

T.C.
SÜLEYMAN DEMİREL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EL YAZISI KARAKTERLERİNİN KAPALI CEBİRSEL
EĞRİLERLE MODELLENMESİ VE SINIFLANDIRILMASI

İhsan PENÇE

Danışman: Doç. Dr. Bayram CETİŞLİ

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
ISPARTA-2012

TEZ ONAYI

İhsan PENÇE tarafından hazırlanan “El Yazısı Karakterlerinin Kapalı Cebirsel Eğrilerle Modellenmesi ve Sınıflandırılması” adlı tez çalışması aşağıdaki jüri tarafından oy birliği ile Süleyman Demirel Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı’nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. Bayram CETİŞLİ

Süleyman Demirel Üniversitesi Bilgisayar Donanımı Anabilim dalı

Jüri Üyeleri :

Doç. Dr. Bayram CETİŞLİ

Süleyman Demirel Üniversitesi Bilgisayar Donanımı Anabilim dalı

Yrd. Doç. Dr. Ecir Uğur KÜÇÜKSİLLE

Süleyman Demirel Üniversitesi Bilgisayar Yazılımı Anabilim dalı

Doç. Dr. Tuncay YİĞİT

Süleyman Demirel Üniversitesi Bilgisayar Yazılımı Anabilim dalı

Prof. Dr. Mehmet Cengiz KAYACAN

Enstitü Müdürü

Not: Bu tezde kullanılan özgün ve başka kaynaktan yapılan bildirişlerin, çizelge, şekil ve fotoğrafların kaynak gösterilmeden kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunundaki Bölümlere tabidir.

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER	i
ÖZET	iii
ABSTRACT	iv
TEŞEKKÜR.....	v
ŞEKİLLER DİZİNİ.....	vi
ÇİZELGELER DİZİNİ	vii
SİMGELER DİZİNİ.....	viii
1. GİRİŞ.....	1
2. KAYNAK ÖZETLERİ.....	5
3. MATERYAL VE YÖNTEM	9
3.1. Ön İşlemden Geçirme.....	9
3.1.1. Euclidean dönüşümler	9
3.1.1.1.Öteleme	10
3.1.2. Affine dönüşümler.....	11
3.1.2.1.Ölçekleme	12
3.1.3. Sınır bölgesinin belirlenmesi.....	13
3.1.4. Karakterin normalize edilmesi	13
3.2. Modelleme.....	15
3.2.1. Genetik algoritmalar.....	16
3.2.2. Parçacık sürü optimizasyonu.....	18
3.2.3. Kapalı cebirsel eğriler	20
3.2.4. Kümeleme tabanlı global optimizasyon	22
3.3. Öznitelik Seçimi ve Sınıflandırma	23
3.3.1. Öznitelik seçimi.....	23
3.3.2. Sınıflandırma.....	23
3.3.2.1.Bayes sınıflandırıcı.....	24
3.3.2.2.Yapay sinir ağları	26
3.3.2.3.Uyarlamalı sinir-bulanık sınıflayıcı	30
4. ARAŞTIRMA BULGULARI VE TARTIŞMA.....	34
4.1. Kapalı Cebirsel Eğriler Kullanılarak Yapılan Çalışmalar.....	34

4.2.	Signomial Kuvvetlerle Diğer Yöntemler Kullanılarak Yapılan Çalışmalar ..	43
4.2.1.	Genetik algoritma sonuçları	44
4.2.2.	Parçacık sürü optimizasyonu sonuçları	45
4.2.3.	Kümeleme tabanlı global optimizasyon sonuçları	46
4.3.	Elde Edilen Katsayıların Yorumlanması	48
5.	SONUÇ	50
	KAYNAKLAR	51
	ÖZGEÇMİŞ	58

ÖZET

YÜKSEK LİSANS TEZİ

EL YAZISI KARAKTERLERİNİN KAPALI CEBİRSEL EĞRİLERLE MODELLENMESİ VE SINIFLANDIRILMASI

İhsan PENÇE

Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Bayram CETİŞLİ

Bu çalışmada el yazısı karakterlerinin uygun cebirsel eğriler ile modellenip denklem şeklinde ifade edilmesi ve elde edilen denklemin kuvvet ve katsayıları kullanılarak karakterlerin sınıflandırılması amaçlanmıştır. Bu sebeple çalışmada çeşitli eğri uydurma yöntemleri ele alınmıştır. Bunlar arasında hız ve başarı oranı göz önünde tutulduğunda kapalı cebirsel eğrilerle modellemenin diğer yöntemlere göre daha ön plana çıktığı görülmüştür. Sınıflandırma işlemi için el yazısı rakamların sekizinci dereceden cebirsel denklemleri elde edilerek denklem katsayıları öznitelik olarak kullanılmıştır. Elde edilen katsayıların değişmez olabilmesi için sadece ölçekleme ve ötelemeye göre normalizasyonu yapılmıştır. Ayrıca dilsel kuvvetli sinir bulanık sınıflayıcı ile öznitelik seçimi yapılmıştır. Çalışmada MNIST el yazısı rakam veri tabanının tamamı ve belli bir kısmı kullanılarak Bayes ve yapay sinir ağları ile önerilen yöntemin tanıma başarısı ölçülmüştür. Çalışmada elde edilen %92.87 değerindeki tanıma oranı umut vaat edip yöntemin geliştirilerek her karakterin bir denklemle ifade edilebileceği açıktır. Böylece karakterleri görüntü biçiminde saklamak yerine katsayılarla saklayarak daha az bellek kullanımı sağlanabilir.

Gelecekte, veri transferinde görüntü yerine transfer edilen katsayılarla veri iletim hızı arttırılabilir. Yeni ve daha verimli bir bilgi ya da görüntü sıkıştırma yöntemi, cebirsel eğri yöntemleri önerilebilir.

Anahtar Kelimeler: El yazısı, karakter tanıma, kapalı cebirsel eğriler, modelleme, Bayes, yapay sinir ağları, uyarlamalı sinir-bulanık sınıflayıcı.

2012, 58 sayfa

ABSTRACT

Master Sc. Thesis

HANDWRITING CHARACTER MODELING WITH IMPLICIT CURVES AND CLASSIFICATION

İhsan PENÇE

**Süleyman Demirel University
Graduate School of Natural Sciences
Department of Computer Engineering**

Supervisor: Assoc. Prof. Dr. Bayram CETİŞLİ

In this study, the modeling of handwriting characters by using the appropriate algebraic curves were aimed, and expressed as an equation. Also, the coefficients and exponents of obtained equations were used for classification of handwritten digits. Therefore, a variety of curve fitting methods were tested in the study. When the speed and the success rate of methods were considered, it was observed that the process of modeling with implicit curves is better than other methods. Coefficients of the equation were used as attribute by obtaining eighth degree algebraic equations of the handwritten digits for the classification process. To be invariant, the normalization of the obtained coefficients was made according to only scaling and translation. Feature selection was also done with neuro-fuzzy classifier with linguistic hedges. In the study, the recognition rate of the method proposed with the Bayes and neural networks was measured by using entire and the certain part of MNIST database of handwritten digits. Recognition rate of 92.87 % was obtained, which in the study are promising, and it is clear that each character can be expressed with an equation by developing the method. By this way, the less memory using can be satisfied by storage of coefficients instead of the storage of image.

In future, the speed of data transport can be accelerated by the using of coefficients in place of images. Also, a new and efficiently data or image compression method can be proposed with algebraic curves.

Key Words: Handwriting, character recognition, implicit curves, modeling, Bayes, neural network, adaptive neuro-fuzzy classifier.

2012, 58 pages

TEŞEKKÜR

Bu çalışma için beni bilgi ve birikimleriyle yönlendiren, çalışmanın her safhasında yardımlarını esirgemeyen danışman hocam Doç. Dr. Bayram CETİŞLİ' ye teşekkürlerimi sunarım.

Çalışmalarında yardımlarını esirgemeyen arkadaşım Melike ŞİŞECİ' ye ve bana her zaman güvenen, attığım her adımda desteklerini arkamda hissettiğim aileme sonsuz saygı ve teşekkürlerimi sunarım.

İhsan PENÇE

ISPARTA, 2012

ŞEKİLLER DİZİNİ

Şekil 1.1. MNIST veri tabanında bulunan bazı örnekler	2
Şekil 3.1. Karakter öteleme işlemi a) Özgün karakter b) Ötelenmiş karakter	11
Şekil 3.2. Koordinat sistemi üzerinde Affine dönüşüm işlemi	11
Şekil 3.3. Ölçekleme işlemi a) Özgün karakter b) Düzenli ölçeklenmiş karakter c) Düzenli olmayan ölçekleme sonucunda elde edilen karakter	13
Şekil 3.4. Sınır bölgesinin belirlenmesi a) Özgün karakter b) Sınır bölgelerinden kesilmiş karakter	13
Şekil 3.5. “3” rakamının normalizasyonu a) Özgün rakam b) Normalize edilmiş rakam.....	15
Şekil 3.6. Genetik algoritmanın genel akış şeması	18
Şekil 3.7. Parçacığın iki boyuttaki hareketi	19
Şekil 3.8. Parçacık sürü optimizasyonu akış diyagramı.....	19
Şekil 3.9. Bir nöronun biyolojik yapısı	26
Şekil 3.10. Nöronun matematiksel modeli	27
Şekil 3.11. Çok katmanlı yapay sinir ağı mimarisi	28
Şekil 3.12. A dilsel terimine uygulanan dilsel kuvvetlerin farklı p değerleri için yorumları	31
Şekil 4.2. “1” rakamının 8. dereceden KCE ile modellenmesi	35
Şekil 4.3. “2” rakamının 8. dereceden KCE ile modellenmesi	35
Şekil 4.4. “3” rakamının 8. dereceden KCE ile modellenmesi	36
Şekil 4.5. “3” rakamının 8. dereceden KCE ile hatalı modellenmesi	37
Şekil 4.6. “4”rakamının 8. dereceden KCE ile modellenmesi	37
Şekil 4.7. “5” rakamının 8. dereceden KCE ile modellenmesi	38
Şekil 4.8. “6” rakamının 8. dereceden KCE ile modellenmesi	38
Şekil 4.9. “7” rakamının 8. dereceden KCE ile modellenmesi	39
Şekil 4.10. “8” rakamının 8. dereceden KCE ile modellenmesi	39
Şekil 4.11. “9” rakamının 8. dereceden KCE ile modellenmesi	40
Şekil 4.12. Arial yazı tipinde “3” rakamının GA ile modellenmesi.....	44
Şekil 4.13. Arial yazı tipinde “4” rakamının GA ile modellenmesi.....	44
Şekil 4.14. Arial yazı tipinde “8” rakamının GA ile modellenmesi.....	45
Şekil 4.15. Arial yazı tipinde “3” rakamının PSO ile modellenmesi	45
Şekil 4.16. Arial yazı tipinde “4” rakamının PSO ile modellenmesi	46
Şekil 4.17. Arial yazı tipinde “8” rakamının PSO ile modellenmesi	46
Şekil 4.18. Arial yazı tipinde “3” rakamının KTGO ile modellenmesi	47
Şekil 4.19. Arial yazı tipinde “4” rakamının KTGO ile modellenmesi	47
Şekil 4.20. Arial yazı tipinde “8” rakamının KTGO ile modellenmesi	47
Şekil 4.21. Arial yazı tipindeki rakamlar	48

ÇİZELGELER DİZİNİ

Tablo 4.1. MNIST veritabanının tamamındaki tanıma oranları.....	41
Tablo 4.2. USBS ile MNIST veritabanının tamamındaki tanıma oranları.....	41
Tablo 4.3. Eğitim ve test seti 1000 için tanıma oranları	42
Tablo 4.4. Eğitim ve test seti 8900 için tanıma oranları	42
Tablo 4.5. Eğitim seti 500 ve test seti 500 için tanıma oranları.....	42
Tablo 4.6. USBS ile eğitim seti 500 ve test seti 500 için tanıma oranları	43
Tablo 4.7. Eğitim seti 4450 ve test seti 4450 için tanıma oranları.....	43
Tablo 4.8. USBS ile eğitim seti 4450 ve test seti 4450 için tanıma oranları	43
Tablo 4.9. Arial yazı tipinde etkili kuvvetlere ait katsayılar.....	49

SİMGELER DİZİNİ

GA	Genetik algoritma
KCE	Kapalı cebirsel eğriler
KTGO	Kümeleme tabanlı global optimizasyon
ÖEEY	Ölçeklenmiş eşlenik eğitim yöntemi
PSO	Parçacık sürü optimizasyonu
RMSE	Ortalama hata kareleri kökü (Root mean square error)
USBS	Uyarlamalı sinir-bulanık sınıflayıcı
YSA	Yapay sinir ağları

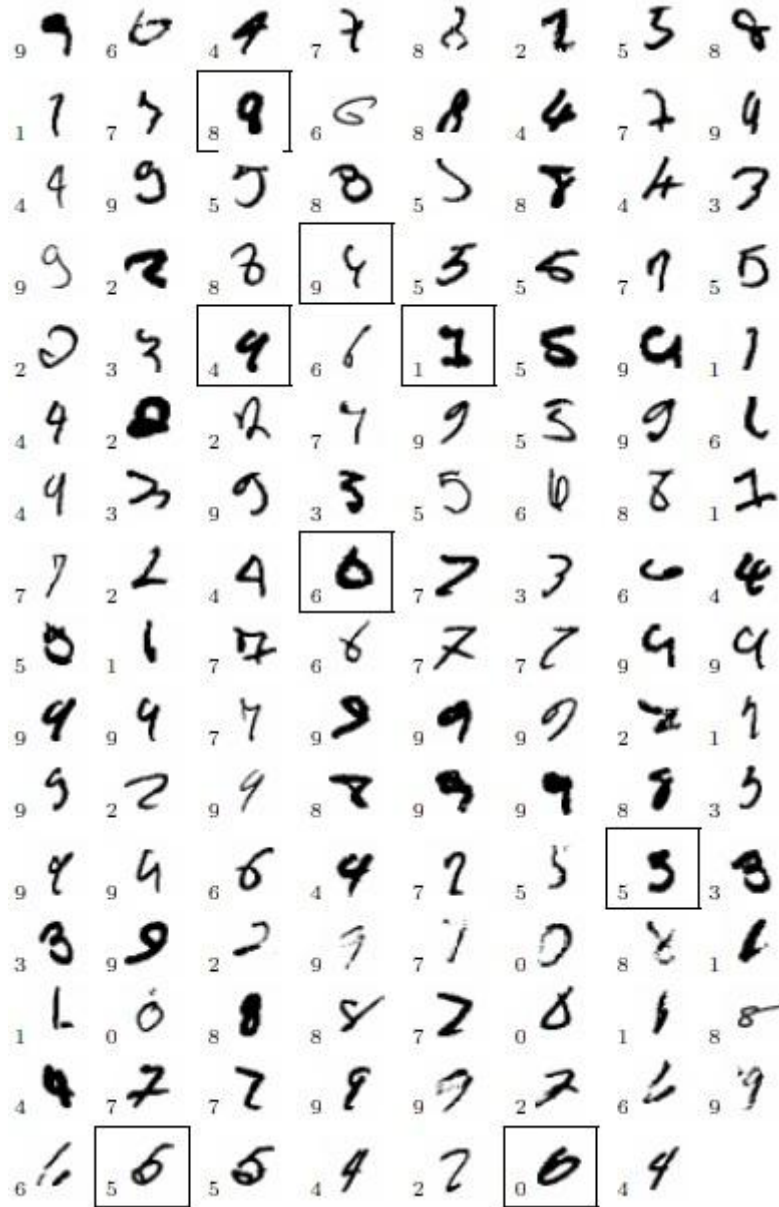
1. GİRİŞ

Karakter tanıma konusu üzerine günümüzde pek çok çalışma olmuştur. Bu çalışmalar genel olarak otomatik karakter tanıma başlığı altında incelenmektedir. Karakter tanıma işlemi pek çok alanda kullanılmakta olup en yaygın örnekleri bankalarda, kredi kartı ve sigorta şirketlerinde, vergilendirme ve tahsilât işlemlerinde, elektronik kütüphanelerde, plaka numaralarını tanımada, postanelerde posta kodlarını tanımada görülmektedir (Candemir, 2004).

Bu çalışmada el yazısı karakterleri üzerinde durulmuştur. El yazısı karakterleri normal karakterlere göre daha zor ayırt edilmekte olup halen daha kesin çözüme ulaştırılamamış bir konu olarak kalmaktadır. Veri seti olarak MNIST (LeCun et al., 1998) veri setinin seçilme sebebiyse fazla sayıda ve çeşitte el yazısı rakamlarını içeriyor olmasıdır. Bu sayede daha gerçekçi ve genel sonuçların alınması amaçlanmaktadır. MNIST veri setinde bulunan bazı el yazısı karakterleri ile bunlar arasında ayırt etmesi zor olan örnekler Şekil 1.1. 'de görülmektedir.

Diğer çalışmalardan farklı olarak bu çalışmada karakterlerin sadece kenarları değil tamamen kendileri 3-boyutlu olarak modellenmeye çalışılmıştır. Bu çalışmada el yazısı karakterlerinin tanınması üç ana bölümde incelenmiştir:

- Ön işlem den geçirme
- Modelleme
- Öznitelik seçimi ve sınıflandırma.



Şekil 1.1. MNIST veri tabanında bulunan bazı örnekler (Keysers, 2006)

Ön işlem den geçirme işlemi görüntü işlemede önemli bir yere sahip olup verilerin gürültüden arındırılması, standart boyut ve sınır kümelerinde çalışılması için sıklıkla kullanılmaktadır. Bu çalışmada da el yazısı karakterler ölçekleme ve ötelemeye göre normalize edilerek modelleme öncesinde ön işlem den geçirilmişlerdir. Bu sayede farklı boyutlarda olup merkezi orijinde olmayan karakterler standart hale getirilmiştir.

Modelleme işlemi için literatürde birçok yöntem kullanılmaktadır. Bunlar arasında Zernike momentler (Lisboa and Perantonis, 1991) ve Fourier tanımlayıcılar (Chung and Wong, 1997; Jeong and Jeong, 1999) en çok kullanılanlarıdır. Bu çalışmada ise genetik algoritma (Goldberg, 1989), parçacık sürü optimizasyonu (Kennedy and Eberhart, 1995), kümeleme tabanlı global optimizasyon ve kapalı cebirsel eğriler (Taubin, 1991; Keren et al., 1994; Civi, 1997; Unel and Wolovich, 2000; Tasdizen et al., 2000) kullanılmıştır. Bunlar arasında kapalı cebirsel eğriler çok hızlı çalışmakla beraber katsayıları da her zaman aynı bulduğundan dolayı sınıflandırma için uygun bulunmuştur. Öte yandan diğer yöntemlerde de karakterlerin modelleme işlemi iyi sonuçlar vermiş olup RMSE değerleri yaklaşık aynı çıkmıştır. Fakat hesaplanan katsayı ve kuvvet değerleri sürekli farklı çıktığından, değişmez öznitelikler elde edilemeyeceğine karar verilmiştir. Bu sebeple, diğer yöntemlerin sınıflandırma için uygun olmadığı anlaşılmıştır. Modelleme işleminde karakterler sınır kümeleri yerine 3-boyutlu ele alınıp karakterin tamamına eğriler oturtulmaya çalışılmıştır. Çalışmada kullanılan veri setindeki bütün el yazısı karakterler için 8. dereceden kapalı cebirsel eğrilerin iyi sonuç verdiği görülmüştür.

Öznitelik seçimi ve sınıflandırma işleminde ise Bayes, yapay sinir ağları (YSA) ve uyarlamalı sinir-bulanık sınıflayıcı (USBS) (Trier et al., 1995; Jain et al., 1996; Cetişli, 2005; Cetişli, 2006) kullanılarak sınıflandırma işlemi yapılmıştır. USBS yönteminin en büyük özelliği bulanık kümelerle uygulanan ve matematiksel olarak üstel kuvvetlere karşılık gelen dilsel kuvvetlerdir. Dilsel kuvvetler, önemli özniteliklerin vurgulanmasını, önemsiz özniteliklerin ise sistemden zayıflatılarak uzaklaştırılmasını sağlamaktadırlar (Cetişli, 2006). Bu sayede 8. dereceden kapalı cebirsel eğrilerle elde edilen 45 öznitelik sayısı yaklaşık yarıya düşürülebilmektedir. Yarıya düşürülen öznitelik sayısının yaklaşık 5. dereceden kapalı cebirsel eğrinin sahip olduğu öznitelik sayısına eşit olduğu düşünüldüğünde USBS yönteminin önemi daha da ön plana çıkmaktadır.

Çalışmada kullanılan MNIST karakter veritabanı için literatürde alınan tanıma oranları incelendiğinde genellikle %90 'nın üzerinde başarı sağlandığı görülmektedir. Bu başarıda, etkili bir şekilde seçilen özniteliklerin rol oynadığı görülmüştür. Bu çalışmada ise başarı oranından çok karakterlerin doğru şekilde modellenmesi ve

denklemlerinin oluřturulması zerine alıřılmıřtır. znitelik olarak sadece elde edilen denklemin katsayıları kullanılmıřtır. Literatrdeki diğerk yntemlere ait zniteliklerin de alıřmaya dhil edilmesiyle tanıma oranının artacağı ařıkrdır.

2. KAYNAK ÖZETLERİ

Karakter tanıma üzerine literatürde birçok çalışma bulunmaktadır. Bu çalışmalarda gerek kişi tarafından oluşturulmuş veritabanları kullanılırken gerekse çok sayıda ve çeşitte örnek içeren MNIST elyazısı karakter veritabanı da kullanılmaktadır. Çalışmalarda çok çeşitli yöntemler kullanılmakla beraber karakterlerin modellenmesi konusunda Bezier eğrileri, B-Spline eğrileri, global-bölgesel yaklaşım (Bézier, 1971; Bahr et al., 2001; Bhattacharya and Singla, 2006) ve kapalı cebirsel eğriler çoğunlukla kullanılmaktadır.

MNIST karakter veritabanı için elde edilen başarı oranları incelendiğinde sanal destek vektörlerinin 2. versiyonu (VSV2) ile %99.44 (Decoste and Scholkopf, 2002), gürbüz görüntü tabanlı öznitelikler ve sınıflamaları ile %99.41 (Teow and Loe, 2002), gradiyent öznitelikleri üzerinde destek vektör sınıflayıcısı (SVC) ile %99.58 (Liu et al., 2002), katlamalı yapay sinir ağları ve elastik bozulma ile %99.60 (Simard et al., 2003), LIRA gri tonlaması ile %99.41 (Kussul and Baidyk, 2004), eğitilebilir öznitelik çıkarıcı ve eğitim setini arttırmak için Affine dönüşümler ile birlikte destek vektör makinası (TFE-SVM) ile %99.46 (Lauer et al., 2005) ve permutasyon kodlama ile görüntü tanıma sistemleri ile %99.56 (Kussul et al., 2005), v.b. gibi çeşitli tanıma oranları sağlandığı görülür.

Bu çalışmada kullanılan kapalı cebirsel eğriler, literatürde çeşitli 2 ve 3-boyutlu nesneleri modellemek için kullanılmakla beraber karakterler üzerinde sadece (Candemir, 2004)' de kullanıldığı görülmüştür. (Candemir, 2004)' de de karakterler sadece sınır bölgelerinden 2-boyutlu olarak modellenmişlerdir. Literatürde bu çalışmada kullanılan 3-boyutlu olarak karakterlerin kapalı cebirsel eğrilerle modellenmesi konusunda bir çalışmaya rastlanmamıştır.

Literatürde kapalı cebirsel eğrilerle modelleme konusu geçmişten günümüze doğru incelendiğinde çoğunlukla 2 ve 3-boyutlu nesneleri modellemek için kullanıldıkları görülmektedir.

(Yazıcıoğlu vd., 2009), cebirsel eğriler kullanarak imge tabanlı görsel geri beslemeli denetim üzerine çalışmışlardır. Çalışmalarında, düzlemsel kapalı eğrilerin örtük

polinom gösteriminin kalibre edilmiş, imge tabanlı görsel geri beslemeli denetimde kullanılmasını önermişlerdir. Yöntemlerinde hedef nesnenin sınırlarının örtük polinom gösterimi, eğri oturtma yöntemi ile elde edilmektedir. Elde edilen cebirsel eğri sonlu sayıda doğrunun çarpımlarının toplamı olarak ayrıştırılmıştır. Bu doğruların kesişim noktaları öznel olarak görsel geri beslemeli denetimde kullanılmaktadır. Kullandıkları robot üzerinde başarılı sonuçlar almışlardır (Yazıcıoğlu vd., 2009).

(Candemir, 2004), kapalı cebirsel eğrilerle karakter tanıma üzerine çalışmıştır. Çalışmasında, karakterlere bir eğri uydurma algoritması ile kapalı cebirsel eğriler uydurulmuştur. Cebirsel eğriler daha sonra ayrıştırma teoremi ile daha basit eğriler olan doğru parçalarına ayrıştırılmışlardır ve doğruların kesişim noktalarından elde edilen ilişkili noktalar kullanılarak kanonik değişmezler hesaplanmıştır. Kanonik değişmezlerden benzerlik değerleri hesaplanarak karakterler sınıflandırılmışlardır. Yöntem, farklı kişilerden elde edilen 143 karakter üzerinde test edilip %77 başarı elde edilmiştir (Candemir, 2004).

(Sahin and Unel, 2004), global olarak stabilize edilmiş 3L eğri uydurma yöntemi üzerine çalışmışlardır. Bazı lineer eğri uydurma yöntemlerinin klasik en küçük kareler yöntemine göre gelişme gösterdiğini ama yinede eksik yada gürültülü verilere karşı global veriyi istikrarlı kılan bir çözüm ortaya koyamadıklarını belirtmişlerdir. Bu sebeple çalışmalarında ridge regresyon yöntemini kullanarak 3L yöntemini güçlendirmişlerdir. Çalışmalarını çeşitli şekillerin sınır kümelerinde test etmişlerdir (Sahin and Unel, 2004).

(Tasdizen et al., 2000), uygulamalar için cebirsel eğrilerin istikrarının artırılması üzerine çalışmışlardır. Çalışmalarında ilk olarak katsayıların ve modellemenin istikrarlı olması için gradient tabanlı yöntem kullanmışlardır. Bu durum ayrıca 3L algoritmasının neden diğer eğri uydurma algoritmalarından daha iyi olduğunu da açıklamaktadır. İkinci olarak ise, yöne göre değişmeyen ridge regresyon yöntemi ile katsayıların ve modellemenin daha da istikrarlı olması sağlanmıştır. Çalışmalarını çeşitli şekillerin sınır kümelerinde test etmişlerdir (Tasdizen et al., 2000).

(Sahin and Unel, 2005), derinlik bilgisi içeren verileri, global olarak stabilize edilmiş cebirsel yüzeyler ile uydurma üzerine çalışmışlardır. Çalışmalarında 3-boyutlu nesneler üzerinde kapalı cebirsel eğriler yöntemini kullanan 3L yöntemini kullanarak yöntemin global olarak istikrarlı olmasına çaba göstermişlerdir. Bunun için 3-boyutlu ridge regresyon yöntemini kullanmışlardır. Geliştirilen yöntemin 3-boyutlu yüzeyler için uygun olduğunu ifade etmişlerdir (Sahin and Unel, 2005).

(Rouhani and Sappa, 2011), 3L algoritmasını kullanan kapalı B-Spline eğri uydurma yöntemi üzerine çalışmışlardır. Çalışmalarında 3L yöntemini kapalı B-Spline eğrilerle birleştirmişlerdir. Bu sayede 3-boyutlu nesneler için daha iyi sonuçlar almışlardır (Rouhani and Sappa, 2011).

(Öden et al., 2001), kapalı polinom ve geometrik öznitelikler kullanarak el tanıma üzerine çalışmışlardır. Çalışmalarında kapalı cebirsel eğriler kullanarak elin sınır kümelerini 3L yöntemi ile modellemişlerdir (Öden et al., 2001).

(Tarel et al., 1998), cebirsel yüzey modelleri ile nokta eşleştirme olmadan serbest şekilli 3-boyutlu nesnelerin duruş tahmini üzerine çalışmışlardır. Çalışmalarında 3-boyutlu yüzeyleri kapalı cebirsel eğrilerle modellemek için 3L yöntemini kullanmışlardır. Hizalama problemi için tensör kasılmalarını kullanmışlardır (Tarel et al., 1998).

(Blane et al., 2000), veriyi kapalı cebirsel eğriler ve yüzeyler ile uydurmak için 3L algoritması kullanılması üzerine çalışmışlardır. Çalışmalarında 2 ve 3-boyutlu karmaşık nesneleri kapalı cebirsel eğrilerle modellemek için yeni bir yöntem olan 3L yöntemini ortaya koymuşlardır. Nesnenin kapalı cebirsel eğrilerle oluşturulmuş sınır bölgesini 3 katmana ayırarak bu bölgede en küçük kareler yöntemini uygulayarak modelleme işlemi yapmışlardır (Blane et al., 2000).

(Carr et al., 2001), radyal baz fonksiyonları ile 3-boyutlu nesnelerin yeniden düzenlenip sunulması üzerine çalışmışlardır. Çalışmalarında 3-boyutlu veriler üzerinde poliharmonik radyal temel fonksiyonları kullanarak tamamlanmamış yüzeyleri tamir ederek yeniden düzenlemişlerdir. Nesnenin yüzeyi daha sonra kapalı

cebirsel eğriler ile tanımlanarak radyal temel fonksiyonları ile modellenmiştir (Carr et al., 2001).

(Taron et al., 2005), yüksek dereceli polinomlar, değişken bant genişlikli çekirdekler ve parametrik olmayan yoğunluk tahmininde belirsizlikler kullanarak şekillerin modellenmesi üzerine çalışmışlardır. Kapalı polinom uzayında şekil modellemesi için yeni bir teknik ortaya koymuşlardır. Serbest şekil deformasyonu modeli kullanarak kapalı polinomlarla yüzeyler uydurulmuştur. Çalışmalarını MNIST veritabanı üzerinde denemişlerdir (Taron et al., 2005).

3. MATERYAL VE YÖNTEM

Bu çalışmada el yazısı karakterler modellenmek ve tanıma oranını arttırmak için çeşitli yöntemlerle işleme sokulmuşlardır. Bu yöntemleri ön işlemden geçirme, modelleme, öznelilik seçimi ve sınıflandırma oluşturmaktadır.

3.1. Ön İşlemden Geçirme

Ön işlemde geçirme işlemi görüntü işlemede vazgeçilmez yere sahip olup, neredeyse bütün çalışmalarda veri için asıl algoritma yazılmadan önce veriye çeşitli ön işlemler uygulanır. Bu işlemin en önemli sebebi verilerin standartlaştırılmak istenmesidir. Çünkü oluşturulan veri seti hiçbir zaman tamamen saf ve birbirleriyle tamamen uyumlu veriler içermez. Bu olumsuzluklara görüntü üzerindeki gürültü, boyutların birbiriyle örtüşmemesi (ölçekleme), renk uzayı farklılıkları, merkezin orijinde olmaması (öteleme) ve eğrilik gibi örnekler verilebilir. Bu çalışmada bu sorunlardan ölçekleme, öteleme ve gürültü giderme üzerinde çalışılmıştır. Görüntü üzerinde karakterden önce bulunan ve modellemeyi olumsuz etkileyen boşluklar gürültü olarak nitelendirilmiş ve bunun çözümü için sınır kümesinin belirlenmesi çalışması yapılmıştır.

Ön işlemde geçirme işlemleri arasında yer alan ölçekleme, öteleme ve sınır kümesinin belirlenmesi işlemleri çalışmada kullanılan veri setindeki görüntülerin sınır aralığını sabitlemek ve orijine oturtmak amacıyla kullanılmışlardır. Ölçekleme işlemi Affine dönüşümler ile öteleme işlemi ise Euclidean dönüşümlerle yapılmaktadır.

3.1.1. Euclidean dönüşümler

Euclidean dönüşümler cisimde yalnızca dönme ve öteleme yapar. Euclidean dönüşüm altında uzunluklar ve açılar değişmez. Cismin şeklinde bir değişikliğe neden olmaz, yalnızca cismin pozisyonu değişir. Euclidean dönüşüm, θ dönme parametresi, t_x x yönünde öteleme, t_y ise y yönünde öteleme parametresi olmak üzere üç parametre ile ifade edilir. İki boyutlu Euclidean dönüşüm matrisi

$$\begin{bmatrix} \cos\theta & \sin\theta & t_x \\ -\sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \text{ veya } \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

şeklinde gösterilir (Candemir, 2004).

Euclidean dönüşüm altında yeni koordinatlar (x', y') Denklem 3.1 ile ifade edilebilir (Candemir, 2004).

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$x' = x\cos\theta - y\sin\theta + t_x \quad (3.1)$$

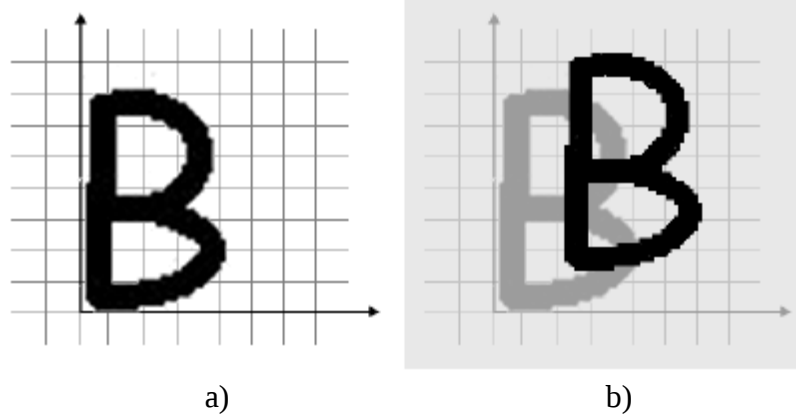
$$y' = x\sin\theta + y\cos\theta + t_y$$

3.1.1.1. Öteleme

Bir karakterin yer değiştirmesi, sınır kümesi üzerindeki tüm $x(k)$ ve $y(k)$ piksellerine, t_x, t_y sabit değişkenlerin eklenmesi ile Denklem 3.2'deki gibi ifade edilebilir (Candemir, 2004). $k = 1, 2, \dots, n$.

$$\begin{aligned} x_{yeni} &= x_{eski} + t_x \\ y_{yeni} &= y_{eski} + t_y \end{aligned} \quad (3.2)$$

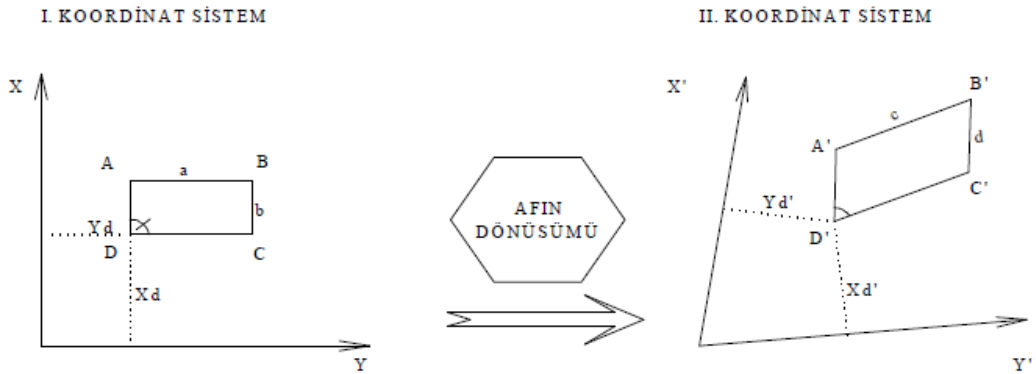
İki boyutta öteleme matrisi $\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$ ile ifade edilebilir. Koordinat düzlemindeki karakterin öteleme sonrasındaki hali Şekil 3.1'de görülmektedir.



Şekil 3.1. Karakter öteleme işlemi a) Özgün karakter b) Ötelenmiş karakter (Candemir, 2004)

3.1.2. Affine dönüşümler

Affine dönüşümlerde nesnenin şeklinde genişleme, daralma veya eğrilme gözükür (Candemir, 2004). Affine dönüşümüne, dik ve eğik koordinatlar arasındaki bir dönüşüm bağıntısıdır denilebilir (Ünsal, 2009). Koordinat düzeminde Affine dönüşüm işlemi Şekil 3.2. ile gösterilebilir.



Şekil 3.2. Koordinat sistemi üzerinde Affine dönüşüm işlemi (Ünsal, 2009)

Affine dönüşüm işleminin özellikleri şunlardır:

- Affine dönüşümlerde açılar ve uzunluklar korunmazlar.
- Doğrular, aynı doğru üzerindeki uzunlukların oranı ve doğruların paralellliği korunur.
- Bir doğru üzerindeki üç nokta, Affine dönüşüm geçirdikten sonra yine aynı doğru üzerindedir.

- Orta nokta Affine dönüşüm geçirdikten sonra yine diğer iki noktanın arasında kalacaktır.
- Kesişen doğrular, Affine dönüşüm geçirdikten sonra yine kesişeceklerdir (Candemir, 2004).

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}}_{\mathbf{Y}} + \underbrace{\begin{bmatrix} p_x \\ p_y \end{bmatrix}}_{\mathbf{T}} \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} m_1 & m_2 & p_x \\ m_3 & m_4 & p_y \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \bar{x} \\ \bar{y} \\ 1 \end{bmatrix} \quad (3.3)$$

Denklem 3.3 'de $\mathbf{A}$ Affine dönüşüm matrisidir; $\mathbf{M}$ lineer dönüşüm ve $\mathbf{T}$ yer değiştirme matrislerini ifade etmektedir. Eğer $\mathbf{M}$, dönme matrisi gibi dikgen (orthogonal) bir matris ise $\mathbf{M}^T \mathbf{M} = \mathbf{M} \mathbf{M}^T = \mathbf{I}$ eşitlikleri geçerlidir. Eğer bir yer değiştirme yoksa dönüşüm lineerdir ve Euclidean dönüşüm olarak adlandırılır. Affine dönüşüm altında yeni koordinatlar (x', y') Denklem 3.4' deki gibi hesaplanır.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m_1 & m_2 & p_x \\ m_3 & m_4 & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \Rightarrow \begin{aligned} x' &= m_1 x + m_2 y + p_x \\ y' &= m_3 x + m_4 y + p_y \end{aligned} \quad (3.4)$$

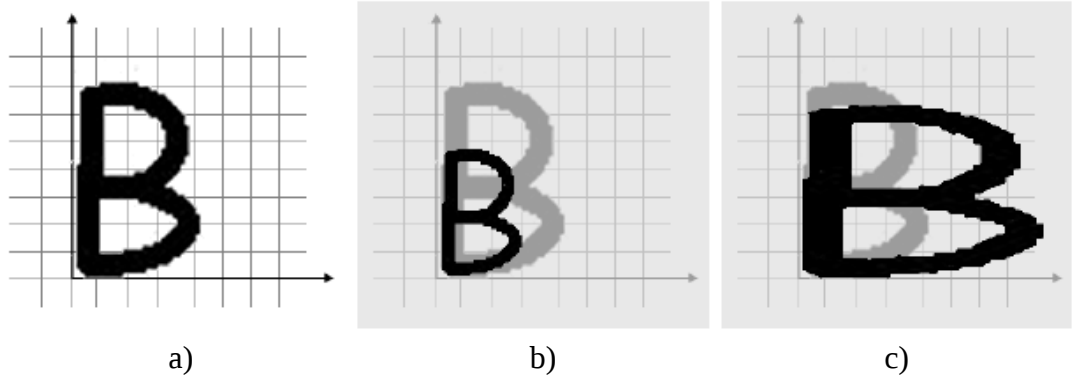
Affine dönüşüm genel olarak dönme, öteleme, genişleme, daraltma, eğme ve yansıma dönüşümlerinden bazılarını içeriyor olabilir (Candemir, 2004).

3.1.2.1. Ölçekleme

Veri tabanında bulunan görüntülerin ebatları birbirlerinden farklı olabilir. Bir karakterin ebadının değiştirilmesi $x(k)$ ve $y(k)$ sınır kümesi noktalarının s_1 ve s_2 katsayıları ile çarpılmasıyla sağlanabilir. Eğer s_1 ve s_2 katsayıları birbirlerine eşit ise düzenli ölçekleme yapılmaktadır. Ölçekleme faktörü s , orijinal karakterin hem x hem de y noktalarına eşit olarak etki eder (Candemir, 2004).

$$\mathbf{A} = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

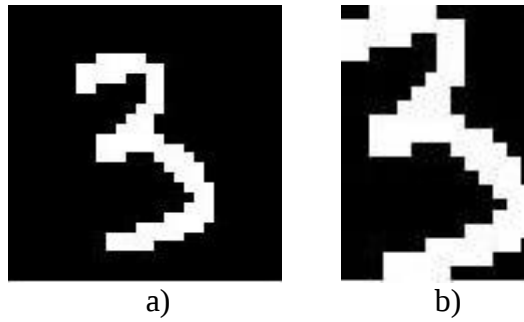
$\mathbf{A}$ düzenli ölçekleme matrisi, $\mathbf{B}$ ise düzenli olmayan ölçekleme matrisidir. Koordinat düzlemindeki karakterin ölçekleme sonrasındaki hali Şekil 3.3.'de görülmektedir.



Şekil 3.3. Ölçekleme işlemi a) Özgün karakter b) Düzenli ölçeklenmiş karakter c) Düzenli olmayan ölçekleme sonucunda elde edilen karakter (Candemir, 2004)

3.1.3. Sınır bölgesinin belirlenmesi

Sınır bölgesinin belirlenmesi işlemi karakterin başlangıç ve bitiş noktalarının tespitidir. Bu sayede karakter, bulunduğu bölgenin dışarısında kalan bölgeden soyutlanabilir. Bu işlem genellikle önceden belirlenmiş bölgeler üzerinde işlem yapılacağına kullanılmaktadır. Yapılan çalışmada el yazısı karakterlerin modellenme sürecinde karakterin etrafındaki boş bölgelerin modellemeyi olumsuz etkilemesi sebebiyle karakterler sınır bölgelerinden kesilmiştir. Sınır bölgesinin belirlenmesi işlemi Şekil 3.4.' de görülmektedir.



Şekil 3.4. Sınır bölgesinin belirlenmesi a) Özgün karakter b) Sınır bölgelerinden kesilmiş karakter

3.1.4. Karakterin normalize edilmesi

Normalizasyon, modellenecek veri kümesini koordinat merkezine taşır, x ve y yönündeki standart sapmaları 1 yapar. Bu işlemin yapılmasındaki bir diğer amaç da,

büyük koordinatlara sahip sınır kümelerine eğri uydurulması durumunda, eğrinin kararlılığının azalması ve gürültüler karşısında bozulmasıdır (Candemir, 2004). Normalizasyon işlemi sonucunda bütün veriler standart sınır kümelerine sahip olmuş olup bunun sayesinde de karakterlerin tanınmasındaki başarı oranı artış göstermiştir.

Veri kümesinde toplam N adet nokta bulunduğunu ve $Y_i = [x_i, y_i]^T$ ile ifade edildiğini varsayalım. Bu durumda tüm koordinat noktalarının toplanıp, toplam nokta sayısına bölünmesi ile veri kümesinin merkezi G elde edilir. Veri kümesinin merkezi Denklem 3.5'deki gibi hesaplanır (Candemir, 2004).

$$G = \frac{1}{N} \sum_{i=1}^N Y_i = \begin{pmatrix} x_c \\ y_c \end{pmatrix} \quad (3.5)$$

Veri kümesinin her bir noktasından ağırlık merkezinin çıkartılmasıyla, verilen veri kümesi koordinat merkezine taşınmış olur. Verinin kovaryans matrisi Denklem 3.6'daki şekilde hesaplanır (Unel, 1999).

$$\Sigma = \frac{1}{N-1} \sum_{i=1}^N (Y_i - G)(Y_i - G)^T \quad (3.6)$$

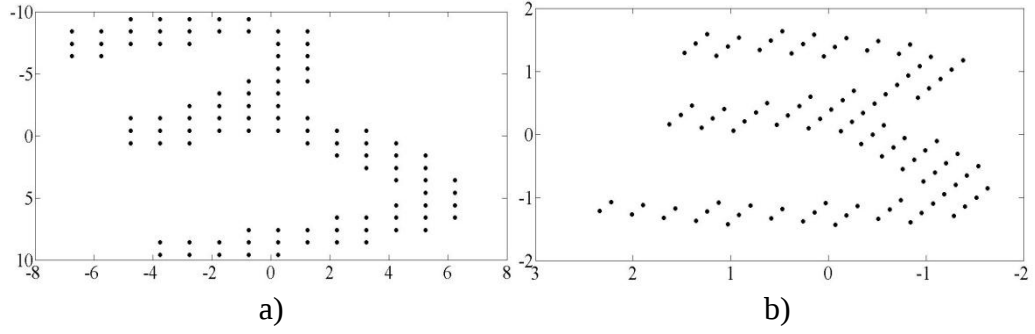
Kovaryans matrisi, pozitif tanımlı ve simetrik olması sebebiyle Denklem 3.7'deki şekilde ayrılabilir. $\mathbf{W}$, kovaryans matrisinin özvektörlerini, $\mathbf{D}$ ise özdeğerlerini tutar (Candemir, 2004).

$$\Sigma = \mathbf{W} \mathbf{D} \mathbf{W}^T \quad (3.7)$$

Elde edilen yeni karakterin veri kümesinde de toplam N adet nokta bulunduğunu ve veri kümesi $\hat{Y}_i = [\hat{x}_i, \hat{y}_i]^T$ ile ifade edildiğini varsayalım. Yeni veri kümesinin merkezi artık orijinde olup $\hat{Y}_i$, Denklem 3.8'deki gibi hesaplanır (Candemir, 2004).

$$\hat{Y}_i = (\hat{x}_i \ \hat{y}_i)^T = \mathbf{D}^{-\frac{1}{2}} \mathbf{W}^T (Y_i - G) \quad (3.8)$$

“3” rakamının özgün ve normalize edilmiş durumları Şekil 3.5.'de görülmektedir.



Şekil 3.5. “3” rakamının normalizasyonu a) Özgün rakam b) Normalize edilmiş rakam

3.2. Modelleme

Modelleme işlemi bir veri kümesinin uygun matematiksel gösterimlerle ifade edilmesidir. Bu çalışmada el yazısı karakterler modellenecek olup modelleme işlemi, karakteri en iyi ifade edebilecek denklemi oluşturmak üzerinedir. Modelleme işlemi için literatürde birçok yöntem olup bu çalışmada genetik algoritma (GA) (Goldberg, 1989), parçacık sürü optimizasyonu (PSO) (Kennedy and Eberhart, 1995), kümeleme tabanlı global optimizasyon ve kapalı cebirsel eğriler (KCE) (Taubin, 1991; Keren et al., 1994; Civi, 1997; Unel and Wolovich, 2000; Tasdizen et al., 2000) açıklanacaktır. Bütün bu yöntemler oluşturdukları eğrilerin hedef veri ile örtüşmesini amaçlar ve bu örtüşmeyi hesaplamak için çeşitli hata oranı hesaplama tekniklerini kullanırlar. Bu çalışmada genel olarak hata oranı hesap etmek için RMSE kullanılmıştır.

RMSE (Ortalama hata kareleri kökü); her veri noktası ile model arasında birbirine uymayan noktaların ortalama oranıdır. RMSE değerlerine bakılarak kalite ölçümü yapılabilmektedir (Karaca, 2007).

Ortalama hata kavramını temel alan ve yaygın şekilde kullanılan RMSE, Denklem 3.9 ile ifade edilmektedir (Yurtoğlu, 2005).

$$RMSE = \sqrt{\frac{1}{T} \sum_{i=1}^T (gerçek\ çıkış_i - hedef_i)^2} \quad (3.9)$$

Burada T hesaplanacak toplam veri sayısını, i hesaplanan verinin indisini belirtmektedir (Yurtoğlu, 2005).

3.2.1. Genetik algoritmalar

Geçmişte araştırmacılar, problemleri parametreler arasındaki ilişkileri deneme yanılma yoluyla çözmeye çalışırken, parametre sayısı artınca çözümsüzlük veya elde edilen çözümü değerlendirememesi problemi ortaya çıkmıştır. Bu tür çok parametrelili problemlerin çözümü için yeni teknikler bulunmaya çalışılmış ve yapay zekâ hem kullanım teknikleri açısından hem de optimizasyon algoritmaları açısından araştırmalarda sık başvurulan bir uygulama alanı olmuştur. Organik olmayan bir sistem olan yapay zekâ, insan düşünüş ve davranış biçiminin bir benzetim modelidir. Bu sebeple geliştirilen yapay zekâ teknikleri de insanın ve çevrenin davranışlarını modelleyerek problem çözmeyi amaçlamıştır (Gürsu ve İnce, 2007).

Darwin'in "en iyi olan yaşar" (survival of the fittest) prensibine dayalı olarak bir popülasyonu oluşturan bireylerin rekabet etmelerini sağlayan, evrimsel süreci taklit eden genetik algoritmalar (GA) ilk olarak John Holland tarafından ortaya atılmıştır (Holland, 1975).

Tabiatla olup biten tüm olaylarda, doğal seçimin olduğu göze çarpar. Birbirinden ayrı muazzam organizmalar, bu aşamalara gelirken sanki optimizasyon algoritmaları oluşmuş ve optimum çözüm elde edilmiştir. Evrim süreci, hayat şartlarına en uygun olanın yaşamasını sağlayan büyük bir algoritmadır (Çelebi, 2003). İteratif ve stokastik bir süreç olan GA, biyolojik süreci modelleyerek fonksiyonları optimize eden evrim algoritmalarıdır. GA' nın işleyişi, her bir ferdi, kromozomlar şeklinde temsil eden başlangıç popülasyonu ile başlar. Başlangıç popülasyonundaki her bir kromozom, problemin olası bir çözümünü temsil eder. Başlangıç popülasyonu genellikle rastgele oluşturulur. Popülasyon sürekli daha iyi çözümler oluşturmaya çalıştığı için, zaman içinde değişir. GA' nın genel yapısı içerisinde, popülasyon içerisindeki her çözüme diğer çözümler ile karşılaştırıldığında ne kadar iyi bir çözüm olduğunu gösteren bir uyum değeri atanır. Uyum değeri ne kadar büyükse bireyin veya organizmanın, sonraki nesilde yaşama şansı ve özelliklerini sonraki nesillere taşıma şansı o kadar fazla olur. GA popülasyonu geliştirilerek en uygun bireyin

oluşturulmasını hedeflemektedir. En iyi çözümü elde edebilmek için, zayıf çözümleri evrimsel bir işleyişe göre eleme yoluna gider. En iyi çözüm yapılan çevrimler sonucunda hâlâ hayatta kalabilmeyi başaran çözümdür (Mitchell, 1998). Bunu sağlamak için uyum değeri yüksek bireylere, popülasyondaki diğer bireylerle eşleşerek üreme fırsatı verilir. Böylece bir önceki neslin bazı özelliklerini taşıyan yeni bireyler meydana getirir. Mevcut neslin en iyi bireylerini seçip, yeni bireyler elde edilerek yeni bir popülasyon oluşturulur. Yeni nesil önceki neslin iyi bireylerinin sahip olduğu özelliklerin büyük bir kısmını taşır. Böylece nesiller boyu iyi özellikler popülasyona taşınır. Eğer GA iyi düzenlenirse popülasyon en uygun çözüme yakınsar (Beasley et al., 1993).

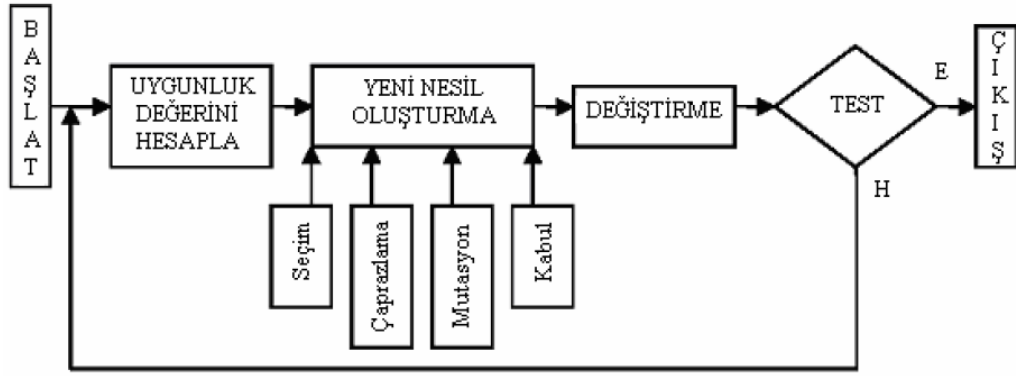
Genetik algoritmada bağımsız parametrelerin kromozomlar içinde kodlanması gerekmektedir. Yığındaki her birey, ikili düzende veya tamsayı olarak kodlanmaktadır (Uygunoğlu ve Yurtcu, 2006).

Genetik algoritmanın yapısı aşağıda genel hatları ile verilmektedir.

1. Olası çözümleri temsil eden başlangıç popülasyonunu oluştur.
2. Popülasyondaki her çözümün uygunluk değerlerini hesapla.
3. Durdurma kriteri sağlanıyorsa algoritmayı durdur. Uygunluğu en iyi olan çözüm kromozomu belirle. Durdurma kriteri sağlanmamışsa aşağıdaki adımlara devam et.
4. Başlangıç popülasyonunu değiştirmek, yeni popülasyonlar elde etmek için genetik algoritma operatörlerini (çaprazlama, mutasyon) uygula.
5. Sonraki nesil için doğal seçim mekanizmasını çalıştır.
6. Adım 2' ye git,

şeklindedir (Gürsu ve İnce, 2007).

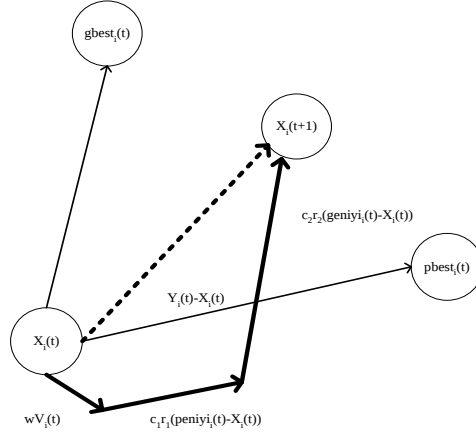
Genetik algoritmanın genel akış şeması Şekil 3.6.'da görülmektedir.



Şekil 3.6. Genetik algoritmanın genel akış şeması (Bolat et al., 2004)

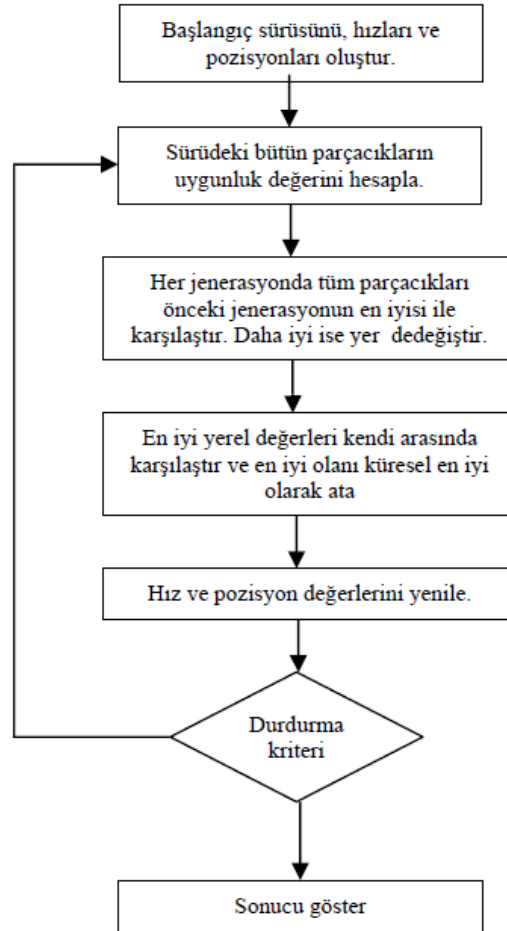
3.2.2. Parçacık sürü optimizasyonu

Parçacık sürü optimizasyonu (PSO), sürü halinde hareket eden balıklar ve böceklerden esinlenerek Kennedy ve Eberhart tarafından geliştirilmiş bir optimizasyon yöntemidir (Kennedy and Eberhart, 1995). Temel olarak sürü zekâsına dayanan bir algoritmadır. Sürü halinde hareket eden hayvanların yiyecek ve güvenlik gibi durumlarda, çoğu zaman rastgele sergiledikleri hareketlerin, amaçlarına daha kolay ulaşmalarını sağladığı görülmüştür (Özsağlam ve Çunkaş, 2008). PSO, bireyler arasındaki sosyal bilgi paylaşımını esas alır. Arama işlemi genetik algoritmalarda olduğu nesil sayısına yapılır. Her bireye parçacık denir ve parçacıklardan oluşan popülasyona da sürü (swarm) denir. Her bir parçacık kendi pozisyonunu, bir önceki tecrübesinden yararlanarak sürüdeki en iyi pozisyona doğru ayarlar. PSO, temel olarak sürüde bulunan bireylerin pozisyonunun, sürünün en iyi pozisyona sahip olan bireyine yaklaştırılmasına dayanır. Bu yaklaşma hızı rastgele gelişen durumdur ve çoğu zaman sürü içinde bulunan bireyler yeni hareketlerinde bir önceki konumdan daha iyi konuma gelirler ve bu süreç hedefe ulaşmaya kadar devam eder (Kennedy and Eberhart, 1995). Bu durum Şekil 3.7.'de görülmektedir.



Şekil 3.7. Parçacığın iki boyuttaki hareketi (Alataş vd., 2007)

PSO, sipariş miktarı belirleme, çizelgeleme problemleri, güç ve voltaj kontrolü, motor parametrelerini belirleme, tedarik seçimi ve sıralama problemleri gibi bir çok optimizasyon problemlerinde başarı ile kullanılmıştır (Wilke, 2005). Şekil 3.8.'de PSO'nun akış diyagramı görülmektedir.



Şekil 3.8. Parçacık sürü optimizasyonu akış diyagramı (Özsağlam ve Çunkaş, 2008)

Algoritma temel olarak aşağıdaki basamaklardan oluşur;

1. Rastgele üretilen başlangıç pozisyonları ve hızları ile başlangıç sürüsü oluşturulur.
2. Sürü içerisindeki tüm parçacıkların uygunluk değerleri hesaplanır.
3. Her bir parçacık için mevcut nesilden yerel en iyi (pbest) bulunur. Sürü içerisinde en iyilerin sayısı parçacık sayısı kadardır.
4. Mevcut nesildeki yerel eniyiler içerisinde küresel en iyi (gbest) seçilir.
5. Pozisyon ve hızlar Denklem 3.10 ve Denklem 3.11 ile yenilenir.
6. Durdurma kriteri sağlanıncaya kadar 2., 3., 4. ve 5. adımlar tekrar edilir (Özsağlam ve Çunkaş, 2008).

$$V_{id} = W * V_{id} + c_1 * rand_1 * (P_{id} - X_{id}) + c_2 * rand_2 * (P_{gd} - X_{id}) \quad (3.10)$$

$$X_{id} = X_{id} + V_{id} \quad (3.11)$$

Burada X_{id} pozisyon ve V_{id} hız değerlerini verirken, $rand_1$ ve $rand_2$ değerleri rastgele üretilmiş sayılardır. P_{id} pbest'i simgelerken, P_{gd} ise gbest'i ifade eder. c_1 ve c_2 katsayıları $0 < c_1, c_2 \leq 2$ öğrenme faktörleridir ve hızlanma katsayıları olarak adlandırılır. Burada $W \in [0.8, 1.2]$ atalet ağırlığıdır ve orijinal PSO hız güncelleme denklemi $W = 1$ 'dir. Atalet ağırlığı, bir önceki hızın ne kadarının bir önceki zaman adımından tutulacağını gösterir. Kısaca, PSO 'da atalet ağırlığı, global ve yerel arama yeteneğini dengelemek için kullanılır. Büyük atalet ağırlığı global arama, küçük ağırlık ise yerel arama yapılmasını kolaylaştırır (Alataş vd., 2007).

3.2.3. Kapalı cebirsel eğriler

Kapalı cebirsel eğriler (KCE), x, y değişkenlerine sahip $f(x, y) = 0$ eşitliğini sağlayan örtük denklemler ile tanımlanır. $z = f(x, y)$ yüzeyinin, $z = 0$ noktalar kümesi bize $f(x, y) = 0$ cebirsel eğrisini verir (Unel and Wolovich, 2000).

Cebirsel bir eğrinin kapalı polinom olarak gösterimi olan $f(x, y)$ Denklem 3.12'deki gibi ifade edilir (Blane et al., 2000).

$$f(x, y) = \sum_{0 \leq i, j, 0 \leq i+j \leq d} a_{ij} x^i y^j \quad (3.12)$$

Burada d , kaçınıcı dereceden KCE ile işlem yapılacağını belirtmektedir. Dördüncü dereceden KCE şu şekilde ifade edilir.

$$f(x, y) = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 + a_{30}x^3 + a_{21}x^2y + a_{12}xy^2 + a_{03}y^3 + a_{40}x^4 + a_{31}x^3y + a_{22}x^2y^2 + a_{13}xy^3 + a_{04}y^4$$

Burada x ve y karakterin koordinat noktalarını gösterip Denklem 3.13' de $\mathbf{X}$ ile ifade edilirken, $\mathbf{a}$ katsayılar vektörünü göstermekte olup Denklem 3.14 ile ifade edilir.

$$\mathbf{X} = [1 \ x \ y \ x^2 \ xy \ y^2 \ x^3 \ x^2y \ xy^2 \ y^3 \ x^4 \ x^3y \ x^2y^2 \ xy^3 \ y^4] \quad (3.13)$$

$$\mathbf{a} = [a_{00} \ a_{10} \ a_{01} \ a_{20} \ a_{11} \ a_{02} \ a_{30} \ a_{21} \ a_{12} \ a_{03} \ a_{40} \ a_{31} \ a_{22} \ a_{13} \ a_{04}] \quad (3.14)$$

Katsayılar vektörü olan $\mathbf{a}$ en küçük kareler yöntemi kullanılarak Denklem 3.15 ile hesaplanır (Cetişli and Edizkan, 2011). En küçük kareler yöntemi karakterin veri kümesinin uydurulacak cebirsel eğriye uzaklığının karesini minimum yapmaya çalışır.

$$\mathbf{a} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \quad (3.15)$$

Burada $\mathbf{Y}$ karakterin veri kümesini ifade etmektedir.

Bu çalışmada kullanılan bütün el yazısı rakamlar 8. dereceden KCE ile modellenmiştir. Yapılan çalışmalarda 8. derecenin rakamları modellemede yeterli olduğu görülmüştür.

n . dereceden bir cebirsel eğrinin toplam katsayısı Denklem 3.16 ile bulunur.

$$\frac{(n+1)(n+2)}{2} \quad (3.16)$$

Polinomlar derecelerine göre adlandırılırlar. Derecesi 1 ise doğru, 2 ise konik, 3 ise kübik, 4 ise kuartik, ...vs. Örtük polinomun katsayısı 1 ise polinom, monic polinom olarak adlandırılır (Candemir, 2004).

3.2.4. Kümeleme tabanlı global optimizasyon

Bu yöntem popülasyon oluşturma tabanlı çalışmakla beraber yeni popülasyonu küme merkezleri etrafından seçmesi ve parametre sayısını duruma göre arttırması ile diğer yöntemlerden ayrılmaktadır. Geliştiren yöntemin mevcut global optimizasyon tekniklerine göre daha hızlı ve daha iyi sonuç verdiği görülmüştür. Bu durumda popülasyondaki elemanların tamamıyla rastgele değiştirilmesi yerine kümeleme sonucu oluşan küme merkezleri etrafından seçilmesi önemli rol oynar. Bunun yanı sıra eğer önceden belirlenen parametre sayısı sistemin performansını yeteri kadar arttıramıyorsa yani belli bir performans değerinde takılı kalıyorsa, yöntem parametre sayısını arttırarak daha iyi sonuçlar üretilmesini sağlamaktadır.

Bu yöntem ile oluşturulan denklemin kuvvetlerini popülasyonda ki en iyi elemanlar oluştururken, katsayılar ise en küçük kareler yöntemi ile Denklem 3.15 deki gibi hesaplanır (Cetişli and Edizkan, 2011).

Oluşturulan yöntemin algoritması şu şekildedir.

1. Önceden belirlenen popülasyon büyüklüğüne göre ilk popülasyonu rastgele elemanlardan oluştur ve popülasyondaki her elemanın RMSE değerini hesapla.
2. Popülasyon-hata oranı ve önceden belirlenen küme sayısına göre bulanık C-ortalamlar kümeleme yöntemi ile kümeleme işlemi yap.
3. Önceki popülasyonun en iyi elemanlarıyla birlikte kümeleme sonucu her bir kümenin merkezinden rastgele seçilen yeni elemanlarla yeni popülasyonu oluştur.
4. Yeni popülasyondaki elemanların değerlerini önceden belirlenmiş sınırlar içinde tut.
5. Yeni popülasyondaki her eleman için RMSE değeri, kuvvet ve katsayıları hesapla.

6. Eğer popülasyondaki en iyi eleman uzun bir süre boyunca sürekli aynı kaldıysa parametre sayısını arttır ve yeni popülasyonun boyutunu düzenle.
7. İterasyon sayısı belirlenen değere gelene kadar veya RMSE değeri çok küçülene kadar 2. ve 6. adımlar arasını tekrarla.

3.3. Öznitelik Seçimi ve Sınıflandırma

Öznitelik seçimi bir veri kümesindeki grupların ayırt edici özellikleri belirlemektir. Bu sayede verinin tamamı yerine daha az sayıda değişken kullanarak bütün veri kümesini ifade edebilmek mümkün olmaktadır. Veri kümesinden elde edilen özniteliklerin kullanışlı ve gerçek hayata uygulanabilir olması için tasarlanan sistemlerin eğitilmesi gerekmektedir. Bu noktada iki tip eğitim yöntemi öne çıkmaktadır. Bunlar öğreticili ve öğreticisiz öğrenmedir (Jang et al., 1997). Öğreticili öğrenmede; eğitim setindeki örneklerin her birinin hedef değerleri bellidir ve sistem bu hedeflere ulaşmak için kendi parametrelerini uyarlar. Hata belli bir değerin altına indiğinde ise öğrenme tamamlanır. Sistem oluşturmada en çok kullanılan yöntem budur (Şişeci, 2012).

3.3.1. Öznitelik seçimi

Bir veri kümesinde bulunan veriler her zaman birbirlerinden kolaylıkla ayrılmazlar. Bu durum veri kümesi ile oluşturulacak ve eğitilecek sistemler için sorun oluşturmaktadır. Bunun için veri kümesini iyi analiz edip verilerin en belirleyici özelliklerini tespit etmek gerekir. Eğer doğru öznitelikler belirlenirse sistemin eğitilip daha sonra karşılaşılabilecek yeni verilere en doğru sonuçlar üretmesi sağlanır.

Bu çalışmada el yazısı karakterlerin kapalı cebirsel eğriler ile modellenmesi sonucu elde edilen katsayılar matrisi öznitelik olarak alınmıştır.

3.3.2. Sınıflandırma

Örüntü tanıma ve sınıflandırma, genel anlamıyla bir veriye, bir fiziksel işleme ya da bir olaya bir sınıf etiketi atanması işidir. Bu atama işlemi ise her zaman bir veriden, bir fiziksel sistemden ya da bir olaydan elde edilen öznitelikler sayesinde

gerçekleştirilir. Bir veri toplama sistemi ile veriye ait bazı özellikler elde edilir ve bu özellikler kullanılarak öznitelik çıkarımı işlemi yapılmaya çalışılır. Benzer durumların benzer özniteliklere sahip olması gerekmektedir (Başaran, 2009). Literatürde çeşitli sınıflayıcılar bulunmaktadır. Bu çalışmada sınıflandırma işlemi için Bayes, YSA ve USBS kullanılmıştır.

3.3.2.1. Bayes sınıflandırıcı

Naive Bayes sınıflandırıcı kuvvetli bağımsız varsayımlarla Bayes teoremini temel alan olasılıklı bir sınıflayıcıdır. Bu olasılık modelinde altı çizilecek nokta bağımsız özellik modeli olmasıdır (Uzun, 2007). Yalın tasarımına ve görünüşte basitleştirilmiş varsayımlara rağmen Naive Bayes sınıflandırıcı gerçek dünya durumlarında beklenenden çok daha iyi sonuçlar vermektedir (Maron, 1961; Domingos and Pazzani, 1997; McCallum and Nigam, 1998; Hand and Yu, 2001; Rish, 2001; Mozina et al., 2004; Menzies et al., 2007).

Bayes sınıflandırmasında kullanılan ölçütler birbirlerinden bağımsız ise tek değişkenli normal dağılım, eğer ölçütler arasında ilişki varsa çok değişkenli normal dağılım kullanılır (Turhan ve Bener, 2007).

Tek değişkenli normal dağılımda, $x \sim N(\mu, \sigma^2)$, veri noktası x , ortalama değer μ ve standart sapma σ ile dağılır. Olasılık yoğunluk fonksiyonu Denklem 3.17'deki şekilde tanımlanır (Alpaydin, 2004).

$$p(x) = \frac{1}{\sqrt{(2\pi)\sigma}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (3.17)$$

Denklem 3.17'e göre, veri noktalarının ortalamadan sapmalarının standart sapmalar cinsinden değerleri üstel fonksiyon içerisinde bir uzaklık birimi olarak kullanılır. Tek değişkenli bir dağılım olduğundan, değişkenler arası ilintiler dikkate alınmaz (Turhan ve Bener, 2007).

Çok değişkenli normal dağılımda ise, $\mathbf{x} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, veri noktaları bütün ölçütlere karşılık gelen değerleri içeren vektörler olarak tanımlanır. Veri noktası $\mathbf{x}$, ortalama

vektörü $\boldsymbol{\mu}$ ve özdeğişinti matrisi $\boldsymbol{\Sigma}$ ile dağılır. Olasılık yoğunluk fonksiyonu Denklem 3.18'deki şekilde tanımlanır (Alpaydin, 2004).

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^{d/2} \boldsymbol{\Sigma}^{1/2}}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right) \quad (3.18)$$

Burada üstel fonksiyon içerisinde kullanılan uzaklık fonksiyonu Mahalanobis uzaklığıdır ve ölçüt değerlerinin doğrusal ilintilerini dikkate alır (Turhan ve Bener, 2007).

Bu dağılımlar ile Bayes teoreminin birleştirilmesi sonucu Bayes sınıflandırıcılar elde edilebilir. Teorik olarak, sınıflandırıcı için olasılık modeli koşullu bir modeldir ve Denklem 3.19 ile gösterilebilir.

$$p(C|F_1, \dots, F_n) \quad (3.19)$$

Burada C , bağımlı sınıf değişkenini ifade ederken $F_1, \dots, F_n$ koşullu birkaç özellik değişkenini ifade eder (Uzun, 2007).

Bayes teoremi, sonsal dağılımın önsel dağılım ve olabilirlikle orantılı olduğunu söyler. Bayes teoremi Denklem 3.20'deki gibi yazılır (Alpaydin, 2004).

$$p(C_i|x) = \frac{p(x|C_i) p(C_i)}{p(x)} \quad (3.20)$$

Bayes teoremini sınıflandırıcı için kullanırsak Denklem 3.21 elde edilir.

$$p(C|F_1, \dots, F_n) = \frac{p(F_1, \dots, F_n|C) p(C)}{p(F_1, \dots, F_n)} \quad (3.21)$$

Bağımsızlık varsayımları üzerine, C sınıf değişkeninin koşullu dağılımı Denklem 3.22'deki gibi yazılabilir (Uzun, 2007).

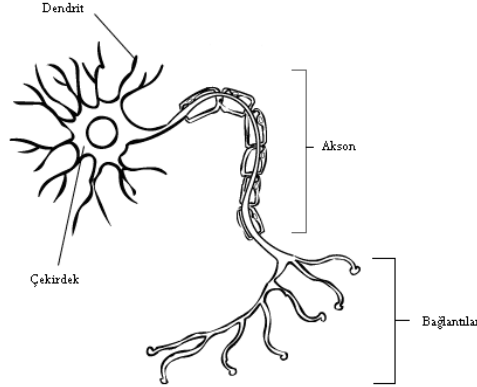
$$p(C, F_1, \dots, F_n) = \frac{1}{Z} p(C) \prod_{i=1}^n p(F_i|C) \quad (3.22)$$

Eğer öznitelik değişkenlerinin değerleri biliniyorsa $Z, F_1, \dots, F_n$ bağlı bir ölçekleme katsayısıdır (Uzun, 2007).

3.3.2.2. Yapay sinir ağları

Yapay sinir ağlarına (YSA) ilişkin mevcut birçok tanım olmakla birlikte genel olarak bilgi teknolojisindeki tanım olarak, YSA insan beyninin çalışma prensibini taklit eden program ve veri yapısı sistemleridir (Akyılmaz ve Ayan, 2006).

Merkezi sinir sisteminin temel işlem elemanı, sinir hücresi (nöron) dir. Sinir hücresi; hücre gövdesi, dendritler ve aksonlar olmak üzere üç bileşenden meydana gelir (Demir, 1997). Dendrit olarak adlandırılan yapı kendisine gelen uyarıları alan nöron girişleridir. Diğer hücrelere bilgiyi transfer eden elemanlar aksonlardır. Yani aksonlar nöron çıkışlarıdır. Akson ile dendrit arasındaki bağlantı ise sinapslar vasıtasıyla gerçekleştirilir. Beyin bütün bu işlemleri elektrokimyasal süreçlerle gerçekleştirir. Yapay sinir ağlarının işleyişi de buna benzer olarak gerçekleşmektedir (Ülker ve Civalek, 2002). Şekil 3.9.'da bir nöronun biyolojik yapısı görülmektedir.

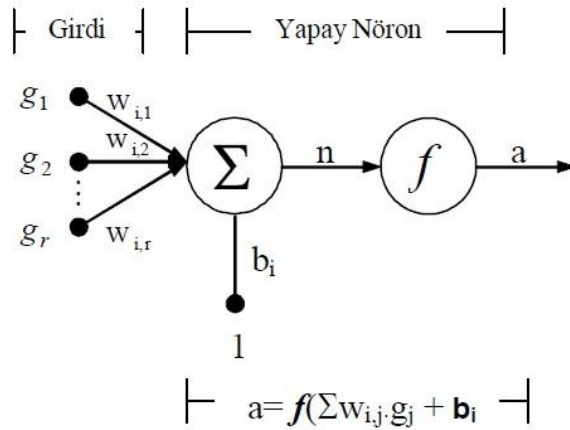


Şekil 3.9. Bir nöronun biyolojik yapısı (Kakıcı, 2009)

YSA her şeyden önce bir mimariye sahiptir. Bu mimari YSA'da veri akışının yönünü, sinirlerin tanımlandığı katmanları, her bir katmanda bulunan işlemci eleman sayılarını, işlemci elemanlar arasında tanımlı sinaptik ağırlıkları, sistemin girdi ve çıktılarını göstermektedir. Bu yönüyle YSA mimarisi aynı zamanda YSAların sınıflandırılmasının da temelini oluşturmaktadır (Akyılmaz ve Ayan, 2006). Bir YSA mimarisi, en az bir girdi bir çıktı katmanı adı verilen iki katmandan oluşur. Bunlara ek olarak genellikle ara katmanlar olarak adlandırılan gizli katmanlar da YSA

mimarisinde kullanılmaktadır. Her bir katmandaki sinirler diğer katmanlardaki sinirler ile sinaptik bağlar ile bağlıdır ve her bir bağ bir ağırlık değerine sahiptir (Haykin, 1994).

YSA'daki her bir sinir kendisine bir önceki sinirlerden gelen bilgiyi veri olarak alıp bir aktivasyon (ya da transfer) fonksiyonu ile işleme tabi tutup varsa bias değeri ile toplayıp oluşan çıktısını kendinden sonra bağlı olduğu sinirlere ilgili sinaptik ağırlıklar ile çarparak iletir. Genellikle bu çıktı değeri bundan sonraki sinirlere bir iletim olup olmayacağını belirleyecek bir eşik değeri ile karşılaştırılır. Bu karşılaştırma aktivasyon fonksiyonu tarafından belirlenir. Bu ardışık işlemler süreci ta ki en son çıktı katmanında çıktı değerlerinin elde edilmesine kadar sürdürülür. Bir yapay sinir kendisine gelen sinyalleri (g_j) geldikleri sinirlerle olan sinaptik bağlantıların ağırlıkları ($w_{i,j}$) ile çarparak toplamakta, buna varsa bias terimini (b_i) de ekleyerek elde edilen sonucu (n) tanımlanan aktivasyon fonksiyonuna (f) göndermektedir. Argümanı n olan aktivasyon fonksiyonu da skaler sinir çıktısı olan a değerini üretmektedir. Şekil 3.10.'da genel olarak herhangi bir ara katmandaki (saklı katman) i -inci yapay sinirin yapısı ve işlem akışı görülmektedir (Akyılmaz ve Ayan, 2006).



Şekil 3.10. Nöronun matematiksel modeli (Haykin, 1994)

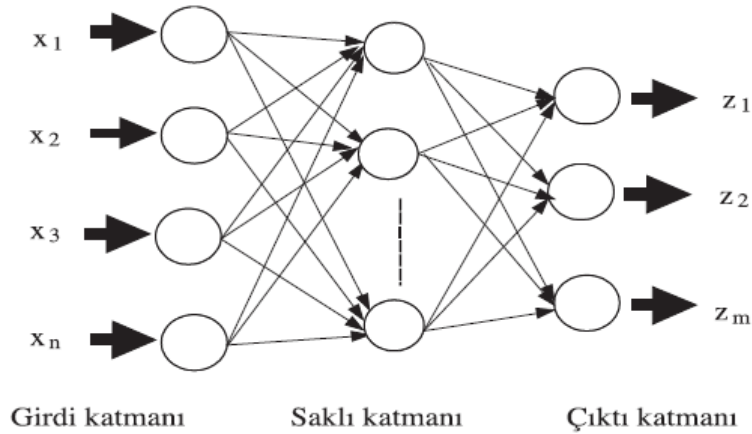
Şekil 3.10. incelendiğinde aktivasyon fonksiyonuna girecek toplam büyüklük Denklem 3.23 ile hesaplanırken, sinirin sonuç çıktısı ise Denklem 3.24 ile bulunur.

$$n = w_{i,1}g_1 + w_{i,2}g_2 + \dots w_{i,r}g_r + b_i = W_g + b_i = \sum w_{i,j}g_j + b_i \quad (3.23)$$

$$j = 1, 2, \dots, r$$

$$a = f(W_g + b_i) \quad (3.24)$$

Literatürde en çok ileri beslemeli sinir ağı yapıları kullanılır. İleri beslemeli YSA da, hücreler katmanlar şeklinde düzenlenir ve bir katmandaki hücrelerin çıkışları bir sonraki katmana ağırlıklar üzerinden giriş olarak verilir. Bilgi, orta ve çıkış katmanında işlenerek ağ çıkışı belirlenir. Bu yapısı ile ileri beslemeli ağlar doğrusal olmayan statik bir işlevi gerçekleştirir. İleri beslemeli üç katmanlı YSA'nın, orta katmanında yeterli sayıda hücre olmak kaydıyla, herhangi bir sürekli fonksiyonu istenilen doğrulukta yaklaştırabileceği gösterilmiştir. Şekil 3.11.'de gösterildiği gibi girdilerin uygulandığı katmana giriş katmanı, çıkışların alındığı katmana çıkış katmanı denir ve bu katmana dış dünyadan erişebilir. Giriş ve çıkış katmanlarının arasında gizli (saklı) katmanlar bulunur. Giriş ve çıkış katmanlarındaki nöron sayıları ele alınan problemin gereklerine göre belirlenir. Dolayısıyla gizli katman sayısındaki ve bu katmanların nöron sayılarındaki belirsizlikleri aşabilmenin tek yolu deneme yanılma yöntemidir (Arslan and İnce, 1996; Fırat ve Güngör, 2004).



Şekil 3.11. Çok katmanlı yapay sinir ağı mimarisi (Ülker ve Civallek, 2002)

Yapay sinir ağları insan zekâsı gibi örneklerle eğitilirler. Ağlar ne kadar çok örnekle eğitilirse problem üzerindeki teşhisi o kadar doğru olur (Özsoy ve Fırat, 2004). Çok tabakalı sinir ağlarının eğitimi için genellikle kullanılan eğitim algoritması hatayı geriye yayma (back-propagation) algoritmasıdır (Civallek, 1999). Bu algoritma,

türevi alınabilir aktivasyon fonksiyonu ile herhangi bir ileri beslemeli sinir ağları için kullanılabilir. Geriye yayma modeli her yerde diferansiyeli alınabilen aktivasyon fonksiyonu gerektirir. Denklem 3.25 'de verilen sigmoid tipli aktivasyon fonksiyonu türevi alınabilir ve sürekli problemlerin çözümünde kullanılmaktadır (Özsoy ve Fırat, 2004).

$$f_1(n) = \frac{1}{1 + e^{-n}} \quad (3.25)$$

Hatayı geriye yayma algoritmasının işlem sırası kısaca şu şekilde verilebilir (Fırat, 2002).

1. İlk önce her bir nöronun çıkış değerleri Denklem 3.24 ile hesaplanır.
2. Daha sonra ağın hatası Denklem 3.26 ile hesaplanır.

$$J_r = \frac{1}{2} \sum_{i=1}^k (d_i - a_i)^2 \quad (3.26)$$

3. Çıkış tabakası için ağırlık değerlerinin değişimleri Denklem 3.27 ve Denklem 3.28'e göre belirlenir.

$$\Delta w_{j,i} = \eta \delta_j a_i \quad (3.27)$$

$$\delta_j = (d_j - a_j) f'(net_j) \quad (3.28)$$

4. Gizli tabakalar için ağırlık değerlerinin değişimi Denklem 3.29 ile hesaplanır.

$$\delta_j = \sum_k (\delta_k - w_{k,j}) f'(net_j) \quad (3.29)$$

5. Değişen değerlere göre ağırlıkların güncellenmesi Denklem 3.30 ile yapılır.

$$w_{i,j}^{yeni} = w_{i,j}^{eski} + \Delta w_{i,j} \quad (3.30)$$

6. Bu adımlar toplam hata kabul edilebilir değere gelinceye kadar tekrarlanır (Fırat, 2002).

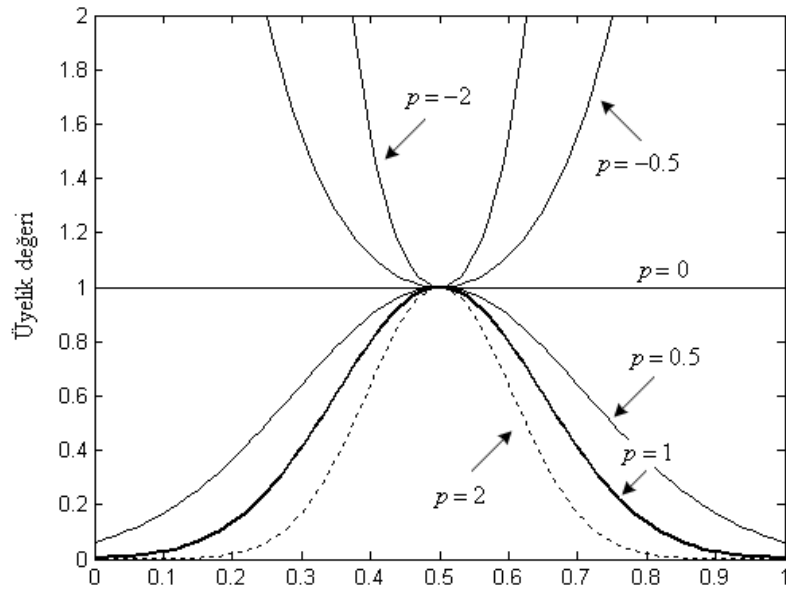
3.3.2.3. Uyarlamalı sinir-bulanık sınıflayıcı

Uyarlamalı sinir-bulanık sınıflayıcı (USBS), sinir ağlarının öğrenme özelliği ile bulanık mantığın insan düşüncesine en yakın eğer-ise çıkarsama mantığını bir araya getiren bir yöntemdir (Cetişli, 2005). Bu sınıflayıcının en büyük özelliği bulanık kümelerle uygulanan ve matematiksel olarak üstel kuvvetlere karşılık gelen dilsel kuvvetlerdir (Cetişli, 2006).

Uyarlamalı sinir-bulanık sınıflayıcı (USBS) sayesinde dilsel kuvvetlerin öznitelikler üzerindeki etkisi ile bazı öznitelikleri kuvvetlendirmek, bazılarını zayıflatmak mümkündür (Cetişli, 2006). Herhangi bir özniteliğin bütün sınıflar için aynı oranda ayırt edici olması beklenemez. Bu nedenle bütün öznitelikler, her sınıflandırma kuralında önemine göre farklı bir kuvvetle ele alınmalıdır. Böylece ayırt edici öznitelikleri seçmenin yanı sıra, seçilen özniteliklerin bazı sınıflar üzerindeki olumsuz etkisi giderilebilir. Bazı durumlarda ise bir öznitelik herhangi bir sınıf için önemli iken diğer sınıflar için önemsiz olabilir. Bu durumda bu özniteliğin önemli olduğu sınıftaki etkisi pekiştirilirken önemsiz olduğu sınıflardaki etkisi azaltılmalıdır. Bulanık mantığın dilsel zenginliği bu işlemi kolaylaştırmaktadır. Örneğin bir özniteliğin herhangi bir sınıftaki değerleri için gerek uzmanlar tarafından gerekse kümeleme yöntemleri ile “iyi”, “kötü” gibi bulanık kümeler tanımlansın. Bu bulanık kümeler kullanılarak yeni bulanık kümeler oluşturulabilir. Bunlardan bazıları “çok kötü”, “çok iyi”, “çok çok kötü”, “çok çok iyi” olabilir. Bu yeni kümeler özgün kümelerle farklı kuvvetler uygulanarak elde edilmektedir. Bu tanımlamalar bulanık kurallar oluştururken bulanık kümelerle uygulanabilir. Böylece klasik bulanık kurallardan daha ayrıntılı ve anlamlı yeni kurallar elde edilir (Cetişli, 2006). Genellikle bulanık kuvvetler, kullanılan dilsel terimlere göre bir değerle sabitlenmektedir. Örneğin “çok” ve “az” anlamları taşıyan yoğunlaşma ve genleşme dilsel kuvvetlerinde olduğu gibi, kuvvetler sırasıyla $p = 2$ ve $p = 0.5$ gibi değerlere sabitlenmektedir (Jang et al., 1997). Ancak USBS da, dilsel kuvvet değerlerinin belli aralıklarda değişmesine izin verilmektedir. Örneğin öznitelik seçimi sırasında kuvvet değerleri $[0 + 1]$ aralığında değişirken sınıflandırma sırasında $[0 + \infty]$ aralığında değişmektedir. Eğer bir özniteliği tanımlayan bulanık kümenin dilsel kuvvet değeri

sıfıra yakınsa o öznitelik önemsiz, bire yakınsa o öznitelik önemli anlamına gelir (Cetişli, 2006).

A dilsel terimine uygulanan dilsel kuvvetlerin farklı p değerleri $\{-2, -0.5, 0, 0.5, 1, 2\}$ için elde edilen kuvvetlendirilmiş A dilsel terimleri Şekil 3.12. 'de görülmektedir. Şekil 3.12. 'de görüldüğü gibi, $p < 0$ durumunda üyelik fonksiyonunun yönü değişmekte ve bunun sonucu olarak da elde edilen üyelik değerleri 1'den büyük olmaktadır. Üyelik değerinin $[0 + 1]$ aralığının dışında olması pek istenilen bir durum değildir. Çünkü bu durumda, elde edilen kümeler bulanık değildir. Bu nedenle $p \geq 0$ olmalıdır (Cetişli, 2006).



Şekil 3.12. A dilsel terimine uygulanan dilsel kuvvetlerin farklı p değerleri için yorumları (Cetişli, 2006)

Sınıflayıcı tasarlanırken parametrelerin uyarlanması, türev tabanlı bir eniyileme yöntemi olan Ölçeklenmiş Eşlenik Eğim Yöntemi (ÖEEY) ile yapılmaktadır (Cetişli, 2006).

USBSde kuralların oluşturulması ve parametrelerin başlangıç değerlerinin belirlenmesi için bulanık C-ortalamalar kümeleme yöntemi kullanılmaktadır. Bunun dışında K-ortalamalar, çıkarsamalı kümeleme yöntemi gibi yöntemler de kullanılabilir (Jang et al., 1997; Cetişli, 2006). Kullanıcı her sınıftaki küme (kural)

sayısını belirterek bulanık C-ortalamalar ile kuralları oluşturmaktadır. USBS, öznitelik seçimi ve sınıflandırma adımlarında farklı sınırlandırma ve kural yapılarıyla çalışır. Birinci adımda bütün dilsel kuvvetler 0.5 değerine kurulur ve her sınıf için sadece bir küme (kural) seçilir. Bu yapıdaki USBS belirlenen adım sayısında öznitelik seçimi için çalıştırılır. Eğitim süresince $0 \leq p_{ij} \leq 1$ olmasına dikkat edilir. Elde edilen ayarlanmış dilsel kuvvetlerle j. özniteliğin seçim değeri Denklem 3.31 ile bulunur (Cetişli, 2006).

$$P_j = \prod_{i=1}^c p_{ij} \quad (3.31)$$

USBS için öznitelik seçimi ve sınıflandırma algoritması şu şekildedir (Cetişli, 2006).

Öznitelik seçimi için;

1. Her sınıf için ortak küme sayısını 1 olarak belirle. Bütün kuvvetleri $p_{ij} = 0.5$ olarak ata. Seçilecek öznitelik sayısı n 'yi belirle.
2. Verideki her sınıfın ortalamasını ve standart sapmasını kullanarak USBS parametrelerinin başlangıç değerlerini belirle ve bulanık EĞER-İSE kurallarını oluştur.
3. USBSyi ÖEEY ile eğit.
4. Her sınıfın en büyük dilsel kuvvet (p_{ij}) değerlerini veren ve bütün sınıflarda en büyük dilsel kuvvet değerlerini veren özniteliklerini belirle.
5. Denklem 3.31 ile P_j değerleri en büyük olan ortak ayırt edici en iyi $n - c$ adet özniteliği ve bireysel ayırt edici olan en iyi c özniteliği seçerek X_{yeni} verisini oluştur.

Sınıflandırma için;

6. Her sınıf için küme (kural) sayısını g olarak belirle. Buna göre toplam kural sayısı $s_1 = g \cdot c$ olur. Bütün kuvvetleri $p_{ij} = 1$ olarak ata.

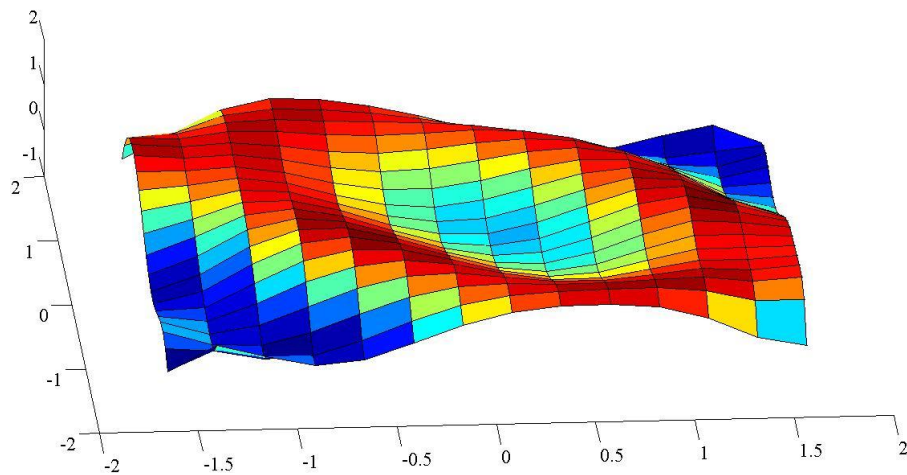
7. Bulanık C-ortalamalar kümeleme yöntemini kullanarak USBS parametrelerinin başlangıç değerlerini belirle ve bulanık EĞER-İSE kurallarını oluştur.
8. USBSyi ÖEEY ile eğit.
9. Eğitim ve test verilerinin sınıflandırma sonuçlarını elde et (Cetişli, 2006).

4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu çalışmada uygulanan farklı yöntemlere göre farklı sonuçlar alınmıştır. Fakat bu sonuçlar içerisinde elde edilen öznitelikler düşünüldüğünde sadece KCE ile elde edilenler sınıflandırma işlemi için uygundur. Çünkü signomial (Duffin et al., 1967) kuvvetlerin uygulandığı diğer yöntemlerde öznitelikleri oluşturan katsayılar sürekli değişmektedir. Bu durum sınıflandırmayı olanaksız kılmıştır.

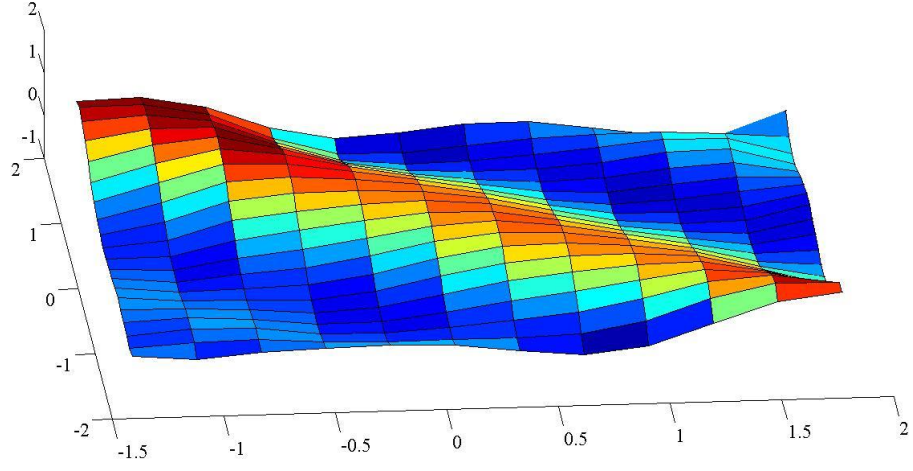
4.1. Kapalı Cebirsel Eğriler Kullanılarak Yapılan Çalışmalar

Çalışmada öncelikli olarak kullanılan KCE yönteminin el yazısı karakterleri modellemede çok hızlı olması ve sınıflandırma işlemi için aynı katsayıları bulması bir avantajdır. Bu durum MNIST karakter veritabanı için öznitelik çıkarmada da açıkça görülmüştür. MNIST içinde bulunan bütün rakamlar (60000 eğitim, 10000 test) 8. dereceden KCE ile modellenmişlerdir. Derecenin seçilmesinde deneme yanılma yöntemi kullanılmıştır. 8. dereceden daha az kuvvetlerde rakamlar düzgün bir şekilde modellenememiştir. 8. dereceden fazla kuvvetlerde ise rakamlar daha iyi modellenmesine karşın elde edilen katsayıların çok fazla ve orantısız olması sınıflandırma başarısını olumsuz etkilemiştir. Bu sebeple rakamların modellenmesinde 8. derecenin uygun olduğunda karar kılınmıştır. Şekil 4.1. - 4.11. arasında MNIST içerisindeki her rakam sınıfından bazı örneklerin 8. dereceden KCE ile modellenmiş durumları görülmektedir.

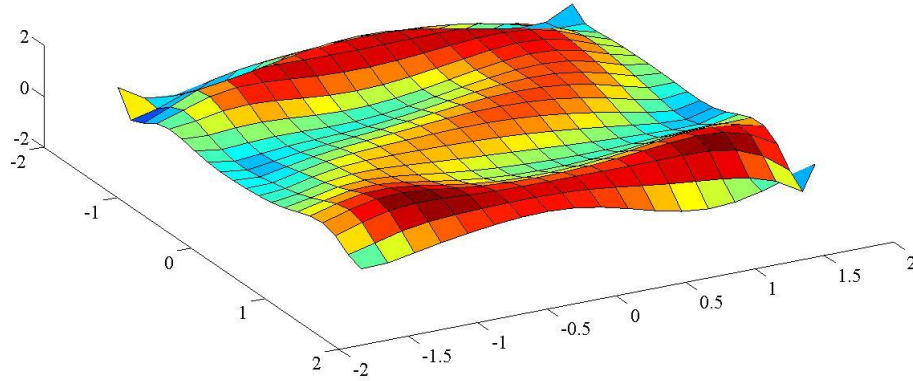


Şekil 4.1. “0” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.1. incelendiğinde KCEnin “0” rakamının ortasındaki boşluğu doğru bir şekilde modelleyebildiği görülmektedir.

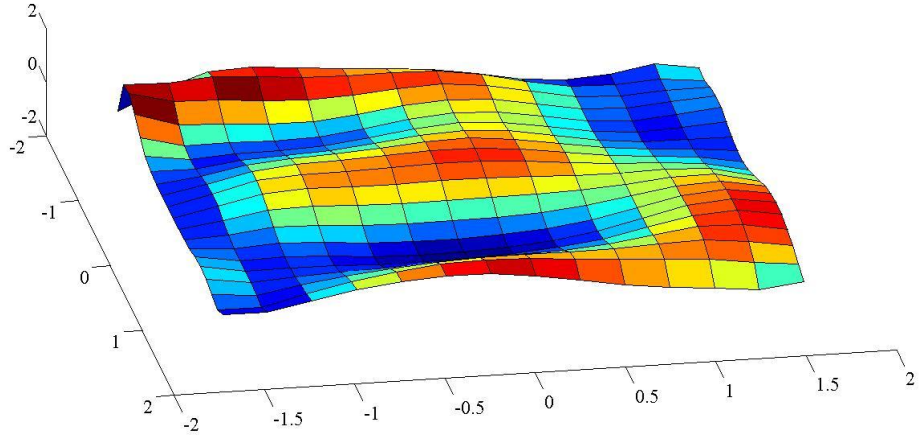


Şekil 4.2. “1” rakamının 8. dereceden KCE ile modellenmesi



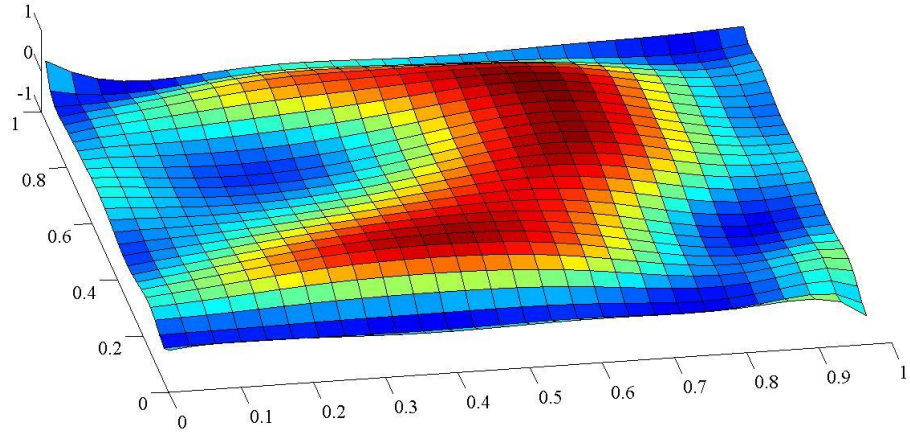
Şekil 4.3. “2” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.3. incelendiğinde “2” rakamı için kıvrımların başarı ile modellenenebildiği görülmektedir.



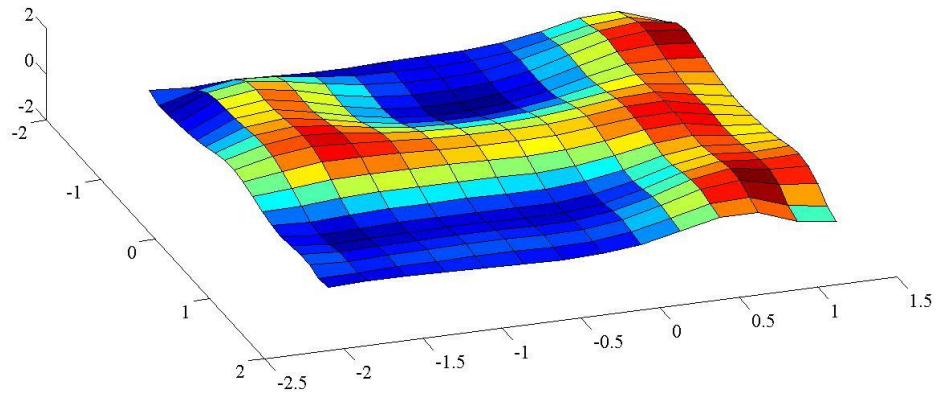
Şekil 4.4. “3” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.4. incelendiğinde “3” rakamının doğru bir şekilde modellendiği görülmüştür. Yapılan çalışmada özellikle “3”, “4” ve “8” rakamları üzerinde durulmuş ve denenen her yöntem ilk önce bu üç rakam üzerinde test edilmiştir. Bu durumun nedenini bu üç rakamın daha zor modellenmesi oluşturmaktadır. “3” rakamında bulunan iki adet boşluğun modellenmesinde zaman zaman sıkıntıya düşülmüştür ve sanki “2” rakamı gibi modellendiği olmuştur. “4” rakamı karmaşık bir yapıya sahip olduğundan “1” ya da “7” rakamı gibi modellendiği görülmüştür. “8” rakamında ise bulunan iki adet boşluk seçici bir durumdur. Yapılan denemelerde görülmüştür ki “3”, “4” ve “8” rakamları eğer başarı ile modellenenbiliyorlarsa geriye kalan rakamlarda bunların bir nevi türevleri olduğu için onlar da modellenenbilmektedir. Çalışmada karşılaşılan olumsuz durumlardan en büyüğü karakterin sınır bölgesi belirlenmeden yapılan denemelerde görülmüştür. Eğer karakter sınır bölgelerinden kesilmez ise karakterin etrafında bulunan boşlukların modellemeyi imkânsız kıldığı görülmüştür. Şekil 4.5. ‘de sınır bölgelerinden kesilmeyerek hatalı modellenmiş “3” rakamı görülmektedir.



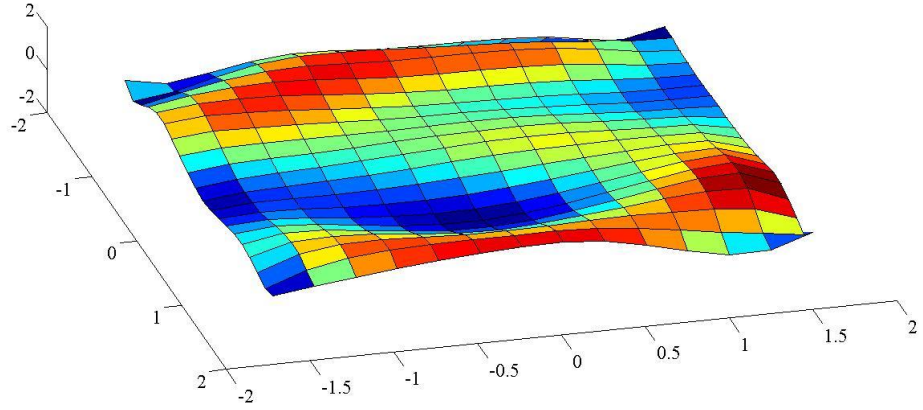
Şekil 4.5. “3” rakamının 8. dereceden KCE ile hatalı modellenmesi

Şekil 4.5. incelendiğinde “3” rakamının etrafındaki boşluk elenmediği için rakamın ara bölgeleri doğru bir şekilde modellenememiştir. Bu durum bütün diğer rakamlarda da görülmektedir. Eğer karakter sınır bölgelerinden kesilmez ise sadece ortada bir tümseğin bulunduğu Şekil 4.5. ‘deki gibi durumların oluştuğu görülür.



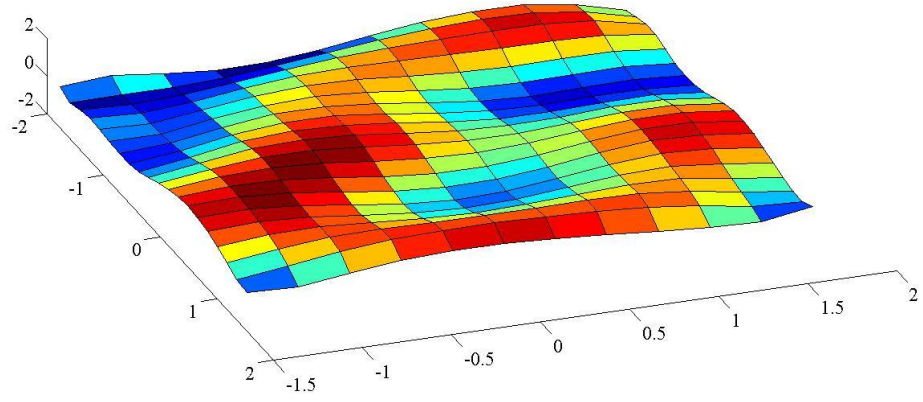
Şekil 4.6. “4”rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.6. ‘da da görüldüğü gibi en zor rakamlardan biri olan “4” rakamı başarı ile modellenebilmiştir. Fakat yine de MNIST veritabanı içerisinde çok farklı şekilde ve kimi zaman insan gözünün bile ayırt edemediği örnekler bulunduğu için rakamlar her zaman bu kadar güzel bir şekilde modellenememektedir.



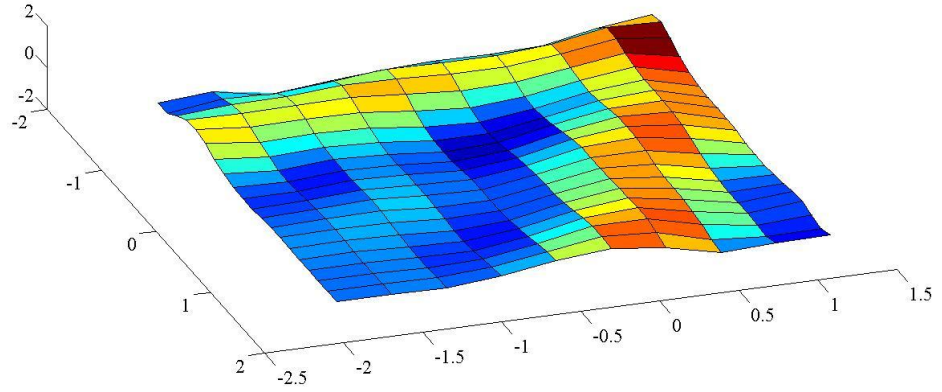
Şekil 4.7. “5” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.7. incelendiğinde rakamın “2” rakamına da çok benzediği görülmektedir. Rakamların ters durumları ya da dönme açılarının önemli olduğu uygulamalarda bu durum tanıma oranını düşürebilmektedir.



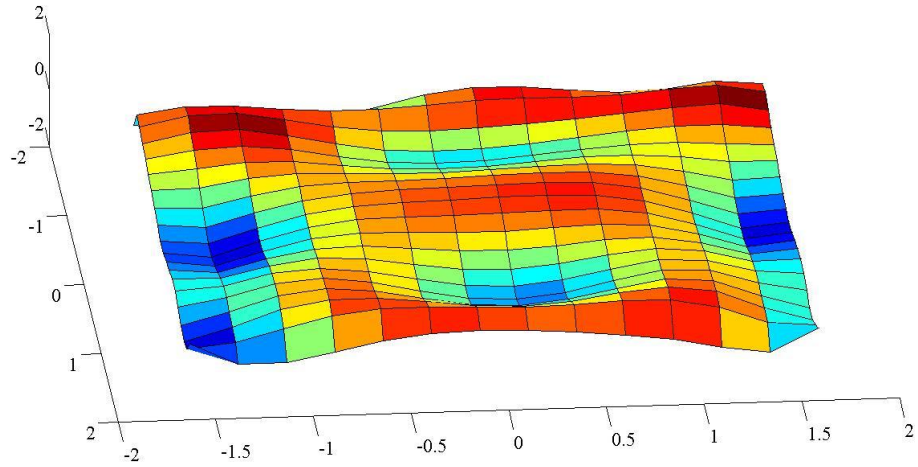
Şekil 4.8. “6” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.7. ‘deki durum Şekil 4.8. ‘de de görülmektedir. “6” rakamı bazı uygulamalarda “9” rakamıyla karışabilmektedir.



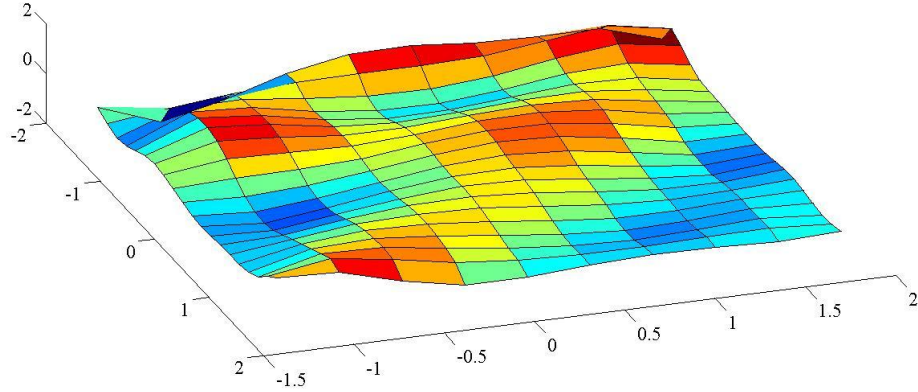
Şekil 4.9. “7” rakamının 8. dereceden KCE ile modellenmesi

KCE ile modellenen bütün karakterler incelendiğinde sınır kümelerinin genellikle aynı aralıkta olduğu gözlemlenir bu duruma normalizasyon işlemleri neden olmuştur. Bu sayede sınır bölgeleri çok büyük olan yani orantısız boyutlardaki rakamların elde edilen katsayıları orantısız olmamıştır. Bu da tanıma oranına olumlu yansımıştır.



Şekil 4.10. “8” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.10. incelendiğinde “8” rakamındaki iki boşluğunda başarılı bir şekilde modellendiği görülmektedir.



Şekil 4.11. “9” rakamının 8. dereceden KCE ile modellenmesi

Şekil 4.1. ve Şekil 4.11. arası incelendiğinde 8. dereceden KCEin rakamları modellemeye başarılı olduğu görülmektedir. Bu sayede her bir rakam polinomlarla da ifade edilebilmiştir. Böylece kısmen de olsa her bir rakamın bir denklemi yazılabilmektedir. Örnek vermek gerekirse Şekil 4.4.’ de görülen “3” rakamının ifadesi Denklem 4.1 de görülmektedir.

$$\begin{aligned}
 f^8(x, y) = & -0.032289x^0y^8 + 0.10065x^1y^7 + 0.064431x^2y^6 + 0.21204x^3y^5 + \\
 & 0.11365x^4y^4 - 0.03687x^5y^3 + 0.14377x^6y^2 - 0.1597x^7y^1 - \\
 & 0.004838x^8y^0 - 0.1056x^0y^7 + 0.18781x^1y^6 + 0.079366x^2y^5 + \\
 & 0.42821x^3y^4 + 0.16302x^4y^3 + 0.13989x^5y^2 + 0.19306x^6y^1 + \\
 & 0.074921x^7y^0 - 0.021569x^0y^6 - 0.81835x^1y^5 - 0.45449x^2y^4 - \\
 & 0.3849x^3y^3 - 0.87644x^4y^2 + 0.87533x^5y^1 - 0.40077x^6y^0 + 0.26322x^0y^5 - \\
 & 1.5776x^1y^4 - 0.44757x^2y^3 - 1.5193x^3y^2 - 0.98411x^4y^1 - 0.42939x^5y^0 + \\
 & 0.30224x^0y^4 + 1.3124x^1y^3 + 1.3701x^2y^2 - 1.2326x^3y^1 + 1.7618x^4y^0 - \\
 & 0.23632x^0y^3 + 3.0939x^1y^2 + 0.96765x^2y^1 + 1.0958x^3y^0 - 0.60391x^0y^2 + \\
 & 0.31341x^1y^1 - 1.7772x^2y^0 + 0.27689x^0y^1 - 1.1245x^1y^0 + 0.74707
 \end{aligned}
 \tag{4.1}$$

Denklem 4.1 ’de elde edilen bu gibi katsayılar, daha sonra çeşitli sınıflayıcılarla eğitilmiş ve tanıma oranları karşılaştırılmıştır. Sınıflandırma aşamasında MNIST veritabanının tamamının kullanılması yanı sıra bir kısmı da kullanılmıştır.

Bundaki amaç eğitim ve test verilerindeki sayının tanıma oranına etkisini ölçmektir. MNIST veritabanında bulunan 10000 adet test verisi içerisinde 1000 eğitim-1000 test, 8900 eğitim-8900 test, 500 eğitim-500 test ve 4450 eğitim-4450 test verisi olmak üzere dört farklı şekilde yeni veritabanları oluşturulmuştur. Bunlardan ilk ikisinde eğitim ve test seti aynı örnekleri içerirken, son ikisinde ise eğitim ve test seti birbirinden ayrılıp eşit sayıda fakat tamamen farklı örnekler içermektedir. Rakamların hepsi 8. dereceden kapalı cebirsel eğrilerle modellenmiş olup öznitelik olarak oluşturulan polinomun en küçük kareler yöntemi ile bulunan katsayıları kullanılmıştır. Sınıflayıcı olarak YSA ve Bayes kullanılmıştır. Ayrıca USBSnin seçtiği önemli özniteliklerden faydalanmak için USBS, kullanılan iki sınıflandırma yöntemine de dâhil edilmiştir. Bu sayede USBSnin önemli öznitelikleri bulma gücü ile Bayes ve YSA'nın sınıflandırmadaki gücü birlikte kullanılmıştır. Nitekim USBS normalde 45 adet olan öznitelik sayısını 23 adete indirmeyi başarmış olmakla beraber başarı oranında da iyi sonuçlar vermiştir. 23 öznitelik sayısı yaklaşık 5. dereceden kapalı cebirsel eğrinin sahip olduğu öznitelik sayısına denk gelmekte olup bunun önemi normalde el yazısı rakamlarının 5. dereceden eğrilerle ifade edilememesinde görülmektedir.

MNIST veritabanının parçalandığı örneklere değinmeden önce Tablo 4.1. ve Tablo 4.2. incelenirse MNIST veritabanının tamamı (60000 eğitim-10000 test) sınıflandırma işlemine dâhil edildiğinde alınan tanıma oranları görülmektedir.

Tablo 4.1. MNIST veritabanının tamamındaki tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
Bayes	45	81.77	82.03
YSA	45	95.80	92.87

Tablo 4.2. USBS ile MNIST veritabanının tamamındaki tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
USBS - Bayes	23	84.89	85.80
USBS - YSA	23	92.15	89.98

Tablo 4.1. de YSA'nın eğitim süresi daha da arttırıldığında eğitim setindeki tanıma oranı %97.31'e kadar çıktığı görülmüştür. Bu durumda test setindeki tanıma oranı %92.59 olmuştur.

MNIST veritabanı parçalandığındaki durumlar incelenecek olursa; eğitim seti ile veri seti aynı seçildiğindeki tanıma oranları Tablo 4.3 ve Tablo 4.4 'de görülmektedir.

Tablo 4.3. Eğitim ve test seti 1000 için tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
Bayes	45	99.00	99.00
YSA	45	98.60	98.60

Tablo 4.4. Eğitim ve test seti 8900 için tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
Bayes	45	85.95	85.95
YSA	45	91.56	91.56

Eğitim seti ve test seti birbirinden bağımsız seçildiğindeki tanıma oranları Tablo 4.5., Tablo 4.6., Tablo 4.7. ve Tablo 4.8. 'de görülmektedir.

Tablo 4.5. Eğitim seti 500 ve test seti 500 için tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
Bayes	45	100.00	57.00
YSA	45	99.80	65.60

Tablo 4.6. USBS ile eğitim seti 500 ve test seti 500 için tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
USBS - Bayes	23	99.00	73.20
USBS - YSA	23	99.80	62.80

Tablo 4.7. Eğitim seti 4450 ve test seti 4450 için tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
Bayes	45	88.06	84.02
YSA	45	94.51	86.26

Tablo 4.8. USBS ile eğitim seti 4450 ve test seti 4450 için tanıma oranları

Sınıflayıcı	Öznitelik Sayısı	Tanıma Oranı (%) Eğitim Seti	Tanıma Oranı (%) Test Seti
USBS - Bayes	23	88.51	85.46
USBS - YSA	23	88.24	80.85

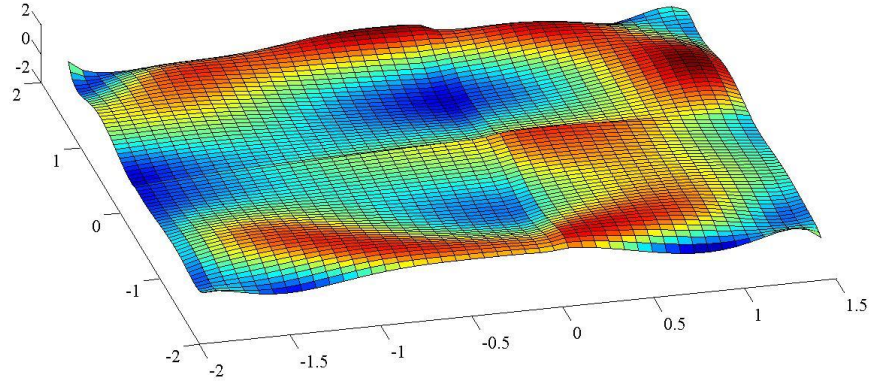
Tablolar incelendiğinde USBS'nin Bayes sınıflandırıcıyla daha iyi sonuç verdiği görülmektedir.

4.2. Signomial Kuvvetlerle Diğer Yöntemler Kullanılarak Yapılan Çalışmalar

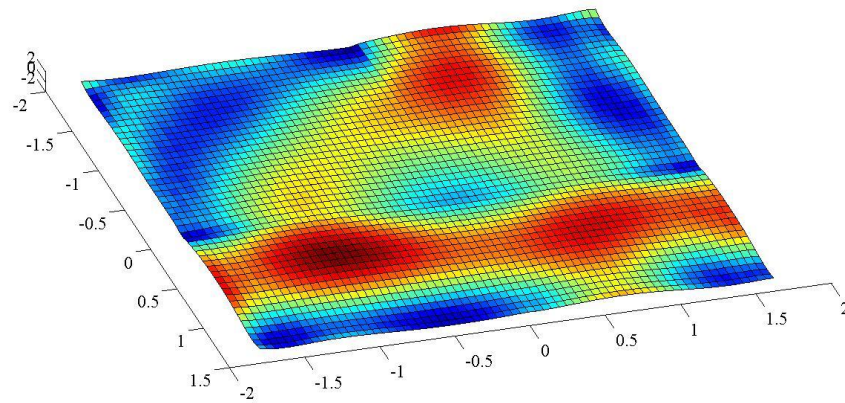
KCE ile modelleme yapılırken alınan kuvvet değerleri bir polinom oluşturması için doğal sayılardan oluşmaktadır. Fakat bu durumun modellemede kısıtlamaya sebep olduğu düşünülüp çalışmada ayrıca kuvvet değerleri reel sayı olabilen signomial (Duffin et al., 1967) kuvvetleri kullanan GA, PSO ve KTGO yöntemleri kullanılmıştır.

4.2.1. Genetik algoritma sonuçları

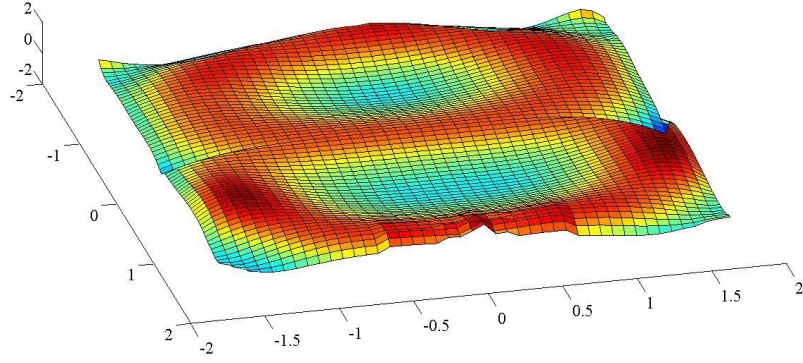
GA kullanılarak oluşturulan popülasyonun reel sayı olabilmesi sağlanabilir. Bu çalışmada 0, 1, 2, ... gibi doğal sayı değerleri yerine 0.13, 1.23, 2.89 gibi reel sayı değerleri kuvvet olarak kullanılmıştır. Çeşitli nesillerden sonra oluşan en iyi popülasyondaki kuvvetler alınarak bu kuvvetlerin en küçük kareler yöntemi ile katsayıları bulunmuştur. Elde edilen denklem rakamları modelleyebilmesine karşın yavaş oluşu ve her seferinde farklı kuvvet değerleri üretmesi sebebiyle sınıflandırma için uygun değildir. Şekil 4.12., Şekil 4.13. ve Şekil 4.14. 'de Arial yazı tipinde GA ile modellenen “3”, “4” ve “8” rakamları görülmektedir.



Şekil 4.12. Arial yazı tipinde “3” rakamının GA ile modellenmesi



Şekil 4.13. Arial yazı tipinde “4” rakamının GA ile modellenmesi

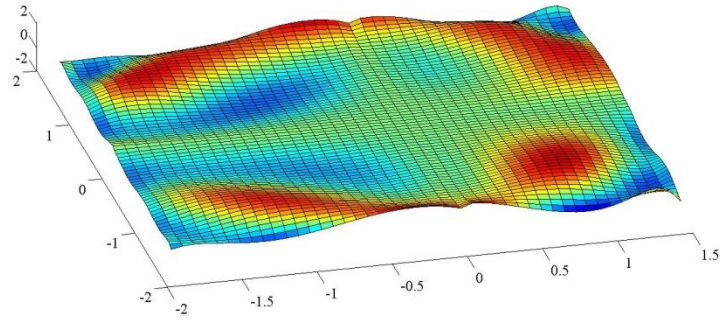


Şekil 4.14. Arial yazı tipinde “8” rakamının GA ile modellenmesi

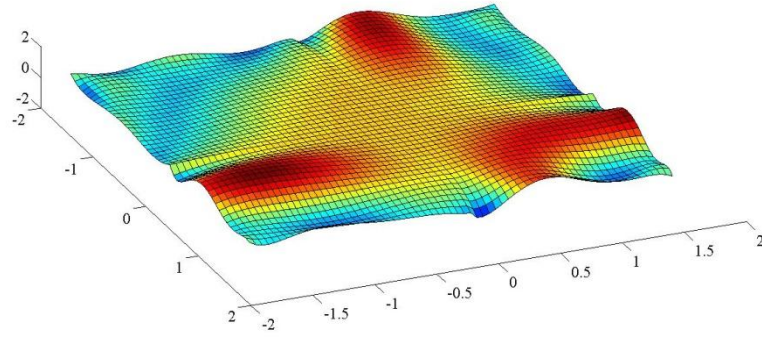
Arial yazı tipi ile test yapılmasındaki amaç bu yazı tipindeki rakamların sade bir görünüme sahip olmalarıdır.

4.2.2. Parçacık sürü optimizasyonu sonuçları

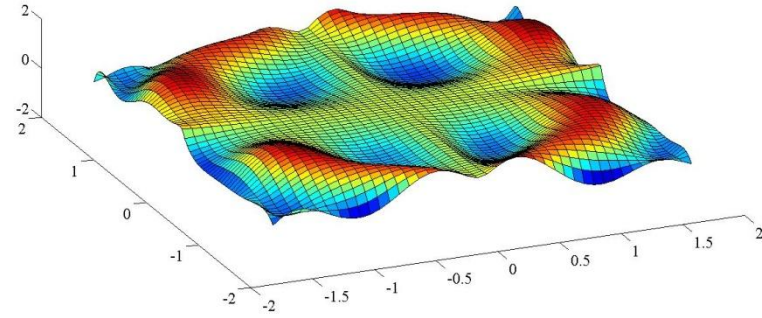
PSO ile modelleme GAYA göre biraz daha hızlı olmasına rağmen sınıflandırma için GA ile aynı sorunları içermektedir. Şekil 4.15., Şekil 4.16. ve Şekil 4.17. ‘de Arial yazı tipinde PSO ile modellenen “3”, “4” ve “8” rakamları görülmektedir.



Şekil 4.15. Arial yazı tipinde “3” rakamının PSO ile modellenmesi



Şekil 4.16. Arial yazı tipinde “4” rakamının PSO ile modellenmesi

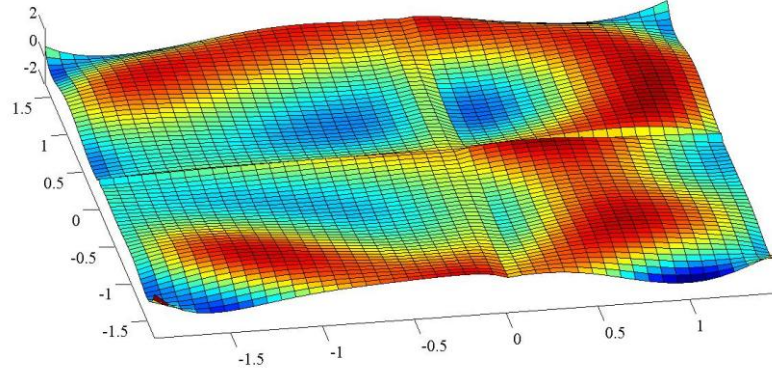


Şekil 4.17. Arial yazı tipinde “8” rakamının PSO ile modellenmesi

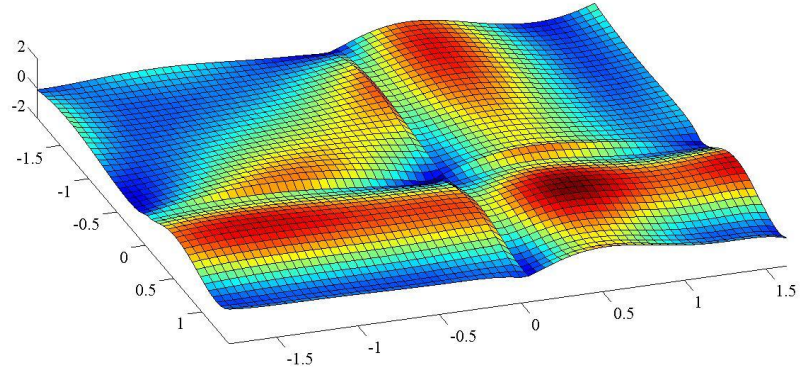
PSO kısmen GAYa göre hızlı çalışmasına rağmen istenilen başarılı modellemeyi tam olarak sağlayamamıştır.

4.2.3. Kümeleme tabanlı global optimizasyon sonuçları

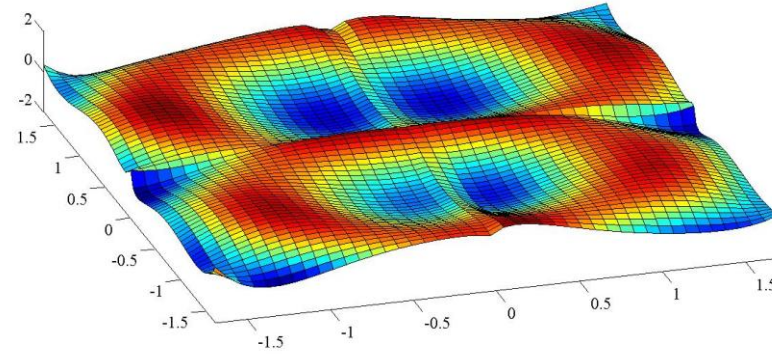
Geliştiren yöntem, mevcut global optimizasyon tekniklerine göre daha hızlı çalışmakta ve daha iyi sonuç vermektedir. Bu durumda yeni nesilde, popülasyon oluşturulurken kümeleme sonucu oluşan küme merkezleri etrafından seçilen elemanlar önemli rol oynar. Bunun yanı sıra sistem belli bir performans değerinde takılı kalıyorsa, yöntem parametre sayısını arttırarak daha iyi sonuçlar üretilmesini sağlamaktadır. Bu yöntemde sınıflandırma için GA ile aynı sorunları içermektedir. Şekil 4.18., Şekil 4.19. ve Şekil 4.20. ‘de Arial yazı tipinde kümeleme tabanlı global optimizasyon (KTGO) ile modellenen üç, dört ve sekiz rakamları görülmektedir.



Şekil 4.18. Arial yazı tipinde “3” rakamının KTGO ile modellenmesi



Şekil 4.19. Arial yazı tipinde “4” rakamının KTGO ile modellenmesi

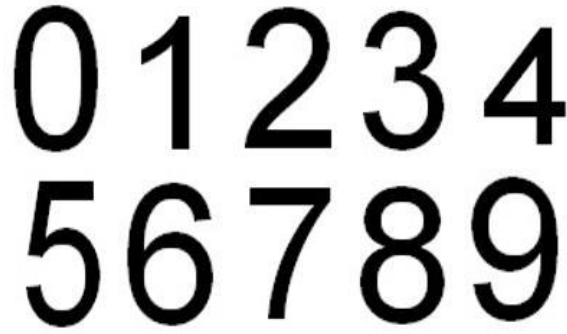


Şekil 4.20. Arial yazı tipinde “8” rakamının KTGO ile modellenmesi

KTGO, diğer GA ve PSO ile karşılaştırıldığında modellemede daha doğru sonuçlar alındığı görülmektedir. Ayrıca diğer yöntemlere göre de biraz daha hızlı çalışmaktadır.

4.3. Elde Edilen Katsayıların Yorumlanması

Elde edilen katsayılar içerisinde bazı kuvvetlere denk gelen katsayıların diğerlerine oranla daha büyük olduğu yani modellemede daha etkin rol oynadığı gözlenmiştir. Bu durumun tespit edilebilmesi için diğer yazı tiplerine göre daha sade örnekler içeren Arial yazı tipi kullanılmıştır. Arial yazı tipindeki rakamlar Şekil 4.21.'de görülmektedir.



Şekil 4.21. Arial yazı tipindeki rakamlar

“0” rakamı için, en yüksek katsayılar x^2y^2 ve x^4y^0 da çıkmıştır. Ayrıca $x^2y^4, x^4y^2, x^6y^0, x^2y^0$ da etkili olmuştur.

“1” rakamı için, x 'li terimlerin katsayıları önemli rol oynamıştır. En etkili katsayılar x^4y^0 ve x^2y^0 olurken $x^3y^0, x^5y^0, x^1y^2, x^1y^0, c$ de etkili olmuşlardır.

“2” rakamı için, x^2y^2 ön plana çıkmakla beraber x^0y^2 de önemli olmuştur. Ağırlıklı olarak y 'li terimlerin katsayıları yüksek çıkmıştır. Bunlar arasında $x^2y^4, x^0y^4, x^0y^3, x^2y^1, x^0y^1, x^2y^0$ bulunmaktadır.

“3” rakamı için, x^2y^2 ön plana çıkmakla beraber x^0y^2, x^0y^4, x^2y^4 de etkili olmuşlardır.

“4” rakamı için, $x^0y^4, x^4y^0, x^0y^2, x^2y^0$ etkili olmuştur. Ayrıca $x^0y^6, x^2y^2, x^0y^3, x^3y^0$ da biraz etkili olmuştur.

“5” rakamı için, x^0y^3, x^0y^2, x^2y^2 etkili olmakla beraber $x^0y^4, x^2y^4, x^0y^5, x^3y^1, x^4y^0, x^0y^1$ de etkili olmuştur.

“6” rakamı için, x^2y^2, x^0y^4, x^0y^2 etkili olmakla beraber $x^2y^4, x^4y^2, x^4y^0, x^0y^3, x^2y^0$ de etkili olmuştur.

“7” rakamı için, x^2y^1, x^0y^4 etkili olmuştur. Ayrıca birazda x^0y^3, x^1y^2 de etkili olmuştur.

“8” rakamı için, $x^2y^2, x^2y^4, x^0y^4, x^0y^2$ etkili olmuştur. Ayrıca x^4y^0 da biraz etkili olmuşlardır.

“9” rakamı için, x^2y^2, x^0y^4, x^0y^2 etkili olmuştur. Ayrıca $x^2y^4, x^4y^2, x^4y^0, x^0y^3$ da biraz etkili olmuştur.

Etkili olarak bulunan kuvvetlere ait katsayılar Tablo 4.9 ‘da görülmektedir.

Tablo 4.9. Arial yazı tipinde etkili kuvvetlere ait katsayılar

Rakam	Kuvvet ve katsayılar									
0	Kuv.	x^2y^2	x^4y^0	x^2y^4	x^4y^2	x^6y^0	x^2y^0			
	Kat.	2.90	2.60	-1.45	-1.59	-1.08	-1.43			
1	Kuv.	x^4y^0	x^2y^0	x^3y^0	x^5y^0	x^1y^2	x^1y^0	c		
	Kat.	2.07	-1.80	-1.14	1.29	-1.01	1.07	0.89		
2	Kuv.	x^2y^2	x^0y^2	x^2y^4	x^0y^4	x^0y^3	x^2y^1	x^0y^1	x^2y^0	
	Kat.	2.61	-1.79	-1.26	1.04	-0.91	-0.83	0.62	-0.80	
3	Kuv.	x^2y^2	x^0y^2	x^0y^4	x^2y^4					
	Kat.	3.39	-2.51	2.34	-2.13					
4	Kuv.	x^0y^4	x^4y^0	x^0y^2	x^2y^0	x^0y^6	x^2y^2	x^0y^3	x^3y^0	
	Kat.	-2.79	-2.19	2.06	1.89	1.32	-1.28	-1.30	-1.07	
5	Kuv.	x^0y^3	x^0y^2	x^2y^2	x^0y^4	x^2y^4	x^0y^5	x^3y^1	x^4y^0	x^0y^1
	Kat.	1.95	-1.92	1.84	1.48	-1.07	-1.19	1.15	1.34	-1.07
6	Kuv.	x^2y^2	x^0y^4	x^0y^2	x^2y^4	x^4y^2	x^4y^0	x^0y^3	x^2y^0	
	Kat.	2.77	2.47	-2.53	-1.67	-1.08	1.68	1.05	-0.94	
7	Kuv.	x^2y^1	x^0y^4	x^0y^3	x^1y^2					
	Kat.	-1.85	1.72	-1.25	-1.21					
8	Kuv.	x^2y^2	x^2y^4	x^0y^4	x^0y^2	x^4y^0				
	Kat.	3.47	-2.43	2.26	-2.87	1.15				
9	Kuv.	x^2y^2	x^0y^4	x^0y^2	x^2y^4	x^4y^2	x^4y^0	x^0y^3		
	Kat.	2.78	2.50	-2.55	-1.69	-1.05	1.63	-1.10		

5. SONUÇ

Bu çalışmada el yazısı rakamların sadece sınır bölgelerini belirten kenarları yerine bir bütün olarak 3-boyutlu olarak kapalı cebirsel eğriler ile modellenip uygun sınıflandırıcı yöntem ile sınıflandırılması üzerinde durulmuştur. Bütün rakamlar 8. dereceden kapalı cebirsel eğrilerle modellenmiş olup öznitelik sayısında çıkan fazlalık USBS'nin yetenekleriyle azaltılmaya çalışılmıştır. MNIST veritabanı üzerinde alınan %92.87 lik tanıma oranı ile özellikle öznitelik sayısı azaltıldığında elde edilen %89.98 lik tanıma oranları umut vaat etmektedir. Ayrıca geliştirilen kümeleme tabanlı global optimizasyon yöntemi de global optimizasyon problemlerine yeni bir soluk getirmekle birlikte popülasyon oluşturmada kümeleme yöntemi kullanması büyük bir avantaj oluşturmaktadır. Çalışmalar ilerisi için umut vaat etmekte olup yöntemin geliştirilmesiyle birlikte her rakamın birer denklemle ifade edilebileceği açıktır.

KAYNAKLAR

- Akyılmaz, O., Ayan, T., 2006. Esnek Hesaplama Yöntemlerinin Jeodezide Uygulamaları. İTÜ Dergisi, 5 (1), 261-268.
- Alataş, B., Akın, E., Özer, A.B., 2007. Kaotik Haritalı Parçacık Sürü Optimizasyon Algoritmaları. XII. Elektrik, Elektronik, Bilgisayar, Biyomedikal Mühendisliği Ulusal Kongresi, Eskişehir.
- Alpaydin, E., 2004. Introduction to Machine Learning. The MIT Press.
- Arslan, A., İnce, R., 1996. The Neural Network Approximation to The Size Effect in Fracture of Cerametic Materials. Engineering Fracture Mechanics, 54 (2), 249-261.
- Bahr, B., Xiao, X., Krishnan, K., 2001. A Real-Time Scheme of Cubic Parametric Curve Interpolations for CNC Systems. Computers in Industry, 45 (3), 309-317.
- Başaran, M., 2009. Hız Kontrol Sistemleri ile Beslenen Asenkron Motorlarda Mekanik Hataların Belirlenmesi. Anadolu Üniversitesi Elektrik-Elektronik Mühendisliği Anabilim dalı, Yüksek Lisans Tezi, 62s, Eskişehir.
- Beasley, D., Bull, D.R., Martin, R.R., 1993. An Overview of Genetic Algorithm:Part1- Fundamentals. Univ. of Michigan, 15 (2), 258-269.
- Bézier, P.E., 1971. Example of an Existing System in the Motor Industry: The Unisurf System. Proc. Roy. Soc. A321, 207-218, London.
- Bhattacharya, R., Singla, P., 2006. Nonlinear Trajectory Generation Using Global Local Approximations. Proceedings of the 45th IEEE Conf. On Decision and Control, pp. 6802-6807.
- Blane, M.M., Lei, Z., Çivi, H., Cooper, D.B., 2000. The 3L Algorithm for fitting Implicit Polynomial Curves and Surfaces to Data. IEEE Trans. Pattern Analysis and Machine Intelligence, 22 (3).
- Bolat, B., Erol, K., İmrak, C., 2004. Genetic Algorithms in Engineering Applications and the Function of Operators. Sigma Journal of Engineering and Natural Sciences, 4, 264-270.
- Candemir, S., 2004. Kapalı Cebirsel Eğrilerle Karakter Tanıma. Gebze Yüksek Teknoloji Enstitüsü, Mühendislik ve Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 64s, Gebze.

- Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R., 2001. Reconstruction and Representation of 3D Objects with Radial Basis Functions. ACM SIGGRAPH 2001, USA.
- Cetiřli, B., 2005. El Yazısı Karakter Tanıma: Dalgacık Moment Özniteliklerinin Yenilenen ANFIS ile Sınıflandırılması. Osmangazi Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi, Eskişehir.
- Cetiřli, B., 2006. Öznitelik Seçiminde Dilsel Kuvvetli Sinir Bulanık Sınıflayıcı Kullanımı. Eskişehir Osmangazi Üniversitesi Müh-Mim. Fak. Dergisi, 19 (2), 109-129.
- Cetiřli, B., Edizkan, R., 2011. Estimation of Adaptive Neuro-Fuzzy Inference System Parameters with the Expectation Maximization Algorithm and Extended Kalman Smoother. Neural Computing and Applications, 20 (3), 403-415.
- Chung, Y., Wong, M., 1997. Handwritten Character Recognition by Fourier Descriptors and Neural Network. IEEE TENCON, Speech and Image Technologies for Computing and Telecommunications.
- Civalek, Ö., 1999. Dairesel Plakların Nöro-Fuzzy Tekniğı ile Analizi. DEÜ Mühendislik Fakültesi Fen ve Mühendislik Dergisi, 1 (2), 13-31.
- Civi, H., 1997. Implicit Algebraic Curves and Surfaces for Shape Modelling and Recognition. Ph.D. Thesis, Bogazici University.
- Çelebi, M., 2003. Yeni Bir Genetik Algoritma Yaklaşımı ile Çıkık Kutuplu Senkron Makinenin Ağırlık Optimizasyonu. Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi, 125s, İstanbul.
- Decoste, D., Scholkopf, B., 2002. Training Invariant Support Vector Machines. Mach. Learn., 46 (1-3), 160–190.
- Demir, Y.K., 1997. Yapay Sinir Ağları ile Ulaştırma Taleplerinin Modellenmesi. İTÜ Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, İstanbul.
- Domingos, P., Pazzani, M., 1997. On the Optimality of the Simple Bayesian Classifier Under Zero-One Loss. Machine Learning, 29, 103–137.
- Duffin, R.J., Peterson, E.L., Zener, C.M., 1967. Geometric Programming: Theory and Application. Wiley, 278p. New York.

- Fırat, M., 2002. Doğrusal ve Kıvrımlı Akarsulara Yerleştirilen Köprü Ayakları Etrafında Meydana Gelen Oyulma Derinliğinin Yapay Sinir Ağları ile Belirlenmesi. Pamukkale Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Denizli.
- Fırat, M., Güngör, M., 2004. Askı Madde Konsantrasyonu ve Miktarının Yapay Sinir Ağları ile Belirlenmesi. İMO Teknik Dergi, 219, 3267-3282.
- Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley.
- Gürsu, B., İnce, M.C., 2007. Genetik Algoritmalar ile Yüksek Gerilim İstasyonlarında Optimum Topraklama Ağı Tasarımı. Fırat Üniversitesi Fen ve Müh. Bil. Dergisi, 19 (4), 511-524.
- Hand, D.J., Yu, K., 2001. Idiot's Bayes - Not So Stupid After All?. International Statistical Review, ISSN 0306-7734, 69 (3), 385-399.
- Haykin, S.S., 1994. Neural Networks: A Comprehensive Foundation. NY: Mcmillan Press.
- Holland, J., 1975. Adaptation in Natural and Artificial Systems. University of Michigan Pres, Ann Arbor, Mich.
- Jain, A.K., Mao, J., Mohiuddin, K., 1996. Artificial Neural Networks: A Tutorial. Accepted to appear in IEEE Computer Special Issues on Neural Computing.
- Jang, J.-S.R., Sun, C.T., Mizutani, E., 1997. Neuro-Fuzzy and Soft Computing. Prentice Hall, Upper Saddle River.
- Jeong, C., Jeong, D., 1999. Handwritten Digit Recognition Using Fourier Descriptors and Contour Information. IEEE TENCON, 6 (99).
- Kakıcı, A., 2009. Yapay Sinir Ağlarının Mimarisi ve Yapı Elemanları. <http://www.ahmetkakici.com/yapay-sinir-aglari/yapay-sinir-aglarinin-mimarisi-ve-yapi-elemanlari/>. Erişim Tarihi: 03.05.2012.
- Karaca, N., 2007. Alçak Çözünürlüklü Fotoğrafların Görüntülenmesi ve Bunların Optimizasyonu ile İlgili Bir Çalışma. Ege Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 111s, İzmir.
- Kennedy, J., Eberhart, R.C., 1995. Particle Swarm Optimization. Proc. of the IEEE Int. Conference on Neural Networks, 4, 1942-1948.
- Keren, D., Cooper, D., Subrahmonia, J., 1994. Describing Complicated Objects by Implicit Polynomials. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16 (1), 38-53.

- Keyzers, D., 2006. Comparison and Combination of State-of-the-art Techniques for Handwritten Character Recognition: Topping the MNIST Benchmark. Tech. Rep., Technical University Kaiserslautern.
- Kussul, E., Baidyk, T., 2004. Improved Method of Handwritten Digit Recognition Tested on MNIST Database. *Image Vision Comput.*, 22 (12), 971–981.
- Kussul, E., Baidyk, T., Wunsch, II. D.C., 2005. Image Recognition Systems with Permutative Coding. *Proceedings of International Joint Conference on Neural Networks (IJCNN2005)*, pp. 1788–1793, Montreal, Canada.
- Lauer, F., Bloch, G., Suen, C.Y., 2005. Increasing the Recognition Rate of Handwritten Digit Classifiers. Technical Report, CENPARMI, Concordia University.
- LeCun, Y., Bottou, L., Bengio, Y., Haffener, P., 1998. Gradient-Based Learning Applied to Document Recognition. *Proc. of the IEEE*, 86 (11), 2278-2324.
- Lisboa, P.J.G., Perantonis, S.J., 1991. Invariant Digit Recognition by Zernike Moments and Third-Order Neural Networks. *Second Int. Conf. Artificial Neural Networks*, pp. 18-20.
- Liu, C.L., Nakashima, K., Sako, H., Fujisawa, H., 2002. Handwritten Digit Recognition Using State-of-The-Art Techniques. *Proceedings of the 8th International Workshop on Frontiers in Handwritten Recognition*, pp. 320–325, Ontario, Canada.
- Maron, M.E., 1961. Automatic Indexing: An Experimental Inquiry. *Journal of the ACM (JACM)*, 8 (3), 404–417.
- McCallum, A., Nigam, K., 1998. A Comparison of Event Models for Naive Bayes Text Classification. In *AAAI/ICML-98 Workshop on Learning for Text Categorization*, Technical Report WS-98-05, AAAI Pres., pp. 41-48.
- Menzies, T., Greenwald, J., Frank, A., 2007. Data Mining Static Code Attributes to Learn Defect Predictors. *IEEE Transactions on Software Engineering*, 33 (1), 2–13.
- Mitchell, M., 1998. *Introduction to Genetic Algorithms*. Bradford Pres, pp. 10-48, London.
- Mozina, M., Demsar, J., Kattan, M., Zupan, B., 2004. Nomograms for Visualization of Naive Bayesian Classifier. In *Proc. of PKDD*, pp. 337-348.

- Öden, C., Erçil, A., Yıldız, V.T., Kırmızıta, H., Büke, B., 2001. Hand Recognition Using Implicit Polynomials and Geometric Features. Audio and Video-Based Biometric Person Authentication Lecture Notes in Computer Science, 2091, 336-341.
- Özsağlam, M.Y., Çunkaş, M., 2008. Optimizasyon Problemlerinin Çözümü için Parçacık Sürü Optimizasyonu Algoritması. Politeknik Dergisi, 11 (4), 299-305.
- Özsoy, İ., Fırat, M., 2004. Kirişsiz Döşemeli Betonarme Bir Binada Oluşan Yatay Deplasmanın Yapay Sinir Ağları ile Tahmini. DEÜ Mühendislik Fakültesi, Fen ve Mühendislik Dergisi, 6 (1), 51-63.
- Rish, I., 2001. An Empirical Study of the Naive Bayes Classifier. IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence.
- Rouhani, M., Sappa, A.D., 2011. Implicit B-Spline Fitting Using the 3L Algorithm. 18th IEEE International Conference on Image Processing (ICIP), pp. 893-896.
- Sahin, T., Unel, M., 2004. Globally Stabilized 3L Curve Fitting. Image Analysis and Recognition Lecture Notes in Computer Science, 3211, 495-502.
- Sahin, T., Unel, M., 2005. Fitting Globally Stabilized Algebraic Surfaces to Range Data. Tenth IEEE International Conference on Computer Vision (ICCV'05), 2, 1083-1088.
- Simard, P.Y., Steinkraus, D., Platt, J.C., 2003. Best Practice for Convolutional Neural Networks Applied to Visual Document Analysis. Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR2003), 2, 958-962, Edinburgh, Scotland.
- Şişeci, M., 2012. Traverten Plaka Taşlarda Sınıfların Kümeleme Yöntemleri İle Belirlenmesi. Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 63s, Isparta.
- Tarel, J.P., Çivi, H., Cooper, D.B., 1998. Pose Estimation of Free-Form 3D Objects without Point Matching using Algebraic Surface Models. IEEE Workshop MB3IA'98, pp. 13-21.
- Taron, M., Paragios, N., Jolly, M.P., 2005. Modelling Shapes with Uncertainties : Higher Order Polynomials, Variable Bandwidth Kernels and non Parametric Density Estimation. Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05), 2, 1659-1666.

- Tasdizen, T., Tarel, J.P., Cooper, D.B., 2000. Improving the Stability of Algebraic Curves for Applications. *IEEE Transactions in Image Processing*, 9 (3), 405-416.
- Taubin, G., 1991. Estimation of Planar Curves, Surfaces and Nonplanar Space Curves defined by Implicit Equations, with Applications to Edge and Range Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13 (11), 1115-1138.
- Teow, L.N., Loe, K.F., 2002. Robust Vision-Based Features and Classification Schemes for Off-Line Handwritten Digit Recognition. *Pattern Recognition*, 35 (11), 2355–2364.
- Trier, O.D., Jain, A.K., Taxt, T., 1995. Feature Extraction Methods for Character Recognition: A survey. Revised July 19.
- Turhan, B., Bener, A., 2007. Yazılım Hata Kestirimi İçin Kaynak Kod Ölçütlerine Dayalı Bayes Sınıflandırması. EMO III. Ulusal Yazılım Mühendisliği Sempozyumu, Ankara.
- Unel, M., 1999. Polynomial Decompositions for Shape Modelling, Object Recognition and Alignment. PhD Thesis, Brown University.
- Unel, M., Wolovich, W.A., 2000. On the Construction of Complete Sets of Geometric Invariants for Algebraic Curves. *Advances in Applied Mathematics*, 24 (1), 65-187.
- Uygunoğlu, T., Yurtcu, Ş., 2006. Yapay Zeka Tekniklerinin İnşaat Mühendisliği Problemlerinde Kullanımı. *Yapı Teknolojileri Elektronik Dergisi (YTED)*, (Teknik Not), 2 (1), 61-70.
- Uzun, E., 2007. İnternet Tabanlı Bilgi Erişimi Destekli Bir Otomatik Öğrenme Sistemi. Trakya Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi, 123s, Edirne.
- Ülker, M., Civalek, Ö., 2002. Yapay Sinir Ağları ile Eksenel Yüklü Kolonların Burkulma Analizi. *Turkish J. Eng. Env. Sci.*, 26, 117-125.
- Ünsal, F.B., 2009. İki Boyutlu Doğrusal Koordinat Dönüşümleri. TMMOB Harita ve Kadastro Mühendisleri Odası 12. Türkiye Harita Bilimsel ve Teknik Kurultayı, Ankara.
- Wilke, D.N., 2005. Analysis of the Particle Swarm Optimization Algorithm. University of Pretoria, Mechanical and Aeronautical Engineering, Master Thesis.

Yazıcıoğlu, A.Y., Çallı, B., Ünel, M., 2009. Cebirsel Eğriler Kullanarak İmge Tabanlı Görsel Geri Beslemeli Denetim. Otomatik Kontrol Ulusal Toplantısı (TOK'09), İstanbul.

Yurtoğlu, H., 2005. Yapay Sinir Ağları Metodolojisi ile Öngörü Modellenmesi: Bazı Makroekonomik Değişkenleri için Türkiye Örneği. DPT – Uzmanlık Tezleri.

ÖZGEÇMİŞ

Adı Soyadı : İhsan PENÇE

Doğum Yeri ve Yılı : Mersin 1987

Medeni Hali : Bekar

Yabancı Dili : İngilizce



Eğitim Durumu (Kurum ve Yıl)

Lise : Dumlupınar Lisesi, MERSİN 2001-2005

Lisans : Süleyman Demirel Üniversitesi 2006-2010

Yüksek Lisans : Süleyman Demirel Üniversitesi 2010-...

Çalıştığı Kurumlar ve Yıl:

Süleyman Demirel Üniversitesi Senirkent Meslek Yüksekokulu (Öğretim Elemanı)
02/2012-...