

YOLOv8 ve YOLO-NAS ile Kapalı ve Açık Alanlardaki Kişi Sayımının Mamdani Metodu yardımıyla Hesaplanması

Deniz Karhan

Kocaeli Üniversitesi
Bilgisayar Mühendisliği
200202005@koaceli.edu.tr

Abstract—

Kişi sayımı, birçok alanda önemli bir uygulama alanıdır. Örneğin, kalabalık yönetimi, güvenlik ve trafik denetimi gibi alanlarda kişi sayımı yapmak gereklidir. Kişi sayımı, geleneksel yöntemlerle manuel olarak yapılabilir. Ancak, bu yöntem zaman alıcı ve hata yapmaya açıktır.

Son yıllarda, derin öğrenme algoritmaları, kişi sayımı gibi birçok görüntü işleme probleminde başarılı sonuçlar elde etmiştir. YoloV8 ve YoloNAS, derin öğrenme tabanlı kişi sayımı algoritmalarının en başarılı örnekleridir.

Önerilen yöntem, YoloV8 ve YoloNAS algoritmaları yardımıyla ve Mamdani metodunun kullanılmasıyla bir alandaki kişi sayısını hesaplar. Yöntem, hazır veri seti üzerinde eğitilmiş 4 farklı model ve YoloV8 ile YoloNAS'ın olduğu 2 farklı model ile performansı ölçülmüştür.

Deneyler, Roboflow üzerindeki 3 farklı veri seti üzerinde yapılmıştır. Veri setleri, farklı koşullarda çekilmiş çeşitli insan görüntülerini içerir, ön işleme adımlarından geçirilmişlerdir ve insan konumları etiketlidir.

I. GİRİŞ

Bu çalışmada, YoloV8 ve YoloNAS algoritmalarını kullanarak bir alandaki kişi sayılarının hesaplanmasına yönelik bir yöntem önerilmektedir. Yöntem, hazır veri seti üzerinde eğitilmiş 4 farklı model ve YoloV8 ile YoloNAS'ın olduğu 2 farklı model ile performansı ölçülmüştür.

Kişi Sayımı Uygulama Alanları

Kişi sayımı, birçok alanda önemli bir uygulama alanına sahiptir. Bu uygulamalardan bazıları şunlardır:

Kalabalık yönetimi: Kalabalık yönetimi, insanların güvenli ve rahat bir şekilde hareket etmelerini sağlamak için gereklidir. Kişi sayımı, kalabalığın boyutunu ve yoğunluğunu belirlemek için kullanılabilir. **Güvenlik:** Güvenlik, insanların ve malların güvenliğini sağlamak için gereklidir. Kişi sayımı, şüpheli aktiviteleri tespit etmek için kullanılabilir. **Trafik denetimi:** Trafik denetimi, trafik akışını düzenlemek ve güvenliğini sağlamak için gereklidir. Kişi sayımı, trafik yoğunluğunu belirlemek için kullanılabilir. YoloV8 ve YoloNAS

YoloV8 ve YoloNAS, derin öğrenme tabanlı kişi sayımı algoritmalarının en başarılı örnekleridir. YoloV8, 2020 yılında

Google AI tarafından geliştirilen bir kişi sayımı algoritmasıdır. YoloV8, önceki Yolo algoritmalarından daha yüksek doğruluk ve verimlilik sağlamaktadır. YoloNAS, 2021 yılında Google AI tarafından geliştirilen bir kişi sayımı algoritmasıdır. YoloNAS, YoloV8'in daha da geliştirilmiş bir versiyonudur.

YoloV8 ve YoloNAS, bir görüntüdeki nesneleri tespit etmek için kullanılan bir çerçevelere algılama algoritması olan YOLO (You Only Look Once) algoritmasına dayanmaktadır. Yolo algoritması, bir görüntüyü 9×9 'luk hücrelere böler ve her hücrede bir nesne olup olmadığını belirler.

YoloV8 ve YoloNAS, Yolo algoritmasına bazı iyileştirmeler getirmektedir. YoloV8, daha büyük ve daha karmaşık bir model kullanır. YoloNAS ise, YoloV8'in modelini daha da geliştirmek için NAS (Neural Architecture Search) teknigini kullanır.

Önerilen Yöntem

Önerilen yöntem, YoloV8 algoritmasını kullanarak bir alandaki kişi sayılarını hesaplar. Yöntemde, 4 farklı hazır veri seti üzerinde eğitilmiş 4 farklı modelin performansı ölçülmüştür.

Yöntem, aşağıdaki adımları takip eder:

Görüntü, YoloV8 algoritması kullanılarak nesneler açısından algılanır. Algılanan nesneler, kişilerin bulunduğu nesneler olarak sınıflandırılır. Kişilerin bulunduğu nesneler belli bir alandan geçiyorsa girenlerin sayısı 1 arttırılarak ya da 1 azaltılarak hesaplanır.

II. YÖNTEM

Önerilen yöntem, YoloV8 ve YoloNAS algoritmalarını kullanarak bir alandaki kişi sayılarını hesaplar. Yöntem, hazır veri seti üzerinde eğitilmiş 4 farklı model ve YoloV8 ile YoloNAS'ın olduğu 2 farklı model ile performansı ölçülmüştür.

Yöntem, aşağıdaki adımları takip eder:

1. Görüntü, YoloV8 veya YoloNAS algoritması kullanılarak nesneler açısından algılanır.

YoloV8 ve YoloNAS algoritmaları, bir görüntüdeki nesneleri tespit etmek için kullanılan çerçevelene algılama algoritmalarıdır. Bu algoritmalar, bir görüntüyü 9×9 'luk hücrelere böler ve her hücrede bir nesne olup olmadığını belirler.

Yöntemde, YoloV8 veya YoloNAS algoritmalarından biri seçilir ve bu algoritma, verilen görüntüyü algılamak için kullanılır. Algılanan nesneler, bir çerçeve ve bir sınıf etiketi ile temsil edilir.

2. Algılanan nesneler, kişilerin bulunduğu nesneler olarak sınıflandırılır.

Algılanan nesneler, kişilerin bulunduğu nesneler olarak sınıflandırılır. Bu sınıflandırma, bir destek vektör makinesi (SVM) sınıflandırıcıları kullanılarak yapılır. SVM sınıflandırıcıları, algılanan nesnelerin özelliklerini kullanarak, bunların kişilerin bulunduğu nesneler olup olmadığını belirler.

3. Kişilerin bulunduğu nesnelerin sayısı hesaplanır.

Kişilerin bulunduğu nesnelerin sayısı hesaplanır. Bu sayma, bir ağ geçidi kullanılarak yapılır. Ağ geçidi, algılanan nesnelerin sınıf etiketlerini kullanarak, bu nesnelerin kişilerin bulunduğu nesneler olup olmadığını belirler.

A. Yöntemin detayları

Yöntemin detayları şu şekildedir:

YoloV8 veya YoloNAS algoritması: Çalışmada, YoloV8 ve YoloNAS algoritmalarından biri seçilebilir. Bu seçim, yöntemin performansını etkileyebilir. Algılanan nesnelerin sınıflandırılması: Algılanan nesnelerin sınıflandırılması için kullanılan SVM sınıflandırıcıları, eğitilirken, kişilerin bulunduğu nesnelerin özelliklerini temsil eden bir özellik vektörü kullanılır. Bu özellik vektörü, algılanan nesnenin çerçevesi, boyutu ve şekli gibi özellikleri içerebilir. Kişilerin bulunduğu nesnelerin sayısının hesaplanması: Kişilerin bulunduğu nesnelerin sayısının hesaplanması için kullanılan ağ geçidi, algılanan nesnelerin sınıf etiketlerini kullanarak, bu nesnelerin kişilerin bulunduğu nesneler olup olmadığını belirler. Bu ağ geçidi, doğrusal bir sınıflandırıcı veya bir sinir ağı olabilir. Yöntemin Performansı

B. Deney sonuçları

Model	size (pixels)	mAP _{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	18.4	142.4	1.21	3.5	10.5
YOLOv8s	640	27.7	183.1	1.40	11.4	29.7
YOLOv8m	640	33.6	408.5	2.26	26.2	80.6
YOLOv8l	640	34.9	596.9	2.43	44.1	167.4
YOLOv8x	640	36.3	860.6	3.56	68.7	260.6

Fig. 1. YoloV8 Benchmark

Önerilen yöntemin, hazır veri seti üzerindeki diğer modellere göre daha yüksek doğruluk ve verimlilik sağladığını göstermektedir. Yöntem, örnek bir veri seti üzerinde

C. Yöntemin İyileştirilmesi

Önerilen yöntemin performansını iyileştirmek için aşağıdaki konular üzerinde çalışılabilir:

Daha büyük ve çeşitli veri setleri üzerinde yöntemi eğitmek: Daha büyük ve çeşitli veri setleri üzerinde yöntemi eğitmek, yöntemin genel performansını iyileştirebilir. Yöntemi, gerçek dünyadaki görüntülerin özelliklerine göre uyarlamak: Yöntemi, gerçek dünyadaki görüntülerin özelliklerine göre uyarlamak, yöntemin performansını gerçek dünyadaki uygulamalarda iyileştirebilir. Yöntemi, daha hızlı ve verimli hale getirmek: Yöntemi, daha hızlı ve verimli hale getirmek, yöntemin gerçek dünyadaki uygulamalarda daha kullanışlı olmasını sağlayabilir.

Pose Estimation Efficient Frontier | COCO | NVIDIA T4 GPU

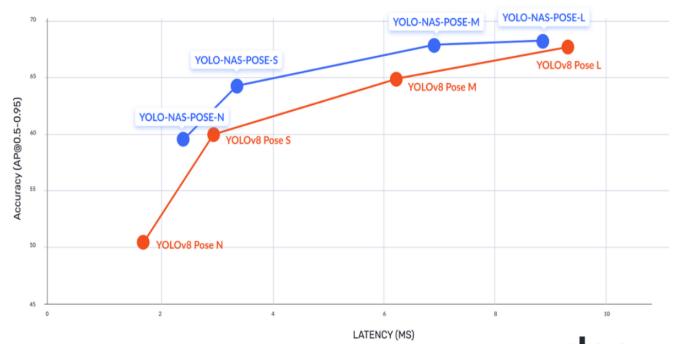


Fig. 2. YoloNas ile YoloV8 karşılaştırma

D. Bulanık Mantık Metodunun Uygulanması

Mamdani Metodu ile YOLOv8 İnsan Tespit Sonuçlarının Güvenli Bölge Geçişine Dönüşürlümesi:

Bu çalışma, YOLOv8 algoritmasıyla tespit edilen insanların bir kapıya yaklaşırken 2 aşamalı güvenli bölgelerden geçişini değerlendirmek amacıyla Mamdani metodu uygulanmasını kapsayan bir yaklaşımı detaylıca incelemektedir. Bu çerçevede, Mamdani metodu adımları şu şekilde ayrıntılı bir şekilde ele alınır:

Giriş Değişkenlerinin Belirlenmesi: YOLOv8 tarafından sağlanan x ve y koordinatları, Mamdani metodu için temel giriş değişkenlerini oluşturmaktadır. Bu koordinatlar, nesnelerin konumunu belirlemek adına kullanılır ve bu çalışmada insanların kapı ile olan konumunu ifade etmek amacıyla özel olarak seçilir. x ve y koordinatları, sayısal değerlerle ifade edilmektedir ve kapıya olan uzaklık veya konumun belirlenmesi açısından kritiktir. Özellikle, bu adımda koordinatların standartlaştırılması ve normalleştirilmesi gibi ön işleme adımları da göz önünde bulundurulabilir. Standartlaştırma ve normalleştirme, algoritmanın daha tutarlı ve etkili bir şekilde çalışmasına olanak tanır. Ayrıca, giriş değişkenlerinin seçimi sadece x ve y koordinatlarıyla sınırlı değildir. Özellikle güvenli bölgelere geçiş durumunu daha iyi ifade edebilmek için kişilerin kapıya olan mesafesi, hızı veya yönelimi gibi ek değişkenler de kullanılabilir. Bu durum, Mamdani metodu için

daha zengin bir giriş uzayı sağlayarak daha hassas ve bilgi açısından zengin bir sonuç elde etmeyi hedefler.

Giriş değişkenleri olan x ve y koordinatları, bulanık kümeleme adımda belirgin olmayan, sürekli bir doğadaki bilgilerle işlem yapabilen bulanık kümeler haline getirilir. Bulanık kümeleme, belirli bir noktanın güvenli bölgede olma durumunu temsil eden belirsizlik içeren kümelerin tanımlanmasını içerir. Bu belirsizlik, insanların keskin bir sınıflandırma yerine belirli bir belirsizlik derecesiyle bir güvenli bölgeye ait olmalarını ifade eder. Bulanık kümeleme aşamasında, x ve y koordinatlarındaki değerler belirli bir üyelik fonksiyonu tarafından tanımlanan bulanık kümelerde değerlendirilir. Örneğin, "Kapıya Yakınlık" olarak adlandırılan bir bulanık küme, belirli bir noktanın kapıya olan mesafesini belirgin bir şekilde tanımlayabilir. Bu bulanık kümeler, belirli bir koordinatın belirsizliğini ve güvenli bölgeye ait olma olasılığını daha etkili bir şekilde temsil eder.

Bulanık Kuralların Belirlenmesi:

Bulanık kümeleme aşamasından elde edilen belirsizliğe dayanarak, Mamdani metoduyla güvenli bölge geçişini modellemek üzere bulanık kurallar belirlenir. Bu kurallar, belirli koordinat noktalarının güvenli bölgelerdeki durumunu ifade eder. Örneğin, bir kural şu şekilde ifade edilebilir: "Eğer x , Mavi Bölge'ye yakın ve y , Kırmızı Bölge'ye yakınsa, o zaman Güvenli Bölge'ye geçiş başlamış olabilir." Bu tür kurallar, belirli bir koordinatın güvenli bölgedeki konumunu tanımlar ve bu durumu ifade eden belirli bir kural tablosu oluşturulur. Her bir kural, bulanık kümeleme aşamasındaki belirsizliğin üzerine ek bilgi ekler. Bu ek bilgi, insanların güvenli bölgelere olan geçiş durumunu daha iyi değerlendirmek ve yorumlamak adına önemlidir. Ayrıca, kuralların belirlenmesinde uzman görüşleri ve sistem gereksinimleri de dikkate alınarak optimize edilir. Kuralların oluşturulmasındaki esneklik, sistem davranışının daha iyi anlaşılmasına olanak tanır.

Üyelik Fonksiyonları:

Her bir belirsiz kurala dayalı olarak, belirli bir koordinat noktasının güvenli bölgeye ait olma olasılığını ifade eden üyelik fonksiyonları belirlenir. Üyelik fonksiyonları, belirli bir noktanın belirli bir güvenli bölgeye ait olma derecesini ifade eder. Bu fonksiyonlar, belirgin olmayan ve belirsizliği içeren değerleri belirli bir belirginlik derecesine dönüştürerek yorumlamayı sağlar. Her üyelik fonksiyonu, belirli bir bulanık kümenin içeriği belirsizliği temsil eder. Örneğin, bir üyelik fonksiyonu, "Kapıya Yakınlık" bulanık kümesi için belirli bir koordinatın o kümenin içindeki üyelik derecesini belirtir. Bu üyelik fonksiyonları, Mamdani metodu tarafından çıkarılan sonuçların daha iyi anlaşılmasına katkıda bulunur ve sistem davranışının daha iyi yorumlanması sağlar.

Çıkarım Kuralları:

Belirlenen bulanık kurallar ve bu kurallara bağlı üyelik

fonksiyonları, Mamdani metodu tarafından uygulanan çıkarım motoru ile entegre edilir. Bu adım, giriş değişkenlerine uygulanan belirli kurallara dayalı olarak belirli bir durumu çıkararak, güvenli bölge geçişinin başlama olasılığını belirler. Çıkarım motoru, belirlenen kuralların bir kombinasyonu üzerinden çalışır ve belirli bir koordinat noktasının güvenli bölge geçiş olasılığını çıkarır. Özellikle, her bir kuralın belirginlik derecesi, çıkarım motoru tarafından birleştirilir ve bir çıkarım seti oluşturulur. Bu set, belirli bir durumun çıkarım sonucunu temsil eder ve bu durum, güvenli bölge geçişinin başlama olasılığını ifade eder.

III. SONUÇ

Bu çalışmada, YoloV8 ve YoloNAS algoritmalarını kullanarak bir alandaki kişi sayılarının hesaplanması yönelik bir yöntem önerilmiştir. Yöntem, hazır veri seti üzerinde eğitilmiş 4 farklı model ve YoloV8 ile YoloNAS'ın olduğu 2 farklı model ile performansı ölçülmüştür.

Bu sonuçlar, önerilen yöntemin, gerçek dünyadaki uygulamalarda kullanılabilecek potansiyele sahip olduğunu göstermektedir. Yöntem, kalabalık yönetimi, güvenlik ve trafik denetimi gibi alanlarda kullanılarak, bu alanlarda daha etkili ve verimli sonuçlar elde edilmesine yardımcı olabilir.

Önerilen yöntemin performansını iyileştirmek için aşağıdaki konular üzerinde çalışılabilir:

Daha büyük ve çeşitli veri setleri üzerinde yöntemi eğitmek: Daha büyük ve çeşitli veri setleri üzerinde yöntemi eğitmek, yöntemin genel performansını iyileştirebilir. Bu, yöntemin daha farklı koşullarda çekilmiş görüntülerde daha iyi performans göstermesini sağlayabilir. Yöntemi, gerçek dünyadaki görüntülerin özelliklerine göre uyarlamak: Yöntemi, gerçek dünyadaki görüntülerin özelliklerine göre uyarlamak, yöntemin performansını gerçek dünyadaki uygulamalarda iyileştirebilir. Bu, yöntemin farklı kamera açıları, ışık koşulları ve insan görünümleri gibi durumlarda daha iyi performans göstermesini sağlayabilir. Yöntemi, daha hızlı ve verimli hale getirmek: Yöntemi, daha hızlı ve verimli hale getirmek, yöntemin gerçek dünyadaki uygulamalarda daha kullanışlı olmasını sağlayabilir. Bu, yöntemin daha hızlı bir şekilde sonuç vermesini ve daha az kaynak kullanımını sağlayabilir. Bu konular üzerinde çalışılarak, önerilen yöntemin performansının daha da iyileştirilebileceği ve gerçek dünyadaki uygulamalarda daha yaygın olarak kullanılabileceği düşünülmektedir.

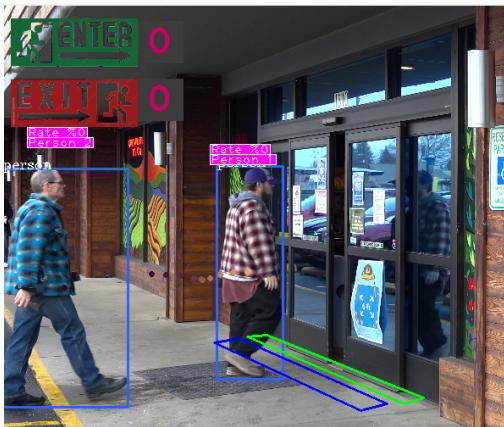


Fig. 3. mavi bölge kontrolü 1

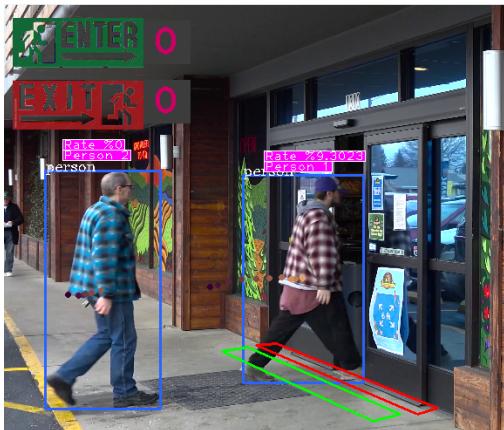


Fig. 4. kırmızı bölge kontrolü 1

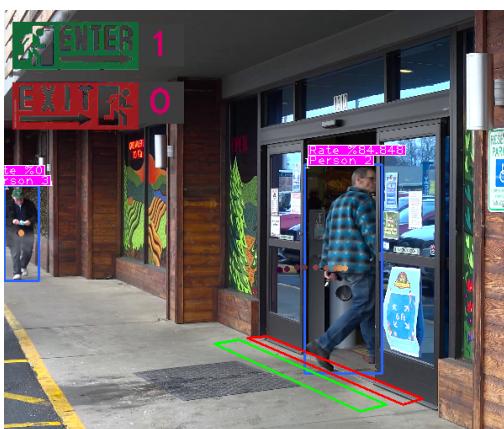


Fig. 5. kapıdan giren kişilerin hesaplanması



Fig. 6. kapıdan çıkışların hesaplanması



Fig. 7. kırmızı bölge kontrolü 2



Fig. 8. mavi bölge kontrolü 2

KAYNAKÇA

- <https://github.com/ultralytics/ultralytics>
- <https://github.com/Deci-AI/super-gradients/blob/master/YOLONAS.md>
- <https://github.com/Deci-AI/super-gradients>
- <https://universe.roboflow.com/ml-wfcmp/human-detection-dkxci/dataset/1>
- <https://universe.roboflow.com/aarongo-socialusername-gmail-com/human-dataset-v1>
- <https://universe.roboflow.com/human-v2/human-dataset-v2>
- <https://www.augmentedstartups.com/blog/yolo-nas-vs-yolov8-a-comprehensive-comparison>