

**ŐEKİL TANIMA PROBLEMİ İÇİN ÇOKLU EŐLEMENİN SONSUZ NORM
UZAKLIĐI ALTINDA GERÇEKLEŐTİRİLMESİ**

CANER ERDEM

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĐİ**

**TOBB EKONOMİ VE TEKNOLOĐİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜŐÜ**

TEMMUZ 2010

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Ünver KAYNAK

Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığımı onaylarım.

Doç. Dr. Erdoğan DOĞDU

Anabilim Dalı Başkanı

Caner ERDEM tarafından hazırlanan ŞEKİL TANIMA PROBLEMİ İÇİN ÇOKLU EŞLEMENİN SONSUZ NORM UZAKLIĞI ALTINDA GERÇEKLEŞTİRİLMESİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. M. Fatih Demirci

Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Tolga Girici

Üye : Yrd. Doç. Dr. Esra Kadiođlu Ürtiř

Üye : Yrd. Doç. Dr. M. Fatih Demirci

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

CANER ERDEM

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Bilgisayar Mühendisliği
Tez Danışmanı : Yrd. Doç. Dr. M. Fatih Demirci
Tez Türü ve Tarihi : Yüksek Lisans – Temmuz 2010

Caner ERDEM

**ŞEKİL TANIMA PROBLEMİ İÇİN ÇOKLU EŞLEMENİN SONSUZ NORM
UZAKLIĞI ALTINDA GERÇEKLEŞTİRİLMESİ**

ÖZET

Çizge eşleme birçok şekil tanıma algoritmasının merkezinde yer alır. Çizge eşleme algoritmalarının büyük bir bölümü çizge düğümleri arasında bire bir bağlantı aramaktadır. Ancak bir çizgenin düğümlerinden oluşan bir alt kümeyi başka bir çizgenin düğümlerinden oluşan bir alt kümeyle çoklu bağlantı ile karşılaştırmak daha gerçekçi bir yaklaşımdır. Bu tezde çizge düğümleri arasında çoklu eşleme yoluyla bir benzerlik ilişkisi aranmıştır. Çoklu eşlemeyi çizgeler üzerinde gerçekleştirebilmek için çizgeler norm uzayda nokta kümeleri biçiminde ifade edilmiştir. Norm uzaydaki nokta kümelerini karşılaştırmak için polinom zamanda çalışan ve doğal yoldan çoklu eşleme yapan kullanışlı algoritmalar vardır. Bu nedenle çizge eşleme problemi nokta eşleştirme problemine indirgenmiştir. Çizge düğümleri arasındaki uzaklıkların izometrik şekilde korunarak norm uzayda noktalar biçiminde ifade edilmesi öncelikli amaçlardan biri olmuştur. Yapılan deneylerin sonuçlarına göre bu yaklaşımın çizge eşleme probleminin çözümündeki etkinliğini arttırmak için referans noktalarıyla eşleme yöntemi geliştirilmiştir. Bu doğrultuda yapılan çalışmaların sonuçları paylaşılmış ve deneysel olarak yöntemin şekil tanımadaki başarımı gösterilmiştir.

Anahtar kelimeler: Şekil tanıma, nesne tanıma, çizge eşleme, metrik uzayda taşıma, EMD, orta eksen dönüştürme, k-ortalamlar algoritması

University : **TOBB University of Economics and Technology**
Institute : **Institute of Natural and Applied Sciences**
Science Programme : **Computer Engineering**
Supervisor : **Associate Professor M. Fatih Demirci**
Degree Awarded and Date : **M.Sc. – July 2010**

Caner ERDEM

**MANY-TO-MANY FEATURE MATCHING UNDER INFINITY NORM
DISTANCE FOR OBJECT RECOGNITION**

ABSTRACT

Graph matching is an important component for many object recognition algorithms. Most of the graph matching algorithms search for one-to-one correspondence between graph vertices. Yet it is more realistic to compare a vertex subset of a graph to a vertex subset of one other graph in a many-to-many fashion. In this thesis, a similarity relation between graph vertices are presented by many-to-many matching approach. To conduct many-to-many matching on graphs, graphs are represented as point sets in normed spaces. There are some useful algorithms to compare point sets that work in polynomial time and permit many-to-many matching in a natural way. Consequently, graph matching problem is stated as that of point matching in this thesis. One of our primary objectives is to maintain isometric embedding of graphs into normed spaces. According to the results of the experiments, matching by reference points technique is developed to increase the efficiency of this approach. The results that are obtained in this manner are explained and the experimental evaluation of the algorithm is demonstrated.

Keywords: Shape recognition, object recognition, graph matching, metric embedding, EMD, medial axis transform, k-means algorithm

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren hocam Yrd. Doç. Dr. M. Fatih Demirci'ye, yine kıymetli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Bilgisayar Mühendislięi Bölümü öğretim üyelerine ve manevi desteklerinden ötürü ailem ve arkadaşlarıma teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÇİZELGELERİN LİSTESİ.....	viii
ŞEKİLLERİN LİSTESİ.....	ix
1. GİRİŞ	1
1.1. Problemin Tanımı	1
1.2. Amaç	2
2. LİTERATÜR ARAŞTIRMASI	4
2.1. Görüntü Eşleme Yaklaşımları.....	4
2.2. Çizge Gösterimi ve İlgili Terimler.....	6
2.3. Çizge Eşleme	7
2.4. Metrik Uzayda Taşıma Yöntemleri.....	8
2.5. Genel Değerlendirme	10
3. l_{∞} NORMU ALTINDA ÇOKLU EŞLEME.....	12
3.1. Orta Eksen Dönüştürme Yöntemi	12
3.2. K-ortalamlar Yöntemi	13
3.3. Çizgelerin Oluşturulması	14
3.4. Çizge Üzerinde Bir Metrik Belirlenmesi	16
3.5. Çizge Metriğinin l_{∞} Norm Uzayına Aktarılması.....	17
3.6. Norm Uzayda İki Dağılımın Karşılaştırılması	18
3.7. Deney Sonuçları.....	20
3.8. Tüm İskeletsel Noktalarla Yöntemin Denenmesi	22
3.9. Tartışma ve Yeni Yöntem.....	24
4. REFERANS NOKTALARIYLA ÇOKLU EŞLEME	26
4.1. Birinci Yöntem: K-ortalamlar Algoritmasıyla Referans Nokta Seçimi	27
4.2. İkinci Yöntem: Ağırlığı Büyük Noktaların Referans Seçilmesi	28
4.3. Üçüncü Yöntem: Ağırlığı En Büyük İki Dallanma Noktasının Referans Alınması ..	30
5. SONUÇLAR ve TARTIŞMA.....	35
5.1. Yapılan Çalışma ve Sonuçlar	35
5.2. Gelecek Çalışmalar	35
KAYNAKLAR	36
ÖZGEÇMİŞ	39

ÇİZELGELERİN LİSTESİ

Çizelge 1. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri	22
Çizelge 2. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri	23
Çizelge 3. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri	24
Çizelge 4. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri	28
Çizelge 5. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri	29
Çizelge 6. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri	30

ŞEKİLLERİN LİSTESİ

Şekil 1. 1) Çizgelerin oluşturulması, 2) Metrik uzaya taşıma, 3) Dağılımların birbirine uzaklığının hesaplanması	3
Şekil 2. Eşbiçimli iki çizge [21].....	8
Şekil 3. Bir dikdörtgenin iskeleti ve iç teğet çemberleri [32]	12
Şekil 4. a) Örnek ikili imge, b) İskeletsel noktalar, c) k-ortalamar yöntemiyle seçilmiş temsilci noktalar.....	13
Şekil 5. k-ortalamar algoritması akış şeması [33]	14
Şekil 6. a) Tüm iskelet noktaları, b) Seçilen düğümler, c) Çizgenin kenarları	15
Şekil 7. a) Örnek imge, b) İskeletsel noktalar, c) Seçilen noktaların yarıçapları.....	16
Şekil 8. Dijkstra Algoritmasının çalışması [24].....	17
Şekil 9. Üç örnek imge; (b), (a) imgesinin değiştirilmiş hali.....	21
Şekil 10. Temsili iskelet. A ve B referans noktaları. v1 ve v2 seçilen iskeletsel noktalar.....	27
Şekil 11. a) Dinozor imgesi, b)İskelet noktaları, c) Referans noktaları	28
Şekil 12. a) Veritabanından bir imge, b) Yarıçapı büyük noktalardan seçilmiş referans noktaları	29
Şekil 13. a) İkili bir imge, b) İmgenin iskeleti, c) Seçilen referans noktaları	30
Şekil 14. İmge veritabanı	31
Şekil 15. Koyu renkler imge sınıfları arasında güçlü bir benzerlik oluşunu gösterir. Köşegen üzerinde kümelenen koyu renk aynı sınıfın imgeleri arasındaki benzerliğin fazla olduğunu işaret eder.	33
Şekil 16. 16 farklı imge için elde edilen benzerlik değerleri. Düşük değerli imge çiftlerinin birbirleriyle aynı sınıftan olması beklenir. Satır ve sütunların en küçük değerleri işaretlenmiştir.....	34

1. GİRİŞ

1.1. Problemin Tanımı

İnsanlar baktıkları bir nesneyi renk, görüş açısı ve boyut gibi etkenlerden bağımsız şekilde tanıyabilirler. Tanıma işini çok hızlı ve hatasız gerçekleştirebildikleri gibi daha önce hiç görmedikleri bir nesneyi de doğru sınıflandırmakta başarılıdırlar. Bilgisayarla görünün bir alt dalı olan nesne tanımadaki düzey ise günümüzde dahi bu başarının çok gerisindedir. Konuyla ilgili araştırmaların devam ettiği nesne tanıma problemi, sorgusu yapılan bir nesnenin veri tabanındaki hangi sınıfla özdeşleştiğini belirlemek biçiminde tanımlanabilir. Bu tezde, nesne tanıma şekil tanıma terimiyle özdeş olarak kullanılmıştır.

Problemin çözümü için uygulanan genel yaklaşım; nesnelerin belirleyici özniteliklerinin çıkarılması, özniteliklerin çizgeler biçiminde ifade edilmesi ve çizgelerin birbirleriyle karşılaştırılarak benzerlik oranlarının bulunmasıdır. Bu yaklaşım yoluyla nesne karşılaştırma işlemi çizge karşılaştırma işlemine indirgenmektedir. Çizgelerin nesne tanımadaki kullanımının önemli nedenlerinden biri nesnelerin yapılarını güçlü şekilde temsil edebilmeleridir. Bir diğer neden ise farklı görüş açılarından elde edilen aynı nesnenin görüntüleri için yakın özellik göstermeleridir. Nesneler çizgeler biçiminde ifade edildiklerinde düğümler nesneden çıkarılan belirleyici öznitelikleri, kenarlar bu öznitelikler arasındaki ilişkileri yansıtır.

Çizge eşleme konusu söz konusu nedenlerden ötürü nesne tanıma probleminin merkezinde yer alır. Bu konunun ele alınmasında nesne tanıma problemine özgü bazı ayrıntılara dikkat edilmelidir. Örneğin birbirlerine çok benzer iki şeklin çizgeleri arasında bire bir eşleşmenin bulunması oldukça güçtür. Bu şu anlama gelir: bir nesnenin belirli bir parçası farklı görüş açılarından elde edilen görüntülerle oluşturulmuş çizgelerde farklı bileşimde düğüm ve kenarlarla temsil edilebilir. Bu nedenle nesne tanıma problemi ele alınırken iki çizge arasında çoklu eşlemenin aranması daha doğru sonuçlar verecektir.

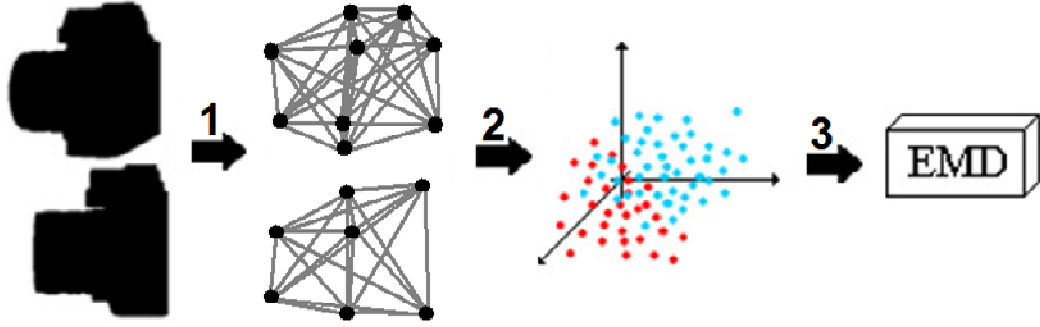
1.2. Amaç

Bu tezle yapılan çalışmanın temel amacı nesne tanıma problemi için çoklu eşlemenin yeni bir yöntem yardımıyla gerçekleştirilmesidir. Bir çizgenin bir alt kümesi başka bir çizgenin herhangi bir alt kümesiyle eşleşebilir. Bu eşleşmenin sağlanabilmesi için çizgeler metrik uzaya taşınarak nokta kümeleri biçiminde ifade edilmiştir.

Çizgelerin metrik uzaya taşınarak çoklu eşlemenin yapılmasında önemli gerekçelerden birisi nokta kümelerini polinom zamanda karşılaştırabilmemizdir. Çizge alt kümelerinin karşılaştırılması ise polinom zamanda çözülemeyen problemler sınıfına girer. Taşıma işleminin izometrik olması bu çalışmanın öncelikli hedeflerinden biri olmuştur. İzometrik bir taşımada çizge üzerinde yer alan iki düğüm arasındaki uzaklığın, metrik uzayda bu düğümlere karşılık gelen noktalar arasındaki uzaklığa eşit olması gerekmektedir. Yapılan deneylerde izometrik taşımanın eşlemede sorunlara neden olduğu görülmüş ve bu sorunları aşmak için farklı bir yöntem önerilmiştir. Bu yöntem referans noktalarıyla çoklu eşlemedir. Bir imgenin farklı görüntüleri arasında büyük değişiklikler göstermeyen noktalarını referans alma üzerine kuruludur. Çizge üzerinde belirlenen referans noktaları diğer çizge düğümlerinin norm uzayda ifade edilmesi için dayanak noktası olmuştur. Böylelikle çizgeler referans noktaları yardımıyla norm uzaya taşınarak, burada oluşturdukları dağılımlar karşılaştırılmıştır.

Ağırlıklı nokta kümelerinin metrik uzaydaki karşılaştırma işlemi Earth Mover's Distance(EMD) [1] algoritması kullanılarak yapılmıştır. EMD iki dağılım arasındaki taşıma problemini çözer ve dağılımlar arasındaki farkın derecesini belirler. Algoritmanın nasıl çalıştığı incelenecek olursa: dağılımlardan birinin kaynakları, diğerinin de bu kaynaklara olan talebi belirlediği düşünülürse; EMD talebi karşılamak için gereken taşımanın en düşük maliyetini hesaplar. Bu maliyet ne kadar fazla ise dağılımlar ve dolayısıyla ait oldukları şekiller birbirinden o derece farklıdır.

Yapmış olduğumuz çalışmanın sonuçlarını görmek için şekillerin silüetlerinden oluşan bir veritabanı ile deneyler yapılmış ve deneylerde başarılı sonuçlar elde edilmiştir. Söz konusu yöntem genel hatlarıyla Şekil 1'de tasvir edilmiştir.



Şekil 1. 1) Çizgelerin oluşturulması, 2) Metrik uzaya taşıma, 3) Dağılımların birbirine uzaklığının hesaplanması

Tezin geriye kalan kısmı şu şekilde düzenlenmiştir: 2. bölümünde nesne tanıma ve ilgili konular hakkında bilgi verilmiştir. 3. bölümde imgelerin çizgeleri oluşturulduktan sonra çizgeler üzerinde metrik bir uzaklık tanımlanmış ve çizge metriği izometrik biçimde norm uzaya taşınmıştır. Norm uzaya taşınan çizgeler çok boyutlu nokta dağılımları halini almıştır ve bu dağılımlar karşılaştırılmıştır. Yöntemin ayrıntıları, deneyler ve deney sonuçları açıklanmıştır. 4. bölümde ise izometrik taşıma işleminin çizge eşlemedeki etkisizliği referans noktalar belirlenerek aşılmıştır. Yeni geliştirilen bu yöntem, ilgili deneyler ve deney sonuçları gösterilmiştir. 5. ve son bölümde sonuçların değerlendirilmesine ve gelecekte yapılabilecek çalışmalara yer verilmiştir.

2. LİTERATÜR ARAŞTIRMASI

Bu bölümde nesne tanıma konusuyla ilgili geçmiş çalışmalara yer verilmiştir. Ayrıca kendi çalışmamız içinde kullandığımız yöntem ve araçlar hakkında ön bilgiler sunulmuştur.

İnsanların görme sistemi, bir nesneye odaklandıklarında o nesnenin iki farklı noktadan alınan görüntülerini birleştirerek tek bir görüntü halinde birleştirir. İki farklı noktadan alınan görüntüler derinlik algısının oluşmasına yardımcı olur ve insanlar nesne tanımada her durumda oldukça başarılıdır.

Nesne tanıma bu çalışmada kullanılan anlamıyla dijital olarak elde edilen iki görüntünün bir bilgisayar programıyla eşleştirilmesidir. Eşleştirme işleminde amaç görüntülerden biri üzerinde yer alan belirleyici noktaların diğer görüntüde aranmasıdır. Bu tanıma biraz daha açacak olursak; görüntü eşleme, bir bilgisayar yardımıyla belirli bir cismin farklı açılardan alınmış görüntülerinin belirleyici noktaları arasındaki benzerlik ilişkisini bulmaktır. Belirleyici noktalar gri renkli çerçeveler, imgelerden çıkarılan basit öznitelikler veya imgelerin üst düzey yapısal tanımları olabilir [2].

2.1. Görüntü Eşleme Yaklaşımları

Görüntü eşleme yaklaşımlarını sınıflandırmak için çok sayıda kıstas olmasına rağmen bunlardan en temel ve en çok benimseneni belirleyici noktaların seçimine göre sınıflandırmadır [2, 3, 4]. Belirleyici noktaların seçimi eşleme algoritmalarına yön veren en önemli unsurdur. Görüntü eşleme süreçleri başlangıçta seçilen yaklaşıma göre gelişir.

Görüntü eşleme yaklaşımlarını sınıflandırmada kullanılan belirleyici noktalar şunlardır: gri renkli çerçeveler, imgelerden çıkarılan basit öznitelikler ve yapısal imge tanımları. Bunların seçimine göre benimsenen üç temel yaklaşım sırasıyla alan-tabanlı eşleme, öznitelik-tabanlı eşleme ve yapısal-tabanlı eşlemedir [2, 7]. Bu yaklaşımların altında yer alan çok sayıda farklı yöntem ve algoritma bulunmaktadır. Bu yöntemlerin verimliliği eşlenecek olan görüntünün seçimine göre farklılık

gösterir. Söz konusu temel üç yaklaşımın dışında melez yaklaşımlar da bulunmaktadır.

Alan-tabanlı eşleme: Alan-tabanlı yaklaşım belirleyici nokta olarak gri renkli piksel çerçevelerini kullanır. Bu yaklaşımın avantajları arasında algoritmalarının sadeliği ve donanım tabanlı gerçekleştirmelerinin kolay olması sayılabilir. En önemli dezavantajı ise görüş açısı değişikliklerine karşı hassas olmasıdır [2].

Bu yaklaşım bir görüntü üstünde yer alan belirli bir çerçevenin çevrelediği piksel grubunun başka bir görüntüde aranmasına dayanır. En iyi durumda bir görüntü üstünde yer alan çerçevenin başka bir görüntünün üzerinde aynen yakalanması gereklidir. Ancak bunun olasılığı çok düşüktür ve iki görüntü arasındaki çerçeveler eşleştirilirken bir benzerlik oranı bulunmasına çalışılır [4].

Bu yaklaşımın uygulanmasına basit bir örnek verecek olursak: bir görüntünün belirli bir konumundan karesel bir çerçeve seçilir ve bu karesel çerçeve içinde yer alan piksel değerleri ikinci bir görüntünün aynı konumdaki çerçevesiyle kıyaslanır. Kıyaslamaya ikinci görüntüdeki çerçevenin yakın çevresiyle devam edilir. Örneğin 5×5 piksellik bir çerçeve ikinci görüntüde 9×9 piksellik bir bölgede aranır. Bire bir eşleşme olasılığı düşük olduğu için iki çerçeve arasındaki ilinti oranına bakılarak benzerliğin derecesi tespit edilir. Pratikte daha karmaşık algoritmalar kullanılmaktadır.

Öznitelik-tabanlı eşleme: Öznitelik-tabanlı yaklaşım imgelerden çıkarılan basit öznitelikleri kullanır. İlk aşamada nokta, köşe, kenar gibi öznitelikler bir program yardımıyla elde edilir. İkinci aşamada elde edilen bu öznitelikler ölçülerine ve uyumluluklarına göre karşılaştırılarak eşleme işlemi tamamlanır.

İlk aşamanın ardından elde edilen özniteliklerden oluşan ve özniteliklerin konum bilgisini de barındıran bir listedir. Bu bir imgenin soyut bir ifadesi biçiminde düşünülebilir. İkinci aşamada birinci aşamada elde edilen listeler karşılaştırılarak iki imge arasındaki benzerlik düzeyi elde edilmeye çalışılır.

Çıkarılan özniteliklerin komşularına göre ayırıcı nitelikte olması, ışık ve gürültü gibi etkenlerden etkilenmemesi ve diğer özniteliklerden mümkün olduğu kadar farklı olması gerekir [8].

Özniteliklerin eşlenmesi bir adım daha ileri götürülerek imgelerden çıkarılan öznitelikler yapısal imge tanımları oluşturmada kullanılmıştır.

Yapısal-tabanlı eşleme: Yapısal-tabanlı eşleme yaklaşımında imgelerden çıkarılan yapısal tanımlar arasında benzerlik ilişkisi aranır [9, 10]. Yapısal tanımlar imgelerden çıkarılan noktalar, çizgiler ve alanlar gibi özniteliklerden ve bunların birbirleriyle olan ilişkilerinden oluşur [11]. Bu yaklaşımla imgenin belirleyici noktalarından oluşan iki listenin karşılaştırılması yerine imgeler bir bütün olarak karşılaştırılmaktadır.

Yapısal tanımlar oluşturulurken imgenin belirli bir bölümünü farklı bileşimdeki öznitelikler ve öznitelikler arasındaki ilişkiler temsil edebilir. Bir ilişki geometrik ve topolojik olarak ifade edilebilir: örneğin geometrik bir ilişki iki kenar arasındaki mesafeyi veya iki çizgi arasındaki açıyı tespit ederken, topolojik bir ilişki hiyerarşik bir temsilde çıkarılan iki öznitelikten hangisinin diğerinin atası veya yavrusu olduğunu tespit eder.

Yapısal-tabanlı eşleme imgelerden çıkarılan özniteliklerin arasındaki ilişkileri de göz önünde bulunduran bir yaklaşımdır. Bizim çalışmamızda bu yaklaşım benimsenmiştir.

2.2. Çizge Gösterimi ve İlgili Terimler

Sonlu bir çizge (V, E) çifti ile gösterilir: V sonlu sayıdaki düğüm kümesi iken E bu düğümler arasındaki kenarları ifade eder. $(u, v \in V)$ olmak üzere bir kenar $e = (u, v)$, iki düğümden oluşur. Bir çizge $G = (E, V)$, tüm kenarlarının $e \in E$ bir ağırlık değeri $w(e) \in \mathbb{R}$ varsa kenar-ağırlıklı çizge olarak adlandırılır. Bir çizgenin boyutu kenar $|E|$ ve düğüm $|V|$ sayıları ile ifade edilir. Kenarların bağlı oldukları düğümler arasındaki ilişkinin yönlü olup olmamasına göre çizgeler yönlü-çizge ve yönsüz-çizge olarak iki sınıfa ayrılır. Yönlü bir çizgede $(u, v) \neq (v, u)$ ilişkisi sağlanır. Yönlü çizgeler genel olarak asimetric ilişkileri belirtmekte kullanılır. Bu tezdeki

çalışmada yönsüz çizgeler kullanılmıştır. Eğer bir çizgenin $G = (E, V)$, tüm düğümleri arasında $(u, v \in V)$, $u \neq v$ ilişkisi sağlanmak üzere bir kenar varsa $(u, v) \in E$, bu çizge tam çizge olarak adlandırılır. Çizgelerdeki düğümler ve kenarlar kimi zaman birden fazla sayıda bilgi ile yüklenebilirler.

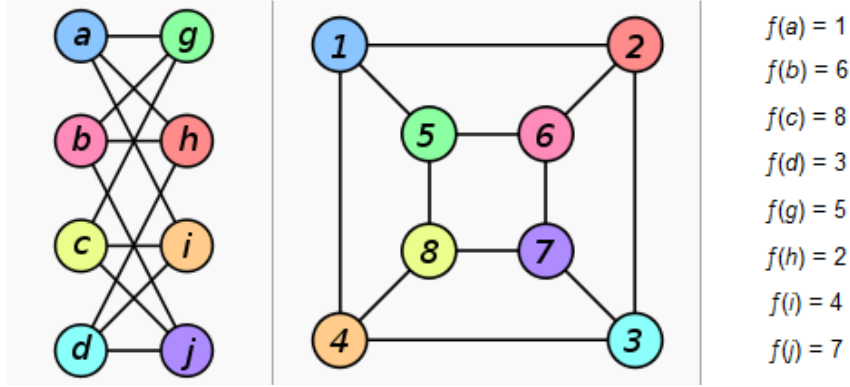
Çizgelerin nesnelere betimlemekte önemli araçlar olduğu bilinmektedir. Bilgisayarla görme dışında moleküler biyoloji, dilbilim, bilgisayar ağları gibi alanlarda çizgelerden yararlanılmaktadır. Çizgeler nesnelere temsil etmekte kullanıldıklarında düğümler genellikle öznitelikleri kenarlar ise bu öznitelikler arasındaki ilişkiyi gösterirler.

Çok sayıda araştırmacı çizgeleri imgelerin yapısal özelliklerini temsil etmekte kullanmışlardır. Dickinson [13] ve Cyr [14] çizgelerden üç boyutlu nesnelere temsil etmek için yararlanırken, bir başka çalışmada [15] bir çizge alt türü olan ağaçlar insanları modellemek için kullanılmıştır.

2.3. Çizge Eşleme

İki imgeden elde edilen çizgelerin eşlenmesiyle imgeler arasındaki benzerliğin bulunması hakkında çok sayıda çalışma yapılmıştır. Bu çalışma içinde de sorgusu yapılan imgeler ve model imgeler çizgeler biçiminde ifade edilmiştir. Nesne tanıma problemi çizge eşleme problemi olarak ele alınmıştır. Çizge eşleme bilgisayarla görme dışındaki alanlarda da kullanılmıştır: moleküler biyoloji [19], bilgisayar ağları [20].

Bire bir çizge eşleşmesi çizge eşbiçimliliği olarak tanımlanmıştır. $G = (V, E)$ ve $G' = (V', E')$ iki çizge olmak üzere, $(u, v) \in E$ ancak ve ancak $(f(u), f(v)) \in E'$ önermesi için bir fonksiyon $f: V \rightarrow V'$ tanımlanabiliyorsa, bu çizgeler eşbiçimlidir (Bkz. Şekil 2). Ayrıca, G' çizgesi G'' çizgesinin bir alt çizgesi ise G çizgesi ile G'' çizgesi arasında alt-çizge eşbiçimliliği vardır.



Şekil 2. Eşbiçimli iki çizge [21]

Çizge eşbiçimliliği nesne tanımda önemli bir konudur. Örneğin iki çizgenin eşbiçimli ortak alt çizgesi ne ölçüde büyükse ilgili imgeler o ölçüde benzerdir [22]. Gerçek uygulamalarda elde edilen imgeler gürültü gibi nesne tanımayı güçleştiren özellikler taşır. Bu nedenle çizgelerdeki benzerlikler imgelerdeki bozulmalar göz önüne alınarak değerlendirilmelidir.

Nesne tanıma uygulamalarında çizge eşbiçimliliği gibi bire bir eşleme yerine çoklu eşlemenin aranması daha uygundur. Bu yönde kullanılan yöntemlerden biri çizge düzeltme uzaklığıdır [25]. Yöntem bir çizgenin, düğüm ve kenar gibi elemanlarını silme, ekleme, yer değiştirme gibi değişikliklerden sonra diğer bir çizgeye dönüştürme üzerine kuruludur. Her değişikliğin ilgili bozulmanın olma ihtimalini göz önüne alan bir maliyeti vardır. Çizgelerden birini diğerine dönüştürmenin toplam maliyeti ne kadar azsa imgeler o kadar birbirine yakındır.

Çizge düzeltme uzaklığı güçlü bir yöntem olmasına karşın birtakım olumsuz yanları vardır. Bunlar arasında hesaplama maliyetinin fazla olması ve güncel algoritmaların kenar ağırlıklarını göz önüne almaması sayılabilir.

2.4. Metrik Uzayda Taşıma Yöntemleri

Teorik bilgisayar biliminde metrik uzaydaki taşıma işlemlerini en az bozulma ile yapmanın yolları aranmaktadır. Taşıma işleminin çizge tabanlı birtakım problemlerin çözümünde kolaylıklar sağladığı görülmüştür. Düşük bozulma ile yapılan çizge taşımayla ilgili en son gelişmelere Indyk'in çalışmasında [26] yer verilmiştir.

Bir X kümesi üzerinde tanımlı $d: X \times X \rightarrow \mathbb{R}$ uzaklık fonksiyonu, aşağıda belirtilen koşulları sağladığında bir metriktir:

- $d(x, y) = 0$ ancak ve ancak $x = y$ ise,
- $d(x, y) = d(y, x)$,
- $d(x, z) \leq d(x, y) + d(y, z)$, $\forall x, y, z \in X$.

Metriklerle çizgeler arasında doğal bir ilişki kurulabilir. Örneğin kenar ağırlıkları verilen bir G çizgesi için metrik bir uzaklık $d_G(i, j)$; $i, j \in V(G)$ olmak üzere i ve j düğümleri arasındaki en kısa yol olarak tanımlanabilir. Tersini düşünülürse verilen bir (X, d) metriği için kenar ağırlıklı bir $G(d)$ çizgesi oluşturulabilir. Bu durumda noktalar çizge düğümlerini belirlerken, noktalar arası uzaklık da çizge kenarlarının ağırlığını belirler.

Norm uzaylar k boyutlu reel uzayı \mathbb{R}^k , Minkowski normlarından l_p biriyle donatarak elde edilir, $1 \leq p \leq \infty$. Verilen bir $x \in \mathbb{R}^k$ noktasının l_p uzaklığı şu biçimde bulunur:

$$\|x\|_p = (\sum_{i=1}^k |x_i|^p)^{1/p} \quad 1 \leq p < \infty.$$

Sonsuz norm uzaklığı l_∞ ise $\|x\|_\infty = \max_{i=1}^k \{|x_i|\}$ olarak hesaplanır. Uzayda verilen iki nokta $x, y \in \mathbb{R}^k$ için, aralarındaki l_p norm uzaklığı $\|x - y\|_p$ olarak ifade edilir.

Bu norm uzaylardan l_2 Öklid uzayını ifade eder. Yapmış olduğumuz çalışmada l_∞ norm uzayına yapılan izometrik çizge taşıma işlemi önemli yer tutmaktadır. Eğer sonlu bir metrik (X, d) için $x, y \in X$ ve $\|f(x) - f(y)\|_p = d(x, y)$ olmak üzere $f: X \rightarrow \mathbb{R}^k$ biçimde bir fonksiyon tanımlanabiliyorsa, bu metrik l_p -metrik olarak adlandırılır. Metrik uzayın k boyutlu olduğu düşünüldüğünde l_p^k biçiminde ifade edilebilir.

Verilen bilgiler doğrultusunda (X, d) ve (X', d') metrik uzaylarının $f: X \rightarrow X'$ biçiminde haritalanması [27] metrik uzayda taşıma işlemine karşılık gelir. Eğer bu taşıma işlemi uzaklıkları değişmeden koruyorsa izometriktir.

Metrik uzaylar zengin matematiksel içeriklerinin yanında algoritmalar yönünden de arařtırmacıların ilgisini çekmektedir. Örneđin çokça bilinen Dolařan Satıř Elemanı problemi bilinen řehirler ve řehirlerarası uzaklıklar için bu řehirlerin hepsinden tam olarak bir kez geçen en kısa yolu bulmaya dayanır. Problemin çözümlü için bütün olası yollar hesaplanarak en kısa yol bulunabilir; ancak oldukça maliyetli bir hesaplamadır. Diđer bir çözümlü ilgili metriđin bir ađaç yapısına tařınması ve ideale yakın bir yol bulunmasıdır. Metrik uzayda tařıma yöntemleri ařađıdaki temel bařlıklar altında deđerlendirilebilir.

Genel metriklerin düşük boyutlu uzaylara tařınması: Amaç orijinal metriđin düşük boyutlu bir temsilini elde etmektir. Orijinal noktaların daha az sayıda ancak anlamlı veriyle ifade edilmesini sađlar.

Genel metriklerin ađaç metriklerine tařınması: Metriklerin norm uzaylarına tařınması yerine ađaç metriđine tařınmasıdır. Birçok arařtırmacı polinom zamanda çözümlenmeyen problemler için bu yöntemi kullanır. Çođunlukla aynı problemlerin ađaçlar için polinom zamanlı çözümleri vardır ve ana problem için yakınsama algoritmaları geliřtirmeye yardımcı olur.

Ađaç metriklerinin düşük boyutlu norm uzaylara tařınması: Birtakım algoritmalar ađaç metrikleri üzerinde çalıřmak yerine, bunların temsil edildiđi düşük boyutlu norm uzayları tercih eder. Norm uzayın boyutu kimi zaman önceden belirlenebilirken kimi zaman ađacın yapısına göre oluşur. Ađaç metriklerinin tařınmasıyla ilgili Gupta'nın çalıřmasına [29] bakılabilir.

Özel metriklerin norm uzaylara tařınması: Kimi özel metriklerin, örneđin düzeltme uzaklıđı, norm uzaylara tařınarak bilinen algoritmalar yardımıyla orijinal metrikteki problemlere çözümlü aranmasıdır [30].

2.5. Genel Deđerlendirme

Çizgeler nesnelerin temsil edilmesinde güçlü araçlardır. Nesnelere çizgeler olarak temsil edildiđinde nesne tanıma problemi çizge eřleme problemine dönüşür. Çizgeleri bire bir eřlemek pratik zorluklardan dolayı kullanıřlı deđildir. Bunun yerine

izgelerin farklı bileşimdeki alt kümelerinin eşleşebileceđi göz önüne alınarak çoklu eşleme yaklaşımı benimsenmiştir.

izgelerin çoklu eşlenmesi düzeltme uzaklıđı yönteminde göz ününde bulundurulmuştur. Bu yöntemin amacı bir çizgenin başka bir çizgeye dönüşmesi için yapılması gereken ekleme, silme, birleştirme, ayırma, yeniden adlandırma gibi işlemlerin en az sayıda yapılmasıdır. Yapılan düzeltmelerin maliyeti çizgeler arasındaki benzerliđin ölçüsüdür. Yöntem çoklu eşleme adına önemli özellikler taşımasına rağmen güncel haliyle bazı pratik eksikleri vardır.

izge metriđinin norm uzaya taşınarak çoklu eşleme yapılması yeni bir yöntem değildir. Bu alanda yapılmış çalışmalardan birinde çizge metriđi norm uzaya taşınarak ilgili çizgeler arasındaki benzerlik aranmıştır [6]. Ancak yapılan taşıma işlemi izometrik değildir, diđer bir deyişle çizge düğümleri arasındaki uzaklık norm uzayda bozulmaktadır. Bizim çalışmamızda metrik taşıma işleminin izometrik olarak sağlanması ve çoklu eşlemenin daha iyi sonuçlarla gerçekleştirilmesi hedeflenmiştir.

Hızlı ve güvenilir bir çoklu eşleme yaklaşımı ucu açık araştırma konularından biridir. Bu tezdeki amacımız çizgeler arasında çoklu eşlemenin söz konusu kıstaslarla sağlanabilmesidir.

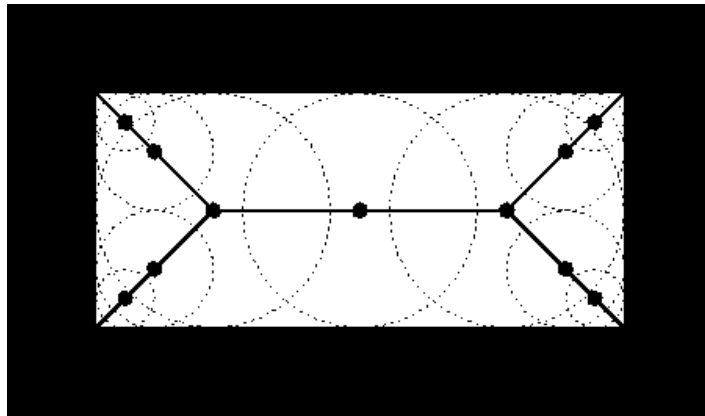
3. l_{∞} NORMU ALTINDA ÇOKLU EŞLEME

Bu tez çalışmasının ilk hedefi nesne tanıma problemi için çoklu eşlemenin l_{∞} normu altında gerçekleştirilmesi olmuştur. Bu hedefe ulaşmak için yapılan çalışmalar ve çalışmalara dayanak teşkil eden teorik alt yapıya bu bölümde yer verilmiştir.

3.1. Orta Eksen Dönüştürme Yöntemi

Orta eksen dönüştürme yöntemi karşılaştıracağımız imgelerin çizgelerini oluşturmak için kullanılmıştır. Bu yöntemle verilen bir ikili imgenin iskeletsel noktaları ve bu noktaların elde edilmesini sağlayan dairelerin yarıçapları elde edilmiştir. İkili imgeler her piksel için yalnız iki değer alabilen imgelerdir. Siyah veya beyaz renkli bir piksel grubu imgenin kendisini betimlerken geri kalan diğer pikseller imgenin arka planını oluşturur.

Orta eksen dönüştürme yöntemi ilk kez 1967 yılında Blum tarafından biyolojik şekillerin tanınabilmesi için ortaya konulmuştur [31]. Orta eksen imgelerin sınırlarına en az iki teğet noktası olan dairelerin merkezlerinin birleşmesiyle oluşur. Topolojik iskelet olarak da adlandırılır. Dairelerin yarıçaplarıyla beraber iskelet üzerindeki noktalar şeklin orijinal halini yeniden oluşturmak için yeterlidir (Şekil 3). İskeletler bağlanırlık, uzunluk, ardıllık, yön ve genişlik gibi bir şeklin topolojik ve geometrik özelliklerini taşırlar.



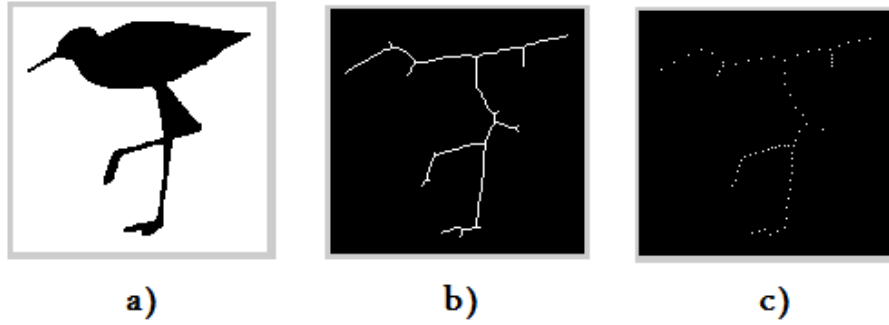
Şekil 3. Bir dikdörtgenin iskeleti ve iç teğet çemberleri [32]

Elimizdeki imgelerden çıkarılan iskeletsel noktalar ve bu noktaları belirleyen yarıçaplar uygulamanın sonraki bölümlerinde imgelerin çizgelerini oluşturmakta

kullanılmıştır. İskeletsel noktalar iki boyutlu düzlemdeki koordinatları ve yarıçaplarıyla beraber saklanmaktadır.

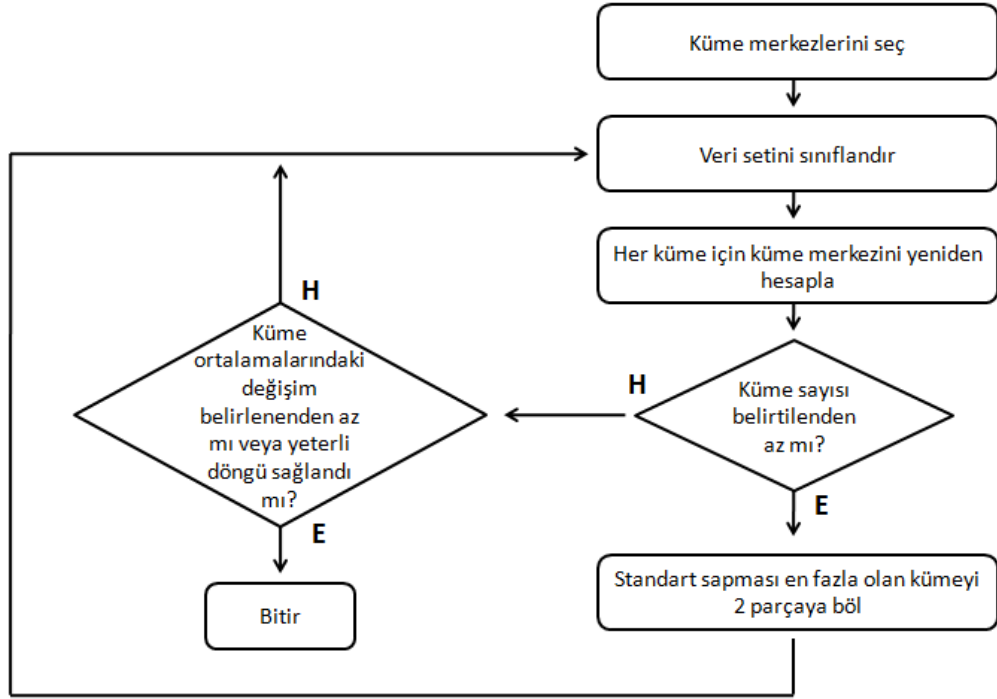
3.2. K-ortalamlar Yöntemi

Bir önceki adımda elde edilen iskeletsel noktaların tümünü uygulamada kullanmanın hesaplama başarımı açısından fazladan yük getireceği düşünülmüş ve k-ortalamlar yöntemiyle iskelet üzerinde temsilci noktalar seçilmiştir. Bu temsilci noktaların iskeletin genel yapısını iyi yansıtması ve bilgi kaybı olmaması gerekmektedir. Yöntemin ayrıntılarına geçmeden önce örnek bir şekil için sonuçlar incelenebilir (Şekil 4). Şekilde (a) bir imge, (b) bu imgeden orta eksen dönüştürme yöntemiyle elde edilmiş iskeletsel noktalar ve (c) k-ortalamlar yöntemiyle belirlenmiş temsilci iskeletsel noktalar.



Şekil 4. a) Örnek ikili imge, b) İskeletsel noktalar, c) k-ortalamlar yöntemiyle seçilmiş temsilci noktalar

K-ortalamlar yöntemi veri analizinde verilerin nasıl kümelendiğini gözlemlemek için kullanılan bir yöntemdir. Bir ortalama etrafında verileri gruplandırma, ardından bir grup oluşturan verilerin ortalamasını bulma olarak döngüsel yürütülen iki adımdan oluşur. Algoritma grup içi ortalamadan sapmalar önceden belirlenen bir seviyenin altına indiğinde sonlanır. İlk başta veri seti için küme merkezleri rastgele belirlenir ve algoritma başlatılır. Algoritmanın akış şeması Şekil 5’te verilmiştir.



Şekil 5. k-ortalamlar algoritması akış şeması [33]

K-ortalamlar algoritmasının her tekrarında küme içi standart sapmalar azalır ve sonuçlar iyileşir. Bir verinin hangi kümeye gireceğini benzerlik ölçütü belirler. Bizim çalışmamızda 2 boyutlu düzlem için iskelet üzerindeki noktaların benzerlik ölçütü Öklid uzaklığıdır.

K-ortalamlar algoritmasının veri seti dışındaki tek parametresi küme sayısıdır. Uygulamamızda küme sayısı iskelet noktaları sayısının 5'te 1'i olarak belirlenmiştir. Algoritma tamamlandığında belirlenen sayıda küme merkezinin koordinatları öğrenilmiş olur. Bu aşamadan sonra küme merkezlerine en yakın iskelet noktaları seçilmiş ve çizge oluşturmak için kullanılmıştır (Şekil 4 - c).

3.3. Çizgelerin Oluşturulması

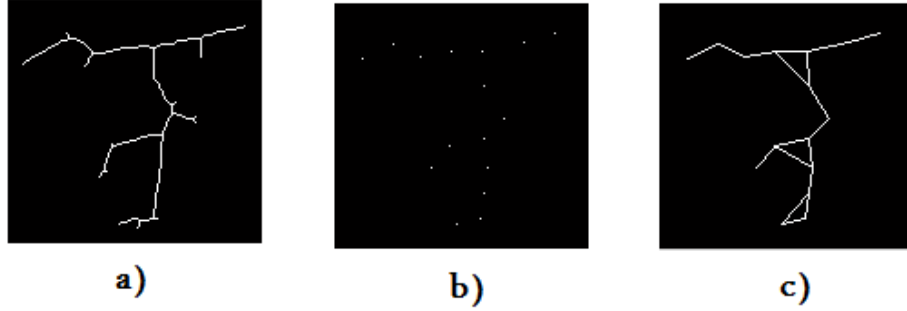
Bir önceki adımda elde edilen temsilci noktalar oluşturulan çizgelerin düğümlerini temsil etmektedir. Hangi düğümler arasında kenar olması gerektiğine bunların arasındaki uzaklıkların istatistiklerine bakarak karar verilmiştir. Bütün noktaların

sayısı n olarak kabul edilirse $\binom{n}{2}$ sayıda uzaklığın ortalaması ve standart sapmasıyla bir eşik değeri oluşturulmuştur:

$$\text{Eşik değeri} = \text{ortalama} - 1,25 \times \text{standart sapma}$$

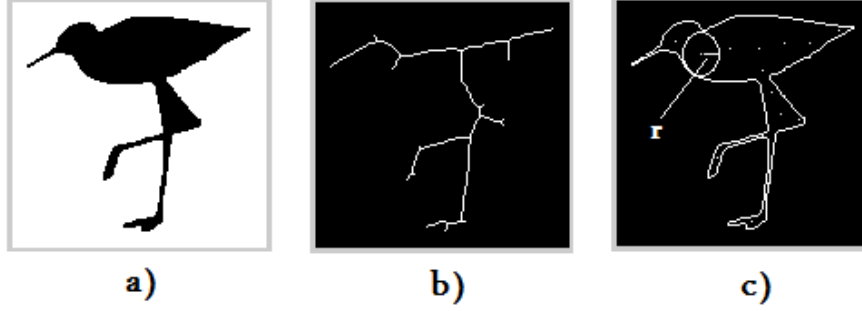
İki nokta arasındaki uzaklık bu eşik değerinin altında ise çizgede bu iki noktaya karşılık gelen düğümler arasına bir kenar yerleştirilmiştir. Kenarların ağırlıkları da bağlandıkları düğümler arasındaki Öklid uzaklığı olarak belirlenmiştir. Örneğin Şekil 6'da 3. imge bu kurala uygun olarak belirtilmiş kenarlardır. Eşik değeri yaptığımız çalışmada genellikle bütün düğümleri çizgeye bağlayacak bir üst sınır oluşturmuştur. Eğer noktalardan bir veya daha fazlası bu eşik değerinin üstünde kalarak çizgeye bağlanmaz ise eşik değeri güncellenmiştir:

$$\text{Eşik değeri} = \text{eşik değeri} + 0,25 \times \text{standart sapma}$$



Şekil 6. a) Tüm iskelet noktaları, b) Seçilen düğümler, c) Çizgenin kenarları

Çizgenin düğüm ve kenarlarını belirledikten sonra düğümlerin ağırlıkları belirlenmiştir. Bu ağırlıklar iskelet noktalarını oluşturan iç teğet çemberlerin yarıçapları olarak hesaplanmıştır (Şekil 7 - c).



Şekil 7. a) Örnek imge, b) İskeletsel noktalar, c) Seçilen noktaların yarıçapları

3.4. Çizge Üzerinde Bir Metrik Belirlenmesi

Tezin 2. bölümünde belirtildiği gibi bir çizge için iki düğüm arasındaki en kısa yol metrik bir uzaklıktır. Böylelikle bir çizge metrik uzayda ifade edilebilir. Çalışmamızda çizgelerin düğümleri arasındaki en kısa yolları bulmak için Dijkstra algoritması kullanılmıştır [28].

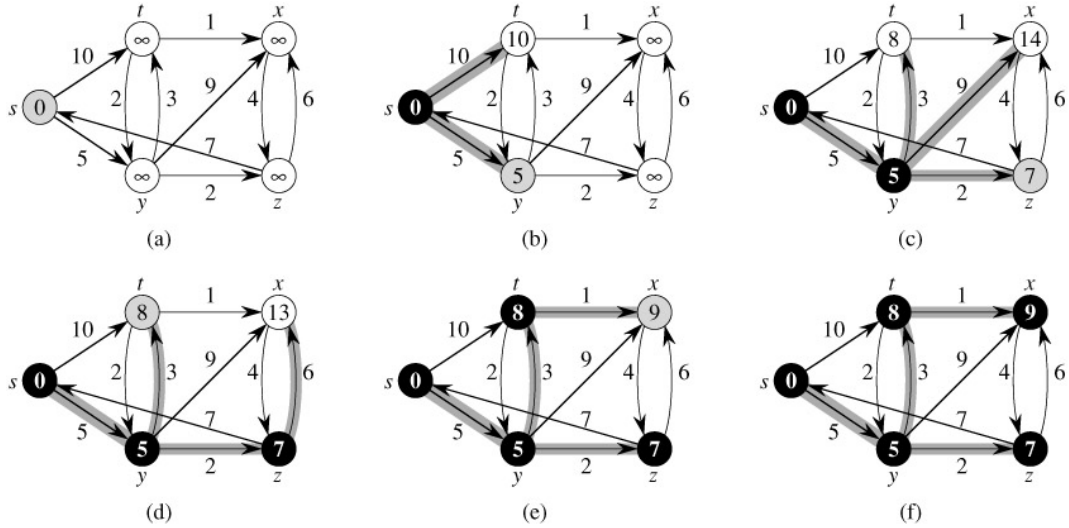
Dijkstra algoritması tek-kaynaklı en kısa yol problemini kenar ağırlıklı, yönlü veya yönsüz bir $G = (V, E)$ çizgesi için çözer. Algoritmanın doğru çalışabilmesi için kenar ağırlıklarının negatif değerler almaması gerekir: $w(u, v) \geq 0, \forall (u, v) \in E$.

Algoritma kaynak düğümden başlayarak bütün diğer düğümlere olan en kısa yolları bulur. Bir hedef düğüme olan en kısa yol arandığında algoritmanın bütün düğümler için çalışmasına gerek olmayabilir; hedef düğüme olan en kısa yol bulunduğu anda algoritma sonlanır. Algoritmanın adımları şöyledir:

1. Bütün düğümlere kaynak düğümden bir başlangıç uzaklığı atanır: kaynak düğüme sıfır diğer tüm düğümlere sonsuz uzaklık.
2. Bütün düğümler ziyaret edilmemiş olarak işaretlenir. Kaynak düğüm güncel düğüm olarak işaretlenir.
3. Güncel düğümün ziyaret edilmemiş tüm komşularına geçici uzaklıkları atanır. Eğer komşu düğümün uzaklığı güncel düğümün uzaklığıyla aralarındaki kenarın ağırlığı toplamından fazlaysa komşunun uzaklığı güncellenir.
4. Güncel düğümün tüm komşuları gezildikten sonra, güncel düğüm ziyaret edilmiş olarak işaretler ve bir daha kontrol edilmez. Bu durumda güncel düğümün uzaklığı son ve kaynaktan olan en kısa uzaklıktır.

- Eğer bütün düğümler ziyaret edildiyse algoritma sonlanır. Aksi takdirde uzaklık değeri en düşük olan ziyaret edilmemiş düğüm yeni güncel düğüm olarak belirlenir ve 3. adımdan devam edilir.

Dijkstra algoritmasının yönlü bir çizgede nasıl ilerlediğini görmek için Şekil 8'e bakabiliriz. Bu örnekte kaynak düğüm s ile gösterilmiştir ve tüm diğer düğümlere olan uzaklıklar ve yollar bulunmuştur. Yolların gösterilmesi için düğümlerin atalarıyla olan kenar bağlantıları gölgelenmiştir: t düğümünün atası ilk adımda ağırlığı 10 olan kenar iken ikinci adımda ağırlığı 3 olan kenar olmuştur ve sonrasında bu düğüme daha kısa bir yol bulunamadığı için değişim olmamıştır. Her adımda siyahla işaretlenmemiş (ziyaret edilmemiş) düğümlerden uzaklığı en düşük olan seçilir ve gri renk ile işaretlenerek komşularının uzaklığı güncellenir.



Şekil 8. Dijkstra Algoritmasının çalışması [24]

Sonuç olarak, imgelerin çizgeleri üzerinde en kısa yol uzaklığı ile bir metrik tanımlandıktan sonra bu çizgeler metrik uzaya taşınabilirler.

3.5. Çizge Metriğinin l_∞ Norm Uzayına Aktarılması

Fréchet'in teoremine göre n sayıda nokta içeren bir metrik uzay (X, d) , izometrik biçimde l_∞^n norm uzayına taşınabilir [23].

İspat: $X = (x_1, \dots, x_n)$ olmak üzere (X, d) bir metrik uzay olsun. (X, d) uzayını $(\mathbb{R}_n, \|\cdot\|)$ norm uzayına izometrik şekilde taşıyabilen bir $f: X \rightarrow \mathbb{R}_n$ fonksiyonu

bulmak yeterli olacaktır. $x_i \in X$ olmak üzere f fonksiyonu aşağıdaki şekilde tanımlanabilir:

$$f(x_i) = (d(x_1, x_i), d(x_2, x_i), \dots, d(x_n, x_i))$$

Bu fonksiyonun $\|f(x_i) - f(x_j)\|_\infty = d(x_i, x_j)$ eşitsizliğini her $x_i, x_j \in X$ için sağladığını göstermek gereklidir. Öncelikle, d bir metrik olduğuna göre $k = 1, \dots, n$ için üçgen eşitsizliğini, $d(x_i, x_k) - d(x_j, x_k) \leq d(x_i, x_j)$, sağlaması gerekir. Buna göre:

$\max_k |d(x_i, x_k) - d(x_j, x_k)| \leq d(x_i, x_j)$, yargısı çıkarılabilir. Bu şu biçimde de yazılabilir:

$$(1) \|f(x_i) - f(x_j)\|_\infty \leq d(x_i, x_j)$$

Öte yandan, $f(x_i) - f(x_j)$ vektörünün j . koordinatı $d(x_j, x_i) - d(x_j, x_j)$ işleminin sonucuna eşittir. O halde, $f(x_i) - f(x_j)$ vektörünün en büyük koordinatı en az $d(x_i, x_j)$ kadar olmalıdır:

$$(2) \|f(x_i) - f(x_j)\|_\infty \geq d(x_i, x_j)$$

Teorem (1) ve (2) numaralı yargılardan ispat edilmiş olur.

Bu teorem yardımıyla elimizdeki çizge metriği n boyutlu sonsuz norm uzayına, l_∞^n , taşınmıştır. Diğer bir deyişle, çizgelerimiz norm uzayda ağırlıklı birer nokta kümesi haline getirilmiştir. Her nokta bir ağırlık ve koordinatlardan oluşan bir vektör ile temsil edilir. Bu taşıma işlemi çizgelerdeki düğümlerin arasındaki uzaklıkları koruyarak norm uzaydaki algoritmalarından yararlanma olanağı vermiştir. Geline aşamada, nesne tanıma problemi norm uzaydaki iki dağılımın karşılaştırılması problemine dönüşür.

3.6. Norm Uzayda İki Dağılımın Karşılaştırılması

Önceki bölümlerde anlatılan adımlardan geçen bir imge norm uzayda bir dağılım halini almıştır. İki imgenin karşılaştırılması için norm uzaydaki dağılımları

arasındaki farkı bulmamız gerekmektedir. Bu farkı bulmak için Earth Mover's Distance(EMD) algoritması kullanılmıştır [18].

Bu algoritmanın çalışabilmesi için dağılımlardaki herhangi iki nokta arasındaki uzaklığı belirleyen bir ölçü bulunmalıdır. Çizgelerimizi taşıdığımız sonsuz norm uzayında bu ölçü iki noktanın koordinatları arasındaki en büyük uzaklıktır. $(x_{i1}, x_{i2}, \dots, x_{in}) \in x_i$ ve $(y_{i1}, y_{i2}, \dots, y_{in}) \in y_i$ iki nokta olmak üzere ararındaki uzaklık:

$$\|x_i - y_i\|_\infty = \max((x_{i1} - y_{i1}), (x_{i2} - y_{i2}), \dots, (x_{in} - y_{in})) \quad (3.1)$$

EMD iki nokta arasındaki bu uzaklık ölçüsünü kullanarak iki dağılım arasındaki uzaklığın derecesini belirler. Algoritmanın ismiyle uyumlu düşünersek, karşılaştıracağımız dağılımlardan birisi belirli bir bölgeye dağılmış toprak kümelerini, diğer dağılım bu toprağın taşınabileceği çukurları ifade eder. EMD bu taşımanın gerçekleşmesi için gereken en düşük maliyeti hesaplar.

Bizim yöntemimizde dağılımlardan her birisi farklı sayıda nokta kümesinden oluşur. Bu noktaların koordinatları ve ağırlıkları vardır. Dağılımlardan herhangi birinin toplam ağırlığı diğerinden fazla olabilir; bu durum algoritmanın çalışmasını engellememektedir. EMD algoritması iyi bilinen taşımacılık problemlerinden birinin çözümü üstüne kuruludur [17]. Bu problemde belirli sayıda tedarikçi farklı miktarda ürüne sahiptir; bu ürünlere farklı ölçülerde ihtiyacı olan tüketiciler vardır. Taşıma problemi en az maliyetli ürün akışını hesaplama üzerine kuruludur.

EMD doğrusal programlama problemi olarak şu şekilde ifade edilir: $P = \{(p_1, w_{p_1}), \dots, (p_m, w_{p_m})\}$ m noktadan oluşan birinci dağılım, p_i noktanın koordinatları, w_{p_i} noktanın ağırlığıdır. $Q = \{(q_1, w_{q_1}), \dots, (q_n, w_{q_n})\}$ n sayıda noktadan oluşan ikinci dağılım ve d_{ij} p_i ve q_i noktaları arasındaki uzaklıktır.

f_{ij} p_i ve q_j arasındaki akış olmak üzere, aşağıdaki amaç fonksiyonunu en düşük yapacak $F = [f_{ij}]$ vektörü bulunmaya çalışılır.

$$MALİYET(P, Q, F) = \sum_{i=1}^m \sum_{j=1}^n f_{ij} d_{ij}$$

Kısıtlar:

$$f_{ij} \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n$$

$$\sum_{j=1}^n f_{ij} \leq w_{p_i} \quad 1 \leq i \leq m$$

$$\sum_{i=1}^m f_{ij} \leq w_{q_j} \quad 1 \leq j \leq n$$

$$\sum_{i=1}^m \sum_{j=1}^n f_{ij} = \min (\sum_{i=1}^m w_{p_i}, \sum_{j=1}^n w_{q_j})$$

Birinci kısıt kaynakların P dağılımından Q dağılımına taşınacağını belirler. İkinci kısıt p_i noktasından çıkan akışı p_i noktasının ağırlığıyla ve üçüncüsü q_j noktasına giren akışı q_j noktasının ağırlığıyla sınırlar. Dördüncü kısıt ise mümkün olan en çok taşımanın yapılması içindir. Taşıma problemi çözüldüğünde iki dağılım arasındaki uzaklık toplam maliyetin toplam akışa bölünerek düzelenmesiyle (normalizasyon)[16] bulunur:

$$EMD(P, Q) = \frac{\sum_{i=1}^m \sum_{j=1}^n f_{ij} d_{ij}}{\sum_{i=1}^m \sum_{j=1}^n f_{ij}}$$

Düzgeleme küçük boyutlu dağılımlar arasındaki farkın ufak gözükmemesinin önüne geçmek için yapılmaktadır.

Şimdiye kadar anlatılan adımlardan geçen bir imge özetle önce bir çizgeye dönüştürülmüş ardından bu çizge üzerinde bir metrik tanımlanarak norm uzaya taşınmıştır. Norm uzayda yapılan karşılaştırmaların nesne tanıma probleminin çözümüne yönelik katkıları bundan sonraki adımlarda tartışılmıştır.

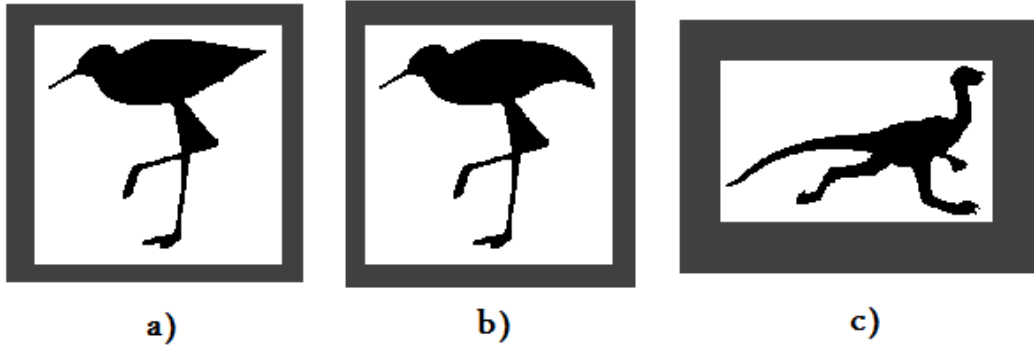
3.7. Deney Sonuçları

İmgelerin çizgelerinin oluşturulmasının ardından çizgelerin norm uzaya aktarılması MATLAB ortamında gerçekleştirilmiştir. Bu aşamadan sonra gerçekleştirilen norm uzaydaki lineer programlama probleminin çözülmesi ve iki imgenin karşılaştırılması için C dili kullanılmıştır.

C dili alt düzey bir programlama dili olmakla beraber oldukça hızlıdır ve EMD algoritması için bu dilde hazırlanmış ve test edilmiş çerçeve bir program vardır [12].

Çerçeve program farklı boyutlardaki norm uzaylarda çalışabilecek şekilde değiştirilmiş ve iki nokta arasındaki uzaklığı sonsuz norm altında hesaplayacak hale getirilmiştir. MATLAB ile yazılmış bir betiğin C diline farklı boyutlarda matrisler göndermesi ve hesaplanan EMD değerini geri alması için MATLAB MEX adıyla bilinen C dilinde yazılmış dış arayüz sınıfı kodlanmıştır. Böylelikle MATLAB ile işlenen iki imgenin arasındaki benzerlik değeri C dili ile kodlanmış bir programa hesaplatıldıktan sonra yine MATLAB ortamına geri alınabilmektedir.

Yöntemimiz Şekil 9'da gösterilen iki kuş imgesi ve bir dinazor imgesi ile sınınmıştır. İkinci imge birinci imgenin değiştirilmiş bir halidir ve algoritmanın denenmesi için kullanılmıştır. Bu iki yakın imgenin kullanılmasının nedeni aralarında bulunan uzaklık değerinin düşük olmasının beklenmesidir. Örneğin, a ve b şekilleri arasındaki uzaklığın a ve c şekilleri arasındaki uzaklıktan daha az olması geliştirdiğimiz yaklaşımın doğru sonuçlar verdiğine dair bir kanıt olabilir. Ancak bu sağlanmıyorsa yaklaşımda bir hata olduğu düşünülebilir.



Şekil 9. Üç örnek imge; (b), (a) imgesinin değiştirilmiş hali

İmgelerden oluşturulan çizgelerin düğüm sayısı imgelerin taşıdıkları norm uzayının boyutunu belirlemektedir. Çizgelerin düğüm sayısı ise her imgenin iskeletsel noktalarının sayısına bağlıdır. Bu durumda ortaya şöyle bir soru çıkmaktadır: Farklı boyutlu iki norm uzaydaki iki ayrı dağılım birbiriyle nasıl karşılaştırılacaktır? Bu sorunu çözmek için boyut indirgeme yöntemleri (Multidimensional Scaling, Principal Component Analysis) kullanılır.

Çalışmamızda iki farklı boyut indirgeme yöntemi denenmiştir: Bunlardan birincisinde düşük boyutlu dağılımdaki noktalar sıfır ile yastıklanarak diğer dağılımın boyutuna yükseltilmiştir. İkinci yöntemde ise yüksek boyutlu dağılımdaki noktaların boyutları düşük boyutlu dağılımdaki noktaların boyutuna indirilmiştir. Boyut indirme yüksek boyutlu dağılımda yer alan noktaların koordinatlarının yeterli bir kısmını silerek gerçekleşmiştir. Silinen miktar iki dağılımın boyutları arasındaki fark kadardır.

Her iki yöntemle yapılan boyut eşitleme için Şekil 9’da verilen imgelerin karşılaştırılmasından alınan sonuçlar Çizelge 1’de yer almaktadır.

Çizelge 1. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri

İmgeler	Sıfırla yastıklama	Silerek küçültme
(a) ve (b)	104,878	99,401
(a) ve (c)	96,825	94,562

Sonuçlar incelendiğinde birbirine çok benzeyen (a) ve (b) imgeleri arasındaki uzaklığın her iki boyut eşitleme yöntemiyle de (a) ve (c) imgeleri arasındaki uzaklıktan daha büyük çıktığı görülür. Bu yöntemimizde bir sorun olduğunu göstermektedir. Bu sorunun kaynağının iskeletsel noktaların seçiminde kullanılan k-ortalamar algoritması olabileceği düşünülmüştür. K-ortalamar yöntemiyle seçilen noktalar bir ölçüde rastgelelik barındırmaktadır, çünkü algoritmanın ilk çalışması sırasında belirlenen küme merkezleri rastgele seçilmektedir. Örneğin, aynı nokta kümesi için algoritmayı iki kez çalıştırsak birbirine yakın ancak farklı küme merkezleri belirlenir. Bu nedenle yöntemimizden bu rastgelelik olgusu çıkarılmış ve yöntem yeniden sınanmıştır.

3.8. Tüm İskeletsel Noktalarla Yöntemin Denenmesi

Çoklu eşlemenin l_∞ normu altında izometrik biçimde gerçekleştirilmesi için ilk başlangıçta planladığımız adımların başarısız olduğu görülmüştür. Bu aşamada k-

ortalamalar algoritmasını yöntemimizden çıkararak iskeletsel noktaların seçimindeki rastgelelik ortadan kaldırılmıştır.

Bu haliyle yöntemimizin işleyişinde iki önemli değişiklik olmuştur. Birincisi k-ortalamalar algoritmasıyla seçilen belirli bir miktarda noktadan çizgenin oluşturulması yerine çıkardığımız iskeletteki tüm noktalar kullanılarak çizgeler oluşturulmuştur. Bir diğer değişiklik ise çizge kenarları belirlenirken hesaplanan eşik değerde meydana gelmiştir. Eşik değer noktalar arası uzaklıkların istatistiklerinden yararlanarak belirlenmemiş, bunun yerine birbirine bitişik iki nokta arasındaki maksimum uzaklık değeri hesaplanarak eşik değeri oluşturulmuştur. Yan yana veya üst üste duran iki nokta arasındaki uzaklık 1 piksel iken birbirlerine göre çapraz konumda olan iki nokta arasındaki uzaklık $\sqrt{2}$ pikseldir. Bu nedenle iskeletsel noktaların tamamı birbirine komşu olduğundan eşik değerini 1,5 olarak belirlemek yeterli olmuştur.

Sözü edilen iki değişiklik dışında yöntemin adımlarında bir değişiklik yapılmamıştır. Kısaca imgelerin orta eksen dönüştürme yöntemiyle iskeletleri çıkarılmış ve bu iskeletler üzerindeki tüm noktalar kullanılarak çizgeler oluşturulmuştur. Çizgeler norm uzayda dağılımlar olarak ifade edilmiş ve karşılaştırılmıştır.

Bu karşılaştırma sonucunda Şekil 9'da yer alan imgeler için Çizelge 2'de yer alan sonuçlar elde edilmiştir.

Çizelge 2. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri

İmgeler	Sıfırla yastıklama	Silerek küçültme
(a) ve (b)	133,665	81,653
(a) ve (c)	100,513	95,705

Çizelgedeki bilgiler incelendiği zaman her iki boyut indirgeme yöntemiyle de istenilen sonuçların alınmadığı görülür. Birbirine çok benzeyen iki şekil arasındaki uzaklığın birbirine uzak iki imgenin arasındaki uzaklıktan çok daha düşük bir değer

alması beklenmiştir. Ancak bu beklenti sonsuz norm uzaklığının hassaslığı ve bu uzaklık ölçüsünde boyut indirgeme yöntemlerinin etkisizliği nedeniyle gerçekleşmemiştir.

3.9. Tartışma ve Yeni Yöntem

İskeletsel noktalardan k-ortalamlar yöntemiyle seçilen temsilcilerle veya iskeletsel noktaların tamamı kullanılarak yapılan karşılaştırmaların başarısız olmasının temel iki nedeni vardır. Birinci neden norm uzaya aktarılan çizgelerin farklı sayıda düğüm içermesi ve bunun norm uzayın boyutuna etki etmesidir. Norm uzaydaki dağılımları aynı boyuta getirmek için kullanılan sıfırla yastıklama yöntemi sonsuz norm uzaklığının hesaplanması gereği iyi sonuçlar vermemektedir. Bu norm altında iki nokta arasındaki uzaklık noktaların tüm koordinatları arasındaki farkların en büyüğü hesaplanarak elde edilir. Bu nedenle sıfırla yastıklanıp düşükten yüksek boyuta çekilen noktalar karşılaştırmalar sırasında büyük uzaklık farklarına yol açmaktadır. Bir diğer yöntem olan silerek küçültme ise uygulamaya büyük bir rastgelelik etkeni katmaktadır. Büyük boyutlu norm uzaydaki noktaların belirli koordinatlarını hesaplamaya katmamak önemli bir bilginin kaçırılmasına neden olabilmektedir.

Şimdiye kadar yaptığımız karşılaştırmaların başarısız olmasının ikinci ve en önemli nedeni sonsuz norm uzaklığıyla gerçekleştirilen karşılaştırma işleminin çizgeler üzerindeki ufak değişikliklere çok duyarlı olmasıdır. Bunu açıklayabilmek için k-ortalamlar algoritmasıyla seçilen temsilci sayısı 50 ile sınırlanmıştır; böylelikle boyut indirgeme yöntemlerinin kullanılmasına gerek kalmamıştır. Oluşan çizgelerin düğüm sayısı 50 olmuştur ve imgeler 50 boyutlu norm uzayda 50 nokta şeklinde ifade edilmiştir. Bu durumda Şekil 9'da yer alan imgeler için karşılaştırma sonuçları Çizelge 3'te verilmiştir.

Çizelge 3. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri

İmgeler	Hesaplanan uzaklık
(a) ve (b)	99,195
(a) ve (c)	91,402

Çizelgeden anlaşıldığı gibi boyut indirgeme yöntemlerinin etkisi olmasa da karşılaştırmalardan beklenen sonuç alınamamıştır. Bu durum (a) ve (b) kuş imgelerinin yalnızca kuyrukları birbirinden farklı olsa da diğer bütün bölümlerinden elde edilen noktaların bu farktan etkilenmesiyle oluşmaktadır. Norm uzaya taşınan noktalar gerçekte temsil ettikleri imgenin aynı bölümünü göstermelerine rağmen aralarında bir uzaklık varmış gibi değerlendirilmektedir. Bu durum sonsuz norm uzaklığının tanımından kaynaklanmaktadır (3.1).

Bahsedilen olumsuzlukların üstesinden gelmek için çizge metriğinin izometrik taşınması fikrinden vazgeçilmiş ve çizgeler referans noktaları yardımıyla norm uzaya taşınarak karşılaştırılmıştır.

4. REFERANS NOKTALARIYLA ÇOKLU EŞLEME

Referans noktalarıyla çoklu eşleme yaklaşımı Veltkamp'ın 2002 yılında yapmış olduğu bir çalışmadan esinlenerek oluşturulmuştur [5]. Söz konusu çalışmada referans olarak belirli sayıda ayırıcı nitelikte nesne kullanılmıştır. Veritabanındaki imgeler bu referans nesnelerle karşılaştırılarak çok boyutlu bir benzerlik vektörü elde edilmiştir. Bu vektörün boyutu kullanılan referans nesne sayısı kadardır. Daha sonra sorgusu yapılan bir nesne veri tabanındaki bütün imgelerle hesaplama açısından maliyetli yöntemlerle karşılaştırılmak yerine referans nesnelerle karşılaştırılmıştır. Bunun ardından sorgusu yapılan nesne için oluşturulan benzerlik vektörü hızlı biçimde veritabanındaki benzerlik vektörleriyle karşılaştırılmıştır. Veri tabanındaki imgelerin benzerlik vektörü önceden oluşturulduğu için yöntem oldukça hızlıdır. Hesaplama açısından maliyeti fazla olan ancak güvenilir yöntemler çevrimdışı gerçekleştirilmiştir.

Referans noktalarıyla çoklu eşleme bir önceki bölümde anlatılan yöntem gibi izometrik bir yöntem değildir. Ancak l_∞ normu altında çoklu eşlemenin gerçekleştirilebilmesi için bu özelliğin göz ardı edilmesi gerekmektedir. Referans noktalarıyla çoklu eşlemenin neden izometrik olmadığını anlamak için Şekil 10 incelenmiştir. Şekilde görünen iskeletin elimizdeki imgelerden birine ait olduğu düşünülebilir. Bu iskelet üzerinde bulunan v_1 ve v_2 noktaları seçilen noktalar, diğer bir deyişle çizge düğümleri olsun. A ve B noktaları ise iskeletten seçilen referans noktaları olsun. Şekilde d ve katlarıyla verilen bilgiler de bu noktaların arasındaki uzaklıkları belirlesin. Bu şekilde oluşturulan bir çizgede v_1 ve v_2 noktaları arasındaki en kısa yol uzaklığı $d + d$ toplamından $2d$ olacaktır. Ancak v_1 ve v_2 noktası norm uzaya A ve B noktalarına olan uzaklıklarına göre taşınırsa, v_1 ve v_2 noktasının koordinatları şu şekilde belirlenir:

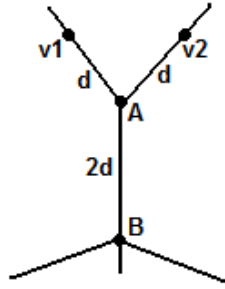
$$p_{v_1} = [(d), (d + 2d)] = [d, 3d]$$

$$p_{v_2} = [(d), (d + 2d)] = [d, 3d]$$

Bu durumda sonsuz norm uzayda bu iki nokta arasındaki uzaklık:

$$\| p_{v_1} - p_{v_2} \|_\infty = \max((d - d), (3d - 3d)) = 0$$

Görüldüğü gibi $v1$ ve $v2$ düğümü arasındaki uzaklık norm uzayda bu düğümlere karşılık gelen noktalar arasında korunmamaktadır. Bu nedenle referans noktalarıyla metrik uzayda yapılan taşıma işlemleri izometrik değildir. Ancak verilen bilgiler, bu yöntemle metrik uzayda taşınan tüm noktalar arasındaki uzaklıkların norm uzayda bozulduğu anlamına gelmez.



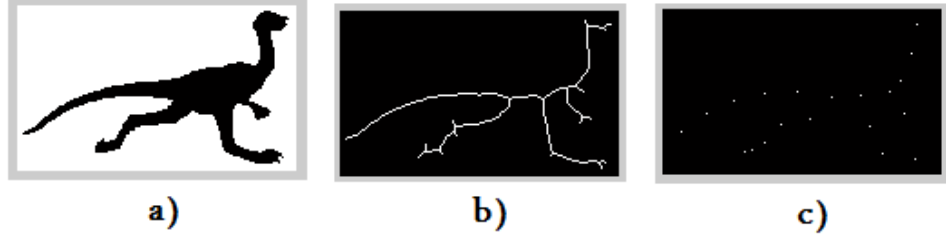
Şekil 10. Temsili iskelet. A ve B referans noktaları. $v1$ ve $v2$ seçilen iskeletsel noktalar

Bu bölüm altında anlatılan metrik taşıma yöntemlerinden sonuncusu izometrik olmamasına rağmen nesne tanıma problemi için etkili sonuçlar ortaya çıkarmıştır.

4.1. Birinci Yöntem: K-ortalamlar Algoritmasıyla Referans Nokta Seçimi

Bu yöntemle bir imgenin iskeleti çıkartıldıktan sonra iskelet üzerinden belirli sayıda referans noktası seçilmektedir. Referans seçimi daha önce açıklanan k-ortalamlar yöntemiyle gerçekleştirilir. İskelet üzerindeki referans noktalar hariç diğer tüm noktalar çizge düğümlerini belirler. Düğüm noktalarının belirlenen referans noktalarına en kısa yol uzaklıkları ise bu noktaların norm uzaydaki koordinatlarını oluşturmaktadır.

Şekil 11 incelendiğinde bir dinazor imgesinin iskeleti ve bu iskelet üzerinden seçilen referans noktaları görülmektedir.



Şekil 11. a) Dinozor imgesi, b)İskelet noktaları, c) Referans noktaları

Yapılan deneylerde Şekil 9’da verilen imgeler için Çizelge 4’te yer alan sonuçlar elde edilmiştir. Hesaplanan uzaklık değerlerine bakıldığı zaman yine beklenen uzaklık farklarının oluşmadığı görülmektedir. Dinozor imgesi ile birinci kuş imgesi arasındaki uzaklık iki kuş imgesi arasındaki uzaklıktan daha az hesaplanmıştır. Bunun nedeninin imgenin kenarlarına yakın bölgelerinden seçilen referans noktaları olabileceği düşünülmüştür. Buradaki uç noktalardan herhangi birisindeki ufak bir değişimin, örneğin kuş imgesinin kuyruğundaki değişim, bütün diğer noktaların kendi aralarındaki uzaklıklara etki ettiğinden şüphe edilmiştir.

Çizelge 4. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri

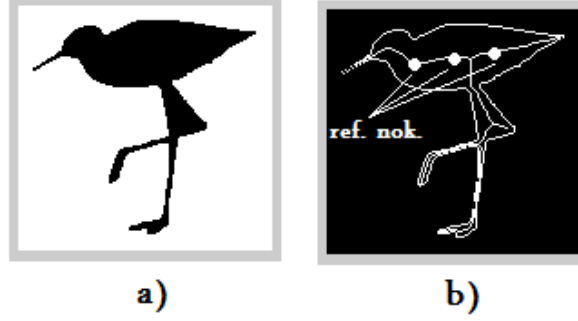
İmgeler	Hesaplanan Uzaklık
(a) ve (b)	114,158
(a) ve (c)	94,751

Bahsedilen nedenlerden ötürü referans noktaların imgelerin kenarlarına yakın noktalardan seçilmemesi gerektiği anlaşılmıştır.

4.2. İkinci Yöntem: Ağırlığı Büyük Noktaların Referans Seçilmesi

Bu yöntemle birinci yöntemden farklı olarak referans noktaları seçilirken bütün iskelet noktaları kullanılmamış ağırlığı büyük noktalardan k-ortalamlar algoritmasıyla seçim yapılmıştır. Bu yolla imgelerin kenarlarına yakın yerlerden referans nokta seçiminin önüne geçilmiştir.

Şekil 12’de k-ortalamlar algoritmasıyla seçilmiş referans noktaları gösterilmektedir. İskelet üzerindeki noktaların yarıçaplarına göre en büyük %25’lik bölümünde kalan noktalar kullanılmıştır. Böylece imgenin kenarlarına yakın ve değişikliklere duyarlı olan bölgelerinden referans noktası seçilmemiş olur.



Şekil 12. a) Veritabanından bir imge, b) Yarıçapı büyük noktalardan seçilmiş referans noktaları
Bu yöntemle Şekil 9’da verilen imgeler üzerinde yaptığımız deneylerde Çizelge 5’te yer alan sonuçlar alınmıştır.

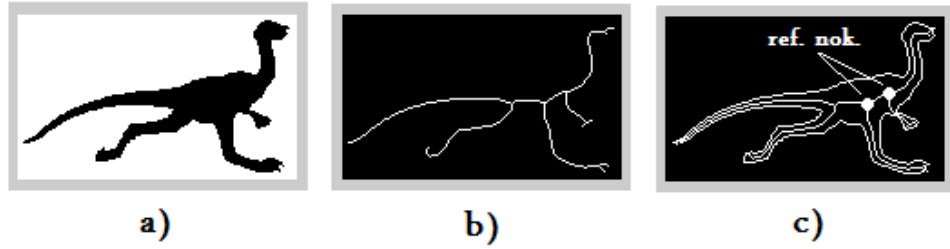
Çizelge 5. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri

İmgeler	Hesaplanan Uzaklık
(a) ve (b)	18,506
(a) ve (c)	18,198

Çizelge incelendiğinde yine istenilen sonuçların elde edilemediği görülmektedir. Bunun nedeni seçilen referans noktalarının k-ortalamlar algoritmasıyla seçilmiş olmasıdır. Bu algoritmayla seçilen referans noktaları birbirine benzer iki imge için yakın konumlarda olmalarına rağmen aralarında yine mesafe bulunmaktadır. Bu nedenle referans noktalarının seçiminde aynı nesnenin imgeleri arasında çok fazla değişiklik göstermeyen noktalar bulunmalıdır.

4.3. Üçüncü Yöntem: Ağırlığı En Büyük İki Dallanma Noktasının Referans Alınması

Aynı nesnenin farklı imgeleri arasında belirgin farklılıklar göstermeyen noktaların referans noktası olarak seçilmesi gerekmektedir. Bunun için imge iskeletlerinin dallanma noktaları referans olarak alınmıştır. Ancak imgelerdeki dallanma noktaları farklı sayılarda olabilir ve bunlardan bir seçim yapmak gerekmektedir. Bu seçim dallanma noktalarından yarıçapları en büyük olan iki noktanın seçilmesiyle yapılmıştır. Şekil 13'te bir imgeden seçilen iki referans noktası görülmektedir.



Şekil 13. a) İkili bir imge, b) İmgenin iskeleti, c) Seçilen referans noktaları

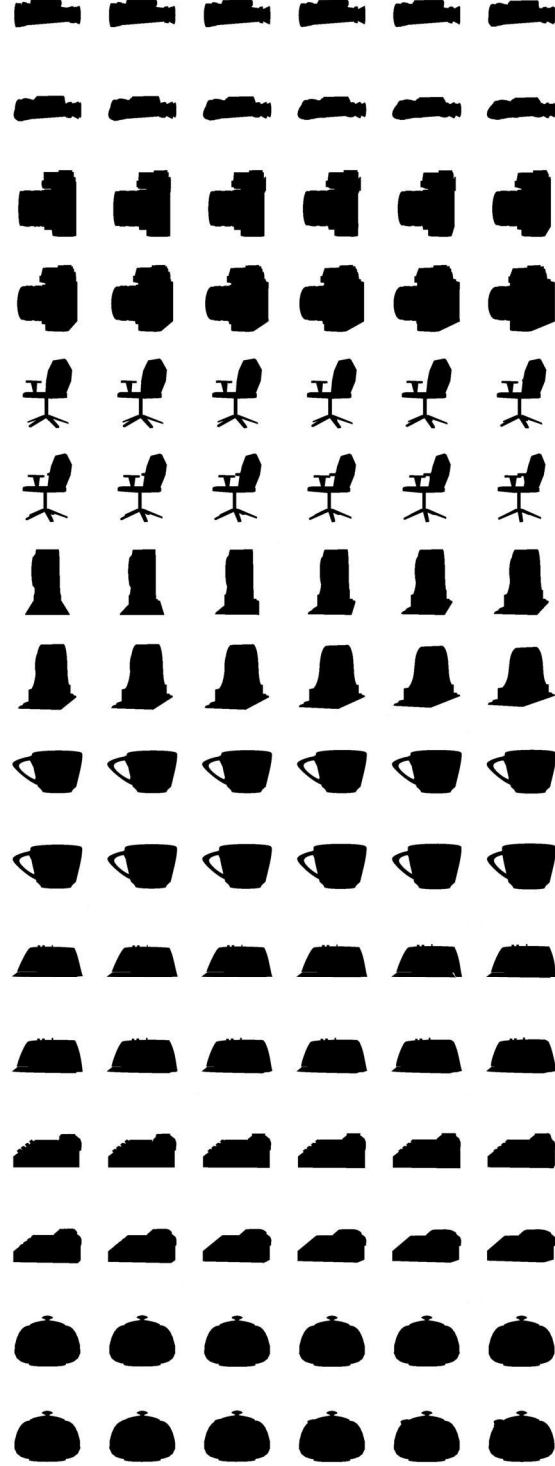
Şekil 9'daki imgeler için yöntem en son haliyle denendiğinde Çizelge 6'da yer alan sonuçlar elde edilmiştir.

Çizelge 6. İmgelerin karşılaştırılmasından elde edilen uzaklık değerleri

İmgeler	Hesaplanan Uzaklık
(a) ve (b)	4,340
(a) ve (c)	35,602

Çizelgeye bakıldığı zaman sonuçların beklentilerle uyduğu görülmektedir. Birbirine yakın iki cisim için hesaplanan uzaklık diğer iki imge arasındaki uzaklıktan oldukça düşük çıkmıştır. Bu yöntemin doğru olabileceğine dair bir işaret olmasına rağmen tek başına yeterli değildir. Bu nedenle elimizdeki imge veritabanından seçilen 8 sınıf ve her sınıfta 12 imge olmak üzere 96 imgelik bir veri seti üzerinde deneyler yapılmıştır (Şekil 14). Deney setindeki imgeler üç boyutlu nesnelerin ikişer

derecelik açılarla düşey eksen etrafında çevrilmesi ve siyah beyaz izdüşümlerinin alınmasıyla oluşturulmuştur.



Şekil 14. İmge veritabanı

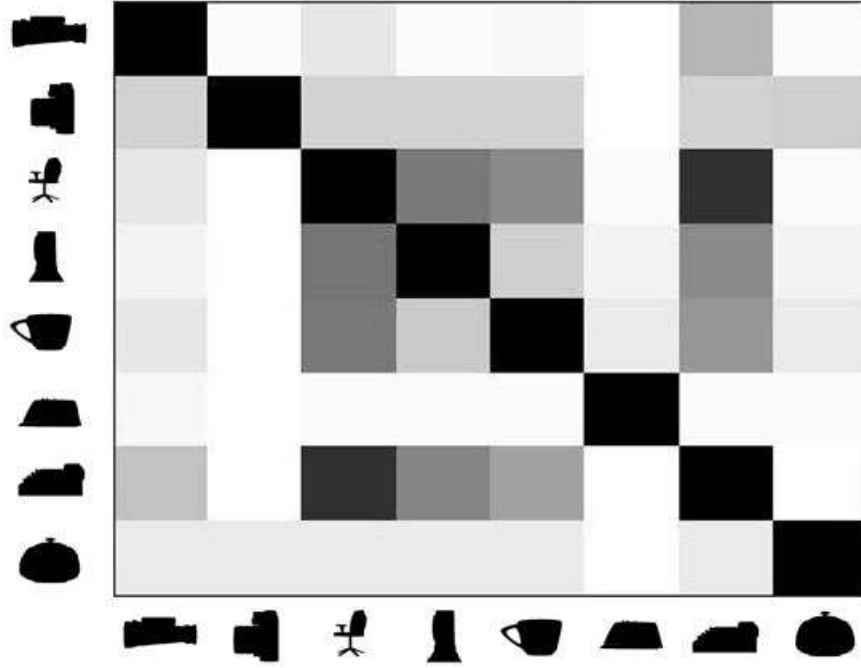
Yapılan deneylerden birincisi en yakın komşu istatistiğini bulmaya yöneliktir. Bu istatistik sorgusu yapılan bir imgenin kendisi dışında veritabanındaki en yakın tahminlerinin sınıflarına bakmaya dayanır. Örneğin birinci en yakın komşu istatistiği sorgusu yapılan bir imgeye en yakın tahminin doğru sınıftan yapılmasını başarı sayar. Veritabanındaki tüm imgeler için başarılı olma sayısı veritabanının büyüklüğüne bölünürse istatistik elde edilmiş olur. Formüle edilirse:

Birinci en yakın komşu istatistiği

$$= \frac{\text{ilk komşuda doğru sınıfa ulaşma(başarı) sayısı}}{\text{toplam imge sayısı}}$$

Geliştirdiğimiz yöntem için birinci en yakın komşu istatistiği %91 çıkmıştır. Bu şu anlama gelir sorgusu yapılan bir nesnenin sınıfı %91 oranla doğru tahmin edilmektedir. İkinci en yakın komşu istatistiği %94 olarak gerçekleşmiştir. Bu değer, sorgusu yapılan bir imgenin ilk iki tahminde sınıfının doğru bilinmesi olasılığıdır.


















Yöntemin bir diğer değerlendirilmesi en kötü tahminin ne kadar uzakta gerçekleştiğine bakılarak yapılır. Kullandığımız veri seti ve yöntemle bu değer 14. sıra olarak gerçekleşmiştir. Bir imge sorgusunda en kötü olasılıkla veri setinin %14'lük bölümünde doğru sınıfa ulaşılmaktadır. Veri setinin kalan kısmının imgenin sınıfını öğrenmek için irdelenmesine gerek kalmamaktadır.



Şekil 15. Koyu renkler imge sınıfları arasında güçlü bir benzerlik olduğunu gösterir. Köşegen üzerinde kümelenen koyu renk aynı sınıfın imgeleri arasındaki benzerliğin fazla olduğunu işaret eder.

Bir başka değerlendirme veri setindeki sınıfların birbirlerine olan ortalama uzaklıklarına bakılarak yapılır. Eğer bir sınıftaki imgelerin kendi sınıfındaki imgelere olan uzaklıkları diğer sınıflarda yer alanlara göre düşükse yöntem iyi sonuçlar veriyordur. Şekil 15'te yer alan uzaklık şablonu 96 imgelik deney setimizden elde edilmiştir. Koyu renklerle belirtilen bölümler uzaklıkların düşük olduğu yerlerdir. Koyu renkli kısımların köşegen üzerinde yer alması tahminlerde elde edilen başarıyı göstermektedir. Uzaklık şablonunda telefon ve sandalye sınıfı arasında güçlü bir benzerlik varmış gibi gözükmektedir. Bunun nedeni izometrik olmayan metrik uzayda taşıma işleminin bu iki sınıfın karşılaştırılmasını olumsuz etkilemesidir.

Deney setindeki her sınıftan seçilen iki birbirinden farklı imge için elde ettiğimiz karşılaştırma sonuçlarına Şekil 16'da yer verilmiştir. Yöntemimizin uygulanmasının ardından iki imgenin karşılaştırılmasından elde edilen değer aynı sınıfın imgeleri için düşük olması beklenmektedir. Şekilde görüldüğü gibi aynı sınıfın imgeleri arasında oluşan uzaklık değerleri farklı sınıftan imgeler arasında oluşanlardan daha düşüktür.

	0,974	34,431	59,619	64,825	42,662	29,590	19,914	19,705
	35,196	1,189	17,115	26,060	16,532	21,292	48,571	55,170
	61,264	18,130	1,392	14,910	30,187	44,882	73,303	66,471
	65,402	26,372	15,300	1,261	23,261	46,762	79,441	80,307
	43,071	17,088	28,791	25,436	0,849	25,904	55,908	62,977
	30,570	20,934	43,635	45,556	25,469	0,925	44,353	40,085
	15,870	47,459	70,999	78,225	54,766	43,415	2,024	9,221
	18,756	55,684	65,766	81,520	63,793	41,981	7,766	0,049
								

Şekil 16. 16 farklı imge için elde edilen benzerlik değerleri. Düşük değerli imge çiftlerinin birbirleriyle aynı sınıftan olması beklenir. Satır ve sütunların en küçük değerleri işaretlenmiştir.

Yapılan deneylerden elde edilen sonuçlar geliştirmiş olduğumuz referans noktalarıyla çoklu eşleme yaklaşımının nesne tanıma probleminin çözümlenmesinde kullanılabileceğini göstermiştir.

5. SONUÇLAR ve TARTIŞMA

5.1. Yapılan Çalışma ve Sonuçlar

Bu tez çalışmasında nesne tanıma problemi için farklı yaklaşımlar denenmiştir. Genel olarak imgelerin orta eksen dönüştürme yöntemi yardımıyla iskeletleri çıkartılmış ve bu iskeletler nesnelerin çizgelerini oluşturmak için kullanılmıştır. Çizgeler üzerinde metrik bir uzaklık olarak en kısa yol uzaklığı tanımlanmıştır. Ardından çizge metriği sonsuz norm uzayına taşınarak nokta kümesi olarak ifade edilmiştir. Sonsuz norm uzayında imgelerden elde edilen iki dağılım karşılaştırılarak imgeler arasındaki benzerliğin derecesi belirlenmeye çalışılmıştır.

Tezin 3. bölümünde ayrıntılı olarak anlatılan metrik uzayda izometrik taşıma işlemi sonsuz norm uzaklığının çizgeler üzerindeki değişikliklere olan duyarlılığı nedeniyle istenilen sonuçları vermemiştir. Bu nedenle 4. bölümde izometrik taşıma fikrinden vazgeçilmiş ve referans noktalarıyla metrik uzayda taşıma işlemi gerçekleştirilmiştir. Bu yöntemle gerçekleştirilen nesne tanıma uygulaması farklı kıstaslara göre değerlendirilmiş ve olumlu sonuçlar alınmıştır.

5.2. Gelecek Çalışmalar

Tezin 4. bölümünde anlatılan referans noktalarıyla çoklu eşleme yöntemi geliştirmeye açık gözükmektedir. Referans noktalarının seçiminde kullandığımız dallanma noktaları yerine, imgeler için daha belirleyici ve aynı nesnenin farklı imgeleri için az değişim gösteren öznitelikler belirlenebilir. Bunun sağlanabilmesi için imgelerin iskeletlerinden yararlanılarak referans noktası belirleme yaklaşımından vazgeçilmesi gerekebilir. Böyle bir durumda imgelerin çizgelerini elde etmek için başka bir yöntem geliştirilmesi düşünülebilir. Eğer bu gerçekleştirilebilirse çizgelerin doğru eşlenmesinde gelişme sağlanabileceği düşünülmektedir.

Çalışmamızda norm uzayda dağılımların karşılaştırılmasında kullanılan sonsuz norm uzaklığı yerine farklı bir uzaklık normu üzerinde denemeler yapılması düşünülebilir. Buna uygun seçilmiş bir metrik taşıma yöntemi ve referans noktası çıkarma algoritmasıyla çoklu eşleme yapılabilecektir.

KAYNAKLAR

- [1] Rubner Y., Tomasi C., Guibas L. J., The earth mover's distance as a metric for image retrieval, *International Journal of Computer Vision*, 40(2), 99-121, 2000.
- [2] Bennamoun M., Mamic G. J., *Object recognition: Fundamentals and case studies*, Springer, New York, 2002.
- [3] Barnard S. T., Fischler M. A., Computational stereo, *ACM Computing Surveys*, 14(4),553-572, 1982.
- [4] Medioni G., Nevatia R., Segment-based stereo matching, *Computer Vision, Graphics and Image Processing*, 31, 2-18, 1985.
- [5] Vleugels J., Veltkamp R. C., Efficient image retrieval through vantage objects, *Pattern Recognition*, 35(1), 69-80, 2002.
- [6] Demirci M. F., Shokoufandeh A., Keselman Y., Bretzner L., Dickinson S., Object recognition as many-to-many feature matching, *International Journal of Computer Vision*, 69(2), 203-222, 2006.
- [7] You J., Bhattacharya P., A wavelet-based coarse-to-fine image matching scheme in a paralel virtual machine enviroment, *IEEE Transactions on Image Processing*, 9(9), 1547-1559,2000.
- [8] Förstner W., Feature extraction in digital photogrammetry, *The Photogrammetric Record*, 14(82), 595-611, 1993.
- [9] Boyer K. L., Kak A. C., Structural stereopsis for 3-d vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2), 144-166, 1988.
- [10] Horaud R., Skordas T., Stereo correspondence through feature grouping and maximal cliques, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11), 1168-1180, 1989.
- [11] Jiang H., Ngo C. W., Graph based image matching, *Proceedings of the 17th International Conference on Pattern Recognition 2004*, 658-661, Cambridge, UK, Ağustos 2004.
- [12] "EMD uygulaması" erişim adresi: <http://ai.stanford.edu/~rubner/emd/default.htm>, erişim tarihi: 25 Nisan 2010.
- [13] Dickinson S., Pentland A., Rosenfeld A., 3-d shape recovery using distributed aspect matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 174-198, 1992.
- [14] Cyr C. M., Kimia B. B., A similarity-based aspect-graph approach to 3d object recognition, *International Journal of Computer Vision*, 57(1), 5-22, 2004.
- [15] Ioffe S., Forsyth D. A., Human tracking with mixtures of trees, *ICCV01*, 690-695, 2001.

- [16] “Düzgeleme” erişim adresi: http://www.metu.edu.tr/cgi-bin/sozluk/turing/sozluk_tr?query=normalization, erişim tarihi: 4 Temmuz 2010.
- [17] Hitchcock F. L., The distribution of a product from several sources to numerous localities, *Journal of Mathematical Physics*, Vol. 20, 224-230, 1941.
- [18] Rubner Y., Tomasi C., Guibas L. J., A metric for distributions with applications to image databases, *Computer Vision*, 59-66, 1998.
- [19] Eckman B. A., Brown P. G., Graph data management for molecular and cell biology, *IBM Journal of Research and Development*, 50(6), 545-560, 2006.
- [20] Dagdeviren O., Erciyes K., Graph matching-based distributed clustering and backbone formation algorithms for sensor networks, *The Computer Journal*, Vol. 10, 2010.
- [21] “Eşbiçimli çizgeler” erişim adresi: http://en.wikipedia.org/wiki/Graph_isomorphism, erişim tarihi: 28 Haziran 2010.
- [22] Barrow H. G., Burstall R. M., Subgraph isomorphism, matching relational structures and maximal cliques, 76(4), 83-84, 1975.
- [23] Linial N., London E., Rabinovich Y., The geometry of graphs and some of its algorithmic applications, *Combinatorica*, 15(2), 215-245, 1995.
- [24] Cormen T. H., Leiserson C. E., Rivest R. L., Stein C., *Introduction to Algorithms*, MIT Press, 2001.
- [25] Liu T. L., Geiger D., Approximate tree matching and shape similarity, *Proceedings of 7th International Conference on Computer Vision*, 456-462, Kerkyra, Greece, 1999.
- [26] Indyk P., Algorithmic applications of low-distortion geometric embeddings, *Foundations of Computer Science*, 10-33, 2001.
- [27] “Mapping” erişim adresi: <http://sozluk.bidb.odtu.edu.tr/> erişim tarihi: 30 Haziran 2010.
- [28] Dijkstra E. W., A note on two problems in connexion with graphs, *Numerische Math*, Vol. 1, 269-271, 1959.
- [29] Gupta A., Embedding tree metrics into low dimensional Euclidean spaces, *Proceedings of the thirty-rst annual ACM symposium on Theory of computing*, 694-700, 1999.
- [30] Ostrovsky R., Rabani Y., Low distortion embeddings for edit distance, *Journal of the ACM*, 54(5), 2007.
- [31] Blum H., *Models for the Perception of Speech and Visual Form*, MIT Press, Cambridge, MA, 1967.

[32] “Orta eksen” eriřim adresi: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/skeleton.htm>, eriřim tarihi: 1 Temmuz 2010.

[33] “K-ortalamlar algoritması” eriřim adresi: <http://www.davutcesmeci.com/k-ortalama-k-means-kumeleme-yontemi/278>, eriřim tarihi: 1 Temmuz 2010.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ERDEM, Caner
Uyruğu : T.C.
Doğum tarihi ve yeri : 11.03.1984 Ankara
Medeni hali : Bekar
Telefon : 0 (312) 292 51 75
Faks : 0 (312) 292 42 80
e-mail : cerdem@etu.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Lisans	ODTÜ/İstatistik Bölümü	2007

İş Deneyimi

Yıl	Yer	Görev
2007-2008	Yapı ve Kredi Bankası	Müfettiş Yardımcısı
2008-2010	TOBB Ekonomi ve Teknoloji Üniversitesi	Araştırma Görevlisi

Yabancı Dil

İngilizce

Yayınlar