

---

*Araştırma Makalesi / Research Article*

---

## **Bul-Tak Oyunağı Şekillerinin Klasik Görüntü İşleme ve Derin Öğrenme Yöntemleri ile Tespiti**

Mehmet DERSUNELİ<sup>1\*</sup>, Taner GÜNDÜZ<sup>2</sup>, Yakup KUTLU<sup>3</sup>

<sup>1,2,3</sup>*Bilgisayar Mühendisliği Bölümü, İskenderun Teknik Üniversitesi, Hatay, Türkiye*  
(ORCID: [0000-0002-9689-2554](https://orcid.org/0000-0002-9689-2554)) (ORCID: [0000-0002-0361-5612](https://orcid.org/0000-0002-0361-5612)) (ORCID: [0000-0002-9853-2878](https://orcid.org/0000-0002-9853-2878))

---

### **Öz**

Bilgisayar görme algoritmaları, teknolojinin ilerlemesiyle daha kullanılabilir hale gelmektedir. Klasik yöntemler olan görüntü işleme ve makine öğrenmesi algoritmaları ile yapılan bilgisayarlı görü uygulamaları halen kullanılsa da gürültü veya istenmeyen ortam değişimleri etkisini sonuçlar üzerinde göstermektedir. Bu çalışmada, bir robotu hareket ettirmek için bilgisayar görme işleminin tepki hızının basit bir problemde nasıl cevap vereceği görülmek istenmiştir. Bu amaçla klasik makine öğrenme yöntemleri ve derin öğrenme algoritmalarından oluşan iki farklı yöntem ile 3 boyutlu geometrik şekiller içeren bul-tak oyunağı üzerinde nesne tespit işlemi gerçekleştirilmiştir. Klasik yöntemde iki farklı algortmada görüntü işleme ile elde edilen öznelikler k-NN algoritması ile sınıflandırılmış, derin öğrenme yönteminde ise nesne tespiti için özelleşmiş olan Yolov4 algoritması kullanılmıştır. Dene ortamında klasik görüntü işleme yöntemi siyah arka planlı test veri setinde %100 başarımlar sağlarken, farklı renk ve desende arka plana sahip ikinci test veri setinde başarımlar %86,25'e düşmüştür. Yolov4 derin öğrenme yöntemi algoritması ise her iki veri setinde de %100 başarımlara ulaşmıştır. Algoritmalar gerçek zamanlı kamera görüntüsü üzerinde çalıştırıldığında klasik yöntem siyah arka planlı bir kare görüntüde 0,06sn'de, farklı renk ve desende arka plana sahip bir kare görüntüde ise 0,04sn'de nesne tespiti yaparken, Yolov4 yöntemi 1,06sn'de nesne tespit işlemi gerçekleştirmiştir.

**Anahtar kelimeler:** Geometrik şekillerin sınıflandırılması, k-NN, Yolov4.

---

## **Detection of Find-Plug Toy Shapes with Classical Image Processing and Deep Learning Methods**

### **Abstract**

Computer vision algorithms are becoming more usable with the advancement of technology. Although computer vision applications with classical methods of image processing and machine learning algorithms are still used, noises or undesirable environment changes show their effects on the results. In this study, we wanted to see how the response speed of computer vision to move a robot would respond in a simple problem. For this purpose, object detection was carried out on the find-and-tack toy containing 3D geometric shapes with two different methods consisting of classical machine learning methods and deep learning algorithms. In the classical method, the features obtained by image processing in two different algorithms are classified with the k-NN algorithm, while the Yolov4 algorithm, which is specialized for object detection, is used in the deep learning method. In the experimental environment, the classical image processing method provided 100% performance in the test dataset with a black background, while the performance decreased to 86.25% in the second test dataset with a different color and patterned background. The Yolov4 deep learning method algorithm, on the other hand, achieved 100% success in both data sets. When the algorithms are run on the real-time camera image, the classical method detects objects in 0.06 sec on a square image with a black background, and 0.04 sec on a square image with a different color and patterned background, while the Yolov4 method takes 1.06 sec. object detection has been performed.

**Keywords:** Classification of geometric shapes, k-NN, Yolov4.

---

---

\*Sorumlu yazar: [mdersuneli@hotmail.com](mailto:mdersuneli@hotmail.com)

Geliş Tarihi: 25.04.2021, Kabul Tarihi: 05.11.2021

## 1. Giriş

Bilgisayarlı görü, insanların görme sistemi ile nesnelere anlamlandırması, bunları değerlendirerek kullanması yeteneğinin bilgisayar sistemlerine kazandırılması ile ilgilenen bir alandır. Bu yeteneği kazanan bilgisayarlı sistemlerin kullanılması, günümüzde çoğalan ve farklılaşan insan ihtiyaçlarına daha hızlı cevaplar verilmesini sağlamıştır.

Fabrikalarda veya lojistik tesislerinde bir yerden bir yere nesne taşıma işlevi gören robot kollar için bilgisayarlı görü sistemlerinden faydalanılmaktadır. Kameralardan alınan görüntülerdeki nesnelere farklı yöntemlerle tespit edilerek konumları bulunmakta ve robot kolla ilgili nesneyi bir yerden bir yere taşıması için ihtiyacı olan verileri sağlamaktadır[1, 2, 3].

Bu kapsamda görüntü işleme ve makine öğrenmesi yöntemleri kullanılarak yapılan bilgisayarlı görü çalışmalarının yanı sıra son dönemde derin öğrenme yöntemleri ile bilgisayarlı görü çalışmalarına yeni bir bakış gelmiştir. Literatürde öncelikle klasik görüntü işleme yöntemleri kullanılarak yapılan bilgisayarlı görü çalışmaları incelendiğinde; Bir çalışmada, kamera ile görüntüleri alınan kare, yuvarlak, dikdörtgen gibi şekillerdeki nesnelere; gri filtre, boyutlandırma, gürültü giderme, ortanca filtre ve ikili resme dönüştürme gibi görüntü işleme teknikleri ile tanımlanmıştır. Ardından bu nesnelere, bilgisayara bağlı olarak çalışan robotik kol ile belirlenen yerlere yerleştiren bir uygulama gerçekleştirilmiştir[4]. Başka bir çalışmada, kamera ile nesnelere açısından bağımsız olarak öz nitelikler çıkarmak için eşik belirleme, kenar belirleme ve bölge büyütme gibi görüntü işleme tekniklerinden yararlanılmış, yapay sinir ağlarında bu öz nitelikler kullanılarak nesnelere tanınmış ve nesnelere ilgili yerlere yerleştirilmesi için bir robot kol kullanılmıştır[5]. Kare, daire, dikdörtgen ve üçgen gibi şekillerin resimlerinin bulunduğu başka bir çalışmada, ikili resme dönüştürme, nesne sınırları çıkarma, nesne alanlarını bulma ve alan filtreleme gibi görüntü işleme teknikleri ile şekiller ayırt edilmiştir. Ardından bunların etrafına bir sınır kutusu çizilmiş ve bu kutu ile şeklin alanının oranları alınarak şeklin ne olduğuna karar veren bir uygulama geliştirilmiştir[6]. Bir çalışmada, kare, daire, üçgen, yıldız, çokgen gibi iki boyutlu şekilleri tespit etmek için bazı öz nitelikler önerilmiş ve bu öz nitelikler k-NN sınıflandırıcı kullanılarak %96,7 başarımla elde edilmiştir[7]. Kare, daire ve üçgen şekillerinin tespit edilmeye çalışıldığı başka bir çalışmada şekillerin çevre ve alan bilgisi ile kullanılarak bir öz nitelik verisi oluşturulmuş ve %85 sınıflandırma başarımla elde edilmiştir[8].

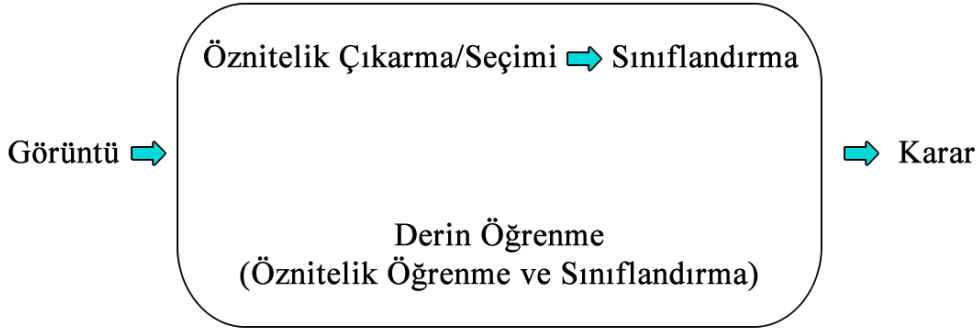
Bu kapsamda derin öğrenme yöntemleri kullanılarak yapılan bilgisayarlı görü çalışmaları incelendiğinde; Derin öğrenme yöntemleri ile dokunsal yüzey tespiti üzerine yapılan bir çalışmada, 4850 adet resimden oluşturulan veri seti Yolov2, Yolov3 ve Yolov3-Dense ile NVIDIA GeForce GTX1080 Ti 11GB ekran kartı üzerinde eğitilmiştir. Sonuç olarak ise başarımlar Yolov2:%69; Yolov3:%78; Yolov3-Dense:%89 şeklinde oluşmuştur[9]. Yapılan diğer bir çalışmada 4 farklı narenciye ürününden oluşturulan 1750 resimlik veri setinde nesne tespiti için Fast R-cnn, Yolov3, Yolov4 ve özelleştirilmiş Yolov4 algoritmaları kullanılmış, eğitimler Quadro P4000, NVIDIA 430.26 üzerinde yapılmış ve sonuç olarak doğruluk oranları Fast R-cnn:%86, Yolov3:%82, Yolov4:%92, Özelleştirilmiş Yolov4:%96 şeklinde oluşmuştur[10]. Başka bir çalışmada ise bulanık su altı görüntülerinden balık tespiti yapılması amaçlanmış bunun için 400 adet balık resminden oluşan bir veri seti kullanılmıştır. Eğitim için Yolov2, Yolov3, Yolov3- Tiny ve MobileNet-SSD Ağları kullanılmış sonuç başarımları ise Yolov2:%78,61, Yolov3-Tiny:%77,98, Yolov3:%71,95, MobileNet-SSD: %88,07 şeklinde gerçekleşmiştir[11]. Tıp alanında yapılan bir çalışmada ise efüzyon sitopatoloji görüntülerinde otomatik çekirdek algılama için Yolov3 derin öğrenme yöntemi kullanılmış sonuç olarak daha önce kullanılmış olan Fast R-Cnn, R-Cnn ve SSD ile kıyaslandığında FP (yanlış pozitif) sayısı artarken 0,060 (sec/img) ile en hızlı tespit zamanına ulaşılmıştır[12]. Bir diğer çalışma ise montaj parçalarının tespiti üzerine yapılmış, çalışmada Yolov3 algoritması 1 sınıf için 312 adet resim verisine uygulanmış, sistem OpenCV kütüphanesi ile gerçek zamanlı uygulamaya çevrilmiş, sonuçta başarımla oranı %84 olarak tespit edilmiştir[13].

Literatürdeki bu çalışmalar incelendiğinde, gerek görüntü işleme ve makine öğrenmesi yöntemleri kullanılarak yapılan bilgisayarlı görü çalışmalarının gerekse derin öğrenme yöntemleri ile gelen yeni bilgisayarlı görü çalışmalarının robotik çalışmaların vazgeçilmez araçları olduğu görülmektedir. Gerçek zamanlı çalışacak robotik uygulamalar için nesne tespiti ve sınıflandırma işleminde 3 boyutlu şekillerin kullanılması önem arz etmektedir. Bundan dolayı bu çalışmada, nesne tespit ve sınıflandırma işlemi için bul-tak adlı oyuncak içerisinde bulunan 3 boyutlu geometrik şekiller kullanılacaktır.

Bul-tak oyuncuğındaki 4 farklı geometrik şeklin tespitinde ilk yöntem olarak klasik görüntü işleme yöntemleri ile öznitelikler elde edilecek ve k-NN algoritması ile sınıflandırılarak nesne tespiti yapılacaktır. Kullanacağımız diğer yöntemde ise Yolov4 derin öğrenme algoritması ile nesne tespiti yapılacaktır. Farklı arka plan ve desenlerde nesne tespiti probleminin klasik görüntü işleme ve derin öğrenme yöntemlerindeki etkisi incelenerek bu yöntemlerde ortaya çıkan başarımların sonuçları karşılaştırılacaktır. Basit olan bu nesne tespiti problemimizde iki yöntemden hangisinin gerçek zamanlı bir uygulamada kullanımının uygun olacağı ortaya konulacaktır.

## 2. Materyal ve Metot

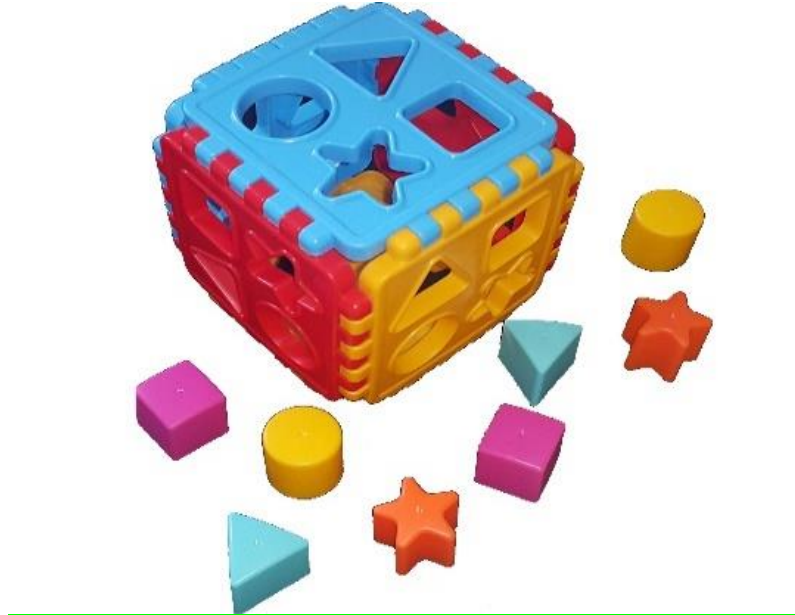
Bu çalışmada iki yaklaşım ayrı ayrı ele alınmıştır. Bu amaçla kameradan alınan görüntüler üzerinde önce klasik görüntü işleme algoritmaları kullanılarak yapılacak çalışmalar ve daha sonra derin öğrenme algoritmaları kullanılarak yapılacak çalışmalar anlatılmıştır. Şekil 1’de yapılacak bilgisayarlı görme algoritmasının genel yapısı verilmiştir.



Şekil 1. Bilgisayarlı görme algoritmasının genel yapısı

### 2.1. Verilerin Elde Edilmesi

Bu çalışmada, Şekil 2’de gösterilen bul-tak isimli 4 adet geometrik şekil (kare, daire, üçgen, yıldız) içeren çocuk oyuncuğı kullanılmıştır. Görüntü alma işlemi 1280 x 720 piksel çözünürlüğe sahip bir web kamerası üzerinden gerçekleştirilmiştir.



Şekil 2. Bul-tak oyuncuğı görseli

Bu çalışmada; arka planın model geliştirilmesine, başarıma ve nasıl çözümler üretileceğine etkisini görmek için iki farklı veri seti oluşturulmuştur. İlk veri seti sabit renkli arka plana sahip görsellerden oluşurken diğer veri seti ise farklı renk ve desene sahip arka planı olan görsellerden oluşmaktadır. Şekil 3 ve Şekil 4'te iki veri setine ait görsellerden örnekler gösterilmiştir.

Bu kapsamda, klasik görüntü işleme yöntemi ile yapılacak çalışma için eğitimde kullanılmak üzere; bu oyuncağın tüm parçalarının tek tek farklı konum ve rotasyonlarda siyah zemin üzerinde üstten görüntüleri alınmıştır. Her şekilden 18'er adet olmak üzere 72 adet görüntü elde edilmiştir. Ardından test için kullanılmak üzere tüm parçaların bulunduğu, yine farklı konum ve rotasyonlarda 80 adet şeklin bulunduğu görüntüler alınmıştır.



Şekil 3. Sabit renkli arka plana sahip veri setinden örnek görüntüler

Farklı renk ve desenli arka plana sahip veri setlerinde yapılacak çalışma için her bir şekilden 100'er adet olmak üzere toplam 400 adet görüntü alınmıştır. Her görselin içerisindeki şekiller etiketlendiği için bu görüntülerin arka planı farklı renk ve desenlerde olup rotasyondan ve ölçekten bağımsız olarak oluşturulmuştur. Veri seti %80 eğitim, %20 test verisi olarak ayrılmıştır.



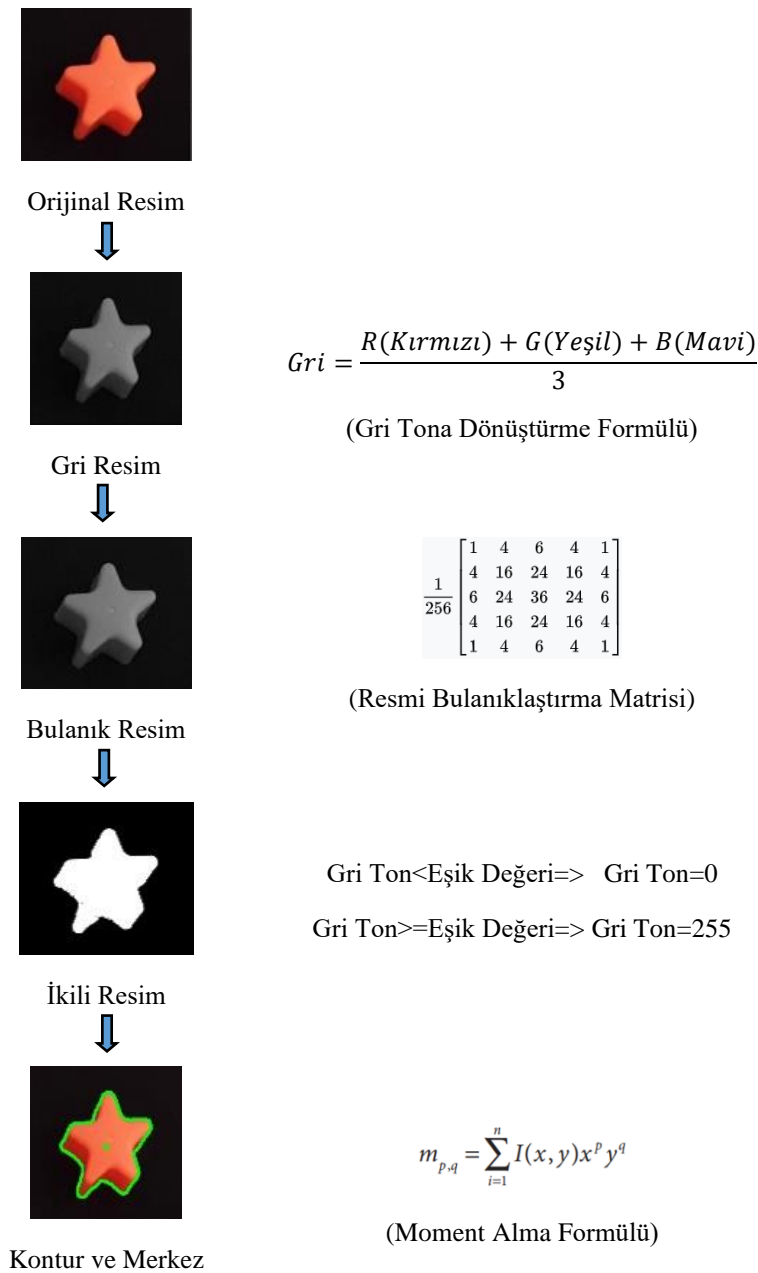
Şekil 4. Farklı renk ve desenli arka plana sahip veri setinden örnek görüntüler

## 2.2. Klasik Yöntemle Nesne Tespiti

Bu yöntemde ilk olarak görüntüler ön işlemlerden (gri tona dönüştürme, bulanıklaştırma, ikili resme dönüştürme ve kontur çıkarma) geçirilmiş, ardından farklı öznelik verileri elde edilmiş ve bu özneliklerin en uygun olanları seçilerek sınıflandırma yapılmıştır. Ardından farklı renk ve desenli arka plana sahip test görüntülerinde de başarımlarını gösterebilecek bir yöntem oluşturulmuştur.

### 2.2.1 Görüntü Ön İşleme

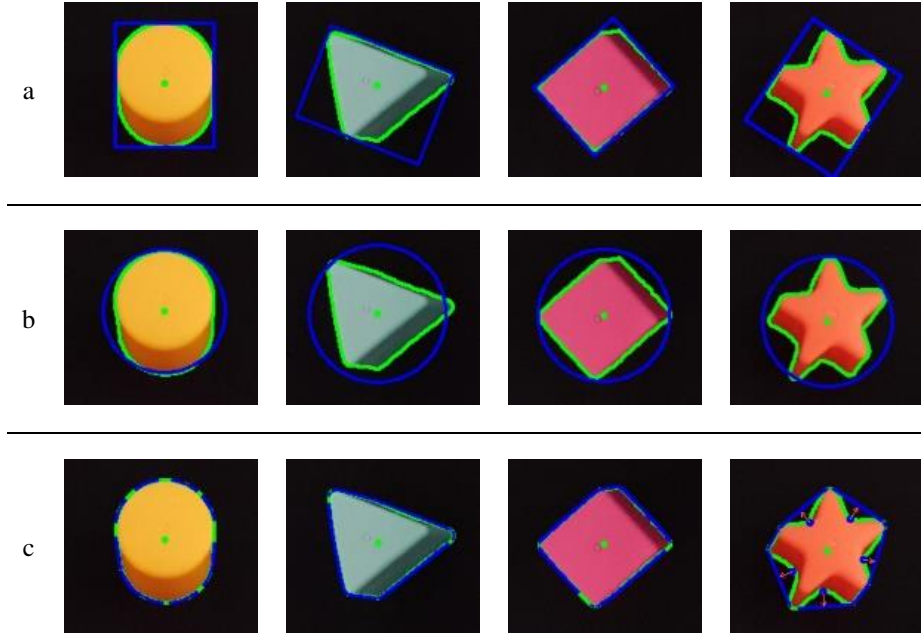
Bu aşamada siyah arka planlı görüntülere sırasıyla gri tona dönüştürme, bulanıklaştırma ve ikili resme dönüştürme işlemleri uygulanmaktadır. Bu işlemler ile arka plan yok edilerek şekillerin ortaya çıkması sağlanmaktadır. Ardından kontur bulma işlemi yapılarak belli bir değerden büyük alana sahip olan şekillerin dış hatlarına çizgi çizdirilmiştir. Şekillerin alanını hesaplamak için moment alma işlemi uygulanmıştır [14]. Görüntü ön işleme adımları Şekil 5’de gösterilmiştir.



Şekil 5. Sabit arka planlı görsellerde görüntü ön işleme adımları

### 2.2.2 Öznitelik Çıkarma

Görüntüler ön işlemlerden geçirildikten sonra elde edilen şekillerden rotasyondan ve uzaklıktan bağımsız olarak öznitelik verileri elde etmek için bazı işlemler yapılmıştır. İlk olarak rotasyonu şekil ile aynı olan bir sınır dikdörtgeni çizdirilmiş, bu dikdörtgenin alanının şeklin alanına oranı alınarak bir öznitelik verisi elde edilmiştir. Bu veriye ek olarak şekillerin çevresine minimum boyutlu sınır çemberi çizdirilerek bu çemberin alanının şeklin alanına oranlanmasıyla bir öznitelik verisi daha elde edilmiştir. Son olarak şekillere dışbükey çizgileri çizdirilerek şeklin gövdesinin bu çizgiye olan sapma miktarlarının ortalaması alınarak bir öznitelik verisi elde edilmiştir. Elde edilen bu öznitelikler Şekil 6'da gösterilmiştir.

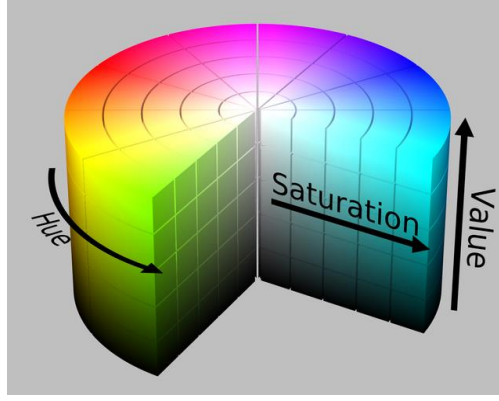


**Şekil 6.** Elde edilen özniteliklerin gösterimi: a. Döndürülmüş sınır dikdörtgeni çizilmiş şekiller, b. Minimum sınır çemberi çizilmiş şekiller, c. Dışbükey çizgileri çizilmiş şekiller

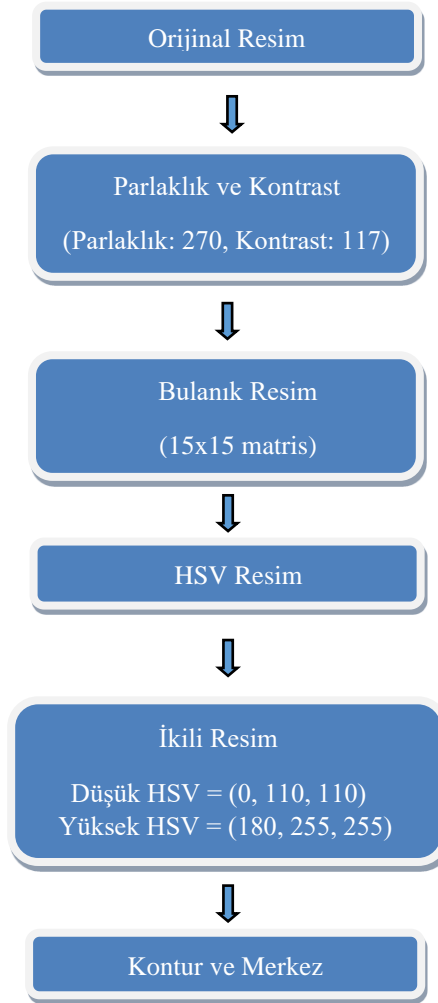
### 2.2.3 Farklı Renk ve Desenli Arka Plana Sahip Görüntülerde Kullanılan Yöntem

Farklı arka plan ve desenlerde nesne tespiti probleminin klasik görüntü işleme yöntemleriyle çözülebilmesi için sabit arka planda kullanılan yöntemden farklı bir yöntem ihtiyacı doğmuştur. Burada tespit edilmesi gereken şekillerin renkleri belirgin olduğu için görüntüler HSV(Hue, Saturation, Value) resme dönüştürülerek renk, parlaklık ve doygunluk parametreleri görüntü ön işleme aşamasında kullanılmıştır(HSV renk uzayı Şekil 7'de görülmektedir). Renkli şekillerin belirginleşmesi için öncelikle resimlerin parlaklık değeri artırılmış ve karşıtlık değeri ise azaltılmıştır. Sonrasında arka plan desenleri ve gürültülerinin azaltılması için 15x15 matris gezdirilerek bulanıklaştırma işlemi gerçekleştirilmiştir, ardından RGB resim HSV resme dönüştürülmüş, son olarak ikili resme çevrilerek kontur ve merkez bulma işlemi gerçekleştirilmiştir. Burada kullanılan görüntü ön işleme adımları Şekil 8'de gösterilmiştir.





Şekil 7. HSV(Renk, Parlaklık, Doygunluk) Renk Uzayı[15]



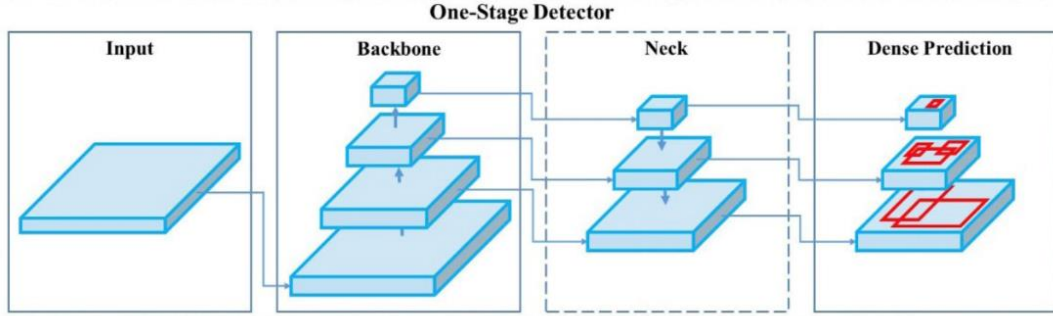
Şekil 8. Farklı renk ve desenlerde kullanılan görüntü ön işleme adımları

### 2.3 Derin Öğrenme Algoritması

Derin öğrenme, makine öğrenmesi yöntemlerinden biridir. Tüm makine öğrenmesi algoritmalarında olduğu gibi verilen girişler ile çıktıları tahmin edecek modelin eğitime olanak sağlamaktadır. Derin öğrenme yöntemleri kullanılarak yapılan bilgisayarlı görü çalışmaları incelendiğinde farklı yaklaşımlar önerilmiştir. Bu çalışmada Yolo algoritması tercih edilmiş ve uygulama sonuçları karşılaştırılmıştır.

Yolo algoritması nesne tespiti için özelleşmiş bir yapay sinir ağı tasarımıdır. Hızlı olması ve doğru tahmin başarımının yüksek olması tercih edilmesinin sebeplerindedir. 106 katmandan oluşan YOLOv4

algoritması terminolojisinde; Şekil 9’da gösterilen yapısı dâhilinde CNNs, residual blocks, skip connections, up-sampling, Leaky ReLU, IOU, non maximum suppression yöntemlerini barındırmaktadır. Öznitelik çıkarma katmanlarında önce resimler 32’nin katı olacak şekilde (416x416, 618x618 vb.) yeniden boyutlandırılıp daha sonra Feature Pyramid Network yöntemi ile öznitelik tespiti yapıp tahmini sınır kutuları çizdirilmektedir. Sınır kutuları içinde [tx ty tw th Po P1 P2...Pc] şeklinde nesnenin merkez noktası, en-boy uzunluğu ve sınıflara ait tahmin değerleri vardır. Sınıflandırma için K-Means sınıflandırıcı kullanılır. Güven endeksleri hesaplanarak en yüksek olanlar alınarak nesne tespiti tamamlanır [16].



Şekil 9. YOLOv4 algoritması görsel yapısı

(One Stage Detector: Tek aşamalı dedektör, Input: Giriş Katmanı, Backbone-Neck: Öznitelik Çıkarım Katmanları, Dense Prediction: Tahmin Katmanı)[16]

## 2.4 Sınıflandırma ve Performans Ölçümleri

Makine öğrenmesinde en çok kullanılan non-parametrik algoritmalarından biri olarak kabul edilen K en yakın komşu algoritması bu çalışmada klasik yöntemde sınıflandırıcı olarak tercih edilmiştir. K en yakın komşu algoritması iki nokta arasındaki uzaklık hesaplaması üzerinden oluşturulmuş basit bir yaklaşımdır. Genellikle Euclid uzaklık kullanılan bu algortmada farklı uzaklık ölçütü olan Minkowski, Manhattan gibi ölçüm yöntemleri de kullanılabilir [17].

Öklid Uzaklığı Hesaplama

$$\sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (1)$$

Sınıflandırma başarımları ölçütleri ise duyarlılık (Duy), belirlilik (Bel), F Puanı ve genel başarımları olarak belirlenmiş ve bu değerlerin hesaplanması; [18]

TP: Etiket (gerçek) değeri 1 ve tahmin sonucu 1 olanlardır, TN: Etiket (gerçek) değeri 0 ve tahmin sonucu 0 olanlardır, FP: Etiket (gerçek) değeri 0 ve tahmin sonucu 1 olanlardır, FN: Etiket (gerçek) değeri 1 ve tahmin sonucu 0 olanlardır.

$$Duy = \frac{\sum_{i=0}^{Farklı\ Sınıf\ Sayısı} Duy_i}{Farklı\ Sınıf\ Sayısı} \times 100\%, \quad Duy_i = \frac{TP_i}{TP_i + FN_i} \quad (2)$$

$$Bel = \frac{\sum_{i=0}^{Farklı\ Sınıf\ Sayısı} Bel_i}{Farklı\ Sınıf\ Sayısı} \times 100\%, \quad Bel_i = \frac{TN_i}{TN_i + FP_i} \quad (3)$$

$$F\ Puanı = 2 \times \frac{duy \times bel}{duy + bel} \quad (4)$$



$$\text{Genel Başarım} = \frac{\Sigma TP + \Sigma TN}{\text{Bütün Gözlemlerin Sayısı}} \times 100\% \quad (5)$$

### 3. Bulgular ve Tartışma

#### 3.1. Klasik Yöntem ile Alınan Sonuçlar

Klasik yöntemde sabit arka plana sahip görüntüler bazı ön işlemlerden geçirilerek öznelikler çıkartılmış ve başarımlar elde edilmiştir. Daha sonra aynı yöntem farklı renk ve desenli arka plana sahip görsellerde uygulandığında yetersiz kaldığı için yeni yaklaşımlar gerektiği görülmüştür. Bu yeni yöntemde ön işlemlere HSV renk uzayı parametreleri eklenerek başarımlar elde edilmiştir.

Sabit arka planlı veri setindeki görüntülerden elde edilen 3 adet öznelik verisinin farklı kombinasyonlarının k-NN algoritmasında k=7 seçilerek kullanılması ile siyah arka planlı test görüntülerinden elde edilen duyarlılık(Duy), belirlilik(Bel), F Puanı ve genel başarımın sonuçları Tablo 1 ve Tablo2'de gösterilmiştir. Bu çalışmada; dönmüş sınır dikdörtgeninin alanı ile şeklin alanının oranı, minimum sınır çemberinin alanı ile şeklin alanının oranı ve dışbükey çizgisi ile şeklin gövdesinin sapma miktarlarının ortalaması öznelik verileri ayrı ayrı kullanıldığında başarımlar sınırlı iken, birlikte kullanıldığında başarımları arttırdığı görülmektedir. İlk öznelik tek başına %82,5 genel başarıma sahip iken (Tablo 1'de verilmiştir), ikinci öznelik eklendiğinde bu başarımın %97'ye yükseldiği görülmüştür(Tablo 2'de verilmiştir). Üçüncü öznelik eklendiğinde bul-tak oyuncağındaki 4 adet geometrik şeklin tespitinde başarımlar %100'e yükselmiştir. Böylece kullanılacak öznelik verisinin yeterli olduğu kanısı oluşmuştur.

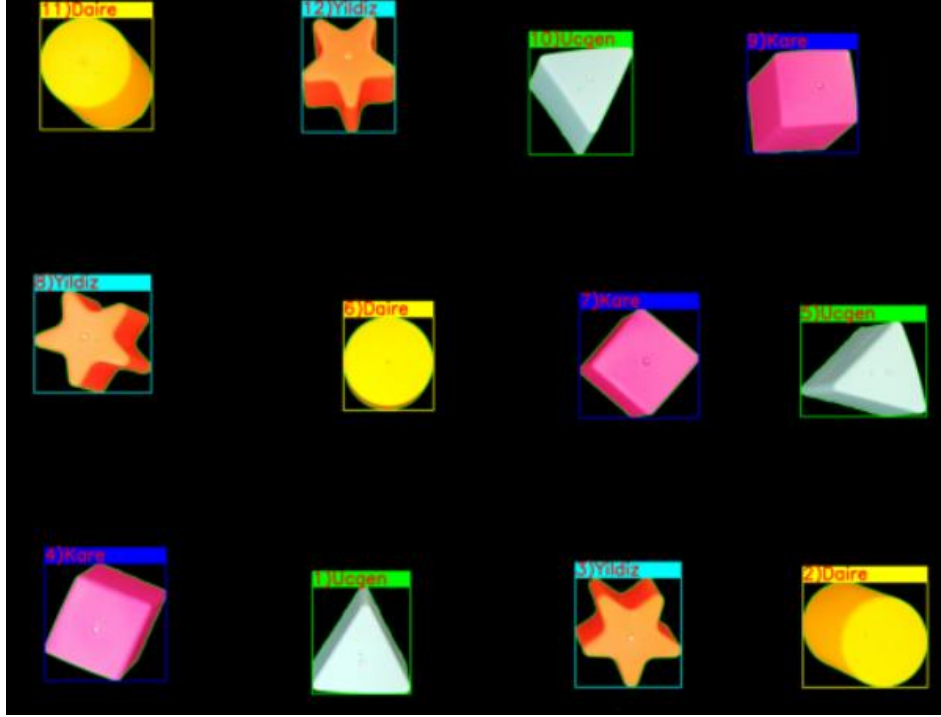
**Tablo 1.** Döndürülmüş sınır dikdörtgeni öznelik verileri kullanılarak elde edilen sonuçlar

Sınıf	Duy	Bel	F Puanı	Genel Başarım
Daire	%100	%86,95	%93,01	
Üçgen	%65	%76,47	%70,27	
Kare	%85	%100	%91,89	%82,5
Yıldız	%80	%69,56	%74,41	

**Tablo 2:** Döndürülmüş sınır dikdörtgeni ve Minimum sınır çemberi öznelik verileri bir arada kullanılarak elde edilen sonuçlar

Sınıf	Duy	Bel	F Puanı	Genel Başarım
Daire	%100	%100	%100	
Üçgen	%100	%90,90	%95,23	
Kare	%100	%100	%100	%97,5
Yıldız	%90	%100	%94,73	

Klasik görüntü işleme algoritması, python'da OpenCV Kütüphanesi [14] kullanılarak siyah arka planlı resimler ve gerçek zamanlı kamera görüntüleri üzerinde çalıştırılarak nesne tespiti başarılı bir şekilde gerçekleştirilmiş olup sonuç çıktısı Şekil 10'da gösterilmiştir.



Şekil 10. Klasik görüntü işleme ile yapılan siyah arka planlı görüntülerde nesne tespiti algoritmasının gerçek zamanlı sonuçları

Farklı renk ve desenlerde arka plana sahip olan test görüntülerinde klasik yöntemdeki ilk yaklaşım arka plandaki çeşitliliklerden dolayı ön işlemlerde yeterince iyi çalışmamış ve çok kötü başarımlar elde edilmiştir. Bu sebeple ön işlemlerin değiştirilmesi gerekliliği doğmuştur. Farklı renk ve desenli görseller için ön işlemlerde HSV renk uzayı parametreleri kullanılmış ve öz nitelikler çıkartılmıştır. Bu yeni yaklaşımla klasik yöntemin k-NN algoritmasında k=7 seçilerek kullanılması ile farklı renk ve desenlerde arka plana sahip görüntülerde %86,25 başarımlar elde edilmiştir (Tablo 3’de elde edilen duyarlılık (Duy), belirlilik (Bel), F Puanı ve genel başarımlar gösterilmiştir).

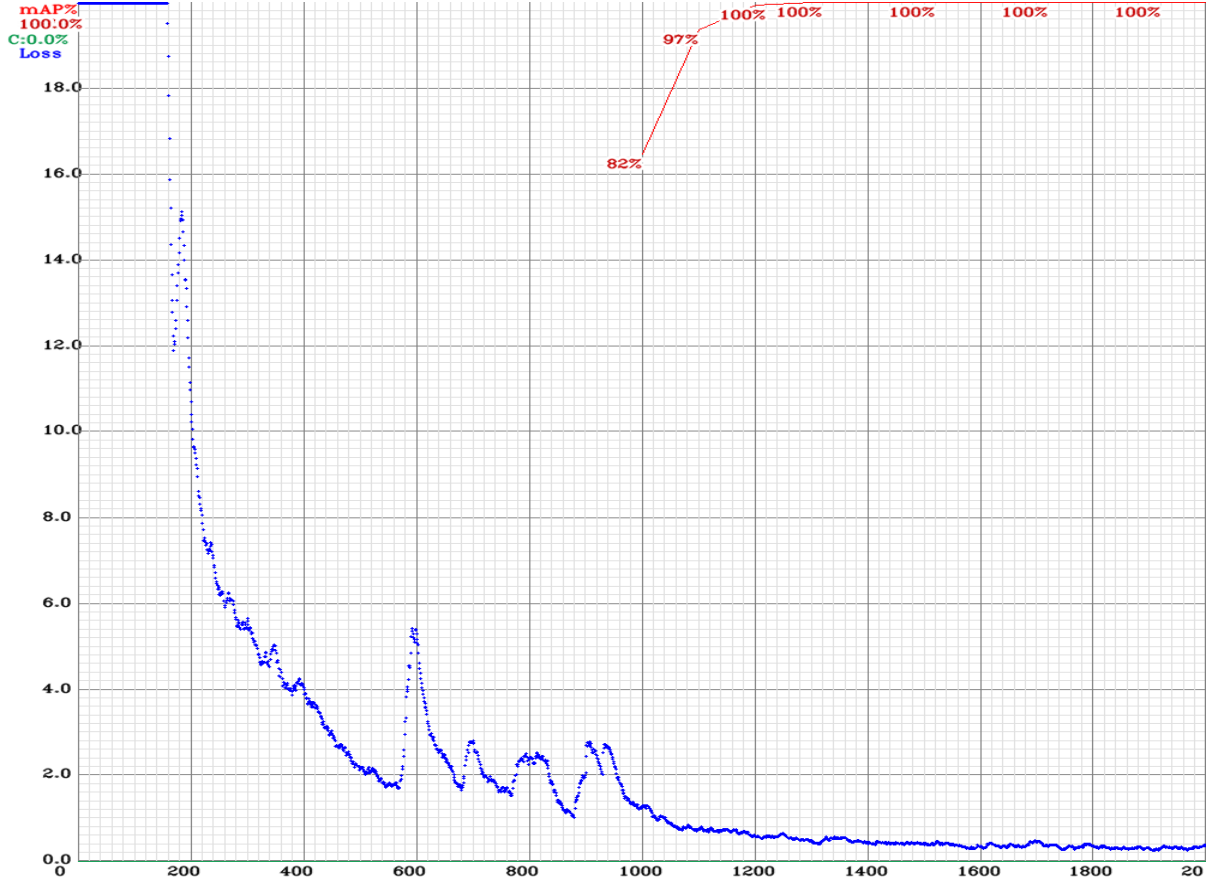
**Tablo 3:** Farklı renk ve desene sahip test görüntülerinde, HSV renk uzayı parametrelerinin kullanılması ile elde edilen sonuçlar

Sınıf	Duy	Bel	F Puanı	Genel Başarımlar
Daire	%90	%90	%90	
Üçgen	%80	%100	%88,88	%86,25
Kare	%75	%100	%85,71	
Yıldız	%100	%74,07	%85,10	

### 3.2. Yolo Algoritması Eğitim Sonuçları

Yolov4 algoritması veri setine göre revize edilmiştir. Bu kapsamda yeniden boyutlandırma işlemi için 416x416 olarak ayarlanmıştır. Filters parametresi  $3*(5+classes)$  ile 4 sınıf için 27 olarak belirlenmiştir. İterasyon sayısı 2000 olarak belirlenmiş ve her iterasyonda 60 resim alınarak her resim 60 ızgaraya bölünmüştür. Öğrenme oranı (learning rate) parametresi 0,0013 olarak belirlenmiştir. Veri seti ve YOLOv4 algoritması Google Drive üzerinden Google Colaboratory’ye aktarılmıştır. Burada Tesla K80 GPU üzerinden dosyalar Unix formatına çevrilerek eğitim gerçekleştirilmiştir.

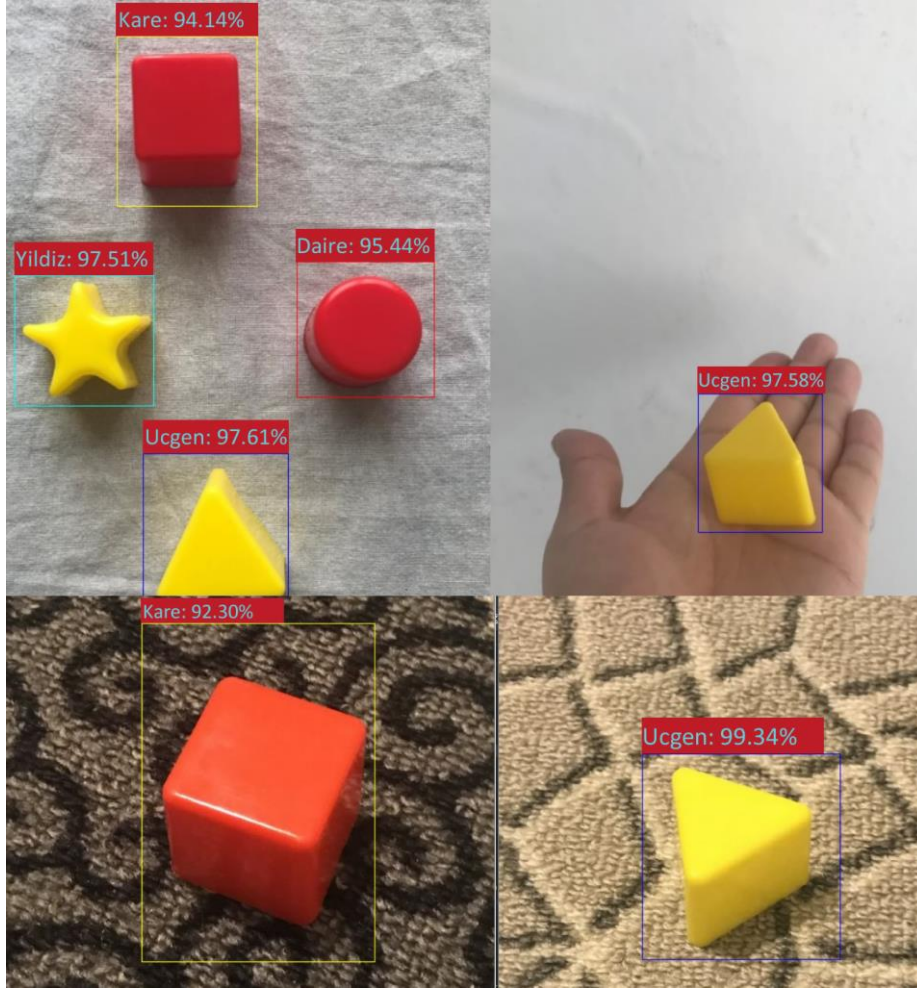
Bu çalışmada öncelikle veri seti makesense.ai uygulaması kullanılarak etiketlenmiştir ve Yolov4 formatına dönüştürülmüştür. Daha sonra gerek sabit arka planlı gerekse desenli veri setleri üzerinde YOLOv4 algoritması ile yapılan eğitim sonucunda kare, daire, üçgen ve yıldız şekilleri %100 doğrulukla sınıflandırılırken duyarlılık, belirlilik, F Puanı değerleri %100 olarak elde edilmiştir. Farklı arka plan desenlerine sahip görseller için Yolov4 derin öğrenme algoritmasının sağladığı öznelik öğrenme becerisi sayesinde desenlerdeki çeşitliliğin nesne tespiti için bir problem olmaktan çıktığı görülmüştür.



Şekil 11. Farklı renk ve desene sahip veri setinde Yolov4 başarımı (mAP: Genel Başarım, Loss: Kayıp Değeri)

Eğitim işlemi her iterasyonda rastgele seçilen 60 resimle yapılmıştır. Eğitim sonuç grafiği incelendiğinde iterasyon sayısı ilerledikçe hata oranının gitgide azaldığı, buna bağlı olarak 1000 iterasyondan itibaren hesaplanan ortalama hassasiyet değerinin %82 ile başlayıp eğitimler ilerledikçe %100'e ulaştığı Şekil 11'de görülmektedir.

Eğitilmiş dosya python'da OpenCV Kütüphanesi [14] ile oluşturulan nesne tanıma algoritmasında kullanılarak resimler ve gerçek zamanlı kamera görüntüleri üzerinden nesne tespiti başarılı bir şekilde gerçekleştirilmiş olup sonuç çıktısı Şekil 12'de gösterilmiştir.



Şekil 12. Yolov4 ile yapılan nesne tespiti algoritmasının gerçek zamanlı sonuçları

#### 4. Sonuç ve Öneriler

Bu çalışmada deneysel ortamda klasik görüntü işleme yöntemleri ile oluşturulan ilk algoritma, siyah arka planda rotasyon ve ölçekten bağımsız alınan resimlerde nesne tespitinde %100 başarımla sonuçlandı. Arka planı farklı renk ve desenlerde olup rotasyon ve ölçekten bağımsız alınan resimlerle oluşturulan ikinci veri setinde ilk algoritma başarımla sonuçlanamadı. Ön işlemlerde iyileştirmeler yapılarak yeniden ele alınmış ve oluşturulan algoritma, ikinci veri setinde nesne tespitinde %86,25 başarımla sonuçlandı.

Yolov4 derin öğrenme yöntemiyle oluşturulan algoritma, her iki veri setinde de %100 başarımla sonuçlandı. Mevcut sistemde klasik yöntemde görüntü işleme açısından yeni yaklaşımlar gerekli olduğu, aksi durumda nesnelerin bulunmasında sorunlar olduğu görülmüştür. Her ne kadar iyileştirmeler yapılsa da yeterli olmamış ve sabit arka planlı veri setindeki başarımla seviyesi ikinci veri setinde yakalanamamıştır. Derin öğrenme algoritmalarında ise arka plan değişikliklerinde algoritmik olarak herhangi bir değişiklik yapılmadığı ve sonuçların etkilenmediği görülmüştür.

Bu algoritmalar, 8gb DDR4 ram, NVIDIA 4gb ekran kartı, 2.4ghz işlemci hızına sahip bir bilgisayarda, 1280x768 çözünürlüğe sahip bir web kamera ile çalıştırılmıştır. Siyah arka planlı bir kare görüntü için klasik yöntem 0,06sn'de sonuca ulaşırken farklı renk ve desende arka plana sahip bir kare görüntü için ise 0,04sn'de nesne tespiti yapmıştır. Yolov4 derin öğrenme algoritmasıyla yapılan teste ise her iki veri setinde de 1,06sn'de nesne tespiti yapmıştır. Burada dikkat edildiği gibi basit problemlerde klasik yöntemler çok hızlı sonuç verse de ortam değişimlerinin başarımları etkilediği görülmektedir. Bunun için daha gelişmiş görüntü işleme algoritmaları kullanılmasının gerekli olduğu, bunun başarımları arttırabileceği gibi tespit süresini de arttırabileceği düşünülmektedir.

Sonuç olarak bu çalışmada çözülmek istenen problem basit olarak arka planı sabit tutulup çözülebileceği durumlarda klasik yöntemde oluşturulan ilk algoritmada kullanılarak hızlı bir şekilde başarılı sonuçlar elde edilmiştir. Problemin daha karmaşık olduğu, arka planların değişkenlik gösterebileceği durumlarda başarımleri açısından Yolov4 derin öğrenme algoritmasının tercih edilmesi daha iyi sonuçlar verecektir. Örneğin gezgin robot, insansız hava araçları gibi mobil olarak hareket halindeki robotik uygulamalarda ciddi bir problem olan bu konu, özellikle yeni geliştirilen GPU destekli gömülü kartlarla birlikte derin öğrenme algoritmalarının uygulanabilirliğinin gelişmesi ile çözüme kavuşacağı düşünülmektedir.

### **Yazarların Katkısı**

Çalışmada tüm yazarlar eşit oranda katkı sunmuştur.

### **Çıkar Çatışması Beyanı**

Yazarlar arasında herhangi bir çıkar çatışması bulunmamaktadır.

### **Araştırma ve Yayın Etiği Beyanı**

Yapılan çalışmada, araştırma ve yayın etiğine uyulmuştur.

### **Teşekkürler**

Bu çalışma İskenderun Teknik Üniversitesinin 2021TF-01 Numaralı Bilimsel Araştırma Projesi(BAP) desteği ile yapılmıştır.

### **Kaynaklar**

- [1] İşçimen B., Atasoy H., Kutlu Y., Yıldırım S., Yıldırım E. 2015. Smart robot arm motion using computer vision. *Elektronika Ir Elektrotechnika*, 21 (6): 3-7.
- [2] İşçimen B., Atasoy H., Kutlu Y., Yıldırım S., Yıldırım E. 2014. Bilgisayar Görmesi ve Gradyan İniş Algoritması Kullanılarak Robot Kol Uygulaması. *Akıllı Sistemlerde Yenilikler Ve Uygulamaları*, Mustafa Kemal University, İskenderun, Hatay, Turkey.
- [3] Kutlu Y., Alanoglu Z., Gökçen A., Yeniad M. 2021. Raspberry pi based intelligent robot that recognizes and places puzzle objects. *arXiv preprint arXiv:2101.12584*.
- [4] Ali H., Seng T. C., Hoi L. H., Elshaikh M. 2012. Development of Vision-Based Sensor of Smart Gripper for Industrial Applications. *IEEE 8th International Colloquium on Signal Processing and its Applications*, 23-25 March, Malacca, 300-304.
- [5] Erdoğan T. 2012. Hareketli Konveyör Üzerinde Kamera Görüntüsü ile Nesne Tanıma ve Nesnelere Yerine Koyma Uygulaması. *Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, İzmir*, 1-93.
- [6] Rege S., Memane R., Phatak M., Agarwal P. 2013. 2D geometric shape and color recognition using digital image processing. *International journal of advanced research in electrical, electronics and instrumentation engineering*, 2(6): 2479-2487.
- [7] Gupta S., Singh Y. J. 2017. Shape detection using geometrical features. *An International Journal of Engineering Sciences*, 26(1): 260-270.
- [8] Zakaria M. F., Choon H. S., Suandi S. A. 2012. Object shape recognition in image for machine vision application. *International Journal of Computer Theory and Engineering*, 4(1): 76-80.
- [9] Aktaş A., Doğan B., Demir Ö. 2020. Derin öğrenme yöntemleri ile dokunsal parke yüzeyi tespiti. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 35(3): 1685-1700.

- [10] Chen W., Lu S., Liu B., Li G., Qian T. 2020. Detecting citrus in orchard environment by using improved YOLOv4. Scientific Programming, 2020: 1-13.
- [11] Akgül T., Çalik N., Töreym B. U. 2020. Deep Learning-Based Fish Detection in Turbid Underwater Images. 28th Signal Processing and Communications Applications Conference (SIU), 5-7 Oct, Gaziantep, 1-4.
- [12] Kılıç B. 2020. Panorama ile Üretilen Plevral Efüzyon Sitopatoloji Görüntüleri Üzerinde Yolov3 ile Otomatik Çekirdek Algılama. Yüksek Lisans Tezi, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Trabzon, 1-50.
- [13] Çağıl G., Yıldırım B. 2020. Bir montaj parçasının derin öğrenme ve görüntü işleme ile tespiti. Zeki Sistemler Teori ve Uygulamaları Dergisi, 3(2): 31-37.
- [14] Bradski G., Kaehler A. 2008. Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly Media, Sebastopol, 1-580.
- [15] HSV color, 2010. HSV color solid cylinder saturation gray, <https://commons.wikimedia.org/w/index.php?curid=9801673>. (Erişim Tarihi: 25.11.2020).
- [16] Bochkovskiy A. 2020. Yolo v4, v3 and v2 for Windows and Linux. <https://github.com/AlexeyAB/darknet>. (Erişim Tarihi: 15.11.2020).
- [17] Cover T. M., Hart P. E. 1967. Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13 (1): 21-27.
- [18] Yayık A., Kutlu Y. 2012. Diagnosis of Congestive Heart Failure Using Poincare Map Plot. 20th Signal Processing and Communications Applications Conference (SIU), 18 - 20 April, Muğla, 1-4.