

El Yazısı ile Yazılmış Kimlik Numarası Tanıma Yöntemi

Anıl ÇELİK¹, Zeynep GÜRLER², Mehmet Kıvılcım KELEŞ^{3*}

Öz

El yazısı metin tanıma, çoğunlukla değişken yazı stilleri ve yazıların yer aldığı koşulsuz ortamlar nedeniyle hala zorlu bir problem olarak karşımıza çıkmaktadır. Literatürdeki çalışmalar, el yazısı örneklerin bulunduğu ortamların genellikle herhangi bir koşula sahip olmaması sebebiyle odaklandıkları el yazısı örneklerine ve ortamlarına göre özelleşmiştir. Bu yüzden farklı yazı tipleri ve ortamlar üzerinde uygulanabilirlikleri düşüktür. Önerilen çalışma, koşulsuz ortamlarda bulunan el yazısı Türkiye Cumhuriyeti Kimlik Numarası'nın (TCKN) tanınmasını hedeflemektedir. TCKN, her Türkiye vatandaşına verilen benzersiz bir kişisel kimlik numarasıdır. TCKN içerisindeki tek bir rakamı yanlış tahmin etmek bütün numaranın yanlış okunmasına yol açacağı için her rakamı doğru tahmin etmek oldukça önemlidir. Önerdiğimiz teknik, dokümanda TCKN yakalama, TCKN içerisinde rakam yakalama ve yakalanan rakamın sınıflandırılması olarak üç ana başlığa ayrılmaktadır. Rakam sınıflandırma aşamasında yakalanan rakamlar bir oto kodlayıcı yardımıyla taslak rakama dönüştürülürken oto kodlayıcıdan elde edilen özniteliklerle sınıflandırılmaktadırlar. Bu işlem, el yazısı rakamların sınıflarına ait en iyi temsile benzemesine çalışarak daha başarılı ayrışmalarını sağlamaktadır. Yapılan deneylerde oto kodlayıcı ile taslağa dönüştürme metodunun sınıflandırma başarımını önemli ölçüde artırdığı görülmüştür.

Anahtar Kelimeler: El yazısı, Türkiye Cumhuriyeti Kimlik Numarası, Oto kodlayıcı.

Handwritten ID Number Recognition Method

Abstract

Handwritten text recognition is still a challenging problem, mostly due to the variable writing styles and unconditional environments in which handwritten text appears. The studies in the literature are specialised to the handwriting samples and environments they focus on, as the environments in which handwritten samples are found are usually unconditioned. Therefore, their applicability to different handwriting types and environments is low. The proposed work aims to recognise the handwritten Turkish Republic Identity Number (TCKN) in unconditional environments. TCKN is a unique personal identification number given to every citizen of Turkey. It is very important to guess each digit correctly, since guessing a single digit in the TCKN will lead to misreading the whole number. Our proposed technique is divided into three main parts: capturing the TCKN in the document, capturing the digit in the TCKN and classifying the captured digit. In the digit classification stage, the captured digits are converted into draft digits with the help of an autoencoder and classified with the attributes obtained from the autoencoder. This process ensures more successful discrimination of handwritten digits by trying to resemble the best representation of their class. In the experiments, it has been observed that the autoencoder to draft conversion method significantly improves the classification performance.

Keywords: Handwritten, Turkish Republic Identity Number, Autoencoder.

^{1,2,3}Anadolu Sigorta, 34805, İstanbul, Türkiye,
acelik@anadolusigorta.com.tr zgurler@anadolusigorta.com.tr kkeles@anadolusigorta.com.tr

*Sorumlu Yazar/Corresponding Author

1. Giriş

El yazısı tanıma, günümüzde sıklıkla karşılaşılan ve üzerinde çalışmalar yapılan problemlerdendir. Özellikle formları işleme ve bilgi çıkarmada el yazısı anlamlandırma birçok projenin temel gereksinimi olmaktadır. Genel olarak formlar, koşullu ve koşulsuz ortam formları olmak üzere iki türde incelenebilmektedir. Bir form üzerinde el yazısı ile doldurulması beklenen bölge net hatlarla belirtilmiş ve farklı formlar arası değişkenlik göstermiyorsa ortam koşullu olarak tanımlanabilmektedir. Koşulsuz ortamlarda ise, el yazısı ile doldurulması beklenen bölge net hatlarla belirtilmemiş olup formlar arası değişkenlik göstermektedir. Bu durum, koşulsuz ortamlarda form üzerindeki konum gibi yapısal bilgilerin işlenmesini zorlaştırmaktadır. Koşullu ortamlarda form işleme çözülmesi mümkün olan bir problem olmakla beraber sektörde sıklıkla koşulsuz ortamlardaki problemlere çözüm aranmaktadır. Bu doğrultuda literatürde farklı ihtiyaçları karşılamaya yönelik koşullu ve koşulsuz ortamlarda çalışan birçok yaklaşım önerilmiştir.

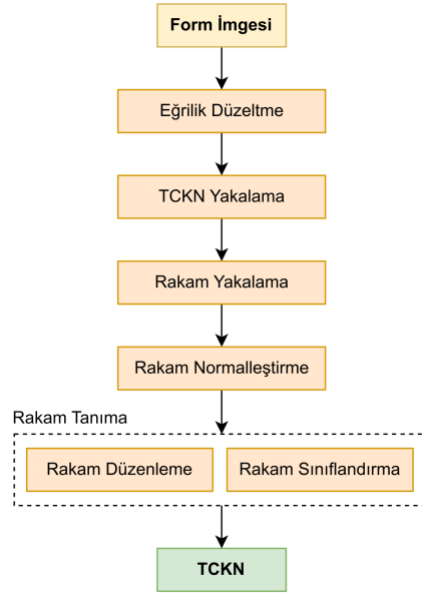
El yazısı tanıma, genel olarak kelime tanıma, banka çeki işleme, zarflardan posta adresi kontrolü, kimlik kartlarının ve posta kodlarının tanınması gibi görevlerde kullanılmaktadır (Neumann ve Matas, 2010; Neumann ve Matas, 2012; Busta ve ark, 2017). Yaklaşımlar, ağırlıklı olarak, el yazısını parça (Basu ve ark, 2009; Roy ve ark, 2016) ve bütün (Madhvanath ve Govindaraju, 2001; Barua ve ark, 2017) olarak işleyen şeklinde ayrılmaktadır. (Tamen ve ark, 2017), Arapça kelime tanıma için Chebyshev momentlerini istatistiksel ve kontur tabanlı özniteliklerle kullanmayı önermiş, Çok Katmanlı Algılayıcı, Destek Vektör Makinesi (DVM) ve Aşırı Öğrenme Makinesi gibi farklı sınıflandırıcıların bir kombinasyonu ile denemeler yapmıştır. (Sueiras ve ark, 2018), İngilizce kelime tanıma için Evrişimsel Sinir Ağı (ESA) ve uzun-kısa süreli bellek kullanan melez bir mimari ile imge sekanslarını metine dönüştüren bir yaklaşım önermiştir. (Disc Bhowmik ve ark, 2008), Bangla kelime tanıma için Genetik Algoritma ile eğitilmiş bir Gizli Markov Modeli kullanmıştır. (Barua ve ark, 2017), Bangla kelime tanınması için, girdilerin parçalarından çıkartılan gradyan tabanlı özniteliklerin, sıralı minimum optimizasyon sınıflandırıcısını önermiştir. (Boukharouba ve Bennia, 2017), Rakamların dikey ve yatay yönlerinden öne çıkan özniteliklerin Freeman Zincir Kodu ile birleştirildiği, DVM tabanlı bir yaklaşım önermiştir. (Mohebi ve Bagirov, 2014), özelleşmiş bir özdüzenleyici harita ile rakam tanınması için bir yaklaşım önermiş ve sonuçları benzer özdüzenleyici harita tabanlı yaklaşımlar ile karşılaştırmıştır. (Siddique ve ark, 2019; Ahlawat ve ark, 2020), Rakam tanıma için kendi özgün ESA yaklaşımlarını önermiştir. (Ghosh ve Maghari, 2017), Rakam tanıma için Derin Sinir Ağı, Derin İnanç Ağı ve ESA kullanarak, karşılaştırmalı bir çalışma gerçekleştirmiştir. (Hirata ve Takahashi, 2020), Önerdikleri ESA mimarisi içerisindeki son evrişimsel katmanı ayrıştırıp birden fazla tam bağlı ağa besleyerek topluluk öğrenmesi tabanlı bir sınıflandırıcı yaklaşım önermiş, sonuçlar kısmında rakam tanımaya da yer vermiştir. (Byerly ve ark, 2021), Rakam

tanıma için ayrı bir ESA mimarisini kapsül ağlarına bağlayan melez bir sınıflandırıcı önermiştir. (An ve ark, 2020), evrişim katmanlarında farklı çekirdek boyutlarına sahip üç farklı ESA modelinin topluluk öğrenmesi ile birleştirildiği bir mimari önermiştir. El yazısı tanıma için literatürde önerilen yaklaşımlar incelendiğinde, koşulsuz ortamlar için sunulan çözümlerin buldukları ortam içerisinde özelleşmiş oldukları gözlemlenmektedir. Bu durum, yaklaşımların yeni başlıklar altında uygulanabilirliğini düşürmektedir. Sunulan çalışmada koşulsuz ortamlardan elde edilen form imgeleri üzerinde el yazısı ile TCKN tanınması yapan yapay zekâ destekli çok kademeli bir yaklaşım önerilmiştir. TCKN tanınmanın diğer el yazısı tanıma çalışmalarından ayrıştığı temel fark, diğer çalışmalara oranla yanlış tanıma yapmanın maliyetinin yüksek olmasıdır. TCKN'nin tek bir basamağını yanlış tanımak, tüm tahminin yanlış olması ile sonuçlanacaktır. Sunulan çalışmada uygulanan adımlar modüler yapıya sahip olup gerektiğinde yeni başlıklara uygulanabilir niteliktedir.

Bu çalışmada önerilen yaklaşımda, ilk aşamada form imgesi üzerinde eğrilik düzeltme uygulanmaktadır. Devamında, iki kademeli olarak TCKN içeren bölge ve içerisinde bulunan rakamlar yakalanmaktadır. Ayrıştırılan rakamlar, konum olarak imgenin ortasına denk gelecek biçimde normalleştirilir. Son olarak rakam imgeleri daha önceden belirlenmiş taslaklara dönüştürülürler ve bu aşamada elde edilmiş olan öznitelikler yardımı ile sınıflandırılırlar. Çalışmanın geri kalanı şu şekilde düzenlenmiştir; önerilen yöntem'de, çalışmaya genel bir bakış sunulduktan sonra, yaklaşımdaki her aşama ayrıntılı kendi başlığı altında açıklanmıştır. Deneysel çalışmalar ve tartışma'da sonuçlar sunulup, yorumlanmış, son olarak gelecekte yapılması planlanan çalışmalar hakkında kısaca bilgi verilmiştir.

2. Materyal ve Metot

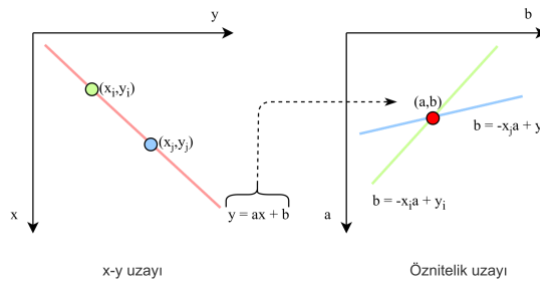
Önerilen yaklaşım, tanıma süreci için birbirini takip eden üç ana kademedен oluşmaktadır; TCKN yakalama, rakam yakalama ve rakam tanıma. Yaklaşımın ana akış diyagramı, Şekil 1'de verilmiştir.



Şekil 1. Önerilen Yaklaşımın Ana Akış Diyagramı

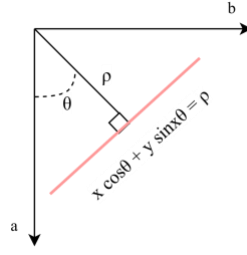
2.1. Eğrilik Düzeltme

Fiziksel formlar genellikle değişen derecelerde eğrilikle dijital ortama aktarılmaktadır. Eğriliğin ilerleyen aşamaların başarımını olumsuz etkilememesi adına önerilen yaklaşımda eğrilik düzeltme uygulanmaktadır. Düzeltmenin yapılabilmesi adına form üzerindeki çizgiler eğim açısının bulunmasında rehber seçilmiştir. Form üzerindeki çizgilerin tespiti Hough dönüşümü ile gerçekleştirilmektedir. Dönüşüm, bir çizgi üzerinde bulunan noktaları öznitelik uzayında kesişen çizgiler olarak temsil etmektedir. Öznitelik uzayında kesişen çizgiler, kesiştikleri noktanın eksen değerleri sayesinde geri dönüştürülebilir niteliğe sahiptir. Hough dönüşümü için kullanılan öznitelik uzayı Şekil 2’de verilmiştir.



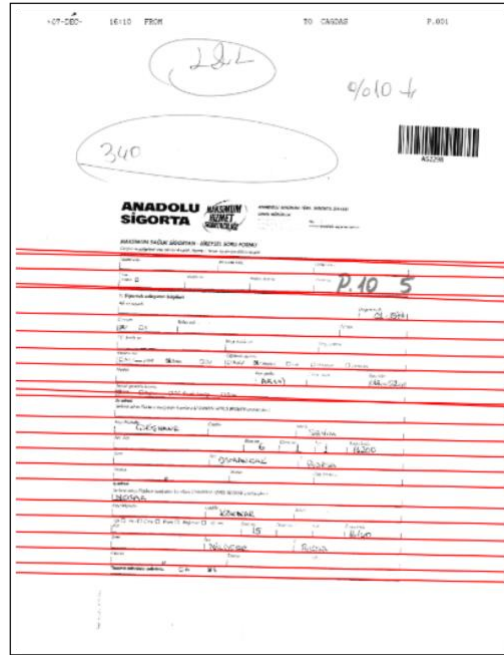
Şekil 2. Hough Dönüşümü için Kullanılan Öznitelik Uzayı

Hough dönüşümünde öznitelik uzayında aynı çizgiye ait noktalar daha kolay kümelenebilmeleri için trigonometrik fonksiyonlar ile temsil edilirler. Öznitelik uzayının trigonometrik fonksiyonlar ile temsili Şekil 3’te verilmiştir.



Şekil 3. Öznitelik Uzayının Trigonometrik Fonksiyonlar ile Temsili

Temsilde aynı çizgiye ait noktaların θ ve ρ değerlerinin yakın olması beklenir. Dönüşüm, benzer θ ve ρ değerlerini kümeleyerek aynı çizgiye ait noktaları tespit eder ve yakalamayı tamamlar. Eğrilik düzeltmede Hough dönüşümü uygulanmadan önce form çizgilerinin daha belirginleştirilmesi adına Canny'nin kenar yakalama yöntemi (Canny, 1986) kullanılmaktadır. Yöntem, siyah-beyaz biçimde beslenen bir imge için kenarları belirginleştirilmiş ikili bir çıktı oluşturur. Devamında, tespit edilen kenarlar üzerindeki çizgiler Hough dönüşümü ile tespit edilir. Son olarak, imge tespit edilen çizgiler arasında en sık geçen açının tersi yönünde döndürülür. Form imgesi üzerinde Hough dönüşümü ile yakalanan çizgiler Şekil 4'te verilmiştir.

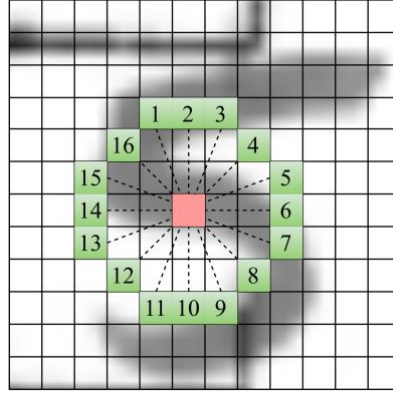


Şekil 4. Form İmgesi Üzerinde Hough Dönüşümü ile Yakalanan Çizgiler

2.2. TCKN Yakalama

Önerilen yaklaşımda ilk aşamada TCKN bölgesinin yakalanması için ORB (Oriented FAST and Rotated BRIEF) (Rublee ve ark, 2011) tabanlı bir yaklaşım denenmiştir. ORB, FAST (Features from Accelerated Segment Test) (Viswanathan, 2009) anahtar noktası yakalama ve BRIEF (Binary

Robust Independent Elementary Features) (Calonder ve ark, 2010) betimleme yaklaşımlarını iyileştirme amacı ile geliştirilmiştir. FAST, bir pikselin anahtar noktası olup olmadığına karar vermek için pikseli ortalayarak çevresine 16 piksel kapsayacak büyüklükte bir daire çizer ve etrafında toplanan piksellerin merkez piksele olan büyüklük-küçüklük ilişkisine bakar. FAST yaklaşımı için anahtar noktası seçme işlemi Şekil 5’te verilmiştir.



Şekil 5. FAST Yaklaşımı için Anahtar Noktası Seçme

FAST, aynı imgenin farklı ölçeklerinde başarılı sonuç vermekte zorlanmaktadır. ORB, FAST yaklaşımını bir imge piramidi ile farklı ölçeklerde gerçekleştirmektedir. Bu sayede çıkartılan anahtar noktalar ölçek değişimlerine karşı gürbüz olur. ORB yaklaşımında BRIEF, FAST tarafından yakalanan anahtar noktaların 128-512 bit arası ikili vektörler olarak kümelenmesi için kullanılmaktadır. FAST gibi ölçek değişimlerinin aksine, BRIEF eğim değiştiği zaman başarılı sonuç vermekte zorlanmaktadır. Bu dezavantajı gidermek adına ORB vektör çıkartma kademesine eğim parametresi eklemiştir. Bu sayede BRIEF vektörleri daha önceden belirlenmiş eğim kademeleri ile oluşturulur ve anahtar noktalarında tespit edilen değişime göre kullanılır.

Anahtar noktası eşleme tabanlı TCKN yakalama yöntemi, ilk aşamada hazır bir taslak form ile yaklaşıma beslenen girdi arasında ORB anahtar noktaları çıkartır. Devamında, iki imgedeki anahtar noktalar arasında mesafe olarak belirli bir eşliğin üstünde kalan noktalar yeterince tanımlayıcı olmadıkları için çıkartılırlar. Geriye kalan anahtar noktalar RANSAC (Fischler ve Bolles, 1981) algoritmasına beslenerek taslak ile girdi imge arasındaki homografi matrisi hesaplanır. Son olarak hesaplanan matris yardımı ile girdi imgeye perspektif dönüşümü uygulanır. Yöntemin çıktısında işlenmiş girdi ile taslak üzerindeki TCKN bölgelerinin aynı hizada olması beklenir. Örnek bir form imgesi üzerinde hizalama öncesi ve sonrası taslak ile örtüşen TCKN bölgeleri Şekil 6’da verilmiştir.

İşlenmemiş İmge

Hızalanmış İmge

Şekil 6. Form İmgesi Üzerinde Hizalama Öncesi(Kırmızı) ve Sonrası(Yeşil) Taslak ile Örtüşen TCKN Bölgeleri

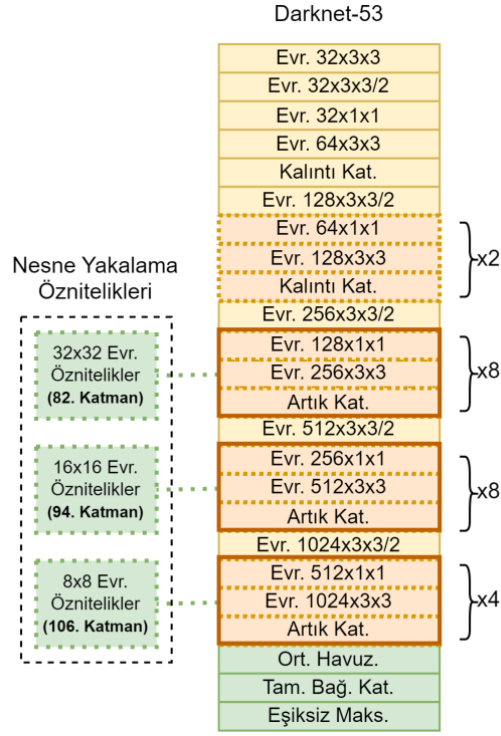
Anahtar noktası eşleme tabanlı yaklaşım koşullu ortamlarda uygulanabilir niteliktedir. Koşulsuz ortamlarda da gürbüz bir yaklaşım geliştirilmesi adına çalışmanın ilerleyen kademelerinde TCKN yakalama bir nesne yakalama problemi olarak tanımlanmış ve bu doğrultuda bir çözüm sunulmuştur. Nesne yakalama yöntemi olarak işlemsel gereksinim ve geriye dönük uyumluluk açısından uygun olan YOLOv3 (Redmon ve Farhadi, 2018) seçilmiştir. YOLO nesne yakalama yaklaşımı ailesi için Pascal VOC (Everingham ve ark, 2010) ve MS COCO (Lin ve ark, 2014) veri kümeleri üzerinde 0.5 eşik değeri için ortalama hassasiyet değerleri Tablo 1’de sunulmuştur.

Tablo 1. YOLO Yöntem Ailesinin Ortalama Hassasiyet Skorları

Yaklaşım	Veri Kümesi	mAP@0.5
YOLOv1	Pascal VOC	63.4
YOLOv2	Pascal VOC	78.6
YOLOv3	MS COCO	57.9
YOLOv4	MS COCO	65.7
PP YOLO	MS COCO	65.2
YOLOv5	MS COCO	68.9

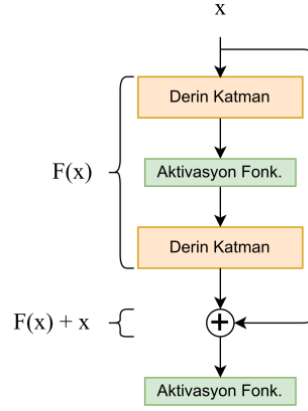
Yapılan ön çalışmada YOLOv3, TCKN ve rakam yakalama problemlerinde 0.5 eşik değeri için, 150 örnek üzerinde 0.99 ortalama hassasiyet skoru elde etmiştir. Yaklaşımın gösterdiği başarımlar sebebiyle YOLO ailesindeki diğer yöntemler değerlendirmeye alınmamıştır. YOLOv3, öznel olarak çıkartıcı olarak Darknet-53 (Redmon ve Farhadi, 2018) mimarisini kullanmaktadır. Darknet-53,

önceki kademesi olan Darknet-19'a (Redmon ve Farhadi, 2017) oranla daha derin bir yapıya sahip olup artık katmanları ile desteklenmiştir. Darknet-53 mimarisi Şekil 7'de sunulmuştur.



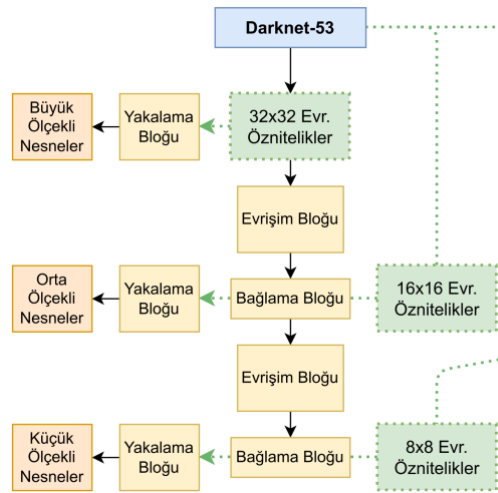
Şekil 7. Darknet-53 ve Mimariden Elde Edilen Öz nitelikler

Darknet-53'te bulunan artık katmanları ilk olarak ResNeT (He ve ark, 2016) içerisinde önerilmiştir. Artık katmanları, yapay sinir ağlarının derinleşmesi sonucu yaşanabilen kaybolan gradyan problemini azaltıcı niteliktedir. Yapay sinir ağlarında geri yayılım aşamasında ağırlıkların güncellenmesi zincir kuralına dayandırılarak gerçekleştirilmektedir. Ağ derinleştikçe uygulanması gereken türev sayısı da artmaktadır. Yapay sinir ağında kullanılan aktivasyon fonksiyonu türevi alındığı zaman beslenen değeri sıfıra yaklaştırma eğilimine sahip ise, geri yayılım sürecini dondurması muhtemel bir sonuçtur. Ağırlık güncellemesi yapamayan yapay sinir ağları için öğrenme süreci beklenilenden daha erken sonlanabilir. Artık katmanları atlama bağlantıları sayesinde geri yayılım aşamasında ağırlık güncellemesi mümkün olmayan katmanlar üzerinden bilgi akışı sağlayabilmektedir. Örnek bir kalıtsal katman üzerindeki temsili akış Şekil 8'de verilmiştir.



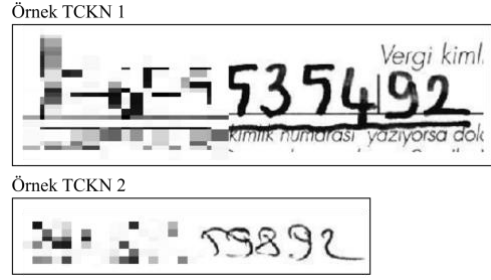
Şekil 8. Kalıtsal Katman Temsili Akış

Şekil 8’de sunulan kalıtsal katman temsilindeki $F(x)$ eşlemesi, geri yayılım aşamasında değer olarak 0 olsa bile bilgi akışı atlama bağlantısı sayesinde devam edebilecektir. YOLOv3 öznitelik piramit ağlarını (Lin ve ark, 2017) anımsatan çok ölçekli bir yakalama yöntemi ile çalışmaktadır. Mimaride imgeler Darknet-53 ağının farklı kademelerinden çıkartılan özniteliklere dönüştürüldükten sonra çok ölçekli nesne yakalama için işlenirler. Darknet-53, başlangıca yakın olan katmanlarda ince detayları işlerken sonlara doğru olan katmanlarda daha kaba hatlar üzerinde durmaktadır. Nesne yakalama bağlamında bu durum yakalamaların farklı ölçeklerde gerçekleştirilebilmesine olanak sağlamaktadır. YOLOv3, Darknet-53 ağından aldığı öznitelikleri 32×32 , 16×16 ve 8×8 evrişimden geçirdikten sonra üç farklı ölçekte nesne yakalamak için kullanmaktadır. Sadeleştirilmiş YOLOv3 mimarisi Şekil 9’da sunulmuştur.



Şekil 9. Sadeleştirilmiş YOLOv3 Mimarisi

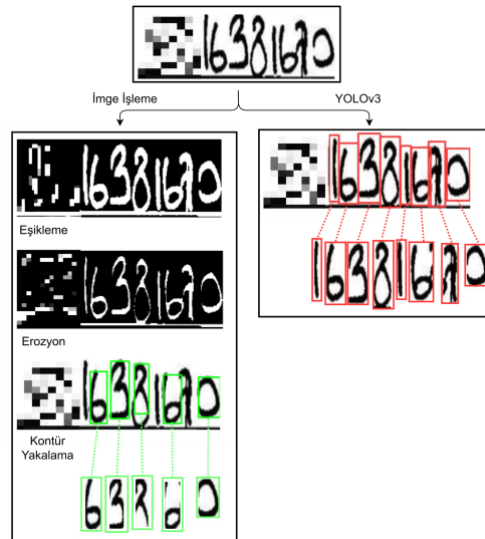
Çoklu ölçekte yakalama TCKN gibi aynı sınıf içerisinde değişken boyutlara sahip olabilen bir nesne için fayda sağlar niteliktedir. Yakalanan TCKN bölgeleri arasındaki oluşabilecek fark Şekil 10’da sunulmuştur.



Şekil 10. TCKN Bölgeleri Arasındaki Değişkenlik

2.3. Rakam Yakalama

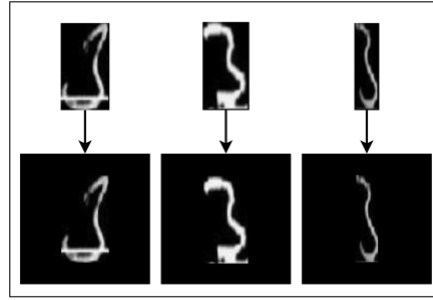
İmge üzerinde TCKN içeren bölge yakalandıktan sonra içerisindeki rakamların bulunduğu bölgelerin de yakalanması gerekmektedir. Önerilen yaklaşımda rakam yakalama için ilk aşamada imge işleme tabanlı bir yöntem denenmiştir. Yöntemde ilk olarak imge siyah-beyaz sonrasında ikili biçime dönüştürülür. Devamında ikili imge 3×3 boyutunda bir çekirdek ile iki kere erozyon operasyonundan geçirilir. Son olarak imge üzerindeki konturlar tespit edilip rakam boyutlarına yakın iç alana sahip olan dışında kalan alanlar filtrelenir. Geriye kalan kontur kümesinin imge içerisindeki rakamların konumları vermesi beklenir. İmge işleme tabanlı yöntem, morfolojik işlemler kullandığı için gereksiz bilgiler ile beraber faydalı olanları da silme eğilimindedir. Yöntem, TCKN yakalamada olduğu gibi çalışmanın ilerleyen aşamalarında nesne tanıma problemi olarak tanımlanmış ve YOLOv3 kullanılarak bu doğrultuda bir çözüm sunulmuştur. İmge işleme ve nesne yakalama tabanlı yöntemlerin rakam yakalama süreçleri Şekil 11’de gösterilmiştir.



Şekil 11. Rakam Yakalama için İmge İşleme ve Nesne Yakalama Karşılaştırması

2.4. Rakam Normalleştirme

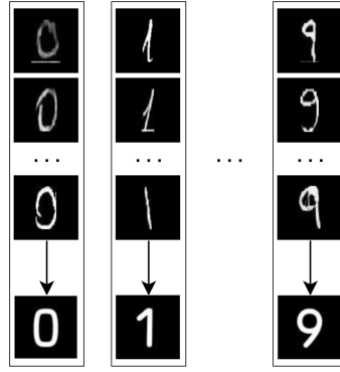
Rakam tanıma aşamasından önce TCKN bölgesi içerisindeki tüm rakamlar imgenin ortasına hizalanacak şekilde normalleştirilir. Bu yöntem sayesinde rakamların nesne yakalama işlemi sonrasında alabileceği farklı boyutların dezavantaj oluşturması engellenmiş olur. Yöntemde ilk olarak imge siyah-beyaz, sonrasında ikili biçime dönüştürülür. Devamında bitset-değil operatörü kullanılarak ikili imgenin piksel değerleri tersine çevrilir ve beyaz pikselleri kapsayan en küçük alan hesaplanır. Son olarak hesaplanan alan içerisinde kalan pikseller bir arka plan imgesi üzerine işlenir. Örnek rakamlar için normalleştirme sonuçları Şekil 12’de verilmiştir.



Şekil 12. Örnek Rakam Normalleştirme Sonuçları

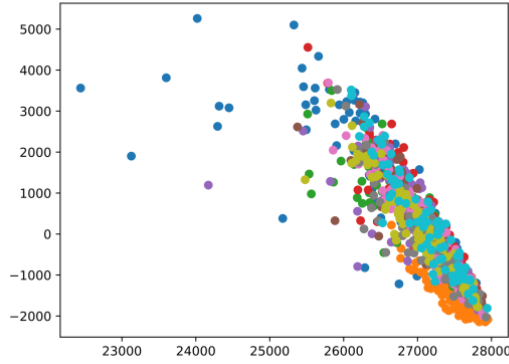
2.5. Rakam Tanıma

Önerilen yaklaşımda rakam tanıma, el yazısı rakamların belirli hatlara sahip bir taslak olarak düzenlenmesi ve düzenleme aşamasında kullanılan yapay sinir ağındaki öznelilikler kullanılarak sınıflandırılması olmak üzere iki aşamada gerçekleştirilmektedir. Rakam düzenleme aşamasındaki temel odak aynı rakama ait örneklerin birbirine benzeşip diğer rakamlardan uzaklaştığı bir biçime dönüştürülmesidir. Bu doğrultuda rakam sınıflandırma başarımını artırmak adına düzenleme kademesinde el yazısı rakamların özgün ve kolay ayrıştırılabilir nitelikte taslaklara dönüştürülmesi önerilmiştir. Taslak seçimi aşamasında hem insan hem de bilgisayar seviyesinde okuma kolaylığı amacı ile geliştirilen “OCR B” (Frutiger, 1980) yazı tipi kullanılmıştır. Örnek el yazısı rakamların taslakları ile eşleştirilmesi Şekil 13’te sunulmuştur.

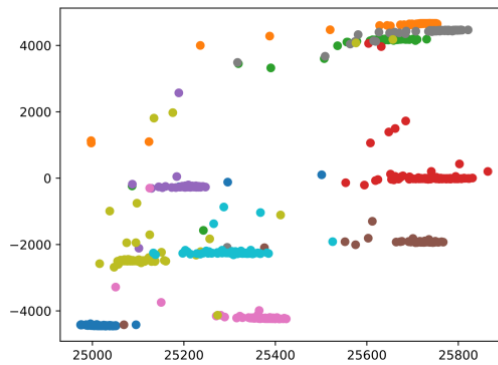


Şekil 13. Rakamların Düzenlenmesi Önerilen Taslaklar ile Eşleştirilmesi

El yazısı rakamlar ve düzenleme yapılması önerilen taslaklar aralarındaki ayrışmanın daha net gözlemlenebilmesi adına Tekil Değer Ayrışımı (TDA) ile temsil edilmiştir. Temsiller, Şekil 14 ve 15'te sunulmuştur.



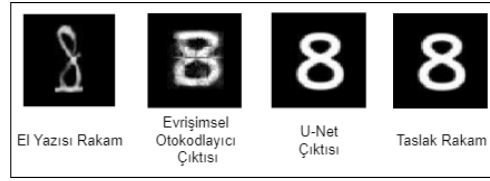
Şekil 14. El yazısı Rakamların TDA ile Temsili



Şekil 15. Taslak Rakamların TDA ile Temsili

TDA sonuçları incelendiğinde el yazısı rakamların birbirleri arasında benzerlik gösterdiği, taslakların ise ayrıştığı net bir şekilde gözlemlenebilmektedir. Yaklaşımında ana düzenleme ve sınıflandırma yöntemi U-Net (Ronneberger ve ark, 2015) olarak belirlenmiştir. İlk olarak biyomedikal görüntü bölütlemesi için geliştirilmiş olan U-Net mimarisi bir köprü ile birbirine

bağlanan dört kodlayıcı bloğu ve dört kod çözücü bloğundan oluşmaktadır. Mimari içerisindeki kodlayıcı ağ her kodlayıcı bloğundaki uzamsal boyutları yarıya indirirken öznitelik kanallarının sayısını iki katına çıkartır. Benzer şekilde kod çözücü ağ uzamsal boyutları ikiye katlar ve öznitelik kanallarının sayısını yarıya indirir. Kodlayıcı ağ, öznitelik çıkarıcı görevi görerek bir dizi kodlayıcı blok aracılığıyla beslenen imgeleri temsil eden öznitelikleri oluşturmayı öğrenir. Ek olarak evrimsel oto kodlayıcı ağlarında ters evrişim işleminin dezavantajlarından olan bulanıklık ve ızgara etkisine karşı gürbüzdür. Kodlayıcı ve kod çözücü blokları arası bilgi taşıyan köprüler kod çözme aşamasında imge hakkında farklı seviyedeki temsilleri kodlayıcı kademesinden aşamalı olarak alırlar. Bu sayede kod çözme sonuçları daha yüksek kalitede elde edilmiş olur. Örnek bir verinin farklı mimarilerle işlenmesi sonucu elde edilen çıktılar ve bu çıktılarla eşleşen taslak Şekil 16'da verilmiştir.



Şekil 16. Örnek Bir Verinin Farklı Mimarilerle İşlenmesi Sonucu Elde Edilen Çıktılar ve Bu Çıktılarla Eşleşen Taslak

Önerilen yaklaşımda kullanılan mimari için öznitelik çıkartıcı olarak ek bir ESA kullanılmamıştır. Üzerinde çalışılan veri kümesi küçük rakam imgelerinden oluştuğu ve imgelerden çıkartılabilecek detay seviyesinin sınırlı olmasından dolayı omurga mimarisi eklenmesi gerekli görülmemiştir. Kullanılan U-Net mimarisi taslağa dönüştürme işlemini gerçekleştirirken aynı zamanda beslenen rakamı sınıflandırmayı da öğrenecek şekilde tasarlanmıştır. Sınıflandırma, dönüştürme işlemi sırasında elde edilen özniteliklerin ayrı bir katmana beslenmesi ile gerçekleştirilmektedir. Bu doğrultuda, kullanılan U-Net ağındaki öznitelikler kodlayıcı blok sonuna eklenen tam-bağlı bir katmana beslenirler. Tam-bağlı katman, kendi kayıp fonksiyonuna sahip olup beslenen rakamları on boyutlu bir vektör çıktısına dönüştürmekle yükümlüdür. Bu sayede öznitelikler sınıflandırma başarımını da dikkate alarak oluşturulmaya teşvik edilmektedir. Kullanılan U-Net mimarisinde iki kayıp fonksiyonu bulunmaktadır; öznitelik üzerinden yapılan tahmin ile referans değer arasında çapraz-entropi ve girdi ile düzenlenmiş taslak imge arasındaki ortalama hata karesi.

Ek olarak, U-Net mimarisininin sınıflandırma kademesini bir ESA ile yapan alternatif hali de sunulmuştur. Kullanılan ESA, rakam imgelerinin boyutunun küçük olmasından dolayı sade olacak şekilde tasarlanmış olup, birbirini takip eden iki evrimsel katman, alt örnekleme, seyreltme ve yoğun katmanlardan oluşmaktadır. Alternatif mimari için kullanılan ESA Şekil 17'de, önerilen ve alternatif mimari ise Şekil 18'de sunulmuştur.

içerisinde istikrarlıdır. TCKN ve rakam yakalama için 1358 eğitim ve 238 test imgesi kullanılmıştır. Rakam sınıflandırma için kullanılan eğitim ve test imgesi sayıları Tablo 2’de sunulmuştur.

Tablo 2. Eğitim ve Test Aşamalarında Kullanılan Rakam Sayıları

Rakam	Eğitim	Test
0	1793	430
1	2048	493
2	2145	552
3	1759	421
4	2013	496
5	1536	356
6	1790	457
7	1399	370
8	1699	438
9	1299	335
Toplam	17481	4348

3.3. Değerlendirme Ölçütleri

TCKN tanıma, beklenti olarak 11 rakamın aynı anda doğru tahmin edilmesi odağında çalıştığı için rakam başına yapılan hata masrafı oldukça yüksek bir konudur. Bu bağlamda doğruluk (Eş. 1) değerleri ile beraber kesinlik (Eş. 2), duyarlılık (Eş. 3) ve F1 skoru (Eş.4) verilmiştir. Doğruluk, kesinlik, duyarlılık ve F1 skoru eşitlikleri için; DP, doğru pozitif tahmin sayısını, DN doğru negatif tahmin sayısını, YP yanlış pozitif tahmin sayısını, YN yanlış negatif tahmin sayısını ifade etmektedir.

$$Doğruluk = \frac{DP + DN}{DP + YP + DN + YN} \quad (1)$$

$$Kesinlik = \frac{DP}{DP + YP} \quad (2)$$

$$Duyarlılık = \frac{DP}{DP + YN} \quad (3)$$

$$F1 = 2 * \frac{Kesinlik * Duyarlılık}{Kesinlik + Duyarlılık} \quad (4)$$

Ek olarak rakam tanıma esnasında rakama özel yapılan yanlış tahminlerin çözümlenebilmesi adına hata matrisleri de sunulmuştur. Hata matrisi rakamlar arasında tereddüt yaşanan durumların net bir şekilde gözlemlenebilmesi adına fayda sağlamaktadır.

3.4. Deneysel Çalışmalar ve Tartışma

Deneysel çalışmalar Windows 10 işletim sistemi, 11. nesil i7 2.30 GHz işlemci, 32 GB RAM ile Python 3.9 ortamında gerçekleştirilmiştir. Çalışmalar TCKN içeren form imgeleri üzerinde gerçekleştirilmiş olup, sonuçlar hem bağımsız hem de 11 rakamın aynı anda doğru tahmin edilmesini kapsayacak şekilde hesaplanmıştır. Karşılaştırmalar önerilen U-Net mimarisi, alternatif hali ve sınıflandırma için kullanılan ESA arasında yapılmıştır. Yaklaşımlar iki hedefli öğrenme yönteminin rekabetçi yöntemler karşısındaki gürbüzlüğünü ölçmesi beklentisi ile seçilmiştir. Deneysel sonuçlar Tablo 3'te sunulmuştur.

Tablo 3. Deneysel Sonuçlar

Metotlar	0	1	2	3	4	5	6	7	8	9	Ortalama	11-Basamak	
Doğruluk	Ön.	%99.76	%100	%99.45	%98.3	%99.79	%99.43	%99.11	%100	%99.07	%99.39	%99.44	%93.4
	U-Net												
	Alt.	%99.29	%99.38	%99.81	%98.3	%99.5	%99.43	%99.33	%100	%98.37	%98.79	%99.26	%91.6
	ESA	%99.06	%99.38	%99.26	%97.56	%98.56	%97.12	%97.58	%98.07	%96.69	%96.6	%98.22	%84.8
Kesinlik	Ön.	%100	%99.19	%100	%99.28	%99.59	%97.52	%99.56	%99.73	%100	%99.1	%99.39	
	U-Net												
	Alt.	%99.07	%100	%99.27	%98.8	%99.79	%98.33	%99.78	%99.46	%99.3	%98.21	%99.2	-
	ESA	%98.38	%98	%98.2	%97.85	%99.39	%97.19	%98.9	%98.37	%98.14	%97.29	%98.17	
Duyarlılık	Ön.	%99.76	%100	%99.45	%98.33	%99.79	%99.43	%99.12	%100	%99.08	%99.4	%99.43	
	U-Net												
	Alt.	%99.3	%99.39	%99.81	%98.34	%99.59	%99.43	%99.34	%100	%98.4	%98.8	%99.24	-
	ESA	%99.07	%99.39	%98.91	%97.67	%98.58	%97.47	%98.68	%98.1	%96.8	%96.71	%97.83	
F1	Ön.	%99.88	%99.59	%99.72	%98.8	%99.69	%98.47	%99.34	%99.86	%99.54	%99.25	%99.41	
	U-Net												
	Alt.	%99.18	%99.69	%99.54	%98.57	%99.69	%98.88	%99.56	%99.73	%98.85	%98.51	%99.22	-
	ESA	%98.72	%98.69	%98.55	%97.74	%98.98	%97.33	%98.79	%98.24	%97.47	%97	%98.15	

Bağımsız rakam tabanlı sonuçlar incelendiğinde önerilen yaklaşımın ortalama doğruluk, kesinlik, duyarlılık ve F1 kulvarlarında rakiplerine oranla daha yüksek başarımlar gösterdiği gözlemlenebilmektedir. Alternatif yaklaşım yakalanan başarımlara yakın değerler elde etmiş, rakam başına elde edilen bağımsız doğruluk, kesinlik, duyarlılık ve F1 skoru kulvarlarında belirli zamanlarda öne çıkmıştır. Son olarak, ESA bağımsız rakam sonuçlarını ortalama %3 seviyesinde geride tamamlamıştır.

Önerilen yaklaşımın temel hedefi TCKN içerisinde bulunan 11 rakamın aynı anda doğru tespit edilmesidir, zira tek bir basamağın yanlış tespit edilmesi tahminin geçersiz sayılmasına sebep olacaktır. 11 basamağın aynı anda doğru tespit edilmesi kulvarında önerilen yaklaşım, %93.4 doğruluk elde etmiştir. Alternatif yaklaşım önerilen yaklaşımı %91.6 doğruluk ile takip etmiş, ESA ise %84.8 doğruluk ile bağımsız rakam seviyesinde yakaladığı başarımları bu kulvarda devam ettirememiştir. Tüm rakamların aynı anda doğru tahmin edilme isabeti bağımsız rakam başarımları ile doğrudan ilişkilidir. Rakam başına yaşanan yanlış tahminleme maliyeti TCKN tanıma başarımları artarak yansımaktadır. Bağımsız rakam tanıma başarımlarında önerilen yaklaşım ile rekabet edebilen

ESA, 11 basamak kademesinde geride kalmıştır. Genel olarak 11 basamak kademesinde sonuçlar incelendiğinde önerilen yaklaşımın rekabetçiler karşısında tercih edilmesinin belirgin bir dezavantajı olmadığı gözlemlenebilmektedir.

Referans Değer	0	429	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	493	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	3	549	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	414	1	3	1	0	0	2	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	495	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	0	354	1	0	0	0	1	0	0	0	0	0	0	0	0
	6	0	1	0	0	1	2	453	0	0	0	0	0	0	0	0	0	0	0	0
	7	0	0	0	0	0	0	0	0	0	0	370	0	0	0	0	0	0	0	0
	8	0	0	0	1	0	2	0	1	434	0	0	0	0	0	0	0	0	0	0
	9	0	0	0	1	0	1	0	0	0	0	0	333	0	0	0	0	0	0	0
	0	1	2	3	4	5	6	7	8	9										
Tahminlenen Değer																				
Önerilen U-Net için Hata Matrisi																				

Referans Değer	0	427	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	490	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	2	1	0	551	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	414	1	2	1	0	2	1	0	0	0	0	0	0	0	0	0
	4	1	0	0	0	494	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	0	354	0	0	0	0	2	0	0	0	0	0	0	0	0
	6	1	0	0	0	0	0	454	0	1	1	0	0	0	0	0	0	0	0	0
	7	0	0	0	0	0	0	0	0	0	370	0	0	0	0	0	0	0	0	0
	8	1	0	1	1	0	1	0	1	431	2	0	0	0	0	0	0	0	0	0
	9	0	0	1	2	0	1	0	0	0	0	331	0	0	0	0	0	0	0	0
	0	1	2	3	4	5	6	7	8	9										
Tahminlenen Değer																				
Alternatif U-Net için Hata Matrisi																				

Referans Değer	0	426	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	490	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	2	3	2	546	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	3	0	0	3	411	1	1	1	1	1	1	2	1	0	0	0	0	0	0	0
	4	0	1	0	0	489	2	1	3	0	0	0	0	0	0	0	0	0	0	0
	5	1	1	1	1	1	0	347	1	0	1	3	0	0	0	0	0	0	0	0
	6	2	1	1	0	2	0	451	0	0	0	0	0	0	0	0	0	0	0	0
	7	0	4	1	2	0	0	0	0	0	363	0	0	0	0	0	0	0	0	0
	8	0	1	1	2	0	5	0	1	424	4	0	0	0	0	0	0	0	0	0
	9	1	0	1	3	0	1	0	0	5	324	0	0	0	0	0	0	0	0	0
	0	1	2	3	4	5	6	7	8	9										
Tahminlenen Değer																				
Evrışimli Sınır Ağı için Hata Matrisi																				

Şekil 19. Hata Matrisleri

Şekil 19’da sunulan hata matrisleri incelendiğinde tereddüt yaşanan rakam sınıflandırma sayısının tüm yaklaşımlar için genel olarak düşük olduğu görülmektedir. Görsel olarak birbirine benzeyen rakamlar arasında karmaşa yaşanması beklenirken hata matrisleri farklı bir tablo çizmektedir. Örneğin önerilen yaklaşım için orijinal değeri 2 olan örneklerin 1 olarak yanlış tahminlendiği gösterilmiştir. Benzer bir durum ESA’da 4 ile 7 rakamları arasında da mevcuttur. Önerilen yaklaşım ile ESA’nın benzer durumlarda karmaşa yaşadığı görülmektedir. Bu durum mimari olarak farklı olmalarına rağmen iki yaklaşımın belirli aşamalarda benzer temsiller öğrendiği olarak yorumlanabilir. Alternatif U-Net mimarisine bakıldığında diğer iki yaklaşımdan farklı durumlarda tereddüt yaşadığı görülmektedir. Değişiklik, mimari olarak önerilen yaklaşıma yakın olmasına rağmen sınıflandırma seviyesindeki ayrışmanın oluşturulan temsili etkilemesi olarak yorumlanabilir.

4. Sonuçlar ve Öneriler

Bu çalışmada sigorta form imgeleri üzerinde uçtan uca TCKN tanıma gerçekleştiren bir yaklaşım önerilmiştir. Önerilen yaklaşım TCKN tanıma sürecini parçalara bölerek ele almış olup modüler niteliğe sahiptir. Çalışma U-Net ağını oto kodlayıcı niteliğinde kullanmış olup hem köprü bağlantılarından faydalanmış, hem de öznetelik seviyesinde temsilleri işleme dahil etmiştir. Düzenleme ve sınıflandırmadan oluşan iki uçlu öğrenme hedefi rakam tanıma başarımını olumlu şekilde etkilemiştir. Koşulsuz ortamda TCKN tanıma problemi için önerilen yaklaşım 11 rakamın aynı anda doğru tahminlemesi kademesinde %93.4 doğruluk elde etmiştir. Ayrıca önerilen

yaklaşımın alternatifini olarak bir yöntem daha sunulmuş, TCKN tanıma problemi üzerinde karşılaştırmaları yapılmıştır. Devamında önerilen mimari ile ESA'nın benzer konularda tereddüte düştüğü tespit edilmiş, mimari olarak farklı olmalarına rağmen yaklaşımların benzer temsiller oluşturdukları gözlemlenmiştir. Önerilen yaklaşım çoklu el yazısı rakam tanıma için yeniden uygulanabilir olup adımlar tekrarlandığında farklı başlıklara çözüm sağlayabilecek niteliktedir. Gelecekteki çalışmalarda yaklaşımın modüler niteliğini bozmadan çoklu rakam tanıma kademesinin imgenin bir rakam dizisine doğrudan dönüştürülerek gerçekleştirilmesi planlanmaktadır.

Yazarların Katkısı

Tüm yazarlar çalışmaya eşit katkıda bulunmuştur.

Çıkar Çatışması Beyanı

Yazarlar arasında herhangi bir çıkar çatışması bulunmamaktadır.

Araştırma ve Yayın Etiği Beyanı

Yapılan çalışmada araştırma ve yayın etiğine uyulmuştur.

Kaynaklar

- Ahluwat S., Choudhary A., Nayyar A., Singh S., Yoon B. (2020). Improved handwritten digit recognition using convolutional neural networks (CNN), *Sensors*, 20(12):3344.
- An S., Lee M., Park S., Yang H., So J. (2020). An ensemble of simple convolutional neural network models for MNIST digit recognition, *arXiv preprint arXiv:2008.10400*.
- Basu S., Das N., Sarkar R., Kundu M., Nasipuri M., Basu DK. (2009). A hierarchical approach to recognition of handwritten Bangla characters, *Pattern Recognition*, 42(7):1467-84.
- Barua S., Malakar S., Bhowmik S., Sarkar R., Nasipuri M., Bangla (2017). Handwritten city name recognition using gradient-based feature, *InProceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications: FICTA, Volume 1*, 343-352.
- Boukharouba A., Bennia A. (2017). Novel feature extraction technique for the recognition of handwritten digits, *Applied Computing and Informatics*, 13(1):19-26.
- Busta M., Neumann L., Matas J. (2017). Deep textspotter: An end-to-end trainable scene text localization and recognition framework, *InProceedings of the IEEE international conference on computer vision*, 2204-2212.
- Byerly A., Kalganova T., Dear I. (2021). No routing needed between capsules, *Neurocomputing*, 463:545-53.
- Canny J. (1986). A computational approach to edge detection, *IEEE Transactions on pattern analysis and machine intelligence*, 679-98.
- Calonder M., Lepetit V., Strecha C., Fua P. (2010). Brief: Binary robust independent elementary features, *In Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, Proceedings, Part IV 11 2010*, 778-792.
- Disc Bhowmik TK., Parui SK., Roy U. (2008). Discriminative HMM training with GA for handwritten word recognition, *In 2008 19th International Conference on Pattern Recognition*.

- Everingham M., Van Gool L., Williams C. K., Winn J., Zisserman A. (2010). The pascal visual object classes (voc) challenge, *International journal of computer vision*, 88:303-38.
- Fischler M. A., Bolles R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, 24(6):381-95.
- Frutiger A. (1980). *Type. Sign. Symbol*. ABC Verlag, Zurich, 50.
- Ghosh M. M., Maghari A. Y. (2017). A comparative study on handwriting digit recognition using neural networks, In *2017 international conference on promising electronic technologies (ICPET)*, pp. 77-81.
- He K., Zhang X., Ren S., Sun J. (2016). Deep residual learning for image recognition, In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778.
- Hirata D., Takahashi N. (2020). Ensemble learning in CNN augmented with fully connected subnetworks, *arXiv preprint arXiv:2003.08562*.
- Lin T. Y., Maire M., Belongie S., Hays J., Perona P., Ramanan D., Dollár P., Zitnick C.L. (2014). Microsoft coco: Common objects in context, In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, Proceedings, Part V 13 2014*, 740-755.
- Lin T. Y., Dollár P., Girshick R., He K., Hariharan B., Belongie S. (2017). Feature pyramid networks for object detection, In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117-2125.
- Madhvanath S., Govindaraju V. (2001). The role of holistic paradigms in handwritten word recognition, *IEEE transactions on pattern analysis and machine intelligence*, 23(2):149-64.
- Mohebi E., Bagirov A. (2014). A convolutional recursive modified self organizing map for handwritten digits recognition, *Neural Networks*, 60, 104-118.
- Neumann L., Matas J. A. (2010). method for text localization and recognition in real-world images, In *Computer Vision–ACCV 2010: 10th Asian Conference on Computer Vision, Queenstown, New Zealand, 8-12, Part III 10*, 770-783.
- Neumann L., Matas J. (2012). Real-time scene text localization and recognition, In *IEEE conference on computer vision and pattern recognition*, 3538-3545.
- Redmon J., Farhadi A. (2017). YOLO9000: better, faster, stronger, In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263-7271.
- Redmon J., Farhadi A. (2018) Yolov3: An incremental improvement, *arXiv preprint arXiv:1804.02767*.
- Ronneberger O., Fischer P., Brox T. (2015). U-net: Convolutional networks for biomedical image segmentation, In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, Proceedings, Part III 18 2015*, 234-241.
- Roy PP., Bhunia AK., Das A., Dey P., Pal U. (2016). HMM-based Indic handwritten word recognition using zone segmentation, *Pattern recognition*, 60:1057-75.
- Rublee E., Rabaud V., Konolige K., Bradski G. (2011). ORB: An efficient alternative to SIFT or SURF, In *2011 International conference on computer vision*, 2564-2571.
- Siddique F., Sakib S., Siddique M. A. (2019). Recognition of handwritten digit using convolutional neural network in python with tensorflow and comparison of performance for various hidden layers, In *2019 5th international conference on advances in electrical engineering (ICAEE)*, 541-546.
- Sueiras J., Ruiz V., Sanchez A., Velez JF. (2018). Offline continuous handwriting recognition using sequence to sequence neural networks, *Neurocomputing*, 289:119-28.
- Tamen Z., Drias H., Boughaci D. (2017). An efficient multiple classifier system for Arabic handwritten words recognition, *Pattern Recognition Letters*, 93:123-32.
- Viswanathan D. G. (2009). Features from accelerated segment test (fast), In *Proceedings of the 10th workshop on image analysis for multimedia interactive services, London, UK, 6-8*.