KASTAMONU ÜNİVERSİTESİ

MÜHENDİSLİK MİMARLIK FAKÜLTESİ

Bilgisayar Mühendisliği

**DERS:** Görüntü İşleme

**KONU:** Karakter Tanıma

**Hazırlayanlar**

Nesrin KAYA (134410018)

Kübra KUKARA (134410016)

**İÇİNDEKİLER**

* GİRİŞ
* YÖNTEMLER

Kelime tanıma teknikleri

- Karakter tabanlı kelime tanıma

- Parçalama tabanlı kelime tanıma

- Kelime şeklini tanıma

Karakter tanıma teknikleri

-Ön işlemler

- Değişmez momentler ve özellik çıkarma

- Sınıflandırma

Karakter tanıma yöntemleri

-Ağırlık merkezine dayalı tanıma

- Korelasyon

- Kontur çıkarma

- Tekleştirme modeline dayalı tanıma

Yapay sinir ağları ile karakter tanıma

* UYGULAMA

YPS ile karakter tanıma örneği

* KAYNAKÇA

**GİRİŞ**

Karakter tanıma problemi, en genel anlatımla, bir takım semboller içeren bir belgeyi bilgisayara görüntü olarak aktarma ve belgeyi oluşturan karakterleri ön işleme vasıtasıyla tespit ettikten sonra önceden bilinen yada tanınan karakterlerle eşleştirme işlemi olarak tanımlanabilir . Karakter tanıma, son yirmi yıl boyunca üzerinde çokça çalışma yapılmış bir araştırma alanıdır. Karakter tanımanın amacı, insanoğlunun okuma özelliğini, insana göre çok daha hızlı çalışan bilgisayara taklit ettirmek olarak belirtilebilir.

Diğer bir tanımlama ile, elle veya makine ile yazılmış yazıların bilgisayar tarafından tanınmasına çok genel terimiyle OCR (Optical Character Recognition, Optik Karakter Tanıma) adı verilmektedir . OCR teknolojisi dünyada pek çok alanda otomasyon amacıyla kullanılmaktadır ve bu konuda çok çalışmalar yapılmıştır . Uygulama alanları arasında bankalarda çeklerin otomatik olarak işlenmesi, postanelerde zarf üzerindeki adreslerin otomatik olarak tanınarak mektupların gidecekleri bölgelere ayrılmaları ve tarayıcılarda bulunur.

Üzeri satırlarca yazı dolu bir kağıt, okuma bilmeyen bir çocuk için bir şey ifade etmediği gibi, bilgisayarlar için de değersiz bir resimden ibarettir. Bilgisayarın, bu resmin içindeki bilgiyi kullanabilmesi için yorumlaması gerekmektedir. Başka bir değişle, bu resimdeki yazıların bilgisayarın anlayacağı, daha kolay ve verimli saklayıp, bulabileceği sembollere çevrilmesi gerekmektedir. Bunun için de kağıt üzerindeki bir metnin doküman işleme denilen bir dizi işlemden geçmesi gerekir. Bu işlemi gerçekleştiren sistemlere Optik Karakter Tanıma (Optical Charecter Recognition-ORC) sistemleri denilir ve birçok ticari OCR sistemi bulunmaktadır. Bu sistemlerin başarısı işlenen dokümandaki karakterlere bağlıdır. Basılı karakterlerde başar oranı yüksekken, düşük kalitede baskılı ve el yazması karakterlerde başarı oranı düşüktür.

Karakter tanıma, öncelikle sinyal ve görüntü işleme algoritmalarıyla arka planın ayrıştırılması, görüntüde var olan gürültünün temizlenmesi gibi işlemler gerektirir. Daha sonra bölümleme adı verilen yazı birimlerinin birbirinden ayrılması işlemleri yapılır. Bu noktada sıkça kullanılan bir yöntem, histogram tabanlı bölümlemedir. Çünkü beyaz bir zemin üzerine siyah ile yazılmış olan yazı karakterlerin ayrıştırılması için seçilecek tek bir eşik değer bölümleme için yeterli olacaktır.

Momentler ve momentlerin polinom fonksiyonları iki boyutlu örüntü tanıma uygulamalarında oldukça önemlidir. Momentler, sınıflandırma için özellik çıkarmada, doku (texture) analizinde ve nesnelerin şekillerinin belirlenmesinde kullanılır. Bu çalışmada, 1960’ların başında Hu’nun , matematiksel moment teorisinden geliştirdiği yedi değişmez momentle Türkçe karakterlerin tanınması amaçlanmıştır. Yapılan bilgisayar benzetimi sonuçlarından, önerilen Türkçe karakter tanıma sisteminin performansının iyi olduğu görülmüştür.

**YÖNTEMLER**

Kelime tanıma teknikleri

Kelime modelleri farklı yollarla gösterilmektedir. En açık yöntem, kurallı kelimelerin bir listesini saklamaktır. Diğer yöntemler n-gramlar (kurallı harf dizilimleri), birinci ve ikinci dereceden harf geçiş olasılıklarını ve n- gram olasılıklarını yakalayan Markov Modelleridir. Karakter tanımada kelime modellerini kullanabilmek için üç farklı yaklaşım tanımlanabilir. Bu yaklaşımlar;

* Karakter tabanlı kelime tanıma

Bu üç aşamalı bir yaklaşımdır. Önce kelime görüntüsü karakter görüntülerine parçalanır. İkinci aşamada, parçalanan karakterler bir karakter tanıma tekniği kullanılarak tanınır. Bağlamsal sonradan işleme (Contextual Postprocessing-CPP) denilen üçüncü aşamada ise sonuç kelimesi beklenilen kelimelerin bir modeli kullanılarak düzeltilir. Örneğin “kulak” kelimesinin “u” karakterini karakter tanıma tekniği “u” mu “v” mi ayırt etmeyebilir. Bir CPP tekniği, “u” harfinin doğru olduğunu belirler, çünkü kulak kelimesi, sözlükte bulunması daha muhtemel bir kelimedir. CPP teknikleri, kelime modellerini ifadelerinde farklılık gösterirler. Sözcükten yararlananlar, tanınan kelime ile sözlükteki her kelime arasındaki uzaklığı ölçmek için kelime dizisi eşleme teknikleri kullanmaktadırlar.

* Parçalama tabanlı kelime tanıma

Bu iki aşamalı bir yaklaşımdır. Bağlamsal bilgi, bireysel karakter görüntülerini tanıma sürecinde kullanılmaktadır. Burada da, ilk aşamada, kelime görüntüsü bireysel karakter görüntülerine parçalanır. Tanımanın ikinci aşamasında, her karakter görüntüsü için özellikler çıkarılır. Kelimelere sınıflama tüm bu karakterlerin özellikleri bütün olarak kullanılarak gerçekleştirilir.

* Kelime şeklini tanıma

Bu kelime görüntüsü karakterlere parçalama kısmını atlayan tek aşamalı bir yaklaşımdır. Özellikler parçalanmamış kelime görüntüsünden elde edilir. Tanıma problemi, verilen kelime görüntüsünü, sözlükteki bir veya daha fazla kelimeye eşleştirmektir.

Tanımayı gerçeklemek için, sözlükteki kelimeler beklenilen kelime özellikleri ile ifade edilir. Böyle kelime özellikleri, beklenilen yazı tipi kümesine dayanmaktedır.

Kelime şeklinin analizi. Hipotez oluşturma ve test etme olarak iki aşamada gerçeklenebilir. Hipotez oluşturma aşamasında , basit bir özellik kümesi kullanılarak, giriş kelimesine benzeyen kelimelerden bir komşuluk oluşturulur. İkinci aşamada, böyle kelimeleri ayırt edebilecek özellikler çıkarılır.

Anlaşıldığı gibi bir karakter dizisinin tanınması her şeyden önce karakterlerin birbirinden iyi bir şekilde dilimlenmesine bağlıdır.

**Karakter tanıma teknikleri**

Geliştirilen Türkçe karakter tanıma sistemi temel olarak üç alt birimden oluşmaktadır. Sistem, Şekil 1’de blok diyagram olarak verilmiştir.



**Şekil 1.** Türkçe karakter tanıma yapısı

Sistemin girişine uygulanan Türkçe karakterler Arial fontunda olup 90x85 boyutundadır. Sistemin ilk birimini oluşturan önişlemler, Türkçe karakterlerin bulunduğu görüntüleri bir dizi görüntü işleme tekniğinden geçirerek, özellik çıkarmadan önce bütün görüntülerin aynı formata girmesini sağlamaktadır. Özellik çıkarma alt birimi, görüntülerden Türkçe karakterleri belirleyecek parametrelerin elde edilmesini ve özellik vektörünün oluşturulmasını sağlamaktadır. Sınıflandırıcı alt birimi ise özellik vektörünü giriş alarak bir çıkış üretmektedir.

* **Ön işlemler**

Örüntü tanıma sistemlerinde, kullanılan bütün önişleme tekniklerinin amacı, benzer örüntülerden benzer anahtar özelliklerin elde edilebilmesi için örüntüyü istenmeyen bileşenlerinden arındırmaktır. Kullanılan bütün görüntüler JPEG formatında olup,90x85 boyutundadır. Çalışmada kullanılan önişlemler ve kullanılma amacı şöyledir; RGB formatında bulunan görüntüler ilk önce gri derinlikli forma dönüştürülmektedir. Görüntülerin bilgisayar ortamına alınması sırasında oluşan gürültülerin giderilmesinde komşuluk piksel değerleri kullanılır. Bu işlem filtreleme olarak adlandırılır ve görüntü işleme uygulamalarında sıklıkla kullanılır. Medyan filtreleme, kenarları koruyarak gürültüleri elimine edebilen nonlineer bir filtredir . Filtreleme, kayan bir pencerenin (3x3’lük bir matris) görüntüdeki her bir pikseli ve o piksele komşu olan diğer piksellerin değerlerini sıralandıktan sonra seçilen medyan değerinin, pencere merkezinde bulunan piksel değeri ile değiştirilmesi işlemidir. Şekil 2’de filtrelenmiş görüntü görülmektedir.

|  |  |
| --- | --- |
| (a) | (b) |

**Şekil 2.** Türkçe karakter a) Gürültülü b) Filtrelenmiş

İmgede bulunan nesnenin arka plandan ayırt edilmesi için kullanılacak olan bölümleme tekniği histogram tabanlıdır. Histogram tabanlı bölümleme işlemi, görüntü işleme uygulamalarında önemli bir önişlemdir. Bölümleme, görüntünün bileşenlerine ayrıştırılması işlemi olarak tanımlanır ve farklı bölümleme yöntemleri literatürde mevcuttur . Burada, görüntüde sadece Türkçe karakter olduğundan histogram eşiklemesi ile Türkçe karakterler arka plandan kolayca alınabilir. Şekil 3’de filtrelenmiş görüntü ve eşiklenmiş hali görülmektedir. Bu aşamadan sonra karakterlere morfolojik inceltme işlemi uygulanabilir, literatürde bu türden uygulamalar mevcuttur [9]. Bu çalışmada herhangi bir inceltme işlemi uygulanmamıştır. Elde edilen görüntülerin son hali, anahtar özellik çıkarımı için kullanılmıştır.

|  |  |
| --- | --- |
| (a) | (b) |

**Şekil 3.** Türkçe karakter a) Filtrelenmiş b) Bölütlenmiş görüntü

* **Değişmez Momentler ve Özellik Çıkarma**

Momentler, uzun zamandır istatistik teorisinde ve klasik mekanikte kullanılmaktadırlar . İstatistikçiler moment olarak ortalama, varyans, çarpıklık oranı ve kurtosis kullanırlarken, mekanikçiler momentleri ağırlık merkezi bulmada ve atalet momenti hesaplamada kullanırlar. 1960’ların başında Hu, matematiksel moment teorisinden yedi değişmez moment geliştirdi. Bu yedi moment, ölçeklendirme, döndürme ve dönüşüm işlemleri altında değişmemekte ve karakter tanıma gibi iki boyutlu örüntü tanıma problemlerinin çözümünde gözle görülür başarı sağlamaktadır. Böylece karakter tanıma, nesne tanıma, radar görüntülerinden uçak ve gemi tanıma gibi uygulamalar ortaya çıkmıştır .

Klasik mekanikte tanımlanan momentlere benzer olarak iki boyutlu (p+q) dereceli ve f(x,y) dağılımlı bir görüntü için moment şöyle tanımlanır;



burada p,q = 0, 1, 2,……değerlerini alabilir. Bu momentler genelde gürültülere bağlı olarak değişkendirler. Bunun sonucunda merkez momentleri tanımlanmıştır ve aşağıdaki gibi ifade edilir;



burada;



olarak tanımlanır. Merkez momentleri, dönüşüm altında değişmezlerdir. Bazen momentlerin boyutlara bağlı olarak normalizasyonu istenebilir. Bu işlem μ00 moment değeri ile gerçekleştirilebilir. Böylece normalize merkez momenti;



olarak tanımlanır. Burada  dir.

Hu, normalize merkez momentlerini kullanarak yedi adet değişmez moment geliştirmiştir. Hu’nun geliştirdiği momentler şöyledir;



|  |  |  |  |
| --- | --- | --- | --- |
| (a) | (b) | (c) | (d) |

**Şekil 4.** Momentleri hesaplanan Türkçe karakter

Tablo 1’de, Şekil 4’deki her bir karakter için hesaplanan moment değerleri sırası ile görülmektedir. Ön işlemlerden geçirilen görüntüler, kendilerini karakterize edecek momentlerin hesaplanmasından sonra yapay sinir ağı sınıflandırıcısının girişlerine verilmiştir. Yapay Sinir Ağı (YSA) girişine uygulanan moment değerleri oldukça küçük olduğundan bu değerlerin logaritmaları alınmıştır.

**Tablo 1.** Moment değerleri

|  |  |
| --- | --- |
| a | 1.1275 3.5427 8.0562 10.3677 19.6363 12.2448 21.9169 |
| b | 1.1275 3.5427 8.0562 10.3677 19.6363 12.2448 21.9169 |
| c | 1.1275 3.5427 8.0562 10.3677 19.6363 12.2448 21.9169 |
| d | 1.3729 5.5033 7.9705 12.1408 23.0868 15.8025 22.8975 |

* **Sınıflandırma**

Örüntü sınıflandırıcılarını geleneksel ve akıllı olarak gruplara ayırmak mümkündür. Geleneksel sınıflandırma algoritmaları istatistiksel bir yapı olan Bayes karar teorisi üzerine kuruludur. Bunların dezavantajları, özellik uzayını sınıflandırma uzayına dönüştürürken, bir gürültünün çıkması ve her bir sınıf için hata kriterinin belli olmamasıdır. Geleneksel sınıflandırıcılara; çok değişkenli Gauss modelleri, en yakın komşu, maksimum olabilirlik, ikili ağaç sınıflandırıcıları ve Fisher’in doğrusal sınıflandırıcıları örnek olarak verilebilir . Buna karşın akıllı sınıflandırma yapıları genellikle YSA tabanlı olup, günümüzde en yaygın kullanılan ve başarımını ispatlamış çok güçlü sınıflandırıcı türleridirler. Özellikle de genelleme yetenekleri, örüntü tanıma uygulamalarının çok büyük boyutlu verileri açısından önemlidir. Bu çalışmada kullanılan sinir ağı modeli ve parametreleri Tablo 2.’de görülmektedir. Tablo 1’de verilen moment değerlerinden birinci sıra eğitim için diğerleri ise test için kullanılmıştır.

**Tablo 2.** Kullanılan YSA modeli

|  |  |  |
| --- | --- | --- |
| **Yapay Sinir Ağı Yapısı** | | |
| YSA modeli | | Çok Katmanlı İleri Beslemeli |
| Katman sayısı | | 3 |
| Katmanlardaki hücre sayısı | | 20-8-3 |
| Başlangıç ağırlıkları ve eşik değerler | | Rasgele |
| Etkinleştirme fonksiyonu | | Tanjant sigmoid – Lineer |
| **YSA Eğitim Parametreleri** | | |
| Öğrenme Kuralı | Levenberg-Marquardt hata geri yayılım | |
| Hedeflenen toplam karesel hata | 10-6 | |

Karakter tanıma yöntemleri

* Ağırlık merkezine dayalı tanıma

Karakterlerin ağırlık merkezlerinin farklı noktalarda olduğu düşüncesinden ortaya çıkmıştır. Ağırlık merkezine dayalı sınıflandırma sabit karakter boyutlarında iyi sonuçlar vermektedir.

Ağırlık merkezi kriteri her ne kadar ayırt edici özellik taşıyorsa da, karakterlerin büyüklükleri değiştiğinde ağırlık merkezleri de kaymaktadır. Bu kaymanın sebebi, büyütme küçültme işlemlerinin piksel bazında yapılmasıdır. Aynı yazı tipinin farklı büyüklüklerinde ağırlık merkezi kaymaktadır.bu nedenle farklı özellik vektörlerine ihtiyaç vardır.

* Korelasyon

Karakter tanıma alanında ilk yaklaşımlardan biri basit optik şablon eşlemedir. Korelasyon karakter tanıma literatüründe sıkça kullanılan bir terimdir. Korelasyon en basit anlamda iki ikili örüntü arasındaki hamming uzaklığının belirlenmesi olarak açıklanabilir.

* Kontur çıkarma

Bu algoritma çerçevelenmiş karakterlerin 4 yönlü kesit eğrilerinin yapılarının incelenmesine dayanmaktadır. Karakter tanıma işlemleri 4 yön izdüşümü bilgilerinin karşılaştırılması ile gerçekleştirilmektedir.

Aramayı hızlandırmak amacı ile her yönde elde edilen uzaklık bilgileri önce toplanarak toplam izdüşüm değerleri elde edilir. Aynı değerli birkaç karaktere rastlandığında ise oluşturulmuş yön bilgilerini içeren 4 vektörün farkları değerlendirilir. Her yönde normalize edilmiş karakterlerin izdüşüm farklarının toplamı alınarak minimum olanı muhtemel sonuç olarak verilir.

* Tekleştirme modeline dayalı tanıma

Bu yöntemde, karakterlerin iskeletleri modellenmeye çalışılmaktadır. Bu modelleme bir çeşit kontur çıkarmadır. Yalnız bu kontur görüntüdeki dikey ve yatay yönden ayrı ayrı sıkıştırılan değişimleri ifade etmektedir. Sonuç olarak bu yöntem, diğer yöntemlerde kullanılan inceltme, ölçekleme problemleri ile karşı karşıya değildir. Yalnızca parlaklık değişimleri tutulduğundan bellek kullanımı ve sınıflama süresi klasik yöntemlere göre daha azdır.

Yapay sinir ağları ile karakter tanıma

Yapay sinir ağı nedir?

* insan beyninin çalışma ilkelerinden ilham alınarak geliştirilmiş, ağırlıklı bağlantılar denilen tek yönlü iletişim kanalları vasıtası ile birbirleriyle haberleşen, her biri kendi hafızasına sahip birçok işlem elemanından (nöronlardan) oluşan paralel ve dağıtık bilgi işleme yapılarıdır.
* YSA’lar gerçek dünyaya ait ilişkileri tanıyabilir, sınıflandırma, kestirim ve işlev uydurma gibi görevleri yerine getirebilirler.

Yapay sinir ağları 2 yöntemle gerçekleştirilir.

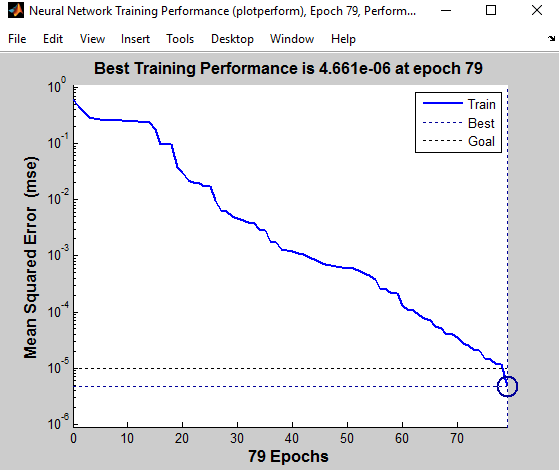
* Kod yazarak yapay sinir ağı oluşturma.
* Araç kutusu yardımı ile yapay sinir ağ oluşturma.

**UYGULAMA**

**YPS ile karakter tanıma örneği:**

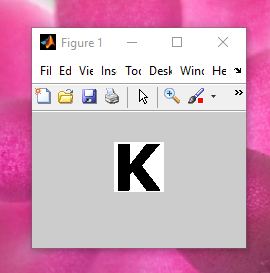
**Öncelikle directory e harflerimizi atıyoruz (K-R)**

* K=im2bw(K); %vectördeki değerleri 0 ve 1 arasında değişen değerlere çeviriyor
* K=imresize(K,[50,50]); % yeniden boyutlandırdık
* K=double(K); %double yaptık
* K=reshape(K,2500,1); %sütun matrisine çevirdik
* R=im2bw(R);
* R=imresize(R,[50,50]);
* R=double(R);
* R=reshape(R,2500,1)
* giris=zeros(2500,2); % sıfırlar matrisi oluşturduk.
* giris(:,1)=K; %giriş matrisine K harfini atıyoruz (1.sutun)
* giris(:,2)=R; %giriş matrisine K harfini atıyoruz (2.sutun)
* target=eye(2); %birim matris oluşturur
* net=newff(minmax(giris),[10 2],{'logsig' 'logsig'},'trainscg'); %agı oluşturuyoruz
* net.trainParam.perf='sse'; % (performan fonk. Hata karelerinin toplamı)
* net.trainParam.epochs=500; % dongu sayısı
* net.trainParam.goal=1e-5; %amaç
* net=train(net,giris,target); %agı eğitiyoruz

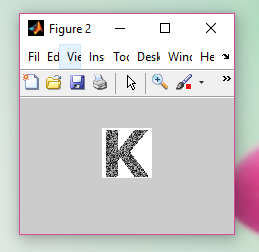
 Ağı egittik.

Şimdi K harfimizi tekrar okutup biraz gürültü ekleyip sonucumuza bakalım.

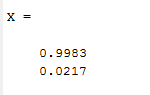
* K=im2bw(K);
* K=imresize(K,[50,50]);
* imshow(K);



* K=K+rand(50,50)\*0.8; %gürültü ekledik
* İmshow(K);



* K=reshape(K,2500,1);
* X=sim(net,K);

 Tarzı bir sonuç bekleriz. 1.sütun 1’e yakın 2.sütun ise 0’a yakın olmalı

**%dosya olusturma**

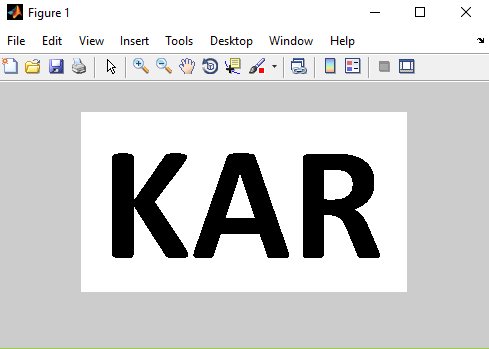
* dosyayeri='karakterler\';
* dosyaturu='.jpg';
* icerik=dir([dosyayeri,'\*',dosyaturu]);
* rsayisi=size(icerik,1);
* A=zeros(2500,rsayisi); %A giriş matrisi oluşturduk.
* for k=1:rsayisi % bu döngü ile sırayla dosya içerisindeki resimleri alıp yeniden boyutlandırma sütun matrisine çevirme işlemleri yapıyor
* string=[dosyayeri,icerik(k,1).name];
* resim=imread(string);
* resim=im2bw(resim);
* resim=imresize(resim,[50,50]);
* resim=double(resim);
* resim=reshape(resim,2500,1);
* A(:,k)=resim;
* end

**%ağın oluşturulması**

* target=eye(k);
* giris=[A];
* [R,Q]=size(giris);
* [s2,Q]=size(target);
* s1=10;
* net=newff(minmax(giris),[s1,s2], {'logsig' 'logsig'},'trainscg');
* net.trainParam.perf='sse';
* net.trainParam.epochs=500;
* net.trainParam.goal=1e-5;
* net=train(net,giris,target);

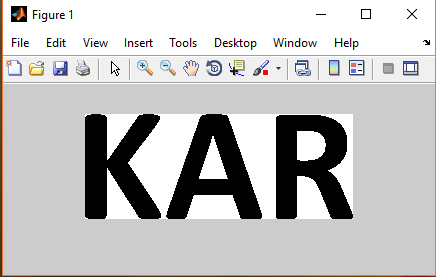
%yazının bulunduğu resmi okutuyoruz

* res=imread('KAR.png');
* res=im2bw(res);
* figure; imshow(res);



%yazıda sıfır olan alanları kırpıyoruz

* [y x]=find(res==0); %resim içinde 0 olan yerleri bulur.
* y1=min(y);
* y2=max(y);
* x1=min(x);
* x2=max(x);
* yenires=imcrop(res,[x1 y1 x2-x1 y2-y1]); %koordinatlara göre kırpma işlemi yapar
* figure; imshow(yenires);



%belirli özelliklere göre harflerin ağa giriş olarak verilmesi

* harfsayisi=round((x2-x1)/80); %burdaki 80 yazıda karakterlerin yaklasık pikseli
* for m=1:harfsayisi
* harf=imcrop(res,[x1+90\*(m-1) y1 90 y2-y1]); %burdaki 90 da diger karaktere geçmek için gerekli piksel sayısı
* harf1=imresize(harf,[50 50]);
* harf2=reshape(harf1,2500,1);
* harf2=double(harf2);
* X=sim(net,harf2);
* sonuc=find(X==max(X));
* if sonuc==1
* fprintf('A',m);
* else if sonuc==2
* fprintf('K',m);
* else if sonuc==3
* fprintf('R',m);
* end
* end
* end
* end



KAYNAKÇA

* <https://www.youtube.com/watch?v=eSkslzGfV-4>
* <https://www.google.com.tr/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwjE4J6zn_bQAhXH7hoKHSHhCoUQFggZMAA&url=http%3A%2F%2Fyzgrafik.ege.edu.tr%2F~ugur%2F09_10_Fall%2FDIP%2FKARAKTER_TANIMA_AU.pdf&usg=AFQjCNEbpQBAG-p4uutZNG5m-2b0I2Yqqg&sig2=Aw4jnqJX592BUX-76G9TEA&bvm=bv.141536425,d.bGg>
* <https://www.researchgate.net/publication/238113628_YAPAY_SINIR_AGLARI_ILE_TURKCE_TIMES_NEW_ROMAN_ARIAL_VE_ELYAZISI_KARAKTERLERI_TANIMA>
* Doğu Anadolu bölgesi araştırmaları 2004

Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü-ELAZIĞ (Abdulkadir ŞENGÜR -İbrahim TÜRKOĞLU)

DEĞİŞMEZ MOMENTLERLE TÜRKÇE KARAKTER TANIMA

* http://www.cedar.buffalo.edu/hwai/hwai\_home.html
* DOÇ.DR. Vasif V. NABİYEV YAPAY ZEKA (Seçkin yayıncılık)