



**T.C.
YALOVA ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**DERİN ÖĞRENME İLE YÖN BELİRLEME
VE CİNSİYET ANALİZİ**

**YÜKSEK LİSANS TEZİ
Hüsnü YAŞAR
175105004**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Doç.Dr. Abdulkadir TEPECİK

MAYIS 2021



**T.C.
YALOVA ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**DERİN ÖĞRENME İLE YÖN BELİRLEME
VE CİNSİYET ANALİZİ**

**YÜKSEK LİSANS TEZİ
Hüsnü YAŞAR
175105004**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Doç.Dr. Abdulkadir TEPECİK

MAYIS 2021

Niřanlıma, aileme, arkadaşlarıma ve saygıdeęer danışman hocama,



ÖNSÖZ

Lisans ve yüksek lisans eğitimim boyunca benden yardımlarını esirgemeyen, akademik anlamda bilgi, birikimi ile yön gösteren ve daima destekteğini gösteren sayın danışman hocam Doç.Dr. Abdulkadir TEPECİK'e teşekkür ederim.

Aileme, nişanlıma ve arkadaşlarıma her zaman yanımda oldukları ve desteklerini hiçbir zaman esirgemedikleri için teşekkür ederim.

Mayıs 2021

Hüsnü Yaşar
(Bilgisayar Mühendisi)





İÇİNDEKİLER

Sayfa No

ÖNSÖZ	İİİ
KISALTMALAR	Vİİ
ÇİZELGE LİSTESİ.....	İX
ŞEKİL LİSTESİ.....	Xİ
ÖZET.....	Xİİİ
ABSTRACT.....	XV
1. GİRİŞ	1
1.1 Tezin Amacı ve Katkısı.....	1
1.2 Literatür Araştırması	2
2. GÖRÜNTÜ İŞLEME.....	5
3. DERİN ÖĞRENME.....	7
3.1 Yapay Sinir Ağları	7
3.1.1 Konvolüsyonel Sinir Ağları (Convolutional Neural Networks)	10
3.1.2 Fast R-CNN.....	12
4. MATERYAL VE YÖNTEM	13
4.1 Materyal	13
4.1.1 Kullanılan Programlar ve Kütüphaneler	13
4.1.2 İnsan Veri Seti.....	13
4.2 Yöntem.....	15
4.2.1 Görüntü Etiketleme	15
4.2.2 İnsan Yön ve Cinsiyet Tespiti	15
5. SONUÇ	21
6. ÖNERİLER.....	27
KAYNAKLAR	31
ÖZGEÇMİŞ	33



KISALTMALAR

CNN	: Convolution Neural Network
SVM	: Support Vector Machine
Fast R-CNN	: Fast Region-based Convolution Neural Network
CuDNN	: The NVIDIA CUDA Deep Neural Network Library
KNN	: K-Nearest Neighbors
L-CNN	: Lightweight Convolution Neural Network
YAA-AÖM	: Yerel Alıcı Alanlar-Aşırı Öğrenme Makinesi
YSA	: Yapay Sinir Ağları
ReLU	: Rectified Linear Unit
TFRecord	: Tensorflow Format Record
GPU	: Graphics Processing Unit
CUDA	: Compute Unified Device Architecture
API	: Application Programming Interface



ÇİZELGE LİSTESİ

Sayfa No

Çizelge 1.1 : Karmaşıklık matrisi.	25
---	----





ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1. Digital bir görüntünün piksellerin belirgin ve değersel görünümü.....	5
Şekil 2.2. İyileştirilmiş görüntü örneği	6
Şekil 2.3. Tanımlanmış insan tespiti	6
Şekil 3.1. YSA model örneği	8
Şekil 3.2. Örnek aktivasyon fonksiyonları girdi-çıkıı grafiğı	9
Şekil 3.3. CNN yapı mimarisi	10
Şekil 3.4. Örnek görüntü verisi üzerinde konvolüsyon işlemleri	11
Şekil 3.5. Fast R-CNN mimari yapısı	12
Şekil 4.1. Görüntü verilerinden alınan sınıf verileri	14
Şekil 4.2. Etiketlenmiş örnek erkek-sağ sınıfı verisi.....	15
Şekil 4.3. Tensorflow depolama formatı (TFRecord).....	16
Şekil 4.4. Eğitim başlangıç değeri	17
Şekil 4.5. Eğitim sonu değeri.....	17
Şekil 4.6. Eğitim süresinde loss(kayıp) oranları	18
Şekil 4.7. Eğitim süresinde saniyedeki adım sayısı	19
Şekil 5.1. Sınıflandırma sonuçlarına örnekler.....	24
Şekil 6.1. Çocuk ve ergen kişi görüntü	27
Şekil 6.2. Fiziki insan çeşitliliğı örneğı.....	28
Şekil 6.3. AVM giriş çıkış görüntüsü	29



DERİN ÖĞRENME İLE YÖN BELİRLEME VE CİNSİYET ANALİZİ

ÖZET

Bu çalışmada kamera görüntülerinden elde edilen, kişilerin yönleri tayin edilerek, bulundukları lokasyonlara giriş ve çıkış bilgilerinin hesaplanması amaçlanmıştır. Tensorflow derin öğrenme algoritması ve Python nesne yönelimli, yorumlamalı programlama dili kullanılarak bu kişilerin cinsiyetlerin belirlenmesi ve tespiti yapılmıştır. Böylece hangi cinsiyetten kişilerin hangi yönlere ve ne kadar yoğunlukla yöneldikleri hesaplanmak istenmiştir. Havaalanı, AVM, istasyon gibi yerlerde bu bilgiler hesaplanmıştır.

Anahtar Kelimeler: Derin öğrenme, video akım yakalama, tensorflow, python, fast r-cnn.



DATA ANALYSIS WITH IMAGE PROCESSING TECHNIQUES AND CLASSIFICATION

ABSTRACT

In this study, it is aimed to calculate the entry and exit information of the locations they are located by determining the directions of the people obtained from the camera images. Gender determination and determination of these individuals were made using Tensorflow deep learning algorithm and Python object oriented, interpretive programming language. Thus, it was wanted to calculate which gender people headed in which direction and how intensely. This information has been calculated in places such as airports, shopping malls and stations.

Keywords: Deep learning, video streaming capture, tensorflow, python, fast r-cnn.





1. GİRİŞ

Dünyada her geçen gün değişen ihtiyaçlar ve zorunluluklar, beraberinde yeni inovasyon ihtiyaçları ve çözümleri yol açmaktadır. Bu durumlar arasında önem arz eden ve en çok görülenlerden birisi de toplumsal hareketliliğin takibidir. Özellikle Covid-19 pandemi dönemi gibi, toplumsal hareketliliğin takibinin ne kadar önemli olduğu göstermektedir. Pandemi veya doğal afet durumunda insanların hareketliliğinin takibi önem arz etmektedir. Özellikle bu hareketlilik içerisinde yön analizi, cinsiyet belirleme, sayım, yetişkinlik analizi, hatta kimlik analizi gelişen teknolojiler ile mümkün kılınmaya ve bir ihtiyaç olmaya başlamıştır.

Pandemi, doğal afet gibi ihtiyaç dışı zorunluluk durumlarında insan hareketlerinin takibinin sağlanması önemli olduğu gibi ticari alanlarda da yaygınlaşan bir ihtiyaç haline gelmiştir. Kar - zarar marjlarını, talep – yoğunluk istatistiklerini göz önüne alıp, analiz edebilmek için de hareketliliğin takibinin gereksinim haline geldiği son yıllarda gözlemlenmiştir. Özellikle ticari kuruluşlar tarafından sahip oldukları lokasyonlardaki mağazalarının insanların ne sıklıkla ziyaret ettikleri, bu yoğunluk içerisinde ne kadar kar ettikleri, hangi profile insanlara daha çok hitap ettiklerini hatta işletmesinde ihtiyaç duyacağı kaynağı ve personeli dahi analiz edebileceği birçok istatistiksel ve gözlemsel verilere ihtiyaç duyduğu belirlenmiştir.

1.1 Tezin Amacı ve Katkısı

Günümüz teknolojilerinde derin öğrenme yöntemlerinin birçok alanda olduğu gibi görüntü işleme ve çözümleme alanında da hem başarı hem de performans yüzdesi artışı ortaya koyduğu araştırılmış ve saptanmıştır. Literatürde, derin öğrenme algoritmaları ile birçok görüntü işleme tekniği uygulandığı gözlemlenmiştir. Kullanılan bu derin öğrenme algoritmaları, dünya üzerinde bu paralelde iş yapan özel şirketler tarafından da açık kaynak kodlu yazılım kütüphaneleri haline getirilerek hız kazanmıştır.

Tez kapsamında, insanların yön tahmini ve cinsiyet analizi sınıflandırma işlemleri gerçek zamanlı olarak tespit edilmesi hedeflenmiştir. Hazır olarak alınan mpii human pose v1 veriseti bu deneysel çalışmada kullanılmıştır. Yön tahmini ve cinsiyet analizi için verilerden öz nitelik çıkarma ve sınıflandırma için CNN, SVM ve Regresyon kombinasyonu olan Fast R-CNN modeli kullanılmıştır. Bu model Tensorflow kütüphanesi ve GPU eğitim süreci için CUDA ve CuDNN toolkitleri ile eğitilmiştir.

1.2 Literatür Araştırması

İnsanların yön tahmini ve cinsiyet analizi hakkında birçok çalışma bulunmaktadır. Ancak yapılan çalışmalara bakıldığında hem yön tahminin hem de cinsiyet analizinin bir arada değil, ayrı ayrı tespit edildikleri görülmektedir. İncelenen literatür çalışmalarında genel olarak insan hareketleri ve nesne tespitleri üzerine araştırmalar yapılmıştır. Bu çalışmalar genel olarak hazır veri setleri, hazır kütüphaneler ve modeller ile gerçekleştirilmiştir. İncelenen literatür araştırmaları aşağıdadır.

Chenghua ve arkadaşları [1], insan hareketi aktivite türlerini tanımak için örüntü tanıma, KNN ve BPNN makine öğrenmesi algoritmalarını kullanarak sınıflandırmalar yapmış, hareket aktivitelerini tespit etmişlerdir.

Choon ve arkadaşları [2], insanların yaş, cinsiyet ve etnik köken gibi demografik özelliklerini kullanarak, hareketsiz görüntü ve videolar üzerinde denetimli öğrenme tekniğiyle nesne algılama, ön işleme, öz nitelik çıkarma ve sınıflandırma yapmışlardır. Bu çalışmada SVM ve Bayes sınıflandırıcıları kullanılarak tespitler yapılmıştır.

Mane ve arkadaşları [3], nesne tanımlama ve tespiti için Tensorflow kütüphanesi ve CNN algoritması kullanarak, çeşitli videolar üzerinde test edip %90 oranında doğruluk payı yakalamışlardır.

Sai ve arkadaşları [4], tehdit edici nesnelerin tespiti ve sayımı için Tensorflow Nesne Algılama Api'si ve Fast R-CNN algoritması kullanmışlardır. Model, tehdit edici nesneleri algılayan iki sınıf için test verileri üzerinde değerlendirilmiştir.

Nikouei ve arkadaşları [5], gerçek zamanlı insan algılama için Raspberry Pi 3 yardımı ile OpenCV kütüphanesini kullanarak, tatmin edici performans elde etmişlerdir. L-CNN algoritması kullanılarak uygun maliyetli bir sınıflandırma yapılmıştır.

Bolme ve arkadaşları [6], Correlation kenar görüntülerinin ilişkilendirmeye dayalı filtre kullanılarak basit insan tespiti modeli oluşturmuşlardır. Değerlendirme PETS 2009 veri kümesinde gerçekleştirilir ve sonuçlar OpenCV kademeli sınıflandırıcı ve son teknoloji ürünü deforme olabilen parçalara dayalı kişi detektörü ile karşılaştırılmıştır.

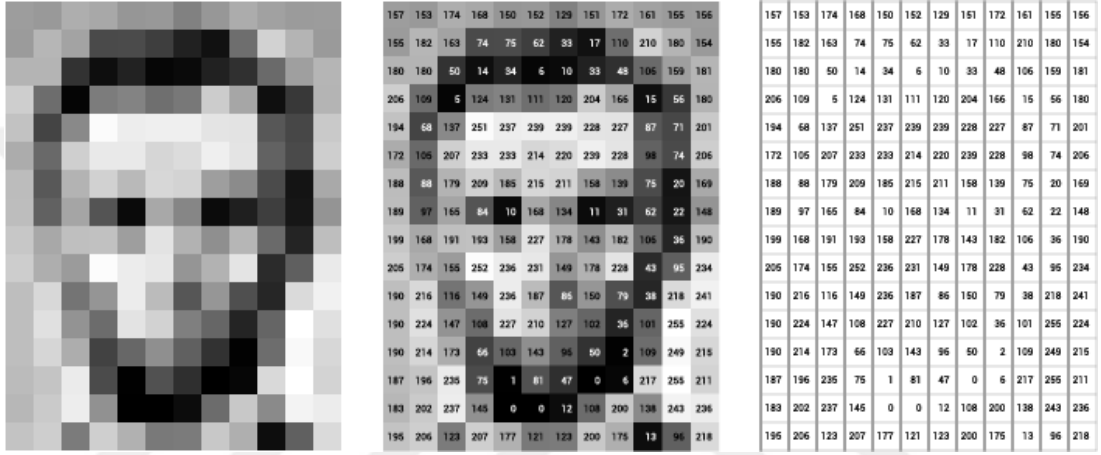
Akbulut ve arkadaşları [7], insan yüz görüntüleri üzerinden cinsiyet tahmini gerçekleştirmişlerdir. Tespit modeli için Yerel Alıcı Alanlar-Aşırı Öğrenme Makinesi ve Evrişimsel Sinir Ağları (Convolutional Neural Networks) kullanılmıştır. YAA-AÖM ve CNN modellerinin tespit sonuçları sırasıyla %80 ve %87 olarak hesaplanmıştır.





2. GÖRÜNTÜ İŞLEME

Görüntü işleme, görüntü verilerin değerlendirildikten sonra elektronik ortamda okunabilir biçimde dönüştürülmesine ve aktarılmasına yönelik olan sinyal işlemeden farklı bir işlemdir [8]. Bir görüntü, x ve y 'nin uzaysal koordinatlar olduğu iki boyutlu bir fonksiyon olarak tanımlanabilir [9]. $f(x,y)$ 'nin herhangi bir çift koordinattaki genliği, o görüntünün yoğunluğu veya gri seviyesi olarak adlandırılır. x , y ve f 'nin yoğunluk değerlerinin tümü sonlu, ayrık nicelikler olduğunda, görüntüye dijital görüntü denir. Bu dijital görüntü üzerindeki görüntü elemanına piksel denir.



Şekil 2.1. Digital bir görüntünün piksellerin belirgin ve değersel görünümü

Görüntü işlemenin sağladığı faydalar, günümüzde güvenlik, sağlık, mimarlık, jeoloji, tarım, otomotiv vb. gibi daha özel birbirinden farklı birçok alanda kullanılabilmektedir [10].

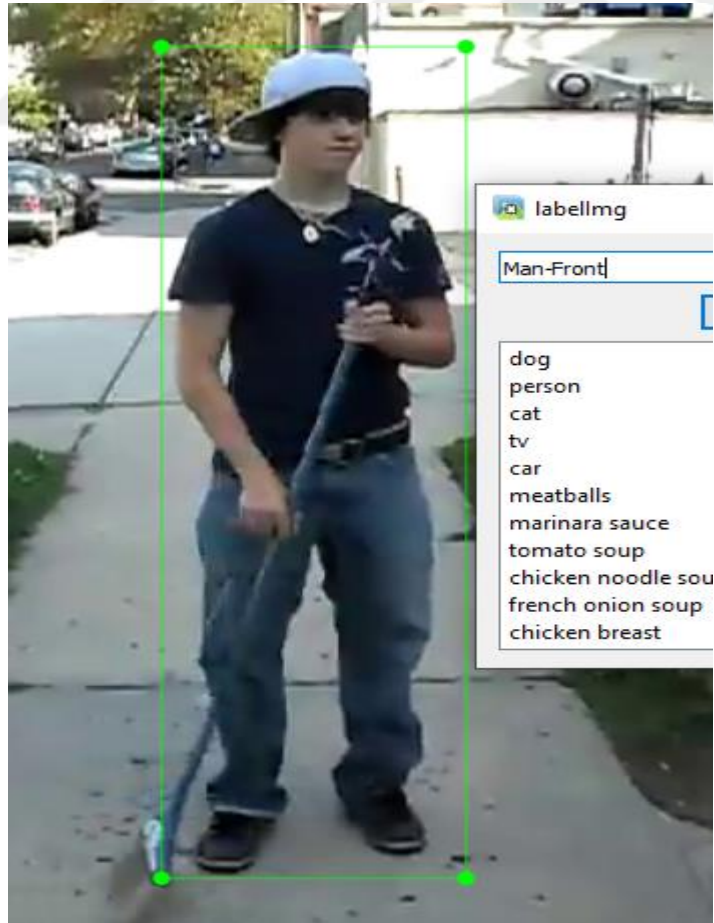
Görüntü işleme teknikleri kullanılarak ihtiyaç duyulan şu işlemler gerçekleştirilebilir:

- Temiz görüntü elde etmek için görüntü üzerindeki gürültülerin arındırılması,
- Daha kaliteli bir görüntü elde edilmesi,



Şekil 2.2. İyileştirilmiş görüntü örneği

- İnsanın ayırt etmesi güç olan nesnelerin tespit edilmesi.(Şekil 2.3)



Şekil 2.3. Tanımlanmış insan tespiti

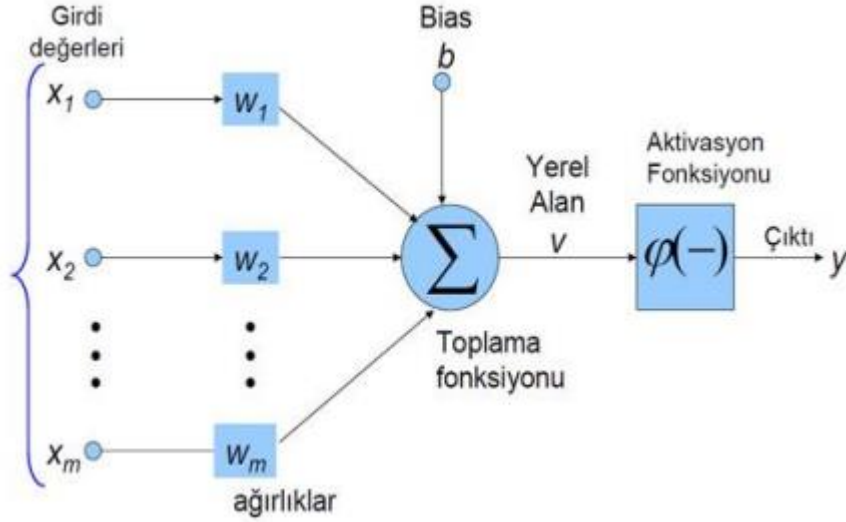
3. DERİN ÖĞRENME

Derin öğrenme, sinir ağları üzerinde birden fazla gizli katmandan oluşan, düşük maliyetli bir makine öğrenme algoritmasıdır. Makine öğrenimi sistemleri, görüntü veya stream üzerinden nesneleri tahmin etmek, sinyaller üzerinden sesi tanımlamak, kullanıcı hareketleri üzerinden ilgi alanlarını eşleştirmek ve eşleşen sonuçlarını önermek v.b için kullanılır. Bu uygulamalar, giderek artan bir şekilde derin öğrenme adı verilen sınıftan yararlanmaktadır [11].

Bilgisayarların görüntü ve konuşma tanıma gibi algısal sorunları çözmesini sağlayan güçlü bir makine öğrenimi biçimidir. Çok büyük veri kümelerindeki kalıpları ve yapıları keşfetmek için birden çok işlem katmanı kullanılır. Yapay sinir ağları gibi derin öğrenme yöntemlerinde, kaç katmanın gerekli olduğuna dair kesin ve hızlı bir kural yoktur, ancak çoğu uzman ikiden fazlasının gerekli olduğu yaygın kabul görmüştür [12].

3.1 Yapay Sinir Ağları

Yapay sinir ağları, insan beyninin nörofiziksel yapısının çalışma mantığından esinlenerek ortaya çıkarılan matematiksel bir modeldir [13]. YSA, insan beyninin öğrenme, optimizasyon, analiz, sınıflandırma, genelleme ve değerlendirme gibi fonksiyonel özelliklerini başarılı bir şekilde uygulamaktadır [14]. YSA girdi ve çıktı değişkenleri, ağırlıklar, toplama fonksiyonu ve aktivasyon fonksiyonundan oluşan bir model yapısıdır. Amaç girdi ve çıktı bilgilerine göre ağın aradaki ilişkiyi öğrenmesini sağlamaktır. Bu şekilde ağ öğreticili öğrenim yöntemi ile eğitilmiş olacaktır. Şekil 3.1’de YSA model örneği görülmektedir.



Şekil 3.1. YSA model örneği

Girdiler: Biyolojik sinir sistemindeki nöronlara gelen yani eğitim için gerekli olan verilerdir.

Ağırlıklar: Yapay hücreye gelen bilgilerin hücre üzerinde etkisine göre değersel olarak büyüyüp küçülen birimdir. Ağın değerinin sıfıra yakın olması o ağın iyi eğitildiğini göstermektedir.

Toplama Fonksiyonu: Hücreye gelen girdiyi hesaplayan fonksiyondur. Yaygın kullanımı ağırlıkların toplamıdır. Bunu dışında çarpım, maksimum, minimum, çoğunluk, kümülatif toplamı vb. gibi toplama fonksiyonları vardır.

$$\text{Net} = \sum_{i=1}^N X_i * W_i \quad (3.1)$$

$$\text{Net} = \prod_{i=1}^N X_i * W_i \quad (3.2)$$

$$\text{Net} = \text{Max}(X_i * W_i) \quad (3.3)$$

$$\text{Net} = \text{Min}(X_i * W_i) \quad (3.4)$$

$$\text{Net} = \sum_{i=1}^N \text{Sgn}(X_i * W_i) \quad (3.5)$$

$$\text{Net} = \text{Net}(\text{eksi}) + \sum_{i=1}^N X_i * W_i \quad (3.6)$$

Toplam (3.1): Girdiler modeldeki ağırlık değerleri ile çarpılır. Net girdi, çarpım değerleri ile toplanarak hesaplanır.

Çarpım (3.2): Girdiler modeldeki ağırlık değerleri ile çarpılır. Net girdi çarpım değerlerinin kendi aralarında çarpılması ile hesaplanır.

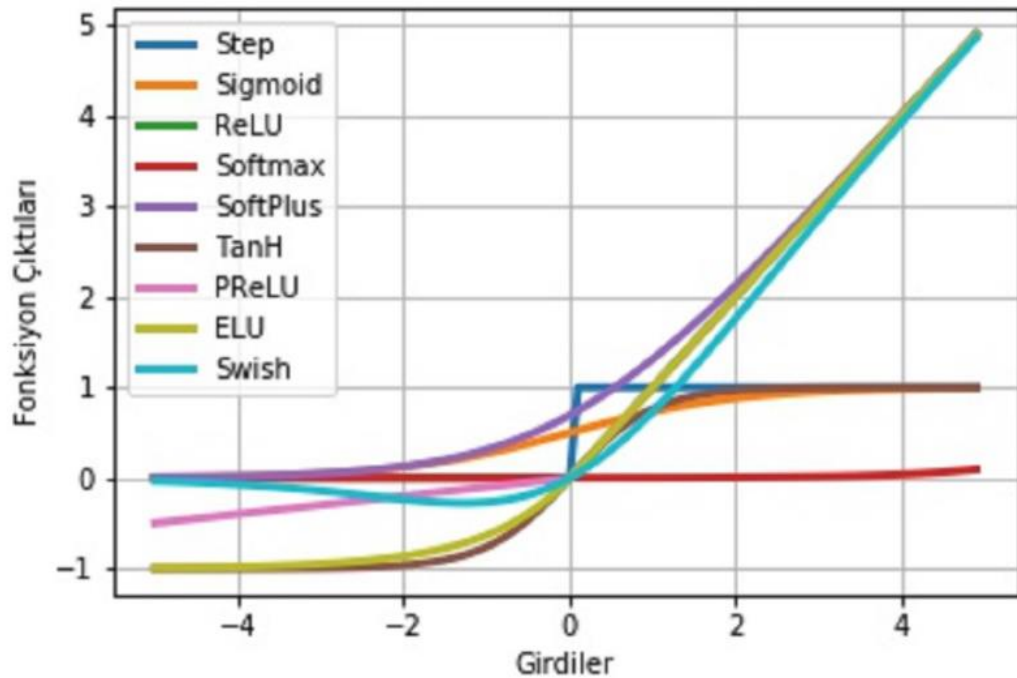
Maksimum (3.3): Girdiler modeldeki ağırlık değerleri ile çarpılır. Net girdi aralarından en büyüğü kabul edilir.

Minimum (3.4): Girdiler modeldeki ağırlık değerleri ile çarpılır. Net girdi aralarından en küçüğü kabul edilir.

Çoğunluk (3.5): Girdiler modeldeki ağırlık değerleri ile çarpılır. Net girdi çarpım sonuçların pozitif ve negatif değerlerinin sayı olarak en büyüğü kabul edilir.

Kümülatif (3.6): Girdiler modelde ki ağırlık değerleri ile toplanır. Net girdi hücreye öncesinde gelen girdiler ile hali hazırdaki girdi değerleri eklenerek hesaplanır.

Aktivasyon Fonksiyonu: Hücreye gelen girdilerin işlenip bu girdiye karşılık olarak çıktısını belirleyen fonksiyondur. Genellikle doğrusal olmayan fonksiyonlar tercih edilir. Doğrusal, Step, Sigmoid, ReLU, Hiperbolik Tanjant Fonksiyonları vb. aktivasyon fonksiyonlarına örnek olarak verilir.

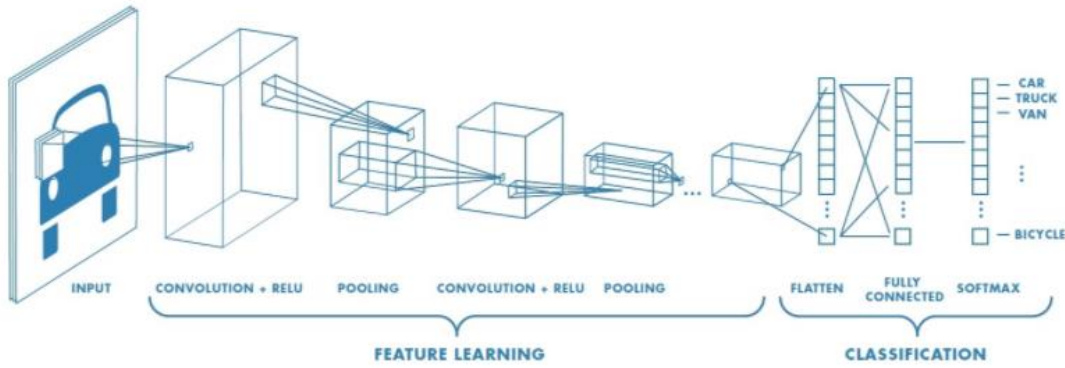


Şekil 3.2. Örnek aktivasyon fonksiyonları girdi-çıkı grafiği

Çıktılar: Aktivasyon fonksiyonundan çıkan veridir. Sonuç ya da başka bir hücre için giriş verisi olabilir.

3.1.1 Konvolüsyonel Sinir Ağları (Convolutional Neural Networks)

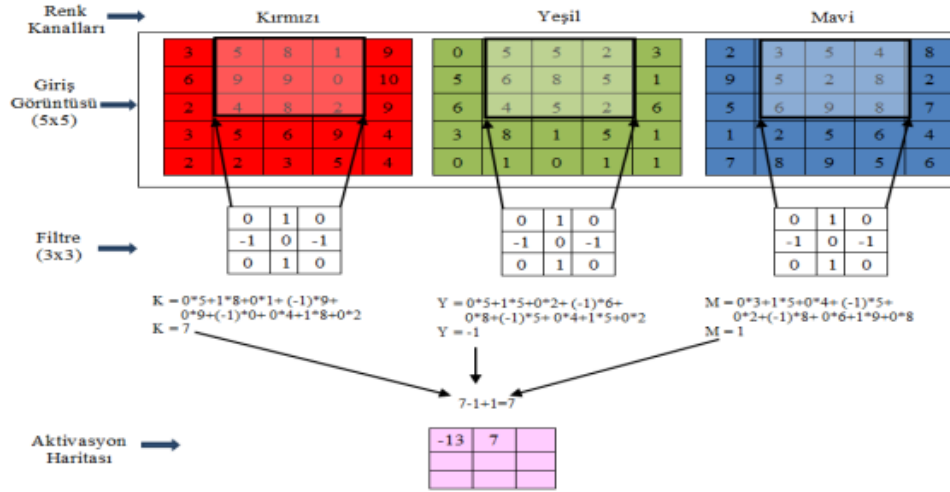
CNN sinir ağı, omurgalı hayvanların görüş simülasyonu taklit edilmesiyle tasarlanmış, Neocognitron ağı yapısına dayanır [15]. Genellikle görüntü işlemede kullanılan bu derin öğrenme algoritmasının, görseller üzerinde özellikleri (feature) ortaya çıkaran ve sınıflandıran farklı katmanları vardır. Bu katmanlar; Konvolüsyon Katmanı, Düzleştirilmiş Doğrusal Birim Katmanı (ReLU), Havuzlama Katmanı ve Tam Bağlantılı Katman'dır. Bu katmanlarda farklı işlemlere uğrayan veri, derin öğrenme modeline girmeden önce hazır hale gelmektedir.



Şekil 3.3. CNN yapı mimarisi

3.1.1.1 Konvolüsyon Katmanı

Erişim Katmanı olarak bilinen bu katman CNN algoritmasının ilk katmanıdır. Görüntü piksellerden oluşmaktadır ve her piksele karşılık matematiksel değerler vardır. Bu katman görselin gerçeğinden daha küçük ve rastgele ölçekli filtre içermektedir. Bu filtre görüntünün diğer bölgelerine de uygulanarak her bölge için farklı öz nitelikler elde edilmektedir [16]. Örneğin, giriş verisi bir renkli görüntü (RGB) $5 \times 5 \times 3$ veri boyutludur. 3×3 bir matris filtre konvolüsyon işlemi için kullanılmaktadır. Filtre görüntü üzerinde belirli adım ile sağa sola kaydırılarak kenar sınırlarına gelindiğinde bir alt basamağa inmektedir. Alt basamağa inildikten sonra tekrar kayma işlemi devam etmektedir. Bu işlem bütün görüntü matrisinin üzerinde gerçekleştirilmektedir. Üç renk kanalında yapılan konvolüsyon işleminin sonunda toplamları aktivasyon haritasını oluşturmaktadır [17].



Şekil 3.4. Örnek görüntü verisi üzerinde konvolüsyon işlemi

3.1.1.2 Düzleştirilmiş Doğrusal Birim Katmanı (ReLU)

Doğrulamalı birim olarak bilinen bu katman konvolüsyon katmanından sonra gelmektedir. Konvolüsyon katmanından çıkan matematiksel verinin aktifleşmesi için doğrusal olmayan yapıya Denklem 3.7’de kullanılan formül ile sokulmaktadır. Bu ReLU işlemi ile ağ daha da hızlanmaktadır. Negatif değerlere sıfır değeri atanarak azaltılmakta ve sadeleştirilmektedir.

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (3.7)$$

3.1.1.3 Havuzlama Katmanı (Pooling)

Aktivasyon katmanından sonra gelen ayrıca yeni bir konvolüsyon katmanı için giriş verilerini sağlayacak katmandır. Havuzlama işleminin temel amacı, her filtrenin çıktısının temsili özelliklerini çıkarmaktır. Mimarinin en önemli parçasıdır; çünkü filtrelenen verilerin vektörlere dönüştürülmesini sağlamaktadır [18]. Filtrelenen matrisler üzerinde maksimum değeri Denklem 3.8’e göre maksimum değeri ya da ortalama değeri alınarak havuzlama işlemi yapılmaktadır. Veri kaybı söz konusu olsa da daha az hesaplama yapmayı ve modelin ezber yapmasını önlemektedir.

$$y = \max(0, x) \quad (3.8)$$

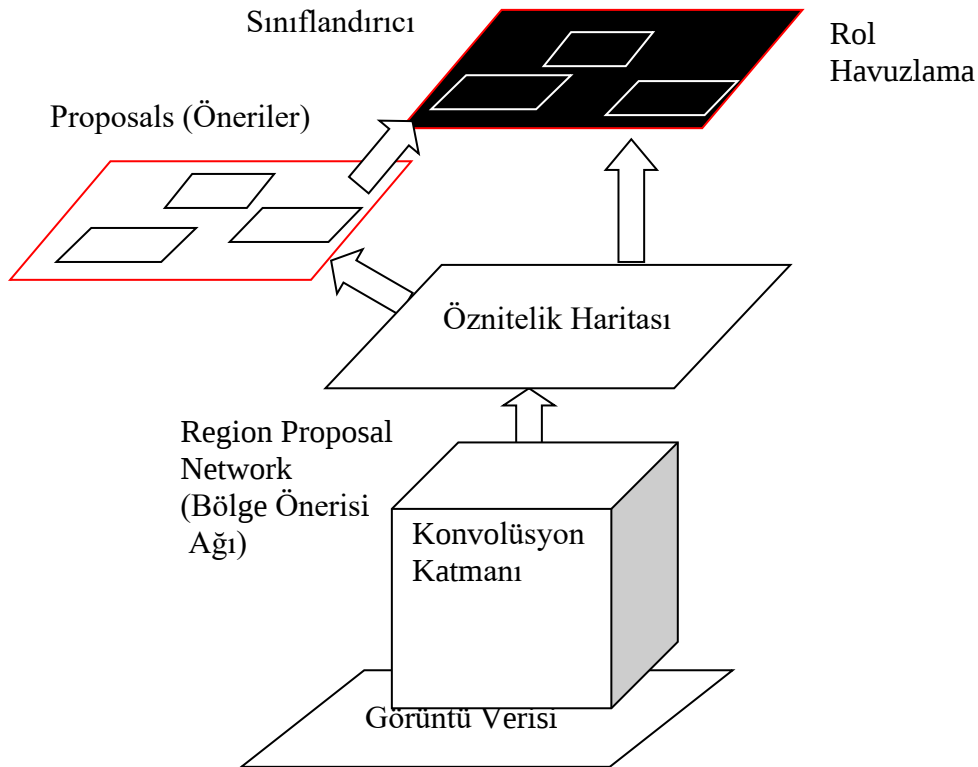
3.1.1.4 Tam Bağlantılı Katman (Fully Connected Layer)

Öncesindeki katmanlardan ve havuzlama katmanından geçen verilerin vektör haline getirildiği ve bir kayıp fonksiyonu sayesinde eğitim için en uygun ağırlıkların hesaplandığı katmandır. Bu katmanda softmax aktivasyon fonksiyonu kullanılarak sınıflandırma tahmini yapılmaktadır.

3.1.2 Fast R-CNN

R-CNN, görüntü verisini binlerce bölge önerilerine bölerek, daha sonra her bölge için CNN modelini uygulayarak öznitelik haritalarını çıkartır ve SVM algoritması ile sınıflandırır.

R-CNN’i daha hızlı hale getirmek için Grishick tarafından 2015 yılında, üç bağımsız modeli ortak eğitilmiş tek bir çerçevede birleştirilerek, Fast R-CNN adı verilen, hesaplama sonuçlarını artıran eğitim prosedürü geliştirilmiştir [19]. Her bölge önerisi için CNN özellik vektörlerini bağımsız olarak çıkarmak yerine, bu model bunları tüm görüntü boyunca tek bir CNN ileri geçişinde toplar ve bölge önerileri bu özellik matrisini paylaşır. Daha sonra aynı özellik matrisi, nesne sınıflandırıcısı ve sınırlayıcı kutu regresörü, öğrenmek için kullanılmak üzere dallanır.



Şekil 3.5. Fast R-CNN mimari yapısı

4. MATERYAL VE YÖNTEM

4.1 Materyal

4.1.1 Kullanılan Programlar ve Kütüphaneler

Bu çalışmada, veri seti için seçilmiş resimleri etiketlemek ve sınıfını belirlemek amacı ile LabelImg adlı Python tabanlı açık kaynak kodlu program kullanılmıştır [20]. Tensorflow tarafında ise derin öğrenme için uçtan uça açık kaynaklı kütüphanesi kullanılmıştır [21].

4.1.2 İnsan Veri Seti

Bu çalışmada, toplamda 757 adet bir ya da birden fazla insan görüntüleri olan görüntü verisi kullanılmıştır. Görüntü verilerini 568 adeti eğitim için kullanılırken, 189 adet veri test için kullanılmıştır. Görüntü verileri alınan mpii human pose v1 veri seti kullanılmıştır [22]. Bu görüntüler üzerinde cinsiyet ve yön kategorisine göre sekiz farklı sınıf oluşturulmuştur. Bu sınıflar; erkek-ön, erkek-sağ, erkek-sol, erkek-arka, kadın-ön, kadın-sağ, kadın-sol ve kadın-arka'dır. Şekil 4.1'de yapılan sekiz farklı sınıfın örnek görüntüleri verilmiştir. (a) erkek-ön, (b) erkek-sağ, (c) erkek-sol, (d) erkek-arka, (e) kadın-ön, (f) kadın-sağ, (g) kadın-ol ve (ğ) kadın-arka sınıflarına örnek olarak görülmektedir.



(a)



(b)



(c)



(d)



(e)



(f)



(g)



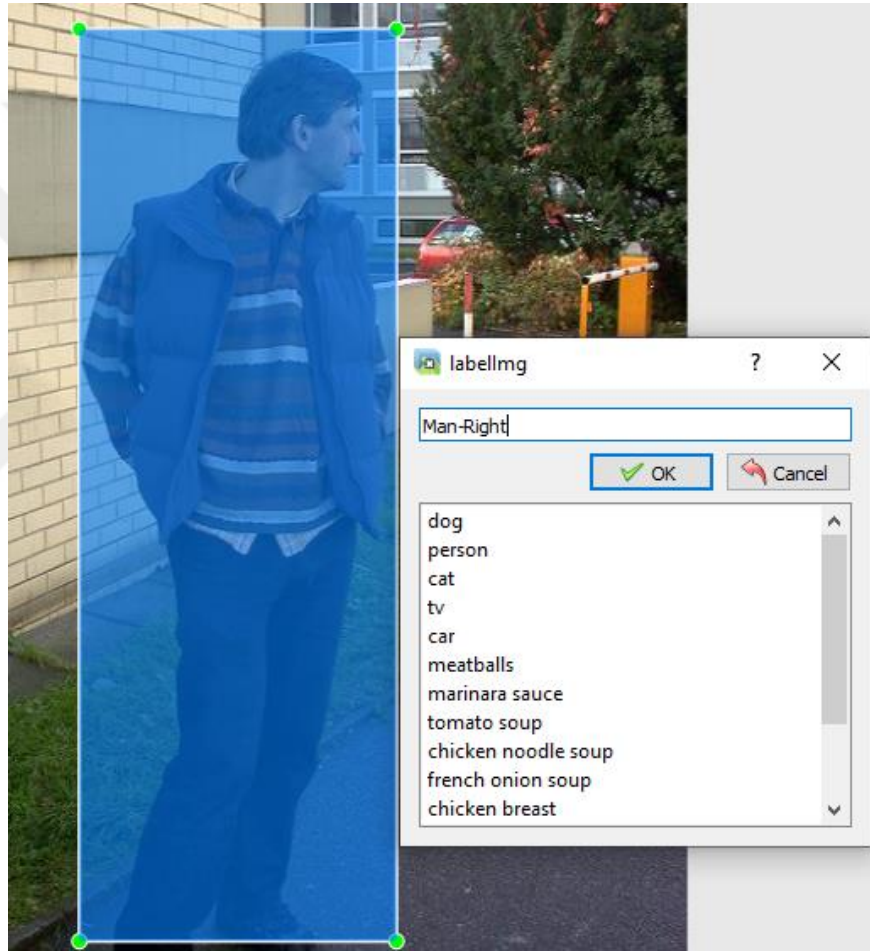
(ğ)

Şekil 4.1. Görüntü verilerinden alınan sınıf verileri

4.2 Yöntem

4.2.1 Görüntü Etiketleme

LabelImg açık kaynak kodlu program ile görüntüleri etiketlenmiş ve sınıflarını tanımlamıştır. Her görüntü verisi programın arayüzden etiketlendikten sonra bir xml dosyası olarak kayıt edilmektedir. Kayıt edilen xml dosyasında görüntü verisinin adı, yolu, kaynağı, resmin genişliği ve yüksekliği, etiketin (sınıfın) adı, etiketin genişliği ve yüksekliği vb. gibi bilgilerin yer almaktadır. Şekilde 4.2’de LabelImg programı üzerinde yapılmış olan bir etiketleme işlemi gözükmemektedir.



Şekil 4.2. Etiketlenmiş örnek erkek-sağ sınıfı verisi

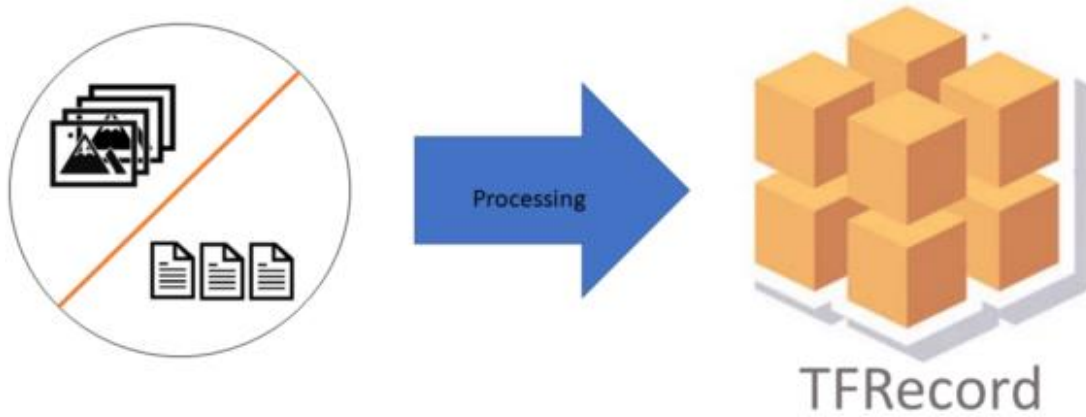
4.2.2 İnsan Yön ve Cinsiyet Tespiti

Etiketlenen görüntü verilerinin genel kısmı eğitim için ayrılmış iken kalan kısmı ise test için kullanılmaktadır. Yapılan araştırmalarda, 1/9 oranı yani görüntülerin %90’ı eğitim için geri kalan %10’u alt limit olarak kullanılmaktadır. Bu oranlar ihtiyaçlara

uygun şekilde deęiřebilmektedir. Bu alıřmada yapılan hesaplamalardan grnt verilerinin en iyi 1/3 ile sonu verdięi grldęnden grnt verilerinin yaklařık olarak %67'si eęitim verisi olarak, %33' ise test verisi olarak kullanılmaktadır. Kullanılan grnt verileri eęitim ve test verileri seildikten sonra hem grnt verileri hem de LabelImg'de oluřturulan xml dosyaları eęitim ve test klasrleri altında ayrılmaktadır.

Tensorflow ktphanesi, kullanılan etiketleri yani sınıfları tam sayı deęerleri ile tanımlandıęı bir etiket haritasına (label map) ihtiya duymaktadır. Bu etiket haritası hem eęitim hem de tespit srelerinde kullanılmaktadır. Blnen eęitim ve test verileri TFRecord formatına dnřtrlmektedir.

TFRecord: İkili dizeler dizisi olarak verilerin depolanması formatıdır. Sıralı bir řekilde grnt verilerini ya da 1 boyutlu vektrleri okumakta ve yazmaktadır.



řekil 4.3. Tensorflow depolama formatı (TFRecord)

Tensorflow ktphanesini alıřtırmak iin bir GPU kullanımı gerekli olmasa da hesaplama yapma konusunda nemli kazanları vardır. GPU kullanımı iin makinenin donanımına uygun CUDA kitaplıklarına ihtiya duyulmaktadır.

Tespit ve sınıflandırma iřlemleri iin konvolsyon sinir aęları tabanlı Fast R-CNN derin ęrenme algoritması kullanılarak model oluřturulmuřtur. Her bir grnt verisi model ierisinde 640x640 piksel olacak řekilde boyutlandırılmaktadır. Modelde sekiz sınıf, etiket haritası ile uygulanmaktadır. Mevcut belleęe baęlı olarak batch size(parti boyutu) iki olarak ayarlanmaktadır. Eęitim toplam yz otuz bin adımda gerekleřecek řekilde model yapılandırılmaktadır.

Şekil 4.4 ve Şekil 4.5’de epoch’ların (devirlerin) başlangıç anı ve bitiş anındaki loss (kayıp) oranları görüntülenmektedir. Sıfıra yaklaşan loss (kayıp) oranı eğitimin başarılı bir şekilde tamamlandığını göstermektedir.

```
INFO:tensorflow:Step 100 per-step time 0.281s loss=2.236
I0420 00:11:20.102019 9220 model_lib_v2.py:648] Step 100 per-step time 0.281s loss=2.236
INFO:tensorflow:Step 200 per-step time 0.266s loss=0.900
I0420 00:11:47.474971 9220 model_lib_v2.py:648] Step 200 per-step time 0.266s loss=0.900
INFO:tensorflow:Step 300 per-step time 0.250s loss=1.529
I0420 00:12:14.830404 9220 model_lib_v2.py:648] Step 300 per-step time 0.250s loss=1.529
INFO:tensorflow:Step 400 per-step time 0.234s loss=0.955
I0420 00:12:41.925459 9220 model_lib_v2.py:648] Step 400 per-step time 0.234s loss=0.955
INFO:tensorflow:Step 500 per-step time 0.250s loss=0.716
I0420 08:36:44.353888 9220 model_lib_v2.py:648] Step 500 per-step time 0.250s loss=0.716
```

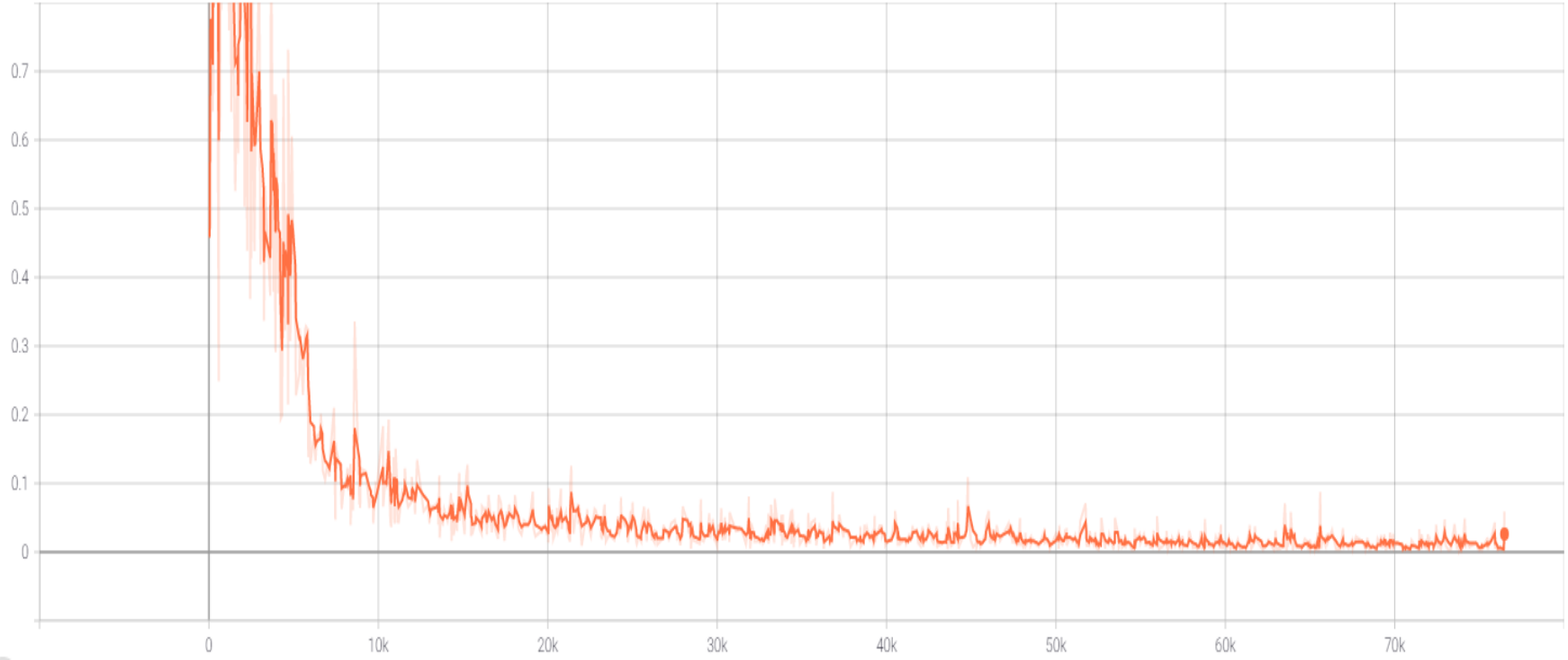
Şekil 4.4. Eğitim başlangıç değerleri

```
I0420 18:02:41.672036 9220 model_lib_v2.py:648] Step 129600 per-step time 0.281s loss=0.019
INFO:tensorflow:Step 129700 per-step time 0.250s loss=0.012
I0420 18:03:07.671263 9220 model_lib_v2.py:648] Step 129700 per-step time 0.250s loss=0.012
INFO:tensorflow:Step 129800 per-step time 0.281s loss=0.021
I0420 18:03:33.934562 9220 model_lib_v2.py:648] Step 129800 per-step time 0.281s loss=0.021
INFO:tensorflow:Step 129900 per-step time 0.266s loss=0.006
I0420 18:04:00.267519 9220 model_lib_v2.py:648] Step 129900 per-step time 0.266s loss=0.006
INFO:tensorflow:Step 130000 per-step time 0.328s loss=0.004
I0420 18:04:29.685884 9220 model_lib_v2.py:648] Step 130000 per-step time 0.328s loss=0.004
(tfenv) D:\MachineLearning\Tensorflow2021\training_demo>
```

Şekil 4.5. Eğitim sonu değerleri

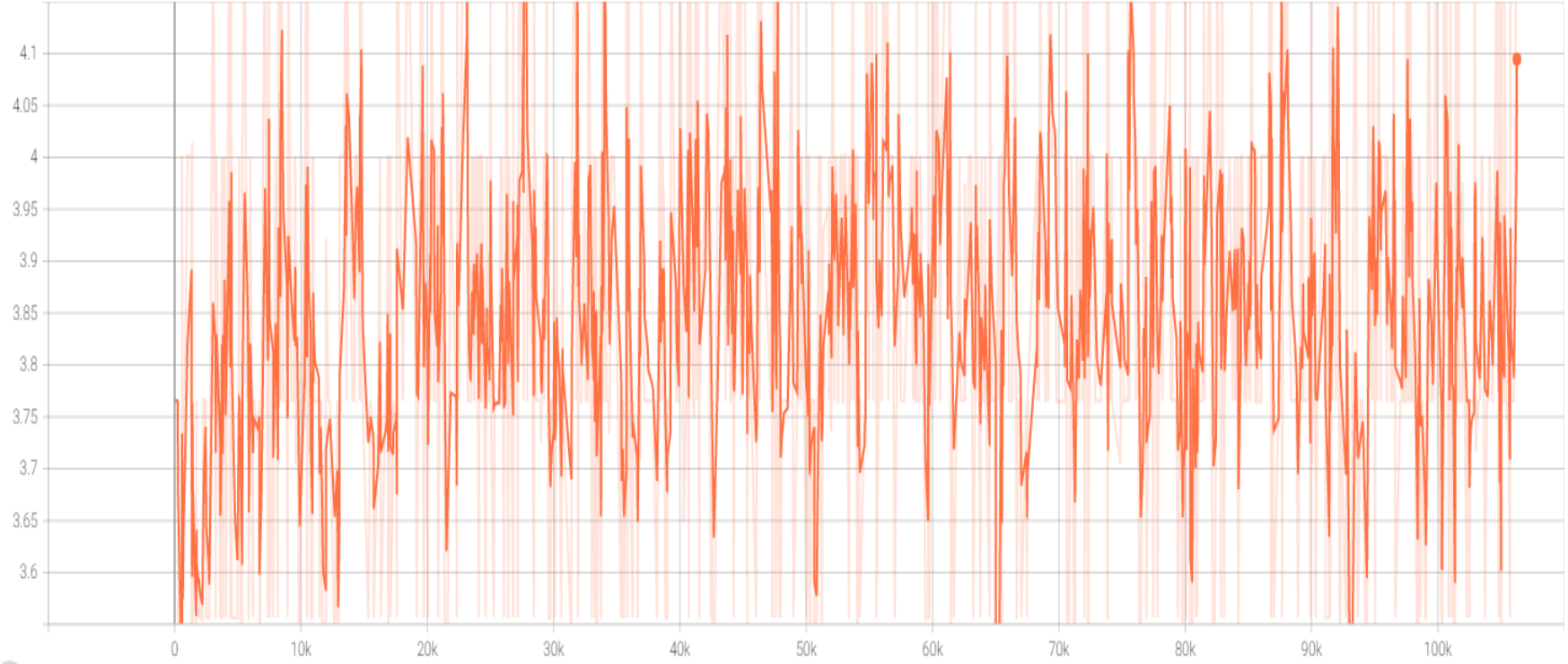
Şekil 4.6’da eğitim boyunca loss(kayıp) oranlarının değişim grafiği, Şekil 4.7’de ise eğitim süresindeki saniyedeki adım sayısı görünmektedir.

Loss/total_loss
tag: Loss/total_loss



Şekil 4.6. Eğitim süresinde loss(kayıp) oranları

steps_per_sec
tag: steps_per_sec



Şekil 4.7. Eğitim süresinde saniyedeki adım sayısı



5. SONUÇ

Görüntü işleme ve derin öğrenme yöntemleri ile yapılan kişi, nesne tanıma gibi çalışmalar artarak ve gelişerek devam etmektedir. Yapılan çalışmalarda cinsiyete göre sınıflandırma, yön tayinleri hatta yüz tanıma gibi birçok uygulama bulunmaktadır. Yüz tanımlama özellikle cep telefonları ve kişisel bilgisayar üzerinde özelleştirilmiş olarak uygulansa da ilerisi için umut ve heyecan verici gözükmemektedir. Sadece insan tanımlama ile kalmayıp bir adım ilerisi olarak otonomlaşan bir teknoloji de hızla gelişmektedir. Bu gelişmeler ile de başarılı çalışmalar yapıldığı gözlemlenmiştir. Performans ve hız kriterleri açısından da giderek manuel yöntemlere göre çok daha iyi sonuçlar verdiği görülmektedir.

Bu çalışmada, insan görüntülerinin elde edilen bilgilerin, işlenerek derin öğrenme tabanlı yön ve cinsiyet belirleme tespiti için bir yaklaşım sunulmuştur. Toplumsal hareketlilik içerisinde kayıt altına alınmış görüntü verileri hazır olarak elde edilmiştir. Çalışmanın amacında yer alan topluluk hareketlerin de bu tespitlerin yapılabilmesi için faydalı bir veri seti kullanılmıştır. Son zamanlarda derin öğrenme uygulamaların ciddi bir hız ve performans kazandıran Tensorflow kütüphanesi ve modelleme için başarılı bir algoritma olan CNN algoritması referans alınmıştır. Hızla gelişen teknoloji ve çalışmalar incelendiğinde CNN modelini tabanına dayanan büyük veriler üzerinde tespitler yapabilecek yeni modellerin çıkarıldığı görülmektedir. Bu bilgilerin referansında derin öğrenme modeli Fast R-CNN modeli üzerinden kurgulanmıştır. Eğitim sürecinin daha hızlı ve sağlıklı olması için de güçlü bir ekran kartı donanımının GPU'su kullanılmıştır. Kullanılan bu modelde önceden etiketlenmiş görüntü verileri 640x640 boyutlandırılarak eğitim için hazırlanmıştır. Eğitim için ayarlanan pipeline (veri hattı) batch size(parti boyutu) donanım birimine göre iki, adım sayısı yüz otuz bin ve sınıf sayısı sekiz olarak belirtilmiştir. Görüntü verilerinden oluşturulan TFRecord verileri de eğitim için ayarlanan pipeline (veri hattı) içerisinde tanımlanmıştır.

Yüz erkek resminde yapılan tespitlerde, erkek ve ön sınıfı için %76, erkek ve arka sınıfı için %68, erkek ve sol sınıfı için %48 erkek ve sağ sınıfı için %40 başarı oranları elde edilmiştir. Erkek ve yön tahmininde bu dört sınıfın ortalaması %58 başarı oranı ile tespit edilmiştir.

Yüz kadın resminde yapılan tespitlerde, kadın ve ön sınıfı için %92, kadın ve arka sınıfı için %72, kadın ve sol sınıfı için %44, kadın ve sağ sınıfı için %44 başarı oranları elde edilmiştir. Kadın ve yön tahmininde bu dört sınıfın ortalaması %63 başarı oranı ile tespit edilmiştir.

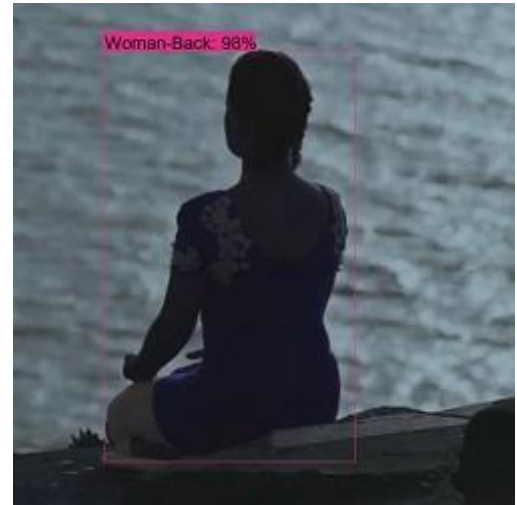
Genel tespit değerlendirmesine bakıldığında erkek cinsiyeti için %73, kadın cinsiyeti tespiti için %85 başarı oranları elde edilmiştir. Ön yönü için %92, arka yönü için %72, sol yönü için %58 ve arka yönü için %50 başarı oranları tespit edilmiştir.

Literatür araştırmasında görüldü ki insan verileri üzerinde hem cinsiyet tahmini hem de yön tespiti yapılan çalışma bulunmamaktadır. Tensorflow kütüphanesini kullanarak yapılan çalışmalarda ise genel olarak hazır veri seti ya da nesne algılama API'leri kullanılmaktadır. API kullanımları büyük veriler üzerinde geliştirilmiş modellerden oluşmaktadır ve tespit başarı oranları yüksektir. Hazır veri seti seçilmeyen çalışmalarda ise genel olarak iki sınıflandırmalı çalışmalar incelenmiştir. Bu çalışmada bir adım ileri düşünülerek insan tespit edildikten sonra yönünü ve cinsiyetini tahmin etme işlemi gerçekleştirilmiştir.

Bu çalışmada, kadın ve erkek cinsiyetlerini doğru tespit etme oranı, erkek için %73, kadın için %85 iken cinsiyet tahmininde doğruluk oranı ortalama %79 olarak hesaplanmıştır. Bu oranlar sırasıyla, YAA-AÖM ve CNN yöntemlerini kullanılarak elde edilen %80, %87 başarı oranlarına oldukça yakındır ve alternatif bir yaklaşımla yapılmıştır [7]. Ayrıca CNN modelinin tahmin sonucuna kıyasla uzak bir başarı oranı olmadığı kabul edilebilir. Bu çalışmada, kullanılan görüntü verileri önceki çalışmanın aksine sadece yüz görüntüleri değil genel olarak bütün beden ile seçilmiş görüntü verileri kullanılmıştır. Bu ise çalışmanın daha zor koşullarda çalıştığı anlamına gelmektedir. Ayrıca çalışmamda tek amaç cinsiyet analizi olmayıp, aynı zamanda yön tahminini de gerçekleştirebilmesi hedeflendiği için görüntüler bütün olarak seçilmiştir. Görüntü verilerinin seçimi göz önüne alındığı zaman, insan yüzü görüntülerine göre daha zor görüntüler üzerinde analiz yapılmıştır ve başarılı tespitler elde edilmiştir. Tespit oranı sonuçları yakın bu iki çalışmada, diğer çalışmaya ek olarak kişinin yön tahlili de elde edilmiştir.

Eğitim tamamlanıp yüz erkek ve yüz kadın görüntü verisi test edilmiştir. Çizelge 1.1' de karmaşıklık matrisi oluşturulmuştur. Sekiz sınıf üzerinde başarı oranları Çizelge

1.1’de yer alan karmaşıklık matrisi üzerinde gözükmetedir. Şekil 5.1’de test sonucunda başarılı sonuçların alındığı sınıflandırmalar gözükmetedir.





Şekil 5.1. Sınıflandırma sonuçlarına örnekler

Çizelge 1.1 : Karmaşıklık matrisi.

İnsan Cinsiyet ve Yön Tespiti	Erkek-Ön	Erkek- Arka	Erkek-Sol	Erkek-Sağ	Kadın-Ön	Kadın-Arka	Kadın-Sol	Kadın-Sağ	Doğruluk
Erkek-Ön	19	1	2	0	3	0	0	0	%76
Erkek-Arka	0	17	2	1	0	1	0	1	%68
Erkek-Sol	1	1	12	4	0	0	5	1	%48
Erkek-Sağ	0	0	5	10	1	1	5	3	%40
Kadın-Ön	1	0	0	0	23	0	1	0	%92
Kadın-Arka	0	0	1	1	2	18	2	1	%72
Kadın-Sol	1	0	1	0	0	0	11	10	%44
Kadın-Sağ	1	0	0	1	1	1	10	11	%44



6. ÖNERİLER

Özellikle ön ve arka yön tespitlerinde daha iyi bir başarı oranı elde edildiği, sağ ve sol içinde benzer başarıların elde edilebileceği gözlemlenmektedir. İnsan dönüş hareketlerinde sağ sol tespitinde ön ya da arka sınıflara benzerlik göstermeleri beklenmektedir. Sonraki çalışmalarda görüntü verilerinin bir takım görüntü filtreleri kullanılarak preprocessing (ön işleme) gerçekleştirilebilir. Yüksek çözünürlükte görüntü verileri kullanarak yön tespit etme yeteneğinin artışına katkı sağlanabilir. Eğitim sonucunda elde edilen modelin dışarı aktarımı ile mobil, Raspberry Pi gibi tek kart bilgisayarlar ya da farklı platformlar üzerinde uygulanabilir, kullanımı yaygınlaştırılabilir.

Bu çalışmada, yetişkinlik analizi doğrultusunda denemeler yapılmış ve kadın, erkek tespitlerinin dışında çocuk sınıfı da dahil edilmiştir. Ancak çocuk verileri kadın ve erkek sınıflarına benzerliğinden dolayı ve yetiştikine yakın yaşlardaki örneklerin doğru bilgi elde etmeye engel olduğu ile karşılaşmıştır. Yetişkinlik analizi için kadın, erkek ve çocuktan daha özel sınıfların oluşturulması daha olumlu sonuçlar oluşturabilir. Örneğin, Şekil 6.1’de yer alan iki çocuk verisi doğrudan çocuk sınıfı olarak değil de, çocuk ve ergen olarak ikiye ayrılabilir. Sadece çocuk sınıfı uygulandığı zaman ikinci görselde yer alan görüntünün erkek olarak sınıflandırılması çok mümkündür.



Şekil 6.1. Çocuk ve ergen kişi görüntü

Bölgesel farklılıklara bakılarak bu çalışmaya etnik kökenlere göre de ayrımlar yapılabilir. Asya, Afrika, Amerikan Indian, Australoid (Avusturalya), Polynesiani (Yeni Zelanda, Hawai ve Pasifik adaları) Doğu Avrupa, Akdeniz, Nordik (Kuzey Avrupa), Alpin (Orta Avrupa) ve Dinarik (Karpaslar ve Kafkaslar) gibi sınıflara ya da daha basite indirgenerek Amerikalı, Avrupalı, Asyalı ve Afrikalı olarak sınıflandırılabilir. Şekil 6.2’de insan çeşitliliğine örnek gösterilmektedir. Böylelik yapılan çalışma bir adım ileri götürölerek cinsiyet, yön tahminlerine artı olarak yetişkinlik seviyesi ve köken tahmini de hesaplabılır hale gelir.



Şekil 6.2. Fiziki insan çeşitliliği örneği

Kişilerin hangi yöne doğru ilerlediği, hangi cinsiyete sahip olduğu, hangi yaş aralıklarında olduğu gibi bilgiler özellikle ticari alanlarda fazlasıyla önem taşımaktadır. Derin öğrenme ile kişilerin tüketim ihtiyaçları belirlenerek bu ihtiyaca göre seçim önerilerin arttığı görölmektedir. Havaalanı, AVM, mağaza, restoran gibi insanların kişisel ihtiyaçlarını giderebilecekleri yerlerde bu bilgiler ile istatistiksel veriler çıkarabilmeleri kolaylaşabilir. Nasıl bir tedarik zincirinde, tedariğin nerelere, ne kadar kısa sürede ve ne kadar tasarruflu bir şekilde teminatı önemli ise, ihtiyaç sektöründe yer alan firmaların da müşteri analizleri bir o kadar önem arz etmektedir. Bu analizler ile sektörün hangi ürünlere talep ihtiyacının çok olduğu, hangi kitlelere daha çok hitap ettiği, hangi saat aralıklarında talebin artışta olduğu ya da hangi mağaza için ne kadar insan kaynağına ihtiyaç duyduğu vb. verilerine sahip olabilir. Şekil 6.3’de AVM girişi ve çıkışı sırasında çekilen görüntü verisi örneği mevcuttur. Kişi üzerindeki yapılan her analizler bu ve buna benzer birçok ihtiyaca cevap bulabilir.



Şekil 6.3. AVM giriş çıkış görüntüsü



KAYNAKLAR

- [1] Chenghua Li, Man Lin, Laurence T. Yang, (2014) Integrating the enriched feature with machine learning algorithms for human movement and fall detection, *The Journal of Supercomputing*. 67, 854-865.
- [2] Ng, C. B., Tay, Y. H., & Goi, B.-M., (2012) Recognizing Human Gender in *Computer Vision: A Survey*. PRICAI 2012: Trends in Artificial Intelligence, 335–346.
- [3] Mane, S., & Mangale, P. S. (2018) Moving Object Detection and Tracking Using Convolutional Neural Networks. *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, 14-15 June 2018.
- [4] Sai, B. N. K., & Sasikala, T., (2019) Object Detection and Count of Objects in Image using Tensor Flow Object Detection API, *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 27-29 Nov. 2019.
- [5] Nikouei, S. Y., Chen, Y., Song, S., Xu, R., Choi, B.-Y., & Faughnan, T. R., (2018) Real-Time Human Detection as an Edge Service Enabled by a Lightweight CNN, *2018 IEEE International Conference on Edge Computing*, 02-07 July 2018.
- [6] Bolme, D. S., Lui, Y. M., Draper, B. A., & Beveridge, J. R., (2009) Simple real-time human detection using a single correlation filter, *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 7-9 Dec. 2009.
- [7] Akbulut, Y., Sengur, A., & Ekici, S. (2017) Gender recognition from face images with deep learning. *2017 International Artificial Intelligence and Data Processing Symposium*, 16-17 Sept. 2017.
- [8] Gonzalez R., Woods R., (2008) Digital Image Processing, third ed., *Pearson Education*, New Jersey.
- [9] Çomak, B., Beycioğlu, A., Başığit, C., Kılınçarslan Ş. (2011) Beton teknolojisinde görüntü işleme tekniklerinin kullanımı. *In 6th International Advanced Technologies Symposium, IATS'11*, 16-18 May 2011, Elazığ, Turkey.

- [10] Samtař G., Gülesin M., (2012) Sayısal Görüntü İşleme ve Farklı Alanlardaki Uygulamaları. *Electronic Journal of Vocational Colleges*, 2, 85-97.
- [11] LeCun, Y., Bengio, Y., & Hinton, G. (2015) Deep learning. *Nature*, 521(7553), 436–444.
- [12] Rusk, N. (2016) Deep learning. *Nature Methods*, 13(1), 35–35.
- [13] Ataseven B., (2013) Yapay Sinir Ağları İle Öngörü Modellemesi. *Öneri Dergisi*, 39, 101-115.
- [14] Yavuz S., Deveci M., (2012) İstatistiksel Normalizasyon Tekniklerinin Yapay Sinir Ağı Performansına Etkisi. *Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 167-187.
- [15] Lo, S.-C. B., Chan, H.-P., Lin, J.-S., Li, H., Freedman, M. T., & Mun, S. K., (1995) Artificial convolution neural network for medical image pattern recognition, *Neural Networks*, 8(7-8), 1201–1214.
- [16] Tumen, V., Soylemez, O. F., & Ergen, B. (2017) Facial emotion recognition on a dataset using convolutional neural network. *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, 16-17 Sept. 2017.
- [17] İnik Ö., Ülker E. (2017) Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri. *Gaziosmanpaşa Bilimsel Araştırma Dergisi*, 6, 85-104.
- [18] Suarez-Paniagua, V., Segura-Bedmar, I. (2018) Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. *BMC Bioinformatics*, 19(S8).
- [19] Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M., (2020) Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, 128, 261-318.
- [20] LabelImg, <https://github.com/tzutalin/labelImg/>, 2020.
- [21] Tensorflow, <https://www.tensorflow.org/>, 2020.
- [22] Mpii human pose v1, <https://pose.mpi-inf.mpg.de>, 2020.

ÖZGEÇMİŞ

Ad Soyad: Hüsnü Yaşar

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

[1] Yasar H., Tepecik A., (2020) 6th World Conference on Computer Engineering, *Direction Determination and Adulthood Analysis with Deep Learning*. 30 Oct–01 Nov 2020, Istanbul, Turkey.