

OTOMATİK PARMAKİZİ TANIMA SİSTEMLERİNDE ÖZELLİK NOKTALARININ TESPİTİNDE YAPAY SİNİR AĞLARININ KULLANILMASI

Necla ÖZKAYA *, Şeref SAĞIROĞLU**

*Erciyes Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 38039/Talas/Kayseri

**Gazi Üniversitesi, Mühendislik Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, 06570/Maltepe/Ankara

Geliş Tarihi : 04.03.2005

ÖZET

Otomatik parmakizi tanıma sistemleriyle kimliklendirme yapılırken, özellik noktaları olarak bilinen parmakizi resimlerdeki hat çizgisi karakteristiklerinden ve bunların birbirleriyle olan ilişkilerinden faydalanılır. Bu yüzden giriş parmakizi resminden özellik noktalarının sorunsuz, güvenilir, hızlı ve otomatik olarak elde edilebilmesi kimliklendirme için çok önemlidir. Bu çalışmada, parmakizi tanımadaki kullanılan özellik noktalarının tespit edilmesine yönelik yapay sinir ağları temelli yeni bir yaklaşım geliştirilmiş ve sunulmuştur. Elde edilen sonuçlar parmakizi resminde özellik noktalarının bulunmasında yapay sinir ağlarının başarılı olduğunu göstermiştir.

Anahtar Kelimeler : Biyometri, Parmakizi, Özellik noktaları, Yapay sinir ağları, Çok katlı perseptronlar.

MINUTIAE EXTRACTION BASED ON ARTIFICIAL NEURAL NETWORKS FOR AUTOMATIC FINGERPRINT RECOGNITION SYSTEMS

ABSTRACT

Automatic fingerprint recognition systems are utilised for personal identification with the use of comparisons of local ridge characteristics and their relationships. Critical stages in personal identification are to extract features automatically, fast and reliably from the input fingerprint images. In this study, a new approach based on artificial neural networks to extract minutiae from fingerprint images is developed and introduced. The results have shown that artificial neural networks achieve the minutiae extraction from fingerprint images with high accuracy.

Key Words : Biometrics, Fingerprint, Minutiae extraction, Artificial neural network, Multilayer perceptrons.

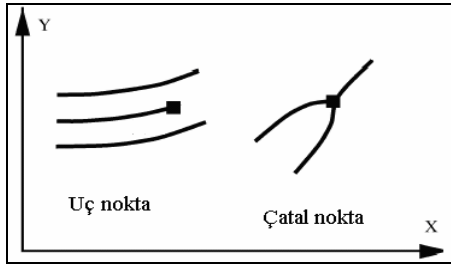
1. GİRİŞ

Günümüzde bilgisayar teknolojilerindeki hızlı gelişme ile birlikte biyometrik sistemler hayatımızda birçok problemin çözümünde kullanılmakta özellikle kişilerin kimliklendirilmesi gereken işlemlerin daha hızlı, kolay ve güvenilir bir şekilde gerçekleştirilmesini sağlamaktadır. Parmakizi

tanıma, retina ve iris tanıma, el geometrisi tanıma, ses tanıma, yüz tanıma, DNA tanıma, imza tanıma gibi tekniklerin tamamını kapsayan biyometri bilimi, yüksek seviyede güvenlik gerektiren alanlarda, giriş çıkışlarda kimlik kontrolünün gerektiği otomatik personel devam kontrol sistemleri gibi uygulamalarda mükemmel çözüm olmakta ve sosyal hayatta oldukça sık karşılaşılan çok hızlı gelişen ve benimsenen bir teknoloji olarak karşımıza

çıkılmaktadır (Jain et al., 1997a, b; Halici et al., 1999; Jain et al., 1999). Güvenilirliği, sistem performansı, düşük maliyeti ve kullanım kolaylığıyla dikkat çeken otomatik parmakizi tanıma sistemleri (OPTS), biyometrik teknikler arasında en yaygın kullanılan teknoloji olarak dikkat çekmektedir (Jain et al., 1997b; Halici et al., 1999; Jain et al., 1999).

Bir OPTS’de parmakizi tanıma genellikle parmakizinde bulunan özellik noktalarının ve bunlara ait parametrelerin karşılaştırılması esasına dayanır (Jain et al., 1997b; Greenberg et al., 2000; Espinosa-Duro, 2002). Parmakizine ait bir çok özellik noktası olmasına rağmen, OPTS’lerde parmakizi tanıma ve karşılaştırmada yalnızca özellik noktaları (minutiae) olarak bilinen uç ve çatal noktalar kullanılarak başarılı sonuçlar elde edilebilmektedir (Jain et al., 1997a,b; Hong et al., 1998; Halici et al., 1999; Jain et al., 1999; Greenberg et al., 2000; Luo et al., 2000; Rusyn et al., 2001; Espinosa-Duro, 2002; Saatci ve Tavsanoglu, 2002). Uç ve çatal noktalar, parmakizi resminde bulunan hat çizgilerinin aniden sonlanması veya çatallaşması şeklinde oluşmaktadır (Halicı et al., 1999; Jain et al., 1997a, b; Jain et al., 1999; Rusyn et al., 2001; Luo et al., 2000; Hong et al., 1998; Espinosa-Duro, 2002). OPTS’de kullanılan iki önemli özellik noktası Şekil 1’de verilmektedir.



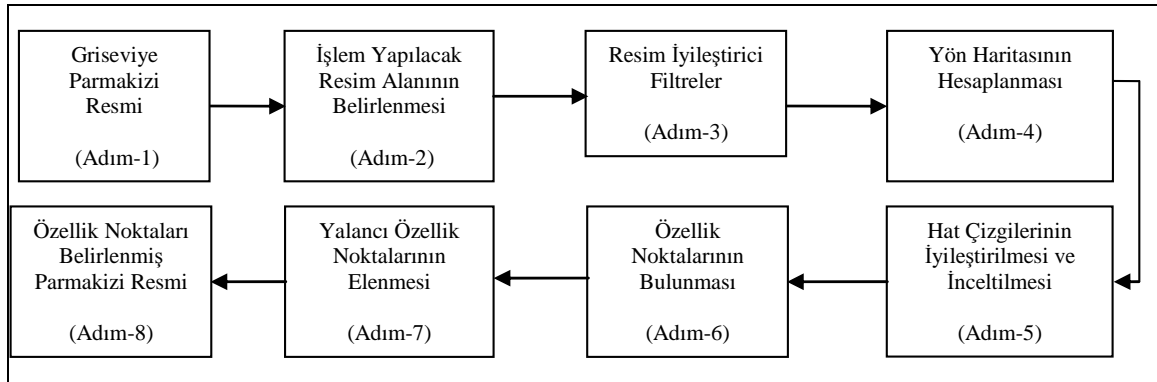
Şekil 1. OPTS’de kullanılan özellik noktaları

Kimliklendirme işleminde bu özellik noktalarının tipleri, yerel yön değerleri, referans noktaya göre açıları, uzaklıkları ve bu nokta ile aralarından geçen

hat çizgisi sayısı gibi parametreler kullanılır (Jain et al., 1997b; Halici et al., 1999; Greenberg et al., 2000; Espinosa-Duro, 2002). Bu değerlerin doğru olarak hesaplanması OPTS’nin performansı ve güvenilirliği açısından çok önemlidir (Jain et al., 1997a,b; Jain et al., 1999; Halici et al., 1999; Greenberg et al., 2000; Özkaya, 2003).

2. ÖZELLİK NOKTALARININ TESPİTİ

Özellik noktalarının tespitine yönelik literatür incelendiğinde farklı yaklaşımların sunulduğu, konuyla ilgili olarak gerçekleştirilen algoritmaların genellikle inceltmiş parmakizi resmi üzerinde işlem yaptığı ve bu işlemin temelde üç adımdan oluştuğu görülmektedir. Bunlar özellik noktalarını bulmaya yönelik ön işlemler, özellik noktalarının bulunması ve yalancı özellik noktalarının elimine edilmesi şeklinde sıralanabilir (Özkaya, 2003). Özellik bulmaya yönelik ön işlemler resimlerin üzerinde işlem yapılacak bölgesinin belirlenip arkaplandan ayrılması, temizlenip iyileştirilmesi ve inceltmesi olarak ifade edilebilir. Özellik noktalarının tespiti, inceltmiş resim üzerinde özellik noktası olabilecek noktaların belirlenmesi olarak tarif edilebilir. Özellik bulma işlemi sonrasındaki işlemler ise elde edilen noktaların gerçek özellik noktası olup olmadığının araştırılması şeklindedir. (Xiao and Raafat, 1991; Ongun, 1995; Alkaya, 1998; Halici et al., 1999; Jain et al., 1999; Özkaya, 2003). Özellik noktalarını bulmaya yönelik genel bir sistemin işlem adımları Şekil 2’de verilmektedir. Özellik noktalarını bulmaya yönelik bir algoritma Espinosa-Duro (2002) tarafından sunulmaktadır. Sözü edilen algoritmada $P(x,y)$ inceltmiş hat çizgisi üzerinde bir piksel ve N_0, \dots, N_7 pikselleri $P(x,y)$ noktasında bulunan pikselin komşu pikselleri olmak üzere uç ve çatal noktalar Eşitlik 1’de verildiği gibi tarif edilmektedir.



Şekil 2. Özellik noktaları bulmaya yönelik genel bir sistemin işlem adımları.

$$P(x, y) = \left\{ \begin{array}{l} \sum_{i=0}^7 N_i = 1 \rightarrow P(x, y): U\check{c} _nokta \\ \sum_{i=0}^7 N_i > 2 \rightarrow P(x, y): \check{C}atal _nokta \end{array} \right\} \quad (1)$$

Aynı amaca yönelik başka bir yaklaşımda (Xiao and Raafat, 1991), P ilgilenilen piksel ve P_1, \dots, P_8 ise P'nin etrafında bulunan sekiz adet komşu piksel olmak üzere uç ve çatal noktalar Eşitlik 2'de verildiği gibi ifade edilmektedir.

$$P(x, y) = \left\{ \begin{array}{l} \sum_{i=1}^8 |P(i+1) - P(i)| = 2 \rightarrow P(x, y): U\check{c} _nokta \\ \sum_{i=1}^8 |P(i+1) - P(i)| = 6 \rightarrow P(x, y): \check{C}atal _nokta \end{array} \right\} \quad (2)$$

Literatürde özellik noktalarının bulunmasına yönelik griseviye resim üzerinde işlem yapan algoritmalar (Sagar and Beng, 1999a; Sagar and Beng, 1999b; Bhanu et al., 2000; Liu et al., 2000) ile YSA ve bulanık mantıkla gerçekleştirilmiş çalışmalar mevcuttur (Ongun, 1995; Maio and Maltoni, 1998; Sagar and Beng, 1999b; Jin et al., 2002;). Bu algoritmaların, gürültülü ve düşük kaliteli resimlerde inceltmiş resim üzerinde çalışan algoritmalara kıyasla daha iyi sonuç verdiği rapor edilmektedir (Sagar and Beng, 1999a; Sagar and Beng, 1999b; Bhanu et al., 2000; Liu et al., 2000;).

Özellik noktalarının tespit edilmesinden sonra özellik noktaları listesinden gerçek özellik noktalarının seçilmesi gerekmektedir. Bu konuda yapılan çalışmalar incelendiğinde işlemin genellikle maske kullanılarak gerçekleştirildiği görülmektedir (Xiao and Raafat, 1991). $N \times N$ boyutundaki bir pencerenin tüm pikseller üzerinde kaydırılarak merkez piksel değerinin değişmesine dayanan bu algoritmaların dezavantajı ise benzer özellik gösteren yalancı özellik noktalarının saptanabileceği ve olabilecek tüm yalancı özellik nokta çeşitleri için ayrı bir maske tanımlanma zorunluluğunun olmasıdır (Xiao and Raafat, 1991; Özkaya, 2003). Gerçekleştirilen çalışmaların bazılarında gerçek özellik noktalarının belirlenmesi konusunda her bir özellik noktasına ait bir takım parametreler hesaplanarak sonuca gidilmiş ve başarılı sonuçlar elde edilebilmiştir (Xiao and Raafat, 1991; Özkaya, 2003).

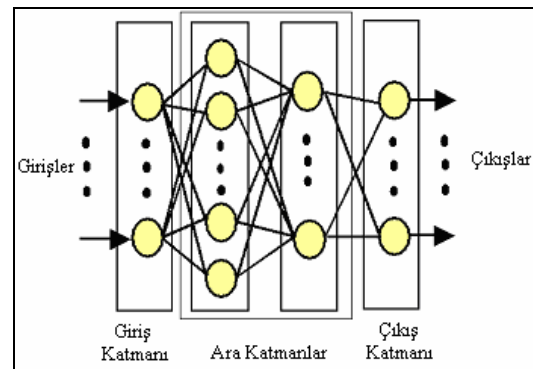
3. YAPAY SİNİR AĞLARI

Yapay sinir ağları (YSA), Yapay sinir ağları (YSA), öğrenme yeteneği, kolayca farklı problemlere uyarlanabilirliği, genelleme yapabilmesi, daha az

bilgi gerektirmesi, paralel yapısından dolayı hızlı işlem yapabilme yeteneği ve uygulayıcıların sistemin girişi ile çıkışı arasındaki ilişkileri formülize etme mecburiyetinin olmayışı gibi pek çok üstünlüğünden dolayı, OPTS'lerde kullanılmaktadır (Ongun, 1995; Maio and Maltoni, 1998; Sagar and Beng, 1999a; Sagar and Beng, 1999b; Jin et al., 2002).

YSA, bir sisteme ilişkin çeşitli parametrelere bağlı olarak tanımlanan girişler ve çıkışlar arasında ilişki kurabilme yeteneğine sahiptir. Bu ilişkinin doğrusal olması zorunlu değildir. Ayrıca YSA, çıkış değerleri bilinmeyen tanımlanmış sistem girişlerine de uygun çıkışlar üretebilmekte, böylece çok karmaşık problemlere bile mükemmel çözüm olabilmektedir.

Literatürde, MLP, LVQ, ART ve SOM gibi birçok YSA yapısı mevcuttur (Sağıroğlu ve ark., 2003). Sunulan bu çalışmada birçok alana başarıyla uygulanmış Çok Katlı Perseptron (MLP) modeli kullanılmıştır. Birçok öğrenme algoritmasının bu ağı eğitmede kullanılabilir olması, bu modelin tercih edilme sebebi olarak açıklanabilir. Bir MLP modeli, bir giriş, bir veya daha fazla ara ve bir de çıkış katmanından oluşur. Bir katmandaki tüm işlem elemanları bir üst katmandaki tüm işlem elemanlarına bağlıdır. Giriş katmanındaki işlem elemanları tampon gibi davranırlar ve giriş sinyalini ara katmandaki işlem elemanlarına dağıtırlar. Ara katmandaki her bir işlem elemanının çıkışı, kendine gelen tüm giriş sinyalleri ile sinyalleri takip eden bağlantı ağırlıklarının çarpımlarının toplamına bir çıkış fonksiyonu uygulanarak elde edilebilir. Bu fonksiyon, basit bir eşik fonksiyonu olabileceği gibi bir sigmoid veya hiperbolik tanjant fonksiyonu da olabilir. Diğer katmanlardaki işlem elemanlarının çıkışları da aynı şekilde hesaplanır. Kullanılan eğitme algoritmasına göre, ağı çıkışı ile arzu edilen çıkış arasındaki hata en aza düşünceye kadar ağı ağırlıkları değiştirilir (Sağıroğlu ve ark., 2003). Genel olarak bir MLP yapısı, Şekil 3'te gösterilmiştir.



Şekil 3. Bir MLP YSA modeli

Bu çalışmada YSA'nın eğitimi için momentum ve öğrenme oranı adaptif geriyayılım (MGY) öğrenme algoritması kullanılmıştır. MGY algoritması, ağırlıkları ve giriş değerleri mevcut ve türevi alınabilir transfer fonksiyonuna sahip olan tüm YSA'ların eğitiminde kullanılabilen bir öğrenme algoritmasıdır. MGY, YSA ağırlıklarının ve bias değerlerinin, adaptif öğrenme oranı ve momentum gözönünde bulundurularak değiştirilmesi esasına dayanır. Bu ise Eşitlik 3a ve 3b'de gösterildiği gibi ifade edilebilir (Anon., 2006).

$$w(t+1) = w(t) + \Delta w(t+1) \quad (3a)$$

$$\Delta w(t+1) = \alpha\beta\delta + \beta\Delta w(t) \quad (3b)$$

Verilen Eşitliklerde $w(t+1)$, ağırlık ve bias değerlerini; $w(t)$, ağırlık ve bias'ın bir önceki adımdaki değerlerini; $\Delta w(t+1)$, ağırlık değişimini; çıkış ve arakatmanlarda farklı şekilde hesaplanan δ ise sistemin çıkış ve arakatmana göre ağırlık değişimini temsil etmektedir.

MGY öğrenme algoritması ile sistem eğitilirken gözönünde bulundurulan parametreler, maksimum epok sayısı, arzu edilen performans değeri, öğrenme oranı (α), öğrenme hızının artma oranı (α_i), öğrenme hızının azalma oranı (α_d), öğrenmedeki maksimum başarısızlık, maksimum performans artışı, momentum sabiti (β), minimum hata değişimi ve grafiksel gösterimde kullanılacak olan epoklar arası mesafe şeklinde sıralanabilir.

Öğrenme sırasında her bir epok için değerlendirme yapılırken performans arzu edilen şekilde artarsa, öğrenme oranı, α_i faktörü gözönünde bulundurularak artırılır. Performans, maksimum performans artış değerinden daha fazla artarsa,

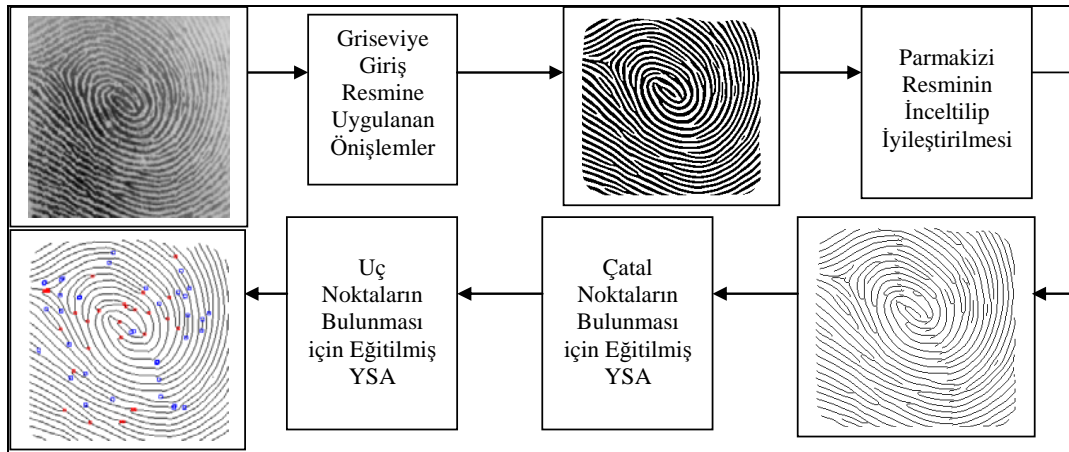
öğrenme oranı, α_d faktörü gözönünde bulundurularak ayarlanır ve performanstaki bu istenmeyen ani değişim YSA'yı etkilemez.

Eğitim işlemi tamamlandığında YSA modeli, çıkış değerlerini, eğitimle ilgili parametreleri ve son epokta elde edilen hata değerlerini verir. Eğitimle ilgili parametreler, sistemin eğitimi için kullanılan epok sayısı, eğitim performansı, geçerlilik performansı, test performansı ve adaptif öğrenme oranı olarak sıralanabilir. Maksimum epok sayısının tamamlanması, eğitim için belirlenen sürenin sonuna gelmesi, sistem performansının istenilen seviyeye gelmesi, eğitim sırasında minimum hata değişim oranının altına düşülmesi veya geçerlilik faktörü kullanılıyorsa bu değerler öğrenmedeki maksimum başarısızlık sayısını aşması durumlarından biri gerçekleştiğinde eğitim işlemi sona erer (Anon., 2006; Sağıroğlu ve ark., 2003).

MGY algoritmasında öğrenme başarısı öğrenme oranı ile yakından ilgilidir. Eğer öğrenme oranı çok büyük ise algoritma salınım yapar ve kararlı davranamaz. Eğer öğrenme oranı çok küçük ise algoritmanın sonuca yakınsaması uzun sürer. Adaptif bir öğrenme oranı, öğrenme adımı ne kadar büyük olursa olsun öğrenmenin kararlı olmasını sağlar. Bu momentum ağırlık yerel optimum noktaları aşmasını sağladığı gibi hatanın düşmesine de yardımcı olur (Sağıroğlu ve ark., 2003).

4. ÖNERİLEN YÖNTEM

Sunulan çalışmada, parmakizi resmine ait özellik noktalarının bulunmasına yönelik YSA temelli bir sistem geliştirilmiştir. Gerçekleştirilen sistemin çalışma prensibi Şekil 4'te verilmektedir.



Şekil 4. YSA temelli sistemin işlem basamakları

Sistem griseviye parmakizi resmini giriş olarak almakta ve çeşitli ön işlemler sonucunda temizlenip iyileştirilmiş resim elde etmektedir. Sonra inceltme işlemini gerçekleştiren sistem, inceltmiş resmi özellik noktalarının bulunması için eğitilmiş YSA'ya giriş olarak vermektedir. YSA çıkışında ise özellik noktaları bulunup işaretlenmiş parmakizi resmi elde edilmektedir. Resmin bilgi taşımayan gereksiz kısımlarından ve gürültüden arındırılması, OPTS için işlem hızının artması, işlem zamanının kısılması, özellik noktalarının bulunmasında sistem performansının ve güvenilirliğinin artması gibi bir takım üstünlükler sağlar. Bu nedenle bir OPTS'de, parmakizi resminin temizlenmesi ve iyileştirilmesi çok önemlidir. Resim ne kadar yüksek kalitede iyileştirilirse OPTS'nin performansı da o kadar artar (Hong et al., 1998; Halici et al., 1999; Greenberg et al., 2000; Saatci ve Tavşanoğlu, 2002; Jin et al., 2002; Özkaya, 2003).

Bu çalışmada resim iyileştirme ve temizleme, resim üzerinde çeşitli etkileri olan farklı maskelerin konvolüsyonundan oluşan bir maske kullanılarak gerçekleştirilmiştir. Bu maske, resim üzerinde yüksek frekanslı bileşenlerin sönümlenmesi etkisi gösteren Ortalama değer maskesi ile kenar bileşenlerin keskinleştirilmesi ve iyileştirilmesine yönelik işlem yapan Laplacian filtrenin konvolüsyonundan oluşur. Uzaysal formda gerçekleştirilen iki boyutlu konvolüsyon işlemi, a ve b konvolüsyon işlemi yapılacak filtreleri temsil etmek üzere Eşitlik 4'te verilen formül ile tanımlanabilir (Gonzalez and Woods, 1992).

$$c(n_1, n_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} a(k_1, k_2) b(n_1 + k_1, n_2 + k_2) \quad (4)$$

Konvolüsyon sonucunda Ortalama değer maskesinin sönümlenme etkisi ile Laplacian filtrenin kenar bilgilerini vurgulama etkisi amaca uygun olarak birleştirilmiştir. Parmakizi resminin temizlenmesi ve iyileştirilmesi için oluşturulan maske Şekil 5'te verilmektedir.

$$\begin{bmatrix} -1 & -4 & -10 & -16 & -19 & -16 & -10 & -4 & -1 \\ -4 & -7 & -13 & -10 & -13 & -10 & -13 & -7 & -4 \\ -10 & -13 & -19 & 2 & -1 & 2 & -19 & -13 & -10 \\ -16 & -10 & 2 & 68 & 74 & 68 & 2 & -10 & -16 \\ -19 & -13 & -1 & 74 & 80 & 74 & -1 & -13 & -19 \\ -16 & -10 & 2 & 68 & 74 & 68 & 2 & -10 & -16 \\ -10 & -13 & -19 & 2 & -1 & 2 & -19 & -13 & -10 \\ -4 & -7 & -13 & -10 & -13 & -10 & -13 & -7 & -4 \\ -1 & -4 & -10 & -16 & -19 & -16 & -10 & -4 & -1 \end{bmatrix}$$

Şekil 5. Parmakizi temizlemede kullanılan maske

Merkez piksel değerinin komşu piksellerin gri seviye ortalaması ile yer değiştirilmesiyle gerçekleştirilen Ortalama değer Eşitlik 5'te, resmin tüm noktalarındaki Laplacian değerinin karesinin toplanması ile ölçülen, hesaplama yükü ve zamanı açısından iyi performans sergileyen ve literatürde çok kullanılan Laplacian filtre ise Eşitlik 6'da verilen formüller ile tanımlanmaktadır (Gonzalez and Woods, 1992).

$$\hat{f}(x, y) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \quad (5)$$

$$\Delta^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (6)$$

Resim üzerine maske uygulandıktan sonra elde edilen temizlenmiş resme, çalışma prensibi Eşitlik 7'de verilen bölgesel ikili dönüşüm algoritması uygulanmıştır.

$$I_{yeni}(n_1, n_2) = \begin{cases} 1 & I_{eski}(n_1, n_2) \geq \text{Ortalama} \\ 0 & \text{Farklı durumlar} \end{cases} \quad (7)$$

Verilen Eşitlikte n_1 ve n_2 ilgili pikselin satır ve sütun numarasını; $I(n_1, n_2)$, n_1 ve n_2 koordinat değerlerine sahip pikselin renk değerini, *Ortalama* ise aynı pikselin içinde bulunduğu bölgedeki elemanların ortalamasını temsil etmektedir.

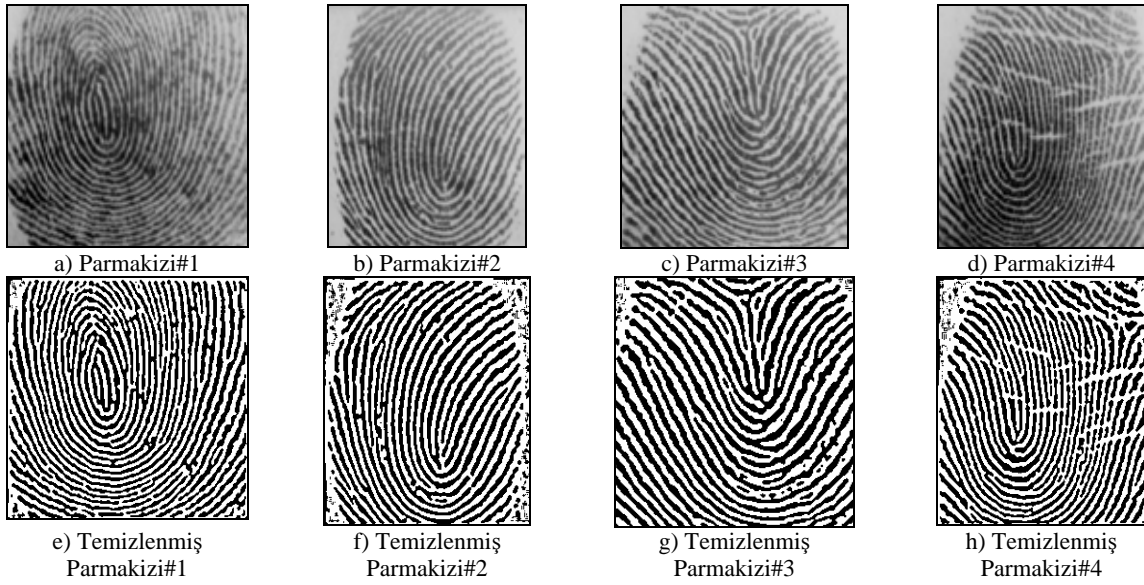
Geliştirilen algoritmanın verilen griseviye parmakizi resimleri için parmakizi temizleme ve iyileştirme adımı sonrasında elde ettiği sonuçlar Şekil 6'da verilmektedir.

Sunulan çalışmada OPTS'nin gerçekleştirdiği bir sonraki işlem, temizlenip iyileştirilen parmakizi resimlerinin inceltmesi ve inceltilen resmin iyileştirilmesidir. İnceltme, parmakizi resmi üzerinde bulunan her bir hat çizgisi genişliğinin daraltılarak 1 piksel ile ifade edilmesi şeklinde tanımlanabilir (Alkaya, 1998; Halici et al., 1999; Greenberg et al., 2000; Jin et al., 2002; Özkaya, 2003).

Bu çalışmada parmakizi resminin inceltmesi için iyi sonuç veren adaptif bir inceltme algoritması kullanılmıştır (Özkaya, 2003). İnceltme işlemini gerçekleştiren algoritmada 3 x 3 boyutunda bir maske kullanılmakta ve maskenin merkezinde bulunan nokta üzerinde işlem yapılmaktadır. Komşuluk mekanizmasına dayalı yöntemde merkezde bulunan siyah piksel etrafındaki siyah noktaların sayısı ve yine bu nokta etrafında siyahtan

beyaza geçiş ve beyazdan siyaha geçiş sayısı göz önünde bulundurularak, inceltme için belirlenen kurallar çerçevesinde ilgili noktalar silinmektedir. Silme işlemi resmin sol üst köşesinden sağ alt köşesine doğru yapılmakta ve silinecek noktalar aynı resim üzerinden silinmektedir. Böylece komşu siyah nokta sayısı hiçbir zaman 8 olmayacak ve inceltilen resim üzerinde kapalı bir çizgi oluşmayacaktır. İnceltilen parmakizi resminden özellik noktalarının doğru bir şekilde elde edilebilmesi için ilgili resimde yalancı özellik noktaları oluşturabilecek kısımların düzeltilmesi gerekmektedir (Xiao and Rafaat, 1991;

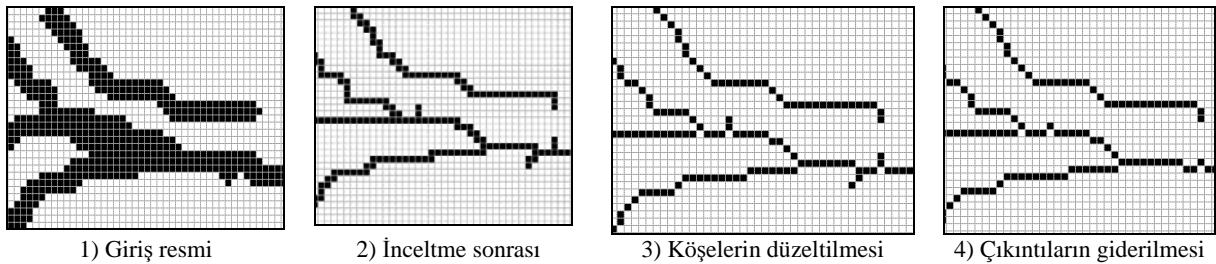
Espinosa-Duro, 2002; Özkaya, 2003). Bu yüzden inceltilen resim tekrar gözden geçirilerek onarılır. Bu onarılma işlemi, inceltilmiş resim üzerinde bulunan hat çizgilerinin düzgünleştirilmesi, hat çizgileri kenarlarında oluşan ufak çıkıntı şeklindeki yapıların ve özellik noktalarının bulunmasını zorlaştıran ayrıntıların giderilmesi, yönleri aynı olan birbirine yakın kırık çizgilerin birleştirilmesi, köşelerin tanımlanıp düzeltilmesi ve özellik noktalarının standartlaştırılması şeklinde sıralanabilir.



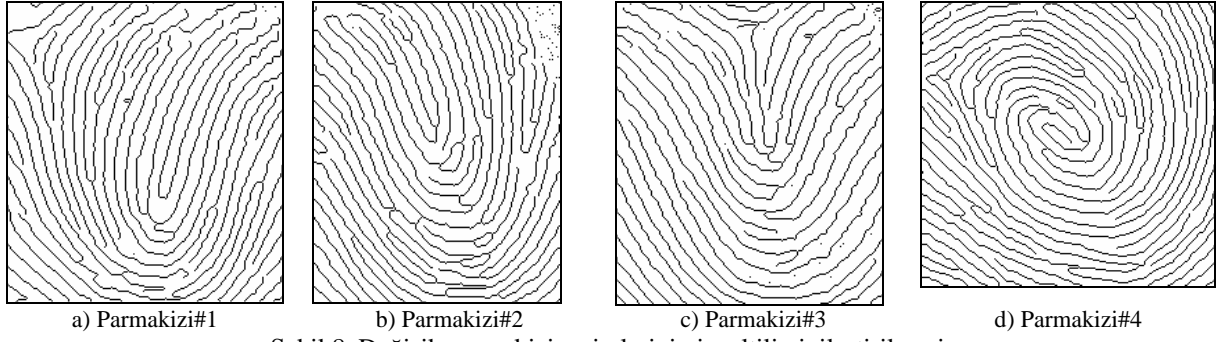
Şekil 6. Değişik parmakizi resimlerinin sunulan yöntemle temizlenmesi ve iyileştirilmesi.

Sistem performansını ve güvenilirliğini arttıran bu özellikler inceltme algoritmasını diğer algoritmalarından ayıran en temel farklardandır. Parmakizi resminin inceltmesiyle ve onarılmasıyla ilgili algoritmanın işlem adımlarının sonuçları sırasıyla Şekil 7'de, verilen inceltme algoritması çeşitli parmakizi örnekleri ile çalıştırıldığında inceltme işlemi sonrasında elde edilen sonuçlar ise Şekil 8'de verilmektedir.

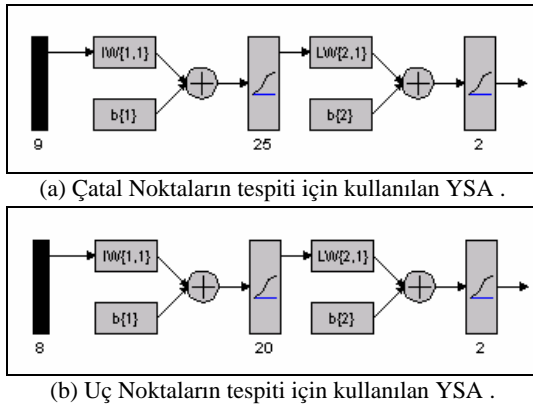
Temizlenip inceltilen parmakizi resmi özellik noktalarının bulunması için hazır hale gelmiştir.



Şekil 7. Geliştirilen inceltme algoritmasının adım adım sonuçları.



Şekil 8. Değişik parmakizi resimlerinin inceltilip iyileştirilmesi.



Şekil 9. Sistemde kullanılan YSA yapıları.

Verilen şekillerde $IW\{\}$ ve $LW\{\}$ YSA'nın ağırlıklarını, $b\{\}$ bias değerlerini, $+$ işareti içeren şekiller toplama fonksiyonlarını, dikey dikdörtgenler ise transfer fonksiyonlarını ifade etmektedir. YSA'nın özellik noktalarını tespit edebilmesi için giriş ve çıkış veri setinin oluşturulması ve sistemin uygun parametrelerle eğitilmesi gerekmektedir. Çatal noktaların tespitine yönelik YSA yapısında giriş işlem elemanı sayısı 9, çıkış işlem elemanı sayısı 2 ve ara katmandaki işlem elemanı sayısı 25 olarak belirlenmiştir.

YSA bileşeninin eğitimi için, parmakizi resminde bulunan çatal ve çatal olmayan noktalar 3×3 boyutunda bir pencere içinde ifade edilir ve vektör formuna çevrilir. Bu sayede YSA giriş veri setleri oluşturulmuş olur. Çıkış için ise "çatal noktadır" veya "çatal nokta değildir" şeklinde iki çıkış değeri verilir. Bu değerler YSA çıkışı olabilecek formata dönüştürüldüğünde $[0 \ 1 ; 1 \ 0]$ şeklinde çıkış veri seti de hazır hale gelmiş olur. Çatal noktaların tespitine yönelik YSA yapısının eğitiminde kullanılan giriş veri seti ve bu girişe uygun çıkış veri seti Tablo 1'de verilmektedir.

Uç noktaların tespitine yönelik YSA bileşeni Sagar and Beng (1999a,b) tarafından önerilen metod kullanılarak gerçekleştirilmiştir. Bu YSA yapısında giriş işlem elemanı sayısı 8 ve çıkış işlem elemanı

sayısı 2, ara katmandaki işlem elemanı sayısı ise 20'dir. Giriş veri seti tanımlanırken öncelikle 5×5 piksel boyutunda bir test penceresi belirlenir. Daha sonra 1×5 piksellik kenar bölgelerden 4 piksel uzağa kadar olan noktalar kontrol edilerek uç nokta olabilecek noktaların temsili gerçekleştirilmiş olur. Çıkış için ise "uç noktadır" veya "uç nokta değildir" şeklinde iki çıkış değeri verilir. Bu değerler YSA çıkışı olabilecek formata dönüştürüldüğünde $[0 \ 1 ; 1 \ 0]$ şeklinde çıkış veri seti de hazır hale gelmiş olur. Uç noktaların bulunması için gerçekleştirilen YSA yapısının eğitiminde kullanılan giriş veri seti ve bu girişe uygun çıkış veri seti Tablo 2'de verilmektedir.

Tablo 1. YSA Giriş ve Çıkış Veri Setleri-1.

Girişler									Çıkışlar	
0	1	0	1	0	1	1	1	0	1	0
1	1	0	1	0	1	0	1	0	1	0
0	1	1	1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1	1	1	0
1	0	1	1	0	1	0	1	0	1	0
1	1	0	0	0	1	1	1	0	1	0
0	1	0	1	0	1	1	0	1	1	0
0	1	1	1	0	0	0	1	1	1	0
1	1	1	1	1	1	1	1	1	0	1
0	1	1	1	0	1	1	1	1	0	1
1	1	0	1	1	1	0	1	1	0	1
1	1	1	1	1	1	1	0	0	0	1
1	1	0	1	1	0	1	1	0	0	1
0	1	1	1	0	1	1	1	0	0	1
1	1	0	1	0	1	0	1	1	0	1
0	1	1	0	1	1	0	1	1	0	1
1	0	1	1	0	1	1	0	1	0	1
1	1	1	1	1	0	1	0	1	0	1
0	1	0	0	1	0	0	1	0	0	1

Giriş-çıkış veri setlerinin tanımlanıp YSA'ya uygulanabilecek formata getirilmesiyle tasarımı tamamlanan sistem, eğitim parametrelerinin de belirlenmesi ile eğitime hazır hale gelmektedir. YSA yapılarının eğitiminde kullanılan parametreler Tablo 3'te verilmektedir. 5000 epok sonunda YSA yapılarının eğitim işlemleri tamamlanmış ve yapılar test edilmeye hazır hale gelmiştir. Sistemin

parmakizi özellik noktalarının bulunmasına yönelik performansını test etmek için kullanılan test resmine her iki yapının ayrı ayrı ürettiği sonuçlar Şekil 10'da, geliştirilen yeni yaklaşım klasik bir yöntemle (Özkaya, 2003) karşılaştırılması neticesinde elde edilen sonuçlar Şekil 11'de sunulmuştur. YSA ile gerçekleştirilen sistemlerin

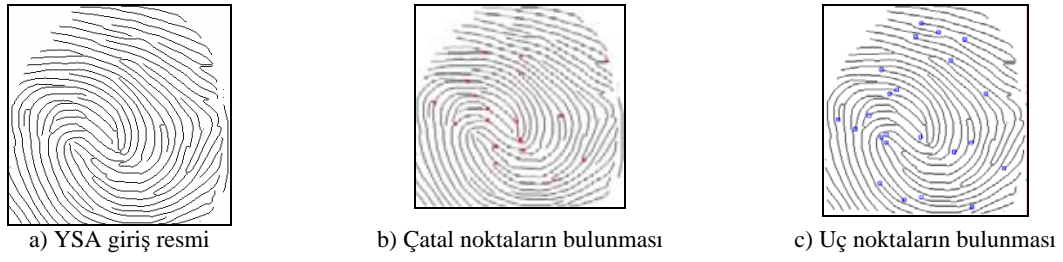
klasik yaklaşımlarla kıyaslandığında aynı sonucu elde edebildiği, buna karşın klasik yaklaşımlardaki algoritma geliştirme aşamasında programcı tarafından gerçekleştirilen zor ve karmaşık işlemleri ortadan kaldıran bir yapıya sahip olduğu görülmektedir.

Tablo 2. YSA Giriş ve Çıkış Veri Setleri-2.

Girişler								Çıkışlar	
0	1	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	0
1	1	0	1	1	1	1	1	1	0
1	1	1	0	1	1	1	1	1	0
1	1	1	1	0	1	1	1	1	0
1	1	1	1	1	0	1	1	1	0
1	1	1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	0	1	0
1	1	1	1	1	1	1	1	0	1
0	0	0	0	0	0	0	0	0	1
0	0	1	1	1	1	1	1	1	0
0	1	0	1	1	1	1	1	1	0
0	1	1	0	1	1	1	1	1	0
0	1	1	1	0	1	1	1	1	0
0	1	1	1	1	0	1	1	1	0
0	1	1	1	1	1	0	1	1	0
0	1	1	1	1	1	1	0	0	1
0	1	1	1	1	1	1	0	0	1
1	0	0	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	0
1	0	1	1	0	1	1	1	1	0
1	0	1	1	1	0	1	1	1	0
1	0	1	1	1	1	0	1	1	0
1	0	1	1	1	1	1	0	0	1
1	0	1	1	1	1	1	1	0	1
1	1	0	0	1	1	1	1	1	0
1	1	0	1	0	1	1	1	1	0
1	1	0	1	1	0	1	1	1	0
1	1	0	1	1	1	0	1	1	0
1	1	0	1	1	1	1	0	0	1
1	1	0	1	1	1	1	0	0	1
1	1	0	1	1	1	1	0	0	1
1	1	0	1	1	1	1	0	0	1
1	1	0	1	1	0	0	1	1	0
1	1	1	1	0	1	0	1	0	1
1	1	1	1	1	0	1	0	0	1
1	1	1	1	1	1	0	0	1	0
1	1	1	1	1	1	0	1	0	1
1	1	1	1	1	1	0	1	0	1
1	1	1	1	1	1	0	0	0	1
1	1	1	1	1	1	0	0	0	1
1	1	1	1	1	1	0	0	0	1
1	1	1	1	1	1	0	0	0	1

Tablo 3. YSA Eğitim Parametreleri

Parametre	Değer
Epok sayısı	5000
Hedef hata	0
α	0.01
α_i	1.05
α_d	0.7
Maksimum başarısızlık	5
β	0.9
Minimum hata değişimi	1e-10



Şekil 10. YSA yapılarından elde edilen test sonuçları.

Griseviye Parmakizi Resimleri

Klasik Yöntemle Özellik Noktaları Bulunmuş Parmakizi Resimleri

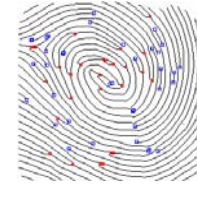
YSA ile Özellik Noktaları Bulunmuş Parmakizi Resimleri



a) Parmakizi #1



b) Parmakizi #1



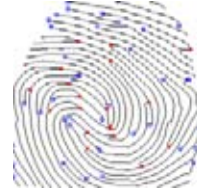
c) Parmakizi #1



d) Parmakizi #2



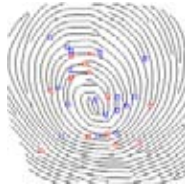
e) Parmakizi #2



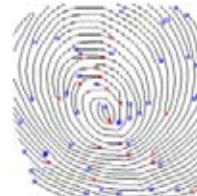
f) Parmakizi #2



g) Parmakizi #3



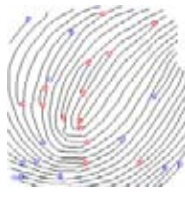
h) Parmakizi #3



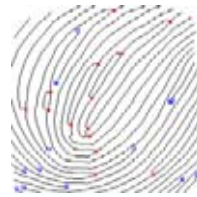
i) Parmakizi #3



j) Parmakizi #4



k) Parmakizi #4



l) Parmakizi #4

Şekil 11. YSA Temelli sistemin klasik yöntemle karşılaştırılması (Özkaya, 2003).

5. SONUÇ

Sunulan çalışmada, otomatik parmakizi tanıma sistemlerinin en önemli işlemlerinden biri olan parmakizi resimlerinde özellik noktalarının bulunmasına yönelik YSA temelli bir sistem gerçekleştirilmiştir. YSA temelli sistemin klasik

yaklaşımlardan en önemli farkı yüksek performansı ve herhangi bir türev hesaplama gereği duyulmamasından dolayı hesaplama karmaşıklığının düşük olmasıdır. Yapay sinir ağları kullanılarak yapılan çalışma sonucunda elde edilen sonuçlar incelendiğinde sunulan yeni yaklaşımın başarılı olduğu görülmektedir. Bu sayede, otomatik parmakizi tanıma sistemleri giriş çıkış kontrolünün

gerektiği her alanda, laboratuvarlar, bina ve şirket giriş çıkışları, güvenlik gerektiren kontrol noktaları, üniversite bünyesinde kampüslerde, fakülte, hastane, yemekhane, kütüphane gibi mekanların giriş-çıkış kapılarında kimlik tesbiti ve benzer güvenlik fonksiyonlarında, bilgisayar ve çok kullanıcıli bilgisayar ağına sahip ve sistemin ortaklaşa kullanıldığı tüm mekanlarda erişim düzeninin sağlanmasında ve kaynakların paylaşımında, ağ güvenliğinin sağlanmasında, fiziksel giriş çıkış kontrol noktalarında, personel devam kontrol sistemlerinde, ticari işlemler ve benzer bir çok alanda, banka güvenliğinde, elektronik fon transferi veya ATM güvenliği gibi işlemlerde, çek ve kredi kartı işlemlerinde, ulusal kimliklendirme sistemlerinde daha güvenilir ve düzenli kimliklendirme için uygulanabilecek ve geniş yelpazede çözümler sunabilecektir.

6. KAYNAKLAR

Alkaya, E. 1998. Enhancement and Preprocessing Techniques For Ridge Extraction in Fingerprint Images, Yüksek Lisans Tezi, Orta Doğu Teknik Üniversitesi, Ankara.

Anonymous, 2006. Matlab 7.0.1. Neural Network Toolbox User's Guide, 2006, <http://www.mathworks.com/access/helpdesk/help/toolbox/nnet/>.

Bhanu B., Boshra, M., Xuejun, T. 2000. Logical Templates For Feature Extraction in Fingerprint Images, 15th International Conference on Pattern Recognition, (2) 846 -850.

Espinosa-Duro, V. 2002. Minutiae Detection Algorithm For Fingerprint Recognition, IEEE Aerospace & Electronics Systems Magazine, (17) 7-10.

Gonzalez, R.C., Woods, R. E. 1992. "Digital Image Processing", Third Edition, Addison-Wesley, Reading, MA.

Greenberg, S., Aladjem, M., Kogan, D., Dimitrov, I. 2000. Fingerprint Image Enhancement Using Filtering Techniques, 15th International Conference on Pattern Recognition, (3) 322 -325.

Halici U., Jain, L. C., Erol, A. 1999. "Introduction to Fingerprint Recognition" Chapter in Jain L. C.; Halici U.; Hayashi, I.; Lee, S.B.; Tsutsui T., 1999, Intelligent Biometric Techniques in Fingerprint and Face Recognition, CRC press, USA.

Hong, L., Wan, Y., Jain, A.K. 1998. Fingerprint Image Enhancement: Algorithms and Performance Evaluation, IEEE Transactions on PAMI , 20, 8, 777-789.

Jain, A. K., Hong, L., Bolle, R. 1997a. On-line Fingerprint Verification, IEEE Transactions on PAMI, 19 (4), 302-314.

Jain, A. K., Hong, L., Pankanti, S., Bolle, R. 1997b. An Identity Authentication System Using Fingerprints, Proceedings of the IEEE, 85, 9, 1365-1388.

Jain, A. K., Prabhakar, S., Hong, L., Pankanti, S. 1999. FingerCode: A Filterbank For Fingerprint Representation and Matching, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (2)193-199.

Jin, A. L. H., Chekima, A., Dargham, J. A., Liao C. F. 2002. Fingerprint Identification and Recognition Using Backpropagation Neural Network, Student Conference on Research and Development SCORED, 98-101.

Liu, J., Huang, Z., Chan, K. L. 2000. Direct Minutiae Extraction From Grey-level Fingerprint Image by Relationship Examination, International Conference on Image Processing, 2, 427-430.

Luo, X., Tian, J., Wu, Y. 2000. A Minutiae Matching Algorithm in Fingerprint Verification, 15th International Conference on Pattern Recognition, 4, 833 -836.

Maio, D., Maltoni, D. 1998. Neural Network Based Minutiae Filtering in Fingerprints Fourteenth International Conference on Pattern Recognition, 2, 1654 -1658.

Ongun, G. 1995. An Automatic Fingerprint Identification System Based On Self organizing Feature Maps Classifier, Yüksek Lisans Tezi, Orta Doğu Teknik Üniversitesi, Ankara, Türkiye.

Özkaya, N. 2003. Otomatik Parmakizi Tanıma Sistemi, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, Erciyes Üniversitesi, Kayseri, Türkiye.

Rusyn, B., Prudyus, I., Ostap, V. 2001. Fingerprint Image Enhancement Algorithm, The Experience of Designing and Application of CAD Systems in Microelectronics CADSM, 193-194.

Saatci, E., Tavsanoğlu, V. 2002. Fingerprint Image Enhancement Using CNN Gabor-type filters, 7th

IEEE International Workshop on Cellular Neural Networks and Their Applications (CNNA), 377-382.

Sagar, V. K., Beng, K. J. A. 1999a. Hybrid Fuzzy Logic and Neural Network Model for Fingerprint Minutiae Extraction, International Joint Conference on Neural Networks IJCNN '99, (5) 3255 -3259.

Sagar, V. K., Beng, K. J. A. 1999b. Fingerprint Feature Extraction by Fuzzy Logic and Neural

Networks, 6th International Conference on Neural Information Processing ICONIP'99, (3) 1138 -1142.

Sađırođlu, Ş., Beşdok, E., Erler, M. 2003. Mühendislikte Yapay Zeka Uygulamaları I: Yapay Sinir Ağları, Ufuk Kitabevi, Türkiye.

Xiao, Q., Raafat, H. 1991. Fingerprint Image Post-Processing: A Combined Statistical and Structural Approach, PR(24), (10) 985-992.