa, Suhartono, 2019).

Sınıflandırma işlemlerinde, giriş görüntüsü genellikle 227x227 piksel boyutunda ve 3 renk kanalıdan (RGB) oluşmaktadır:

$$X \in \mathbb{R}^{227 \times 227 \times 3} \quad (28)$$

Evrışim katmanları, görüntü üzerinde filtreler (kernels) uygulayarak özellik haritaları (feature maps) çıkarmaktadır:

$$h_{ij}^{(l)} = f(\sum_{m,n} W_{mn}^{(l)} \cdot X_{(i+m)(j+n)} + b^{(l)}) \quad (29)$$

Burada $h_{ij}^{(l)}$, l 'inci katmandaki aktivasyon, $W_{mn}^{(l)}$, filtre ağırlıkları, X giriş verisi ve $b^{(l)}$ bias terimidir. Aktivasyon fonksiyonu f ise genellikle ReLU ($\text{ReLU}(x) = \max(0, x)$) olarak kullanılmaktadır. Bununla birlikte, max-pooling katmanları, her bir özellik haritasının alt örneklemeler (subsampling) yaparak boyutunu azaltır:

$$h_{ij}^{(l)} = \max_{(m,n) \in \text{pool}} h_{(i+m)(j+n)}^{(l-1)} \quad (30)$$

Burada pool, havuzlama bölgesini ifade etmektedir. Tam bağlı katmanlar, tüm nöronların birbirine bağlı olduğu katmanlardır. Bu katmanlarda matris çarpımı ve aktivasyon fonksiyonu kullanılarak sınıflandırma yapılır:

$$h^{(l)} = f(W^{(l)} \cdot h^{(l-1)} + b^{(l)}) \quad (31)$$

Burada $W^{(l)}$, ağırlık matrisi, $h^{(l-1)}$, önceki katmanın aktivasyonları ve $b^{(l)}$ bias terimidir.

Son katman, ikili sınıflandırma sonucunu olasılık olarak veren sigmoid katmanıdır:

$$P(y = 1|X) = \sigma(z) = \frac{1}{1+e^{-z}} \quad (32)$$

Burada z , tam bağlı katmanın çıktısıdır. Modelin çıktısı, belirli bir eşik değeri (genellikle 0.5) ile karşılaştırılarak sınıf etiketi olarak atanır:

$$\hat{y} = \begin{cases} 1 & \text{eğer } P(y = 1|x) \geq 0.5 \\ 0 & \text{eğer } P(y = 1|x) < 0.5 \end{cases} \quad (33)$$

Model Performans Değerlendirme

Bu çalışmada, kullanılan makine öğrenmesi ve derin öğrenme modellerinin sınıflandırma performansı, Doğruluk (Accuracy), Hassasiyet (Precision), Recall (Duyarlılık) ve F1 Skoru (F1-Score) metrikleri kullanılarak değerlendirilmiştir. Bu metrikler, modelin doğruluk, hassasiyet, duyarlılık ve genel performansını ölçmek için kullanılmıştır. Aşağıda, her bir hesaplanma yöntemleri sırasıyla Denklem 34-37'de gösterilmektedir. Bu denklemlerde, DP doğru pozitif, DN doğru negatif, YP yanlış pozitif ve YN yanlış negatif sayısını ifade etmektedir.

$$\text{Doğruluk} = \frac{DP+DN}{DP+DN+YP+YN} \quad (34)$$

$$\text{Hassasiyet} = \frac{DP}{DP+YP} \quad (35)$$

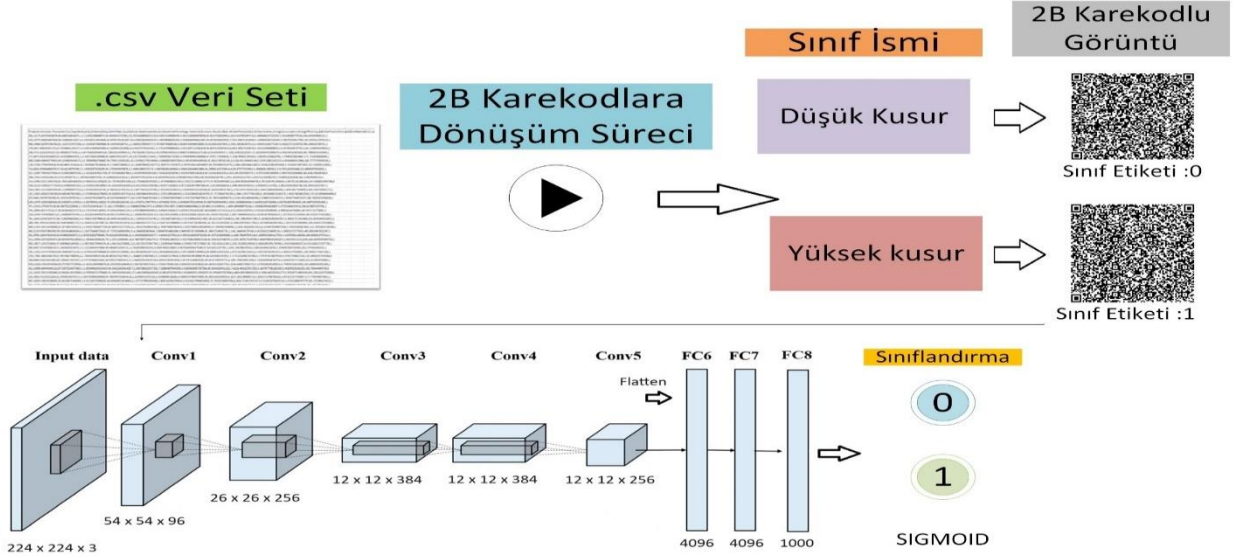
$$\text{Duyarlılık} = \frac{DP}{DP+YN} \quad (36)$$

$$\text{F1 Skoru} = 2 \cdot \frac{\text{Hassasiyet} \cdot \text{Duyarlılık}}{\text{Hassasiyet} + \text{Duyarlılık}} \quad (37)$$

Önerilen Metodoloji

Bu çalışmada, üretim süreçlerindeki kusur oranlarını doğru sınıflandırmak için AlexNet modeli kullanılarak yenilikçi bir metodoloji önerilmiştir. Metodoloji, veri dönüşümü, model eğitimi ve performans değerlendirmesi adımlarını içerir. Sayısal veriler Python kullanılarak QR kodlarına dönüştürülmüş, her veri noktası bir QR koduna çevrilip 224x224 piksel boyutunda RGB formatına getirilmiştir. Bu işlem, PIL ve qrcode kütüphaneleri ile gerçekleştirilmiştir. Ardından, QR kodları AlexNet modeli ile sınıflandırılmıştır. Bu yöntem, verilerin görsel temsillerini oluşturarak AlexNet'in güçlü desen tanıma yeteneklerinden faydalanmayı sağlar.

Şekil 2: Önerilen metodoloji



Önerilen metodolojide, sınıflandırma işlemi için kullanılan AlexNet modelinin katmanları ve parametreleri Tablo 2’de gösterilmektedir.

QR kod dönüşüm süreci, sayısal verilerin görsel formatta temsil edilmesini sağlayarak derin öğrenme algoritmalarının karmaşık desenleri daha etkili bir şekilde öğrenmesini mümkün kılmaktadır. Özellikle AlexNet gibi CNN modelleri, görsel veri girişine dayalı çalıştığı için bu yöntem, modelin performansını artırmakta ve daha yüksek doğruluk oranları elde edilmesini sağlamaktadır. Bu yaklaşım, verilerin doğrudan sayısal formda işlendiği geleneksel yöntemlere kıyasla daha zengin bir bilgi aktarımı sunmaktadır. QR kodlar, verilerdeki ilişkileri iki boyutlu bir görsel düzlemde organize ederek modelin öğrenme sürecine katkıda bulunmaktadır.

Tablo 2: AlexNet modelinin katmanları ve parametreleri

Katman (Type)	Çıkış Şekli (Output Shape)	Parametre Sayısı
conv2d_1 (Conv2D) Aktivasyon: ReLU	(None, 54, 54, 96)	34,944
max_pooling2d_4 (MaxPooling2D)	(None, 26, 26, 96)	0
conv2d_2 (Conv2D) Aktivasyon: ReLU	(None, 26, 26, 256)	614,656
max_pooling2d_5 (MaxPooling2D)	(None, 12, 12, 256)	0
conv2d_3 (Conv2D) Aktivasyon: ReLU	(None, 12, 12, 384)	885,120
conv2d_4 (Conv2D) Aktivasyon: ReLU	(None, 12, 12, 384)	1,327,488
conv2d_5 (Conv2D) Aktivasyon: ReLU	(None, 12, 12, 256)	884,992
max_pooling2d_6 (MaxPooling2D)	(None, 5, 5, 256)	0
flatten (Flatten)	(None, 6400)	0
dense_1 (Dense) Aktivasyon: ReLU	(None, 4096)	26,218,496
dropout_5 (Dropout)	(None, 4096)	0
dense_2 (Dense) Aktivasyon: ReLU	(None, 4096)	16,781,312
dropout_6 (Dropout)	(None, 4096)	0
dense_3 (Dense) Aktivasyon: ReLU	(None, 1000)	4,097
dropout_6 (Dropout)	(None, 1000)	0
dense_4 (Dense) Aktivasyon: sigmoid	(None, 1)	1,001

ÇALIŞMANIN BULGULARI

Bu çalışmada, üretim süreçlerindeki kusur oranlarını sınıflandırmak için hem makine öğrenmesi modellerinin hem de derin öğrenme modeli olan AlexNet modelinin hiperparametre ayarları Random Search hiperparametre algoritması ile belirlenmiştir. Çalışmada kullanılan her modelin bu yöntemle belirlenen hiperparametre ayarlamaları Tablo 3'te gösterilmektedir.

Tablo 3: Tüm modellerin random search algoritması kullanılarak belirlenen hiperparametre ayarları

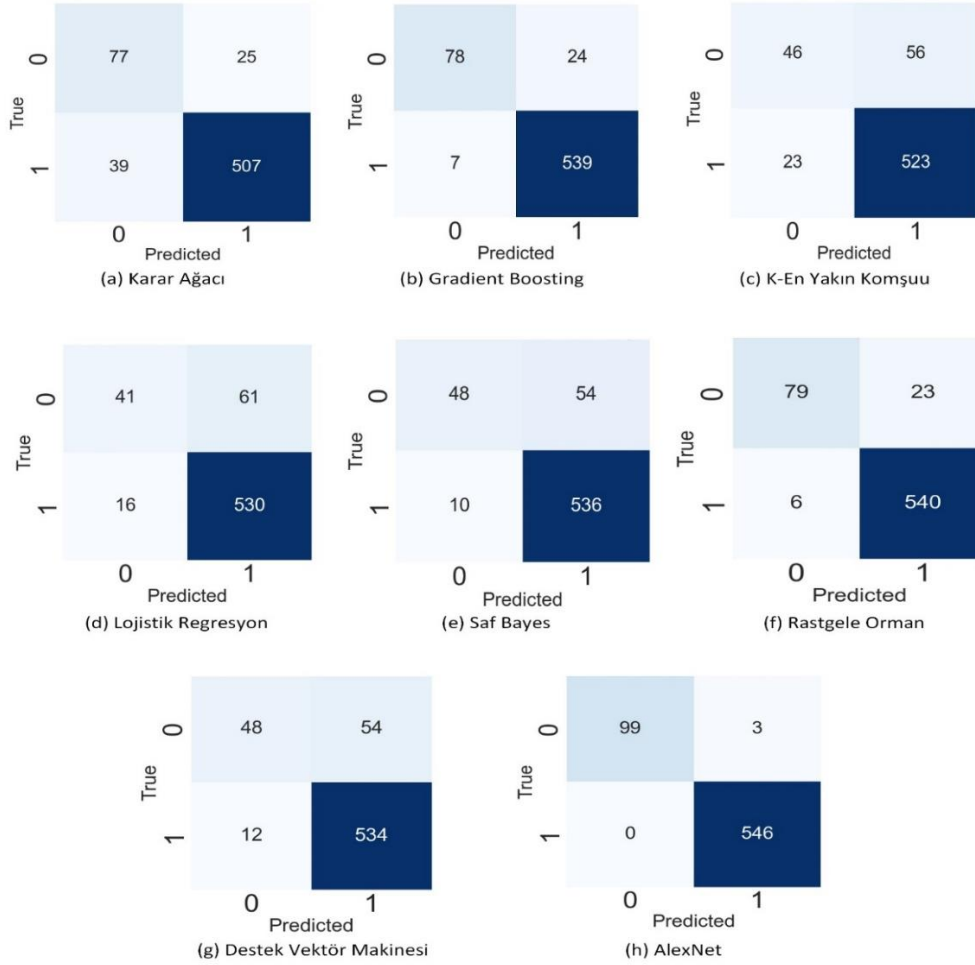
Model	Hiperparametre	Aranan Değer	Nihai Değer
Lojistik Regresyon	Düzenleme Parametresi	0.01, 0.1, 1, 10, 100	1
	Çözümleyici	'newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'	'lbfgs'
	Maksimum İterasyon	100, 200, 500	200
Rastgele Orman	Ağaç Sayısı	100, 200, 500	200
	Maksimum Derinlik	None, 10, 20, 30, 40, 50	20
	Minimum Örnek Sayısı	2, 5, 10	2

	Minimum Yaprak Sayısı	1, 2, 4	1
	Maksimum Özellik Sayısı	'auto', 'sqrt', 'log2', None	'auto'
	Bootstrap	True, False	True
Gradient Boosting	Ağaç Sayısı	100, 200, 500	200
	Öğrenme Oranı	0.001, 0.01, 0.1, 0.2	0.1
	Maksimum Derinlik	3, 4, 5	4
	Alt Örnekleme	0.5, 0.7, 1.0	1.0
	Maksimum Özellik Sayısı	None, 'sqrt', 'log2'	'sqrt'
		C	0.1, 1, 10, 100
Destek Vektör Makinesi	Kernel	'linear', 'poly', 'rbf', 'sigmoid'	'poly'
	Gamma	'scale', 'auto'	'scale'
	Derece (poly kernel için)	2, 3, 4	3
K-En Yakın Komşu	K (Komşu Sayısı)	3, 5, 7, 9, 11	5
	Ağırlıklar	'uniform', 'distance'	'uniform'
	Mesafe Ölçütü	'euclidean', 'manhattan', 'chebyshev', 'minkowski'	'euclidean'
Karar Ağacı	Maksimum Derinlik	None, 10, 20, 30, 40, 50	30
	Minimum Örnek Sayısı	2, 5, 10	5
	Minimum Yaprak Sayısı	1, 2, 4	2
	Maksimum Özellik Sayısı	None, 'sqrt', 'log2'	None
Saf Bayes	Smoothing Parametresi	1e-9, 1e-8, 1e-7, 1e-6, 1e-5	1e-9
AlexNet	Öğrenme Oranı	0.01, 0.001, 0.0001	0.0001
	Epok Sayısı	10, 20, 30	20
	Batch Boyutu	32, 64, 128	64
	Optimizör	'SGD', 'Adam', 'RMSprop'	'Adam'
	Dropout Oranı	0.5, 0.7	0.5
	Ağırlık Çürüme	1e-4, 1e-5, 1e-6	1e-4
	Beta1	0.9	0.9
	Beta2	0.999	0.999
	Epsilon	1e-8, 1e-7	1e-8
		Son Katman Aktivasyonu	'softmax', 'sigmoid'

Modellerin hiperparametre ayarlamaları yapıp modeller eğitildikten sonra, modellerin performansını daha detaylı incelemek amacıyla, her model için karmaşıklık matrisleri oluşturulmuştur. Karmaşıklık

matrisleri, modellerin sınıflandırma başarısını doğru pozitif, doğru negatif, yanlış pozitif ve yanlış negatif sayıları üzerinden değerlendirme imkânı sağlamaktadır. Aşağıda Şekil 3'te, her modelin test sonuçlarına ilişkin karmaşıklık matrisleri sunulmaktadır.

Şekil 3: Modellerin test sonuçlarına ilişkin karmaşıklık matrisleri



Şekil 3'teki karmaşıklık matrislerine göre, modellerin test sonuçları özetle şöyledir: Karar Ağacı düşük kusurda 77 DP, 25 YN; yüksek kusurda 507 DN, 39 YP. Gradient Boosting düşük kusurda 78 DP, 24 YN; yüksek kusurda 539 DN, 7 YP. K-En Yakın Komşuu düşük kusurda 46 DP, 56 YN; yüksek kusurda 523 DN, 23 YP. Lojistik Regresyon düşük kusurda 41 DP, 61 YN; yüksek kusurda 530 DN, 16 YP. Naif Bayes düşük kusurda 48 DP, 54 YN; yüksek kusurda 536 DN, 10 YP. Rastgele Orman düşük kusurda 79 DP, 23 YN; yüksek kusurda 540 DN, 6 YP. Destek Vektör Makinesi düşük kusurda 48 DP, 54 YN; yüksek kusurda 534 DN, 12 YP. AlexNet düşük kusurda 99 DP, 3 YN; yüksek kusurda 546 DN, 0 YP ile en yüksek

doğruluğa sahiptir. Bu sonuçlar, AlexNet'in üstün performansını göstermektedir. Tablo 4'te sınıflandırma raporları sunulmaktadır.

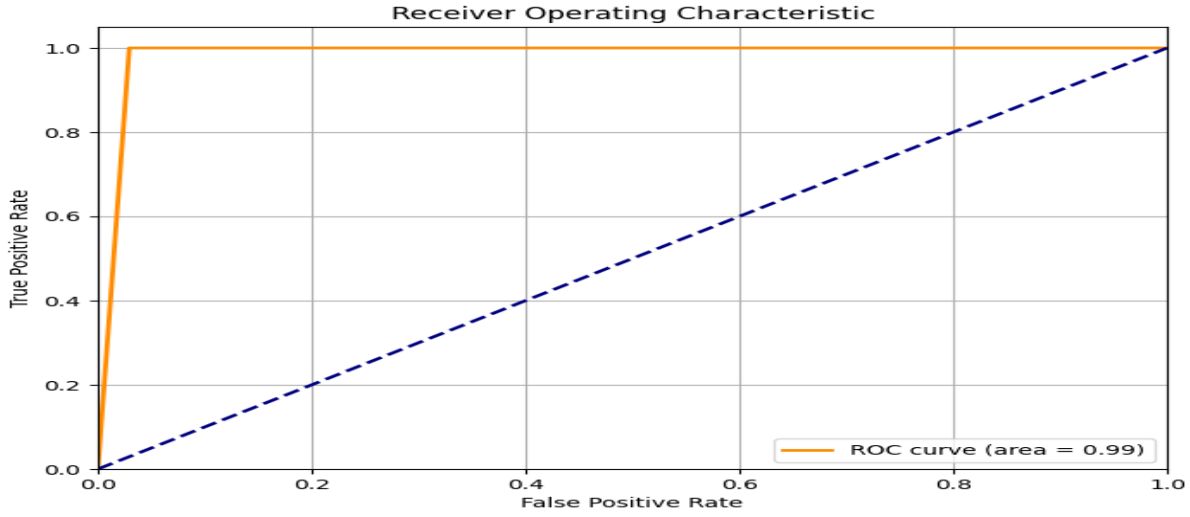
Tablo 4: Modellerin test sonuçlarına ilişkin sınıflandırma raporları

Model	Sınıf Etiketi	Hassasiyet	Duyarlılık	F1-skoru	Destek
Karar Ağacı	0	0,66	0,75	0,71	102
	1	0,95	0,93	0,94	546
	Doğruluk			0,90	648
	Makro Ortalama	0,81	0,84	0,82	648
	Ağırlıklı Ortalama	0,91	0,90	0,90	648
Gradient Boosting	0	0,92	0,76	0,83	102
	1	0,96	0,99	0,97	546
	Doğruluk			0,95	648
	Makro Ortalama	0,94	0,88	0,90	648
	Ağırlıklı Ortalama	0,95	0,95	0,95	648
K-En Yakın Komşu	0	0,67	0,45	0,54	102
	1	0,90	0,96	0,93	546
	Doğruluk			0,88	648
	Makro Ortalama	0,78	0,70	0,73	648
	Ağırlıklı Ortalama	0,87	0,88	0,87	648
Lojistik Regresyon	0	0,72	0,40	0,52	102
	1	0,90	0,97	0,93	546
	Doğruluk			0,88	648
	Makro Ortalama	0,81	0,69	0,72	648
	Ağırlıklı Ortalama	0,87	0,88	0,87	648
Saf Bayes	0	0,83	0,47	0,60	102
	1	0,91	0,98	0,94	546
	Doğruluk			0,90	648
	Makro Ortalama	0,87	0,73	0,77	648
	Ağırlıklı Ortalama	0,90	0,90	0,89	648

Rastgele Orman	0	0,93	0,77	0,84	102
	1	0,96	0,99	0,97	546
	Doğruluk			0,96	648
	Makro Ortalama	0,94	0,88	0,91	648
	Ağırlıklı Ortalama	0,95	0,96	0,95	648
Destek Vektör Makinesi	0	0,80	0,47	0,59	102
	1	0,91	0,98	0,94	546
	Doğruluk			0,90	648
	Makro Ortalama	0,85	0,72	0,77	648
	Ağırlıklı Ortalama	0,89	0,90	0,89	648
AlexNet	0	1,00	0,97	0,99	102
	1	0,99	1,00	1,00	546
	Doğruluk			1,00	648
	Makro Ortalama	1,00	0,99	0,99	648
	Ağırlıklı Ortalama	1,00	1,00	1,00	648

Tablo 4'te sunulan sonuçlar, üretim süreçlerindeki kusur oranlarını sınıflandırmada kullanılan modellerin performansını göstermektedir. AlexNet modeli, düşük kusur (0) sınıfında %100 hassasiyet, %97 duyarlılık ve %99 F1-skoru; yüksek kusur (1) sınıfında ise %99 hassasiyet, %100 duyarlılık ve %100 F1-skoru ile %100 genel doğruluk oranına ulaşmıştır. Bu sonuçlar, AlexNet'in diğer modellere kıyasla en yüksek performansı sergilediğini göstermektedir. Rastgele Orman ve Gradient Boosting modelleri, düşük kusur sınıfında sırasıyla %93 ve %92 hassasiyet, %77 ve %76 duyarlılık ile güçlü bir performans göstermiş, yüksek kusur sınıfında ise %96 üzerinde hassasiyet ve %99 duyarlılık değerlerine ulaşmıştır. Bu modellerin genel doğruluk oranları sırasıyla %96 ve %95'tir. Diğer yandan, Destek Vektör Makinesi ve Saf Bayes modelleri, düşük kusur sınıfında %80 ve %83 hassasiyet; %47 duyarlılık ile sınırlı bir başarı elde etmiş, ancak yüksek kusur sınıfında %91–%98 arasında hassasiyet ve duyarlılık değerleri ile tatmin edici sonuçlar sunmuştur. K-En Yakın Komşu ve Lojistik Regresyon, düşük kusur sınıfında sırasıyla %67 ve %72 hassasiyet; %45 ve %40 duyarlılık ile diğer modellere kıyasla daha düşük bir performans sergilemiş, ancak yüksek kusur sınıfında %90–%97 arası performans göstermiştir. Karar Ağacı, düşük kusur sınıfında %66 hassasiyet, %75 duyarlılık ile dengeli bir performans sergilerken, yüksek kusur sınıfında %95 hassasiyet ve %93 duyarlılık ile iyi sonuçlar almıştır. Bu bulgular, AlexNet'in QR kod dönüşüm süreci ile diğer modellere kıyasla daha üstün bir sınıflandırma kabiliyeti sunduğunu açıkça ortaya koymaktadır. Şekil 4'te AlexNet'in ROC eğrisi gösterilmiştir.

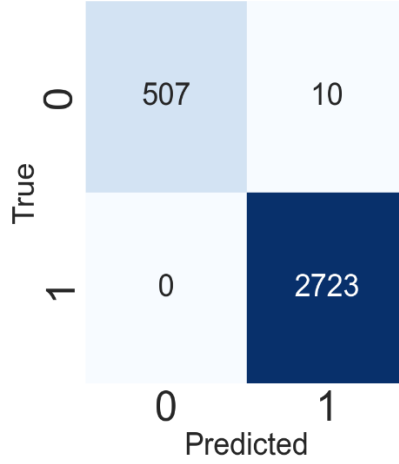
Şekil 4: AlexNet modelinin ROC eğrisi



Şekil 4'teki AlexNet modelinin ROC eğrisi, modelin sınıflandırma başarısını göstermektedir. ROC eğrisi, doğru pozitif oranı (TPR) ile yanlış pozitif oranı (FPR) arasındaki ilişkiyi grafikte sunar. Eğrinin altındaki alan (AUC) modelin genel performansını gösterir. AlexNet modelinin ROC eğrisi neredeyse mükemmel olup, AUC değeri 0.99'dur. Bu, modelin hem düşük hem de yüksek kusur sınıflarını yüksek doğrulukla ayırt edebildiğini ve sınıflandırma performansının mükemmel yakın olduğunu göstermektedir. AlexNet, yüksek doğruluk oranları ve düşük yanlış pozitif oranları ile üretim süreçlerindeki kusur tespitinde üstün başarı sergilemiştir. Bu bulgular, AlexNet'in karmaşık veri setlerinde etkili olduğunu ve kalite kontrolünü optimize etmede büyük potansiyele sahip olduğunu vurgulamaktadır.

Elde edilen bulgulara göre en başarılı model AlexNet olmuştur. AlexNet'e tüm veri seti üzerinde 5 katlı çapraz doğrulama uygulanmıştır. Bu yöntem, modelin genelleme yeteneğini değerlendirmek ve aşırı öğrenme (overfitting) sorunlarını önlemek için kullanılır. Veri seti beş eşit parçaya bölünerek her parça bir kez test, diğer dört parça eğitim için kullanılır. Böylece model, beş farklı eğitim ve test veri seti üzerinde eğitilip test edilir (Gemert vd., 2006). 5 katlı çapraz doğrulama, modelin performansını daha güvenilir ve genel geçer şekilde değerlendirmeyi sağlar. Bu yöntem, modelin farklı veri seti bölmelerindeki tutarlı performansını değerlendirmeye olanak tanır ve ortalama performans ölçütlerini sunar (Tran vd., 2023). Şekil 5'te, AlexNet modelinin 5 katlı çapraz doğrulama sonuçlarına ilişkin karmaşıklık matrisi gösterilmektedir.

Şekil 5: AlexNet modelinin 5 katlı çapraz doğrulama sonucuna ilişkin karmaşıklık matrisi



Şekil 5'te sunulan AlexNet modelinin 5 katlı çapraz doğrulama karmaşıklık matrisi, modelin düşük kusur (0) ve yüksek kusur (1) sınıflarını sınıflandırma başarısını göstermektedir. Model, düşük kusur sınıfında 507 doğru pozitif (DP) ve 10 yanlış negatif (YN), yüksek kusur sınıfında ise 2723 doğru negatif (DN) ve 0 yanlış pozitif (YP) tespit etmiştir. Bu sonuçlar, AlexNet modelinin yüksek kusur sınıfında hatasız ve düşük kusur sınıfında yüksek doğrulukla performans sergilediğini göstermektedir. Modelin genelleme yeteneği güçlü olup, farklı veri seti bölmelerinde tutarlı performans sergilemektedir. Bu bulgular, AlexNet'in üretim süreçlerinde kalite kontrol ve kusur tespiti için etkili bir araç olduğunu vurgulamaktadır. Tablo 5'te modelin çapraz doğrulama sonucuna ilişkin sınıflandırma raporu sunulmuştur.

Tablo 5: AlexNet modelinin çapraz doğrulama sonucuna ilişkin sınıflandırma raporu

Sınıf Etiketi	Hassasiyet	Duyarlılık	F1-skoru	Destek
0	1,00	0,98	0,99	517
1	1,00	1,00	1,00	2723
Doğruluk			1,00	3240
Makro Ortalama	1,00	0,99	0,99	3240
Ağırlıklı Ortalama	1,00	1,00	1,00	3240

Tablo 5'teki AlexNet modelinin 5 katlı çapraz doğrulama sınıflandırma raporu, modelin düşük kusur (0) ve yüksek kusur (1) sınıflarındaki performansını göstermektedir. Düşük kusur sınıfında hassasiyet %100, duyarlılık %98 ve F1-skoru %99 olarak kaydedilmiştir. Yüksek kusur sınıfında ise hassasiyet, duyarlılık ve F1-skoru değerlerinin tamamı %100'dür. Modelin genel doğruluğu %100 olup, makro ortalama hassasiyet %100, duyarlılık %99 ve F1-skoru %99, ağırlıklı ortalama metriklerinin tamamı %100'dür. Bu sonuçlar,

AlexNet modelinin hem düşük hem de yüksek kusur sınıflarında üstün performans sergilediğini ve yüksek kusurları hatasız tespit ettiğini göstermektedir. Modelin %100 genel doğruluğu, yüksek doğruluk ve güçlü genelleme yeteneğini ortaya koymaktadır. Bu bulgular, AlexNet'in üretim süreçlerinde kalite kontrol ve kusur tespiti için etkili ve güvenilir bir araç olduğunu vurgulamaktadır.

AlexNet modeliyle elde edilen %100 doğruluk oranının temel nedenleri, veri setinin boyutu ve yapısına dayanmaktadır. Veri setindeki kusurların belirgin ve tutarlı olması, modelin desen tanıma yeteneğini artırmış ve daha kesin sonuçlar üretmesine olanak tanımıştır. Ayrıca, denetimli öğrenme sürecinde kullanılan optimize edilmiş hiperparametreler, modelin yüksek doğruluk oranına katkıda bulunmuştur. Ancak bu sonuç, veri setinin spesifik yapısından kaynaklanıyor olabilir ve genellenabilirliği sınırlı olabilir.

SONUÇ VE ÖNERİLER

Üretim süreçlerindeki kusur oranlarının minimize edilmesi ve kalite kontrolünün optimize edilmesi, endüstriyel verimlilik ve rekabet avantajı açısından büyük önem taşımaktadır. Gelişen teknolojiler ve artan veri miktarı ile birlikte, üretim süreçlerinde toplanan verilerin etkili bir şekilde analiz edilmesi ve bu verilerden anlamlı sonuçlar çıkarılması kritik hale gelmiştir. Bu çalışma, makine öğrenmesi ve derin öğrenme algoritmalarının üretim süreçlerindeki kusur oranlarını yüksek doğrulukla sınıflandırma potansiyelini incelemektedir. Özellikle sayısal verilerin karekod formatında dönüştürülerek AlexNet gibi derin öğrenme modelleri ile analiz edilmesi, kalite kontrol süreçlerinin iyileştirilmesi ve üretim verimliliğinin artırılması açısından önemli bir yenilik sunmaktadır. Bu yenilikçi metodoloji, diğer endüstriyel süreçlerde ve farklı veri setlerinde de geniş bir kullanım potansiyeline sahiptir ve üretim sektöründe kalite kontrol süreçlerinin iyileştirilmesi ve operasyonel verimliliğin artırılması açısından önemli katkılar sunmaktadır. Çalışmanın bulguları, kullanılan modeller arasında AlexNet'in, en yüksek performansı sergileyerek, hem düşük kusur (0) hem de yüksek kusur (1) sınıflarını yüksek doğrulukla sınıflandırdığını göstermiştir. AlexNet modelinin 5 katlı çapraz doğrulama sonuçları, modelin genelleme yeteneğinin güçlü olduğunu ve özellikle yüksek kusur sınıflarını hatasız bir şekilde tespit ettiğini göstermiştir. Çapraz doğrulama sonuçlarına göre, modelin doğruluk oranı %100 olup, makro ve ağırlıklı ortalama metriklerinde de üstün bir başarı sergilemiştir.

Bu çalışmada kullanılan AlexNet modeli, üretim süreçlerindeki kusur oranlarını sınıflandırmada diğer makine öğrenmesi modellerine göre üstün performans sergilemiştir. Literatürdeki benzer çalışmaların bulguları ile bu çalışmanın bulguları karşılaştırıldığında, Ademuji vd. (2017), üretim sistemlerindeki hata teşhis uygulamalarında makine öğrenmesi algoritmalarını incelemiştir. Bayesian Ağları (BN), Yapay Sinir

Ağları (ANN), Destek Vektör Makineleri (SVM) ve Gizli Markov Modelleri (HMM) kullanarak yüksek hata sınıflandırma oranları elde etmişlerdir. Örneğin, Demetgül (2013)'ün çalışmasında SVM ve karar ağaçları algoritmaları, üretim sistemlerinde %100 sınıflandırma oranı elde etmiştir. Bu çalışmada da benzer şekilde AlexNet modeli, doğruluk oranı %100 olarak en yüksek performansı göstermiştir. Bu sonuç, derin öğrenme algoritmalarının karmaşık veri setlerinde üstün performans sergileyebileceğini göstermektedir. Bununla birlikte, Lee vd. (2017), yarı iletken üretim süreçlerinde arıza tespiti ve sınıflandırılması için FDC-CNN modelini kullanmış ve %97.9 genel doğruluk oranı elde etmişlerdir. Bu çalışma, derin öğrenme modellerinin üretim süreçlerinde başarılı olduğunu göstermektedir. Ancak, AlexNet modelinin %100 doğruluk oranı, FDC-CNN modelinden daha yüksek bir performans sergilemiştir. Bu durum, AlexNet modelinin üretim süreçlerindeki kusur tespitinde daha etkili olduğunu göstermektedir. Öte yandan, Essid vd. (2018) çalışmasında, metal kutuların üretiminde oluşan kusurların otomatik tespiti ve sınıflandırılması amacıyla derin sinir ağları kullanılmıştır. Otoenkoder derin sinir ağı mimarisi ile %99.80 doğruluk oranı elde edilmiştir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, sayısal verilerin 2 boyutlu karekod görüntülerine dönüştürülerek derin öğrenme algoritmaları ile analiz edilmesinin başarısını vurgulamaktadır. Escobar ve Morales-Menendez (2018), yüksek uyumlu üretim ortamlarında kalite kontrolü için 11-düzenlemeli lojistik regresyon kullanarak %100 hata tespit oranına ulaşmıştır. Benzer şekilde, bu çalışmada AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. Her iki çalışmada da yüksek doğruluk oranları elde edilmesine rağmen, AlexNet modelinin derin öğrenme algoritmaları kullanılarak daha karmaşık veri setlerinde başarılı olması dikkat çekicidir. Kim vd. (2018), dengesiz veri kümeleriyle üretim kalitesi koşullarını sınıflandırmak için maliyet duyarlı karar ağacı topluluklarını kullanmış ve AdaC2 tabanlı yöntemle en iyi sonuçları elde etmişlerdir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, derin öğrenme algoritmalarının dengesiz veri kümeleriyle çalışmada da etkili olabileceğini göstermektedir. Yang vd. (2020), üretim süreçlerinde ürün kusurlarını tespit etmek için derin öğrenme yöntemlerinin kullanımıyla %99.00 doğruluk oranı elde etmişlerdir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, sayısal verilerin karekod formatında dönüştürülerek kullanılmasıyla derin öğrenme algoritmalarının güçlü desen tanıma yeteneklerinden faydalandığını göstermektedir. Kotsiopoulos vd. (2021), endüstriyel üretim süreçlerinde düşük kaliteli kaynakların tespiti için L1-regularize edilmiş Lojistik Regresyon kullanarak %99.80 doğruluk oranı elde etmişlerdir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, derin öğrenme algoritmalarının üretim süreçlerinde daha karmaşık ve çeşitli veri setlerinde başarılı olabileceğini göstermektedir. Chien vd. (2022), akıllı yarı iletken üretiminde arıza tespiti ve sınıflandırılması için derin öğrenme teknikleri kullanarak %98.3 doğruluk oranı elde etmişlerdir. Bu çalışmada ise AlexNet modeli ile

%100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, derin öğrenme algoritmalarının farklı endüstriyel uygulamalarda etkili bir şekilde kullanılabileceğini göstermektedir. Blondheim (2022), elektronik bileşenlerin, boruların, kaynak parçalarının ve tekstil malzemelerinin yaygın kusurlarını sınıflandırmak için CNN ve diğer derin öğrenme yöntemleri kullanarak yüksek doğruluk oranları elde etmiştir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, derin öğrenme algoritmalarının üretim süreçlerinde daha yaygın ve etkili bir şekilde kullanılabileceğini göstermektedir. Singh ve Desai (2023), makine görüşü ve derin öğrenme algoritmaları kullanarak otomatik yüzey kusuru tespit sistemi geliştirmiş ve %100 doğruluk oranı elde etmişlerdir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. Her iki çalışmada da yüksek doğruluk oranları elde edilmesine rağmen, AlexNet modelinin sayısal verilerin karekod formatında dönüştürülerek kullanılmasıyla daha etkili olduğu görülmektedir. Herzog vd. (2024), lazer bazlı metal eklemeli imalatta hataların tespiti için süreç izleme ve makine öğrenmesi teknikleri kullanarak %95 doğruluk oranı elde etmişlerdir. Bu çalışmada ise AlexNet modeli ile %100 doğruluk oranı elde edilmiştir. AlexNet modelinin daha yüksek performans göstermesi, derin öğrenme algoritmalarının üretim süreçlerinde daha karmaşık ve çeşitli veri setlerinde başarılı olabileceğini göstermektedir. Sonuç olarak, literatürdeki mevcut çalışmalarla karşılaştırıldığında, bu çalışmada kullanılan AlexNet modelinin üretim süreçlerindeki kusur tespitinde üstün performans sergilediği ve sayısal verilerin 2 boyutlu karekod görüntülerine dönüştürülerek derin öğrenme algoritmaları ile analiz edilmesinin yenilikçi ve etkili bir yaklaşım olduğu görülmektedir. Bu bulgular, derin öğrenme algoritmalarının karmaşık veri setlerinde yüksek doğruluk sağlayarak üretim süreçlerindeki kalite kontrolünü optimize etme potansiyelini vurgulamaktadır.

Bu çalışmanın bazı kısıtlılıkları bulunmaktadır. Kullanılan veri seti belirli bir üretim sürecine ait olup, sadece 3240 veri girişi içermektedir. Daha geniş ve çeşitli veri setlerinin kullanılması, modelin genelleme yeteneğini ve diğer üretim süreçlerine uygulanabilirliğini artırabilir. Ayrıca, derin öğrenme modellerinin eğitimi ve kullanımı yüksek hesaplama gücü gerektirmektedir; bu da sınırlı kaynaklara sahip kuruluşlar için maliyetli olabilir. Çalışma sadece ikili sınıflandırma (düşük ve yüksek kusur) üzerine odaklanmıştır; daha karmaşık çok sınıflı sınıflandırma problemleri için modelin performansı test edilmemiştir. Sayısal verilerin karekod formatına dönüştürülmesi, veri ön işleme sürecini karmaşıklaştırabilir ve ek hesaplama maliyeti getirebilir. Bu dönüşüm sürecinin farklı veri setleri ve uygulamalar için geçerliliği sınırlı olabilir. Çalışmanın bulguları belirli bir üretim sürecine dayanmaktadır ve diğer endüstriyel süreçler için genelleme yapılmadan önce dikkatle değerlendirilmelidir. Gelecekte, daha geniş ve çeşitli veri setleri kullanılarak, farklı üretim süreçleri ve endüstrilerde model performansının test edilmesi önerilmektedir. Derin öğrenme modellerinin performansını artırmak için farklı model mimarileri ve hiperparametre ayarları denenebilir.

Ayrıca, transfer öğrenme gibi teknikler kullanılarak model performansı iyileştirilebilir. Gerçek dünya verileri, daha karmaşık ve heterojen yapıya sahip olabileceğinden, modelin genellenebilirliği gerçek dünyada test edilmeye ihtiyaç duymaktadır. Örneğin, çeşitli üretim hatlarındaki farklı kusur türleriyle modelin performansını karşılaştırmak, yöntemin uygulanabilirliğini değerlendirmek açısından önemlidir. Bu yöntem, daha geniş veri setleriyle test edildiğinde performans kaybı yaşayabilir. Ancak QR kod dönüşüm süreci, genellenebilirliği artırmak için yeni veri türlerine kolayca adapte edilebilir. Çalışmanın bulguları endüstriyel uygulamalara entegre edilerek, gerçek üretim ortamlarında test edilmelidir. Bu, modelin pratikteki etkinliğini değerlendirmek için önemlidir. Kullanıcı dostu yazılımlar ve araçlar geliştirilerek, derin öğrenme modellerinin endüstride yaygın olarak kullanımı sağlanabilir. Yöntem, üç boyutlu (3D) görüntüler gibi daha karmaşık veri türlerine uyarlanabilir. Bu durumda, 2D CNN modelleri yerine 3D CNN veya Transformer tabanlı modeller tercih edilebilir. Farklı kusur türlerinin modellenmesi, QR kod çözümlemesine ek olarak ses, ses veya sıcaklık verilerini de içerebilecek daha geniş bir sistem entegrasyonu ile sağlanabilir. Ayrıca, endüstri çalışanlarının derin öğrenme ve makine öğrenmesi konularında eğitilmesi, bu teknolojilerin benimsenmesini kolaylaştıracaktır. Sonuç olarak, bu çalışma, derin öğrenme modellerinin üretim süreçlerinde kusur tespiti ve kalite kontrolü için etkili bir araç olabileceğini göstermektedir. Gelecekteki çalışmalar, bu modellerin performansını artırarak, üretim süreçlerinde daha yaygın ve etkili kullanımlarını sağlayacaktır.

YAZAR BEYANI/AUTHOR STATEMENT

Araştırmacılar makaleye ortak olarak katkıda bulunduğunu bildirmiştir. Araştırmacılar herhangi bir çıkar çatışması bildirmemiştir.

Researchers have jointly contributed to the article. Researchers have not declared any conflict of interest.

KAYNAKÇA

Abdallah, M., Joung, B. G., Lee, W. J., Mousoulis, C., Raghunathan, N., Shakouri, A., & Bagchi, S. (2023). Anomaly detection and inter-sensor transfer learning on smart manufacturing datasets. *Sensors*, 23(1). doi: 10.3390/s23010486

Ademujimi, T. T., Brundage, M. P., & Prabhu, V. V. (2017). A review of current machine learning techniques used in manufacturing diagnosis. In M. F. Zaeh (Ed.), *Advances in Production Management Systems. The Path to Intelligent, Collaborative and Sustainable Manufacturing: IFIP WG 5.7 International Conference, APMS 2017, Hamburg, Germany, September 3-7, 2017, Proceedings, Part I* (pp. 407-415). Springer International Publishing. doi: 10.1007/978-3-319-66923-6_48

- Belnoue, J. P., Nixon-Pearson, O. J., Thompson, A. J., Ivanov, D., Potter, K. D., & Hallett, S. R. (2018). Consolidation-driven defect generation in thick composite parts. *Journal of Manufacturing Science and Engineering*, 140(7). doi: 10.1115/1.4039555
- Blondheim, D. Jr. (2022). Improving manufacturing applications of machine learning by understanding defect classification and the critical error threshold. *International Journal of Metalcasting*, 16(2), 502-520. doi: 10.1007/s40962-021-00637-0
- Borkovskaya, V., & Passmore, D. (2018). Application of failure mode and effects analysis in ecology in Russia. *MATEC Web of Conferences*, 193, 05026. doi: 10.1051/mateconf/201819305026
- Breitenbach, J., Eckert, I., Mahal, V., Baumgartl, H., & Buettner, R. (2022). Automated defect detection of screws in the manufacturing industry using convolutional neural networks. *Proceedings of the Annual Hawaii International Conference on System Sciences*. doi: 10.24251/hicss.2022.151
- Bullejos, M., Cabezas, D., Martín-Martín, M., & Alcalá, F. (2022). A k-nearest neighbors algorithm in Python for visualizing the 3D stratigraphic architecture of the Llobregat River Delta in NE Spain. *Journal of Marine Science and Engineering*, 10(7). doi: 10.3390/jmse10070986
- Burt, J. R., Torosdagli, N., Khosravan, N., RaviPrakash, H., Mortazi, A., Tissavirasingham, F., & Bağcı, U. (2018). Deep learning beyond cats and dogs: Recent advances in diagnosing breast cancer with deep neural networks. *The British Journal of Radiology*. doi: 10.1259/bjr.20170545
- Cao, Y., & Chun-hai, Z. (2012). Algorithm with weighted attributes for unresolved exception in decision tree induction algorithm. *2012 Second International Conference on Business Computing and Global Informatization*. doi: 10.1109/bcgin.2012.140
- Chen, Y., Peng, X., Kong, L., Dong, G., Remani, A., & Leach, R. (2021). Defect inspection technologies for additive manufacturing. *International Journal of Extreme Manufacturing*, 3(2). doi: 10.1088/2631-7990/abe0d0
- Chien, C. F., Hung, W. T., & Liao, E. T. Y. (2022). Redefining monitoring rules for intelligent fault detection and classification via CNN transfer learning for smart manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, 35(2), 158-165. doi: 10.1109/TSM.2022.3164904
- Demetgul, M. (2013). Fault diagnosis on production systems with support vector machine and decision trees algorithms. *The International Journal of Advanced Manufacturing Technology*, 67, 2183-2194. doi: 10.1007/s00170-012-4639-5

- Dudyrev, E., & Kuznetsov, S. O. (2021). Decision concept lattice vs. decision trees and random forests. In *Formal Concept Analysis: 16th International Conference, ICFCA 2021, Strasbourg, France, June 29–July 2, 2021, Proceedings 16* (pp. 252-260). Springer International Publishing. doi: 10.48550/arxiv.2106.00387
- Escobar, C. A., & Morales-Menendez, R. (2018). Machine learning techniques for quality control in high conformance manufacturing environment. *Advances in Mechanical Engineering*, 10(2), 1687814018755519. doi: 10.1177/1687814018755519
- Essid, O., Laga, H., & Samir, C. (2018). Automatic detection and classification of manufacturing defects in metal boxes using deep neural networks. *PLOS ONE*, 13(11), e0203192. doi: 10.1371/journal.pone.0203192
- Gemert, J., Snoek, C., Veenman, C., & Smeulders, A. (2006). The influence of cross-validation on video classification performance. *Proceedings of the ACM International Conference on Multimedia*. doi: 10.1145/1180639.1180786
- Goel, E., & Abhilasha, E. (2017). Random forest: A review. *International Journal of Advanced Research in Computer Science and Software Engineering*, 7(1), 251-257. doi: 10.23956/ijarcsse/v7i1/01113
- Herzog, T., Brandt, M., Trinchi, A., Sola, A., & Molotnikov, A. (2024). Process monitoring and machine learning for defect detection in laser-based metal additive manufacturing. *Journal of Intelligent Manufacturing*, 35(4), 1407-1437. doi: 10.1007/s10845-023-02119-y
- Hidayaturohman, Q. A., Clarke, H. G., Taflan, G. Y., & Sancaktar, I. (2023). A comparative study of machine learning approaches to heart disease prediction: An empirical analysis. *Preprint version*. doi: 10.21203/rs.3.rs-3098962/v1
- Karthikeyan, T., & Thangaraju, P. (2013). Analysis of classification algorithms applied to hepatitis patients. *International Journal of Computer Applications*, 62(15), 25-30. doi: 10.5120/10157-5032
- Kharoua, R. E. (2024). Predicting manufacturing defects dataset. *Kaggle*. <https://www.kaggle.com/datasets/rabieelkharoua/predicting-manufacturing-defects-dataset> adresinden erişildi.
- Kim, A., Oh, K., Jung, J. Y., & Kim, B. (2018). Imbalanced classification of manufacturing quality conditions using cost-sensitive decision tree ensembles. *International Journal of Computer Integrated Manufacturing*, 31(8), 701-717. doi: 10.1080/0951192X.2017.1407447
- Kotsiopoulos, T., Sarigiannidis, P., Ioannidis, D., & Tzovaras, D. (2021). Machine learning and deep learning in smart manufacturing: The smart grid paradigm. *Computer Science Review*, 40. doi: 10.1016/j.cosrev.2020.100341

- Lee, K. B., Cheon, S., & Kim, C. O. (2017). A convolutional neural network for fault classification and diagnosis in semiconductor manufacturing processes. *IEEE Transactions on Semiconductor Manufacturing*, 30(2), 135-142. doi: 10.1109/TSM.2017.2676245
- Liu, J., Cui, J., & Chen, C. (2023). Online efficient secure logistic regression based on function secret sharing. *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. doi: 10.1145/3583780.3614998
- Lv, M., Zhou, G., He, M., Chen, A., Zhang, W., & Hu, Y. (2020). Maize leaf disease identification based on feature enhancement and DMS-robust AlexNet. *IEEE Access*, 8, 57952-57966. doi: 10.1109/access.2020.2982443
- Mai, J., Chen, Z., Yi, C., & Ding, Z. (2021). Human activity recognition of exoskeleton robot with supervised learning techniques. *Preprint version*. doi: 10.21203/rs.3.rs-1161576/v1
- Natekin, A., & Knoll, A. (2013). Gradient boosting machines: A tutorial. *Frontiers in Neurorobotics*, 7. doi: 10.3389/fnbot.2013.00021
- Nuhu, A. A., Zeeshan, Q., Safaei, B., & Shahzad, M. A. (2023). Machine learning-based techniques for fault diagnosis in the semiconductor manufacturing process: A comparative study. *The Journal of Supercomputing*, 79(2), 2031-2081. doi: 10.1007/s11227-022-04730-x
- Oktaria, A. S., Prakasa, E., & Suhartono, E. (2019). Wood species identification using convolutional neural network (CNN) architectures on macroscopic images. *Journal of Information Technology and Computer Science*, 4(3), 274-283. doi: 10.25126/jitecs.201943155
- Olorunlambe, K. A., Hua, Z., Shepherd, D. E. T., & Dearn, K. D. (2021). Towards a diagnostic tool for diagnosing joint pathologies: Supervised learning of acoustic emission signals. *Sensors*, 21(23), 8091. doi: 10.3390/s21238091
- Papageorgiou, K., Theodosiou, T., Rapti, A., Papageorgiou, E. I., Dimitriou, N., Tzovaras, D., & Margetis, G. (2022). A systematic review on machine learning methods for root cause analysis towards zero-defect manufacturing. *Frontiers in Manufacturing Technology*, 2. doi: 10.3389/fmtec.2022.972712
- Papernot, N. (2018). Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *Preprint version*. doi: 10.48550/arxiv.1803.04765
- Rafdi, A., Mawengkang, H., & Efendi, S. (2021). Sentiment analysis using Naïve Bayes algorithm with feature selection particle swarm optimization (PSO) and genetic algorithm. *International Journal of Advances in Data and Information Systems*, 2(2). doi: 10.25008/ijadis.v2i2.1224

- Rizal, R. A., Purba, N. O., Siregar, L. A., Sinaga, K. P., & Azizah, N. (2020). Analysis of tuberculosis (TB) on X-ray images using SURF feature extraction and the k-nearest neighbor (KNN) classification method. *Journal of Applied Information and Communication Technologies*, 5(2). doi: 10.32497/jaict.v5i2.1979
- Ruiz, L., Torres, M., Gómez, A., Díaz, S., González, J. M., & Cavas, F. (2020). Detection and classification of aircraft fixation elements during manufacturing processes using a convolutional neural network. *Applied Sciences*, 10(19). doi: 10.3390/app10196856
- Safavian, S., & Landgrebe, D. A. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3), 660-674. doi: 10.1109/21.97458
- Singh, S. A., & Desai, K. A. (2023). Automated surface defect detection framework using machine vision and convolutional neural networks. *Journal of Intelligent Manufacturing*, 34(4), 1995-2011. doi: 10.1007/s10845-021-01878-w
- Singh, S. A., Kumar, A. S., & Desai, K. A. (2023). Comparative assessment of common pre-trained CNNs for vision-based surface defect detection of machined components. *Expert Systems with Applications*, 218. doi: 10.1016/j.eswa.2023.119623
- Tran, H., Friendship, R., & Poljak, Z. (2023). Classification of group A rotavirus VP7 and VP4 genotypes using random forest. *Frontiers in Genetics*, 14. doi: 10.3389/fgene.2023.1029185
- Wakayama, R., Murata, R., Kimura, A., Yamashita, T., Yamauchi, Y., & Fujiyoshi, H. (2015). Distributed forests for MapReduce-based machine learning. *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*. doi: 10.1109/acpr.2015.7486509
- Wang, L., Wang, X., & Li, X. (2007). Inference and learning in hybrid probabilistic network. *Frontiers of Computer Science in China*, 1(4), 429-435. doi: 10.1007/s11704-007-0041-0
- Wang, R., & Chen, N. (2020). Defect pattern recognition on wafers using convolutional neural networks. *Quality and Reliability Engineering International*, 36(4), 1245-1257. doi: 10.1002/qre.2627
- Wang, W., Li, T., & Tu, Z. (2019). Multiple fingerprints-based indoor localization via GBDT: Subspace and RSSI. *IEEE Access*, 7, 80519-80529. doi: 10.1109/access.2019.2922995
- Wen, L., Wang, W., & Huo, W. (2020). RegBoost: A gradient boosted multivariate regression algorithm. *International Journal of Crowd Science*, 4(1), 60-72. doi: 10.1108/ijcs-10-2019-0029
- Wermelinger, J. (2023). Opportunities and challenges of supervised machine learning for the classification of motor evoked potentials according to muscles. *BMC Medical Informatics and Decision Making*, 23(1). doi: 10.1186/s12911-023-02276-3



- Xue-jun, S., & Yang, G. (2006). Linear programming approach for the inverse problem of support vector machines. *Proceedings of the 2006 International Conference on Machine Learning and Cybernetics*. doi: 10.1109/icmlc.2006.258544
- Yang, J., Li, S., Wang, Z., Dong, H., Wang, J., & Tang, S. (2020). Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges. *Materials*, 13(24). doi: 10.3390/ma13245755
- Yoshida, K., Iwasawa, T., Sano, N., Tanaka, M., & Suzuki, T. (2017). Construction of defect detection system for image data using machine learning and image processing. *Total Quality Science*, 3(2), 46-58. doi: 10.17929/tqs.3.46