



# **İZ DESENİ TEMSİLİ İLE PARMAK İZİ İNDEKSLEME YÖNTEMİ**

**Murat UYSAL**

**Karabük Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalında  
Doktora Tezi  
Olarak Hazırlanmıştır**

**KARABÜK  
Haziran 2015**

Murat UYSAL tarafından hazırlanan “İZ DESENİ TEMSİLİ İLE PARMAK İZİ İNDEKSLEME YÖNTEMİ” başlıklı bu tezin Doktora Tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Salih GÖRGÜNOĞLU

Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir. 16/ 06/ 2015

Ünvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Prof. Dr. Mehmet AKBABA (KBÜ)

Üye : Prof. Dr. Fatih Vehbi ÇELEBİ (YBÜ)

Üye : Doç. Dr. Halim Haldun GÖKTAŞ (YBÜ)

Üye : Yrd. Doç. Dr. İlhami Muharrem ORAK (KBÜ)

Üye : Yrd. Doç. Dr. Salih GÖRGÜNOĞLU (KBÜ)

...../...../2015

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Doktora derecesini onamıştır.

Prof. Dr. Nevin AYTEMİZ

Fen Bilimleri Enstitüsü Müdürü

*“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”*

Murat UYSAL

## **ÖZET**

**Doktora Tezi**

### **İZ DESENİ TEMSİLİ İLE PARMAC İZİ İNDEKSLEME YÖNTEMİ**

**Murat UYSAL**

**Karabük Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Yrd. Doç. Dr. Salih GÖRGÜNOĞLU**

**Haziran 2015, 74 sayfa**

Bu çalışmada özellik noktaları üçlüsü temelli parmak izi indeksleme sistemlerinin performansını arttırmak için kullanılabilir iz deseni temsili önerilmiştir. Önerilen temsil yöntemi, özellik noktaları üçlülerinin oluşturduğu üçgenel bölgelerdeki parmak izi desenini ifade etmektedir. İz deseni temsili parmak izindeki elastik distorsiyon, döndürme veya yer değiştirmeden etkilenmemekte ve sabit uzunlukta sayısal bir değerle ifade edilmektedir. Fakat özellik noktaları tespit hatalarına karşı duyarlıdır. İz deseni temsili kullanmak maksadıyla geliştirilen özellik noktaları üçlüsü temelli indeksleme uygulaması ile yapılan deneyler özellik noktaları üçlüsü temelli indekslemenin iz deseni kullanılmadan da parmak izlerinin sadece özellik noktalarıyla temsil edildiği sistemlerde çalışabilecek diğer yöntemlerden daha başarılı olduğunu ortaya koymuştur. Özellik noktalarının türünün tespitindeki hatalar iz desenini ve iz deseninin temsiliyi değiştirirse de özellik noktaları üçlüsü temelli bir indeksleme sisteminde iz deseni temsilleri aday parmak izlerinin puanlanmasında kullanılmış, indeksleme sisteminin daha iyi performans göstermesi sağlanmıştır. Özellik noktaları üçlüsü temelli indeksleme sistemlerinde kullanılabilir olan

zellik noktası tr ve zellik noktaları iftleri arasındaki iz izgileri sayısının da aynı puanlama yntemiyle kullanıldıklarında faydalı olduęu gsterilmiř fakat iz deseni temsili kullanıldığında en iyi sonular alınmıřtır.

**Anahtar Szckler** : İz deseni temsili, parmak izi indeksleme, parmak izi tanıma.

**Bilim Kodu** : 902.1.014

## **ABSTRACT**

**Ph. D. Thesis**

### **A FINGERPRINT INDEXING METHOD USING RIDGE PATTERN REPRESENTATION**

**Murat UYSAL**

**Karabük University**

**Graduate School of Natural and Applied Sciences**

**Department of Computer Engineering**

**Thesis Advisor:**

**Asst. Prof. Dr. Salih GÖRGÜNOĞLU**

**June 2015, 74 pages**

Ridge pattern representation which can be used to raise performance of a minutia triplet based fingerprint indexing system has been proposed. The proposed representation method represents fingerprint ridge pattern within a triangular area formed by a minutia triplet. It is robust to elastic distortion, rotation and translation invariant, and has a constant length. Nevertheless, it is sensitive to minutia detection errors. The experiments performed with the minutia triplet based indexing application, which was developed to use ridge pattern representation, have revealed that minutia triplet based indexing is successful even if it is used without ridge pattern representation compared to the systems in which fingerprints are represented with minutiae templates. Although minutia type detection errors change ridge pattern and ridge pattern representation, ridge pattern representations were used for candidate fingerprint impression scores, then the indexing system performed better. It has shown that minutia type and ridge counts between minutia pairs, too, in minutia triplet based systems are useful for indexing when they are used with the same score

calculation method but the best results has been obtained when we use ridge pattern representation.

**Key Words** : Ridge pattern representation, fingerprint indexing, fingerprint recognition.

**Science Code** : 902.1.014



## TEŐEKKÖR

Bu alıőmayı yapabilmem iin baőından beri tÖm imkanları bana sađlayan danıőman hocam Yrd. Do. Dr. Salih GÖRGÖNOĐLU'na, tez alıőması boyunca fikirlerinden istifade ettiđim tez komitesi üyeleri Prof. Dr. Mehmet AKBABA ve Prof. Dr. Fatih Vehbi ELEBi'ye, kıymetli yardımlarından dolayı iő arkadaşlarım Serkan ATAMAN ve Osman Nuri GÖKE'ye, yine alıőmayı yürütebilmem iin gerekli izin taleplerimi hi bir zaman geri evirmeyen kurum amirim Sayın Ali GÖCÖK'e teőekkürlerimi sunarım.

## İÇİNDEKİLER

	<b>Sayfa</b>
KABUL.....	iii
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER.....	ix
ŞEKİLLER DİZİNİ.....	xi
ÇİZELGELER DİZİNİ.....	xv
SİMGELER VE KISALTMALAR DİZİNİ.....	xvi
BÖLÜM 1.....	1
GİRİŞ.....	1
BÖLÜM 2.....	3
LİTERATÜR TARAMASI.....	3
BÖLÜM 3.....	18
İZ DESENİ TEMSİLİ.....	18
3.1. İZ DESENİ TEMSİLİNİN ELDE EDİLİŞİ.....	18
3.2. İZ DESENİ TEMSİLİNİN ÖZELLİKLERİ.....	20
3.3. İZ DESENİ TEMSİLİNİN ÖZELLİK NOKTASI TESPİT HATALARINA KARŞI DUYARLILIĞI.....	23
BÖLÜM 4.....	27
ÖNERİLEN İNDEKSLEME YÖNTEMLERİ.....	27
4.1. ÖZELLİK NOKTALARI ÜÇLÜSÜ TEMELLİ İNDEKSLEME METODU.....	28
4.1.1. ÖZELLİK NOKTALARI ÜÇLÜSÜNÜN ÖZELLİKLERİ.....	29
4.1.2. İNDEKSLERİN ÇIKARILMASI.....	29
4.1.3. YEDEK İNDEKSLER.....	30
4.1.4. İNDEKSLERİN DEPOLANMASI.....	32

	<b><u>Sayfa</u></b>
4.1.5. SORGU CEVABININ ÇIKARILMASI.....	33
4.1.6. METODUN UYGULANMASI.....	35
4.2. İZ DESENİ TEMSİLİ İLE ÖZELLİK NOKTALARI ÜÇLÜSÜ TEMELLİ İNDEKSLEME.....	40
BÖLÜM 5.....	44
DENEY SONUÇLAR.....	44
5.1. MCC TEMELLİ İVE UYGULANAN ÖZELLİK NOKTALARI ÜÇLÜSÜ TEMELLİ İNDEKSLEMELERİN KARŞILAŞTIRILMASI.....	44
5.2. İZ DESENİ TEMSİLİNİN ETKİSİ.....	50
5.3. İZ DESENİ TEMSİLİ İLE ÖZELLİK NOKTASI TÜRLERİNİN ETKİLERİNİN KARŞILAŞTIRILMASI.....	53
5.4. İZ DESENİ TEMSİLİ İLE İZ ÇİZGİLERİ SAYILARININ ETKİLERİNİN KARŞILAŞTIRILMASI.....	57
BÖLÜM 6.....	68
SONUÇ.....	68
KAYNAKLAR.....	70
ÖZGEÇMİŞ.....	74

## ŞEKİLLER DİZİNİ

	<b>Sayfa</b>
Şekil 1.1. Parmak izi indeksleme sistemi.....	1
Şekil 2.1. Parmak izi kodu (FingerCode).....	3
Şekil 2.2. Çekirdek, delta ve paralel yönsel görüntüler.....	4
Şekil 2.3. Özellik noktası ve etrafında oluşturulan yapılar.....	5
Şekil 2.4. Etiketlerinden faydalanılacak ilişkili yapılar.....	6
Şekil 2.5. Özellik noktaları üçlüsü.....	6
Şekil 2.6. İndeks, kimlik ve indeks puanlarının tutulduğu veri yapıları.....	7
Şekil 2.7. Özellik noktalarının delaunay üçgenleştirmesi.....	8
Şekil 2.8. Delaunay üçgenleştirmesi adımları.....	9
Şekil 2.9. Özellik noktaları üçlüsü ve uzunluklar.....	10
Şekil 2.10. Özellik noktalarının üzerinde bulunduğu kuadratik eğriler.....	10
Şekil 2.11. İz çizgilerinin çatallanma noktalarının değişik durumları.....	11
Şekil 2.12. Özellik noktalarının birinin hafif kaymasıyla delaunay üçgen yapısının değişmesi.....	12
Şekil 2.13. Özellik noktaları dörtlüsü.....	12
Şekil 2.14. Minutia – Cylinder- Code (MCC) temsili.....	13
Şekil 2.15. LSH uygulaması örneği.....	14
Şekil 2.16. Yüksek dereceli delaunay üçgenleri.....	16
Şekil 2.17. Delaunay ve genişletilmiş delaunay üçgenleri.....	16
Şekil 3.1. İz çizgilerinin iz deseni temsiline katılması.....	19
Şekil 3.2. İz deseninin elde edilmesi.....	19
Şekil 3.3. İz görüntüsü ve gerdirilmiş hali.....	20
Şekil 3.4. Delaunay üçgenleştirmesi ile temsil edilen iz deseni parçaları.....	21
Şekil 3.5. Birinci derece delaunay üçgenleştirmesi ile temsil edilen iz deseni parçaları.....	22
Şekil 3.6. Genişletilmiş delaunay üçgenleştirmesi ile temsil edilen iz deseni parçaları.....	22
Şekil 3.7. Bir özellik noktasının yanlış tespit edilmesi ile iz deseni temsiline değişmesi.....	24
Şekil 4.1. Özellik noktaları üçlüsü ve aralarındaki vektörler.....	29

	<u>Sayfa</u>
Şekil 4.2. Vektör uzunlukları için bin'ler ve yedek indekslerin üretildiği değerler.....	31
Şekil 4.3. Bağlı özellik noktası yönleri için bin'ler ve yedek indekslerin üretildiği değerler.....	32
Şekil 4.4. MINDTCT çalışma adımları.....	35
Şekil 4.5. Saptanan iki türden özellik noktasının konumları.....	36
Şekil 4.6. Düşük güvenilirlikli özellik noktaları filtrelenmediğinde üçgen ağı.....	37
Şekil 4.7. Düşük güvenilirlikli özellik noktaları filtrelendiğinde üçgen ağı.....	38
Şekil 4.8. Özellik noktasının türünün yanlış tespit edilmesiyle özellik noktaları arasında kalan iz çizgilerinin sayılarının değişmesi.....	40
Şekil 4.9. Kayıp alanlar barındıran üçgensel iz desenleri.....	42
Şekil 5.1. FVC2000DB2 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	46
Şekil 5.2. FVC2000DB3 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	46
Şekil 5.3. FVC2002DB1 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	47
Şekil 5.4. FVC2004DB1 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	47
Şekil 5.5. NISTDB4 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	48
Şekil 5.6. NISTDB14 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	48
Şekil 5.7. FVC2000DB2 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	50
Şekil 5.8. FVC2000DB3 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	51
Şekil 5.9. FVC2002DB1 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	51
Şekil 5.10. FVC2004DB1 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	52
Şekil 5.11. NISTDB4 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.....	52

Şekil 5.12. NISTDB14 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	53
Şekil 5.13. FVC2000DB2 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	54
Şekil 5.14. FVC2000DB3 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	54
Şekil 5.15. FVC2002DB1 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	55
Şekil 5.16. FVC2004DB1 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	55
Şekil 5.17. NISTDB4 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	56
Şekil 5.18. NISTDB14 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	56
Şekil 5.19. FVC2000DB2 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	57
Şekil 5.20. FVC2000DB3 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	58
Şekil 5.21. FVC2002DB1 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	58
Şekil 5.22. FVC2004DB1 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	59
Şekil 5.23. NISTDB4 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	59
Şekil 5.24. NISTDB14 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.....	60
Şekil 5.25. FVC2000DB2 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indekisleme performansları.....	61

## Sayfa

Şekil 5.26. FVC2000DB3 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.....	61
Şekil 5.27. FVC2002DB1 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.....	62
Şekil 5.28. FVC2004DB1 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.....	62
Şekil 5.29. NISTDB4 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.....	63
Şekil 5.30. NISTDB14 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.....	63
Şekil 5.31. FVC2000DB2 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.....	64
Şekil 5.32. FVC2000DB3 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.....	65
Şekil 5.33. FVC2002DB1 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.....	65
Şekil 5.34. FVC2004DB1 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.....	66
Şekil 5.35. NISTDB4 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.....	66
Şekil 5.36. NISTDB14 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.....	67

## ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 3.1. Değişik parmak izi veritabanlarındaki aramalar için kayıp oranları.....	25
Çizelge 5.1. Farklı $c$ değerleri için tutma oranları.....	45
Çizelge 5.2. Parmak izi başına ortalama arama süresi.....	49



## **SİMGELER VE KISALTMALAR DİZİNİ**

### **SİMGELER**

c : Yedek indeks miktarı katsayısı

### **KISALTMALAR**

MCC: Minutia-Cylinder Code (Özellik Noktası-Silinder Kodu)

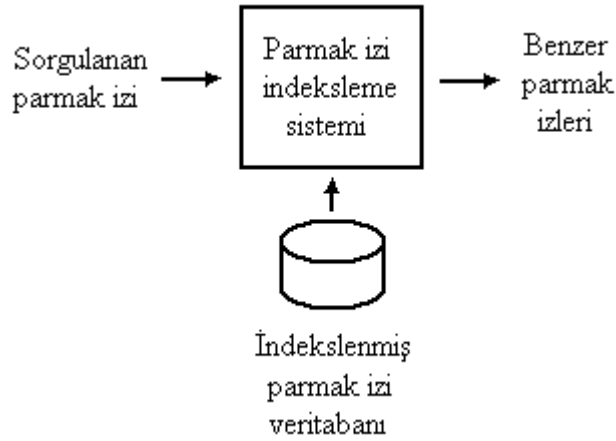
LSH: Locality Sensitive Hashing (Konum Duyarlı Seçme)

ÖNÜ: Özellik Noktaları Üçlüsü

## BÖLÜM 1

### GİRİŞ

Sorgulanan parmak izinin veritabanında hangi parmak iziyle eşleşebildiğinin her bir parmak iziyle karşılaştırma yapılarak bulunması, büyük parmak izi veritabanlarının kimliklendirme sistemleri için çok uzun zaman almaktadır. Bu problemi çözmek için bir yolu parmak izlerini sınıflandırarak arama uzayını daraltmaktır. Sınıflandırma temelli tekniklerin seçiciliği büyük ölçüde sınıfların sayısına ve parmak izlerinin bu sınıflardaki doğal dağılımına bağlıdır [1]. Parmak izlerinin biçimsel yapısına (sağa döngü, sola döngü, kemerli, halka vb.) bağlı olarak yapılan sınıflandırmada, sınıfların sayısı azdır, parmak izleri de bu sınıflara eşit olarak dağılmamıştır ve hiç bir sınıfa dahil edilemeyecek parmak izleri de olabilmektedir [2]. Arama uzayını daraltmanın diğer bir yolu ise Şekil 1.1'de gösterilmiş olan indeksleme yöntemini uygulamaktır.



Şekil 1.1. Parmak izi indeksleme sistemi.

Parmak izi tanıma sistemlerinin parçası olan parmak izi indeksleme sistemleri, kendisi tarafından indekslenmiş parmak izlerinden, sorgulanan parmak izlerine benzeyen parmak izlerini bezerlik sırasına göre sıralanmış halde sonuç kümesi olarak vermektedir. Böylece parmak izi tanıma sisteminde, indeksleme sisteminin çıktısı olan daha az sayıdaki parmak izleri birer birer sorgulanan parmak izi ile doğrulama

işlemine tabi tutulur. Bu şekilde parmak izi tanıma sistemlerinin işlem zamanı kısalmır. Parmak izi indekslemesinde parmak izinden çıkartılmış çeşitli özellikler kullanılmaktadır. Örneğin indeksleme için yönelim alanı kullanılmıştır [3-4]. Başka bir çalışmada ise gabor filtresi cevabından elde edilen parmak izi kodu (FingerCode) kullanılmıştır [5]. Bazı çalışmalarda değişik tipteki filtre cevapları kullanılmıştır [6-7]. Parmak izinden elde edilen özellik noktalarının (uç ve çatal) kullanıldığı çok sayıda çalışma mevcuttur [8-20]. [21-22] iz çizgilerinin yöneliminden başka iz frekanslarını da indeksleme amaçlı kullanmıştır.

Çoğu indeksleme algoritması, parmak izini temsil etmek ve indeksler oluşturmak için özellik noktalarını temel alarak kullanır [20]. Halbuki parmak izini temsil edebilecek yapı inceltirilmiş iz çizgileridir [23]. Fakat iz çizgilerini kullanan çalışmaların sayısı çok azdır.

Özellik noktaları etrafındaki iz çizgilerinin etiketlenerek indekslemede kullanıldığı bir çalışma yapılmıştır [9]. Özellik noktalarıyla beraber, üzerinde özellik noktasını tutan iz çizgilerinin ucuna yakın kısmının kuadratik eğri parametrelerini indeks elde etmede kullanan çalışma [14] vardır. Bu çalışmada iz çizgilerinin özellik noktalarına yakın yerlerinin şekli kullanılır ama iz çizgilerinin deseni kullanılmaz. İz çizgilerinin eğrilik değerinin indeksleme ve sınıflandırmada faydalı olacağını gösteren çalışma [24] da iz çizgilerinin birbirlerine göre durumunu indekslemede kullanmaz.

Bu çalışma kapsamında parmak izi çizgilerinin desen temsili için bir yöntem önerilmiş, önerilen temsil yöntemi uygulanan parmak izi indeksleme yöntemiyle beraber kullanılmıştır. İz deseni temsili kullananabilmek için uygulanan indeksleme yöntemi iz desenini kullanmadan da üstün başarı göstermiştir. İz deseni ile beraber kullanıldığında indeksleme başarısının daha da iyi olduğu görülmüştür.

Parmak izi indekslemesi ile ilgili literatürde geçen çalışmalar ayrıntılı olarak Bölüm 2 'de anlatılmıştır. Önerilen iz deseni temsili ise Bölüm 3' te anlatılmıştır. Bölüm 4' te gerçekleştirilen indeksleme yöntemi verilmiş, Bölüm 5' te ise yapılan deney sonuçları aktarılmıştır.

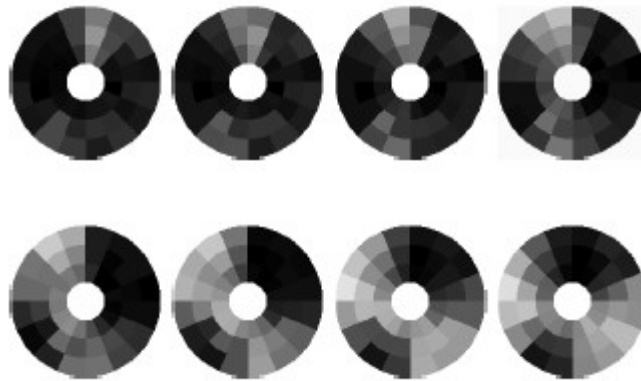
## BÖLÜM 2

### LİTERATÜR TARAMASI

Literatürde geçen parmak izi indeksleme çalışmalarının bir kısmı yönelim alanı temellidir. Bunun örneklerinden biri olarak yönelim alanı modellenmesi Fourier açılımı ile yapılmış ve Fourier açılımı katsayılarından elde edilen vektörler indekslemede kullanılmıştır [3].

Yönelim alanlarına karşılık gelecek özellik vektörlerini çıkarmak için Fourier açılımı yerine döndürmeye duyarsız olduğu bildirilen kutupsal kompleks moment'ler kullanılmıştır [4]. Bu durumda indekslemenin Fourier açılımıyla yapılan yönelim modellenmesinden daha başarılı olduğu bildirilmiştir.

İndeksleme için yönsel alan, parmak izi kodu (FingerCode) ve özellik noktaları üçlüleri birlikte kullanılmıştır [5]. Bu çalışmada yönsel alandan elde edilen özellik vektörünün boyutu PCA veya Karhunen – Loève dönüşümü ile düşürülmektedir. 30 boyutlu özellik vektörünün en iyi sonucu verdiği bildirilmiştir.



Şekil 2.1. Parmak izi kodu (FingerCode) [5].

Parmak izi kodu şu şekilde elde edilmektedir: Çekirdek-eşmerkezli 20 'şer piksel genişliğinde ve 16 dilime ayrılmış 3 diskin kapladığı alan 8 farklı yöne ayarlanmış

Gabor filtrelerinden geçirilir. Gabor filtresi cevabının her parça için standart sapması hesaplanarak Şekil 2.1 'deki gibi 384 elemanlı parmak izi kodu elde edilir. Özellik noktaları üçlüleri hash tablosu ile kullanılmaktadır.

Bu çalışmada üç farklı yöntem indeksleme için denemiştir. Yönsel alan kullanıldığında en iyi sonuçların elde edildiği, özellik noktaları üçlüleri kullanıldığında elde edilen sonuçların literatürde bildirilenlerden daha kötü olduğu belirtilmiştir. Bu üçünün kombinasyonu indeksleme için kullanıldığında en iyi sonucun elde edildiği, fakat özellik noktaları üçlülerinin bu sonuca kayda değer katkısının olmadığı ifade edilmiştir.

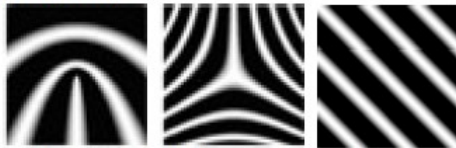
Daha farklı filtre cevaplarının da kullanıldığı çalışmalar mevcuttur. Örnek olarak iz görüntülerindeki tekil nokta korelasyonundan faydalanılarak indeksleme yapılmıştır. Korelasyon hesabında MACE (Minimum Average Correlation Energy) filtresi kullanılmıştır. Bu filtre tekil nokta etrafındaki yönsel görüntüden yola çıkarak hesap yapmaktadır [6].

Diğer bir çalışmada parmak izlerinde Şekil 2.2 'de gösterildiği gibi bulunabilecek çekirdek, delta ve paralel yönsel görüntü formunu saptamak üzere, izin yönsel görüntüsü sırasıyla (2.1a), (2.1b) ve (2.1c)' de eşitlikleri verilen filtrelerden geçirilmiştir [7]. Burada  $i$  sanal birimdir.  $x$  ve  $y$  pikselin bulunduğu konumu,  $\sigma$  standart sapmayı ifade etmektedir.

$$h_c = (x + iy) \exp(-(x^2 + y^2)) / (2\sigma^2) \quad (2.1a)$$

$$h_d = (x - iy) \exp(-(x^2 + y^2)) / (2\sigma^2) \quad (2.1b)$$

$$h_p = \exp(-(x^2 + y^2)) / (2\sigma^2) \quad (2.1c)$$

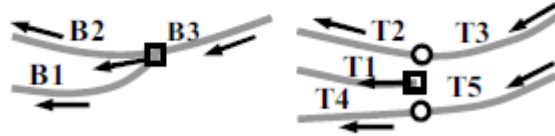


Şekil 2.2. Çekirdek, delta ve paralel yönsel görüntüler [7].

Filtrelerin cevabı farklı izlerde çekirdek ve delta noktaları için farklı olmaktadır. Parmak izi görüntüleri çekirdek noktaları referans alınarak hizalanır. Sonra elde edilen filtre cevaplarının blok ortalamaları alınarak boyutları düşürülür. Ortaya çıkan değerler indekslemede kullanılır.

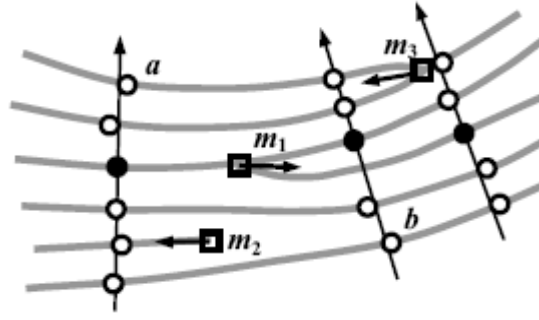
Bir başka çalışmada [8] indeks olarak özellik noktalarının çekirdekten uzaklığı ve üzerinde bulunduğu çekirdek-eşmerkezli  $5^{\circ}$  'lik dilim numarası birleşimi kullanılmıştır.

İz çizgilerini dikkate alarak yapılan çalışmada [9] özellik noktaları etrafında özellik noktası ve etrafındaki iz çizgilerinden müteşekkil birer yapı oluşturulmuştur. Bu yapı özellik noktasını, iz çizgisinin çatallanma noktası için 3 iz çizgisi ve iz çizgisi uç noktası için uç noktasının üzerinde olduğu iz çizgisi ve onun da etrafında olan iki iz çizgisini barındırır. Uç noktası etrafındaki iz çizgileri, iz çizgisinin uç noktası hizasından ikiye bölünür. Parçalara ayrılmış iz çizgileri Şekil 2.3 'te görüldüğü gibi göreceli konumlarına göre etiketlenir. İz çizgilerinin akış yönleri özellik noktasının yönüne göre belirlenir [9].



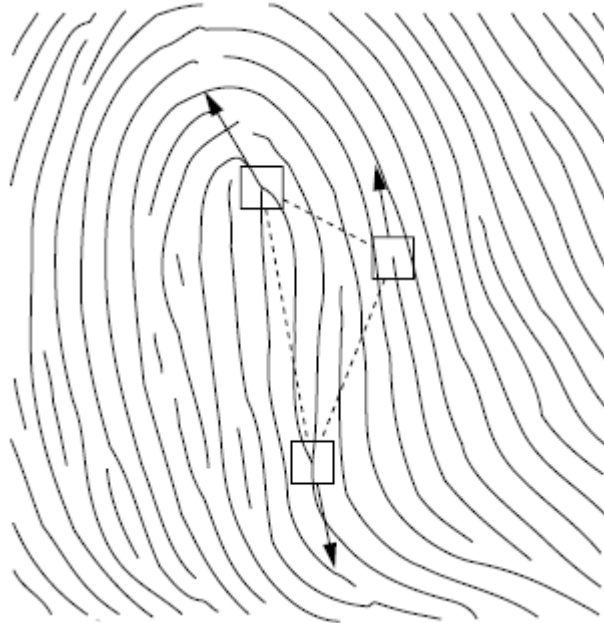
Şekil 2.3. Özellik noktası ve etrafında oluşturulan yapılar [9].

Birbiriyle Şekil 2.4'teki gibi ilişkisi bulunan yapıların ortaya çıkardığı etiketler, aralarda kalan iz çizgisi sayısı gibi değerlerle çok sayıda indeks oluşturulur. Sonuçlar [10] ve [11]'de verilen yöntemlerle karşılaştırıldığında daha ayırıcı ve doğrusal olmayan bozulmalara karşı dayanıklı bir indeksleme yapıldığı belirtilmiştir. İz çizgilerini ele alış tarzının [10]'da belirtildiği gibi görüntü kalitesindeki bozulmadan etkilenmeyeceği bildirilmiştir. Ayrıca sorgulama sırasındaki işlemlerin hızından bahsedilmiştir.



Şekil 2.4. Etiketlerinden faydalanılacak ilişkili yapılar. [9].

Parmak izi indeksleme özellik noktaları konum ve açı değerleriyle temsil edilmiştir [10]. İzdeki Şekil 2.5' teki gibi özellik noktaları üçlüleri indeks değeri hesaplanmasında kullanılmaktadır. Belli bir mesafeden daha fazla birbirine uzak olmayan nokta üçlüleri kullanılarak indeks sayısının özellik noktası sayısının üçlü kombinasyonlarının sayısına kadar ulaşmasının önüne geçilmiştir. (Fakat yine de arama işlemlerinin kompleksitesin üst sınırı  $O(N^3)$  olmaktadır.) Tam bir indeks 9 bileşenden oluşmaktadır. Bunlar: her bir kenarın uzunluğu, her bir nokta çifti arasındaki iz çizgileri sayısı ve referans olarak alınan bir kenara göre olan özellik noktaları açılarıdır.

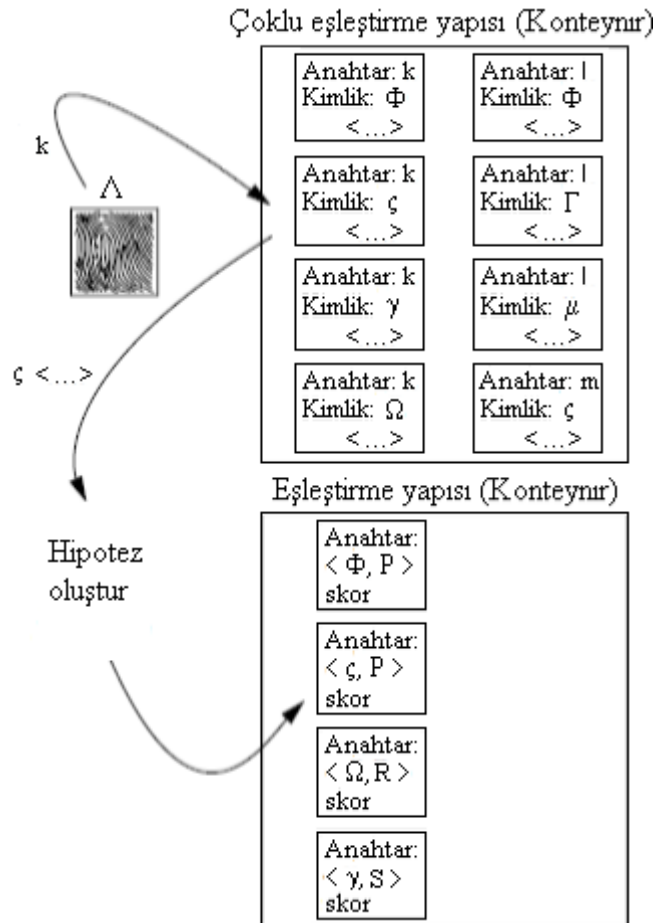


Şekil 2.5. Özellik noktaları üçlüsü [10].

Uygulanan yöntemin sonuçları izin yer değiştirmesinden veya döndürülmesinden

etkilenmemektedir. Noktaların oluşturduğu üçgenin en büyük kenarından başlanarak saat yönünde etrafında dolaşarak elde edilen sıraya göre indeks değerleri hesaplanır. Böylelikle aynı parmağın aynı nokta üçlüsü için farklı izlerde aynı sıra izlenerek aynı indeks değeri hesaplanır.

Saklanan izlere ait kimlik değerleri onlara karşılık gelen indekslerle çoklu eşleştirme veri yapısında tutulur. Sorgulanan iz için hesaplanan bir indekse karşılık, çoklu eşleştirme yapısı aday izleri belirten kimlikler kümesi verir. Anahtar olarak kimlik, veri olarak oy sayısını tutan bir eşleştirme yapısı kullanılarak sorgulanan izden çıkan her indeks için çoklu eşleştirmenin verdiği kimliklerin eşleştirme yapısındaki karşılığı olan girdideki oy değeri 1 artırılır.



Şekil 2.6. İndeks, kimlik ve indeks puanlarının tutulduğu veri yapıları [10].

Böylece sorgulanan iz için Şekil 2.6' daki eşleştirme yapısı (iz kimliği, indeks puanı) ikililerini tutar. Belirli eşik değerinin üstündeki indeks puanına sahip girdilerin anahtar

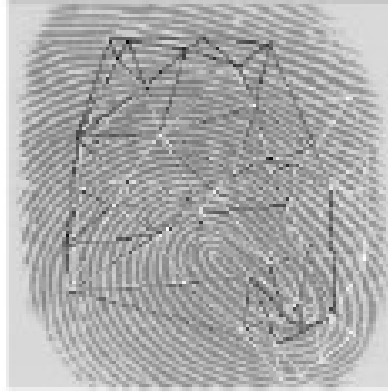


değerlerinden sorgulanan parmak izi ile karşılaştırılacak izlere ait kimlikler elde edilir.

Bir başka çalışmada da [11] özellik noktaları üçlülere indekslemede kullanılmıştır. Araştırmacılar [10] ile karşılaştırma yaparak indeksleme yöntemlerinin konum değişikliği ve döndürmeden etkilenmemesinin yanı sıra ölçekleme ve bir takım bozulmalardan da etkilenmediğini belirtmişlerdir. [10]'da yapılan çalışmayı ölçeklemeyi dikkate almaması ve özellik noktaları arasındaki iz çizgileri sayısının tespitinin iz görüntüsünün kalitesine bağlı olması sebebiyle zayıf bulmaktadırlar.

İndeksleme elemanları olarak şunlar kullanılmıştır: Üçgenin en küçük açısı, üçgenin ortanca açısı, üçgenin belli iki kenarından elde ettikleri vektörlerin çarpımı olan vektörün yönü (buna üçgenin yönelimi denmektedir), özellik noktalarının tiplerine karşılık gelen 0 ve 1 değerlerinin sıralı biçimde dizilmesinden ortaya çıkan ikilik sistemdeki sayı (buna üçgenin tipi denmektedir), üçgenin yönü, en uzun kenarın uzunluğu. Hatalara karşı daha sağlam indeksleme için bazı geometrik kısıtlamalara da gidilmiştir.

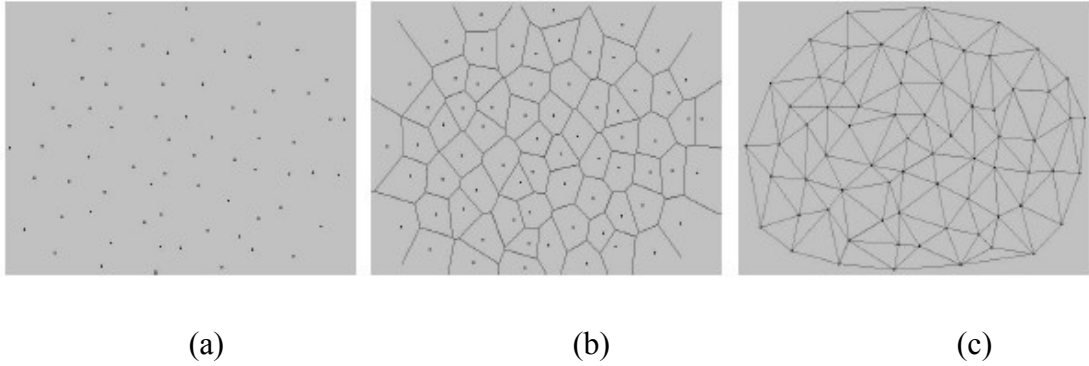
Bir diğer çalışmada [12] ise [11]'de yapılan çalışma ile karşılaştırma yapılmış, N özellik noktası sayısı iken  $O(N^3)$  mertebesinde indeks olacağına vurgu yapılmıştır. Örnek özellik noktaları kümesi için Delaunay üçgenleştirmesi Şekil 2.7 ile verilmiştir.



Şekil 2.7. Özellik noktalarının delaunay üçgenleştirmesi [12].

Özellik noktalarından Delaunay üçgenleştirmesi ile benzersiz bir topolojik yapı oluşturulur. Bu yapının üçgenleri indeksleme için kullanılır. Bunun avantajları şu şekilde sıralanmaktadır: N tane özellik noktasının Delaunay üçgenleştirmesinden  $O(N)$  mertebesinde üçgen ortaya çıkar. Bu, indekslemenin bellek gereksinimini oldukça düşürür. Ayrıca arama zamanını da kısaltır.

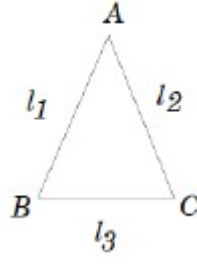
Özellik noktalarından elde edilen Delaunay üçgenleri ayırdedici bir yapıdır. Bu yapı parmak izindeki özellik noktalarının komşulukları ile oluşturdukları yapıyı yansıtarak ait olduğu parmak izine has bir yapı ve ona has indeks değerleri çıkarmaya müsaittir. Gürültü sebebiyle bir özellik noktasının yok olması veya yerinin değişmesi Delaunay üçgenleri yapısının tümünü değil sadece özellik noktasının üzerinde bulunduğu bölgeyi yerel olarak etkiler. Delaunay üçgenleştirmesi için burada kullanılan yöntem özellik noktalarından voronoi diagramını elde etmek ve Voronoi diagramını Delaunay üçgenlerine dönüştürmektir. (Voronoi diagramı ile Delaunay üçgenlerinin birinden diğeri elde edilebilir.)



Şekil 2.8. Delaunay üçgenleştirmesi adımları, a) Özellik noktaları, b) Voronoi diagramı, c) Delaunay üçgenleri [12].

Bu işlemin karmaşıklığı  $O(N \log(N))$  olmaktadır. Voronoi hücrelerinin merkezlerindeki noktaları komşu hücrelerin merkezlerindeki noktalarla birleştiren doğru parçaları Delaunay üçgenleştirmesini yapar.

İndeksler şöyle elde edilir: Şekil 2.9' da örneği verilen özellik noktaları üçlüsünün oluşturduğu üçgenin kenarları  $l_1 < l_2 \leq l_3$  olacak şekilde uzunluk sırasına dizilir. Sonra eşitlik (2.2a), eşitlik (2.2b) ve eşitlik (2.2c) ile verilen değerler hesaplanır.



Şekil 2.9. Özellik noktaları üçlüsü ve uzunluklar [12].

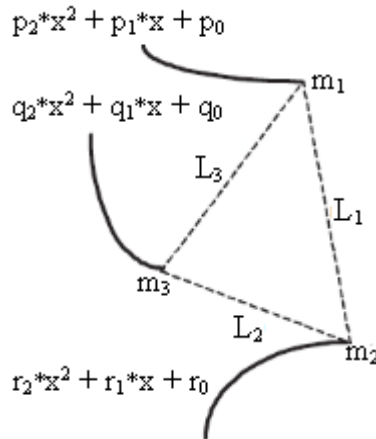
$$0 \leq \frac{l_1}{l_3} \leq 1 \quad (2.2a)$$

$$0 \leq \frac{l_2}{l_3} \leq 1 \quad (2.2b)$$

$$-1 \leq \cos(A) \leq 1 \quad (2.2c)$$

Burada A en kısa iki kenar arasındaki açıdır. Elde edilen değerler ölçeklendirilir. Sonra da kuantizasyona tabii tutulurlar. Böylece indeks elemanları hesaplanmış olur. Buradaki indeks elemanlarının gürültüye karşı sağlam kalabildiği belirtilmektedir. Parmak izi doğrulaması ile ilgili yapılan bir çalışmada [13], özellik noktalarından elde edilen Delaunay üçgenlerinin ait olduğu iz yapısını diğerlerinden ne kadar ayırt edici bir şekilde temsil edebildiği gösterilmiştir.

Bir başka çalışmada Şekil 2.10' da örneği verilen özellik noktalarının Delaunay üçgenleri ile üçgeni meydana getiren özellik noktalarının üzerinde bulunduğu iz çizgisinin kuadratik eğri parametreleri indekslemeye katılmıştır [14].

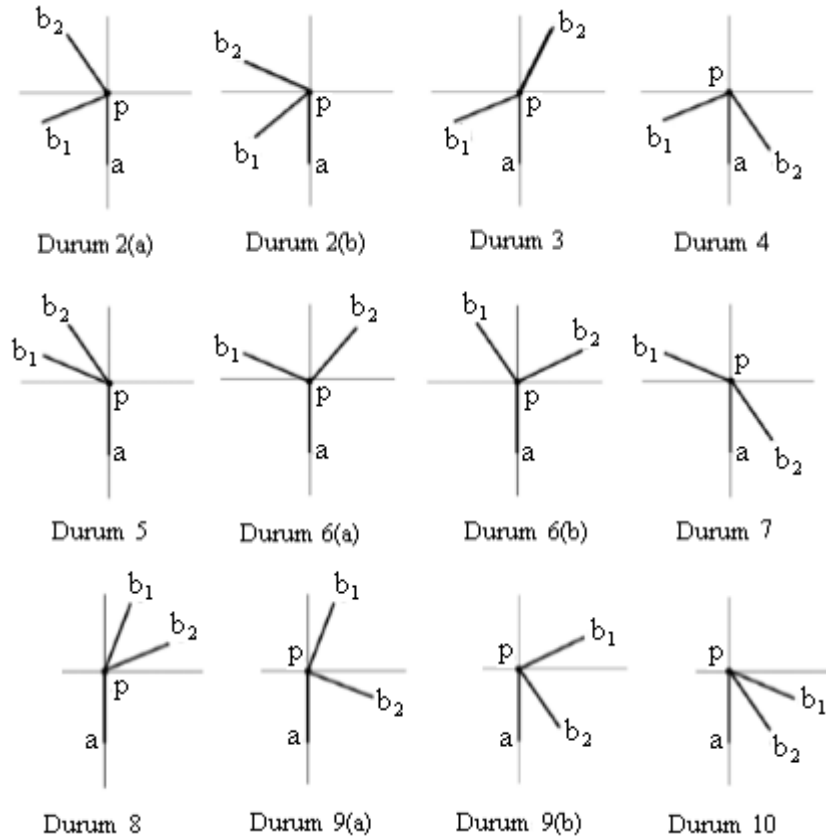


Şekil 2.10. Özellik noktalarının üzerinde bulunduğu kuadratik eğriler [14].

Delaunay üçgeni ve ilgili iz çizgileri parametreleri ile 9 boyutlu indeksler meydana getirilir. Veritabanındaki tüm indeksler  $k$  - means kümeleme algoritmasına tabi tutulur. Sorgulanan parmak izinin indeks değerlerinin ait olduğu kümelerdeki parmak izleri sonuç olarak elde edilir.

Diğer bir çalışmada indeks elemanları olarak Delaunay üçgenleştirilmesi yapılan özellik noktaları üçlüleri kullanılmıştır. Her özellik noktası etrafındaki en yakın  $k$  komşuluğundaki gözenek noktalarının özellik noktasından en çok ve en az uzaklık değerleri ve yine en yakın  $k$  komşuluğundaki gözenek noktalarının özellik noktalarından mesafelerinin ortalaması indeks elemanı olarak kullanılmıştır [15].

Özellik noktalarından elde edilen Delaunay üçgenlerinden başka özellik noktalarının detayları da [16]'da indekslemeye katılmıştır.

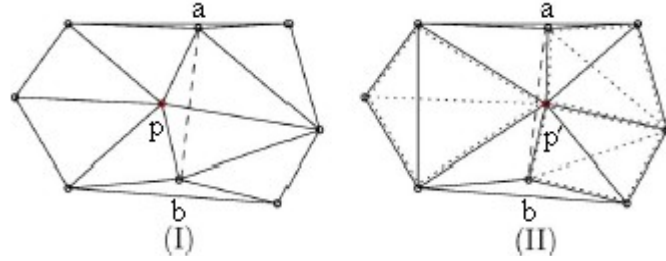


Şekil 2.11. İz çizgilerinin çatallanma noktalarının değişik durumları [16].

İz çizgilerinin çatal noktaları için çatal uçlarının çatallanma merkezinde çizilen

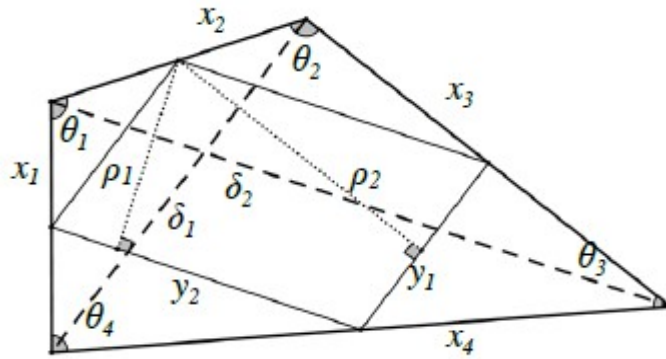
koordinat drtlklerine yerleřimleri Őekil 2.11 'de gsterilen farklı durumlarda olabilmektedir. zellik noktaları detayları olarak bu durum bilgileri indekslemeye katılmıştır. İz grntsndeki bozulmalardan Delaunay çgen yapısının fazla etkilenmedięi iddiasının yaptıkları deney sonucuna gre doęru olmadıęını Őekil 2.12'de rneęini vererek bildirmişlerdir.

Buna karřı dřk dereceli (0 ve 1. derece) Delaunay çgen yapısını nermişlerdir. Delaunay çgen yapısını nerdikleri bir algoritmadan geçirerek bu yapıyı elde etmektedirler. 0 ve 1. derece Delaunay çgenleri zellik noktalarındaki kçük kaymalarla yapısını deęiřtirmemektedirler.



Őekil 2.12. zellik noktalarının birinin hafif kaymasıyla Delaunay çgen yapısının deęiřmesi [16].

[17]'de ise indeksleme iin Őekil 2.13 'teki gibi zellik noktaları drtlleri kullanılmıştır. Fakat ibkey drtgenler dikkate alınmamıştır.



Őekil 2.13. zellik noktaları drtls [17].

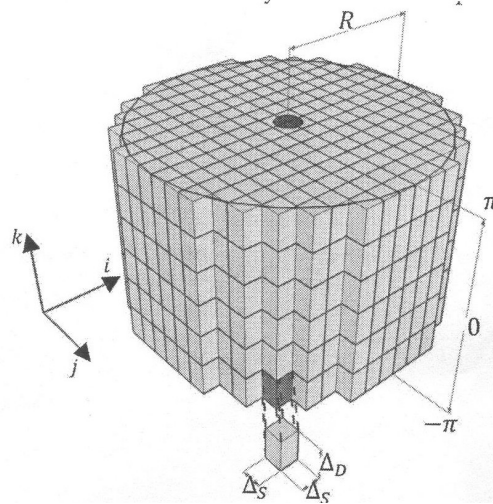
İndeks elemanı olarak řunlar kullanılmıştır:

$\varphi_1 = \theta_1 - \theta_3$  ve  $\varphi_2 = \theta_2 - \theta_4$  dörtgenin karşılıklı iç açılarının farkları: Özellik noktalarının birinin konumundaki kayma karşılıklı iki açıyı beraber değiştirdiğinden farktaki değişme daha az veya sabit olur.

$d_1$  ve  $d_2$ , köşegen uzunlukları: Bir veya karşılıklı iki özellik noktasının konumlarında kayma olursa diğer karşılıklı iki özellik noktasını birleştiren köşegenin uzunluğu değişmez.

Ayrıca iç paralelkenarın yükseklikleri kullanılmıştır. Bundan başka dörtgen ile paralelkenarın kenarları ve alanları kullanılarak elde edilen kompozit global özellik kullanılmıştır..

Bir diğer önemli çalışmada [18] yine özellik noktası temelli ama tamamen farklı bir yaklaşım ile indeksleme yapılmıştır. Minutia Cylinder – Code - MCC adını verdikleri temsil yöntemi  $m$  özellik noktasının etrafındaki özellik noktalarının bağıl konumlarını üç boyutlu uzaydaki  $x, y$  koordinatları, yön bilgilerini ise üç boyutlu uzaydaki yükseklik bilgisi olarak kodlar. Bu temsilde diğer özellik noktaları  $m$  özellik noktası merkezli bir silindir tarafından çevrilmiştir. Silindir her biri  $[-\pi, \pi]$  aralığındaki ( $m$  özellik noktasının yönüne göre) yön farkına karşılık gelen dilimlere ayrılır. Her dilim hücrelere ayrıştırılmıştır. Şekil 2.14 böyle bir silindiri göstermektedir.



Şekil 2. 14. Minutia – Cylinder- Code (MCC) temsili [18].

MCC' nin bit bazlı temsilinde her özellik noktasının konum ve yön bilgilerine karşılık gelen silindir hücreleri doludur. Diğerleri boştur. Silindir dolu hücrelerin 1'e diğerlerininin 0'a karşılık geldiği binary vektör olarak işlem görür. Bu vektör indekslemede kullanılmaya müsaittir ve önerilen yöntemin temelini oluşturur. Her özellik noktasından MCC temsili ile binary vektör elde edilir.

MCC temsili döndürmeden ve yer değiştirmeden etkilenmez.. Ten bozulmalarına karşı sağlamdır. Sabit uzunluktadır.

İndekslemede Konum Duyarlı Hashing (Locality Sensitive Hashing) – LSH yöntemi kullanılır. Uygulanan LSH'in basit örneği Şekil 2.15 ile beraber şöyle anlatılmıştır:

	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$	$l_9$	$l_{10}$	$l_{11}$	$l_{12}$	$l_{13}$	$l_{14}$	$l_{15}$	$f_{H_1}$	$f_{H_2}$	$f_{H_3}$
$v_1$	0	1	1	1	0	0	1	0	1	0	0	0	0	1	0	7	4	0
$v_2$	0	0	1	0	1	0	1	0	0	0	0	1	0	0	1	2	5	3
$v_3$	1	0	1	0	1	0	0	0	0	1	0	0	0	1	0	0	6	6
$v_4$	0	0	0	0	0	1	1	0	1	0	1	1	1	0	0	3	1	0
$v_5$	1	1	1	0	1	0	0	0	1	0	0	1	0	0	0	5	5	6
$v_6$	1	0	1	0	0	1	0	0	0	1	0	0	0	1	1	0	6	5
$v_7$	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	3	3	0
$v_8$	1	0	0	0	0	1	1	1	0	1	0	0	1	0	0	2	2	4
$v_9$	0	1	0	1	1	0	0	0	1	1	0	0	0	0	1	5	2	3
$v_{10}$	0	1	1	0	0	0	1	1	1	0	1	0	0	0	1	7	4	1

$H_1$		$H_2$		$H_3$	
0 = 000 <sub>2</sub>	{ $v_3, v_6$ }	0 = 000 <sub>2</sub>	$\emptyset$	0 = 000 <sub>2</sub>	{ $v_1, v_4, v_7$ }
1 = 001 <sub>2</sub>	$\emptyset$	1 = 001 <sub>2</sub>	{ $v_4$ }	1 = 001 <sub>2</sub>	{ $v_{10}$ }
2 = 010 <sub>2</sub>	{ $v_2, v_8$ }	2 = 010 <sub>2</sub>	{ $v_8, v_9$ }	2 = 010 <sub>2</sub>	$\emptyset$
3 = 011 <sub>2</sub>	{ $v_4, v_7$ }	3 = 011 <sub>2</sub>	{ $v_7$ }	3 = 011 <sub>2</sub>	{ $v_2, v_9$ }
4 = 100 <sub>2</sub>	$\emptyset$	4 = 100 <sub>2</sub>	{ $v_1, v_{10}$ }	4 = 100 <sub>2</sub>	{ $v_8$ }
5 = 101 <sub>2</sub>	{ $v_5, v_9$ }	5 = 101 <sub>2</sub>	{ $v_2, v_5$ }	5 = 101 <sub>2</sub>	{ $v_6$ }
6 = 110 <sub>2</sub>	$\emptyset$	6 = 110 <sub>2</sub>	{ $v_3, v_6$ }	6 = 110 <sub>2</sub>	{ $v_3, v_5$ }
7 = 111 <sub>2</sub>	{ $v_1, v_{10}$ }	7 = 111 <sub>2</sub>	$\emptyset$	7 = 111 <sub>2</sub>	$\emptyset$

	$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$	$l_9$	$l_{10}$	$l_{11}$	$l_{12}$	$l_{13}$	$l_{14}$	$l_{15}$	$f_{H_1}$	$f_{H_2}$	$f_{H_3}$
$v_5$	1	1	1	0	0	0	1	0	1	0	1	0	0	1	0	7	4	4

Şekil 2.15. LSH uygulaması örneği [18].

15 bit uzunluğunda  $v_1, v_2, v_3, \dots, v_{10}$  vektörleri olsun. Bir  $v = [i_1, i_2, i_3, \dots, i_{10}]$

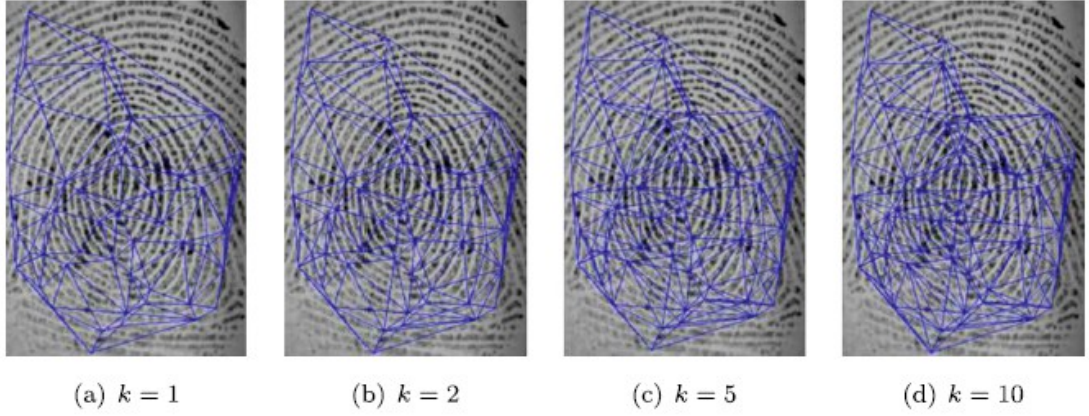
vektörünün daha düşük boyutlu (burada 3 boyutlu) projeksiyonları  $H_1 = \{i_2, i_7, i_{10}\}$ ,  $H_2 = \{i_3, i_{10}, i_{12}\}$ ,  $H_3 = \{i_1, i_5, i_{15}\}$  olsun.  $H_1, H_2, H_3$  binary vektörlerini bunların karşılığı olan doğal sayılara eşleyen  $f_{H_1}, f_{H_2}, f_{H_3}$  hash fonksiyonları olsun.  $v_1, v_2, v_3, \dots, v_{10}$  binary vektörleri ve bu vektörler üzerindeki  $f_{H_1}, f_{H_2}, f_{H_3}$  hash fonksiyonlarının verdiği değerler aşağıdaki tabloda gösterildiği gibi olsun. Sorgulanan parmak izine ait vs binary vektörü ve onun üzerinde  $f_{H_1}, f_{H_2}, f_{H_3}$  fonksiyonlarının verdiği değerlere göre  $\{v_1, v_{10}, v_8\}$ , sonuç kümesi olarak bulunur.

Hamming mesafesine göre yapılan benzerlik hesabında  $v_1$  ve  $v_{10}$  en çok benzeyen vektörler,  $v_8$  ise yanlış aday olarak bulunur.

MCC sadece özellik noktalarının konum ve yön bilgilerini kullanır. Bu çalışmada [18] MCC yönteminin diğer metodlara göre daha az özellik kullanıldığı ve daha başarılı olduğu belirtilmiştir. MCC kısa zamanda diğer araştırmacıların dikkatini çekmiştir. Araştırmacılar çalışmalarının başarısını MCC'ninki ile de karşılaştırmaya başlamıştır.

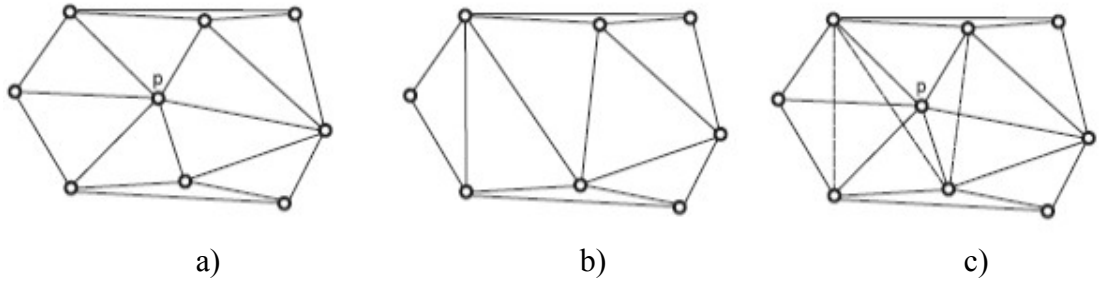
Özellik noktaları üçlüsü temelli bir başka indeksleme çalışmasında [19] düşük (0 ve 1) dereceli Delaunay üçgenlerinin sayısının az olup buna bağlı olarak parmak izi tanımlamada yetersizliğinin olduğu, bazı önemli geometrik özelliklerin bu yapı içinde kaybedildiği, dolayısıyla indeksleme işleminin doğruluğunun etkilendiği bildirilmiştir. Buna karşı 1. dereceden daha yüksek Delaunay üçgenleştirmesi önerilmiştir. Şekil 2.16 'daki gibi daha yüksek dereceli Delaunay üçgenleştirmesi daha fazla sayıda üçgen ortaya çıkarır. Kenarları düşük kaliteli bölgelerden geçen üçgenler indekslemeye katılmaz. Eleme sebebiyle oluşan üçgen azlığı yüksek dereceli Delaunay yapısının sağladığı daha fazla üçgen ile telafi edilir.





Şekil 2.16. Yüksek dereceli Delaunay üçgenleri [19].

Genişletilmiş Delaunay üçgenlerinin [20] önerildiği çalışmada Delaunay üçgenlerinin yapısının kararlı olmadığına, düşük dereceli dereceli Delaunay üçgenlerinin ise sadece özellik noktası konumlarındaki kayma etkilerini azaltabildiğine vurgu yapılarak genişletilmiş delaunay üçgenleri yapısı önerilmiştir. Önerdikleri yapı Delaunay üçgenlerinden türetilir ve Delaunay üçgenlerinin tümünü içerir. Delaunay üçgenlerine ek olarak Şekil 2.17' de örneği verildiği gibi her bir özellik noktası eksik olduğunda ortaya çıkabilecek üçgenler de katılır. Böylece tespit edilemeyen özellik noktalarına karşı kararlılık sağlanmaya çalışılır.



Şekil 2.17. Delaunay ve genişletilmiş Delaunay üçgenleri, a) Delaunay üçgenleri, b) P yokken Delaunay üçgenleri, c) Genişletilmiş Delaunay üçgenleri [20].

İz çizgilerinin yönelim ve frekans bilgilerinin kullanıldığı bir çalışma mevcuttur [21]. Bu çalışma [21] pek kullanılmayan iz çizgileri frekansının ayırt ediciliğine dikkat çekmiştir. Yerel yönelim ve frekans değerlerinden başka ortalama frekans ve yatay simetriyi gösteren bir değer de kullanılmıştır.

Hem birinci seviye hem de ikinci seviye özelliklerin beraber kullanıldığı çalışmada

[22], MCC [18] ile iz çizgilerinin yönelim ve frekans bilgilerinin [21] birleştirilerek kullanılması önerilmiştir.

## BÖLÜM 3

### İZ DESENİ TEMSİLİ

Parmak izini temsil edebilecek yapı iskeletleştirilmiş iz çizgileridir [23]. Fakat çoğu parmak izi karşılaştırma yöntemi özellik noktası temelli olduğu gibi [8], çoğu indeksleme algoritması da parmak izini temsil etmek ve indeksler oluşturmak için özellik noktalarını temel alarak kullanır [20].

Parmak izinin yerel ölçekteki bölgeleri, elastik bozulmadan izin tamamına göre nispeten daha az etkilenir [16,25]. Herhangi 3 özellik noktası [10-11] yerine Delaunay üçgenleri [12] indekslemede kullanılırsa parmak izinin daha yerel bölgeleri işleminden geçirilerek indekslemeye katılmış olur.

Delaunay üçgenleştirmesine katılan bir özellik noktasının aynı parmağa ait başka bir izde konumunun kaymasının ya da tespit edilememesinin etkisi yerel ölçekte kalsa da [12], Delaunay üçgenleri gürültü ve bir takım bozulmalara duyarlıdır [16]. Düşük dereceli Delaunay üçgenleştirmesi [16] ve genişletilmiş Delaunay üçgenleştirmesi [20] çabaları da buna çözüm bulmak içindir. Dolayısıyla parmak izinin daha yerelleşmiş kısımları değerlendirildiğinde bozulmanın indekslemeye etkisinin daha az olması beklenmektedir.

Bu çalışma kapsamında parmak izinin üçgensel bölgeleri içindeki iz desenleri temsil edilmiş ve bu temsile iz deseni temsili denilmiştir.

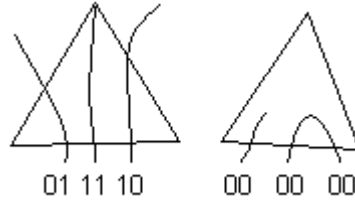
#### 2.1. İZ DESENİ TEMSİLİNİN ELDE EDİLİŞİ

Üçgen içindeki iz deseninin temsili şöyle elde edilir: Üçgenin her bir kenarı için temsilden 32 bit ayrılır. Her bir iz çizgisi için 2'şer bit, iz çizgisi tarafından kesilen üçgen kenarına düşen 32 bit' ten ayrılır.

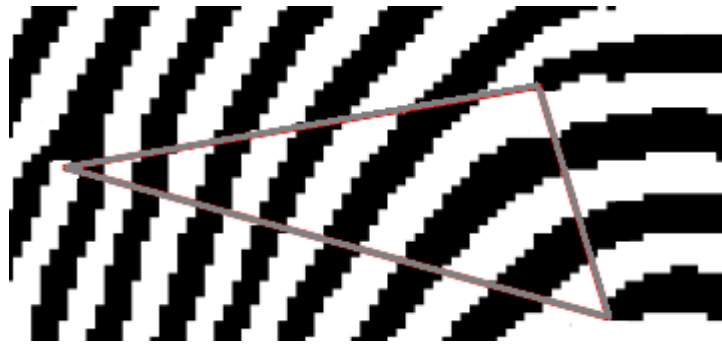
İncelenen kenarı kesen iz çizgileri sırayla kenarı kestiği yerden üçgenden çıkıncaya kadar izlenir. Üçgeni diğer kestiği nokta veya noktaların durumuna göre aşağıdaki gibi değer alır.

- Yine aynı kenardaysa 00,
- Sol taraftaki kenardaysa 01,
- Sağ taraftaki kenardaysa 10,
- Hem sağ hem sol taraftaki kenardaysa 11

Binary değerleri indeksin ayrılan yerine yazılır. İzlenen iz çizgisi içeride kalıyor, diğer kenarları kesmiyorsa 00 binary değeri yazılır. Anlatılan durumların örnekleri Şekil 3.1 ile resmedilmiştir.



Şekil 3.1. İz çizgilerinin iz deseni temsiline katılması.



En kısa kenardan başlayarak saat yönünün tersinde devam edildiğinde kenar başına düşen indeks kısımları:

```
01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10 10 10 10 10 00 00 00 00 00 00 00 00 00 00 00
01 01 01 01 01 10 10 00 00 00 00 00 00 00 00 00
```

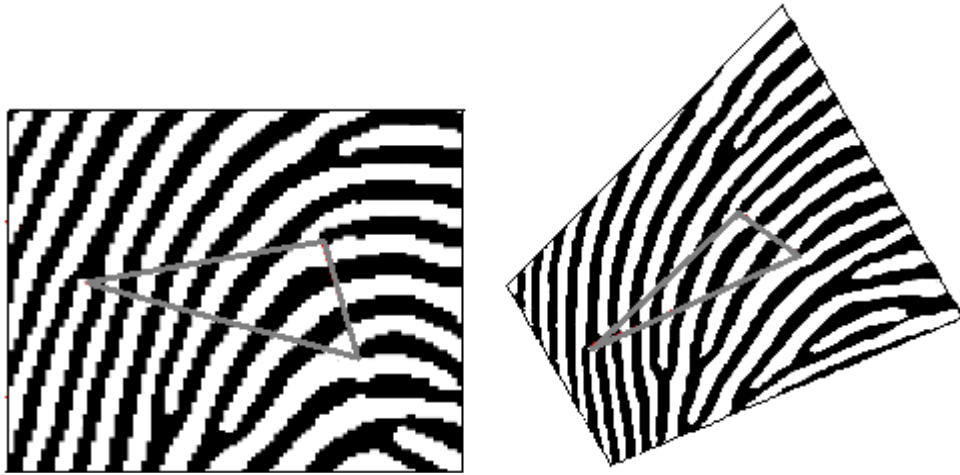
Şekil 3.2. İz deseninin elde edilmesi.

16'dan az iz çizgisi olması halinde 32 bit'ten eksik kalan yerlere 00 binary değerleri yazılır. 16'ıncı iz çizgisinden sonrakiler ise dikkate alınmaz.

Üç kenar için bu işlemler tekrarlanıp Şekil 3.2 ile örneği verilen 96 bit'lik binary değer elde edilir. Özellik noktalarının tipleri, oluşturduğu üçgenin şekli gibi elde edilebilecek diğer değerler için 32 bit daha katılarak 128 bit'lik temsil değerleri elde edilir. Bu çalışmada iz deseni temsillerinin son 32 bit'lik kısmı boş tutulmuştur.

### 3.2. İZ DESENİ TEMSİLİNİN ÖZELLİKLERİ

İndeks oluşturmada kullanılan üçgensel bölgenin sınırları deri basınç ve elastığına bağlı bozulmalara karşı değişse de iz desenine bağlı indeksler bu değişimlere karşı sağlam durur. Ayrıca iz deseni temsili döndürmeden ve yer değiştirmeden etkilenmez. Şekil 3.3' te de gösterildiği gibi iz görüntüsü hangi yönde esnetilirse esnetilsin iz çizgileri üçgenin aynı kenarlarını keser. Bu sebeple aynı iz deseni temsili elde edilir.



Şekil 3.3. İz görüntüsü ve gerdirilmiş hali.

Üçgen ağının stabilitesi elastik bozulmaya duyarlı olsa da Delaunay üçgenleri yerine elastik bozulmalara karşı daha kararlı oldukları bildirilen düşük dereceli (0. ve 1. derece) Delaunay üçgenleri [12] kullanılarak da indeksler elde edilebilir. Ayrıca tespit edilemeyen özellik noktaları problemine karşı daha kararlı olduğu bildirilen genişletilmiş Delaunay üçgenleri [20] de kullanılabilir.

Üçgenleri kesebilecek iz çizgileri sayısı sabit olmasa da üçgen kenarı başına belli sayıdan (bu çalışma kapsamında 16) sonrakiler dikkate alınmadığından iz çizgileri temsilleri sabit uzunluktadır.

Yan yana olan iz çizgileri genelde aynı kenarları kestiği için elde edilen iz deseni temsili birbirinin tekrarı olan 2 bit'lik parçaları içerir. Dolayısıyla sıkıştırmaya müsaittir.

Üçgensel bölgeler özellik noktaları kümesi üzerinde yapılacak Delaunay üçgenleştirmesi ile elde edilirse özellik noktalarının arasında kalan iz çizgileri deseninin tamamı temsil edilmiş olur. Şekil 3.4 bunun örneğini FVC2000 DB1 veritabanınındaki [26] 101\_1.tif adlı parmak izi resmi için göstermektedir.



Şekil 3.4. Delaunay üçgenleştirmesi ile temsil edilen iz deseni parçaları.

Üçgenleştirilmiş bölgenin dışındaki alanın da iz deseni temsil edilmek istenirse izlerin resim kenarında kalan uç noktaları özellik noktaları olarak eklenirse iz çizgileri deseninin tümü temsil edilebilir. Bu çalışma parmak izinin temsilden ziyade yapılan temsillerin indekslemede kullanılması ile ilgili olduğundan bu konu incelenmemiştir.

Birinci derece Delaunay üçgenleri [12] ya da genişletilmiş Delaunay üçgenleri [20] ile iz deseni temsil edilmek istenirse aynı parmak izi resmi için temsil edilecek desen parçaları sırasıyla Şekil 3.5' ve Şekil 3.6' daki gibi olur.



Şekil 3.5. Birinci derece Delaunay üçgenleştirmesi ile temsil edilen iz deseni parçaları.



Şekil 3.6. Genişletilmiş Delaunay üçgenleştirmesi ile temsil edilen iz deseni parçaları.

İz deseni temsili elastik bozulmalara karşı sağlam durabilse de özellik çıkarma

hataları sebebiyle özellik noktalarının yanlış tesbit edilmesi iz deseninin değişmesine sebep olur. Özellik noktası türü sağlam bir özellik değildir [27] ve bu iz deseni temsilinin parmak izi kimliklendirme sistemleri için başarısını tehdit eden bir durumdur.

### **3.3. İZ DESENİ TEMSİLİNİN ÖZELLİK NOKTASI TESPİT HATALARINA KARŞI DUYARLILIĞI**

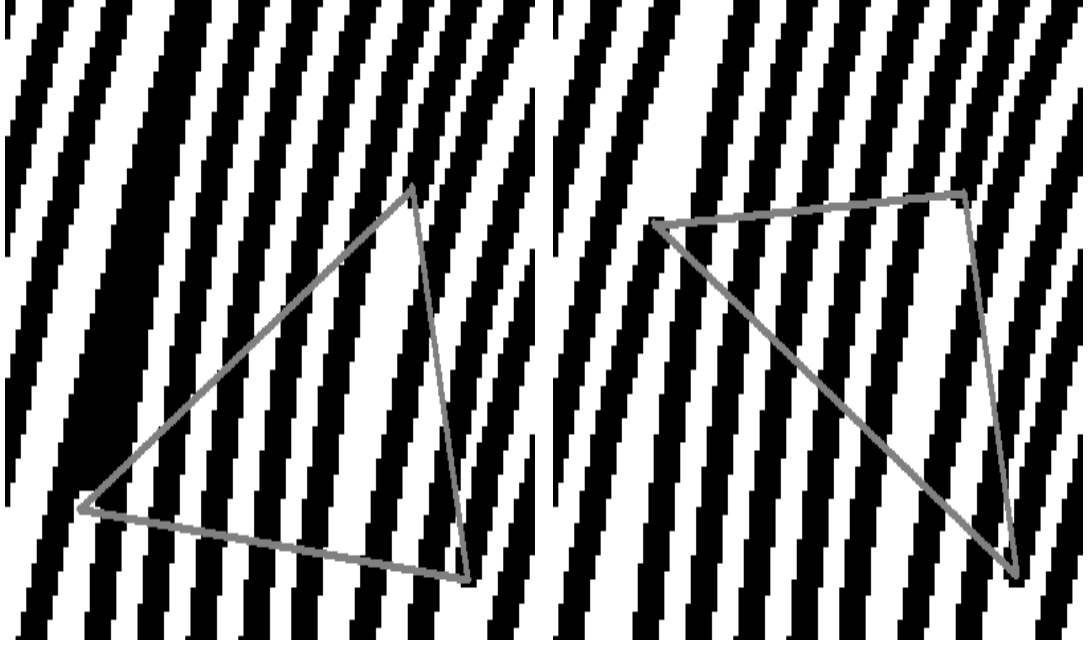
Özellik noktası türü (uç, çatallanma) sağlam bir özellik değildir [27]. Ayrıca özellik noktası türünün olduğundan farklı tespit edilmesi iz desenini ve iz deseni temsilini değiştirir. İz deseni temsili özellik noktası türleri bilgisini taşımasa bile çoğunlukla temsil ettiği iz deseni yanlış özellik noktası türü tespiti sebebiyle değiştiği için iz deseni temsili değişir.

Ayrıca özellik noktası türünün farklı tespit edilmesi iz deseni temsilinin birden fazla yerindeki bit ikililerinin değerlerinin değişmesine sebep olur. Çünkü genellikle bir iz çizgisi, iz deseni temsilinde birden fazla yerde temsil edilir.

Şekil 3.7, özellik noktalarından biri iz çizgisi ucu yerine iz çizgisi çatallanması olarak tespit edilen üçgensel iz deseni temsilinin nasıl değiştiğini gösterir. İz deseni temsilini değiştiren bit ikilileri ve özellik noktası türlerinin son çeyrekte gösterildiği varsayılarak değeri değişen özellik noktası bitleri altı çizili olarak gösterilmiştir.

Bir biti bile değişen iz deseni temsili indekslemede aday parmak izlerinin kaçırılmasına sebep olabilir. Bu yüzden iz deseni temsili özellik noktası türü tespit hatalarına karşı duyarlıdır.





```

101000000000000000000000000000000000000000000000000
010101010100000000000000000000000000000000000000000
101010101001010000000000000000000000000000000000000
001000000000000000000000000000000000000000000000000

```

(a)

```

101100000000000000000000000000000000000000000000000
110101010101000000000000000000000000000000000000000
101010101011010000000000000000000000000000000000000
011000000000000000000000000000000000000000000000000

```

(b)

Şekil 3.7. Bir özellik noktasının yanlış tespit edilmesi ile iz deseni temsilinin değişmesi, a) Özellik noktası doğru tespit edildiğinde, b) Özellik noktası yanlış tespit edildiğinde.

Bu problemin de etkisiyle veritabanında karşılığı hiç bulunamayan parmak izleri olabilmektedir. Bu durumun etkisini ölçmek için şöyle bir deney yapıldı: İz deseni temsilleri indeks olarak kullanılarak parmak izleri indekslendi. İndeksler veritabanında kaynağı olan özellik noktaları üçlülerinin oluşturduğu üçgenin kenar uzunlukları, alanı, binarize edilmiş bağlı özellik noktaları yönleri ve ait oldukları parmak izinin kimlik bilgisi ile beraber tutuldu. (Üçgenleştirme Delaunay üçgenleştirmesi ile yapıldı.) Sorgulanan parmak izinden elde edilen aynı tür bilgilerle indekslenmiş veritabanlarında aramalar yapıldı.

Çizelge 3.1, üç farklı veritabanı için yapılan deneyin sonuçlarını özetlemektedir. Buna göre FVC2000 DB2 [26], FVC2000 DB3 [26] ve FVC2002 DB1 [28] veritabanlarında yapılan aramaların sırasıyla %15, %9 ve %5' i hiç sonuç vermemektedir. (Bu oranlar tabloda kayıp oranı olarak belirtilmiştir.)

% 30'a kadar artan penetrasyon oranlarına karşı tutma oranlarının 1'e tümleyeninin arandığında karşılığı bulunamayan parmak izlerinin oranlarına (kayıp oranına) yaklaşması, karşılığı bulunamayan parmak izleri olmaması durumunda %30 penetrasyon oranı için %100 tutma oranına çok yaklaşılabileceği fikrini vermektedir. Penetrasyon oranı indeksleme sisteminin verdiği aday parmak izi sayısının veritabanındaki tüm parmak izlerinin sayısına oranıdır.

Çizelge 3.1. Değişik parmak izi veritabanlarındaki aramalar için kayıp oranları.

Penetrasyon oranı (%)	Tutma oranı (%)		
	FVC2000 DB2	FVC2000 DB3	FVC2002 DB1
1	67	60	86
5	80	73	92
10	82	76	93
15	83	80	94
20	84	84	95
25	85	86	95
30	85	89	96
<b>Kayıp oranı (%)</b>	15	9	5

Aramalar için kullanılan algoritma veritabanındaki tüm parmak izlerini puan değerlerine göre sıralamaktadır. Puanı sıfır olan parmak izleri de izin verilen penetrasyon oranı için sonuç listesinde olabildiği için FVC2002 DB1 veritabanı [28] için %30 penetrasyon oranı için %95'den büyük penetrasyon oranı hesaplanmıştır. Bu sonuçlara göre en az bir parmak izinin aranması sonunda sonuç listesinde verilen sıfır puanlı parmak izinin tesadüfen doğru parmak izi olarak bulunduğu anlaşılıyor.

Arama sonunda karşılığı bulunamayan parmak izleri için küçük farklılıkların tolere edilmesi faydalı olabilir. Bu, indeksleme işleminin seçiciliğini azaltabilir ama sadece arama sonunda sıfır puanlı ya da çok düşük puanlı parmak izleri sonuç kümesini oluşturduğu zaman yapılırsa bu probleminden kurtulabilir.

Küçük farklılıkların tolere edilmesi için iz deseni temsilinin küçük farklılıklar için daha az sayıda farklılık ortaya çıkarması sağlanabilir.

## BÖLÜM 4

### ÖNERİLEN İNDEKSLEME YÖNTEMLERİ

Bir çok indeksleme metodu [10-12,14-20] gibi, özellik noktalarını kullanmaktadır. Özellik noktalarını kullananlardan [10-12,14-16,19-20] özellik noktaları kümesinden özellik noktaları üçlüleri oluşturur. Fakat Minutia Cylinder Code (MCC) [18], her bir özellik noktası için oluşturulur. MCC [18] sadece özellik noktalarının konum ve yön bilgilerini kullanır.

MCC ile ilgili olarak diğer metodların genellikle daha fazla özellik kullandığı, MCC ile indekslemenin neredeyse tüm durumlarda diğerlerinden daha iyi performans gösterdiği bildirilmiştir [18]. Mesela özellik noktaları arasındaki iz çizgileri sayıları [10,19-20]' de, özellik noktaları tipi [11]' de, özellik noktaları yakınlarındaki iz çizgileri iskeletinden üretilen özellik noktaları detayı [16]' da kullanılmıştır. Yapılan diğer bir çalışma [22] iz çizgilerinin frekansını ve yönelimini kullanmayı önermiştir.

Özellik noktaları çiftleri arasındaki mesafe ve yön ayrık değerlere sahip değildir. Ayrık değerleri gerektiren metodlar kutulama (binning) mekanizması kullanırlar. Kutulama değer aralıklarına göre gruplandırma yapma yöntemidir. Her bir gruba kutu denir. Kutulara karşık gelen değer aralıkları birbirleriyle örtüşmez. Distorsiyonu tolere etmek için büyük boyutlu kutular kullanılırsa indeksleme uzayı indirgenir ve indeksleme algoritmasının performansı düşer [11]. Bu yüzden [11]'deki indeks elemanları özellik noktaları üçlülerinin daha fazla özelliğinden türetilmiştir.

Genişletilmiş delaunay üçgenleri ile yapılan indekslemede [20] distorsiyon düşünülerek sorgulanan parmak izleri için fazladan özellik vektörleri üretilmektedir. MCC yapısında ise distorsiyona ve küçük özellik çıkarma hatalarına karşı dayanıklı olması için her bir özellik noktasının katkısı keskin olmayan bir fonksiyon ile tanımlanır [18].

Bu çalışma kapsamında önce sadece özellik noktaları bilgilerini kullanan özellik noktaları üçlüsü temelli indeksleme uygulaması gerçekleştirilmiş, sonra bu uygulamaya Bölüm 3' te tarif edilen iz deseni temsilleri de eklenerek iz deseni temsillerinin faydası gösterilmiştir. İlk uygulamanın ayrı başlık altında tutulmasının sebebi son yıllarda literatürde önemli yer tutmaya başlayan MCC [18] ile indeksleme başarısını geride bırakmış ve parmak izlerinin sadece özellik noktaları ile temsil edilip iz deseni temsillerinin elde edilemeyeceği sistemlerde kullanılabilecek müstakil bir indeksleme yöntemi olabileceğinden dolayıdır.

#### 4.1. ÖZELLİK NOKTALARI ÜÇLÜSÜ TEMELLİ İNDEKSLEME METODU

Burada önerilen indeksleme metodu özellik noktaları üçlülerini kullanılır. Özellik noktalarının 2 boyutlu uzayda konumu ve yönü vardır. İki tür özellik noktası olabilir. Ya iz çizgisinin ucudur ya da iz çizgisinin çatallanma noktasıdır. Fakat önerilen yöntemde özellik noktası tipi dikkate alınmaz.

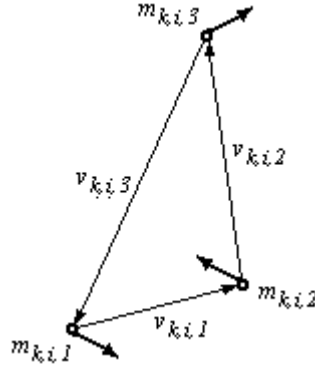
Özellik noktaları üçlülerini oluşturmak için özellik noktaları kümesi üzerinde düşük dereceli delaunay üçgenleştirilmesi [16] işlemleri yapılır. Her bir delaunay üçgeninin köşeleri kendilerinden bir indeks çıkarılan özellik noktaları üçlüsüdür.

İndekslerin çıkarıldığı özellikleri tanımlamak için aşağıdaki tanımlamaları yapmak gerekir:

$T_{k,i}$  ( $k \in \{1, 2, 3, \dots, m\}$ ,  $i \in \{1, 2, 3, \dots, n_k\}$ ) ( $m_{k,i,1}$ ,  $m_{k,i,2}$ ) saat yönünün tersinde giderken en yakın özellik noktaları ikilisi olmak şartıyla indekslenmiş bir veritabanındaki  $f_k$  parmak izine ait ( $m_{k,i,1}$ ,  $m_{k,i,2}$ ,  $m_{k,i,3}$ ) özellik noktası üçlüsü olsun.

Ayrıca  $v_{k,i,1}$ ,  $v_{k,i,2}$ ,  $v_{k,i,3}$  sırasıyla  $m_{k,i,1}$ 'den  $m_{k,i,2}$ 'e,  $m_{k,i,2}$ 'den  $m_{k,i,3}$ 'e ve  $m_{k,i,3}$ 'den  $m_{k,i,1}$ 'e tanımlanmış vektörler olsun.

Özellik noktaları üçlüsü ve arasındaki vektörler Şekil 4.1' de de gösterilmiştir.



Şekil 4.1. Özellik noktaları üçlüsü ve aralarındaki vektörler.

#### 4.1.1. ÖZELLİK NOKTALARI ÜÇLÜSÜNÜN ÖZELLİKLERİ

İndeksleri elde etmek için kullanılan özellikler vektör uzunlukları, vektörlerle özellik noktası yönlerinin aralarında kalan açılar ve özellik noktalarının birbirlerine göre bağlı açılarıdır.

Bir  $T_{k,i}$  özellik noktaları üçlüsü için  $l_{k,i,1}$ ,  $l_{k,i,2}$  and  $l_{k,i,3}$  vektör uzunlukları sırasıyla  $v_{k,i,1}$ ,  $v_{k,i,2}$ ,  $v_{k,i,3}$  vektörlerinin uzunluklarıdır. Vektörlerin özellik noktası yönlerine göre yaptığı açılar  $\alpha_{k,i,1}$ ,  $\alpha_{k,i,2}$ ,  $\alpha_{k,i,3}$ , sırasıyla  $v_{k,i,1}$  ile  $\theta_{k,i,1}$ ,  $v_{k,i,2}$  ile  $\theta_{k,i,2}$  ve  $v_{k,i,3}$  ile  $\theta_{k,i,3}$  arasındaki açılardır. Birbirlerine göre bağlı özellik noktası yönleri  $\beta_{k,i,1} = \theta_{k,i,1} - \theta_{k,i,2}$ ,  $\beta_{k,i,2} = \theta_{k,i,2} - \theta_{k,i,3}$  and  $\beta_{k,i,3} = \theta_{k,i,3} - \theta_{k,i,1}$  dir.

Elde edilen  $T_{k,i}$ 'nin 9 boyutlu ( $l_{k,i,1}$ ,  $l_{k,i,2}$ ,  $l_{k,i,3}$ ,  $\alpha_{k,i,1}$ ,  $\alpha_{k,i,2}$ ,  $\alpha_{k,i,3}$ ,  $\beta_{k,i,1}$ ,  $\beta_{k,i,2}$ ,  $\beta_{k,i,3}$ ) vektöründen  $I_{k,i}$  indeksi çıkarılır.

#### 4.1.2. İNDEKSLERİN ÇIKARILMASI

Vektör uzunlukları özellik noktaları arasındaki mesafeleri temsil eder. Elastik distorsiyon kısa mesafeler için küçük ama uzun mesafeler için daha büyüktür. Dolayısıyla uzun vektörlere karşılık gelen kutular daha çok toleransa izin vermelidir. Bu sebeple daha uzun vektörlere daha geniş kutular karşılık gelmelidir.

FVC2002 DB1 [28] ile bu çalışma kapsamında yapılan deneylere göre 7 piksel'den az özellik noktaları arası mesafelerin olması çok nadir olduğundan (%1'den az) en

küçük vektörlere karşılık gelecek kutu genişliği 7 pikselden daha büyük ve özellik noktaları arası mesafeler dağılımının bir kısmını kapsayabilecek genişlikte olmalıdır.

Önerilen yöntem vektör uzunlukları için 8 kutu ile uygulanmaktadır. Bunlardan ilkinin genişliği 16 piksel'dir. İkinci kutunun genişliği 9 piksel ve geri kalanların sonuncusu hariç her bin kutu öncekinden 2 piksel daha geniştir. 100 piksel'den uzun her mesafe (0'dan başlayanlardan) 7 olarak kutulanır.

$L$  vektör uzunluğu  $l$  için herhangi bir kutu olsun ( $L \in \{0, 1, 2, \dots, 7\}$ ) ve  $W_L$   $L$ 'nin genişliği olsun. Öyleyse piksel biriminden  $W_L$  eşitlik (4.1) ile ifade edilir.

$$W_L = \begin{cases} 16, & L = 0 \\ 7 + 2L, & 0 \leq L < 7 \end{cases} \quad (4.1)$$

Buna göre  $L$  de eşitlik (4.2)'deki gibi ifade edilebilir.

$$L = \begin{cases} 0, & 0 < l < 16 \\ j, & (3 + j)^2 \leq l < (4 + j)^2 \quad (j \in \{1, 2, \dots, 6\}) \\ 7 & l \geq 100 \end{cases} \quad (4.2)$$

O halde  $T_{k,i}$  özellik vektörünün  $l_{k,i,1}$ ,  $l_{k,i,2}$ ,  $l_{k,i,3}$ ,  $\alpha_{k,i,1}$ ,  $\alpha_{k,i,2}$ ,  $\alpha_{k,i,3}$ ,  $\beta_{k,i,1}$ ,  $\beta_{k,i,2}$ ,  $\beta_{k,i,3}$  bileşenleri kutulanarak sırasıyla  $L_{k,i,1}$ ,  $L_{k,i,2}$ ,  $L_{k,i,3}$ ,  $A_{k,i,1}$ ,  $A_{k,i,2}$ ,  $A_{k,i,3}$ ,  $B_{k,i,1}$ ,  $B_{k,i,2}$  ve  $B_{k,i,3}$  elde edilir. Bu değerler ikilik sistemde uç uca eklenerek  $I_{k,i}$  indeksi elde edilir.

#### 4.1.3. YEDEK İNDEKSLER

$f$  toleranslı  $\tau$  olan bir özellik vektörü,  $F$   $f$ 'nin  $F_l$  den  $F_u$ 'ya kadar olan değerlerini kaplayan bir kutu ( $F \in \{0, 1, 2, \dots, Fn-1\}$ ) ve  $f_p$   $F$ 'ye karşılık gelen  $f$ 'nin bir değeri olsun .

$\tau$ 'nin bir kutunun yarısını kaplamadığı varsayılarak,  $f_p - F_l < c\tau/2$  ise  $F$  için üretilen indeksin yanında  $F-1$  değeri için ilave bir indeks yedek olarak üretilir ( $0 \leq c \leq 1$ ). Benzer şekilde eğer  $F_u - f_p < c\tau/2$  ise  $F+1$  değeri için fazladan bir indeks yedek indeks olarak üretilir. Eğer  $c = 0$  ise hiç yedek indeks üretilmez. Yukarıdaki açıklamalar tolerans değerinin  $F_l$  ya da  $F_u$ 'ya bağlı olabileceği dikkate alınarak düşünülmelidir.

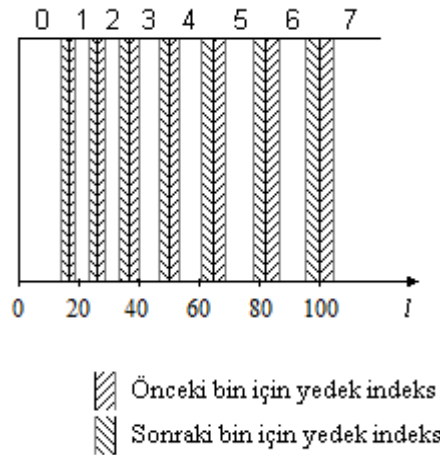
Yöntem  $T_{k,i}$  özellik vektörlerinin  $l_{k,i,1}$ ,  $l_{k,i,2}$ ,  $l_{k,i,3}$ ,  $\alpha_{k,i,1}$ ,  $\alpha_{k,i,2}$  ve  $\alpha_{k,i,3}$  bileşenleri için üretilen yedek indekslerle uygulanmaktadır.

Vektör uzunlukları için tolerans değerleri kendisi için indeks üretilen kutuya bağlıdır.  $\tau_l$  ve  $\tau_u$  sırasıyla  $F_l$  ve  $F_u$ 'daki tolerans değerleri olsun ve sırasıyla  $L-1$  ve  $L+1$  için yedek indekslerin gerekip gerekmediğini test etmek için kullanılsın. Bu değerler deneysel olarak bulunup eşitlik (4.3)'teki gibi ifade edilmiştir.

$$\tau_l = \begin{cases} 6, & L = 0 \\ 0,066L^2 + 1,2L + 5,4, & 1 < L \leq 7 \end{cases} \quad (4.3)$$

$$\tau_u = 0,066(L+1)^2 + 1,2(L+1) + 5,4 \quad (0 \leq L \leq 7)$$

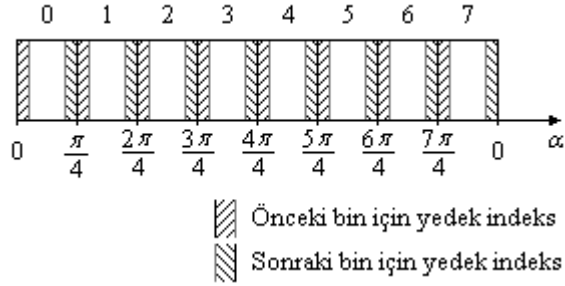
Vektör uzunluklarına karşılık gelen kutular ile önceki ve sonraki kutular için yedek indekslerin üretileceği değerler Şekil 4.2' de gösterilmiştir.



Şekil 4.2. Vektör uzunlukları için bin'ler ve yedek indekslerin üretildiği değerler.



Özellik noktası yönlerine göre vektör açıları için sabit bin genişliği kullanıldığı gibi  $\pi / 10'$  luk sabit tolerans değeri uygulanmıştır. Şekil 4.3 uygulanan modelde yedek indekslerin üretildiği değerleri ve vektör açıları için bin'leri göstermektedir.



Şekil 4.3. Bağlı özellik noktası yönleri için bin'ler ve yedek indekslerin üretildiği değerler.

Birbirlerine göre bağlı özellik noktası yönleri yedek indekslerin üretilmesi için dikkate alınmamaktadır. Çünkü yapılan deneyler bunun önemli bir etkisinin olmadığını göstermiştir.

Ayrıca aynı özellik vektöründeki birden fazla bileşenin yedek indeks gerektirmesi ihtimali ihmal edilmiştir. Çünkü deneyler bu durumu hesaba katmanın faydasının olmadığını göstermiştir.

Yedek indekslerin sayısı kabaca tolerans değerlerinin bin genişliklerine oranından tahmin edilebilir.

#### 4.1.4. İNDEKSLERİN DEPOLANMASI

$f_k$  'nın her  $I_{k,i}$  indeksi hash olarak kullanılır. Hash tablosunda bunlara karşılık gelen değerler ilgili parmak izinin ID numarasını  $k$ ,  $T_{k,i}$  için  $l_{k,i,1}$ ,  $l_{k,i,2}$ ,  $l_{k,i,3}$ ,  $\alpha_{k,i,1}$ ,  $\alpha_{k,i,2}$ ,  $\alpha_{k,i,3}$  özellik vektörü bileşenlerini, referans özellik noktası olarak  $m_{k,i,1}$  'in konum ve açı bilgilerini içerir. İndeksler 27 bitlik sayılar olduğu için hash tablosu  $2^{27}$  anahtar içerir.

Her  $I_{k,i}$  indeksi ile  $T_{k,i}$  özellik noktaları üçlüleri arasında bire bir eşleme vardır. Çünkü veritabanındaki parmak izleri için yedek indeksler üretilmemektedir.

#### 4.1.5. SORGU CEVABININ ÇIKARILMASI

Distorsiyonla başa çıkmak amacıyla sorgulanan parmak izleri için yedek indeksler çıkarılmasına izin verilmiştir.

$f_q$  sorgulanan bir parmak izi  $I_{q,j}$  ( $j \in \{1, 2, 3, \dots, n_q\}$ )  $f_q$  'dan elde edilen indeksler olsun. Sorgulanan  $f_q$  'ya karşılık aday listesini elde etmek için veritabanındaki her  $f_k$  parmak izi için  $S_{k,q}$  skoru hesaplanır. Vektör uzunlukları için  $dsl_{k,i,q,j,t}$  ve özellik noktalarına göre vektör açıları için  $d\alpha_{k,i,q,j,t}$  benzemezlik değerleri 7 ise iki vektörün birbirlerine yakın olmadığını, aksi takdirde vektörlerin birbirine yakın ya da daha yakın olduğunu ifade eder. Bu değerler eşitlik (4.4) ve eşitlik (4.5)'te sırayla verilmiştir.

$$dsl_{k,i,q,j,t} = \begin{cases} 0, & l_{k,i,t} - 0,75\tau_l < l_{q,j,t} < l_{k,i,t} + 0,75\tau_u \\ 1, & l_{k,i,t} - \tau_l < l_{q,j,t} < l_{k,i,t} + \tau_u \quad (t \in \{1,2,3\}) \\ 7, & \text{aksi takdirde} \end{cases} \quad (4.4)$$

$$d\alpha_{k,i,q,j,t} = \begin{cases} 0, & \alpha_{k,i,t} - 0,6\frac{\pi}{10} < \alpha_{k,i,t} < \alpha_{k,i,t} + 0,6\frac{\pi}{10} \\ 1, & \alpha_{k,i,t} - \frac{\pi}{10} < \alpha_{k,i,t} < \alpha_{k,i,t} + \frac{\pi}{10} \quad (t \in \{1,2,3\}) \\ 7, & \text{aksi takdirde} \end{cases} \quad (4.5)$$

Dolayısıyla  $f_k$ 'nin  $T_{k,i}$  ve  $f_q$ 'nin  $T_{q,j}$  özellik vektörleri arasındaki sözde mesafe  $ds_{k,i,q,j}$  benzemezlik değeri eşitlik (4.6)'daki gibi ifade edilir.

$$d_{k,i,q,j} = \sum_{t=1}^3 dsl_{k,i,q,j,t} + \sum_{t=1}^3 d\alpha_{k,i,q,j,t} \quad (4.6)$$

$T_{k,i}$  ve  $T_{q,j}$ 'nin referans özellik noktaları arasında en çok  $\pi/4$  radyan açı ve 256 piksel yer farkı varsa eşitlik (4.7) ile ifade edilen  $dsm_{k,i,q,j}$  değeri 0 olur. Bu,  $T_{k,i}$  ve  $T_{q,j}$  arasında izin verilen miktarda dönme ve yer farkı olabileceği manasına gelir.

$$dsm_{k,i,q,j} = \begin{cases} 0, & |\theta_{k,i,1} - \theta_{q,j,1}| \leq \frac{\pi}{4} \quad \wedge \quad \|m_{k,i,1} - m_{q,j,1}\| \leq 256 \\ 1, & \text{aksi takdirde} \end{cases} \quad (4.7)$$

Eğer tüm  $l_{k,i,1}, l_{k,i,2}, l_{k,i,3}, \alpha_{k,i,1}, \alpha_{k,i,2}, \alpha_{k,i,3}$ , sırasıyla  $l_{q,j,1}, l_{q,j,2}, l_{q,j,3}, \alpha_{q,j,1}, \alpha_{q,j,2}, \alpha_{q,j,3}$  değerlerine yakınsa ve dönme ile yer değiştirme izin verilen değerleri aşmıyorsa  $f_q$  için  $f_k$ 'nin skoru eşit  $I_{k,i}$  ve  $I_{q,j}$  indeksleri için artırılır. Her  $T_{k,i}$  ve  $T_{q,j}$  çifti için  $s_{k,i,q,j}$  skor artırım ifadesi eşitlik (4.8) ile verilmiştir.

$$s_{k,i,q,j} = \begin{cases} \frac{1}{1 + 0,55 ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \quad \wedge \quad dsm_{k,i,q,j} = 0 \\ 0, & \text{aksi takdirde} \end{cases} \quad (4.8)$$

$f_q$  için  $f_k$ 'nin toplam skoru (4.9) ile hesaplanır.

$$S_{k,q} = \sum_j \sum_i \frac{s_{k,i,q,j}}{n_k} \quad (I_{k,i} = I_{q,j}) \quad (4.9)$$

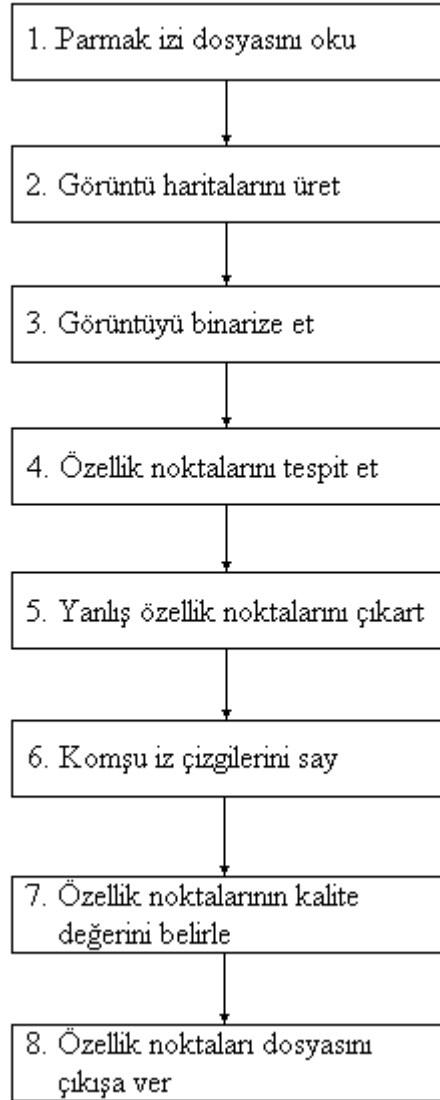
$I_{q,j}$  indeksi  $f_k$ ' dan birden fazla indeksle eşleşebilir. Böyle bir durumda en yakın  $I_{q,j}$  'nin seçilmesi gerekir. Fakat karmaşıklığı arttırmamak için hepsi değerlendirmeye alınmaktadır.

Sonunda parmak izleri puanlarına göre azalan yönde sıralanırlar. En yüksekten başlayarak izin verilen sayıda parmak izi sorgulanan parmak izi için aday listesi olarak verilir.

Yukarıda anlatılan özellik noktaları üçlüsü temelli indeksleme yöntemi değişik veritabanlarında MCC tabanlı indeksleme yöntemiyle karşılaştırmalı olarak uygulanmıştır.

#### 4.1.6. METODUN UYGULANMASI

Metod parmak izi resimlerinin 500 ppi çözünürlükte olduğu varsayımıyla uygulanmıştır. Fakat vektör binleme ile ilgili sabitleri değiştirerek değişik çözünürlükler için de uygulanabilir. Parmak izi görüntüleri işlenmeden önce iyileştirilmemiş, doğrudan özellik noktası tespit aşamasına geçilmiştir. NIST DB4 [30] ve NIST DB14 [31] resimlerine has kenarlardan 50' şer piksel çıkarılmıştır. Çünkü bu resimler parmak izi görüntüsünden daha büyüktür ve kenarlarında yazılar ve çizgiler ihtiva eder. Karşılaştırmanın yapılacağı MCC temelli indeksleme çalışmasında [18] da deneyler öncesinde bu işlem yapılmıştır.



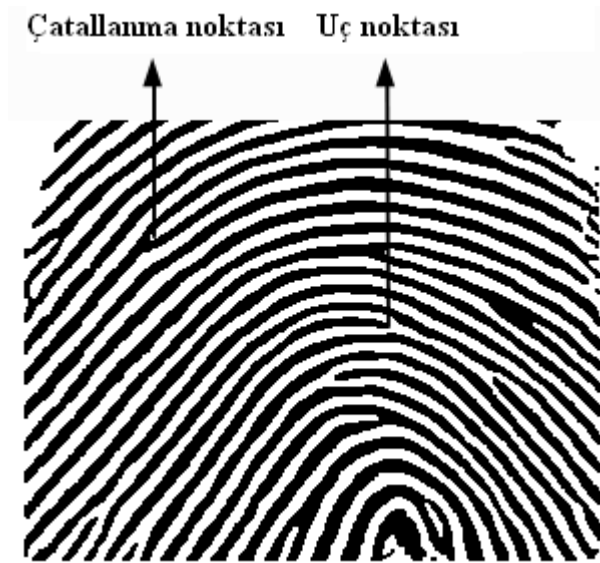
Şekil 4.4. MINDTCT çalışma adımları [32].

Özellik noktalarının tespiti için MINDTCT (Sürüm 4.1.0) kullanılmıştır. MINDTCT [32] NIST (National Institute of Standards and Technology) tarafından dağıtılan nbis [33] (NIST Biometric Image Software) paketindedir. Çalışma adımları Şekil 4.4 ile gösterilmiştir. Çıktı olarak verdiklerinden bu çalışmada özellik noktalarının konum, yön ve güvenilirlik bilgilerinden faydalanılmıştır.

MINDTCT 'in ürettiği diğer çıktılar yön haritası, düşük kontrast haritası, düşük akış haritası, yüksek eğri haritası ve kalite haritasıdır [32]. Ayrıca MINDTCT Şekil 4.4'te belirtilen ara işlemlerden biri olarak binarizasyon da yapar.

Açık kaynak kodlu olan bu programın kodları değiştirilerek parmak izinin binarize edilmiş halini çıktı olarak vermesi sağlanmıştır ve iz deseni temsilini çıkarmak için bu binarize edilmiş çıktı kullanılmıştır. Ayrıca FVC veritabanlarına [26,28] ait parmak izi resimlerini okuyabilmesi sağlanmıştır.

Bu özellik noktası saptama aracı özellik noktalarının konumlarını türleriyle uyum içinde tespit etmektedir. Bütün uç noktalarının konumları siyah renkli uç noktalarının üzerinde olurken, bütün çatallanma noktalarının konumları beyaz renkli bölgeler üzerinde olur. (İz deseni temsilini çıkartmak için hazırlanan algoritma bu özellikten faydalanmaktadır.) Şekil 4.5 bu durumun örneklerini göstermektedir.

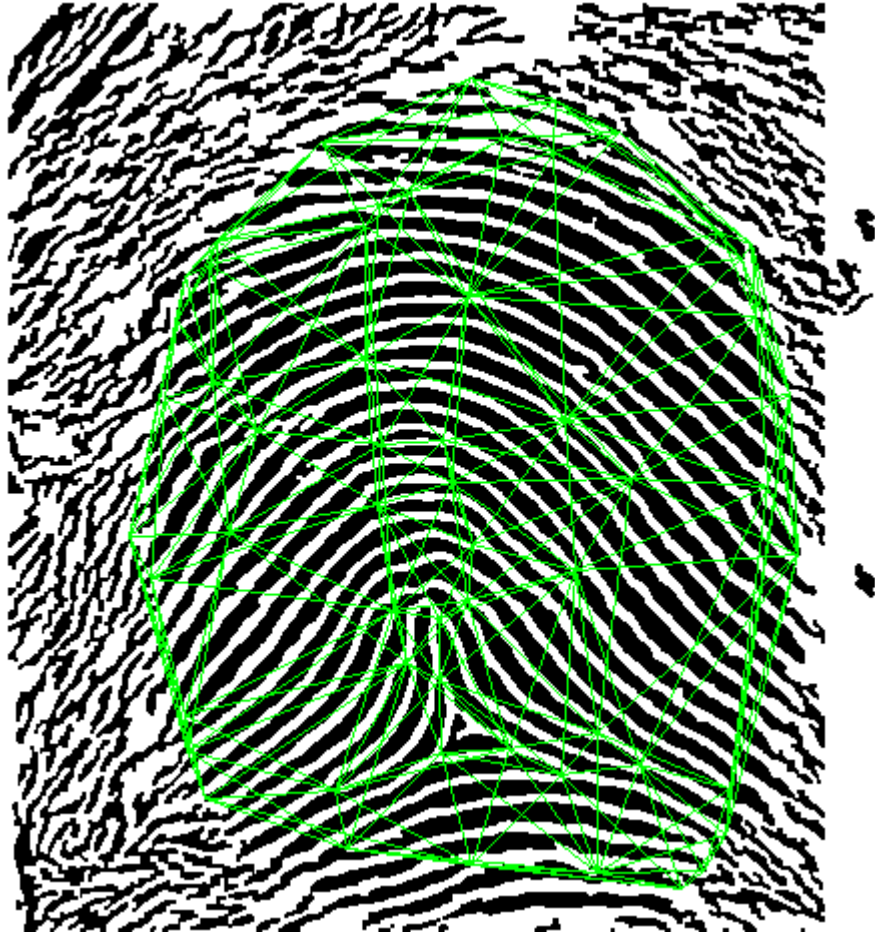


Şekil 4.5. Saptanan iki türden özellik noktasının konumları.

MINDTCT [32] özellik noktalarını 0 - 1 aralığındaki güvenilirlik değerleriyle verir. 0.15' ten küçük güvenilirlik değerlerine sahip özellik noktaları sahte özellik noktalarını çıkarmak için filtrelenmiştir. Çünkü sahte özellik noktaları çıkarılmadığı zaman delaunay üçgen ağı daha farklı oluşur. Şekil 4.6 ile Şekil 4.7 FVC2000 DB3 veritabanındaki [26] 101\_1.tif adlı parmak izi resminin düşük güvenilirlik değerine sahip özellik noktalarının filtrelendiğinde ve filtrelenmediğinde üçgen ağının aldığı şekilleri göstermektedir.



Şekil 4.6. Düşük güvenilirlikli özellik noktaları filtrelenmediğinde üçgen ağı.



Şekil 4.7. Düşük güvenilirlikli özellik noktaları filtrelendiğinde üçgen ağı.

Özellik noktalarının yönleri MINDTCT [32] tarafından pozitif dişey eksenden başlayıp saat yönünde artan 0 - 31 aralığındaki tam sayılarla temsil edilir. Önerilen metod gereğı doğrudan 0 - 7 aralığındaki tam sayılara dönüştürölmektedirler. Özellik noktalarna göre vektör açılarını hesaplayabilmek içinse önce 0 -  $2\pi$  aralığındaki kayan noktalı sayılara dönüştürölmektedirler.

Kendilerine karşılık gelen deęerlerle beraber her indekslenmiş parmak izi resminin indeksi hash tablosu için anahtar deęerleri olarak bir dosyada tutulmuştur. Ait oldukları izlere göre gruplanan sorgulanacak parmak izi resimlerinin indeksleri de kendilerine karşılık gelen deęerlerle beraber başka bir dosyada tutulmuştur. Dosyaların ikincisi yedek indeksleri de ihtiva etmiştir.

İkinci dosya tek seferlik deneyle sorgulatmak için veritabanının sorgulanacak tüm

parmak izlerinin indeks, kimlik ve diğer bilgilerini tutar. Birinci dosya ise veritabanının indekslenmiş parmak izleri yerine geçecek tüm izlerinin indekslerini ve karşılık gelen kimlik bilgisi ile diğer değerlerini tutar.

Metodu denemek için yazılan bir program:

- İki dosyayı yükler. Birinci dosya bilgilerini kullanarak hash tablosunu oluşturur.
- İkinci dosyada indeksleri yer alan her bir parmak izi için
  - İndeksleri indekslenmiş parmak izlerinin indekslerine karşılık gelen kimlik bilgisi ve ilgili değerleriyle doldurulmuş hash tablosunda arar.
  - İndekslenmiş parmak izlerinin skorlarını hesaplar ve en yüksek skorlulardan izin verilen sayıda parmak izini skorlarına göre azalan yönde aday parmak izleri listesi olarak verir.

MCC temelli yazılım [18] konulduğu web sistesinden [34] indirilmiştir. İki metodu karşılaştırmak için MINDTCT [32] çıktısı yine güvenilirlik değerlerine göre filtrelenerek MCC temelli indeksleme yazılımına kendi gerektirdiği biçimde verilmiştir. Bu yapılırken özellik noktası yönleri pozitif yatay eksenden başlayarak saat yönünün tersinde artan  $0 - 2\pi$  aralığındaki değerlere dönüştürülmüştür.

Metodu test etmek için FVC2000 DB2 [26], FVC2000 DB3 [26], FVC2002 DB1 [28], NIST DB4 [30] ve NIST DB14 [31] ile deneyler yapılmıştır.

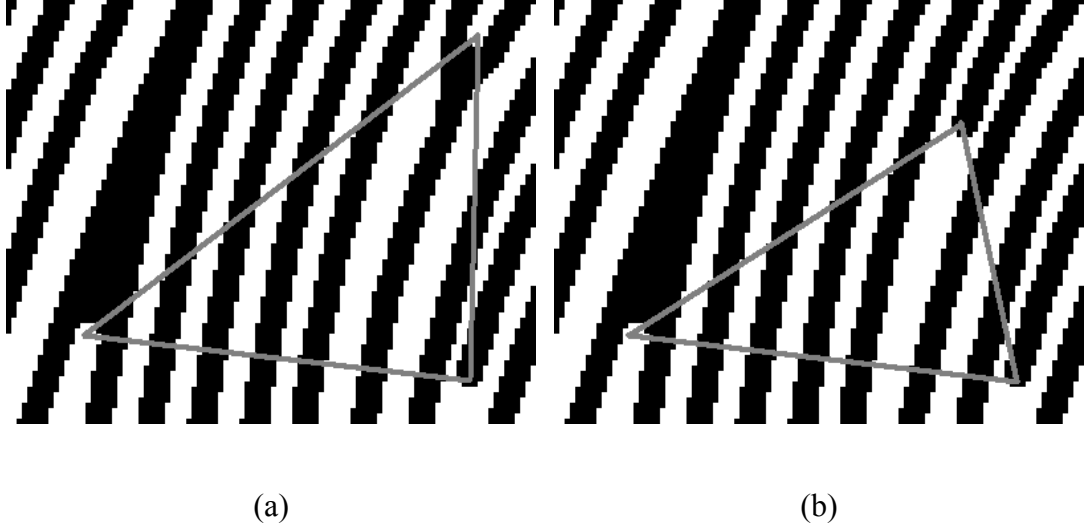
Adı verilen FVC veritabanlarının hepsinde 100' er parmağa ait 1' den 8' e kadar numaralanmış 8 farklı parmak izi bulunur. Her parmağın 1 numaralı izi indekslenmiş ve geri kalan 7 izden rastgele seçilen biri sorgu izi olarak kullanılmıştır. Sorgu parmak izinin numarası C yalancı rastgele sayı fonksiyonu rand tarafından 0 tohum değeri verilerek elde edilmiş ve her sorgu için tekrar rastgele sayı üretilmiştir.

NIST DB4 [30] 2000 çift parmak izi içerir. Her çift için birincisi indekslenmiştir, ikincisi sorgu izi olarak kullanılmıştır. NIST DB14 [31] numaralanmış 27000 çift parmak izi içerir. Son 2700 çiftin ikincileri sorgu izi olarak kullanılmıştır. Önce son 2700, sonra son 24000 çiftin birincileri indekslenip üzerlerinde aramalar yapılmıştır.



## 4.2. İZ DESENİ TEMSİLİ İLE ÖZELLİK NOKTALARI ÜÇLÜSÜ TEMELLİ İNDEKSLEME

Özellik noktası türü sağlam bir özellik değildir [27] ve özellik noktası türünün yanlış tespit edilmesi özellik noktaları arasındaki iz çizgilerinin yanlış sayılmasına sebep olur. Şekil 4.8 bunun örneğini göstermektedir. Benzer şekilde 3.4'te anlatıldığı gibi bu durum üçgensel bölgenin içinde kalan iz deseninin ve iz deseni temsilinin değişmesine de sebep olur. Ayrıca iz çizgilerinden birinde olabilecek kopukluk sahte özellik noktalarının oluşmasına sebep olabilir.



Şekil 4.8. Özellik noktalarının türünün yanlış tespit edilmesiyle özellik noktaları arasında kalan iz çizgilerinin sayılarının değişmesi, a) Alt kenardan başlayarak iz çizgileri sayıları 7-2-6, b) Alt kenardan başlayarak iz çizgileri sayıları 7-2-5.

MCC [18,27] yapısına özellik noktası türü dahil edilmemiştir. [11]'de ise iz çizgileri sayısının kullanılmasından kaçınılmıştır ama özellik noktası türü kullanılmıştır. Aslında yukarıda anlatıldığı gibi bu üç özelliğin sağlamlığı birbiriyle ilgilidir. Yine de özellik noktası üçlülerini kullanan metodların çoğunda bu özelliklere ihtiyaç duyulmuştur. Örneğin iz çizgisi sayıları [10,19-20]'de, özellik noktası türleri [11,16]'de indeksler oluşturulurken kullanılmıştır. Özellik noktası türü ya da iz çizgisi sayısı kullanmayan özellik noktası üçlülerini temelli indeksleme çalışmalarının örnekleri ise [12,14]'tür.

Bu çalışmada iz deseni temsili [35] ile özellik noktası üçlülere temelli indeksleme metodunun başarısının artırılması önerilmiştir. İz deseni temsili indekslemeye katılmamış indeksleme sonucunda elde edilen sonuç kümesinin eleman sayısının azaltılması için kullanılmıştır.

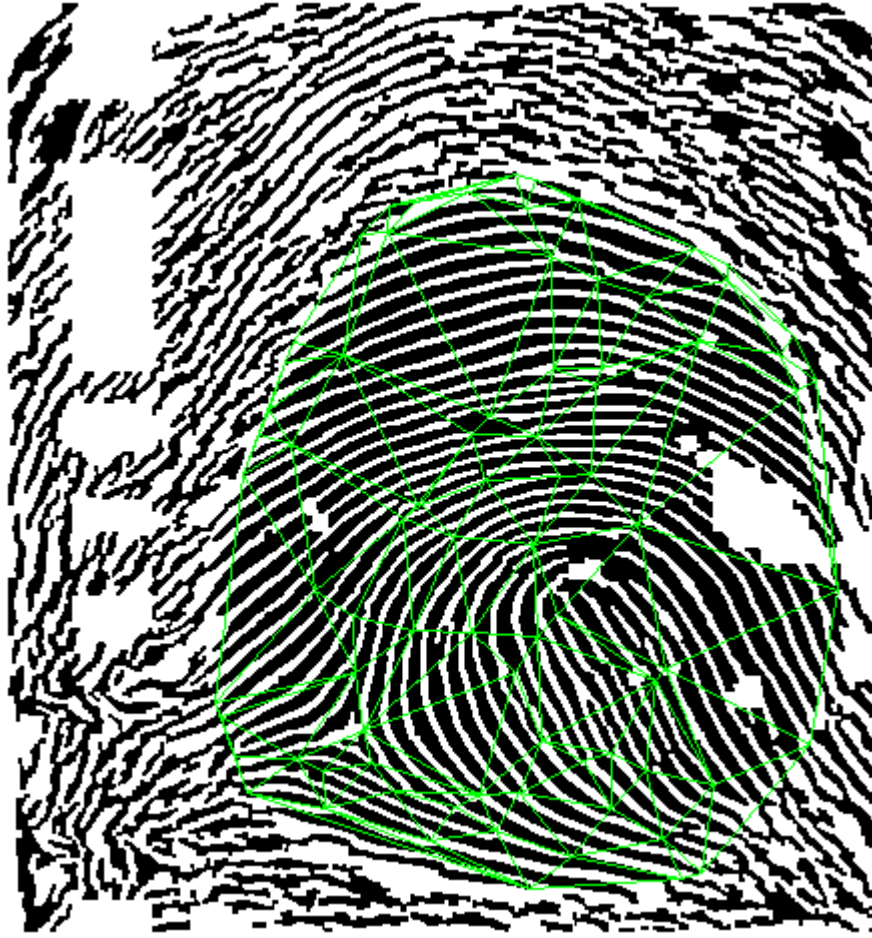
Eşitlik (4.1)'de tanımlanan  $T_{k,i}$  özellik noktaları ile çevrilmiş üçgenel alanın iz deseni temsili  $P_{k,i}$  olsun. İz deseni temsili kullanıldığında indekslenen parmak izi  $f_k$ 'nin her  $I_{k,i}$  indeksine hash tablosunda karşılık gelen girdide  $k, l_{k,i,1}, l_{k,i,2}, l_{k,i,3}, \alpha_{k,i,1}, \alpha_{k,i,2}, \alpha_{k,i,3}, m_{k,i,1}$ 'in konum ve açı bilgileri ile beraber  $P_{k,i}$  tutulur.

Karşılaştırılan üçgenel bölgeler için iz desenlerinin eşit olmasını kriter olarak koymak için eşitlik (4.8)'de verilen ifadeye  $P_{k,i} = P_{q,j}$  kriteri eklenerek skor artırım ifadesi eşitlik (4.10)'daki hale getirilebilir.

$$s_{k,i,q,j} = \begin{cases} \frac{1}{1 + 0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \quad \wedge \quad dsm_{k,i,q,j} = 0 \quad \wedge \quad P_{k,i} = P_{q,j} \\ 0, & \text{aksi takdirde} \end{cases} \quad (4.10)$$

Fakat yapılan deneyler  $P_{k,i} = P_{q,j}$  kriterinin (4.10)'daki haliyle eklenmesinin tutma oranlarını azalttığını göstermiştir. Aksi halde iz deseni temsiline indekslemeye katılması düşünülebilirdi. O zaman hem sonuç kümesi olarak daha az sayıda aday parmak izi yüksek doğruluk oranlarıyla verilir hem de indekslemenin etkisiyle sorgu cevapları daha çabuk elde edilebilirdi.

Eşitlik (4.10)'daki ifade karşılaştırılan özellik noktaları arasındaki iz desenlerinin de aynı olmasını şart koşar ki eşleşmesi gereken iz desenlerinin örnek olarak verilen FVC2000 DB3 veritabanındaki [26] 110\_2.tif adlı parmak izi resmi için Şekil 4.9'deki gibi kayıp alanları olduğu zaman iz çizgileri doğru izlenemeyeceğinden iz deseni temsilleri eşit çıkmaz ve eşleşme olmaz.



Şekil 4.9. Kayıp alanlar barındıran üçgensel iz desenleri.

Bunun için eşitlik (4.8)'deki ifade eşitlik (4.11)'de verilen biçime dönüştürülmüştür.

$$s_{k,i,q,j} = \begin{cases} \frac{1}{1+0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \wedge dsm_{k,i,q,j} = 0 \wedge P_{k,i} \neq P_{q,j} \\ \frac{2}{1+0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \wedge dsm_{k,i,q,j} = 0 \wedge P_{k,i} = P_{q,j} \\ 0, & \text{aksi takdirde} \end{cases} \quad (4.11)$$

Eşitlik (4.11)'deki ifade, sorgulanan  $f_q$  parmak izindeki  $T_j$  özellik noktası üçlüsü ile veritabanında indekslenmiş  $f_k$  parmak izindeki  $T_i$  özellik noktası üçlülerinin iz deseni temsilleri eşit olduğunda, skor arttırım değerinin iki katı yapıldığı manasına gelir. İz deseni temsilleri eşit olmadığında skor arttırım değeri sıfır yapılmaz. İz deseni temsilleri eşit olmayan üçgensel bölgeleri oluşturan özellik noktaları üçlülerinin

diğer geometrik özelliklerine bakılarak aday parmak izinin skor değeri üzerinde katsayısı düşük de olsa katkı yapma şansı elde etmesini sağlar.

Skor arttırım değeri eşitlik (4.8) yerine eşitlik (4.11)'deki gibi iz desenlerinin eşit olup olmamasına göre, eşitlik (4.12)'deki gibi özellik noktası türlerinin eşit olup olmamasına göre ve eşitlik (4.13)'teki gibi iz çizgisi sayılarının eşit olup olmamasına göre hesaplanarak eşitlik eşitlik (4.8)'in geçerli olduğu 4.1'de anlatılan metotla karşılaştırmalı olarak deneyler yapılmıştır.

$$s_{k,i,q,j} = \begin{cases} \frac{1}{1+0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \wedge dsm_{k,i,q,j} = 0 \wedge M_{k,i} \neq M_{q,j} \\ \frac{2}{1+0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \wedge dsm_{k,i,q,j} = 0 \wedge M_{k,i} = M_{q,j} \\ 0, & \text{aksi takdirde} \end{cases} \quad (4.12)$$

Eşitlik (4.12)'deki  $M_{k,i}$  ve  $M_{q,j}$  ise sırasıyla benzerlik testi yapılan  $T_{k,i}$  ve  $T_{q,j}$  özellik noktası üçlüleri için  $\{0,1\}$  kümesinin elemanlarıyla kodlanmış özellik noktası türü üçlüleridir.

$$s_{k,i,q,j} = \begin{cases} \frac{1}{1+0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \wedge dsm_{k,i,q,j} = 0 \wedge R_{k,i} \neq R_{q,j} \\ \frac{2}{1+0,55ds_{k,i,q,j}}, & ds_{k,i,q,j} < 7 \wedge dsm_{k,i,q,j} = 0 \wedge R_{k,i} = R_{q,j} \\ 0, & \text{aksi takdirde} \end{cases} \quad (4.13)$$

Eşitlik (4.13)'teki  $R_{k,i}$  ve  $R_{q,j}$  sırasıyla benzerlik testi yapılan  $T_{k,i}$  ve  $T_{q,j}$  özellik noktası üçlüleri için özellik noktaları arasında kalan iz çizgisi sayısı üçlüleridir.

## BÖLÜM 5

### DENEY SONUÇLARI

MCC temelli indeksleme [18] ile bu çalışma kapsamında uygulanan özellik noktaları üçlüsü temelli indeksleme karşılaştırmalı olarak denenmiştir. Ardından uygulanan özellik noktaları üçlüsü temelli indeksleme sonuçları iz deseni temsili ile daha başarılı hale getirilmiştir. Son olarak iz deseni temsilinin kullanılmasının etkisiyle özellik noktası türünün ve iz çizgileri sayısının kullanılmasının etkisini karşılaştırmak için deneyler gerçekleştirilmiş penetrasyon ve tutma oranları elde edilmiştir.

Penetrasyon, aday listesi olarak elde edilen parmak izi sayısının veritabanındaki tüm parmak izlerinin sayısına oranı manasına gelir. Tutma oranı (hit rate) ise başarılı deneme sayısının tüm denemelerin sayısına oranıdır.

#### 5.1. MCC TEMELLİ VE UYGULANAN ÖZELLİK NOKTALARI ÜÇLÜSÜ TEMELLİ İNDEKSLEMELERİN KARŞILAŞTIRILMASI

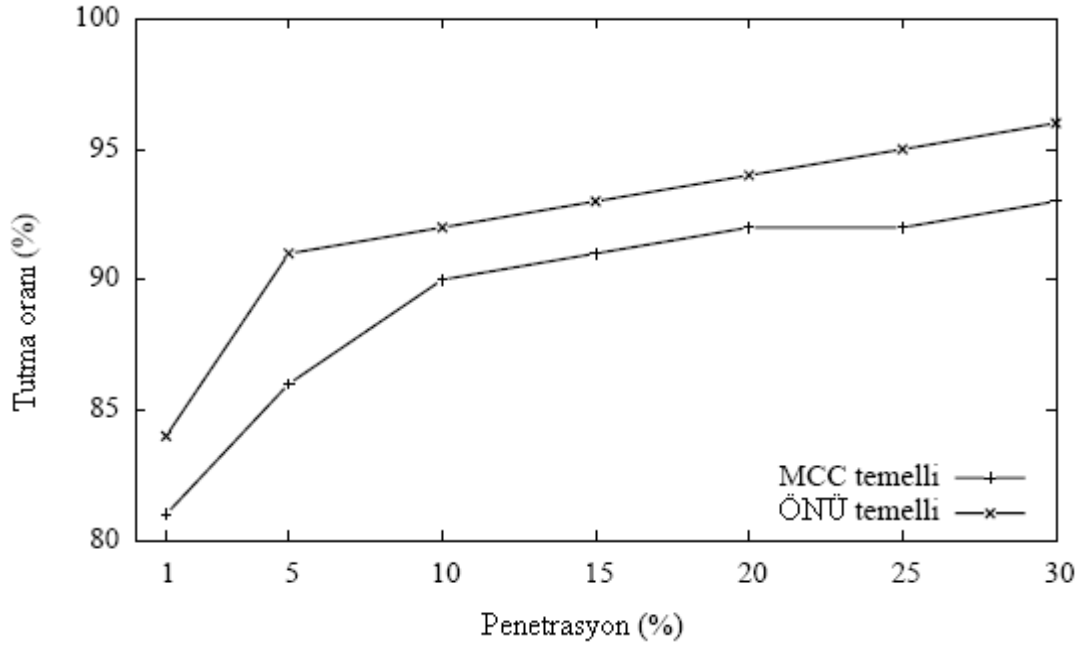
Uygulanan özellik noktaları üçlüsü temelli yazılım ve bulunduğu web sitesinden [34] indirilen MCC temelli indeksleme [18] yazılımı veritabanlarında test edilmiştir.

Özellik noktaları üçlüsü temelli indekslemede yedek indeks miktarı katsayısı  $c$ , 0,4 olarak atanmıştır. En yüksek tutma oranlarını elde edebilmek için  $c$  değeri şöyle belirlenmiştir: Her bir veritabanında elde edilen ve Çizelge 5.1'de verilen tutma oranlarının 0,01 ve 0,05 penetrasyon oranına karşılık gelenlerinden en yüksek olanlarını elde etmeye yarayan, en küçük ve birbirini izleyen iki farklı  $c$  değeri tespit edilmiştir. Bu değerler FVC2000DB2 için 0,40 ve 0,45, FVC2000DB3 için 0,20 ve 0,25, FVC2002DB1 için 0,35 ve 0,40, FVC2004DB1 için 0,55 ve 0,60'tır. Tespit edilen  $c$  değerlerinin ortalaması alınarak 0,4 değeri bulunmuş ve tüm

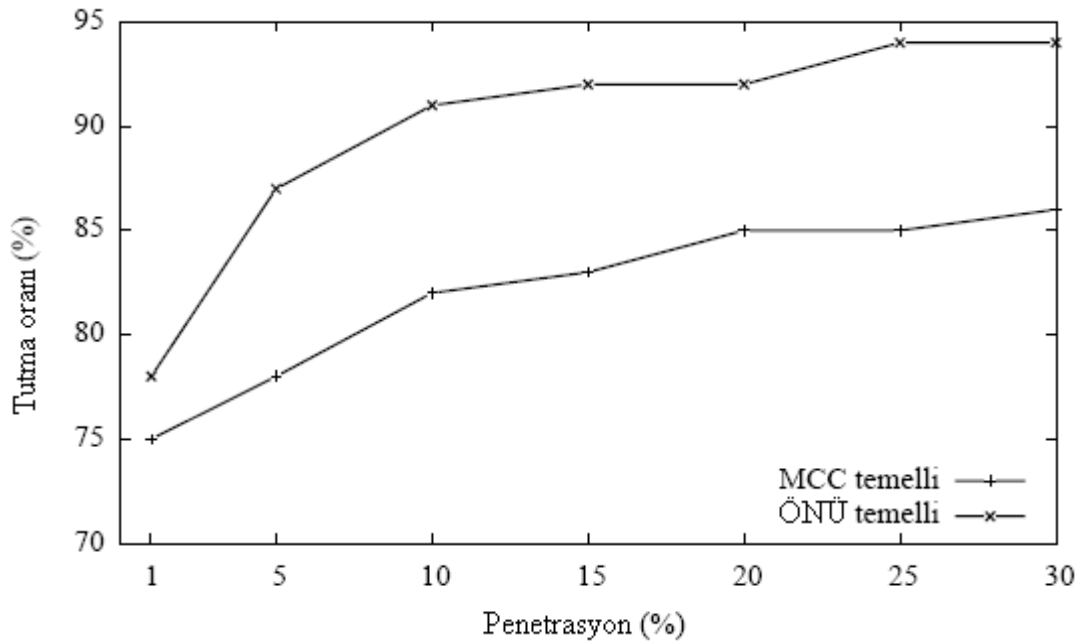
veritabanlarındaki deneylerde kullanılmıştır. Elde edilen sonuçlara göre uygulanan özellik noktaları üçlüsü temelli indeksleme MCC temelli indekslemeye [18] göre denenen tüm veritabanlarında üstün başarı göstermiştir. Sonuçlar Şekil 5.1'den Şekil 5.6'ya kadar gösterilmiştir.

Çizelge 5.1. Farklı  $c$  değerleri için tutma oranları.

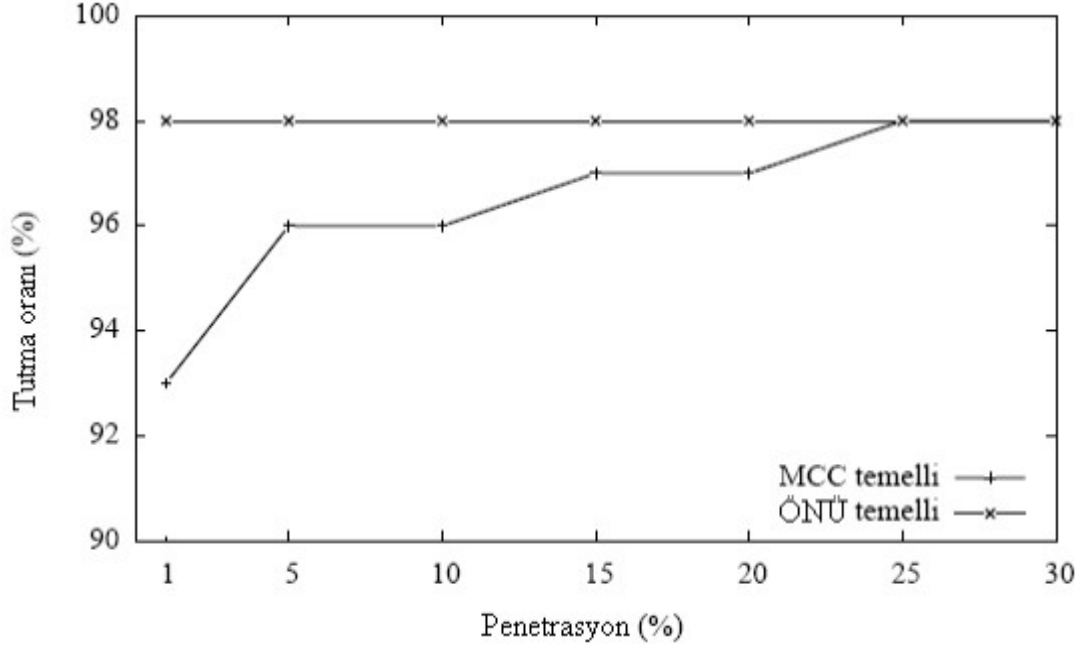
	Tutma oranı	Yedek indeks miktarı katsayısı, $c$																			
		0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00
FVC2000 DB2	1	83	81	82	83	83	82	83	<b>84</b>	<b>84</b>	85	84	85	85	84	83	83	83	83	83	83
	5	89	89	89	89	89	89	90	<b>91</b>	<b>91</b>	89	89	89	89	89	89	89	89	89	89	89
	10	90	90	90	90	92	93	93	<b>92</b>	<b>92</b>	92	92	92	92	92	92	92	92	92	92	92
	15	91	90	91	91	92	93	93	<b>93</b>	<b>93</b>	93	93	93	94	94	94	94	94	94	94	94
	20	94	93	92	92	92	93	93	<b>94</b>	<b>94</b>	94	94	93	94	94	94	94	94	94	94	94
	25	95	95	95	96	95	95	94	<b>95</b>	<b>94</b>	94	94	94	94	94	94	94	94	94	95	95
FVC2000 DB3	1	76	77	78	<b>80</b>	<b>79</b>	80	78	78	77	78	77	76	76	78	77	77	77	77	78	78
	5	83	84	86	<b>87</b>	<b>87</b>	87	88	87	86	87	88	87	87	87	86	86	85	85	85	85
	10	88	88	88	<b>89</b>	<b>89</b>	89	89	91	90	90	90	90	90	90	90	90	90	90	91	92
	15	90	89	90	<b>90</b>	<b>90</b>	90	92	92	92	93	93	92	91	92	92	93	91	91	91	92
	20	91	92	92	<b>93</b>	<b>93</b>	92	92	92	92	94	94	94	95	95	95	95	95	94	93	92
	25	92	93	94	<b>93</b>	<b>94</b>	94	94	94	93	94	94	94	95	96	96	96	96	96	96	96
FVC2002 DB1	1	96	97	97	97	97	97	<b>98</b>	<b>98</b>	98	98	98	98	98	98	98	98	98	98	98	98
	5	97	97	97	97	98	98	<b>98</b>	<b>98</b>	98	98	98	98	98	98	98	98	98	98	98	98
	10	97	97	97	97	98	98	<b>98</b>	<b>98</b>	98	98	98	98	98	98	98	98	98	98	98	98
	15	97	97	97	97	98	98	<b>98</b>	<b>98</b>	98	98	98	98	98	98	98	98	98	98	98	98
	20	98	97	97	97	98	98	<b>98</b>	<b>98</b>	98	98	98	98	98	98	98	98	98	98	98	98
	25	98	98	98	97	98	98	<b>98</b>	<b>98</b>	98	98	98	98	98	98	98	98	98	98	98	98
FVC2004 DB1	1	65	67	70	71	72	73	74	74	75	76	<b>78</b>	<b>78</b>	78	78	78	78	78	78	78	78
	5	79	79	79	79	80	80	82	81	83	83	<b>83</b>	<b>83</b>	83	82	82	82	82	83	83	83
	10	81	85	83	81	83	84	86	86	86	86	<b>86</b>	<b>86</b>	86	86	86	86	86	86	86	86
	15	83	85	85	85	87	87	86	86	86	86	<b>86</b>	<b>86</b>	86	86	86	86	86	86	86	86
	20	84	86	86	86	87	87	88	88	88	88	<b>87</b>	<b>87</b>	87	87	87	87	87	87	87	87
	25	86	88	88	88	88	87	89	89	89	88	<b>88</b>	<b>88</b>	88	88	88	88	88	88	89	89



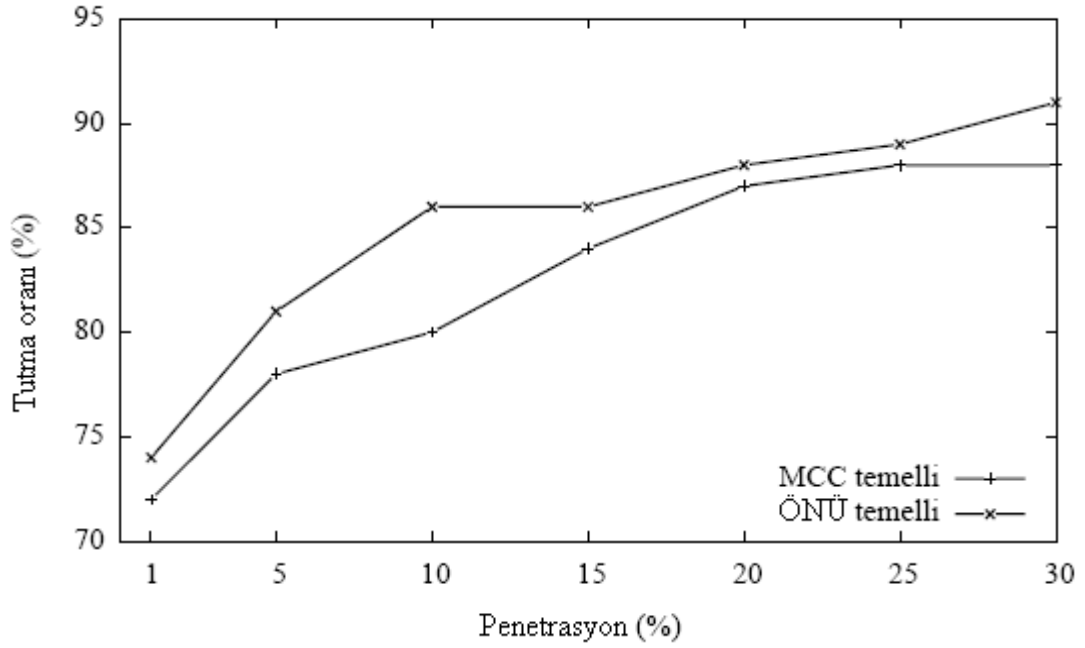
Şekil 5.1. FVC2000DB2 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indekisleme performansları.



Şekil 5.2. FVC2000DB3 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indekisleme performansları.



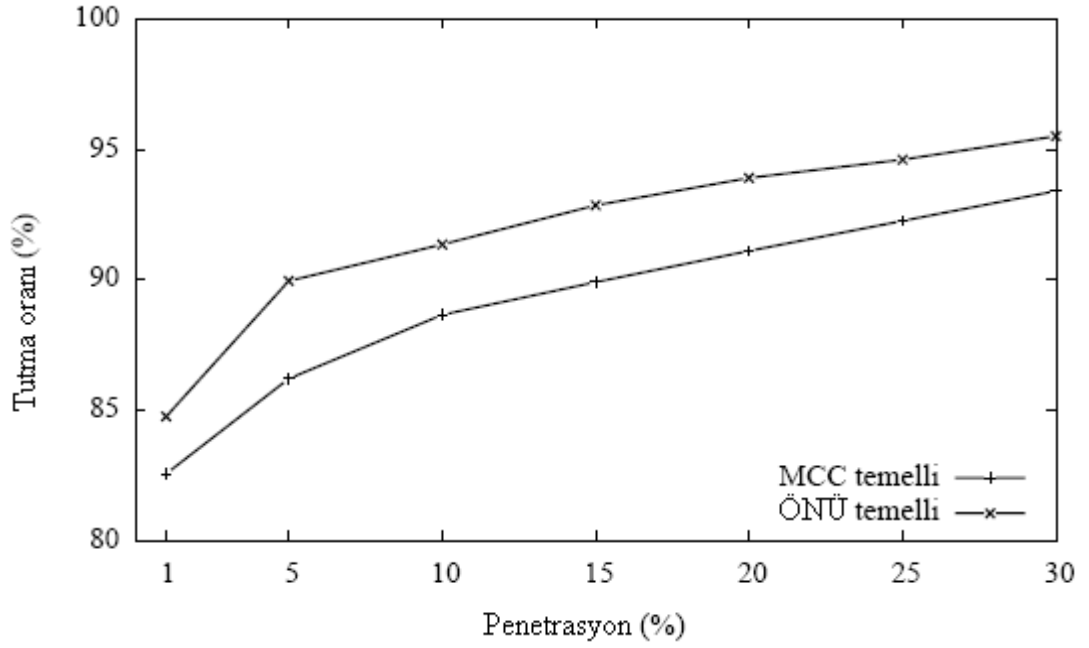
Şekil 5.3. FVC2002DB1 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indekisleme performansları.



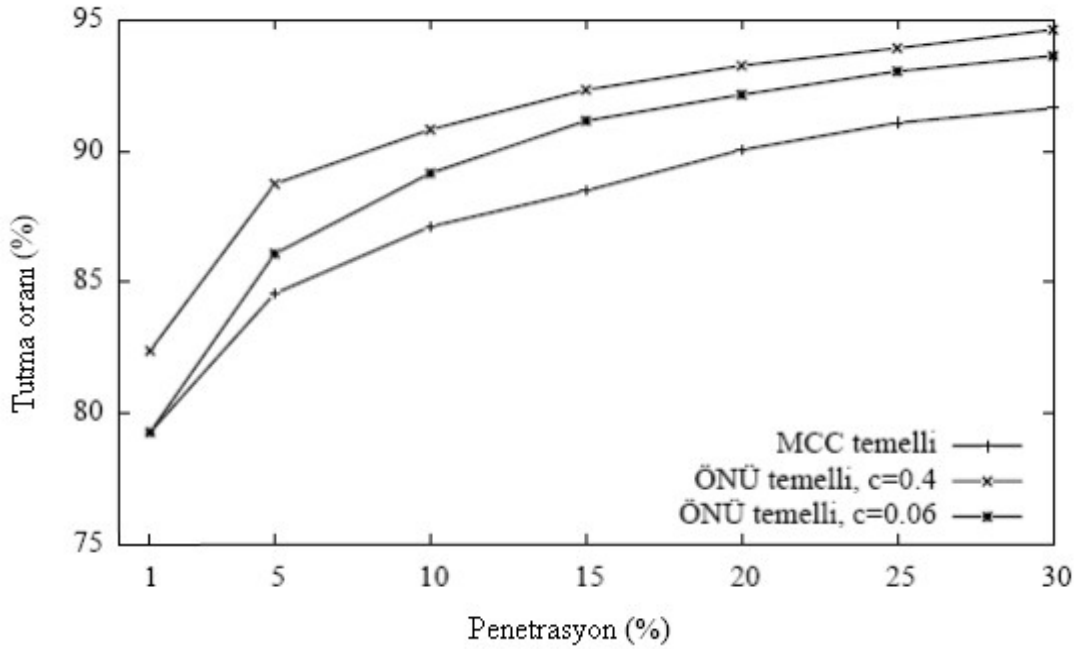
Şekil 5.4. FVC2004DB1 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indekisleme performansları.

FVC2002DB1 için 0,01 penetrasyon oranı için bile özellik noktaları üçlüsü temelli yöntem ile en yüksek tutma oranı sağlandığı için tutma oranı penetrasyon oranına göre değişiklik göstermemiştir.





Şekil 5.5. NISTDB4 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indekisleme performansları.



Şekil 5.6. NISTDB14 için MCC temelli ve özellik noktaları üçlüsü temelli yöntemlerin indekisleme performansları.

NISTDB14 için son 2400 izde son 2700 iz araştırılarak yapılan deneyde neredeyse aynı tutma oranları elde edilmiştir. O yüzden sonuçlara burada yer verilmemiştir.

Zaman ölçümleri MCC temelli indekisleme için yazarlarının [18] yaptığı gibi

NISTDB14'ün son 2700 iz çiftinin ve son 24000 iz çiftinin birinci izleri üzerinde son 2700 iz çiftinin ikinci izleri aratılarak yapılmıştır. Ölçüm sonuçları Çizelge 5.2'de özetlenmiştir.

İki veri seti üzerinde yapılan zaman ölçümlerine göre uygulanan metodun MCC temelli indeksleme [18] ile benzer kompleksiteye sahip olduğu görülmüştür.

Çizelge 5.2. Parmak izi başına ortalama arama süresi.

Veri seti	Yöntem		
	MCC temelli	Özellik noktaları üçlüsü temelli	
		c=0,06	c=0,4
Son 2700 çiftin ilk izleri	10 ms.	10 ms.	19 ms.
Son 2400 çiftin ilk izleri	101 ms.	118 ms.	215 ms.

MCC temelli indekslemeye [18] göre daha yüksek tutma oranları veren  $c=0,4$  değerinden başka  $c=0,06$  değeri ile deneylerin tekrar edilmesi Şekil 5.6'dan da görüldüğü gibi  $c$ 'nin bu değeri için 0,01 penetrasyon oranına karşılık MCC temelli indeksleme [18] ile aynı tutma oranını vermesindedir; deklare edilen süreyle aynı zaman aralığında sonuçları vermesinden değildir.

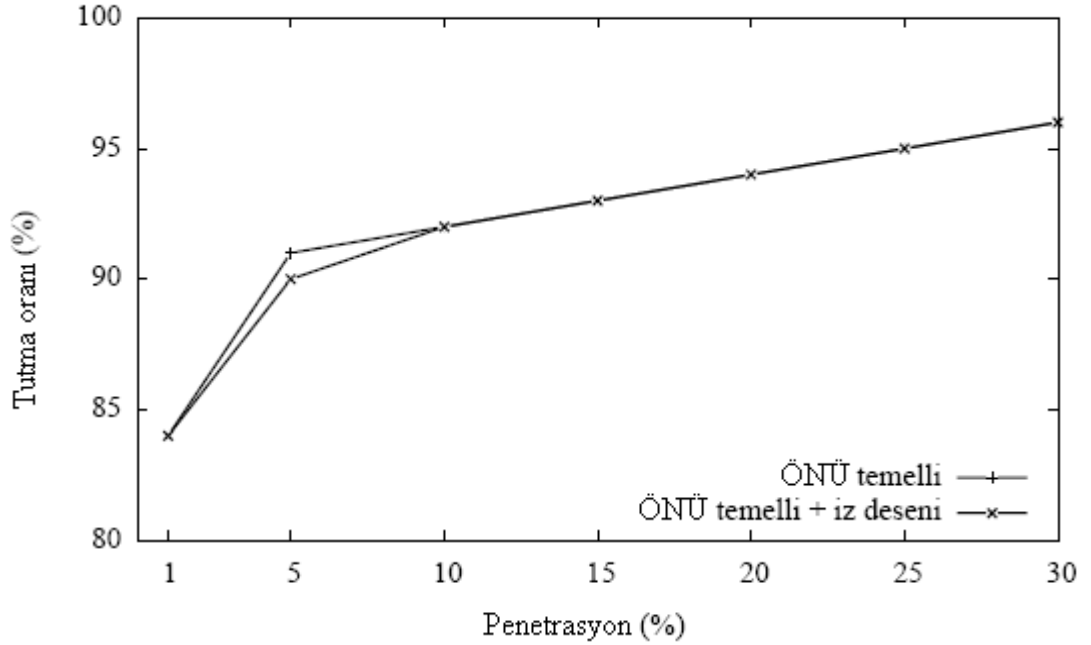
Zaman ölçümü sonuçlarına göre hız konusunda adil kıyas yapmak zordur. Çünkü iki metodun denendiği makineler aynı değildir. İndirilen yazılım arama işlemlerinden başka hem aranacak hem de normalde veritabanında zaten bulunan parmak izlerine ait özellik noktalarını yüklediğinden ve bunları MCC temsillerine dönüştürdüğünden sadece arama işlemleri için geçen süreyi aynı makinede ölçerek bulma imkanı yoktur. MCC temelli yazılım için yapılan zaman ölçümünün 2.66 GHz.'lik Intel Quad Core işlemcili makinede yapıldığı ve yazılımın çok işçikli (multi-threaded) olduğu bildirilmiştir [18]. Bu çalışmanın deneyleri ise Intel Pentium 3.4 GHz.'lik Core 2 i3 işlemcili makinede yapılmıştır. İşlemlerin paralelleştirilmesi ile ilgili bir çalışma yapılmamıştır. Bu sebeple iki metodun zaman ölçümlerinin bire bir karşılaştırılmasından kaçınılmıştır.

## 5.2. İZ DESENİ TEMSİLİNİN ETKİSİ

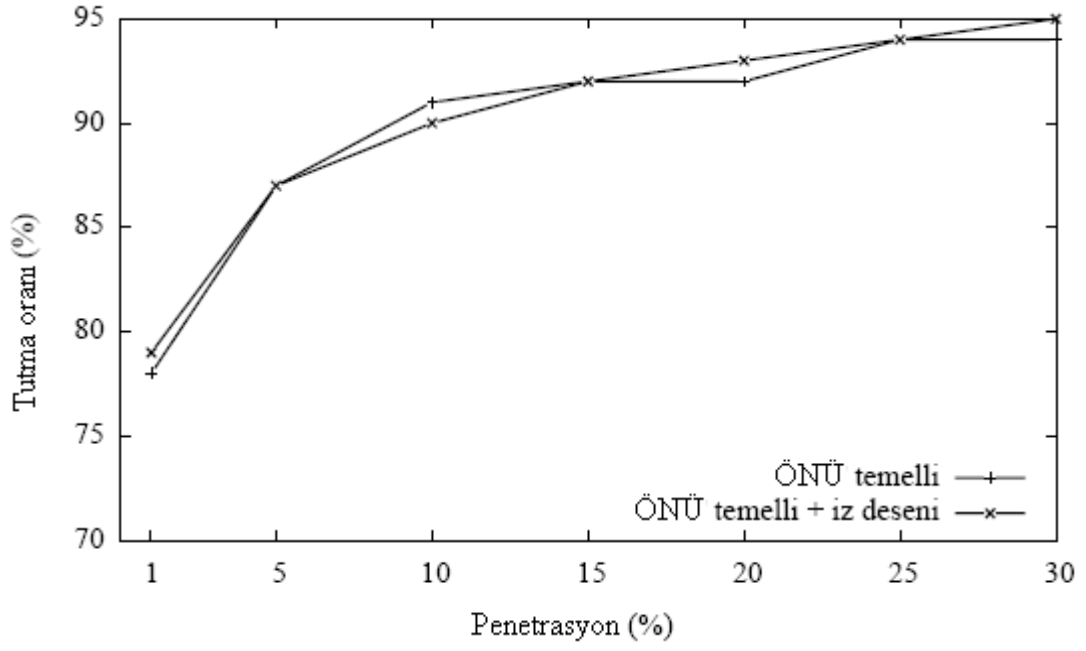
Uygulanan özellik noktaları üçlüsü temelli indekslemenin iz deseni temsili de kullanıldığında özellikle düşük penetrasyon oranları için genellikle daha yüksek tutma oranları sağladığı görülmüştür. Sadece FVC2000DB2 için penetrasyon oranı 0.01 iken ve FVC2002DB1 için her durumda iken tutma oranı aynı kalmıştır. Daha yüksek penetrasyon oranlarında iz deseni temsili kullanıldığında tutma oranı nadiren daha düşük çıkmıştır. Fakat indeksleme sistemi için düşük penetrasyon oranlarında yüksek tutma oranı sağlayabilmek daha önemlidir.

İz deseni temsilinin (4.11)'deki gibi kriter olarak eklenmesi karşılaştırılan özellik noktaları üçlüleri başına fazladan bir eşitlik testinin daha yapılmasını gerektirir. Ortalama arama süresinin ne kadar uzadığını öğrenmek için yapılan deneyde NISTDB14'ün son 2700 iz çifti üzerindeki aramalarda geçen ortalama sürenin 19 ms.'den 21 ms.'ye çıktığı belirlenmiştir.

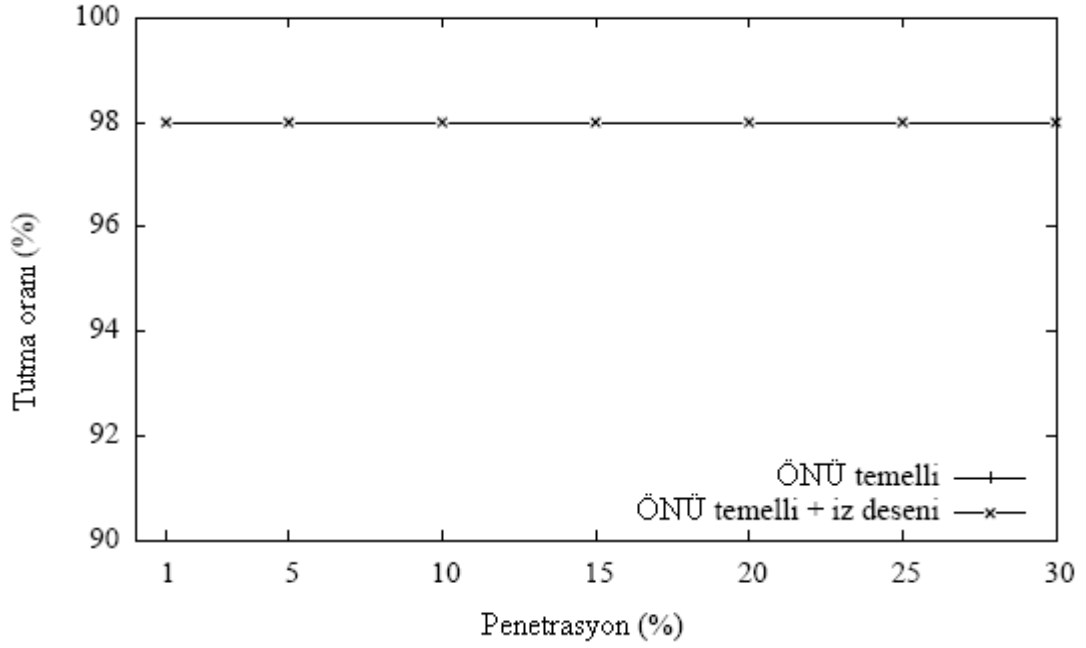
Tutma oranlarının verildiği sonuçlar ise Şekil 5.7'den Şekil 5.12'ye kadar verilmiştir.



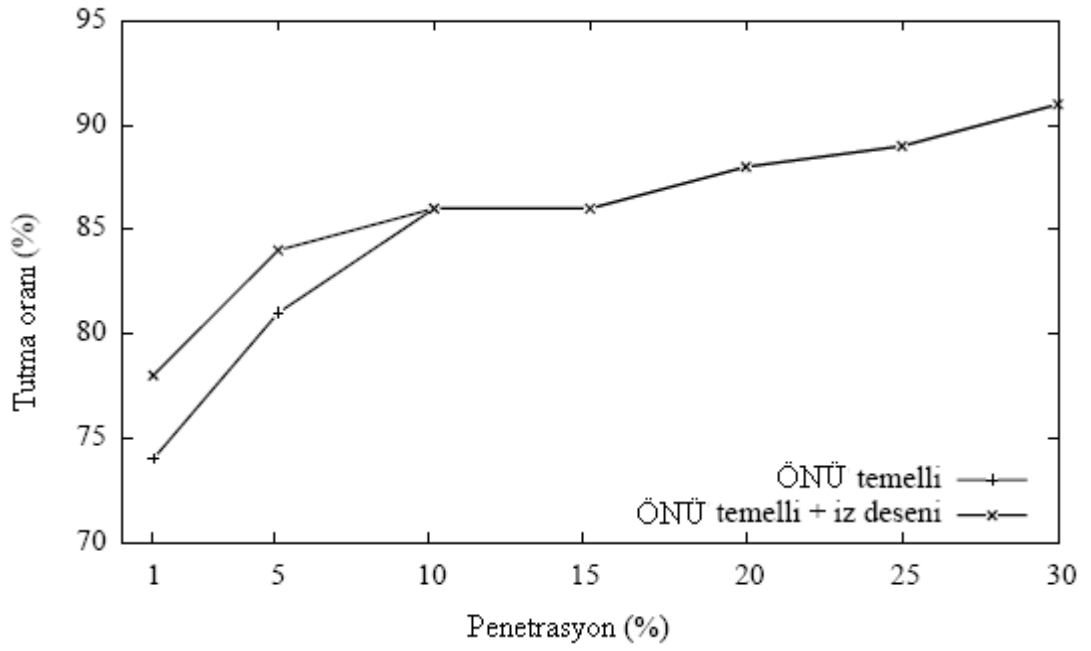
Şekil 5.7. FVC2000DB2 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.



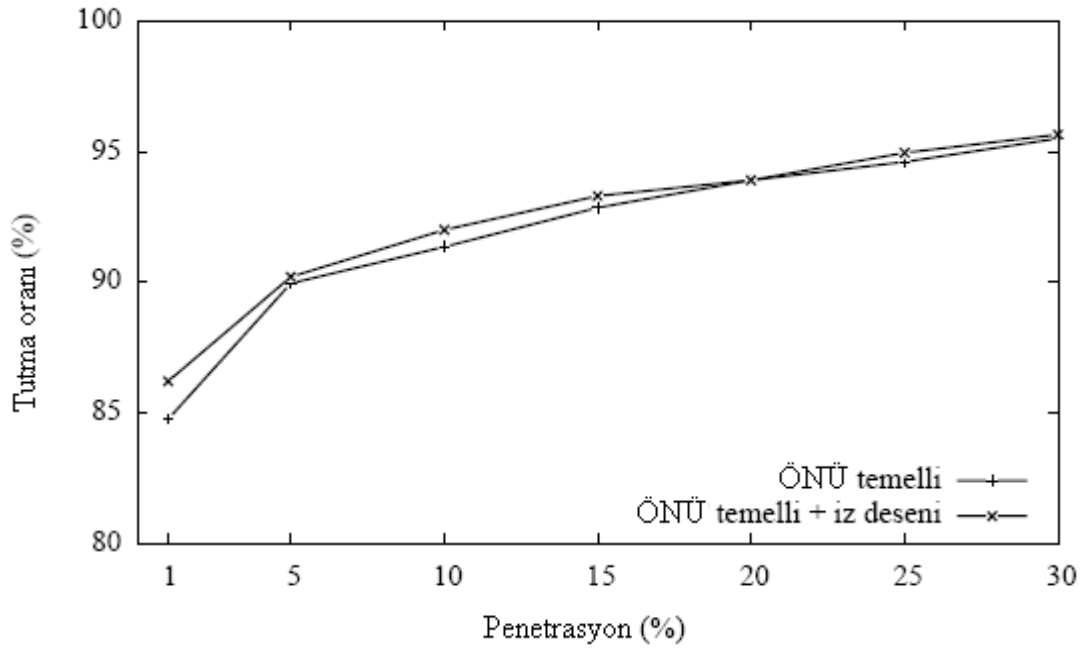
Şekil 5.8. FVC2000DB3 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.



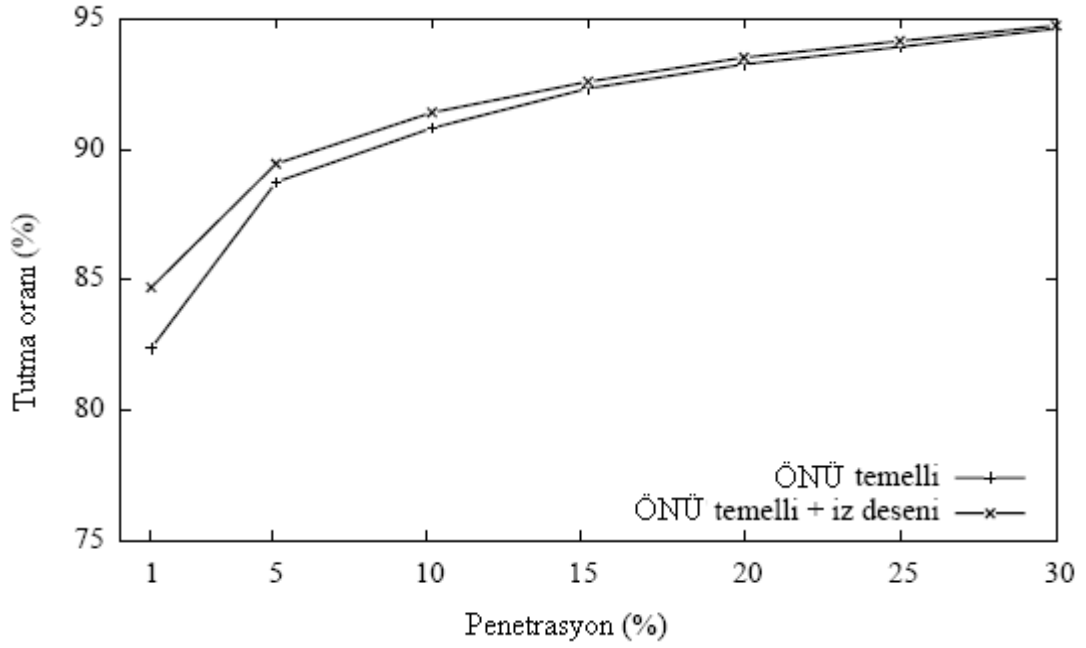
Şekil 5.9. FVC2002DB1 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.



Şekil 5.10. FVC2004DB1 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.



Şekil 5.11. NISTDB4 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.

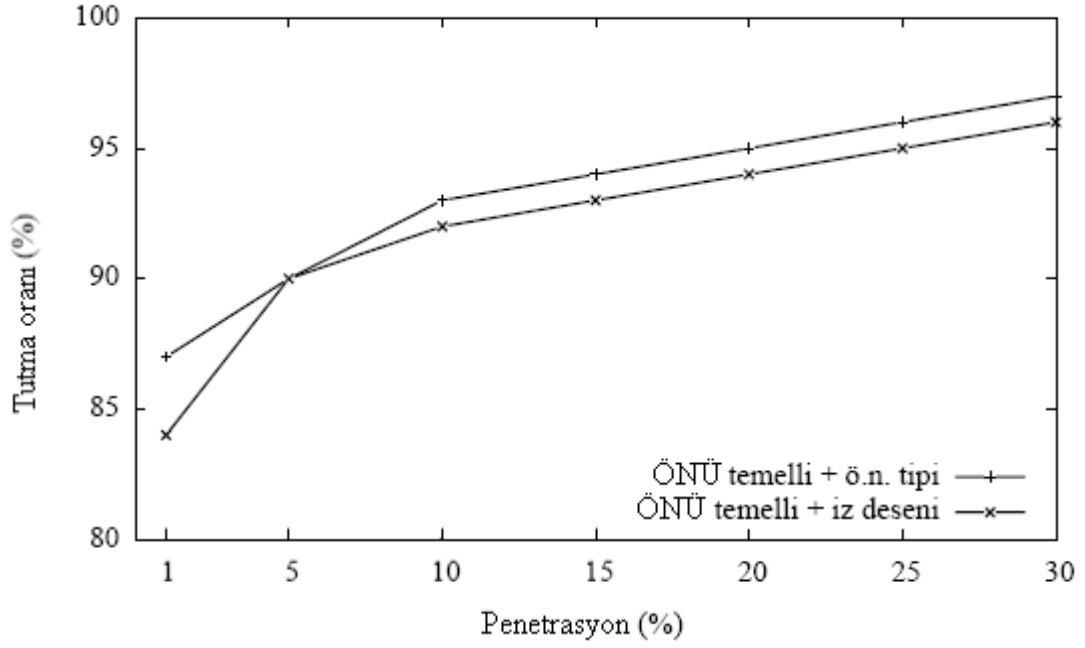


Şekil 5.12. NISTDB14 için iz deseni temsili kullanıldığında ve kullanılmadığında özellik noktaları üçlüsü temelli yöntemlerin indeksleme performansları.

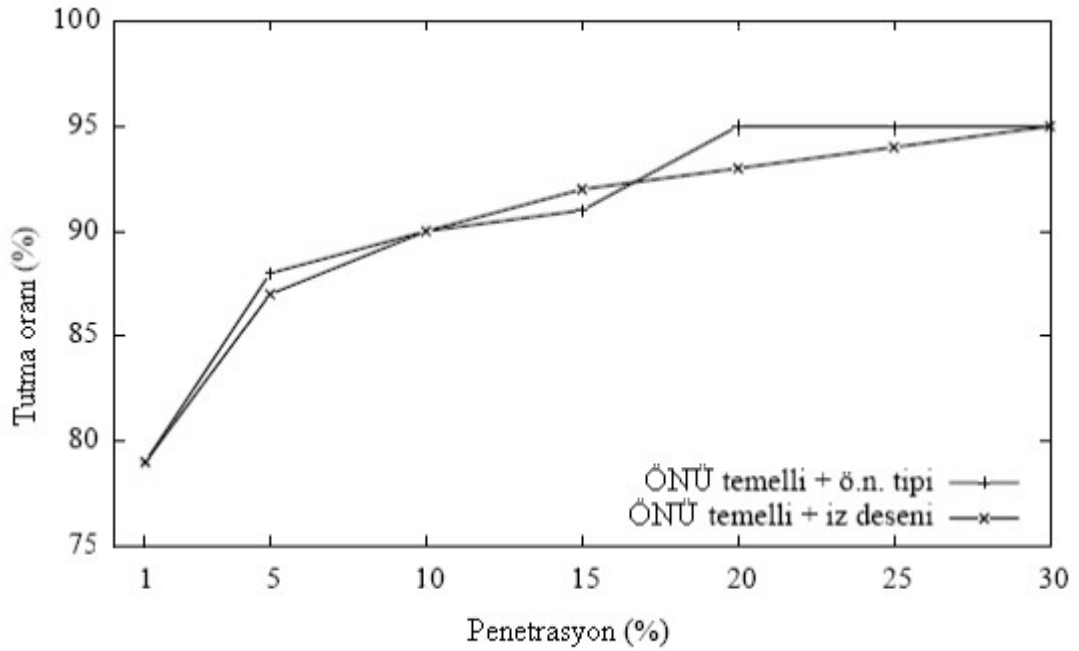
### 5.3. İZ DESENİ TEMSİLİ İLE ÖZELLİK NOKTASI TÜRLERİNİN ETKİLERİNİN KARŞILAŞTIRILMASI

Hem özellik noktalarını kullanan indeksleme metodlarında [11,16] faydalanılan özelliklerden biri olduğu için hem MCC temelli indekslemede [18] kaçınılan bir özellik olduğundan iz deseni temsilinin özellik noktası türüne göre üstün olup olmadığını irdelemek için deneyler yapılmıştır.

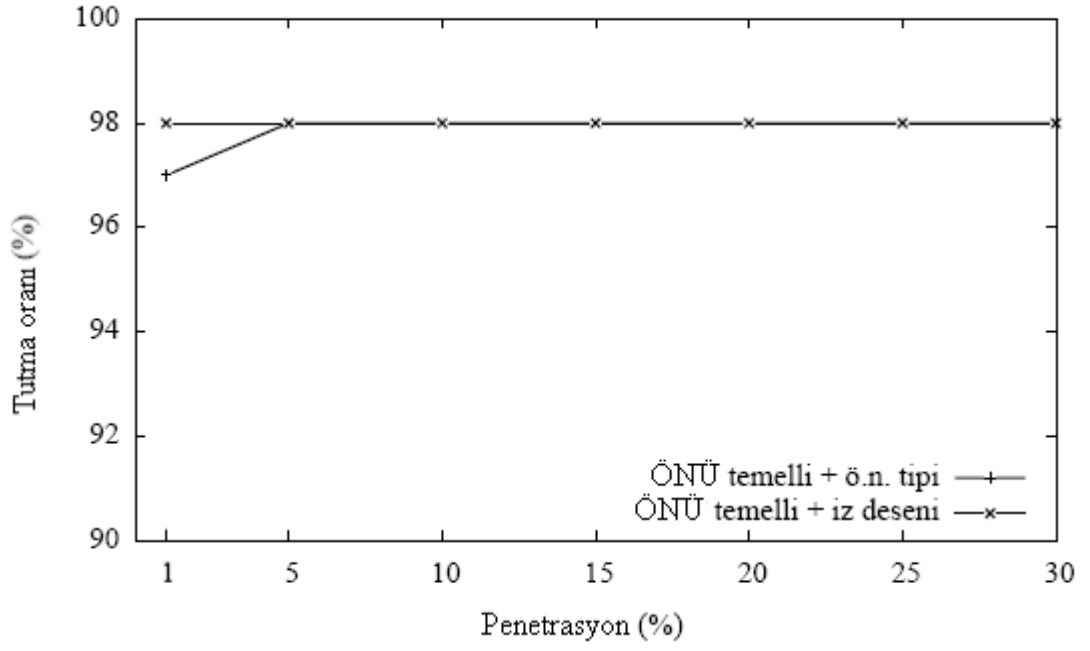
Bu deneyler sonunda sadece FVC2000DB2 veritabanı için özellik noktası türünün kullanılmasının daha yüksek tutma oranlarını sağladığı, FVC2000DB3 veritabanı için birbirine yakın tutma oranlarını sağladığı, diğer tüm veritabanları için ise iz deseni temsilinin daha yüksek tutma oranlarını sağladığı görülmüştür. Sonuç olarak genelde iz deseni temsilinin indeksleme sisteminin seçiciliğini özellik noktası türüne göre daha çok arttırdığı söylenebilir. (4.11) ve (4.12) ile verilen ifadeler aynı iş yüküne sebep olduğundan, iz deseni temsili yerine özellik noktaları üçlüleri kullanıldığında ortalama arama süresinin belirlenmesi için zaman ölçümü deneyleri yapılmamıştır. Sonuçlar Şekil 5.13'den Şekil 5.18'e kadar verilmiştir.



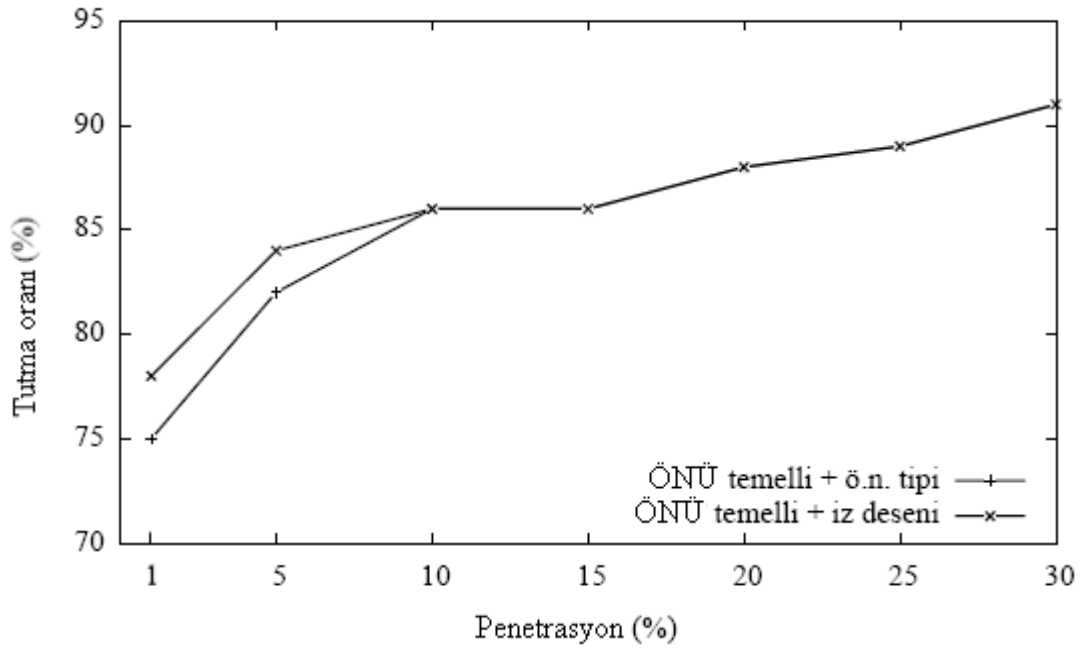
Şekil 5.13. FVC2000DB2 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.



Şekil 5.14. FVC2000DB3 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.

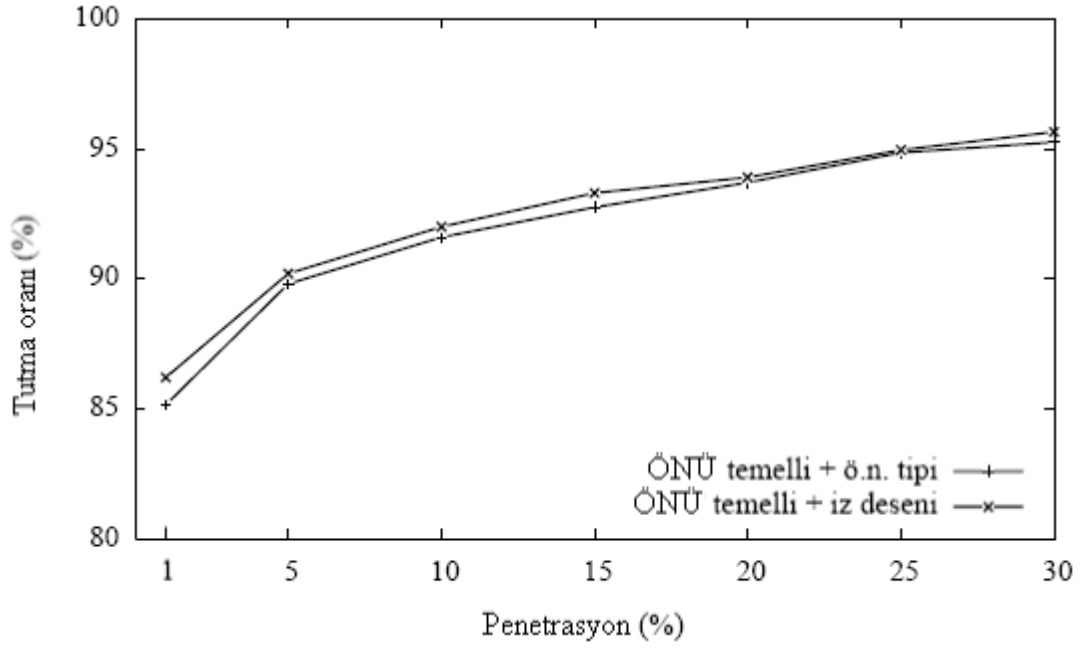


Şekil 5.15. FVC2002DB1 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indeksleme performansları.

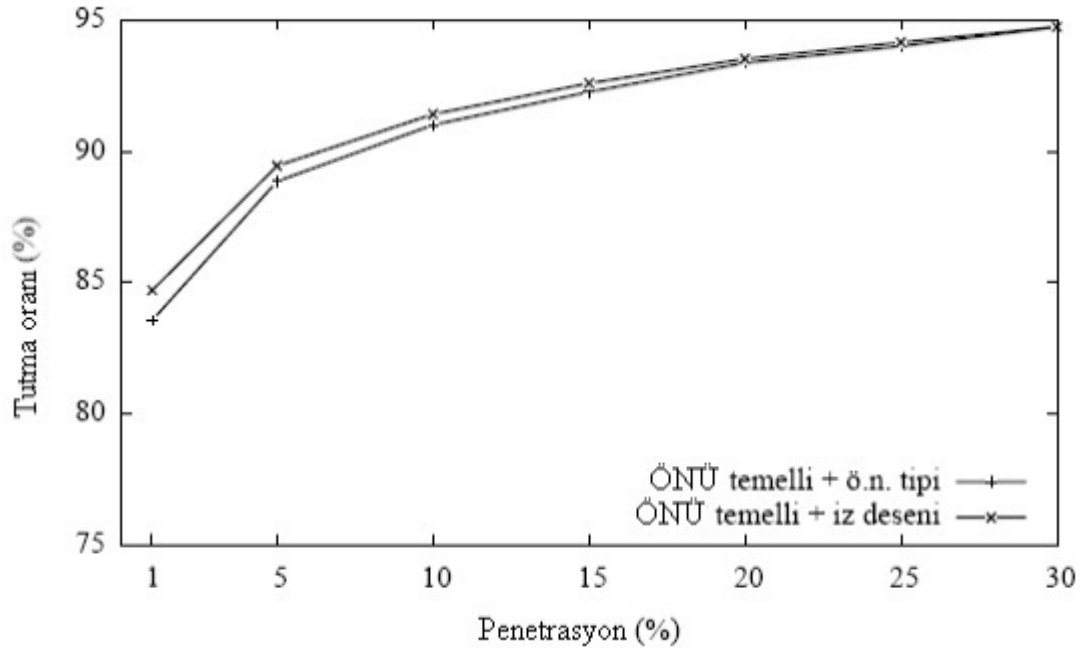


Şekil 5.16. FVC2004DB1 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indeksleme performansları.





Şekil 5.17. NISTDB4 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.

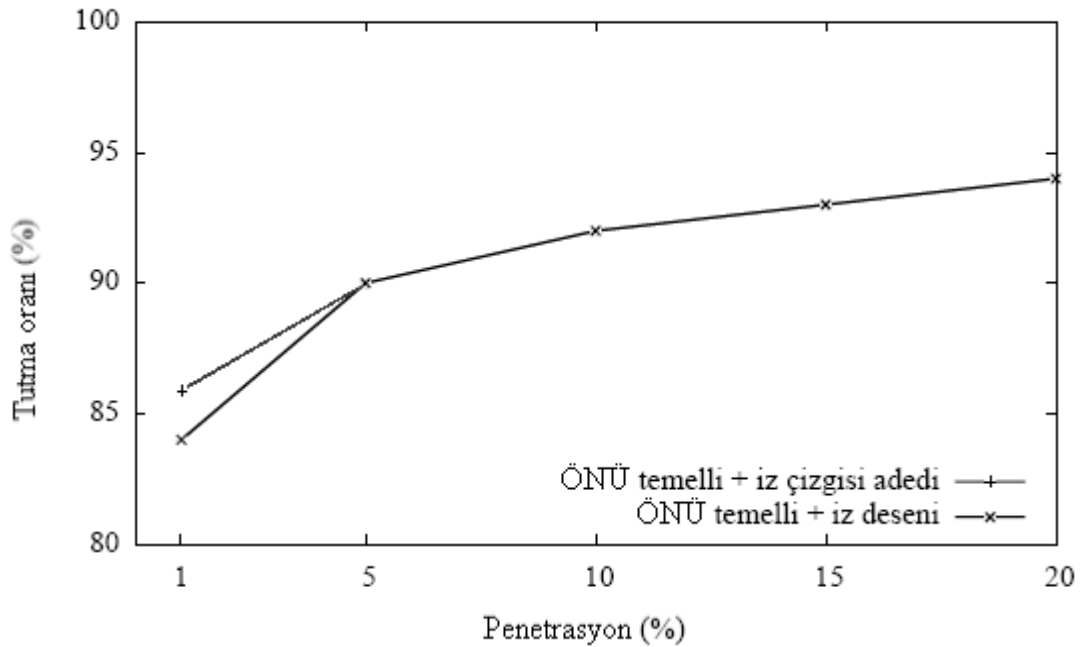


Şekil 5.18. NISTDB14 için iz deseni temsili kullanıldığında ve özellik noktaları türü kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.

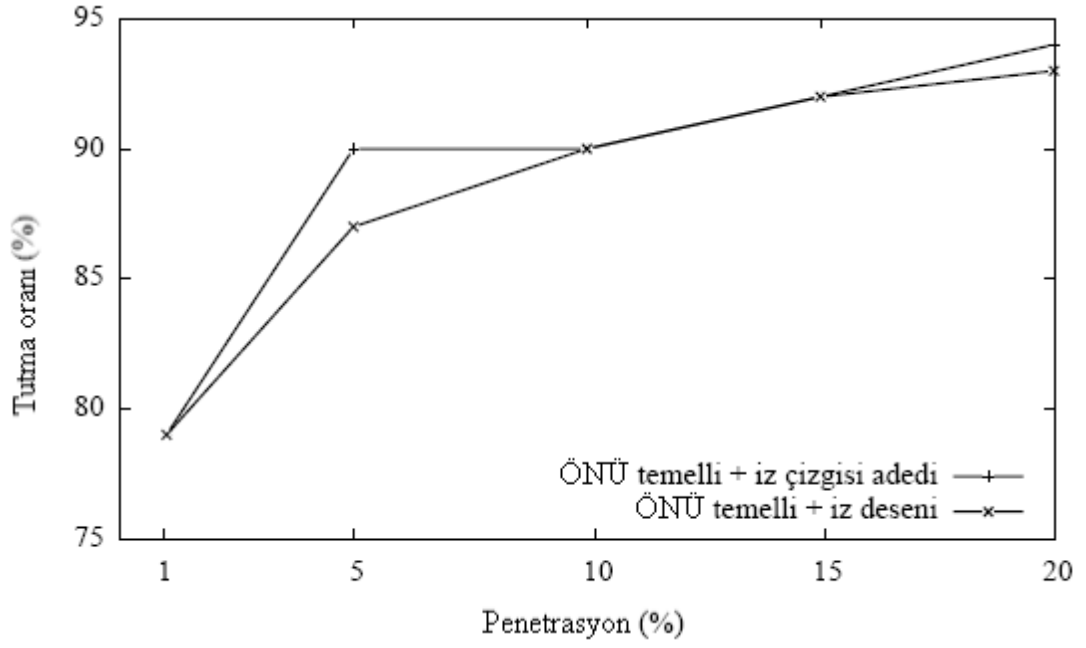
#### 5.4. İZ DESENİ TEMSİLİ İLE İZ ÇİZGİLERİ SAYILARININ ETKİLERİNİN KARŞILAŞTIRILMASI

İz deseni temsilinin yapısı incelendiğinde iz çizgileri sayısından ne kadar fazlasını verebildiği akla soru olarak gelebilir. İz deseni temsilinden iz çizgileri sayısı elde edilebilir. Bunun dışında iz deseni temsili iz çizgilerinin birbiriyle nasıl bağlı olduğu bilgisini de verir. İşte bu bilginin faydası iz deseni temsilinin faydasının iz çizgileri sayısından ne kadar fazla olduğundan anlaşılır.

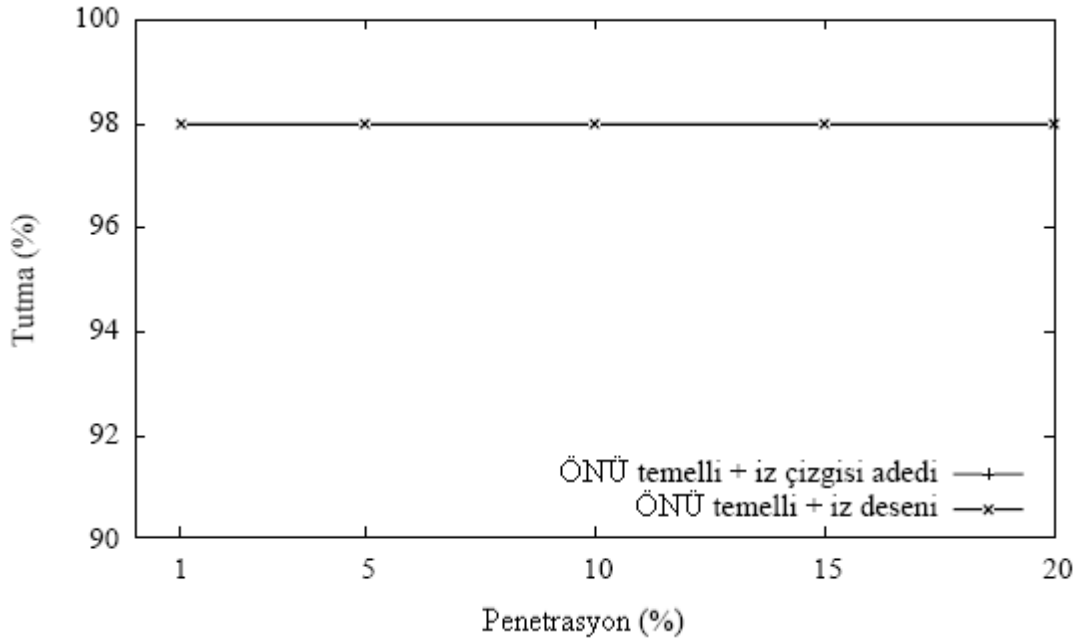
İz deseni temsilinden iz çizgileri sayıları elde edilebilse de adil kıyas için iz çizgileri sayıları doğrudan parmak izi resimlerinden elde edilerek karşılaştırma deneyleri yapılmıştır. Sonuçlara göre iz çizgileri sayıları FVC2000DB2 veritabanında iz deseni temsilinden daha başarılı çıkmıştır. Diğer tüm FVC veritabanlarında özellikle düşük penetrasyon oranları için ikisi de aynı tutma oranlarını sağlamıştır. Daha kalabalık olan NIST veritabanlarında ise iz deseni temsili, iz çizgileri sayılarına göre az da olsa üstünlük göstermiştir. Elde edilen sonuçlar Şekil 19'dan Şekil 24'e kadar verilmiştir.



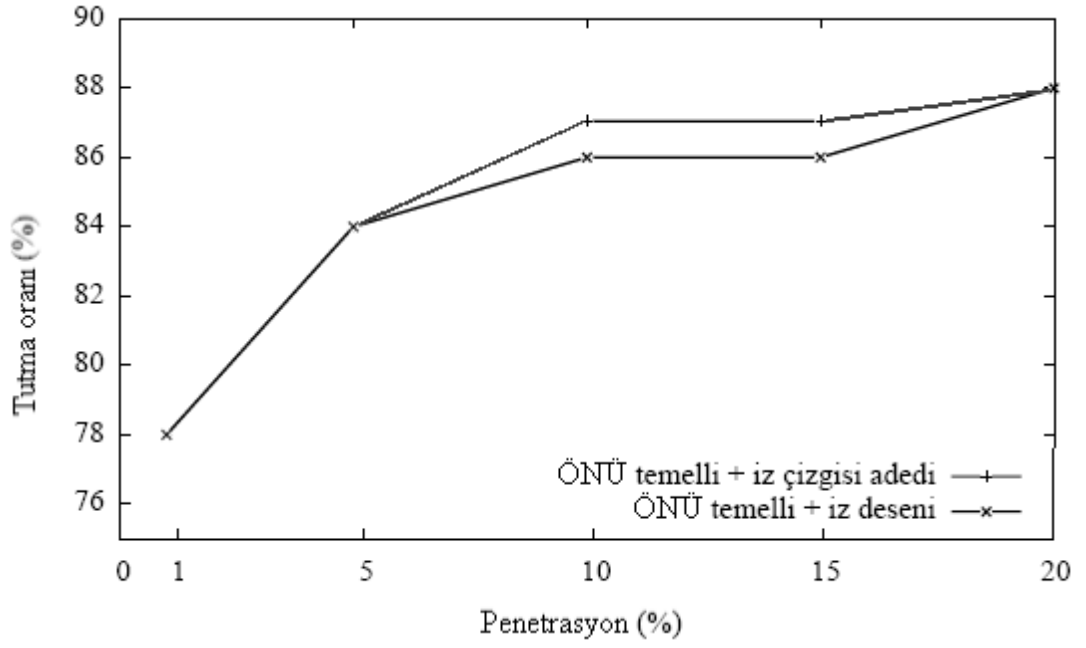
Şekil 5.19. FVC2000DB2 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.



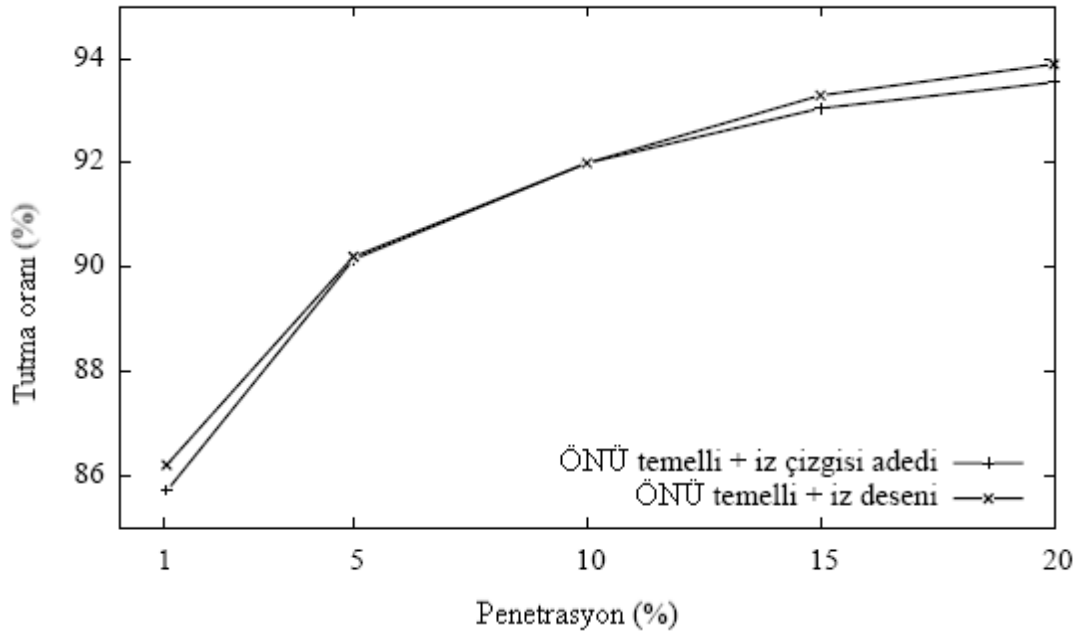
Şekil 5.20. FVC2000DB3 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.



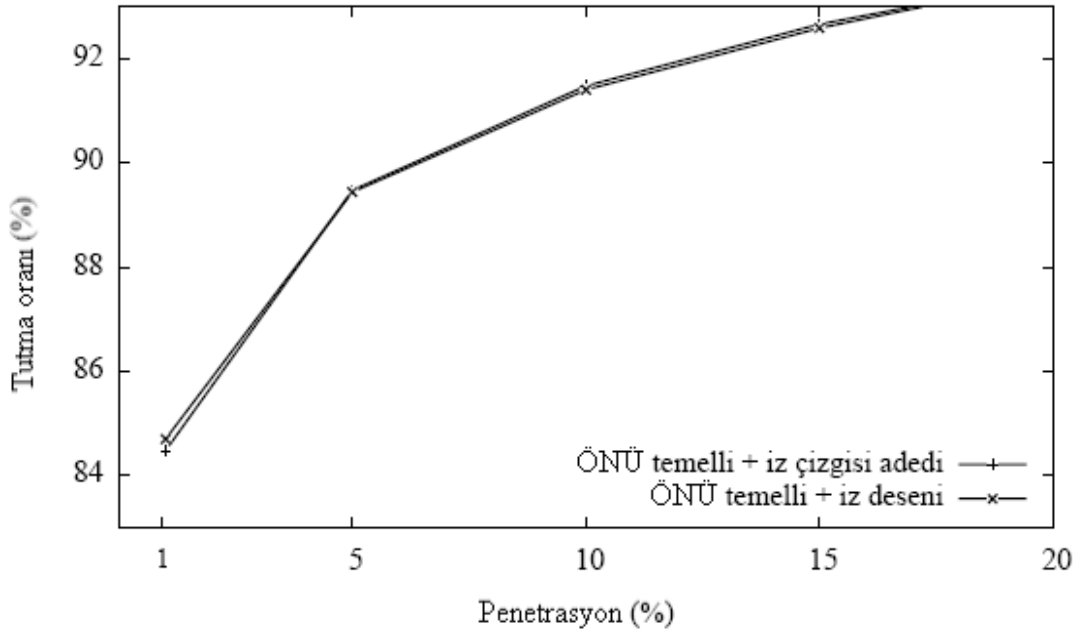
Şekil 5.21. FVC2002DB1 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indekisleme performansları.



Şekil 5.22. FVC2004DB1 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indeksleme performansları.



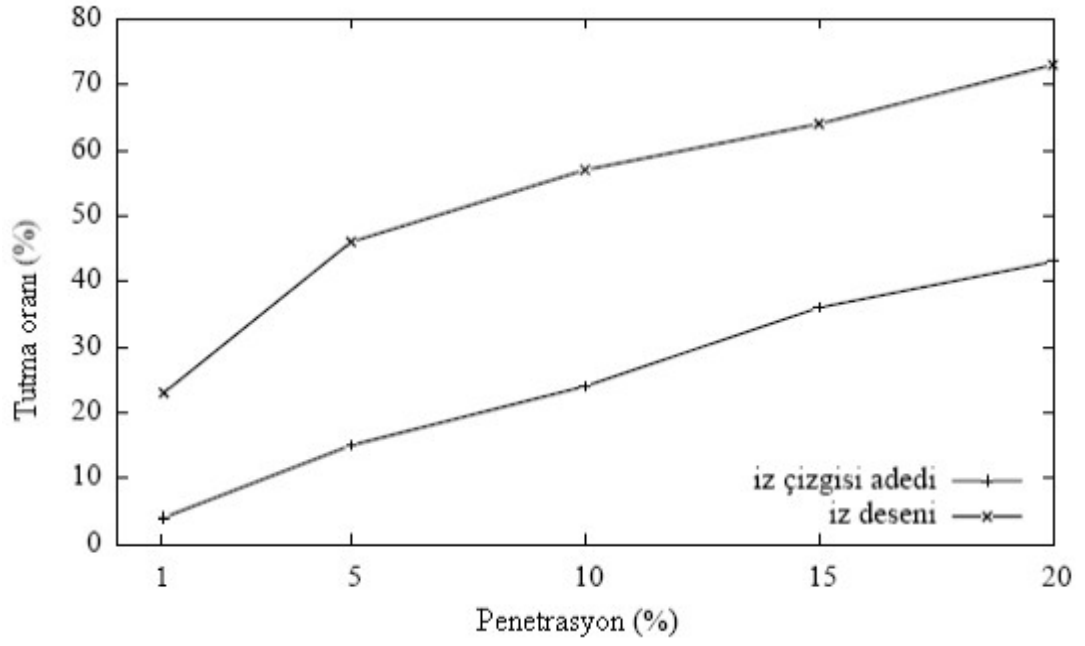
Şekil 5.23. NISTDB4 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indeksleme performansları.



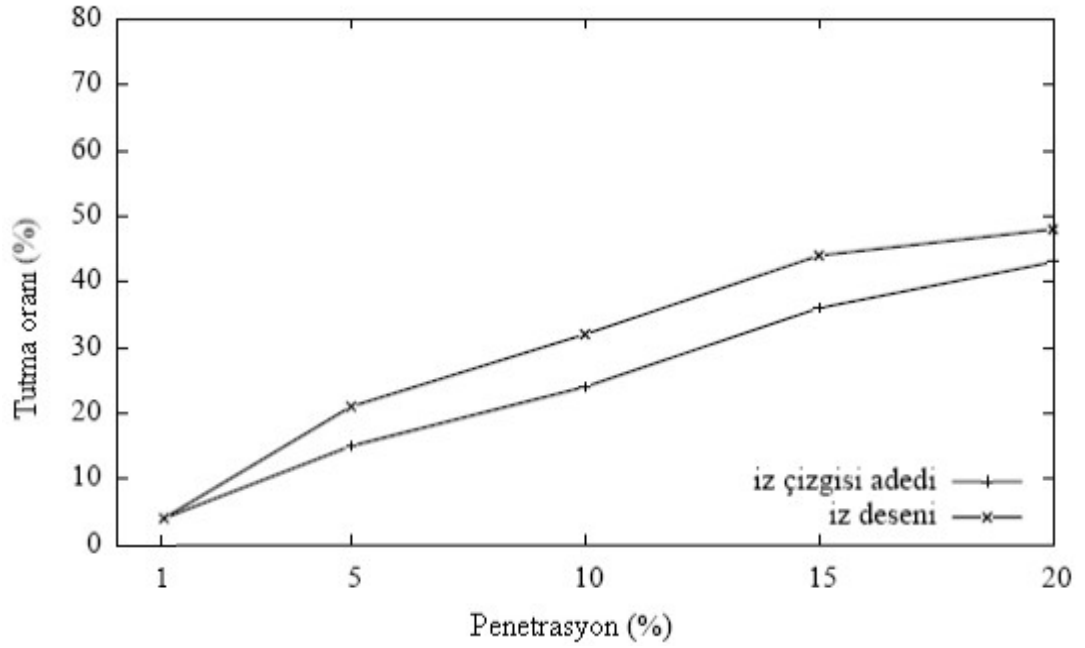
Şekil 5.24. NISTDB14 için iz deseni temsili kullanıldığında ve iz çizgileri sayıları kullanıldığında özellik noktaları üçlüsü temelli yöntemin indeksleme performansları.

FVC2000DB2 veritabanı için iz deseni temsilinin %1 penetrasyon oranına karşı %2 daha düşük tutma oranı göstermesi, diğer FVC veritabanlarında üstünlüğünü belli edememesi FVC veritabanlarının 100'er parmağa ait izleri tutan küçük veritabanları olmasından ve tesadüfen sadece 1 parmak izinin tutturulması veya kaybedilmesinin sonuçları %1 kadar değiştirmesiyle açıklanabilir. Kalabalık NIST veritabanları ile yapılan deneyler iz deseni temsilinin üstünlüğünü göstermiştir.

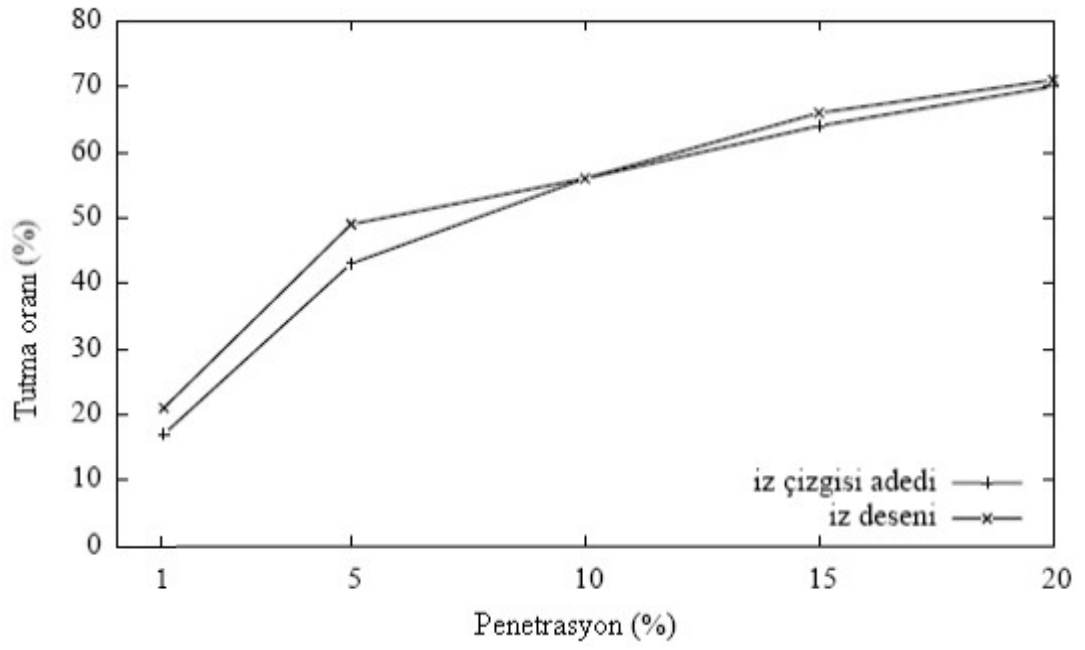
İz deseni temsilinin iz çizgisi sayılarına göre seçicilik farkı küçümsenmemelidir. Her iki durumda da özellik noktalarının birbirlerine göre bağıl açıları ve aralarındaki mesafeler gibi özellikler indekslemede kullanıldığından iz deseninin üstünlüğünü iz çizgileri sayılarına göre belirgin olarak gösterememesi iz deseni temsilinin yakalayıp iz çizgileri sayılarının yakalayamadığı parmak izlerinin indeksleme etkisiyle elenmesiyle açıklanabilir. Sadece iz deseni temsili ve sadece iz çizgileri sayıları ile yapılan indeksleme deneyleri iz deseni temsilinin daha üstün olduğunu göstermiştir. Yapılan deney sonuçları Şekil 5.25'ten Şekil 5.30'a kadar verilmiştir.



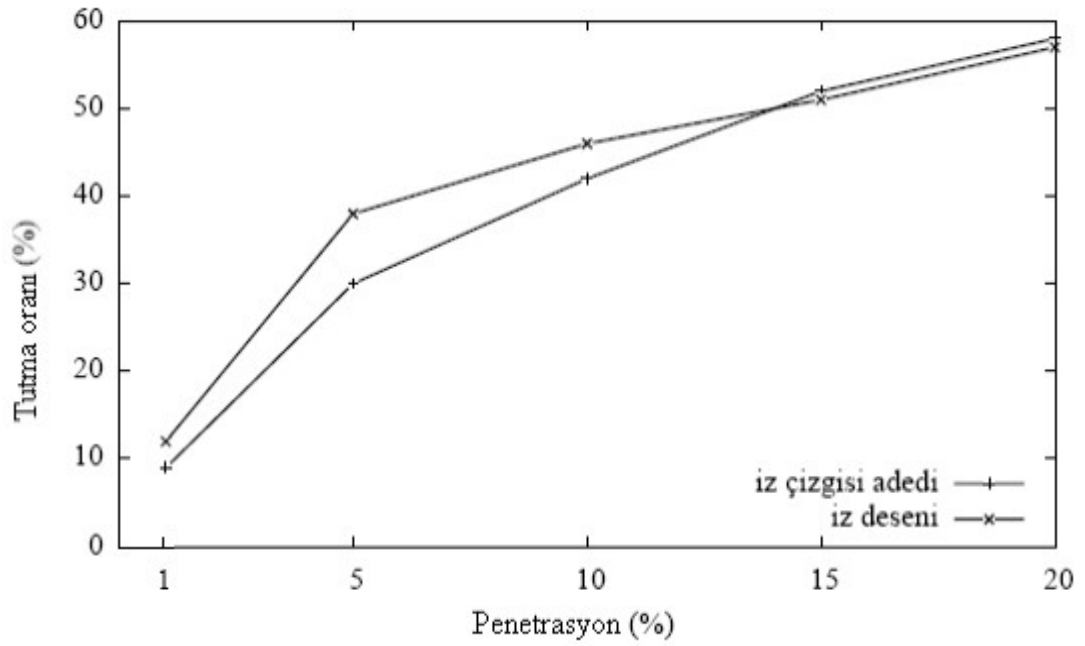
Şekil 5.25. FVC2000DB2 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.



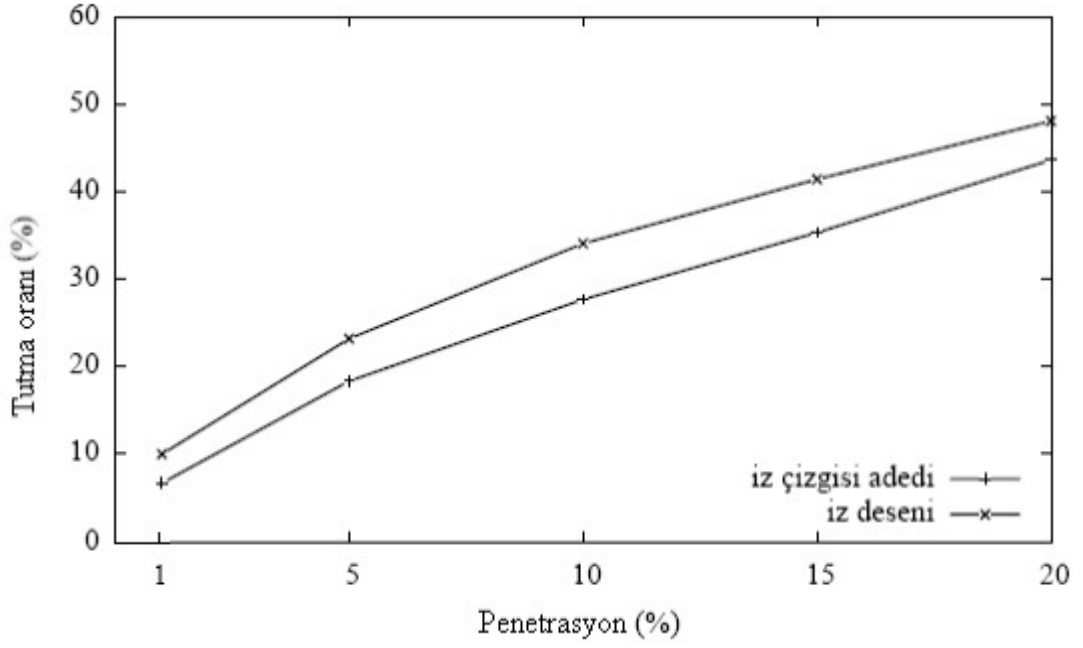
Şekil 5.26. FVC2000DB3 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.



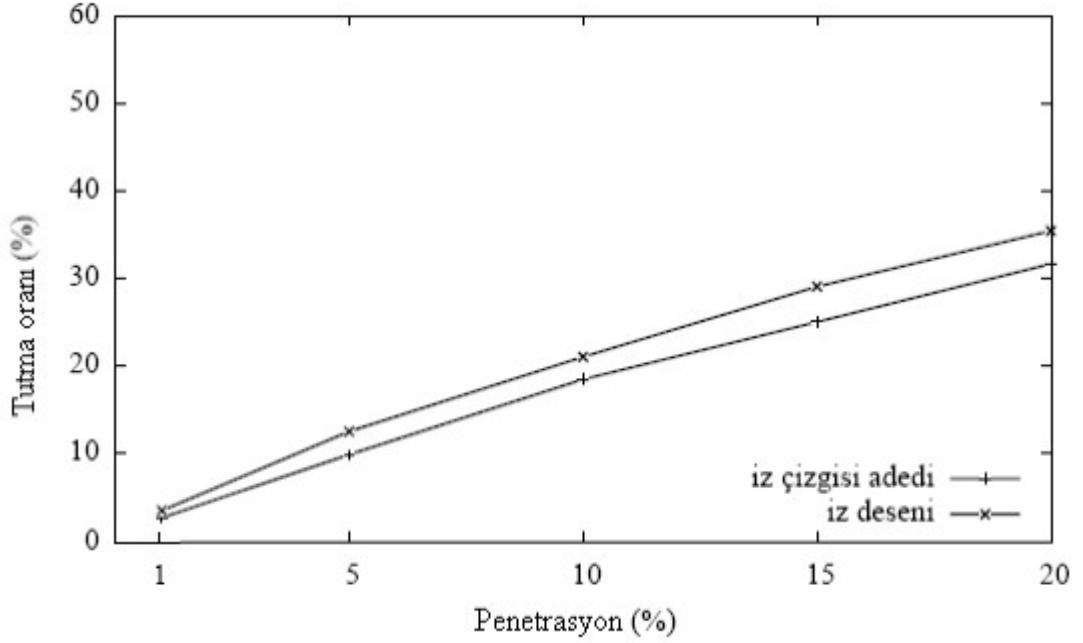
Şekil 5.27. FVC2002DB1 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.



Şekil 5.28. FVC2004DB1 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.



Şekil 5.29. NISTDB4 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.



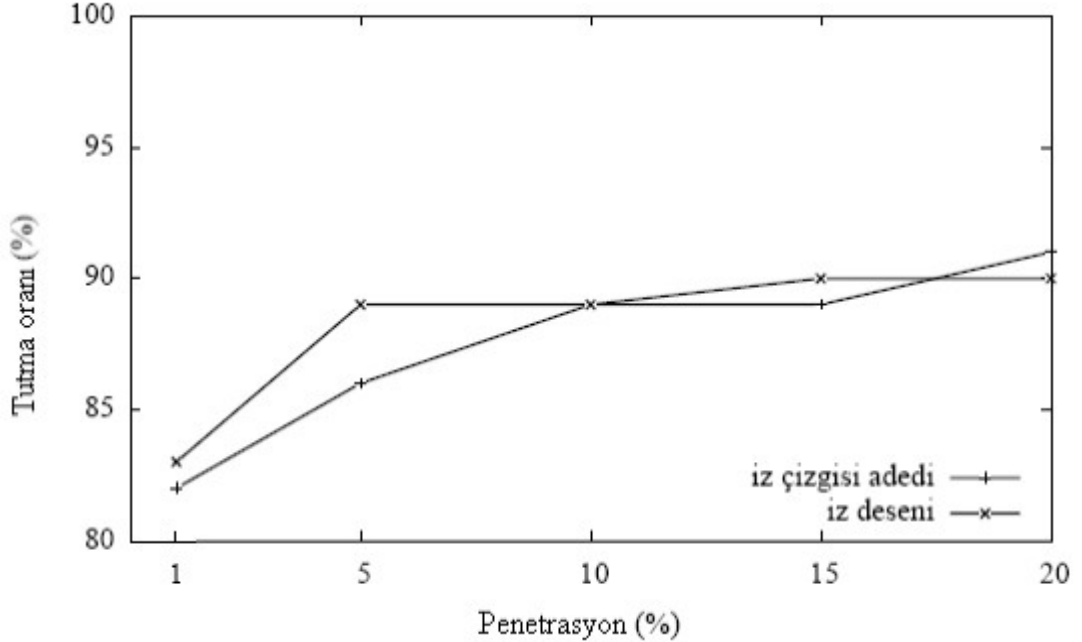
Şekil 5.30. NISTDB14 için sadece iz çizgileri sayısı kullanılarak ve sadece iz deseni kullanılarak yapılan indeksleme performansları.

Ayrıca Şekil 19'dan Şekil 24'e kadar verilen sonuçlara göre özellik noktalarına ait diğer özellikler kullanıldığında iz deseni temsilinin iz çizgileri sayılarına göre daha faydalı sayılmayacağı da akla gelebilir. Fakat bu özellikler iz deseni temsili ve iz

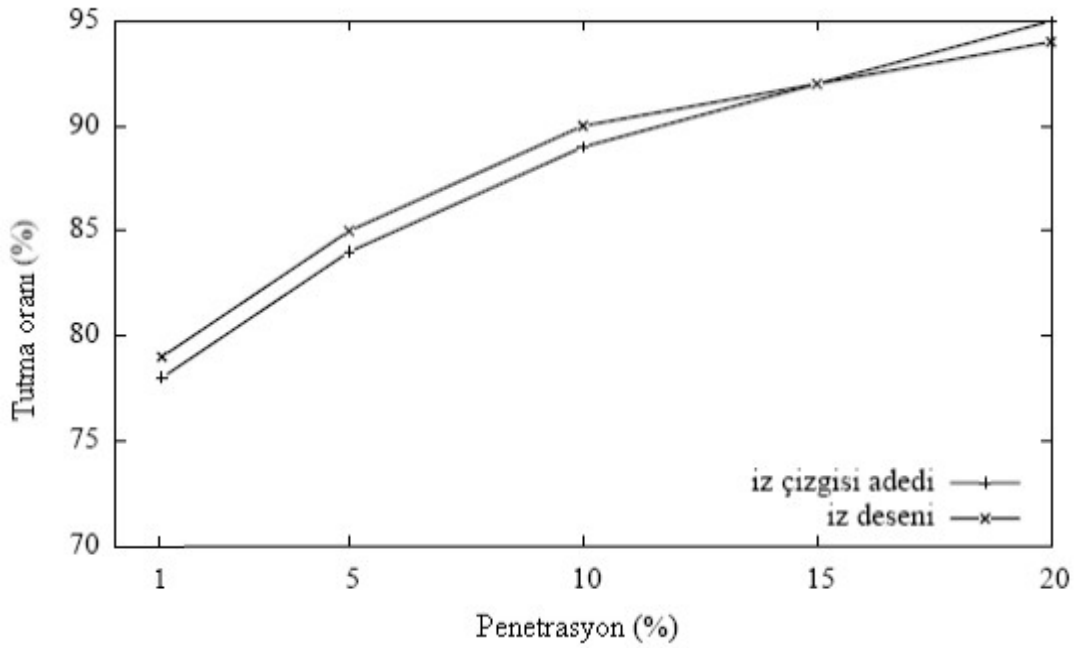


çizgileri sayıları ile ayrı ayrı yapılan indekslemelerde indekslemeye katılmayıp sonuç kümesinin daraltılmasında kullanıldığı zaman Şekil 5.31'den Şekil 5.36'ya kadar verilen deney sonuçlarından da görülebileceği gibi iz deseni temsilinin iz çizgileri sayılarına göre yine daha seçici olduğu ortaya çıkmaktadır. Sonuçlara göre FVC2002DB1 ve FVC2004DB1 haricindeki tüm veritabanlarında diğer özellikler sonuç kümesinin eleman sayısının azaltılması için kullanıldığında iz deseni temsilinin iz çizgileri sayısına göre belirgin olarak daha seçici olduğu anlaşılmaktadır. Sadece FVC2004DB1 veritabanı için bunun aksi bir durum olmakta, FVC2002DB1 veritabanı için düşük penetrasyon oranları için iz deseni temsili ve iz çizgileri sayıları aynı seçiciliğe sahip olmaktadır.

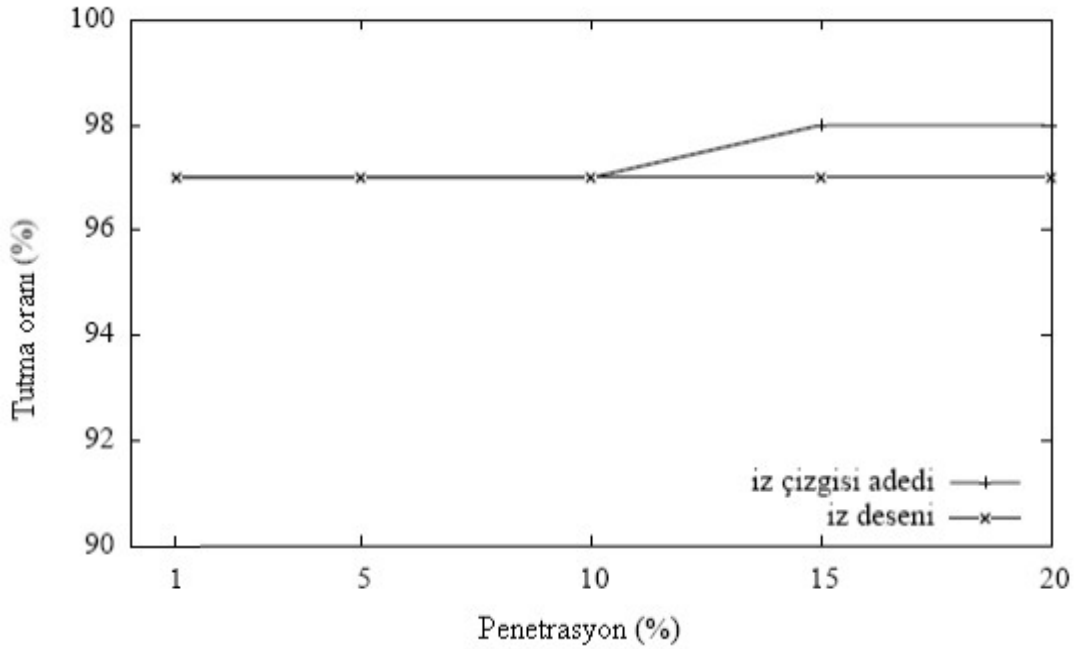
Sonuç olarak iz deseni temsili iz çizgileri sayısından diğer özellikler kullanıldığında da daha seçicidir. Özellik noktalarının diğer özellikleri indekslemeye katıldığında indekslemenin etkisiyle iz deseni temsilinin bulunmasını sağlayacağı fakat iz çizgileri sayılarının bulunmasında faydalı olamayacağı parmak izleri elenmektedir.



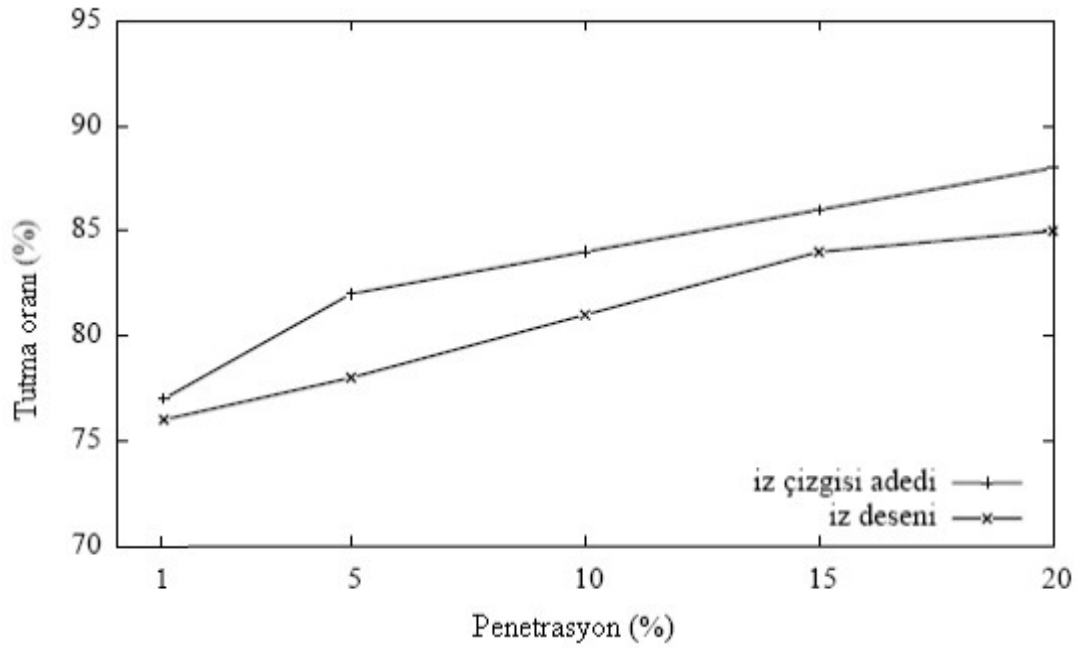
Şekil 5.31. FVC2000DB2 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.



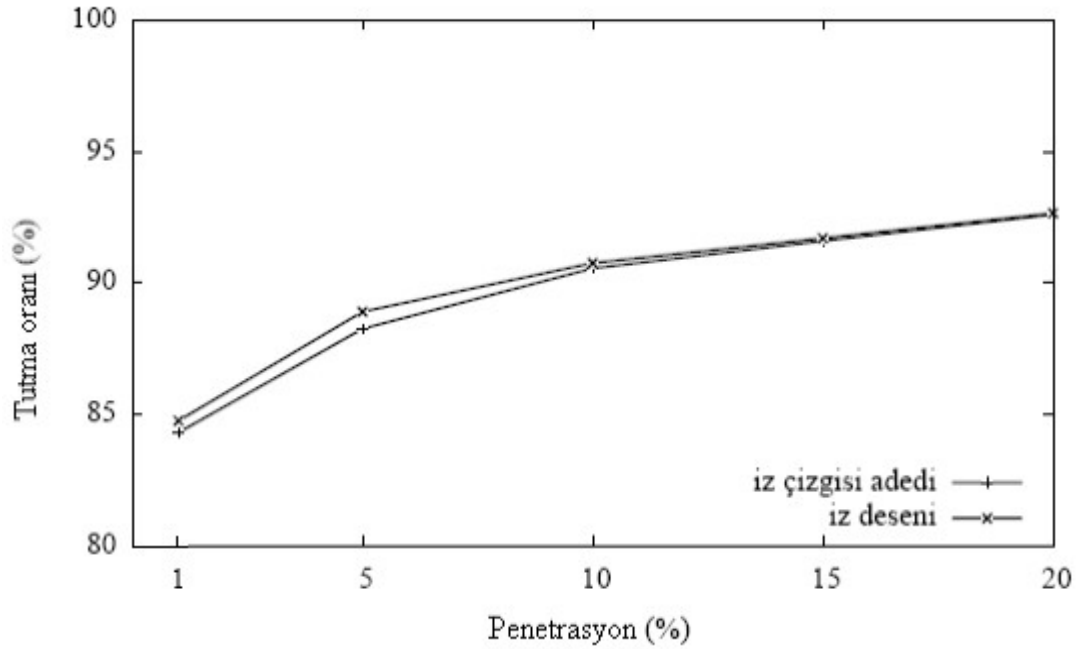
Şekil 5.32. FVC2000DB3 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indekisleme performansları.



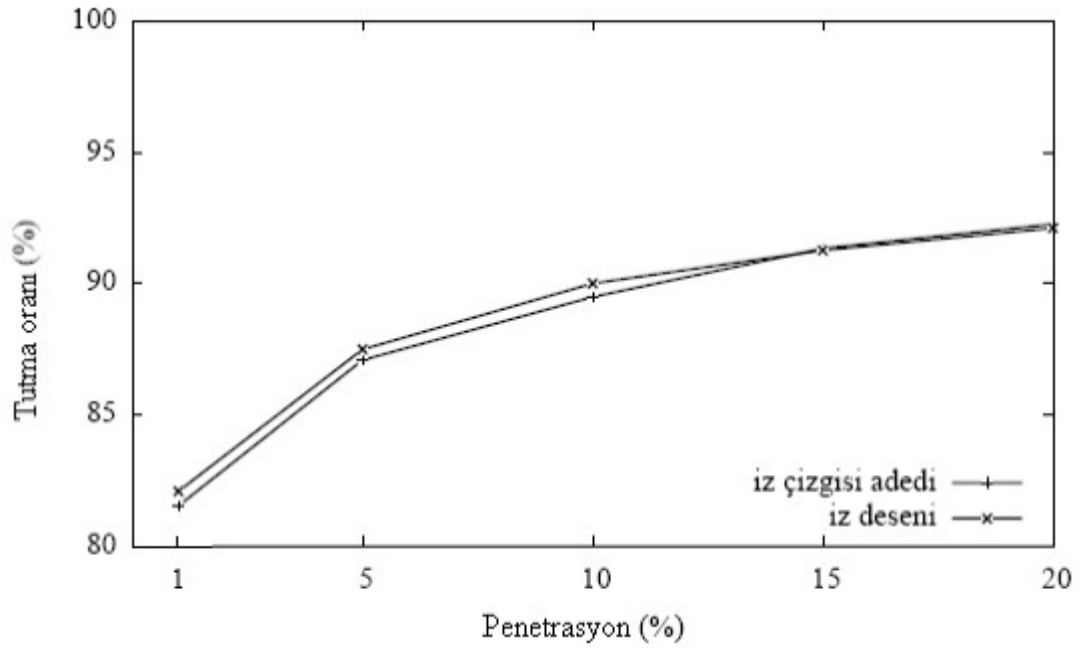
Şekil 5.33. FVC2002DB1 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indekisleme performansları.



Şekil 5.34. FVC2004DB1 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indekisleme performansları.



Şekil 5.35. NISTDB4 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indekisleme performansları.



Şekil 5.36. NISTDB14 için iz çizgileri sayısı kullanılarak ve iz deseni kullanılarak yapılan indeksleme performansları.

## BÖLÜM 6

### SONUÇ

Bu çalışma kapsamında özellik noktaları üçlüsü temelli parmak izi tanıma problemlerinde kullanılabilir iz deseni temsili önerilmiştir. Önerilen temsil metodu indekslemede kullanılmamış ama indeksleme sisteminin sorgulamalara karşı verdiği aday listelerinin doğru sonuçları içerme ihtimalini kuvvetlendirmek için kullanılmıştır.

İz deseni temsiline ya da iz desenini bünyesindeki iz çizgilerini yeniden inşa ettirebilecek kadar temsil edebilen başka bir yapıya literatürde daha önce yer almış çalışmalarda rastlanılmamıştır. İz deseni temsili iz çizgilerinin desenini tümüyle temsil eden özgün bir gösterim yöntemidir.

İz deseni temsili kullanabilmek için yine bu çalışma kapsamında gerçekleştirilen özellik noktaları üçlüsü temelli indeksleme uygulamasının iz deseni temsili kullanılmadan da literatürde geçen MCC temelli indeksleme [18] uygulamasından daha iyi sonuçlar çıkardığı görülmüştür. Parmak izlerinin sadece özellik noktalarıyla temsil edildiği sistemler ve birbiriyle uyumlu çalışabilmesi için parmak izi temsillerinin aktarımının özellik noktaları bilgileriyle yapıldığı sistemler düşünüldüğünde parmak izlerinin sadece özellik noktalarını okuyarak indeksleme yapan MCC temelli indekslemenin [18] diğer yöntemlerin neredeyse hepsinden üstün performans sergilemesi, ayrıca bu başarısını diğer yöntemlerin özellik noktası bilgilerinden başka özellikler de kullanmasına rağmen göstermesi çok önemlidir. Bu sebeple MCC temelli indeksleme [18] kısa zamanda araştırmacıların ilgisini çekmiştir. Burada uygulanan özellik noktası üçlüsü temelli indeksleme de sadece özellik noktaları bilgileriyle indeksleme yapmaktadır ve yapılan deneylerle MCC temelli indekslemeden [18] daha yüksek tutma oranlarını sağladığını göstermiştir.

İz deseni temsili kullanıldığında ise indeksleme performansının arttığı, özellik noktaları üçlüsü temelli diğer çalışmalarda uygulanan özellik noktası türü ve iz çizgileri sayılarının da faydalı olduğu, iz deseni temsilinin ise daha faydalı olduğu anlaşılmıştır.

İleride iz deseni temsili ile toleranslı aramaların yapılması, parmak izi doğrulamasının yapılması, parmak izlerinin temsili ve yeniden inşası ile ilgili çalışmalar yapılabilir.

İz deseni ile parmak izi temsilinin başarılı olması halinde temsil edilen parmak izlerinin yeniden inşası yapılmadan yönelim haritalarının, çekirdek ve delta noktalarının doğrudan iz deseni temsillerinden elde edilmesi ve parmak izinin sınıflandırılması için metodlar geliştirilebilir. Ayrıca gizliliğin sağlanması için yeniden inşası mümkün olmayan iz deseni temsilleri geliştirilebilir.

Özellik noktaları üçlüleri arasında kalan üçgensel iz deseni yerine tek özellik noktası etrafındaki iz desenlerinin temsiline yönelik çalışmalar denenebilir.

## KAYNAKLAR

1. Matoni, D., Maio, D and Jain, A., "Handbook of Fingerprint Recognition", *Springer – Verlag*, (2003)
2. Lumini, A., Maio, D. and Maltoni, D., "Continuous versus Exclusive Classification for Fingerprint Retrieval", *Pattern Recognition Letters*, 18, 1027-1034, (1997).
3. Wang, Y., Hu, J. and Phillips, D., "A Fingerprint Orientation Model Based on 2D Fourier Expansion (FOMFE) and Its Application to Singular-Point Detection and Fingerprint Indexing", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29 (4), 573-585, (2007).
4. Liu, M. and Yap, P. T., "Invariant Representation of Orientation Fields for Fingerprint Indexing", *Pattern Recognition*, 45 (7), 2532-2542, (2012).
5. De Boer, J., Bazen, A. M. and Gerez, S.H., "Indexing Fingerprint Databases Based on Multiple Features," *ProRISC 2001, Annual Workshop on Circuits, Systems and Signal Processing*, 300-306, (2001).
6. Liu, T., Zhu, G., Zang, C. and Hao, P., "Fingerprint Indexing Based on Singular Point Correlation", *Proceedings of IEEE International Conference on Image Processing*, 3, 293-296, (2005).
7. Li, J., Yau, W. Y. and Wang, H., "Fingerprint Indexing Based On Symmetrical Measurement", *Proceedings of the 18th International Conference on Pattern Recognition*, 1, 1038-1041, (2006).
8. Umarani, J., Viswanathan, J., Gupta, A. K. and Gupta, P., "Minutiae Based Geometric Hashing for Fingerprint Database", **8th International Conference, ICIC 2012**, Huangshan, China, 422-427, (2012).
9. Feng, J. and Cai, A., "Fingerprint Indexing Using Ridge Invariants", *Proceedings of the 18th International Conference on Pattern Recognition*, 4, 433-436, (2006).
10. Germain, R. S., Califano, A. and Colville, S., "Fingerprint Matching Using Transformation Parameter Clustering", *IEEE Computational Science and Engineering*, 4 (4), 42-49, (1997).

11. Bhanu, B. and Tan, X., "A Triplet Based Approach for Indexing of Fingerprint Database for Identification", *Proceedings of the Third International Conference on Audio- and Video-Based Biometric Person Authentication*, 205-210, (2001).
12. Bebis, G., Deaconu, T. and Georgiopoulos, M., "Fingerprint Identification Using Delaunay Triangulation", *In Proceedings of 1999 IEEE International Conference on Intelligence, Information, and Systems*, 452-459, (1999).
13. Liu, N., Yin, J. Y. and Zhang, H., "A Fingerprint Matching Algorithm Based On Delaunay Triangulation Net", *The Fifth International Conference on Computer and Information Technology*, CIT 2005, (2005).
14. Ross, A. and Mukherjee, R., "Augmenting Ridge Curves with Minutiae Triplets for Fingerprint Indexing", *Proceedings of SPIE Conference on Biometric Technology for Human Identification IV*, (2007).
15. Singh, R., Vatsa, M. and Noore, A., "Fingerprint Indexing using Minutiae and Pore Features", *Proceedings of the 2009 International Conference on Image Processing, Computer Vision, & Pattern Recognition, IPCV 2009*, (2009)
16. Liang, X., Bishnu, A. and Asano, T., "A Robust Fingerprint Indexing Scheme Using Minutia Neighborhood Structure and Low-Order Delaunay Triangles", *IEEE Transactions on Information Forensics and Security*, 2 (4), 721-733, (2007).
17. Iloanusi, O.; Gyaourova, A. and Ross, A., "Indexing fingerprints using minutiae quadruplets," *(CVPRW), 2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, .127-133, (2011).
18. Cappelli, R., Ferrara, M. and Maltoni, D., "Fingerprint Indexing Based on Minutia Cylinder-Code", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33 (5), 1051-1057, (2011).
19. Muñoz-Briseño, A., Gago-Alonso, A. and Hernández-Palancar, J., "Fingerprint Indexing with Bad Quality Areas", *Expert Systems with Applications*, 40 (5), . 1839-1846, (2013).
20. Gago-Alonso, A., Hernández-Palancar, J., Rodríguez-Reina, E. and Muñoz-Briseño, A., "Indexing and Retrieving in Fingerprint Databases Under Structural Distortions", *Expert Systems with Applications*, 40 (8), 2858-2871, (2013).
21. Cappelli, R., "Fast and Accurate Fingerprint Indexing Based on Ridge Orientation and Frequency", *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 41 (6), (2011)



22. Cappelli, R. and Ferrara, M., "A fingerprint retrieval system based on level-1 and level-2 features", *Expert Systems with Applications* 39 (2012) 10465–1047
23. Feng, J., Ouyang, Z. and Cai, A., "Fingerprint matching using ridges", *Pattern Recognition*, 39 (11), 2131-2140, (2006).
24. Biswas, S., Ratha, N. K., Aggarwal, G. and Connell, J., "Exploring Ridge Curvature for Fingerprint Indexing", *Proceedings of the 2nd IEEE International Conference on Biometrics: Theory, Applications and Systems*, . 1-6, (2008).
25. Zhang, W., Wang, S. and Wang, Y., "Structure Matching Algorithm of Fingerprint Minutiae Based on Core Point", *Journal of Acta Automatica Sinica*, 29 (6), 354-363, (2003).
26. Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L. and Jain, A. K., "FVC2000: Fingerprint Verification Competition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (3), 402-412, (2002).
27. Cappelli, R., Ferrara, M. and Maltoni, D., "Minutia Cylinder-Code: A New Representation and Matching Technique for Fingerprint Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (12), (2010)
28. Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L. and Jain, A. K., "FVC2002: Second Fingerprint Verification Competition", *Proceedings of 16th International Conference on Pattern Recognition*, 3, 811-814, (2002).
29. Shuai, X., Zhang, C. and Hao, P., "Fingerprint Indexing Based on Composite Set of Reduced SIFT Features", *Proc. Int'l Conf. Pattern Recognition*, 1-4, (2008)
30. Watson, C. I. and Wilson, C. L., "NIST Special Database 4, Fingerprint Database", *US National Institute of Standards and Technology (NIST)*, (1992)
31. Watson, C. I., "NIST Special Database 14, Fingerprint Database", *US National Institute of Standards and Technology (NIST)*, (1993)
32. Watson, C. I., Garris, M. D., Tabassi, E., Wilson, C. L., McCabe, R. M., Janet, S. and Ko, K., "User's Guide to NIST Biometric Image Software (NBIS)", *National Institute of Standards and Technology*, 48-72, (2007)
33. Internet: National Institute of Standards and Technology, "NBIS Home Page", <http://www.nist.gov/itl/iad/ig/nbis.cfm> (2014)
34. Internet: University of Bologna, "Biometrics System Laboratory", <http://biolab.csr.unibo.it> (2015)

35. Uysal, M. and Görgünođlu, S., "Ridge Pattern Representation for Fingerprint Indexing", *Elektronika ir Elektrotechnika*, 20 (7), 2014

## **ÖZGEÇMİŞ**

Murat UYSAL 1978 yılında Zonguldak / Çaycuma'da doğdu. 1996 yılında Zonguldak Teknik Lisesi - Bilgisayar Donanımı Bölümünden mezun oldu. 2002 yılında Orta Doğu Teknik Üniversitesi Bilgisayar Öğretmenliği Bölümündeki eğitimini tamamlayarak mezun oldu. 2006 yılında Zonguldak Karaelmas Üniversitesi Elektrik Elektronik Mühendisliği Bölümündeki Yüksek Lisans eğitimini tamamladı. 2010 yılında Karabük Üniversitesi Bilgisayar Mühendisliği Bölümünde Doktora çalışmasına başlamıştır. 2002 yılından beri Milli Eğitim Bakanlığına bağlı okullarda bilgisayar öğretmeni olarak görev yapmaktadır.

### **ADRES BİLGİLERİ**

Adres: Çaycuma Mesleki ve Teknik Anadolu Lisesi  
Pehlivanlar Mah.  
67900 Çaycuma / ZONGULDAK  
Tel : 0372 615 15 47  
Faks: 0372 615 25 06  
e-posta: uysal.tr@gmail.com