

**BİRDEN FAZLA OPTİMAL ÇÖZÜM İLE ŞEKİL ÖZETİ
OLUŞTURMA**

MARLEN AKİMALİEV

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

**TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ARALIK 2012

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Ünver KAYNAK
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Doç. Dr. Erdoğan DOĞDU
Anabilim Dalı Başkanı

MARLEN AKİMALİEV tarafından hazırlanan BİR DEN FAZLA OPTİMAL ÇÖZÜM İLE ŞEKİL ÖZETİ OLUŞTURMA adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. M.Fatih DEMİRCİ
Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Murat Özbayoğlu

Üye : Yrd. Doç. Dr. M.Fatih DEMİRCİ

Üye : Doç. Dr. Bülent TAVLI

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.

Marlen Akimaliev

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Bilgisayar Mühendisliği
Tez Danışmanı : Yrd. Doç. Dr. M.Fatih DEMİRCİ
Tez Türü ve Tarihi : Yüksek Lisans – Aralık 2012

Marlen Akimaliev

BİRDEN FAZLA OPTİMAL ÇÖZÜM İLE ŞEKİL ÖZETİ OLUŞTURMA

ÖZET

Şekil özeti oluşturma örüntü tanıma, bilgisayarla görü ve endüstriyel tasarım gibi birçok alanda araştırmacıların karşılaştığı problemlerdendir. Bir şekil kümesi verildiğinde, yeni geliştirilen şekil özeti oluşturma metodu bu şekillerin özetini, şekillerin öznitelikleri arasındaki eşlemeleri kullanarak bulur. Eşlemeler yaygın olarak bilinen taşıma probleminin optimal çözümüne dayanarak elde edilir. Bir taşıma problemi için birden fazla optimal çözüm bulunduğu göz önüne alındığında, bu çalışma bütün optimal çözümleri yerel komşuluk ilişkilerini ne kadar koruduğuna dayanarak derecelendirir ve en yüksek dereceye sahip çözümü kullanarak özet şekli oluşturur. Sistem üzerinde yapılan deneysel değerlendirme önerilen yaklaşımın bir önceki şekil özeti oluşturma tekniğine karşılık, daha iyi sonuçlar verdiğini göstermiştir.

Anahtar Kelimeler: şekil özeti oluşturma, taşıma problemi, birden fazla optimal çözüm.

University : TOBB University of Economics and Technology
Institute : Institute of Natural and Applied Sciences
Science Programme : Computer Engineering
Supervisor : Assistant Prof. M.Fatih DEMİRCİ
Degree Awarded and Date : M.Sc. – DECEMBER 2012

Marlen Akimaliev

SHAPE ABSTRACTION THROUGH MULTIPLE OPTIMAL SOLUTIONS

ABSTRACT

One of the most important problems facing researchers in various fields such as pattern recognition, computer vision, and industrial design is the shape abstraction. A recent shape abstraction method generates an abstracted shape through the correspondences between their features from a given set of shapes. The correspondences are obtained based on the optimal solution of a well known transportation problem. This paper considers the case where multiple optimal solutions exist for one problem and ranks all optimal solutions based on how much they preserve the local neighborhood relations. Afterwards, the abstracted shape is created using the solution with the highest rank. Experimental evaluation of the framework demonstrates that the proposed approach compares favorably with the previous shape abstraction technique.

Keywords: shape abstraction, transportation problem, multiple optimal solutions.

TEŞEKKÜR

Bu çalışmayı tamamlamamda çok katkısı olan, bana karşı manevi desteklerini esirgmeden daima destek olan danışmanım Yrd. Doç. Dr. M.Fatih DEMİRCİ'ye en içten teşekkürlerimi sunarım. Araştırma boyunca onunla çalışmak bana çok büyük katkı sağladı.

Bu tez 109E183 proje numarasına sahip, TÜBİTAK Kariyer projesi tarafından desteklenen ve Yrd. Doç. Dr. M.Fatih DEMİRCİ koordinatörlüğünde yürütülen "Şekil tanıma problemi için Çizge-Tabanlı Endeksleme ve Eşleme" isimli proje kapsamında meydana gelmiştir. Bu proje süresince maddi desteklerinden dolayı TÜBİTAK'a, projeden sonra bana maddi ve manevi her konuda destek olan TOBB ETÜ'ye teşekkürlerimi sunarım.

Düzeltilmelerde çok emeği geçen Uğur Şahin'e ve 181 labındaki tüm arkadaşlara teşekkür ediyorum.

Ayrıca, her zaman yanımda olan ve her konuda bana destek olan eşim, Gulshat'a teşekkürlerimi sunuyorum.

Rahmetli Babam AKİMALI KASIMALİEV'e ithaf edilmiştir.

İÇİNDEKİLER

1 GİRİŞ	1
1.1 Problem tanımı	1
1.2 Bu tezin sağladığı katkı ve organizasyonu	6
2 ÖNCEKİ ÇALIŞMALAR	9
2.1 Kapalı yüzeyler	9
2.2 Şekil gramerleri	13
2.3 Fonksiyonel nesne tanımlamaları	18
2.4 Çizgeler	23
3 ÖZET ŞEKİL ÇIKARMA İŞLEMİ	28
3.1 Genel şekil tanıma problemi	28
3.2 Şeklin iskeletsel çizgesini çıkarma	30
3.3 Örneklerden iskeletsel özet oluşturma	31
4 ŞEKİL ÖZETİ İÇİN OPTİMAL ÇÖZÜMLERİ BULMA	37
4.1 Olası bir başlangıç çözümü belirleme	37
4.2 İlk optimal çözümü bulma	38

4.3	Alternatif optimal çözümleri bulma	38
5	ÖZET ŞEKİL ÇIKARMA	47
6	DENEYLER	49
6.1	Kullanılan veritabanları	49
6.2	Değerlendirme	49
7	ÖZET VE SONUÇLAR	53
	ÖZGEÇMİŞ	59

ŞEKİLLERİN LİSTESİ

- 1.1 Bu görüntüde iki tane insansı şeklin oturduğu ve birbirine baktığı heykel temsil edilmektedir. Yerel öznitelikler düzeyinde bakıldığında bu şekiller bir anlam taşımamaktadır. Ancak, daha özet (soyut) bir düzeyde bakıldığında, şeklin belirgin kısımları ortaya çıkmaya başlamakta ve böylece nesne sınıflandırma işlemi kolaylaşmaktadır. Özet kısımları oluşturan yerel öznitelikler örneklerden alınmamıştır ve daha önce hedef bir model (örn. insan) kullanılmamıştır. Bu görüntü [13] çalışmasından alınmıştır. 2
- 1.2 Şekil 1.1’de sağ tarafta oturan insansı figürün özet şekli. Görüntüdeki yerel özniteliklerin (dış hatların) özet şeklin sınırları ile bire bir denk (ya da tam hiza halinde) olmadığına dikkat edilmelidir. Bu görüntü [13] çalışmasından alınmıştır. 3
- 1.3 Birden fazla optimal çözümü bulunan taşıma problemi örneği. Birinci nokta dizisi, $P = \{p_1, \dots, p_5\}$, ikinci nokta dizisi, $Q = \{q_1, \dots, q_{10}, \dots, q_{19}\}$ ’un arasındadır. $i = \{1, \dots, 5\}$, $j = \{1, \dots, 19\}$ ve $k = \{1, \dots, 10\}$ için $d|p_i, q_i| = d|p_i, q_{i+10}| = d|p_5, q_{5+k}|$ ve $w(p_i) = w(q_j)$ ise, birden fazla optimal çözüm bulunmaktadır. 6
- 1.4 Birinci satırda (a), (b) ve (c) şıklarında üç tane optimal çözüm sonucu elde edilen üç tane farklı nokta tekabülleri görülmektedir. İkinci satırda bu tekabüller sonucunda oluşturulan üç tane değişik nokta dizisi ve son satırda ise Şekil 1.3’deki aynı girdi dizileri için bu nokta dizilerinden çıkartılan üç tane değişik özet şekil görülmektedir. Optimal çözümler aynı optimal değere sahip olmasına rağmen, tekabülleri farklıdır. 7

1.5	(a) ve (c) girdi şekilleridir ve bu şekillerden oluşan özet şekil ortada, (b)'de görülmektedir.	8
2.1	Düzlemsel dış hatlar ile temsil edilen araba ve gözyaşı damlası şekli [7].	10
2.2	Araba şeklini gözyaşı damlası şekli ile ağırlıklı özetleme (ortalama) (a) 70/30, (b) 50/50, ve (c) 30/70 [7].	12
2.3	Yirmi tane çay fincanı şekli [7].	13
2.4	Çay fincanı şekillerinin (a) ortası, (b) medyanı, ve (c) modu [7].	13
2.5	Görüntü ayrıştırma işlemi örneği gösterilmektedir. Ayrıştırma çizgesi dikey oklarla gösterilen yapısal ayrışmayı ve yatay oklarla gösterilen uzaysal ve fonksiyonel ilişkilerin sayısını içermektedir. [31] çalışmasından bir alıntıdır.	14
2.6	Görüntü ayrıştırmadaki özyinelemeli yukarıdan-aşağı/aşağıdan-yukarı hesaplama işlemleri gösterilmektedir. Mutfak sahnesinde dört tane dikdörtgen (kırmızı) aşağıdan-yukarı işleminden geçerek kabul edilmiştir ve sırası ile A, B ve C kutupsal olmayan düğümler tarafından temsil edilen üretim kuralları etkinleştirilmiştir. Böylece, yukarıdan-aşağı aramada adayların (mavi) sayısı tahmin edilir. Tam yukarı oklar aşağıdan-yukarı bağlamayı gösterirken, aşağı oklar ise yukarıdan-aşağı tahmini göstermektedir [36].	16
2.7	Kulplu bardak görüntüsü [33].	19
2.8	Kural oluşturma motoru tarafından belirlenen <i>if – then</i> kuralı [33]. . .	20
2.9	Bardak nesnesinin fonksiyonel tanımı. Bu anlamsal ağ İngilizce tanımlar kullanarak oluşturulmuştur. AKO = A Kind Of, HQ = Has Quality [33].	22
2.10	Fiziksel görünümün anlamsal ağını gösteren belli bir bardak örneği. Anlamsal ağın çoğu görüntüden ACRONYM kullanılarak çıkartılabilir. Bazı görünmeyen detaylar doğal dil arayüzü kullanılarak eklenebilir [33].	23

2.11	Çizgenin komşuluk matrisinin kromozom gösterimi: (a) asıl çizge, (b) komşuluk matrisi, ve (c) kromozom [15].	25
3.1	Kolay tanınan bazı şekil silüetleri.	29
3.2	Aynı varlığı temsil eden iki şekil.	29
3.3	Balık ve at şekli ve onların iskeletsel noktaları (şekillerin iç kısımları).	30
3.4	Bir dikdörtgen üzerinde ateşin yayılması örneği [4].	31
3.5	İki tane örnekten şekil prototipini çıkartma işlemi. İlk iki sütun değişik görüntü rotasyonuna ve parça eklememesine maruz kalmış iki tane örnek bölgeyi göstermektedir. Orta (medyan) dalları çizgedeki düğümleri oluşturmak için bölümlenmiştir (elips şeklinde) ve eşleme sonucu elde edilen düğüm tekabülleri aynı renk ile gösterilmiştir. Birinci sütundaki insan şeklinin tek parça sağ bacağı (şekil açısından) ikinci sütundaki insan şeklinin iki parça sağ bacağına eşleştiği gibi düğümler arasındaki tekabüller çoklu olabilir. Aynı şekilde, birinci sütundaki iki parça sol kol kısmı ikinci sütundaki tek parça sol kol kısmı ile eşleşmiştir. Üçüncü ve altıncı sütunlarda ikili örneklerden elde edilen özet şekil prototipleri görülmektedir [11].	32
3.6	P ve Q dağılımlarının verildiği bir taşıma problemi örneği. f_{ij} , p_i ve q_j noktaları arasındaki akışı ve d_{ij} uzaklığı göstermektedir. w_{p_i} (veya w_{q_i}), p_i (veya q_i)'nin ağırlığıdır.	33
3.7	Bir dizi çaydanlık görüntüsü için şekil prototiplerini oluşturma örneği. Asıl görüntüler en altta (alt düzey) gösterilirken, onların ortalamaları iç düğümleri belirtmektedir. İkili ortalama işlemi, prototipin soyunun herhangi birinden yeterince farklı oluncaya kadar devam ettirilir. Buradaki durumda, altı tane görünümünden iki tane prototip oluşturulmuştur [11].	36
4.1	İki tane dağılımdan oluşan taşıma problemi örneği, $P = \{(p_1, 30), (p_2, 20), (p_3, 40)\}$ ve $Q = \{(q_1, 10), (q_2, 20), (q_3, 30), (q_4, 30)\}$. Her (p_i, q_j) ikilisinin uzaklığı onları birleştiren doğru üzerinde verilmiştir.	39

6.1	56 tane şekilden oluşan Aslan ve Tari veritabanı. Her biri için 4'er şekil olmak üzere 14 tane sınıf vardır.	50
6.2	26 tane şekilden oluşan Rutgers Tools veritabanı (İlk satır 18 alet şeklini ve son satır 8 biyolojik şekli göstermektedir).	50
6.3	MPEG-7 veritabanı için bazı nesnelere gösterilmiştir. 40 tane sınıfın görüntüleri verilmiştir (toplam 70 tane sınıftan oluşmaktadır). . .	51
6.4	ETU veritabanı için 10 tane sınıfın görüntüleri verilmiştir.	51
6.5	Birbirine yakın iki tane nesne görüntüsü ve bu iki tane görüntüden oluşturulan özet görüntü gösterilmektedir. İlk iki ve son iki sütun farklı iki tane görüntüyü ve onların iskeletlerini temsil etmektedir. Ortadaki iki sütun ise bu iki tane görüntüden oluşturulan şekil özetini ve iskeletini ifade etmektedir.	52

ÇİZELGELERİN LİSTESİ

- 3.1 Şekil 3.6’de verilen taşıma problemi için oluşturulan parametre tablosu. f_{ij} , p_i ve q_j noktaları arasındaki akışı ve d_{ij} de uzaklığı göstermektedir. Her bir hedef noktasının ağırlığı tablonun aşağısında ve her bir kaynak noktasının ağırlığı tablonun sağ tarafında verilmiştir. δ_{ij} ve Δ_{ij} değişkenleri optimal çözüm hesaplanırken kullanılmıştır. Detaylar için Algoritma 1 ve Algoritma 2’ye bakınız. 35
- 4.1 Verilen problem için düzenlenen başlangıç parametre tablosu. Her hücrenin üst-sol köşesi maliyeti (uzaklık), üst-sağ köşesi maksimum uzaklıklar arasındaki farkı, alt-sol köşesi maliyet farkını ve alt-sağ köşesi her p_i ve q_j ikilisi arasındaki akışı belirtmektedir. 40
- 4.2 Russell’s approximation metodu kullanılarak δ değerlerinin hesaplanması. δ değerleri köşeli parantezlerde verilmiştir. 40
- 4.3 f_{31} ’e 10 değerinde yeni bir akış belirlendikten sonra. 41
- 4.4 Yeni bir akış belirlendikten sonraki değerlendirme. 41
- 4.5 Elde edilen olası başlangıç çözümü. Akışlar şu şekilde belirlenmiştir: $f_{12}:20$, $f_{13}:10$, $f_{24}:20$, $f_{31}:10$, $f_{33}:20$ ve $f_{34}:10$ 42
- 4.6 u_i (solda), v_j (yukarıda) ve Δ (parantezde) değerleri hesaplanırken. 42
- 4.7 En küçük Δ değerini belirleme (Δ_{21}) ve kapalı döngüyü oluşturma. Kapalı döngü $\pm\Theta$ ile işaretlenerek oluşturulmaktadır. Burada, kapalı döngü şu şekildedir: f_{21} , f_{24} , f_{31} ve f_{34} 43

4.8 Akış deęişiklikleri sonrasındaki Çizelge. f_{21} , f_{24} , f_{31} ve f_{34} akışları deęişmiştir.	43
4.9 Sonraki tekrarlama, Δ deęerlerini belirleme.	43
4.10 Son optimal çözüm. Bu çözüm için akış matrisi: $f_{12}:10$, $f_{13}:20$, $f_{21}:10$, $f_{23}:10$, $f_{32}:10$ ve $f_{34}:30$	45
4.11 Δ deęerlerini belirle, $\Delta_{ij} = 0$ alternatif optimal çözümü vermektedir. Δ_{11} ve Δ_{22} sıfır deęerindedir ve bu da demektir ki bu problemin iki tane alternatif çözümü mevcuttur.	45
4.12 Sıfır deęerli Δ_{11} için kapalı döngü oluşturma. Kapalı döngü şu şekildedir: f_{11} , f_{13} , f_{21} ve f_{23}	45
4.13 Şekil 4.1'de verilen problem için ilk alternatif optimal çözüm. Akışlar şu şekilde bulunmuştur: $f_{11}:10$, $f_{12}:10$, $f_{13}:10$, $f_{23}:20$, $f_{32}:10$ ve $f_{34}:30$	46
4.14 İkinci alternatif optimal çözüm bulunduktan sonraki akış matrisi: $f_{13}:30$, $f_{21}:10$, $f_{22}:10$, $f_{32}:10$ ve $f_{34}:30$	46

1. GİRİŞ

1.1 Problem tanımı

Nesne sınıflandırma işlemi bir görüntü içindeki bir nesnenin örneklerini bulma ve tanıma süreci olarak tanımlanabilir. Bu süreç genel nesne tanıma (generic object recognition) olarak da adlandırılmaktadır. Diğer yandan, özel nesne tanıma (specific object recognition) işlemi özel ve bireysel nesnelere ele almaktadır. Genel nesne tanıma örneği olarak insan, çocuk, köpek, bisiklet veya tencere gibi birkaç sınıf verilebilir. Özel nesne tanıma ise, "Albert Einstein" veya "benim arabam" gibi spesifik bir nesne ile örneklendirilebilir. Aslında, bilgisayarla görü çalışma alanının asıl amaçlarından biri genel nesne tanıma ve sınıflandırma olmasına [21] rağmen, bu hedefe hala ulaşamamıştır. Diğer taraftan, farklı pozlardaki ve iyi bir görüş alanı koşulları altındaki bilinen nesnelere tanıma işlemi önemli bir ilerleme kaydetmiştir [20].

Bir çok nesne sınıflandırma yöntemi vardır ve onlardan biri de şekil tabanlı nesne sınıflandırma yöntemidir. Şekil tabanlı yöntemlerde nesnelere alan, silüetleri, kuşatan kutusu ve bulunan alanın gradyanı gibi iki boyutlu bilgiler kullanılmaktadır. Şekil tabanlı nesne sınıflandırma metodlarından biri de şekil özeti oluşturmaktır ve araştırmacıların karşılaştıkları en önemli problemlerden biridir.

Şekil tabanlı yerel (lokal) öznitelikleri (dış hat ya da bölgeler) bir araya getirme işlemi, 1960'lerde, şekil tabanlı nesne tanıma alanında çalışan araştırmacıların kabul ettiği ilk ve önemli bir adımdır [13]. Şekil 1.1'de örnek bir görüntü verilmiştir. Bu görüntü insan gözüyle bakıldığında, iki tane insansı şeklin (figürün) oturduğu ve birbirine baktığı hemen anlaşılmaktadır. Fakat, nesne sınıflandırma



Şekil 1.1: Bu görüntüde iki tane insansı şeklin oturduğu ve birbirine baktığı heykel temsil edilmektedir. Yerel öznitelikler düzeyinde bakıldığında bu şekiller bir anlam taşımamaktadır. Ancak, daha özet (soyut) bir düzeyde bakıldığında, şeklin belirgin kısımları ortaya çıkmaya başlamakta ve böylece nesne sınıflandırma işlemi kolaylaşmaktadır. Özet kısımları oluşturan yerel öznitelikler örneklerden alınmamıştır ve daha önce hedef bir model (örn. insan) kullanılmamıştır. Bu görüntü [13] çalışmasından alınmıştır.

işlemi için bu o kadar da kolay bir iş değildir. Şekildeki yerel özniteliklere tek tek bakıldığında, görünüm ve biçim açısından şüphesiz ki birbiriyle alakasızdır. Ancak, bu yerel öznitelikler bir araya getirilip özetlendiğinde, belirgin kısımlar ve biçimler ortaya çıkmaya başlar ve daha önce görülmemiş örnek bir nesnenin (bu durumda çok çarpık bir insan heykeli) bilinen bir sınıf (insan) olarak tanınmasını kolaylaştırır.

Görüntü (ya da öznitelik) özeti oluşturma işlemi görüntü öznitelikleri kümesi çıkarma işlemi ile başlar ve daha sonra bunun üzerinden özet oluşturulur. Eğer oluşturulan özet, kısım tabanlı ise (görüntüde oluşabilecek üst üste oturma ya da karışıklık olması halinde nesne tanıma için gerekli olan temsil lokallığı sağlanacak şekilde), yerel özniteliklerin algısal olarak, özet kısımlarla örtüşecek (map) şekilde koleksiyonlar halinde gruplandırılması gerekmektedir. Özniteliklerin grup haline getirilebilmesi için aralarında doğal (non-accidental) ilişkilerin olması gerekmektedir. Bu ilişkiler görüntü tabanlı (renk ve doku yakınlığı) olabilir,



Şekil 1.2: Şekil 1.1’de sağ tarafta oturan insansı figürün özet şekli. Görüntüdeki yerel özniteliklerin (dış hatların) özet şeklin sınırları ile bire bir denk (ya da tam hiza halinde) olmadığına dikkat edilmelidir. Bu görüntü [13] çalışmasından alınmıştır.

ancak, görüntü özelliği sınıf genelleştirmesi için çok nadir kullanılmaktadır. Örneğin, görüntüdeki heykel başka renge boyanmış ya da çizgiler ve lekeler ile dokunmuş olsaydı, tanıma işlemi etkilenmemiş olurdu.

Gelişigüzel (tesadüf olmayan) birbirine bağlı öznitelikleri bir araya getirme nesne sınıflandırma işlemi için gereklidir ama yeterli değildir. Şekil 1.1’de, iki tane insansı şeklin herhangi birindeki gövde kısmını oluşturan çeşitli yerel öznitelikleri bir araya getirme işlemi gerçekten de son derece zor ve önemli bir problemdir. Görüntü iyileştirilerek belirgin şekil özniteliklerin bir araya getirildiğini kabul edersek, tipik bir tanıma sisteminin gruptaki belirgin görüntü öznitelikleri ve belirgin model öznitelikleri arasında bire bir tekabül oluşturması beklenir. Ancak, asıl problem burada yatmaktadır. Yerel görüntü öznitelikleri (nokta, yama, dış hat ya da bölge gibi) arasında bire bir tekabüllerin olduğunu kabul edersek, bu, oluşan modelin görüntünün şablonu olmasından fazla bir şey katmayacaktır.

Gövdeyi oluşturan yerel öznitelik grubu ve sezgisel bir insan modelinin gövde "kısmı" arasındaki asıl tekabül yerel görüntü öznitelikleri düzeyinde değil, daha çok şekil özniteliklerinin özet düzeyinde yatmaktadır. Örneğin, Şekil 1.2’de görüldüğü gibi, oturan bir insan modelinin böyle bir özeti verilmiştir ve eliptik bölüm ile gösterilen kısım gövdeye denk gelmektedir. Bire bir tekabül olduğu varsayılırsa, heykelin gövdesini oluşturan birçok sayıda yerel öznitelik (uzun, "belirgin" dış hatlar dahil olmak üzere), modelin gövdesi ile tekabül etmeden

önce özetinin oluşturulması gerekmektedir. Bu özet, görüntüde, açık olarak görülmemektedir (özet, bir araya getirilmiş görüntü özniteliklerinin altkümesi değildir) ve böyle özet oluşturma açık bir model gerektirdiği gibi (bu durumda, eliptik bir şekil), model, herhangi bir nesne ya da sahne düzeyinde bir bilgi olmadığını varsaymaktadır.

Şekil özeti oluşturma problemi yeni bir problem değildir ve sınıflandırma alanındaki, nesne şeklinin yüksek düzeydeki özetini çıkartarak modelleri oluşturan araştırmacıların daha yaygın olarak kabul ettiği (ancak daha etkin bir şekilde çözülemeyen) ve üzerinde çalışıldığı bir problemdir. Son 40 yıl içerisinde, gerçek görüntülerden şekil özetini etkin bir şekilde elde edememe, araştırmacıları giderek daha az özet oluşturma gerektiren ya da hiç gerektirmeyen nesne tanıma alanına sürüklemiştir. Bu evrimi anlamak, bizi özet oluşturma problemine odaklandırmakla kalmayıp, şimdiye kadar, sınıflandırma alanındaki araştırmacıların sağladığı birçok katkıyı belirlememize yardımcı olur.

Bu problem, şimdiye kadar, birçok farklı bağlamda incelenmiştir. Kapalı yüzeyler (closed surfaces) [7], şekil gramerleri (shape grammars) [36], fonksiyonel nesne tanımlamaları (functional object descriptions) [33] ve çizgeler (graphs) [15] gibi yaklaşımlar bu çalışmalardan sadece bazılarıdır. Bu teknikler Bölüm 2’de detaylı olarak anlatılacaktır.

Daha önce bahsedildiği gibi, bu problem bazı çalışmalar tarafından geniş olarak incelenmiştir. Ancak, bu teknikler gerçek nesnelerin görüntülerinden şekil özetlerini etkin bir şekilde elde edememektedir [13]. 1970’lerde, bu problem ile ilgili çok önemli birkaç ilke ortaya çıkmıştır:

1. Nesne sınıfını tanımlamada şeklin (örn. dış hat) önemi.
2. Bakış açısından bağımsız olan 3 boyutlu temsilin önemi.
3. Simetrisinin ve öznitelik gruplamadaki diğer doğal (non-accidental) ilişkilerin önemi.
4. Etkin indeksleme (sorgu olarak hesaplanabilecek nesne modellerinden adayları seçme süreci), nesne boğumlanması ve birbirine girmiş nesnelere tanımda yardımcı olması için paylaşılabilir parçalardan oluşan dağıtılmış

temsilere ve modelleme karmaşıklığını gidermeye yardımcı olan, onlar arasındaki ilişkilere olan gereklilik.

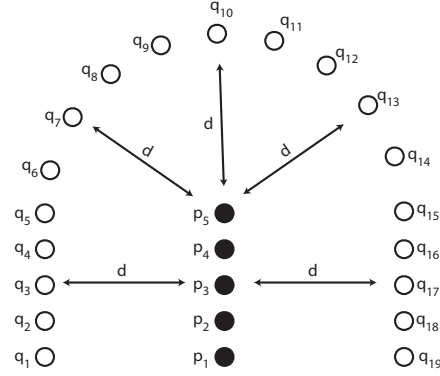
5. Kısım/bütün hiyerarşi ve özet hiyerarşiler dahil olmak üzere, hiyerarşik temsillere olan gereklilik.
6. Büyük veritabanlarda ölçeklenebilirliğe olan gereklilik, yani, "bulma" ve hedef tanıma problemi daha genel tanıma (büyük veritabanında) probleminin özel bir durumudur ve büyük veritabanı üzerinde lineer arama (bir nesne için bir detektör) kabul edilir bir durum değildir;
7. Değişken yapıya olan gereklilik, yani, bir sınıfa ait örneklerin parçalarının (kısımlarının) sayısı, kimlikleri ve eklentileri örnekler arasında değişebilir.

Bir nesnenin şeklini veren görüntüdeki dış hatlar görünümünden daha çok sınıf için genel olan bir özneliktir. Ancak, daha genel olan özneliklerdeki problem onların artan belirsizlikleridir. Küçük bir dış hat parçası (örn. dış hatlardaki kesiklikleri ayırtmada eğri parçalara ayrılırken meydana gelen) için sınıfa özel çok az bir bilgi taşımaktadır. Daha önce belirtildiği gibi, çözüm, birbirine yakın ve ilişkili dış hatları daha belirgin yapılar haline gelecek şekilde bir araya getirmektir.

Dış hatların ne kadar belirgin yapılacağı tamamen probleme bağlıdır. Probleme göre birinde en az gruptandırma işlemi ile hedefe ulaşılırken, diğerinde daha çok, alandan bağımsız olacak şekilde etkili bir gruptandırma işlemi yapmak gerekmekte ve problemin karmaşıklığı artmaktadır. Bir diğer önemli sorun da, büyük bir veritabanı üzerinde tanıma için genel sınıflandırma işlemi yapmaktır. Görüntü dış hatları ile çalışırken, etkili bir biçimde büyük bir veritabanı birkaç tane adaya indirgeyecek şekilde onları daha büyük, daha belirgin indeksleme yapıları haline dönüştürülmesi gerekmektedir.

Modellerin eklemmeden bağımsız olması için indeksleme yapılarının doğal olarak nesnenin kısımlarına tekabül etmesi gerekmektedir. Ayrıca, kısım boyutları azaltılarak sınıflar arasında parçaların aynı özelliği taşıması için bir şekilde kısımlardaki belirsizliği azaltmak, yani indeksleme yapılarının güçlerinin artırılması gerekmektedir. Bu demek oluyor ki, indeksleme yapıları yeterli oldukça kısımların bir araya getirilmesi kolaylaşmaktadır.

Son zamanlarda, görüntü ve şekil özeti oluşturma mekanizmalarının geliştirilmesi



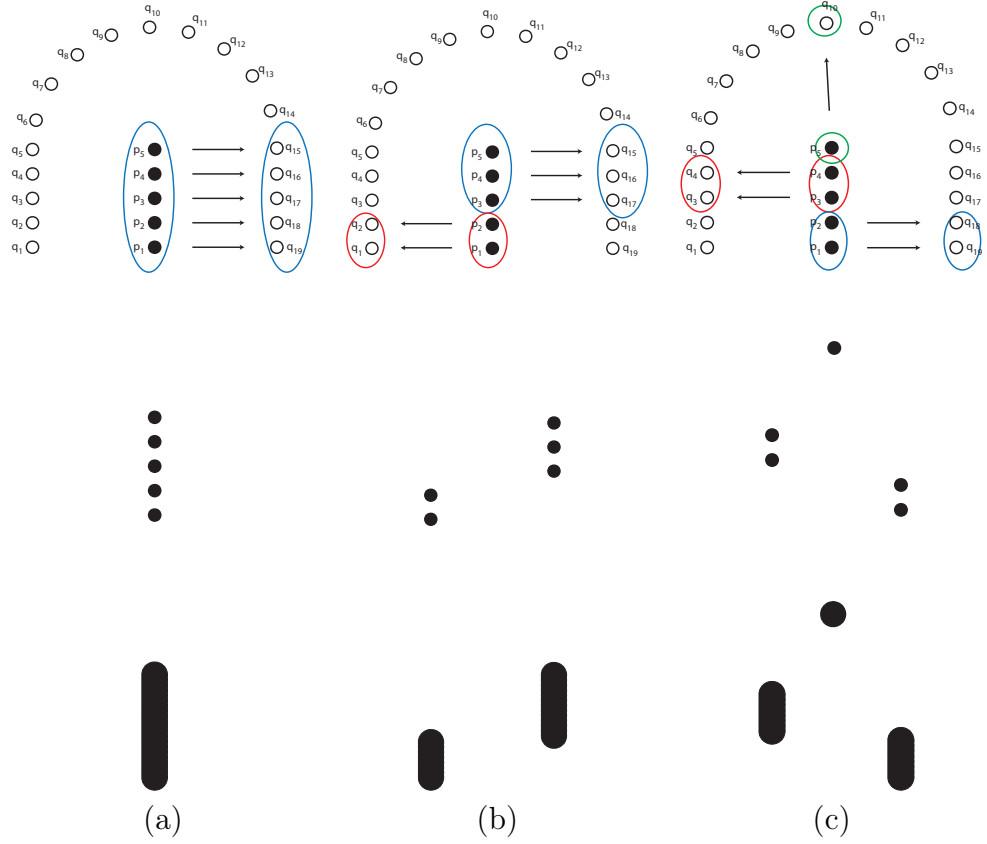
Şekil 1.3: Birden fazla optimal çözümü bulunan taşıma problemi örneği. Birinci nokta dizisi, $P = \{p_1, \dots, p_5\}$, ikinci nokta dizisi, $Q = \{q_1, \dots, q_{10}, \dots, q_{19}\}$ 'un arasındadır. $i = \{1, \dots, 5\}$, $j = \{1, \dots, 19\}$ ve $k = \{1, \dots, 10\}$ için $d|p_i, q_i| = d|p_i, q_{i+10}| = d|p_5, q_{5+k}|$ ve $w(p_i) = w(q_j)$ ise, birden fazla optimal çözüm bulunmaktadır.

yerine, görüntü ilk önce özet hiyerarşisinde yukarı doğru (özeti simüle ederek) ve sonrasında oluşan modeli özet hiyerarşisinde aşağı doğru (sınıflandırmadan uzaklaştırarak) işlem yapılarak, özet oluşturma işleminde açıklık oluşmuştur. Ancak, son çalışmalar açık bir şekilde sınıflandırmaya doğru yol almaktadır. Ve yeni geliştirilen yöntemler özet oluşturma açığını bir şekilde kapatmaya çalışıyor olsa bile, görüntü öznitelikleri model özniteliklerine sıkı bir biçimde bağlıdır ve kritik olan özet oluşturma problemine olan ilgi yetersiz kalmaktadır.

1.2 Bu tezin sağladığı katkı ve organizasyonu

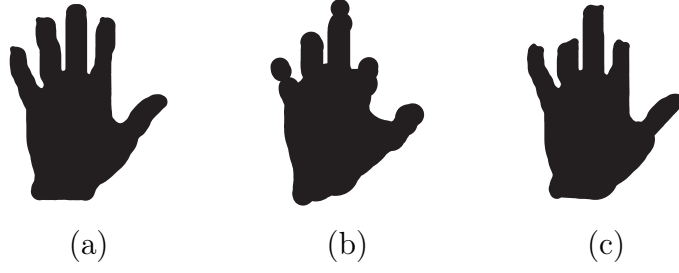
Demirci ve ark. [10] öznitelikleri çoklu tekabül eden bir örnek dizisinden şekil özeti prototipini elde etmek için yeni bir yöntem sunmuştur. Girdi örnekleri çizge olarak temsil edildikten sonra, çizge çiftleri arasındaki düğüm tekabülleri [11] çalışmasındaki eşleme metodu kullanılarak hesaplanmaktadır. Daha sonra, bu tekabüller kullanılarak özet çizge elde edilmektedir. Bu özet çizgedeki düğüm noktalar tekabül eden öznitelik dizilerini ve kenarlar özetler arasındaki özellikleri göstermektedir.

Yöntemin [10] güçlü olmasına rağmen, etkinliği elde edilen nokta tekabüllerine bağlı kalmaktadır. Elde edilen tekabüller iyi bilinen taşıma problemi [26] tarafından bulunan optimal çözüme dayanmaktadır. Yüksek boyutlu uzayda verilen



Şekil 1.4: Birinci satırda (a), (b) ve (c) şıklarında üç tane optimal çözüm sonucu elde edilen üç tane farklı nokta tekabülleri görülmektedir. İkinci satırda bu tekabüller sonucunda oluşturulan üç tane değişik nokta dizisi ve son satırda ise Şekil 1.3'deki aynı girdi dizileri için bu nokta dizilerinden çıkarılan üç tane değişik özet şekil görülmektedir. Optimal çözümler aynı optimal değere sahip olmasına rağmen, tekabülleri farklıdır.

büyük nokta dağılım dizileri için bu dağılımlar arasındaki taşıma probleminden bulunan global optimal çözüm sayısı birden fazla olabilir. Bütün optimal çözümler aynı optimal değere sahip iken, nokta tekabülleri farklı olabilir. Şekil 1.3'teki örnek şekilde görüldüğü gibi, $P = \{p_1, \dots, p_5\}$, verilen birinci nokta dizisidir ve $Q = \{q_1, \dots, q_{10}, \dots, q_{19}\}$, verilen ikinci nokta dizisinin ortasında bulunmaktadır. Birinci dizideki her noktanın, ikinci dizide tekabül ettiği noktaya aynı uzaklıkta olduğu ve her noktanın aynı [universal] ağırlığa sahip olduğu varsayılırsa, yani, $i = \{1, \dots, 5\}, j = \{1, \dots, 19\}$ ve $k = \{1, \dots, 10\}$ için $d|p_i, q_i| = d|p_i, q_{i+10}| = d|p_5, q_{5+k}|$ ve $w(p_i) = w(q_j)$ ise, aynı optimal değere sahip birden fazla optimal çözüm bulunmaktadır. Bunun gibi üç tane çözüm ile elde edilen üç tane değişik nokta tekabülü ve çıkarılmış özetler Şekil 1.4'te verilmiştir.



Şekil 1.5: (a) ve (c) girdi şekillerdir ve bu şekillerden oluşan özet şekil ortada, (b)'de görülmektedir.

[10]'da şekil özeti [11]'deki eşleme metodu tarafından bulunan tek bir optimal çözüme dayanarak oluşturulmuştur. Eşleme metodunun birden fazla optimal çözüm verebileceği durumu göz önüne alınmamıştır. Bu eksiklik hesaba katılarak, bu tez çalışmasında bütün optimal çözümler kullanılarak bir önceki şekil özeti oluşturma algoritması genişletilmiştir. Bütün optimal çözümler, noktaların ve onların tekabüllerinin yerel komşuluk ilişkilerini ne kadar koruduğuna dayanarak derecelendirilmiştir. Daha sonra en yüksek dereceye sahip çözüm kullanılarak şekil özeti oluşturulmuştur. Şekil 1.5'de iki tane girdi şekli ve bu şekillerden elde edilen şekil özeti görülmektedir.

Deneyssel değerlendirme, önerilen yaklaşımın bir önceki şekil özeti oluşturma tekniğine göre daha iyi sonuçlar verdiğini göstermektedir. Bu çalışmanın ilk versiyonu [3]'de yayınlanmıştır.

Bu tezin geri kalan kısmı şu şekilde organize edilmiştir; Bölüm 2'de önceki çalışmalar anlatılmıştır. Bölüm 3'te ise sunulan şekil özeti oluşturma metoduna yer verilmiştir. Bölüm 4'te bulunan bütün optimal çözümleri derecelendirme algoritmasından bahsedilmiştir. Özet şekli çıkarma işlemi Bölüm 5'te anlatılmıştır. Bölüm 6'da yapılan deneyler sunulduktan sonra, son olarak, Bölüm 7 özet ve sonuçlara bırakılmıştır.

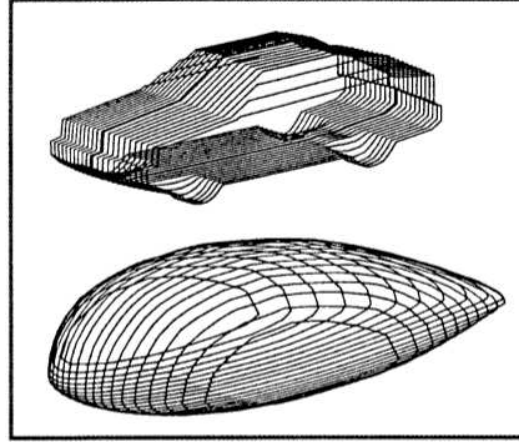
2. ÖNCEKİ ÇALIŞMALAR

Şekil özeti çıkarma problemi, şimdiye kadar, birçok farklı bağlamda incelenmiştir. Bu bağlamların bazıları şunlardır: kapalı yüzeyler(closed surfaces) [7], şekil gramerleri(shape grammars) [36], fonksiyonel nesne tanımlamaları(functional object descriptions) [33] ve çizgeler(graphs) [15]. Bu teknikler bu bölümde detaylı olarak anlatılmıştır.

2.1 Kapalı yüzeyler

Ürünlerin birçok türünde, ürünün zamanla kalıbının değişikliğe uğraması bir çeşit düzenli değişkenliği göstermektedir. Bu değişiklik normalde kademelidir ve eğilim olarak bilinmektedir. Eğilimler, bir dönemin sosyal, kültürel ve teknik evrimini yansıtmaktadır. Tasarımcıların kalıpta olan eğilimleri takip etmesi ya da araştırması için ideal bir araç gerekmektedir. Bu motivasyondan yola çıkarak Chen ve ark. şekil özeti oluşturma işlemi sunmuştur ve uygulamıştır [7].

Şekil özeti oluşturma işlemi bir grup şekilden özgün bir temsil çıkarma sürecidir. Bu işlem, tasarımcıları için özellikle iki yönden yararlıdır: birincisi kalıp analizi aracı olarak ve ikincisi kalıp oluşturma aracı olarak kullanılabilir. Birinci kullanım, şekil özeti oluşturma işleminden elde edilen sonuç asıl şeklin özelliklerini korur varsayımına dayanmaktadır. Bu varsayım, kalıp eğilimini tahmin ve analiz etmede şekil özeti oluşturma işleminin kullanılmasını mümkün hale getirmektedir. Kalıp oluşturma işleminde ise, şekil özeti oluşturma işlemi çok değişik şekillerin özeti olarak yeni nesnelere oluşturmak için kullanılabilir. Böylece, tasarımcı ilkel şekilleri birleştirerek daha gelişmiş çözümler ortaya koyabilir ve elde edilen sonuçlar geliştirilmekte olan kalıp için yeni bir bakış açısı sunabilir. Örneğin,



Şekil 2.1: Düzlemsel dış hatlar ile temsil edilen araba ve gözyaşı damlası şekli [7].

araba ve gözyaşı damlası şekilleri özet oluşturma işleminden geçirilerek araba gövdesi tasarımının aerodinamik yönleri geliştirilebilir. Örnek için Şekil 2.1'e bakınız.

[7]'de şekil özeti oluşturma probleminin nesne modeli düzeyinde çözülmesi amaçlanmıştır. Girdi olarak 2 boyutlu düzlemsel çokgenler veya bir dizi düzlemsel dış hat ile temsil edilen 3 boyutlu nesnelere alınmıştır. Özet oluşturma süreci iki adımdan oluşmaktadır; bir grup şekil arasında düğüm tekabülleri kurulur ve bu tekabüller kullanılarak şekil özeti oluşturulur. Tekabüller şekiller arasındaki eşlemeyi kurar ve şekillerin hangi kısımlarının özetlenmesi gerektiğini belirler. Tekabüller sözdizimsel ve anlamsal olarak iki düzeyde kurulabilir ve bu çalışma sözdizimsel düzeyi kullanmıştır. Sözdizimsel düzeyde, şekiller birer geometrik varlık (entity) olarak ele alınmaktadır ve özet oluşturma işlemi boyut ve uzaklık gibi nesnelere ait geometrik özellikleri üzerinden yapılmaktadır.

Şekil özeti oluşturma probleminde tekabülleri belirleme işlemi biraz belirsiz bir süreçtir. Buna sebep de tekabül bulma işlemi insan yargısını içeren bir konudur; şekillerin nasıl eşleşmesi gerektiği ile ilgili iyi anlaşılması ve evrensel olarak geçerli bir tanım bulunmamaktadır. Bu çalışmada kullanılan tekabülleri bulma metodu, iki tane şeklin ortalamasını (özetini) iki girdi şekil ile sınırlı olan silindirik hacmin orta bölümü olarak belirler. Yüzey rekonstrüksiyon metodları optimal sonuçlar üreten ve buluşsal metodları içeren olarak iki sınıfa ayrılır ve bu çalışmada buluşsal metodları içeren yaklaşım kullanılmıştır. Bu yaklaşımda şekil uyumluluğu problemi vardır ve ikiden fazla şekle uygulanamamaktadır.

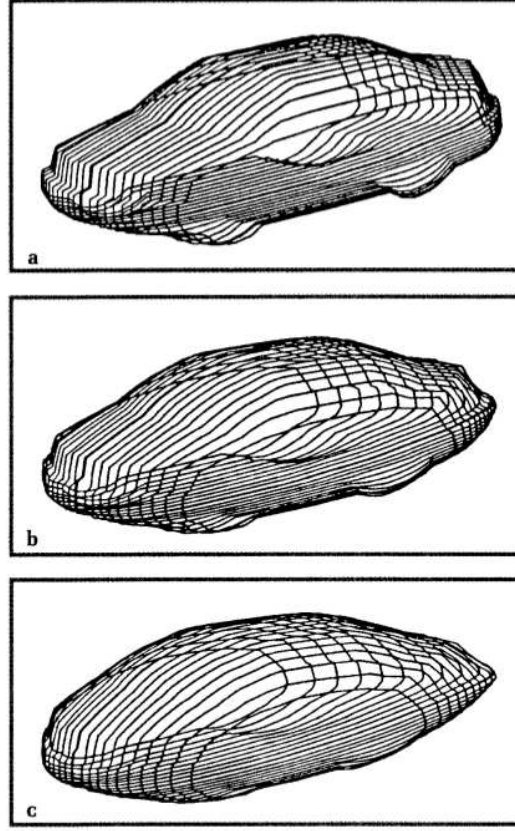
Art arda olan dış hat döngüleri genellikle çok fazla değişiklik göstermediğinden şekil uyumluluk problemi yüzey rekonstrüksiyon metodunu fazla ilgilendirmemektedir.

İkiden fazla şekli özetleme (ortalama) problemi şekiller arasında dolaylı olarak bire bir tekabülleri oluşturma ile çözülebilir. Nirengi (triangulation) metodunda, bir dış hat üzerindeki her düğüm diğer dış hat üzerindeki birkaç düğüme tekabül edebilir. Bu çalışmadaki yaklaşımda, ikiden fazla şekil üzerinde işlem yapmak için diğer tüm şekiller ile eşleşmesi için düğüm sayısı en fazla olan bir taban şekil seçilmiştir. Her eşleşme oluşturulduktan sonra bütün şekillerin düğüm sayısı taban şeklin düğüm sayısına eşit olacak şekilde işlem yapılmaktadır. Bu işlem dolaylı olarak tekabülleri tespit eder ve özet (ortalama) oluşturma işlemindeki tekabül bulma işlemini bir önışleme olarak yapmaktadır.

Bire bir ilişkinin sağlanması için ek bir geçiş yapmak gerekmektedir. İlk geçişte, diğer dış hattaki eşleşmenin bulunması için fazla düğümü bulunan dış hattın her bir düğümü üzerinden sırayla geçilmektedir. Birbirine en yakın uzaklıkta olan düğümler eşlenmektedir. Bu geçişten sonra ilk dış hattaki her düğümün tam birer tekabülü bulunmuş olacaktır. Bir sonraki geçişte ise daha az düğüm bulunan dış hat incelenecektir. Şayet, bir düğümün birden fazla tekabülü varsa, en iyi eşleşmenin seçilmesi için onun her tekabülü incelenmektedir. Bu karara yine uzaklıklar karşılaştırılarak varılmaktadır. Geri kalan tekabüller için dış hat üzerinde her biri için benzersiz eşleme olacak şekilde birer tane yeni düğüm eklenmektedir. Yeni düğümlerin pozisyonları eskilerle orantılı olacaktır.

Bu algoritma sadece bir döngülü geçiş sağlanabilen şekiller üzerinde çalışabilmektedir. Şeklin dalları varsa, bir döngülü geçiş sağlanabilen şekle dönüştürülmesi gerekmektedir. Düğümlerinin dağılımı düzgün olmayan, aynı sayıda düğümü bulunan şekiller için optimal bir şekilde çalışmamaktadır. Eğer aynı sayıda düğümleri bulunan girdi şekilleri verildiyse, algoritma yeni bir ekleme yapmayacaktır ve sırası ile düğümler üzerinde eşleme yapacaktır.

Tekabüller bulunduktan sonra şekiller özetleme (ortalama) işlemi için hazır durumdadır. Özetleme süreci aynı anda her bir tekabül (her şekilden tekabül eden noktalar grubu) dizisi üzerinde işlem yapma ve o diziden özet noktayı çıkarmayı gerektirmektedir. Özet şekli oluşturmak için bütün dizilerden çıkartılan özet

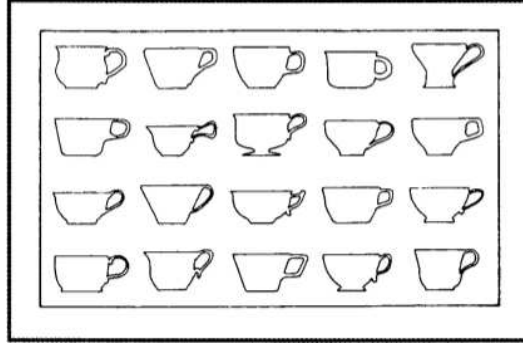


Şekil 2.2: Araba şekli gözyaşı damlası şekli ile ağırlıklı özetleme (ortalama) (a) 70/30, (b) 50/50, ve (c) 30/70 [7].

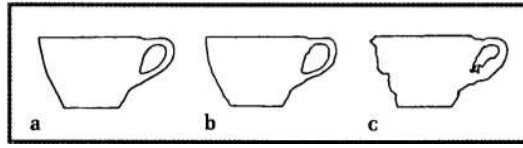
noktalar orijinal şekildeki gibi birbirine bağlıdır (yani, orijinal şekildeki topoloji bilgisi özet şekilde de korunmaktadır). Orta, medyan ve mod gibi üç çeşit özet elde edilebilmektedir.

Orta şekil tekabül eden noktaların koordinatlarının aritmetik ortalamasından elde edilmektedir. Medyan şekil ise ayrı ayrı x , y ve z doğrultularında sıralama ile ve her bir sıralamanın medyan değerini toplama ile bulunmaktadır. Mod şekil de her bir nokta için aralık atama ile hesaplanmaktadır. Bu aralık iki noktanın üst üste gelmemesi için arasındaki aralığı kontrol etmek için kullanılmaktadır. Üç çeşit şekil örneği için Şekil 2.4'e bakınız.

Şekil özetlerini aynı ağırlıkta elde etmenin dışında, her bir şekil için isteğe bağlı olarak bir ağırlık verilebilmektedir. Verilen ağırlığa bağlı olarak ara değeri hesaplama (interpolate) ya da tahmin etme (extrapolate) gibi bir dizi sonuç elde edilebilmektedir. İkidenden fazla şekil üzerinde ara değeri hesaplama ya da tahmin etme işlemini gerçekleştirmek için her bir tekabül dizisi için eğri (curve-fitting)



Şekil 2.3: Yirmi tane çay fincanı şekli [7].



Şekil 2.4: Çay fincanı şekillerinin (a) ortası, (b) medyanı, ve (c) modu [7].

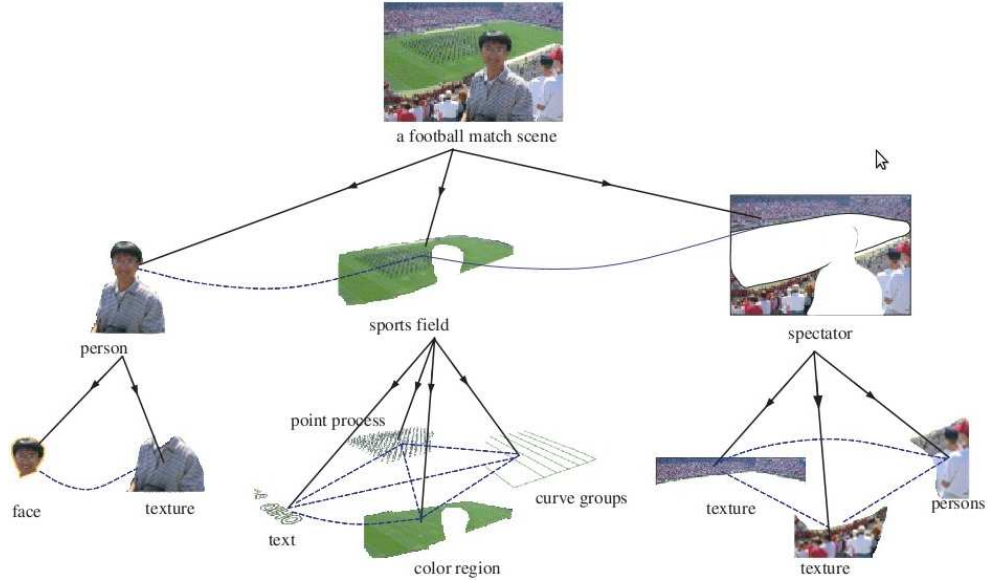
teknikleri uygulanabilir. Ara değeri hesaplama işlemi bir şekilden diğer şekle olan akıcı (düzgün) geçişi göstermektedir. Tahmin etme işlemi ise şekillerin eğilimini görmeye yardımcı olabilir. Ağırlıklı özetleme örneği olarak Şekil 2.2 verilebilir.

2.2 Şekil gramerleri

Bilgisayarla görü'de, 1960 -1970 yılları arasındaki çalışmalardan başlayarak, görüntülerin içeriğini anlamak her zaman ana problemlerden biri olmuştur. Şekil 2.5'te görüldüğü gibi, görüntü ayrıştırma işlemi [31], doğal dil anlayışına benzetilerek, girdi görüntüsünün en muhtemel tercümesi olarak ayrık çizgeyi çıkarmaktır. Bu ayrık çizge sahnenin içeriği için ağaç yapısındaki bir ayrışmayı ve hiyerarşinin her düzeyindeki bağlamlar için düğümler arasındaki uzaysal ve fonksiyonel ilişkilerin sayısını içermektedir.

Görüntü ayrıştırma ve yapılandırılmış yöntemler 1970'lerde çok büyük zorluklarla karşılaşmış, ve o zamandan bu yana çalışmalara yer verilmemiştir. Sözdizimsel ve grammer çalışmaları çoğunlukla perde arkasında ele alınmıştır. Bu zorluklar bugün bile zorluğunu korumaktadır.

1.Problem: Sağlam çıkarımda bulunabilmek için bilgisayarda temsil edilmesi



Şekil 2.5: Görüntü ayrıştırma işlemi örneği gösterilmektedir. Ayrıştırma çizgesi dikey oklarla gösterilen yapısal ayrışmayı ve yatay oklarla gösterilen uzaysal ve fonksiyonel ilişkilerin sayısını içermektedir. [31] çalışmasından bir alıntıdır.

gereken gerçek dünya sahneleri ile ilgili büyük miktarda görsel bilgi mevcuttur. Örneğin, kullanım için en az 3,000 tane nesne kategorisi vardır ve birçok kategoride geniş içkategorik yapısal varyasyonlar mevcuttur. Burada, anahtar sorular şunlardır: nesne kategorisi nasıl tanımlanır, araba mı ya da ceket mi? ve bu kategoriler tutarlı bir çerçeve içerisinde nasıl temsil edilir?

2.Problem: Hesaplama karmaşıklığı çok büyük. Şekil 2.5'e bakılırsa, girdi görüntüsünün çok sayıda nesne içerebildiği görülmektedir. İnsanın görme özelliğinin [30] sahne sınıflandırmadan kenar algılama işlemine kadar aynı anda her düzeyde hesaplama yapabildiği ve sağlam hesaplama yapabilmek için birden fazla görsel rutinleri benimsediği bilinmektedir. Bu tüm işlemler çok kısa bir sürede ($\leq 400ms$) gerçekleşmektedir. Buna karşılık, birçok örüntü tanıma ve makine öğrenme algoritması ileri beslemelidir ve bilgisayarla görü sistemleri çıkarım için nadiren yeterli görsel bilgiye sahiptir.

Burada, anahtar sorular şunlardır: binlerce kategoriye hitap edebilen sağlam bir hesaplama nasıl başarılabilir? Ve bu aşağıdan yukarı ve yukarıdan aşağı prosedürler nasıl koordine edilebilir? Sağlam bir hesaplama yapabilmek için görü algoritmaları bütün kategoriler için genel olan basit işlemler ve yapılara dayanmalıdır.

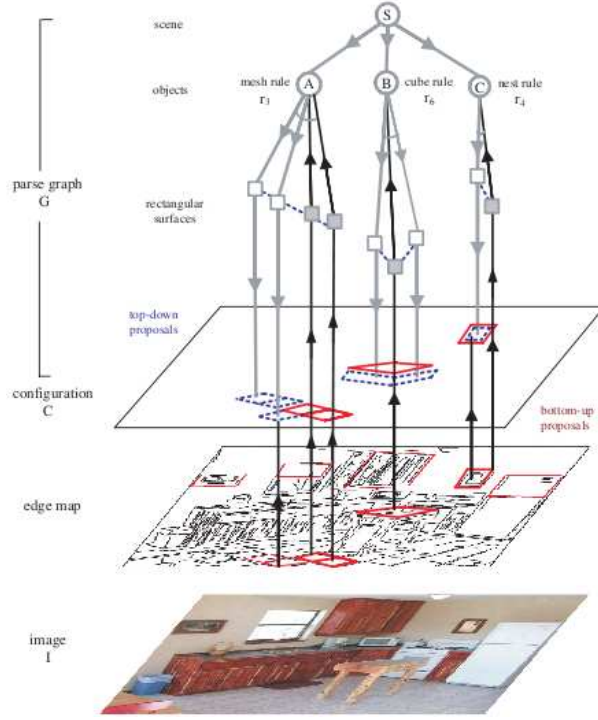
3.Problem: Görüntü ayrıştırma çalışmalarını pasif duruma düşüren en belirgin neden, erken sözdizimsel ve yapısal metotlardaki işlenmemiş pikseller ve sembolik işaret temsilleri arasındaki anlamsal boşluk olmuştur. Yani, işlenmemiş görüntülerden semboller tam olarak çıkartılamaz anlamına gelmektedir. Bu nedenden dolayı, bu alandaki çalışmalar zamanla görünüme dayalı metotlara doğru kaymıştır.

Görünüm tabanlı yöntemler ve makine öğrenme algoritmalarının dikkate değer bir ilerleme kaydetmesine rağmen, onların kendiliğinden gelen problemleri yapısal metotlarla tamamlanabilir. örneğin, onlar dizgisel ve üretken yapıların olmasından çok fazla ayrıştırma örnekleri gerektirir. Genellikle, onlar spesifik ayrıştırma setlerine olması gerektiğinden daha çok uymaktadır ve yeni örnekler veya özellikle çok fazla içkategorik varyasyona sahip yapılandırmaya kolaylıkla genelleme yapılamamaktadır.

Bu çalışma, temsil etme, öğrenme ve tanıma için birleşik bir çerçeve olarak stokastik ve bağlama duyarlı bir grameri tanıtmıştır. Bu çerçeve bir önceki bölümde bahsi geçen problemleri ve birçok mevcut modeli ve algoritmayı bütünleştirmektedir.

1.Hedef: Görsel bilgi gösterimi ve nesne kategorizasyonu için ortak bir çerçeve. Dil bilgisi alanında çalışılan gramerler, çok geniş yapılandırma ve örnekler dizisi üreten anlatım gücü için daha çok bilinmektedir. Örneğin, nispeten daha küçük kelimeler oluşturan dilin, üretim kuralları kullanan, paylaşılan ve kullanılabilir elemanları.

Bu çalışmada görüntü grameri Ve-Veya çizge temsilinde somutlaştırılmıştır. Her Veya-düğüm noktaları alternatif alt yapılandırmalarını göstermektedir ve Ve-düğümüleri ise bir dizi bileşene ayrıştırılır. Bu Ve-Veya çizgesi hem hiyerarşik ayrışmaları, hem de uzaysal ve fonksiyonel ilişkiler için bağlamları temsil etmektedir. Hiyerarşik ayrışmalar sahnelerden başlayıp nesnelere, parçalar, primitifler ve kutupsal ve kutupsal olmayan düğümlerden oluşturulan piksellerden elde edilmiştir. Bu, üretim kurallarını temsil etmek için alternatif bir yoldur ve tüm olası ayrışma ağaçlarını içermektedir. Daha sonra, maksimum olabilirlik kestirimi kullanılarak örneklerden öğrenilebilen Ve-Veya çizgesi için olasılıksal



Şekil 2.6: Görüntü ayrıştırmadaki özyinelemeli yukarıdan-aşağı/aşağıdan-yukarı hesaplama işlemleri gösterilmektedir. Mutfak sahnesinde dört tane dikdörtgen (kırmızı) aşağıdan-yukarı işleminden geçerek kabul edilmiştir ve sırası ile A, B ve C kutupsal olmayan düğümler tarafından temsil edilen üretim kuralları etkinleştirilmiştir. Böylece, yukarıdan-aşağı aramada adayların (mavi) sayısı tahmin edilir. Tam yukarı oklar aşağıdan-yukarı bağlamayı gösterirken, aşağı oklar ise yukarıdan-aşağı tahmini göstermektedir [36].

model tanımlanmaktadır. Böylelikle, bütün yapısal ve bağlamsal bilgi Ve-Veya çizgesinde (ve gramer de aynı şekilde) temsil edilmektedir. Bu nesne kategorizasyonu problemini de çözmektedir. Her nesne kategorisi, *bulunan grupta olasılığının örneklerin doğal frekansını oluşturması ile beraber gramer tarafından oluşturulan tüm geçerli yapılandırmalar dizisi olarak tanımlanabilir.*

Bu olasılık modeli, aralıklı kodlama (wavelet coding) ve stokastik bağlam dışı gramerler (SCFG) gibi popüler generatif modelleri ve Markov rastgele alanlar ve grafiksel modeller gibi tanımlayıcı modelleri birleştirir. İlki tekrar yapılandırma için generatif hiyerarşiyi temsil eder ve diğeri bağlamı modeller.

2.Hedef: Ölçeklenebilir ve özyinelemeli yukarıdan-aşağı/aşağıdan-yukarı hesaplama. Ve-Veya çizge gösterimi düğümlerinin iki tür özyinelemeli yapısı vardır ve düğüm sayısı ve nesne kategorilerine kadar ölçeklenebilmektedir. Örneğin, Veya-düğümü bir nesneyi, arabayı, temsil etsin, o zaman arabanın farklı görünümüleri (ön, yan,

arka tarafı vs.) için bir dizi çocuk düğümü olacaktır. Yeni bir çocuk düğüm eklenerek yeni görünüm meydana gelecektir.

Şekil 2.6 zaman adımımda, yapım aşamasında bir ayrıştırma çizgesini göstermektedir. Bu basit gramer bir tane primitif kullanmaktadır: dikdörtgen yüzeyler görüntü düzlemi üzerine yansıtılmıştır. Gramer kuralları, çeşitli örgü, lineer, yuvalama, kübik yapıların hizalanmaları gibi grupları temsil etmektedir. Mutfak sahnesinde dört tane dikdörtgen (kırmızı) aşağıdan-yukarı işleminden geçerek kabul edilmiştir ve sırası ile kutupsal olmayan A, B ve C düğümleri tarafından temsil edilen üretim kuralları etkinleştirilmiştir. Böylece, yukarıdan-aşağı aramada adayların (mavi) sayısı tahmin edilir. Tam yukarı oklar aşağıdan-yukarı bağlamayı gösterirken, aşağı oklar ise yukarıdan-aşağı tahmini göstermektedir.

Bir girdi görüntü verildiğinde, görüntü ayrıştırma işlemi çıktı tercümesi olarak anında en muhtemel ayrışma çizgesini oluşturur ve bu ayrışma çizgesi Veya-düğümlelerinde seçim yapıldıktan sonraki Ve-Veya çizgesinin alt çizgesidir.

3.Hedef: Küçük örnekleri öğrenme ve genelleme. Bu Ve-Veya çizge gösterimi üzerinde tanımlı olasılıksal modelin her kategori için nispeten daha küçük ayrıştırma dizisinden öğrenilebilir ve daha sonra çok sayıda yapılandırma oluşturmak için Monte Carlo simülasyonu üzerinde örneklendirilebilir. Bu aslında geleneksel doku sentezi deneyi için Minimax entropi ilkesi tarafından elde edilen bir uzantıdır. Burada, gözlenen dokudan farklı, ancak gözlenen dokuya algısal olarak eşdeğer olan yeni doku örnekleri oluşturulmaktadır. Minimax entropi ilkesi öğrenme şeması [17]'deki Ve-Veya çizgelerini genişletmektedir. Bu çizgeler deneyde öngörülemeyen nesne örneklerini kapsamak için bileşim aracılığı ile yeni yapılandırma üretebilir. Bu genelleme özelliği çoğunlukla ayrımcı makine öğrenme metodlarında eksik olmaktadır.

4.Hedef: Anlamsal boşluğu doldurmak için görsel sözlük eşleme. Semboller ve pikseller arasındaki iyi tanınmış anlamsal boşluğu doldurmak için gramer her düzeydeki görsel kavramlar için bir dizi görsel sözlük içermektedir. Bu sözlükler için yapılan iki önemli gözlem şunlardır.

1. Sözlüklerin elemanları çizge birleşimi aracılığı ile düzenlenmektedir. Sözlüğün en alt düzeyinde her birinin başka primitiflerle bağlanması için açık

bağlantısı ile küçük çizgedeki çapa noktaları sayısı olmak üzere bir dizi görüntü primitifi bulunmaktadır. Bu primitifler parçalar ve nesnelere için daha büyük çizge yapıları oluşturmak için birleştirilebilir, aynı çocukların Lego parçalarını topladığı gibi.

2. Görü, konuşmada nesnelere rastgele ölçeklerde görünebildiği gibi, diğer sensörlerden farklıdır. Buna göre, her düğümün örnekleri herhangi bir boyutta ortaya çıkabilir. Ve-Veya çizgesinde her düzeydeki kutupsal olmayan düğümler direk görüntü primitifleri olarak sınırlanabilirler. Böylece, aynı düğümde, ölçekler üzerindeki örnekler arasında geçiş söz konusu olabilir.

Literatürde düşük seviyedeki primitiflerin nasıl olması gerektiği konusunda çeşitli tartışmalar olmasına rağmen, farklılıklar gerçekte yok denecek kadar küçüktür. Bu lokal primitiflerden anlam çıkarma belirsizliğinin daha büyük yapılar kullanılarak yukarıdan-aşağı hesaplama üzerinden çözülmesi tavsiye edilir. Son olarak, primitifler ilkel bir eskiz çizge gösterimi oluşturmak için birleştirilmektedir ve girdi görüntüsü için bütün piksellerin açıklaması oluşturulacaktır. Bu, anlamsal boşluğu kapatmaktadır.

2.3 Fonksiyonel nesne tanımlamaları

Görü sistemleri için, nesnelere neye benzediğini tanımlama işlemi çok zor bir işittir. Nesnelere niçin gerekli olduğu ve ne işe yaradıklarını belirlemek daha kolaydır. Görü sistemleri, nesnelere tanımak için, gerekli olduğunda fonksiyonel tanımlamaları kullanabilir ve kendileri mümkün olduğunca fiziksel tanımlamaları öğrenebilmelidirler.

Mesela, birçok bardak türü vardır: bazılarında kulp olur, bazıları da kulpsuz olur; bazılarının düz silindirik gövdesi olur, bazıları da yivlidir; bazıları porselenden yapılmıştır, bazıları strafordan yapılmıştır ve bazıları da metaldir. Nesnelere sayısız fiziksel özellikleri vardır. Ancak, fonksiyonel olarak bütün bardaklar bir şey içmek için kullanılır. Fonksiyonel açıdan bardak nesnesinin ne olduğunu tanımlamak daha kolaydır.



Şekil 2.7: Kulplu bardak görüntüsü [33].

Daha açık olmak gerekirse, bardak için; açık kaplardır, stabil bir şekilde ayakta durabilir ve kaldırılabilir gibi tanımlamalar verilmiş olsun. Şekil 2.7’de görüldüğü gibi nesnenin kulpu, yukarıyı gösteren içbükey yüzeyi ve düz bir dibi vardır. Ayrıca, nesnenin hafif olduğu da bilinmektedir. Kâse, tuğla ve çanta nesnelerinin bulunduğu kabul edilirse, bardak nesnesinin arandığı sonucuna varılabilir. Ve bu verilen bardak örneğine dayanarak bir fiziksel bardak modeli oluşturulabilir.

Winston ve ark. ilk önce fonksiyonel açıklamalar kullanarak nesnenin fiziksel tanımını yapmıştır ve daha sonra, fonksiyonel açıklamalar ve tanıma ait özel eylemler kullanarak fiziksel modelleri çıkartmıştır [33]. Model çıkartma işlemi fiziksel örnek ve fonksiyonel açıklamaya ek olarak bazı geçmiş örnekleri (precedents) de kapsamaktadır:

- Fiziksel örneğin olması zorunludur. Aksi takdirde hangi geçmiş örneklerin uygun olduğu anlaşılabilir.
- Geçmiş örneklerin de olması zorunludur. Aksi takdirde fiziksel örneğin hangi görünüşünün uygun olduğu anlaşılabilir.

Kıyas yolu ile öğrenme ve kısıt transferi teorisini gerçekleştiren sistemin yaptığı görevler şunlardır:

- Bir problemi ve geçmiş örneği tanımlayan İngilizce kelimeleri anlamsal bir ağdan oluşan bağlantılara çeviren doğal bir dil arayüzü kullanmaktadır.
- Eşleme motoru, geçmiş örnek ve problem kısımları arasındaki tekabülleri tespit etmektedir.

```

Rule
  RULE-1
  if
    link found using CAUSE structure
    link found using CAUSE structure
    .
    .
    .
  then
    link to be shown to hold
  case
    names of all precedents used

```

Şekil 2.8: Kural oluşturma motoru tarafından belirlenen *if – then* kuralı [33].

- Analoglaştırma motoru problemdeki sorgulanan bağlantının problemdeki verilen bağlantılar tarafından desteklenip desteklenmediğini belirlemektedir. Bunu gerçekleştirmek için, analoglaştırma motoru geçmiş örnekten alınan *CAUSE* (neden, sebep) bağlantılarını probleme transfer etmektedir.
- Kural oluşturma motoru problemde ortaya çıkartılan geçmiş örnekteki nedensel yapı bölümünü yakalamak için *if – then* kuralı düzenlemektedir. *If – then* kuralındaki *then* kısmı problemdeki sorgulanan bağlantı tarafından gelmektedir. *If* kısımları ise transfer edilen *CAUSE* (neden, sebep) yapısı tarafından belirlenen bağlantılar tarafından gelmektedir. Bu kural Şekil 2.8’de görülmektedir.

Daha karmaşık durumlarda, sadece bir tane geçmiş örnekten yeterli nedensel yapı oluşturulamamaktadır. Bu nedenle, bir kaç tane geçmiş örneğin birleştirilmesi gerekmektedir. Transfer edilen *CAUSE* (neden, sebep) yapısında problemdeki yeni oluşturulan bağlantıya çıkan yolun bulunamadığı durumda yeni bir geçmiş örnek tespit edilmektedir. Bu çalışmada kullanılan bütün örnekler birden fazla geçmiş örnek kullanmaktadır.

ACRONYM tanıma odaklı modelleme sistemidir ve içeriği aşağıda verilmiştir:

- ACRONYM nesnelerin nasıl alan doldurduğunu genelleştirilmiş silindirleri kullanarak tanımlar.
- ACRONYM bir izleyicinin ne gördüğünü temsil etmek için dizginleri ve

elipsleri kullanır.

- ACRONYM belirli görünüm noktalarından nesnelere neye benzediğini tahmin eder.
- ACRONYM aşağıdan yukarı analizini denetlemek için yukarıdan aşağı tahminini sağlar.

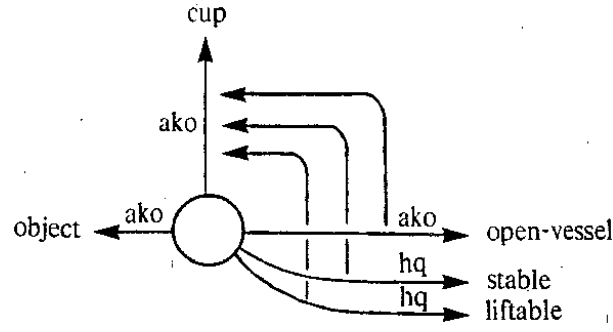
Bu bölümde ANALOGY'nin öğrenme işlemi ve ACRONYM'in fiziksel temsili işleminin sentezini kapsayan adımlar kısaca anlatılmıştır. Sonraki bölümde ise bu adımlar bir bardağın fiziksel modelinin tanımlanması ve öğrenme sistemi ile tanınması belirli bir senaryo üzerinde örneklendirilerek açıklanmıştır. Sistem, burada, kullanılan bardak türü için ACRONYM temsil ilkelerini ifade etmektedir ve daha sonra *if – then* şeklinde bir model oluşturmaktadır. Sentezdeki adımlar şu şekildedir:

1. Tanınacak olan nesneyi fonksiyonel açıdan açıkla.
2. Fiziksel bir örnek göster.
3. Fiziksel örneğin fiziksel tanımını artır.
4. Fonksiyonel gerekliliklerin gelişmiş fiziksel tanım ile sağlandığını göster, ve böylece nesneyi belirle.
5. Fiziksel tanımlanan kavramın fiziksel modelini oluştur.

Aşağıda sıradan bir bardağın neye benzediğini öğrenme işlemi gösterilerek öğrenme sürecinin adımları anlatılmıştır.

İlk adım, bardak kavramını kaldırılabilirlik, sabitlik, ve erişilebilir bir konteyner olarak kullanılabilirlik gibi fonksiyonel nitelikler cinsinden tarif etmektir. Bu işlem İngilizce kullanılarak aşağıdaki gibi yapılmaktadır:

X bir tanım olsun. X bir nesnenin tanımıdır. Nesne sabit, kaldırılabilir ve açık bir kap olduğu için bardaktır. X'i hatırla.



Şekil 2.9: Bardak nesnesinin fonksiyonel tanımı. Bu anlamsal ağ İngilizce tanımlar kullanarak oluşturulmuştur. AKO = A Kind Of, HQ = Has Quality [33].

Elbette diğer daha ayrıntılı tanımlar kullanılabilir; ama bu çalışmadaki tanımlar yeterli olarak sayılmıştır.

İngilizce tanımlar Şekil 2.9'daki gibi anlamsal bir ağa dönüştürülmektedir.

Sonraki adım ise Şekil 2.10'da görüldüğü gibi bir bardak örneği göstermektir. ACRONYM, böylesine görsel bir bilgiyi Şekil 2.9'daki anlamsal ağa dönüştürme kapasitesine sahiptir. Bu çalışma, ACRONYM'den tam olarak yararlanmadığı için aşağıdaki gibi İngilizce bir tanım kullanılmıştır:

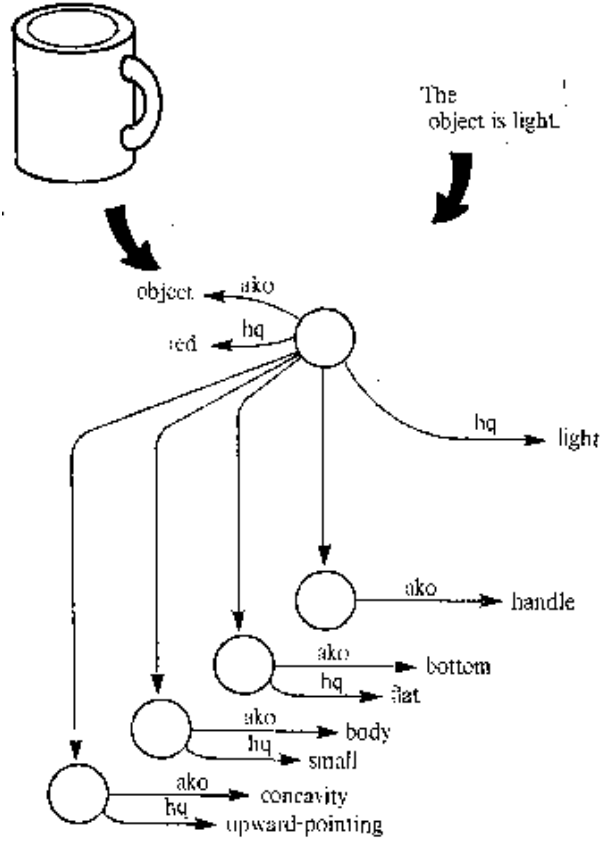
E bir alıştırma olsun. E, kırmızı bir nesne ile ilgili bir alıştırmadır. Nesnenin hacmi küçüktür. Nesnenin altı düzdür. Nesnenin yukarı doğru bir çukurluğu bulunmaktadır. Nesnenin kulpu vardır.

Burada dikkat edilmesi gereken nokta, belli bir bardak için yapılan tanımdaki özellikler fonksiyonel değil, fizikseldir.

Sonraki adımda, fiziksel örneğin fiziksel tanımı arttırılmaktadır. Bu, görüşten elde edilemeyen fiziksel özellikleri ve bağlantıları belirtmek için yardımcı olmaktadır.

Nesne hafiftir.

Görüldüğü gibi bu çalışmada fonksiyonel nesne tanımlamaları kavramı kullanılarak şekil özeti oluşturulmuş ve fiziksel nesne tanımlamalarını öğrenmek için şekil benzetmeleri kullanılmıştır. Basit şekil özetlerini oluşturmak için



Şekil 2.10: Fiziksel görünümün anlamsal ağını gösteren belli bir bardak örneği. Anlamsal ağı çoğu görüntüden ACRONYM kullanılarak çıkartılabilir. Bazı görünmeyen detaylar doğal dil arayüzü kullanılarak eklenebilir [33].

mekanizmalar mevcut olmadığı için, fonksiyonel nesne tanımları tabanlı teknikler kendi etkinliğini kaybetmiştir.

2.4 Çizgeler

Çizgeler, nesne ve kavramların gösterimi için genel olarak kullanılan güçlü bir veri yapısıdır. Çizge gösteriminde, genellikle, düğümler nesnelere ya da nesne parçalarını temsil ederken, kenarlar ise nesnelere veya nesne parçaları arasındaki ilişkileri tanımlamaktadır. Çizgeler bazı ilginç değişmezlik özelliklerine sahiptir. Örneğin, bir kağıt üzerine çizilen çizge çevirme, döndürme ya da dönüştürme işlemine tabi tutularak yansıması elde edilirse, yansımadaki çizge matematiksel anlamda asıl çizgenin aynısıdır. Bu değişmezlik özelliklerinin parçaları ve onların

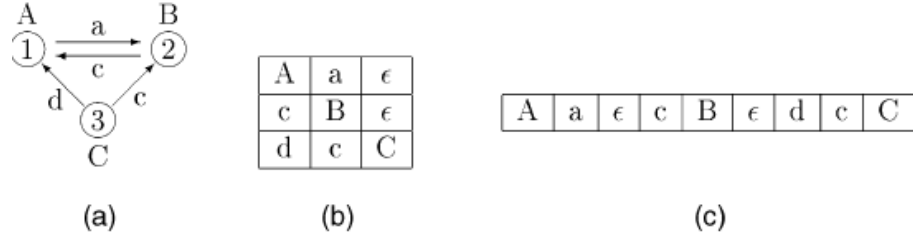
ilişkileri bakımından nesnelere modelleme için çok uygun olması, çizgeleri örüntü tanıma ve bilgisayarla görü uygulamaları için çok cazip bir hale getirmektedir.

Nesne prototipi öğrenmede, aynı nesnenin gürültülü örnek dizisi verilir ve temsili bir model çıkarma amaçlanır. Bu bağlamda, medyan kavramının çok yararlı olduğu ortaya çıkmaktadır. Eğer üzerinde işlem gerçekleştirilen nesnelere temsili için çizgeler kullanılırsa, bir çizge dizisinin medyanını bulma işi ile karşı karşıya kalınmaktadır. Bir çizge dizisi verilirse, medyan, dizideki bütün çizgelere olan mesafelerinin toplamı en küçük olan çizgedir. Burada, dizi medyan ve genelleştirilmiş medyan çizgeleri arasındaki fark anlaşılabilir. Fark, ayrı ayrı aranan medyanın olduğu çizgelerin uzayındadır.

Jiang ve ark. medyan kavramını çizgeler alanına genişletmiştir [15]. Çizge mesafesi bakımından, bir çizge dizisinin dizi medyanı ve genelleştirilmiş medyanı yeni kavram olarak tanımlanmıştır. Her iki türdeki medyan çizgelerinin özelliklerine bakılmıştır. Daha kompleks genelleştirilmiş medyan çizgelerin hesaplanması işlemi için genetik arama algoritması geliştirilmiştir.

Genelleştirilmiş medyan çizgenin hesaplanması son derece karmaşık bir işittir. Pratik nedenlerden ötürü, makul bir süre içinde, iyi kalitede yaklaşık çözümler sunan bir arama stratejisine ihtiyaç vardır. Burada bahsi geçen iş, bilgisayarla görüde çok öneme sahip ayrık optimizasyon problemlerinin geniş bir sınıfına aittir.

Genetik arama teknikleri biyolojik evrim teorisi kavramlarından esinlenmiştir. Onlar, arama, optimizasyon, makine öğrenme gibi zor işlemler için başarıyla uygulanan genel amaçlı optimizasyon metotlarıdır. Özellikle, genetik algoritmaların genel NP-complete problemlere uygulanan yaklaşık çözümler için yararlı olduğu kanıtlanmıştır [16]. Genetik algoritmaların çizge eşleme problemlerini çözdüğü durumlara rastlanmıştır [9], [32]. Genetik arama, eldeki ayrık optimizasyon problemi için olası çözümleri temsil eden kromozom nüfusunu koruma fikridir. Uygunluk olarak adlandırılan maliyet fonksiyonu çözüm adayları ile ilişkilidir. Başlangıç bir nüfus verilmiş olsun. Genetik algoritmalar nüfustaki kromozomları değiştirmek ve yeni bir nüfus oluşturmak için genetik operatörler kullanır. Çaprazlama genetik operatörü, çözüm adaylarını geçerli nüfustan rastgele çözümler ve seçilmiş yapılandırma konumlarında çözümleri değiştirme işlemi kapsamaktadır. Mutasyon işlemi bireysel çözüm adaylarının bileşen sembollerini rastgele



Şekil 2.11: Çizgenin komşuluk matrisinin kromozom gösterimi: (a) asıl çizge, (b) komşuluk matrisi, ve (c) kromozom [15].

değiştirerek nüfusa yeni bilgileri tanıtmaktır. Bir durdurma koşulu sağlanana kadar nüfus oluşturma işlemi devamlı tekrarlanır.

[9] çalışmasında belirtildiği gibi genetik algoritmaların diğer ayrık optimizasyon metotlarına göre bir takım avantajları vardır. Genetik arama algoritmasının eşsiz özelliklerinden biri çaprazlama operatörünün olmasıdır. Bu operatör etkili bir şekilde yerel tutarlı alt çözümleri birleştirerek küresel tutarlı bir çözüm üretmektedir. Ek olarak, genetik operatörler küresel yakınsama için daha yüksek bir olasılık sağlamaktadır.

Bir g çizgesini kromozom açısından temsil etmek için basit bir çözüm, kendi matrisini doğrusallaştırarak elde etmektir, örneğin; kendi satırlarını birleştirmek. Şekil 2.11'de örnek verilmiştir. g aday medyan çizgesinin kromozom temsilinin uygunluk fonksiyonu, $SOD(g)$ olarak seçilmiştir. Bu kromozom temsili, kavramsal olarak basittir; ancak ciddi bir eksikliği vardır. Her iterasyonda ve nüfustaki her bireysel kromozom için genelleştirilmiş medyan çizgesindeki g adayını temsil etmek için S kümesindeki her g_i 'nin $d(g, g_i)$ değerinin hesaplanması gerekmektedir. Bu hesaplama g 'nin büyüklüğünde üsteldir. Dolayısıyla, çok büyük hesaplama karmaşıklığı olan bir prosedür ile karşı karşıya kalınmaktadır.

Şekil 2.11'deki basit kromozom temsilinden ortaya çıkan büyük hesaplama karmaşıklığını gidermek için başka bir çözüm geliştirilmiştir. Sadece g aday çizgesini kromozomda kodlamak ve ayrı ayrı g ile g_i ler arasında optimal eşleme bulmak yerine, g çizgesi ve g_i ler için bütün eşlemeler direk kromozomda kodlanmaktadır. Böylece, genetik algoritma tarafından sadece genelleştirilmiş medyan çizgesi için potansiyel g adayını oluşturmak değil, S kümesinde g ve g_i ler arasında optimal eşlemeleri hesaplama işlemi de gerçekleştirilmektedir.

Bu çalışmadaki arama stratejisi genetik algoritmalarda genellikle kullanılan standart operatörlere dayanmaktadır. Buradaki algoritmada, uygunluk orantılı seçim uygulayıcı, rulet tekerleği örnekleme yavrularını oluşturmak için seçilir. Bu kavramsal olarak, kromozunun uygunluğuna alan olarak eşit her bir kromozom için bir dilim dairesel rulet tekerleği belirlemek ile aynıdır. Rulet tekerleği döndürülmektedir ve top kama şeklindeki bir dilim kalanına geldiğinde, topa karşılık gelen kromozom seçilmektedir.

Çaprazlama olasılığı ile iki tane yavruyu oluşturmak için seçilmiş bir çift kromozom keyfi bir konumda (üniform olasılık ile seçilen) çaprazlanır. Genetik algoritmalarda genellikle kullanılan bu tek nokta çaprazlama operatörü, bu durumda geliştirilmelidir. Ortaya çıkan kromozomlar tarafından temsil edilen dönüşümler tutarsız olabilir. Bu durumda, kromozomların değiştirilmesi gerekmektedir. Genel olarak konuşulması gerekirse, aynı g 'deki iki veya daha fazla düğüm aday g 'deki aynı düğüm ile eşlendiği zaman tutarsız durum ortaya çıkar. g_1 'in hem birinci, hem de ikinci düğümü g 'nin ikinci düğümü ile eşleşmiştir. Değişiklik tutarsız kromozomun ilgili konumları tutarlı hale gelene kadar rastgele değiştirilerek gerçekleştirilir.

Mutasyon işlemi dizideki her sayı mutasyon olasılığı ile rasgele değiştirilerek gerçekleştirilir. Yine, tutarlılık testi için ve gerekli olduğu durumlarda, çaprazlama operatörüne benzer bir değiştirme işlemi uygulanmaktadır.

Bu çalışma verilen g_i çizgeleri bilinmeyen genelleştirilmiş medyan çizgenin iyi bir yaklaşımıdır varsayımını takip etmektedir. Bu yüzden, onlar başlangıç nüfusun parçası olarak kullanılmıştır. Başlangıç nüfusuna geri kalanları ise rastgele oluşturulmuş kromozomlardır. g_i girdi çizgesini başlangıç nüfusa girmek bütün girdiler için g_i 'nin ECGM'sini hesaplamayı ifade eder. Bu nedenle, başlangıç durumu verildikten sonra, medyan çizge dizisi dolaylı olarak hesaplanmaktadır. Onun mevcut olması genetik arama tarafından bulunan yaklaşık genelleştirilmiş medyan çizgesinin kalitesini değerlendirmeye yardımcı olur.

Aşağıdaki iki kriterden biri yerine getirilinceye kadar nüfus evrim süreci devam ettirilir. İlk kriter, nesillerin maksimum sayısına ulaşılmış olmasıdır. İkincisi ise, eğer nüfus çok homojen bir hale gelirse, işlem de sona erer. Mevcut uygulamada, homojenlik, bir nüfusun ortalama araçları ve uygunluk değerlerinin varyansı ile

ölçülmektedir. Onların çarpımı, belirlenmiş bir eşik değerinden küçük ise, o zaman evrimi durdurmak için nüfusun yeteri kadar homojen olduğu varsayılabilir.

Bilgisayarla görü ve örüntü tanımada sadece geleneksel olarak vektör uzayı için uygulanan bir çok teknik vardır. Onların çizge alanına genişletilmesi gelecek araştırmalar için zorlu bir görevdir. Burada tanıtılan medyan çizge kavramı gelecekteki gelişmeler için yararlı bir araç olabilir.

3. ÖZET ŞEKİL ÇIKARMA İŞLEMİ

Bu bölümde ilk önce genel şekil tanıma problemi tanımlanmıştır ve şeklin iskeletsel çizgesini çıkarma işlemi anlatılmıştır. Daha sonra, örnek şekiller dizisi üzerinden şekil-tabanlı kategorik modelini otomatik öğrenen, Demirci ve ark.'nın önerdiği çalışma [11], "Örneklerden iskeletsel özet oluşturma" başlığı altında anlatılmıştır. Daha sonra [11]'de kullanılan EMD algoritmasına yer verilmiştir. Son olarak, bir önceki çalışmanın eksiklikleri belirtilerek, sunulan çalışmaya giriş yapılmıştır.

3.1 Genel şekil tanıma problemi

Bir şekil için temsil işlemi, belirlenen kurallarıyla beraber, şekil ya da şeklin bazı yönlerinin herhangi bir özel şekil için şemanın nasıl uygulanacağını tanımlayan formal bir şemadır [22]. Bir şeklin tanımı o şeklin üzerinde kullanılan gösterim şemasıdır [22]. Örneğin, şeklin anları öznitelik olarak kullanılırsa, anları hesaplama formülü gösterimin bir parçası iken, formül kullanılarak şekilden elde edilen sayısal değer onun tanımıdır. Şekil eşleme iki şekil arasındaki tekabülü bulma problemidir. Şekil tanıma ise bilinmeyen bir şeklin hangi kategoriye ait olduğunu bulma problemidir. Genel şekil tanıma, uygulama alanındaki şeklin değişkenliği ve doğasında herhangi bir kısıtlama olmadığını ima etmektedir.

Şekil bilgisinin, tanıma sürecinde çok önemli bir rol oynadığı bilinmektedir. Şekil 3.1'de görüldüğü gibi insanlar bu nesnelere şekillerin silüetleri ve ilkel

çiziminden hangi kategoride olduğunu kolaylıkla çıkarabilmektedirler. Bu görüntülerde renk, gölgelendirme, hareket veya doku bilgisi bulunmadığından tanıma problemi sadece şekil bilgisi kullanılarak çözülebilir.



Şekil 3.1: Kolay tanınan bazı şekil silüetleri.

Genel şekil gösterim şemaları için bazı genel gereksinimler bulunmaktadır. Birinci olarak, şekil etki alanında serbest ve öngörülemeyen bir değişkenlik söz konusudur. Genel şekil tanıma için, gösterimin, şekil etki alanındaki bu büyük derecedeki değişkenliği yakalama özelliğine sahip olmalıdır. Kaydırma, artırıp düşürme, döndürme, bakış yönü, eklemleme vb. gibi değişik görsel dönüşümler altında benzer tanımlamalar sağlamalıdır. İkincisi, algısal olarak benzer olan nesnelere gösterimde benzer tanımlamalara sahip olmalılar. Bu, istikrar (denge) gerekliliği olarak bilinmektedir. Üçüncüsü, tanımlamalar benzersiz olmalıdır. Aksi takdirde, farklı şekiller benzer tanımlamalara sahip olabilmektedir.

Mevcut şekil temsili şemaları çok farklı kriterlere göre sınıflandırılabilir. Temsil şemalarının çoğu istikrar gereksinimini sağlamamaktadırlar ve görsel dönüşümler altında güçlü değildir. Şekil 3.2'deki şekiller için gösterim işini göz önünde bulunduralım. İnsanlar bu şekillerin aynı varlığı temsil ettiğini algırlar, bu nedenle, temsil şemasının onların benzerliğini yakalaması beklenmektedir.



Şekil 3.2: Aynı varlığı temsil eden iki şekil.

3.2 Şeklin iskeletsel çizgesini çıkarma

Bir düzlem bölgenin yapısal şeklini temsil etmek için kullanılan önemli yaklaşımlardan biri, bölgeyi çizgeye indirgemektir. Bu indirgeme işlemi, inceltme (*skeletonizing* olarak da adlandırılmaktadır) algoritması kullanarak bölgenin iskeletini elde etmektir. İnceltme işlemi görüntü işlemede çok önemli rollerden birini oynamaktadır.

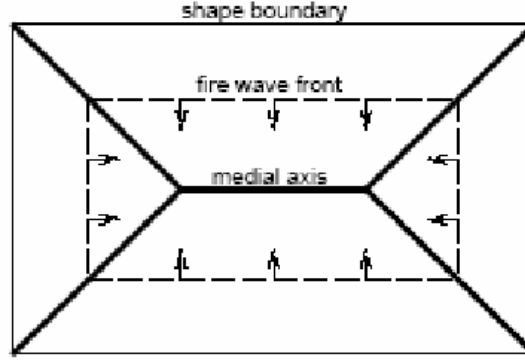
Bir bölgenin iskeleti Blum [6] tarafından önerilen orta eksen dönüşümü (MAT - medial axis transform) yaklaşımı tarafından tanımlanabilir. Sınırı B olarak belirlenen bir R bölgesinin orta eksen dönüşümü şu şekildedir: R 'deki her p noktası için, onun B 'deki en yakın komşusu belirlenir. Eğer, p 'nin birden fazla böyle komşusu var ise, p noktası R 'nin orta eksenine (iskeletine) aittir denilmektedir. "En yakın" kavramı uzaklığın tanımına bağlıdır. Şekil 3.3'te Euclid uzaklığı kullanılan örnekler görülmektedir.



Şekil 3.3: Balık ve at şekli ve onların iskeletsel noktaları (şekillerin iç kısımları).

Bir bölgenin orta eksen dönüşümü (MAT) "kır yangın kavramı" olarak bilinen sezgisel bir tanıma dayanmaktadır. Bir görüntü bölgesini üniform, kuru ot kır olarak tanımlayalım ve kenarlarından ateşe tutulduğunu varsayalım. Tüm yangın cepheleri aynı hızda bölgeye doğru ilerleyecektir. Aynı anda birden fazla yangının ulaştığı noktalar kümesi bize orta eksen dönüşümünü (MAT) vermektedir. Bir dikdörtgen şekli için Şekil 3.4'te örnek verilmiştir.

Eksen tabanlı temsilleri kullanmanın ana avantajı geometrik özelliklerinden ötürü onların şekillerin algısal özelliklerini yakalama özelliğidir. Şekil 3.3'te at şeklinin her bir parçası orta eksen dönüşümü tanımlamasında açık hale getirilmiştir. Orta eksen bölümlerinin organizasyonu benzer olduğu için tanımlamalar da benzerdir. Diğer artı tarafı da eksen tabanlı temsillerin hem bölge hem de sınır bilgisini



Şekil 3.4: Bir dikdörtgen üzerinde ateşin yayılması örneği [4].

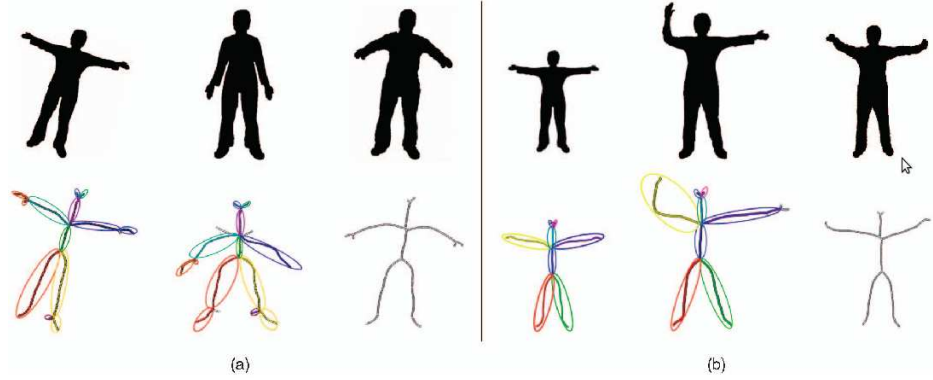
yakalamasıdır.

3.3 Örneklerden iskeletsel özet oluşturma

Nesne sınıflandırma işlemi sınıf içi görünüm ve şekil değişikliklerinden bağımsız olan prototip modellerine ihtiyaç duymaktadır. Prototip şekil modelleme, etki alanında bu nesnenin eklemleme olarak değişmeyen, genel parça yapısını yakalayan modele dönüşmektedir. Belirli bir sınıfa ait bir örnek dizisi tek bir prototiple değiştirilebilir (ya da hiyerarşik olarak bir prototip altında toplanabilir). O zaman, nesne tanıma işleminin karmaşıklığı kabadan-ince arama/indeksleme tekniklerine indirgenebilir.

Önceki birçok şekil sınıflandırma tekniği, örnekleri bire bir tekabülde aynı öznitelikleri paylaşan kategorik modellerle sınırı kalmaktadır. Bu özniteliklerin ilgili konumları, yüzler, motosikletler veya arabalar gibi sınırlı olan kategorilerde, biraz değişebilmesine rağmen, bu modeller parça eklemleme, görüntü dönüştürme, veya ölçek değiştirme işlemlerinden bağımsız değildir.

Çizgeler şeklin kullanışlı yapısal özetini temsil etmektedir. Bire bir öznitelik tekabülü örnekler dizisi kategorisi genişliğinde dağıtılsa, bu ilgili çizgeler genişliğinde çoklu düğüm tekabülüne dönüşmektedir. Eğer çoklu tekabüller bulunabilirse, onlar kullanılarak kategorik model için gerekli olan özellikleri karşılayan prototip çizgenin özet öznitelikleri (düğümler) ve ilişkileri (kenarlar) oluşturulabilir.

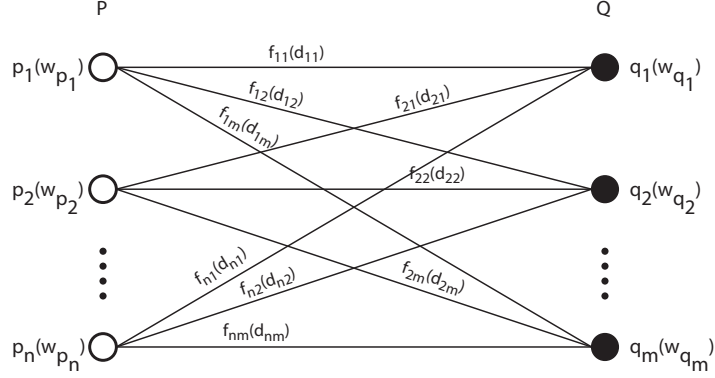


Şekil 3.5: İki tane örnekten şekil prototipini çıkartma işlemi. İlk iki sütun değişik görüntü rotasyonuna ve parça eklememesine maruz kalmış iki tane örnek bölgeyi göstermektedir. Orta (medyan) dalları çizgedeki düğümleri oluşturmak için bölümlenmiştir (elips şeklinde) ve eşleme sonucu elde edilen düğüm tekabülleri aynı renk ile gösterilmiştir. Birinci sütundaki insan şeklinin tek parça sağ bacağı (şekil açısından) ikinci sütundaki insan şeklinin iki parça sağ bacağına eşleştiği gibi düğümler arasındaki tekabüller çoklu olabilir. Aynı şekilde, birinci sütundaki iki parça sol kol kısmı ikinci sütundaki tek parça sol kol kısmı ile eşleşmiştir. Üçüncü ve altıncı sütunlarda ikili örneklerden elde edilen özet şekil prototipleri görülmektedir [11].

Demirci ve ark. örnek şekiller dizisi üzerinden şekil-tabanlı kategorik modeli otomatik öğrenen yeni bir çerçeve önermiştir [11]. Her örneğin çizge temsili varsayılarak, algoritma örnek çizgeler çifti arasında çoklu tekabülleri hesaplayarak işleme başlamaktadır. Bu tekabüllerden, düğümleri ilgili özniteliğin özetlerini ve kenarları özetler arasındaki bağı gösteren bir prototip çizge oluşturulmaktadır. Bu ikili özetleme prosedürü, şekiller kümesini sınıflara ayıran ve özet sınıf prototiplerini hesaplayan hiyerarşik kümeleme sürecinin ana işlemidir.

Söz konusu çalışmada ilk önce şekiller, orta eksen çizgesi (MAT ile ilgili detaylı bilgi için Bölü 3.2'e bakınız) açısından tanımlanmaktadır ve daha sonra oluşturulan iki tane çizgenin düğümleri arasındaki eşlemeyi bulma problemine yönelmektedir. Daha önceki çizge eşleme çalışmaları [27], [25], [14], [28], iki çizge arasındaki bire bir eşlemeyi bulmaya odaklanmıştır. Ancak, Şekil 3.5'te görüldüğü gibi tekabüller genellikle bire bir değil, aksine, çokludur. Kısım tekabüllerinin bire bir olması gerektiği zaman bile, gürültü ve parçalara ayırma hatalarından dolayı onların öznitelikleri arasında çoklu tekabüller oluşabilmektedir. Bu durum, orta eksen çizgeleri için, bir çizgedeki tek düğüm diğer çizgedeki fazladan parçalanmış düğüm grubuna denk gelebilir anlamına gelmektedir.

Orta eksen çizgeleri arasındaki çoklu parça tekabüllerini bulmak için kullanılan



Şekil 3.6: P ve Q dağılımlarının verildiği bir taşıma problemi örneği. f_{ij} , p_i ve q_j noktaları arasındaki akışı ve d_{ij} uzaklığı göstermektedir. w_{p_i} (veya w_{q_i}), p_i (veya q_i)'nin ağırlığıdır.

algoritma, Demirci ve ark.'nın çoklu eşleme [12] çalışmasının genişletilmiş halidir. Söz konusu yaklaşımda, eşleşecek olan iki tane çizgenin düğümleri sabit boyutlu Euclid uzayında temsil edilir. Daha sonra temsil edilen düğümler (noktalar) arasında çoklu eşleme bulunur. Prosedürün ilk adımı [1] ve [23]'teki düşük-bozuklu gömme (low-distortion embedding) tekniği kullanılarak gerçekleştirilmektedir. İkinci adımda ise EMD (Earth Mover's Distance) algoritması dönüşüm altında kullanılmaktadır.

EMD algoritması lineer programlama probleminin özel bir türü olan taşıma probleminin [2] çözümüne dayanmaktadır. EMD her bir dağılımın her bir noktası d -boyutlu uzayda verilen koordinatları ve ağırlığı ile temsil edilen iki dağılım arasındaki uzaklığı (benzerliği) verir. Birinci dağılımdaki her elemanın kaynağı ve ikinci dağılımdaki her elemanın onların konumlarındaki hedefi gösterdiği varsayılın. EMD algoritması bir dağılımı diğerine taşımak için gerekli olan minimum iş miktarını bulur. Bu problem aşağıdaki lineer programlama problemi olarak formalize edilebilir. $P = (p_1, w_{p_1}), \dots, (p_n, w_{p_n})$ birinci dağılım, $Q = (q_1, w_{q_1}), \dots, (q_m, w_{q_m})$ ikinci dağılım olsun. Burada, p_i (veya q_i), i 'nci elemanın pozisyonu, w_{p_i} (veya w_{q_i}), i 'nci elemanın ağırlığı ve d_{ij} , p_i ve q_j arasındaki uzaklıktır. f_{ij} ($1 \leq i \leq n, 1 \leq j \leq m$), p_i noktasından q_j noktasına taşınan akış miktarı olsun. Şekil 3.6 bu formalizasyonu göstermektedir.

Bu problemin amacı toplam maliyeti en aza indiren akış değerlerini bulmaktır:

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

öyle ki:

$$f_{ij} \geq 0, 1 \leq i \leq n, 1 \leq j \leq m \quad (3.1)$$

$$\sum_{i=1}^n f_{ij} \leq w_{pi} \quad (3.2)$$

$$\sum_{j=1}^m f_{ij} \leq w_{qj} \quad (3.3)$$

$$\sum_{i=1}^n \sum_{j=1}^m f_{ij} = \min \left(\sum_{i=1}^n w_{pi}, \sum_{j=1}^m w_{qj} \right) \quad (3.4)$$

Bu kısıtlamalar P 'den Q 'ye giden akışların sıfırdan büyük ve iki yönlü olmasını, P dağılımındaki her kaynak noktası tarafından gönderilen toplam akış miktarının ve Q dağılımındaki her hedef noktasına aktarılan toplam akış miktarının ağırlıkları ile sınırlı olmasını, ve P dağılımından Q dağılımına gönderilen akışın en yüksek miktarda olmasını garanti altına almaktadır. Bir taşıma problemi parametre tablosu açısından değerlendirilebilir. Şekil 3.6'daki formülasyon kullanılarak oluşturulan parametre tablosu Çizelge 3.1'de görülmektedir. Bu problemin çözümü sonraki bölümde anlatılmıştır.

[12] çalışmasında, orta eksen çizgelerinin, çoklu parça eşleme için uyarılmasının sebebi birkaç önemli gözleme dayanmaktadır. İlk olarak, bölgeler, onların "parçaları" düzeyinde eşlemeye çalışılmaktadır. Bir parçanın, orta eksen çizgesinin bağlı alt çizgesini temsil ettiği için, iki tane orta eksen çizgesi arasındaki çoklu eşlemeye bağlı alt çizgeyi diğer bir bağlı alt çizgeye denk getirmesi gerekmektedir. İkincisi, iki bölgenin özeti, bölgeler arasındaki birçok ortak "parça"dan oluşmalıdır. Sonuçta, bağlı alt çizgeleri bağlayan maksimum sayıda eşlemeye ihtiyaç duyulmaktadır. Son olarak, eşleşen "parçalar"ın birbirine oldukça benzer olması gerekiyor. Bu yüzden, orta eksen çizgelerinin bağlı alt çizgeleri arasında maksimum, kısmi çoklu eşleme oluşturulması söz konusudur. Öyle ki, eşleşen alt çizgelerin düğümlerine denk gelen orta eksen kısımlarının toplamı birbirine oldukça benzer olmalı. Sonuç olarak, ortaya çıkan çoklu

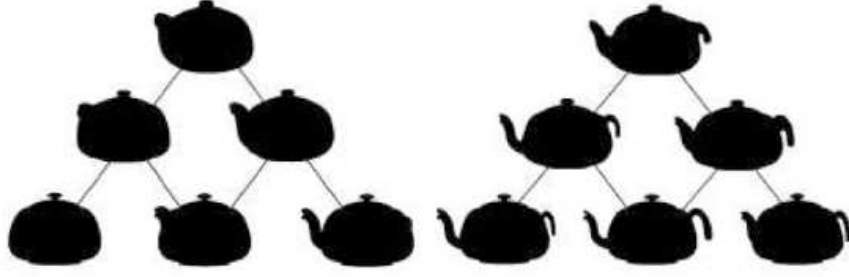
	q_1 'e		q_2 'ye		q_3 'e		...	q_m 'e		Hedef
p_1 'den	d_{11}	δ_{11}	d_{12}	δ_{12}	d_{13}	δ_{13}	...	d_{1m}	δ_{1m}	w_{p_1}
	Δ_{11}	f_{11}	Δ_{12}	f_{12}	Δ_{13}	f_{13}	...	Δ_{1m}	f_{1m}	
p_2 'den	d_{21}	δ_{21}	d_{22}	δ_{22}	d_{23}	δ_{23}	...	d_{2m}	δ_{2m}	w_{p_2}
	Δ_{21}	f_{21}	Δ_{22}	f_{22}	Δ_{23}	f_{23}	...	Δ_{2m}	f_{2m}	
p_3 'ten	d_{31}	δ_{31}	d_{32}	δ_{32}	d_{33}	δ_{33}	...	d_{3m}	δ_{3m}	w_{p_3}
	Δ_{31}	f_{31}	Δ_{32}	f_{32}	Δ_{33}	f_{33}	...	Δ_{3m}	f_{3m}	
...
p_n 'den	d_{n1}	δ_{n1}	d_{n2}	δ_{n2}	d_{n3}	δ_{n3}	...	d_{nm}	δ_{nm}	w_{p_n}
	Δ_{n1}	f_{n1}	Δ_{n2}	f_{n2}	Δ_{n3}	f_{n3}	...	Δ_{nm}	f_{nm}	
Kaynak	w_{q_1}		w_{q_2}		w_{q_3}			w_{q_m}		

Çizelge 3.1: Şekil 3.6'de verilen taşıma problemi için oluşturulan parametre tablosu. f_{ij} , p_i ve q_j noktaları arasındaki akışı ve d_{ij} de uzaklığı göstermektedir. Her bir hedef noktasının ağırlığı tablonun aşağısında ve her bir kaynak noktasının ağırlığı tablonun sağ tarafında verilmiştir. δ_{ij} ve Δ_{ij} değişkenleri optimal çözüm hesaplanırken kullanılmıştır. Detaylar için Algoritma 1 ve Algoritma 2'ye bakınız.

eşleme çerçevesi, dönüştürme, ölçekleme, görüntü döndürme, eklememe, ve sınıf içi deformasyon gibi küçük değişimlere karşı dayanıklı olarak tasarlanmıştır (Şekil 3.5). Ayrıca, EMD algoritmasına ek akış kısıtlamaları dahil edilerek iki orta eksen çizgesi arasında kısmi eşleme bulunabilir.

Bir nesne sınıfı ile ilişkili bir görünüm dizisinin temsili özetlerini oluşturmak için daha önce bahsedilen ikili özet oluşturma modeli kullanılmıştır. Çoklu eşleme algoritması iki görünüm arasındaki benzerliği sayısal ölçüm olarak sağladığından, birbirine çok benzeyen ikililer dizisi defalarca kullanılarak özet görünümler oluşturulabilir. Ortaya çıkan ve özet hiyerarşisi olarak adlandırılan hiyerarşik yapı, bir nesne sınıfı dizisi için, bu dizideki nesnelerin birbirine oldukça benzer temsili prototiplerini yakalamaktadır. Şekil 3.7 altı tane çaydanlık görünümü için oluşturulan iki şekil prototipini göstermektedir.

Başarılı olmasına rağmen, [12] çalışmasının bazı kısıtlamaları vardır. İlk olarak, kaydırma, artırıp düşürme, döndürme, bakış yönü, eklememeden bağımsız olarak şekil prototipleri oluşturması ve çoklu parça eşleme tekabülleri bulmasına rağmen, silüetlerin doğru bölümlenmiş ya da ayrılmış şekilde sunulmasına ihtiyaç duyulmaktadır. İkincisi ise, özetlenmiş orta eksen çizgeleri nesne sınıflandırmasını destekleyecek şekilde tasarlanmıştır, ancak, görselleştirme ve



Şekil 3.7: Bir dizi çaydanlık görüntüsü için şekil prototiplerini oluşturma örneği. Asıl görüntüler en altta (alt düzey) gösterilirken, onların ortalamaları iç düğümleri belirtmektedir. İkili ortalama işlemi, prototipin soyunun herhangi birinden yeterince farklı oluncaya kadar devam ettirilir. Buradaki durumda, altı tane görünümünden iki tane prototip oluşturulmuştur [11].

sıkıştırma işlemini desteklememektedir. Bu yüzden, yeniden oluşturma hatasını optimize etmek için hiçbir girişimde bulunulmamıştır. Son olarak, bu çalışmada sunulan metot güçlü olmasında rağmen, etkinliği elde edilen nokta tekabüllerine bağlıdır. Elde edilen tekabüller EMD [26] algoritması tarafından bulunan tek optimal çözüme dayanmaktadır. Yüksek boyutlu uzayda verilen büyük nokta dağılım dizileri için bu dağılımlar arasındaki EMD'nin verdiği global olarak optimal çözüm sayısı birden fazla olabilir. Son kısıtlamadan yola çıkılarak bu tezde yeni bir çalışma sunulmuştur. Sözkonusu yaklaşımda girdi çizgilerinin düğümleri ilk önce geometrik bir uzay içinde temsil edilir. Bu temsil çizge eşleme problemini nokta eşleme problemi olarak yeniden formüle eder. Temsil edilen düğümler(noktalar) daha sonra Earth Mover's Distance(EMD) [26] algoritması kullanılarak eşleştirilirler. Bu tekabüller baz alınarak ortalama çizge hesaplandığından iyi bir tekabül elde etmek algoritmanın başarımında önemli rol oynamaktadır. Diğer yandan, farklı tekabül kümeleri farklı şekil özetlerinin oluşmasına yol açmaktadır. İlk önce EMD algoritmasında aynı girdi nokta dizileri için farklı tekabüllerin nasıl elde edildiği anlatılacaktır.

4. ŞEKİL ÖZETİ İÇİN OPTİMAL ÇÖZÜMLERİ BULMA

Yukarıda, Bölüm 3'te verilen problemin çözümünü bulma işlemi iki aşamada gerçekleşmektedir: Olası bir başlangıç çözümü belirleme ve optimal çözümü bulma. Bizim asıl amacımız ilk optimal çözümü kullanarak alternatif optimal çözümleri bulma olduğu için sonraki bölümlerde bu aşamalar anlatılacaktır.

4.1 Olası bir başlangıç çözümü belirleme

Olası bir başlangıç çözümü, yukarıda EMD formülasyonunda belirtilen (3.2) ve (3.3) kısıtlamalarını sağlayan sıfırdan büyük f_{ij} akış kümesidir. Kaynak olarak n ve hedef olarak m tane nokta verilmiş olsun. Eğer bu problem için bulunan akış sayısı $n + m - 1$ ise bu olası çözüm başlangıç çözümü olarak kabul edilir. North-West corner kuralı, Lowest cost entry metodu ve Russell's approximation metodu, olası bir başlangıç çözümünü bulmak için kullanılan metodlardan bazılarıdır. Russell's approximation metodu ile bulunan çözüm, diğer metotlar ile bulunan çözümlere göre optimal çözüme daha yakındır [19]. EMD algoritması Russell's approximation metodunu kullanmaktadır ve bu metot burada kısaca anlatılacaktır. Bu algoritma, ilk önce kaynak ve hedef çiftleri arasında uzaklığı maksimum olan çiftleri belirler, daha sonra, bunlar arasında farkı en küçük olan (p_i, q_j) kaynak ve hedef çiftini seçerek başlar. Bu noktalar arasında mümkün olan maksimum akış belirlenir ve kısıtlamaların bir tanesi bozulana kadar akış oluşturulabilecek diğer noktalar ile devam edilir. Bu süreç Algoritma 1'de özetlenmiştir. Olası bir başlangıç çözüm bulunduktan sonra, ilk

Algorithm 1 Olası bir başlangıç çözümü belirleme için kullanılan Russell's Approximation Metodu

- 1: Her (p_i) kaynağı için, p_i noktasından hedef noktaya uzaklığı maksimum olan (q_{max}) hedefi belirle.
 - 2: Her (q_j) hedefi için, q_j noktasından kaynak noktaya uzaklığı maksimum olan (p_{max}) kaynağı belirle.
 - 3: Her (p_i, q_j) kaynak ve hedef ikilisi için, Adım 1'de belirtildiği gibi q_{max} 'ı bul, ve Adım 2'de belirtildiği gibi p_{max} 'ı bul. $min = \min\{w_{p_{max}}, w_{q_{max}}\}$, $max = \max\{w_{p_{max}}, w_{q_{max}}\}$ olsun. $\delta_{ij} = min - max$ değerini hesapla.
 - 4: En küçük değeri olan δ 'yı hesapla. (Bu δ_{kl}) olsun ve p_k noktasından q_l noktasına maksimum akışı gönder.
 - 5: $min_{weight} = \min\{w_{p_k}, w_{q_l}\}$ olsun. min_{weight} 'i hem w_{p_k} ve w_{q_l} 'den çıkar, ve ağırlık değeri sıfır olan noktayı (kaynak veya hedef) sil.
 - 6: Kaynak veya hedefte hiçbir nokta kalmayncaya kadar Adım 1-5'i tekrarla.
-

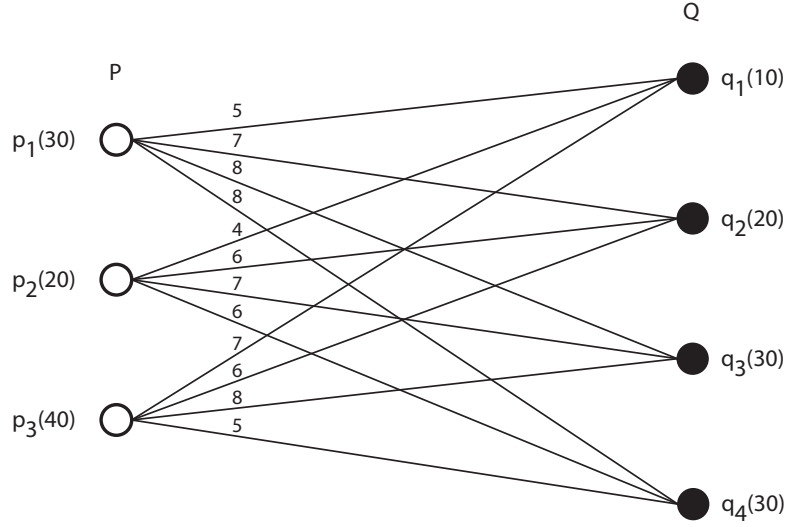
optimal çözüm bulma işlemi ile devam edilecektir.

4.2 İlk optimal çözümü bulma

Olası bir başlangıç çözüm elde edildikten sonra ilk optimal çözüm bulma işlemi ile devam edilir. EMD algoritması optimal çözümü hesaplamak için MODI (Modified Distribution) [34] metodunu kullanmaktadır. Bu metot ilk önce Russell's Approximation metodu ile bulunan başlangıç çözümü ile başlar ve bu çözümün daha da iyileştirilmesinin mümkün olup olmadığını belirler. Bu işlem sonucunda kaynak ve hedef noktaları arasında yeni bir akış matrisi oluşturulur. Algoritma, oluşan bu yeni akış matrisini kullanarak bu işlemi geçerli akış matrisi optimal çözümü verene kadar devam ettirir. Bu süreç Algoritma 2'de özetlenmiştir.

4.3 Alternatif optimal çözümleri bulma

İlk optimal çözüm bulunduktan sonra, o çözüm kullanılarak alternatif optimal çözümler bulunabilir. Bölüm 3'te belirtildiği gibi EMD algoritmasının sonucu toplam maliyeti vermektedir. Alternatif optimal çözümleri bulmak için ilk bulunan optimal çözümün sıfır katkı sağlayarak geliştirilmesinin mümkün olup olmadığı belirlenir. Başka bir deyişle, toplam maliyeti değiştirmeyen diğer akış



Şekil 4.1: İki tane dağılımdan oluşan taşıma problemi örneği, $P = \{(p_1, 30), (p_2, 20), (p_3, 40)\}$ ve $Q = \{(q_1, 10), (q_2, 20), (q_3, 30), (q_4, 30)\}$. Her (p_i, q_j) ikilisinin uzaklığı onları birleştiren doğru üzerinde verilmiştir.

matrislerine bakılır ve böylece alternatif optimal çözümler elde edilir. Yukarıda bahsedilen tüm süreç aşağıda örnek ile detaylı olarak anlatılacaktır.

$P = \{(p_1, 30), (p_2, 20), (p_3, 40)\}$ ve $Q = \{(q_1, 10), (q_2, 20), (q_3, 30), (q_4, 30)\}$ şeklinde verilmiş üç tane kaynak ve 4 tane hedef noktasından oluşan Şekil 4.1'de görüldüğü gibi bir taşıma problemi verilmiş olsun. Burada amaç P ve Q dağılımları arasında toplam maliyeti minimize eden $F = [f_{ij}]$, $(1 \leq i \leq 3, 1 \leq j \leq 4)$ akış matrisini bulmaktır. Bu problemin parametre tablosu Çizelge 4.1'de görüldüğü gibi düzenlenmiştir.

Algoritma, ilk aşamada, Russell's approximation metodunu kullanarak olası başlangıç çözümü bulur. Algoritma 1'de verildiği gibi Russell's approximation metodu önce bütün ikililer arasında uzaklığı minimum olan (p_i, q_j) kaynak ve hedef ikilisini belirler ve bu noktalar arasında maksimum akışı gönderir. Çizelge 4.2'te görüldüğü gibi bütün ikililer arasında maksimum uzaklıkta olan kaynak ve hedef ikilisinin farkı (ya da δ) hesaplanır. Bu durum için en küçük fark -1'dir (aynı değerde olması durumunda en küçük fark isteğe bağlı belirlenebilir) ve (p_3, q_1) ikilisi belirlenmiştir. Bu ikili için akış matrisinde 10 değerinde yeni bir akış oluşturulmuştur. Bu işlem çizelgenin değişmesine yol açmaktadır ve yeni bir çizelgede ağırlığı sıfıra dönüşen q_1 noktası silinmiştir ve $w_{p_3} = 30$ olmuştur (Çizelge 4.3). Yukarıdaki süreç, belirlenecek akış kalmayıncaya kadar f_{ij}

	q_1		q_2		q_3		q_4		Kaynak
p_1	5	δ_{11}	7	δ_{12}	8	δ_{13}	8	δ_{14}	30
	Δ_{11}	f_{11}	Δ_{12}	f_{12}	Δ_{13}	f_{13}	Δ_{14}	f_{14}	
p_2	4	δ_{21}	6	δ_{22}	7	δ_{23}	6	δ_{24}	20
	Δ_{21}	f_{21}	Δ_{22}	f_{22}	Δ_{23}	f_{23}	Δ_{24}	f_{24}	
p_3	7	δ_{31}	6	δ_{32}	8	δ_{33}	5	δ_{34}	40
	Δ_{31}	f_{31}	Δ_{32}	f_{32}	Δ_{33}	f_{33}	Δ_{34}	f_{34}	
Hedef	10		20		30		30		

Çizelge 4.1: Verilen problem için düzenlenen başlangıç parametre tablosu. Her hücrenin üst-sol köşesi maliyeti (uzaklık), üst-sağ köşesi maksimum uzaklıklar arasındaki farkı, alt-sol köşesi maliyet farkını ve alt-sağ köşesi her p_i ve q_j ikilisi arasındaki akışı belirtmektedir.

	q_1		q_2		q_3		q_4		Kaynak
p_1	5	[-1]	7	[-1]	8	[0]	8	[0]	30
	f_{11}		f_{12}		f_{13}		f_{14}		
p_2	4	[0]	6	[0]	7	[-1]	6	[-1]	20
	f_{21}		f_{22}		f_{23}		f_{24}		
p_3	7	[-1]	6	[-1]	8	[0]	5	[0]	40
	f_{31}		f_{32}		f_{33}		f_{34}		
Hedef	10		20		30		30		

Çizelge 4.2: Russell's approximation metodu kullanılarak δ değerlerinin hesaplanması. δ değerleri köşeli parantezlerde verilmiştir.

akış matrisinde yeni akışlar belirlenerek tekrarlanır ve sonucunda Çizelge 4.5'te görüldüğü gibi olası başlangıç çözümü elde edilir.

EMD algoritması, ikinci aşamada ise optimal çözümü bulmak için MODI metodunu kullanır ve Çizelge 4.5'da elde edilen çözüm ile işleme başlar. Algoritma 2'de bahsedildiği gibi MODI metodu sıfırdan farklı her f_{ij} için u_i ve v_j değerlerini belirler ve sıfır değerindeki her f_{ij} için Δ_{ij} uzaklık farkını hesaplar (Çizelge 4.6'ye bakınız, Δ 'lar parantezde verilmiştir).

En küçük Δ_{ij} değeri -4 olduğu için bu durumda buna denk gelen sıfır değerindeki f_{ij} akışı f_{21} 'dir. f_{21} 'den başlayan ve f_{21} 'de biten bir kapalı döngü oluşturulmuştur. Çizelge 4.7'de görüldüğü üzere, f_{21} akışı $+\Theta$, f_{24} , f_{34} ve f_{31} akışları sırası ile $-\Theta$, $+\Theta$ ve $-\Theta$ olarak işaretlenmiştir. Bu kapalı döngü içerisindeki sıfırdan

	q_1	q_2	q_3	q_4	Kaynak
p_1	5	7	8	8	30
p_2	4	6	7	6	20
p_3	7	6	8	5	30
Hedef	0	20	30	30	

Çizelge 4.3: f_{31} 'e 10 değerinde yeni bir akış belirlendikten sonra.

	q_1	q_2	q_3	q_4	Kaynak
p_1	5	7 [-1] f_{12}	8 [0] f_{13}	8 [0] f_{14}	30
p_2	4	6 [0] f_{22}	7 [-1] f_{23}	6 [-1] f_{24}	20
p_3	7	6 [-1] f_{32}	8 [0] f_{33}	5 [0] f_{34}	30
Hedef	0	20	30	30	

Çizelge 4.4: Yeni bir akış belirlendikten sonraki değerlendirme.

farklı değerdeki $-\Theta$ olarak işaretlenmiş akışlar arasındaki en küçük akış $f_{31} = 10$ 'dur. Döngü içerisindeki bu akışlar üzerinde şu şekilde değişiklik yapılacaktır: $+\Theta$ ile işaretlenmiş akışlara 10 eklenecek ve $-\Theta$ ile işaretlenmiş akışlardan 10 çıkartılacaktır. Bu değişim için Çizelge 4.8'a bakınız. Yukarıdaki süreç, verilen çözüm için bütün $\Delta_{ij} \geq 0$ durumları sağlanıncaya kadar devam eder ve bunun sonucunda optimal çözümü verir. Optimal çözüm Çizelge 4.10'de verilmiştir.

Alternatif optimal çözümleri belirlemek için ilk optimal çözüme ulaşırken en son elde edilen Δ_{ij} değerleri ve akış matrisi (ya da Çizelge 4.10) kullanılacaktır. Bölüm 4.3'te bahsedildiği gibi ne kadar sıfır değerli Δ_{ij} varsa o kadar alternatif optimal çözüm var demektir. Çizelge 4.11 verilen çözüm için Δ_{ij} değerlerini göstermektedir. İki tane sıfır değerli Δ_{ij} vardır ve böylece bu problem için iki tane alternatif optimal çözüm bulunmaktadır. İlk alternatif optimal çözümü bulmak için bu durumda ilk önce en küçük Δ_{ij} olan Δ_{11} seçilmiştir. Bu Δ_{11} 'e denk gelen akış f_{11} 'dir. Algoritma 2'de verildiği gibi burada da kapalı döngünün belirlenmesi gerekmektedir. f_{11} akışı için kapalı döngü şu şekilde

	q_1	q_2	q_3	q_4	Kaynak
p_1	5	7 20	8 10	8	30
p_2	4	6	7	6 20	20
p_3	7 10	6	8 20	5 10	40
Hedef	10	20	30	30	

Çizelge 4.5: Elde edilen olası başlangıç çözümü. Akışlar şu şekilde belirlenmiştir: $f_{12}:20$, $f_{13}:10$, $f_{24}:20$, $f_{31}:10$, $f_{33}:20$ ve $f_{34}:10$.

	q_1	q_2	q_3	q_4	Kaynak
	$v_1 = 7$	$v_2 = 7$	$v_3 = 8$	$v_4 = 5$	
p_1 $u_1 = 0$	5 (-2) f_{11}	7 20	8 10	8 (3) f_{14}	30
p_2 $u_2 = 1$	4 (-4) f_{21}	6 (-2) f_{22}	7 (-2) f_{23}	6 20	20
p_3 $u_3 = 0$	7 10	6 (-1) f_{32}	8 20	5 10	40
Hedef	10	20	30	30	

Çizelge 4.6: u_i (solda), v_j (yukarıda) ve Δ (parantezde) değerleri hesaplanırken.

bulunmuştur: f_{11} akışı $+\Theta$, f_{13} , f_{23} ve f_{21} akışları sırası ile $-\Theta$, $+\Theta$ ve $-\Theta$ olarak işaretlenmiştir (Çizelge 4.12'e bakınız). Bu kapalı döngü içerisindeki sıfırdan farklı değerdeki $-\Theta$ olarak işaretlenmiş akışlar arasındaki en küçük akış $f_{21} = 10$ 'dur. Döngü içerisindeki bu akışlar üzerinde şu şekilde değişiklik yapılacaktır: $+\Theta$ ile işaretlenmiş akışlara 10 eklenecek ve $-\Theta$ ile işaretlenmiş akışlardan 10 çıkartılacaktır. Bu değişim Çizelge 4.8'da görüldüğü gibi bize ilk optimal çözümü vermektedir. İkinci alternatif çözümü elde etmek için aynı süreç uygulanmaktadır ve Çizelge 4.14'teki çözüm elde edilmiştir.

	q_1	q_2	q_3	q_4	Kaynak
p_1	5 (-2)	7 20	8 10	8 (3)	30
p_2	4 + Θ (-4)	6 (-2)	7 (-2)	6 - Θ 20	20
p_3	7 - Θ 10	6 (-1)	8 20	5 + Θ 10	40
Hedef	10	20	30	30	

Çizelge 4.7: En küçük Δ değerini belirleme (Δ_{21}) ve kapalı döngüyü oluşturma. Kapalı döngü $\pm\Theta$ ile işaretlenerek oluşturulmaktadır. Burada, kapalı döngü şu şekildedir: f_{21} , f_{24} , f_{31} ve f_{34} .

	q_1	q_2	q_3	q_4	Kaynak
p_1	5	7 20	8 10	8	30
p_2	4 10	6	7	6 10	20
p_3	7	6	8 20	5 20	40
Hedef	10	20	30	30	

Çizelge 4.8: Akış değişiklikleri sonrasındaki Çizelge. f_{21} , f_{24} , f_{31} ve f_{34} akışları değişmiştir.

	q_1	q_2	q_3	q_4	Kaynak
p_1	5 (2)	7 20	8 10	8 (3)	30
p_2	4 10	6 (-2)	7 (-2)	6 10	20
p_3	7 (4)	6 (-1)	8 20	5 20	40
Hedef	10	20	30	30	

Çizelge 4.9: Sonraki tekrarlama, Δ değerlerini belirleme.

Algorithm 2 MODI metodu kullanarak optimal çözümü bulma algoritması

- 1: Algoritma 1 tarafından elde edilen olası başlangıç çözüm ile başla. Bu çözüm sonucu oluşan p_i ve q_j noktaları arasındaki akış matrisi f_{ij} olsun.
 - 2: $u_1 \leftarrow 0$.
 - 3: Sıfırdan farklı olan her f_{ij} için, $v_j \leftarrow d_{1j}$.
 - 4: Hesaplanan her u_i, v_j, f_{ij} kullanarak kalan u_i ve v_j değerlerini şu şekilde hesapla: $d_{ij} = u_i + v_j$.
 - 5: $\Delta_{ij} = d_{ij} - (u_i + v_j)$ formülünü kullanarak sıfır değerli her f_{ij} akış için Δ_{ij} maliyet farkını hesapla.
 - 6: Δ_{ij} değerlerini şu şekilde değerlendir:
 - (i) Eğer bütün $\Delta_{ij} > 0$, f_{ij} akış matrisini döndür (çözüm optimaldir ve [unique]'tir).
 - (ii) Eğer bütün $\Delta_{ij} \geq 0$ ve en az bir tane $\Delta_{ij} = 0$, f_{ij} akış matrisini döndür (çözüm optimaldir ve alternatif optimal çözümler mevcuttur).
 - (iii) Eğer en az bir tane $\Delta_{ij} < 0$, çözüm optimal değildir ve daha iyileştirilmesi gerekmektedir. Adım 7 ile devam et.
 - 7: $\Delta_{kl} \leftarrow$ en küçük Δ ve f_{kl}, Δ_{kl} 'e denk gelen akış olsun.
 - 8: Kaynak ve hedef noktaların ağırlıklarını dengelemek için şu şekilde f_{kl} ile başlayan ve f_{kl} ile biten bir kapalı döngü oluştur:
 - (i) f_{kl} 'i $+\Theta$ olarak işaretle (f_{kl} yeni bir akış kabul ediyor).
 - (ii) f_{kl} yeni bir akış kabul ettiği zaman döngüyü ağırlıkları etkilenmiş kaynak ve hedef noktaları ile oluştur.
 - (iii) Döngü içerisinde f_{kl} ile başlayarak tüm noktaları sırası ile ziyaret et ve $+\Theta$ ve $-\Theta$ olarak yer değiştirerek işaretle.
 - (iv) $\min\{f_{loop}\} \leftarrow$ en küçük $-\Theta$ olarak işaretlenmiş akış.
 - (v) $\min\{f_{loop}\}$ değerinde her $+\Theta$ ile işaretlenmiş akışları artır ve her $-\Theta$ ile işaretlenmiş akışları eksilt (bu adım toplam maliyeti $\Delta_{kl} \times d_{kl}$ değerinde eksiltmektedir).
 - 9: Tüm $\Delta_{ij} \geq 0$ olana kadar Adım 1-8'i tekrarla.
-

	q_1	q_2	q_3	q_4	Kaynak
p_1	5	7 10	8 20	8	30
p_2	4 10	6	7 10	6	20
p_3	7	6 10	8	5 30	40
Hedef	10	20	30	30	

Çizelge 4.10: Son optimal çözüm. Bu çözüm için akış matrisi: $f_{12}:10$, $f_{13}:20$, $f_{21}:10$, $f_{23}:10$, $f_{32}:10$ ve $f_{34}:30$.

	q_1	q_2	q_3	q_4	Kaynak
p_1	5 (0)	7 10	8 20	8 (2)	30
p_2	4 10	6 (0)	7 10	6 (1)	20
p_3	7 (3)	6 10	8 (1)	5 30	40
Hedef	10	20	30	30	

Çizelge 4.11: Δ değerlerini belirle, $\Delta_{ij} = 0$ alternatif optimal çözümü vermektedir. Δ_{11} ve Δ_{22} sıfır değerindedir ve bu da demektir ki bu problemin iki tane alternatif çözümü mevcuttur.

	q_1	q_2	q_3	q_4	Kaynak
p_1	5 (0)	$+\Theta$ 7 10	8 20	$-\Theta$ 8 (2)	30
p_2	4 10	$-\Theta$ 6 (0)	7 10	$+\Theta$ 6 (1)	20
p_3	7 (3)	6 10	8 (1)	5 30	40
Hedef	10	20	30	30	

Çizelge 4.12: Sıfır değerli Δ_{11} için kapalı döngü oluşturma. Kapalı döngü şu şekildedir: f_{11} , f_{13} , f_{21} ve f_{23} .

	q_1	q_2	q_3	q_4	Kaynak
p_1	5 10	7 10	8 10	8	30
p_2	4	6	7 20	6	20
p_3	7	6 10	8	5 30	40
Hedef	10	20	30	30	

Çizelge 4.13: Şekil 4.1’de verilen problem için ilk alternatif optimal çözüm. Akışlar şu şekilde bulunmuştur: $f_{11}:10$, $f_{12}:10$, $f_{13}:10$, $f_{23}:20$, $f_{32}:10$ ve $f_{34}:30$.

	q_1	q_2	q_3	q_4	Kaynak
p_1	5	7	8 30	8	30
p_2	4 10	6 10	7	6	20
p_3	7	6 10	8	5 30	40
Hedef	10	20	30	30	

Çizelge 4.14: İkinci alternatif optimal çözüm bulunduktan sonraki akış matrisi: $f_{13}:30$, $f_{21}:10$, $f_{22}:10$, $f_{32}:10$ ve $f_{34}:30$.

5. ÖZET ŞEKİL ÇIKARMA

$S = \{s_1, \dots, s_r\}$ aynı problem için elde edilen farklı optimal çözümler kümesi olsun. Bu çözümler şekil özeti oluşturma açısından komşuluk ilişkilerinin ne kadar korunduğuna bakılarak sıralanır. İyi bir optimal çözümde, birinci girdi kümesindeki komşu noktaların tekabül ettiği ikinci girdi kümesindeki noktaların da komşu olması gerekir. Örnek olarak, Şekil 1.4'te görülen iki farklı optimal çözümden (a) kısmındaki çözüm, (b)'ye göre komşuluk ilişkilerini koruduğundan dolayı daha iyi bir çözümdür. Yukarıda bahsedildiği gibi, $P = \{(p_1, w_{p1}), \dots, (p_n, w_{pn})\}$ ve $Q = \{(q_1, w_{q1}), \dots, (q_m, w_{qm})\}$ girdi dağılımları olsun. $N_p = \{n_1, \dots, n_k\}$ herhangi bir $p \in P$ ($N_p \in P$) noktasının k tane en yakın komşusu olsun. Optimal çözüm $s' \in S$ N_p 'yi herhangi bir $M_p = \{m_1, \dots, m_l\}$ ($M_p \in Q$) kümesine tekabül ettiğini varsayalım. Optimal çözümleri sıralamak için N_p ve M_p kümelerindeki ortalama uzaklık arasındaki farkın mutlak değeri kullanılmaktadır. Biçimsel olarak, N_p ve M_p arasındaki eşleme kalitesi aşağıdaki gibi hesaplanmıştır:

$$\begin{aligned} D_{N_p} &= \frac{1}{k^2} \sum_{i=1}^k \sum_{j=1}^k d_{n_i, n_j} \\ D_{M_p} &= \frac{1}{l^2} \sum_{i=1}^l \sum_{j=1}^l d_{m_i, m_j} \\ D_{N_p M_p} &= |D_{N_p} - D_{M_p}| \\ Q_{N_p M_p} &= \frac{1}{1 + D_{N_p M_p}} \end{aligned}$$

burada d_{n_i, n_j} (veya d_{m_i, m_j}) noktalar arasındaki uzaklıktır.

Genel olarak, optimal çözüm $s' \in S$ ile ilişkili olan kalite değeri aşağıdaki gibi hesaplanmıştır:

$$Q(s') = \sum_{p \in P} Q_{N_p M_p}.$$

Optimal çözümler sıralandıktan sonra, en üst sıradaki optimal çözüm ile bulunan tekabüller baz alınarak şekil özeti oluşturma prosedürü başlatılır. Bu tekabüllere dayalı yönlü ortalama (oriented averaging) işlemi kullanılarak bir özet oluşturulur. Özellikle, her tekabül için nitelikleri normalize edilen, eşlemeye ilişkili noktaların niteliklerinin ağırlıklı ortalaması olan bir nokta eklenir.

Daha biçimsel olarak, $P' = \langle p'_1, \dots, p'_k \rangle$, tek bir $q' \in Q$ noktası ile eşleşmiş P dağılımındaki noktaların alt kümesi olsun. Her bir (P', q') tekabülü için sıralanmış ortalama diziyi hesaplamak için özete bir tane a noktası eklenmektedir ve bu noktanın yarıçapı (r_a), x ve y koordinatları (x_a, y_a) gibi özellikleri (attributes) şu şekilde hesaplanmaktadır:

$$\begin{aligned} r_a &= \frac{1}{2w_a} \left[\sum_{p' \in P'} r_{p'} \times f_{p',q'} + r_{q'} \right], \\ x_a &= \frac{1}{2w_a} \left[\sum_{p' \in P'} x_{p'} \times f_{p',q'} + x_{q'} \right], \\ y_a &= \frac{1}{2w_a} \left[\sum_{p' \in P'} y_{p'} \times f_{p',q'} + y_{q'} \right], \end{aligned}$$

burada, $w_a = \sum_{p' \in P'} f_{p',q'}$ 'dir ve $f(p', q')$, EMD algoritması ile elde edilen p' noktasından q' noktasına gönderilen akıştır.

6. DENEYLER

Bu bölümde ilk önce deneyler için kullandığımız veri dizileri sunulmuştur. Daha sonra sunulan metodun bir önceki şekil özeti oluşturma algoritması ile karşılaştırılması dahil olmak üzere deneysel değerlendirilmesi anlatılmıştır.

6.1 Kullanılan veritabanları

Deneyler için beş tane değişik veri dizisi kullanılmıştır. Aslan ve Tari veritabanı [5], her bir sınıfta 4 tane görüntü olmak üzere 14 tane sınıftan oluşmaktadır (Şekil 6.1). Rutgers Tools veritabanında [29] 26 tane (18 alet ve 8 biyolojik) şekil vardır (Şekil 6.2). Üçüncü veritabanı olarak 20 tane sınıfın her birinde 70 değişik görüntü olmak üzere 1400 silüetten oluşan MPEG-7 şekil veritabanı [18] kullanılmıştır. Şekil 6.3'te veritabanından alınan bazı sınıflar görülmektedir. Kullanılan son veritabanı ise ETU veritabanıdır (Şekil 6.4). ETU, 10 tane sınıf içinde 72 tane görüntü olmak üzere 720 silüetten oluşmaktadır.

6.2 Değerlendirme

Veritabanı görüntüleri için eşleme algoritması ile hesaplanan uzaklık kullanılarak birbirine en çok benzeyen ikililer alt dizisi ve onlara ait özet görüntüler seçilmiştir. Karşılaştırma için hem sunulan hem de bir önceki yaklaşım kullanılarak özet görüntüler çıkarılmıştır. Bir önceki yaklaşım, özet görüntüyü EMD algoritması ile bulunan ilk tekabülü kullanarak elde ederken [11], sunulan yaklaşım Bölüm 5'te bahsedilen kritere göre hesaplanan en yüksek derecedeki optimal çözümün verdiği



Şekil 6.1: 56 tane şekilden oluşan Aslan ve Tari veritabanı. Her biri için 4'er şekil olmak üzere 14 tane sınıf vardır.



Şekil 6.2: 26 tane şekilden oluşan Rutgers Tools veritabanı (İlk satır 18 alet şekli ve son satır 8 biyolojik şekli göstermektedir).

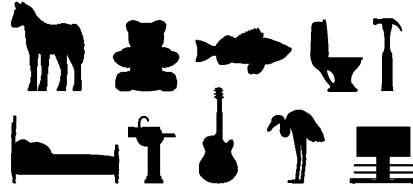
tekabülü kullanmaktadır.

İlk deneyde, yukarıda belirtilen veri kümesinden alınan bazı sınıfların şekil özetleri oluşturulmuştur. Şekil 6.5'te ilk iki ve son iki sütun farklı iki tane görüntüyü ve onların iskeletlerini temsil etmektedir. Ortadaki iki sütun ise bu iki tane görüntüden oluşturulan şekil özetini ve iskeletini ifade etmektedir. Burada, şekil özetini oluşturmak için en yüksek dereceye ait optimal çözüm kullanılmıştır.

Sonraki deneyde ise sorgu görüntüsünün ve bu görüntüye en benzer olan veritabanı görüntüsünün aynı sınıfta olup olmadığı değerlendirilmiştir. Burada sorgu görüntüleri bütün görüntüler için (her sınıftaki) oluşturulan özet görüntülerdir. Veritabanı görüntüleri ise asıl görüntülerdir (her sınıf için). Sorgu görüntüsü ve veritabanı görüntüsü aynı nesne sınıfında ise, yani, sorguladığımız özet görüntünün sınıfı, kendisine en benzer görüntünün sınıfı ile aynı ise karşılaştırma başarılı olarak değerlendirilmiştir. Her veritabanı için sonuçlar








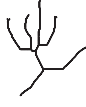





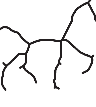
























Şekil 6.3: MPEG-7 veritabanı için bazı nesnelere gösterilmiştir. 40 tane sınıfın görüntüleri verilmiştir (toplam 70 tane sınıftan oluşmaktadır).



Şekil 6.4: ETU veritabanı için 10 tane sınıfın görüntüleri verilmiştir.

aşağıda verilmektedir. Aslan ve Tari veritabanı için bir önceki yaklaşım ile karşılaştırma deneyin 85.7%'inde başarılı iken, sunulan yaklaşım için bu oran 91.1%'e yükselmiştir. Diğer veritabanları için sonuçlar şöyledir: Rutgers Tools veritabanı için: 76% ve 100%, MPEG-7 veritabanı için: 89.8% ve 94.4%, ve ETU veritabanı için: 99.3% ve 100%. Genel olarak bu sonuçlar sunulan yaklaşımın şekil özeti oluşturma işlemine katkı sağladığını göstermektedir.

Nesne tanıma deneylerinde, sunulan metodun sağladığı gelişmenin, bulunan optimal çözüm sayısı ile doğrudan ilişkili olduğu göz önüne alınmalıdır. Sadece bir tane optimal çözüm olduğu durumda bir önceki ve sunulan algoritma aynı özeti vermektedir. Birbirine en çok benzeyen çiftler için, veritabanı kullanılarak, özet görüntülerin oluşturulduğu dikkate alınmalıdır. Daha büyük veri kümesi için birden fazla optimal çözüm sayısının ve önerilen çalışmanın performansının artacağı beklenmektedir.

Nesne 1	İskelet 1	Özet	İskelet	Nesne 2	İskelet 2
					
					
					
					
					
					

Şekil 6.5: Birbirine yakın iki tane nesne görüntüsü ve bu iki tane görüntüden oluşturulan özet görüntü gösterilmektedir. İlk iki ve son iki sütun farklı iki tane görüntüyü ve onların iskeletlerini temsil etmektedir. Ortadaki iki sütun ise bu iki tane görüntüden oluşturulan şekil özetini ve iskeletini ifade etmektedir.

7. ÖZET VE SONUÇLAR

Nesne sınıflandırma işlemi, bir görüntü içindeki bir nesnenin örneklerini bulma ve tanıma sürecidir. Bir çok nesne sınıflandırma yöntemi vardır ve bunlardan biri de şekil tabanlı nesne sınıflandırma yöntemidir. Şekil tabanlı yöntemlerde nesnelerin alanı, silüetleri, kuşatan kutusu ve bulunan alanın gradyanı gibi iki boyutlu bilgiler kullanılmaktadır. Şekil tabanlı nesne sınıflandırma metodlarından biri de şekil özeti oluşturmaktır ve araştırmacıların karşılaştıkları en önemli problemlerden biridir. Bu problem, şimdiye kadar, birçok farklı bağlamda incelenmiştir. Kapalı yüzeyler (closed surfaces) [7], şekil gramerleri (shape grammars) [36], fonksiyonel nesne tanımlamaları (functional object descriptions) [33] ve çizgeler (graphs) [15] gibi yaklaşımlar bu çalışmalardan sadece bazılarıdır. Bu teknikler Bölüm 2’de detaylı olarak anlatılmıştır.

Genel olarak bu tezin organizasyonu şu şekildedir. 3. bölümde ilk önce genel şekil tanıma problemi tanımlanmıştır ve şeklin iskeletsel çizgesini çıkarma işlemi anlatılmıştır. Daha sonra, örnek şekiller dizisi üzerinden şekil-tabanlı kategorik modelini otomatik öğrenen, Demirci ve ark.’nın önerdiği çalışma [11], "Örneklerden iskeletsel özet oluşturma" başlığı altında anlatılmıştır. Ayrıca, [11]’de kullanılan EMD algoritmasına yer verilmiştir. Son olarak, bir önceki çalışmanın eksiklikleri belirtilerek, sunulan çalışmaya giriş yapılmıştır. 4. bölümde, Bölüm 3’te verilen problemin çözümünü bulma işlemi iki aşamada anlatılmıştır. 5. bölümde önce elde edilen farklı optimal çözümler kümesi için komşuluk ilişkilerinin ne kadar korunduğuna bakılarak sıralama yapılmaktadır. Daha sonra en üst sıradaki optimal çözüm ile bulunan tekabüller baz alınarak şekil özeti oluşturma prosedürü anlatılmıştır. 6. bölümde ise ilk önce deneyler için kullandığımız veri dizileri sunulmuştur ve sunulan metodun bir önceki şekil özeti oluşturma algoritması ile karşılaştırılması dahil olmak üzere deneysel

değerlendirilmesi anlatılmıştır.

Demirci ve ark. özet çizgeyi yaygın olarak tanınmış taşıma probleminden elde edilen optimal çözüme dayalı düğüm tekabülünden hesaplamıştır [11]. Birden fazla optimal çözümün olduğu duruma dikkate alınarak, bu tezde, bütün optimal çözümler, eşleşmiş kümeler arasındaki yerel komşuluk ilişkilerini ne kadar koruduğuna dayanarak sıralanmıştır. Bir önceki yaklaşım [11], özet çizgeyi, bulunan ilk optimal çözümü kullanarak hesaplar iken, önerdiğimiz yaklaşım en yüksek dereceye sahip olan optimal çözümü kullanarak özet çizgeyi hesaplar. Bu tekniğin deneysel değerlendirmesi bir önceki çalışmaya göre daha başarılı sonuçlar vermiştir.

Gelecek çalışmalar olarak neler yapılabilir: Ek karşılaştırma deneyleri dahil olmak üzere, daha büyük veri kümeleri kullanılarak kapsamlı deneyler yapılabilir. Elde edilen sonuçlar, diğer şekil-tabanlı (sadece orta eksenli değil) tanıma algoritmaları ile karşılaştırılabilir. Optimal çözümleri hesaplamak için çeşitli kriterler, örneğin, karmaşıklık ve yürütme zamanı vs. incelenebilir. En önemlisi yakınsama algoritmaları kullanılarak problem için optimal çözüme yakın birçok çözüm bulunarak değerlendirilebilir.

KAYNAKLAR

- [1] R. Agarwala, V. Bafna, M. Farach, M. Paterson, and M. Thorup. On the approximability of numerical taxonomy (fitting distances by tree metrics). *SIAM J. Comput.*, 28(3):1073–1085, Feb. 1999.
- [2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*, pages 4–7. Prentice Hall, Englewood Cliffs, New Jersey, 1993.
- [3] M. Akimaliev and M. F. Demirci. Shape abstraction through multiple optimal solutions. In *Proceedings of the 7th international conference on Advances in visual computing - Volume Part II*, ISVC’11, pages 588–596, Berlin, Heidelberg, 2011. Springer-Verlag.
- [4] C. Aslan. Disconnected skeletons for shape recognition. Master’s thesis, Department of Computer Engineering, Middle East Technical University, Ankara, Turkey, 2005.
- [5] C. Aslan and S.Tari. An axis based representation for recognition. *ICCV*, 2(10):1339–1346, 2005.
- [6] H. Blum. Biological shape and visual science (part i). *Journal of Theoretical Biology*, 38(2):205 – 287, 1973.
- [7] E. Chen and R. Parent. Shape averaging and its applications to industrial design. *IEEE Computer Graphics and Applications*, 9(1):47–54, January 1989.
- [8] Z. Chi and S. Geman. Estimation of probabilistic context-free grammars. *Comput. Linguist.*, 24(2):299–305, June 1998.

- [9] A. D. J. Cross, R. C. Wilson, and E. R. Hancock. Inexact graph matching using genetic search. *Pattern Recognition*, 30(6):953–970, 1997.
- [10] F. Demirci, A. Shokoufandeh, and S. Dickinson. Skeletal shape abstraction from examples. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):944–952, 2009.
- [11] F. Demirci, A. Shokoufandeh, Y. Keselman, L. Bretzner, and S. Dickinson. Object recognition as many-to-many feature matching. *International Journal of Computer Vision*, 69(2):203–222, 2006.
- [12] M. F. Demirci, A. Shokoufandeh, Y. Keselman, L. Bretzner, and S. Dickinson. Object recognition as many-to-many feature matching. *Int. J. Comput. Vision*, 69(2):203–222, Aug. 2006.
- [13] S. Dickinson. *The Evolution of Object Categorization and the Challenge of Image Abstraction*, pages 1–37. Cambridge University Press, 2009.
- [14] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching, 1996.
- [15] X. Jiang, A. Munger, and H. Bunke. On median graphs: properties, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10), 2001.
- [16] K. A. D. Jong, K. A. De, J. William, and W. M. Spears. Using genetic algorithms to solve np-complete problems, 1989.
- [17] J. Porway, Z. Y. Yao, and S. C. Zhu. Learning an and-or graph for modeling and recognizing object categories. Technical report, Department of Statistics, 2007.
- [18] L. J. Latecki, R. Lakämper, and U. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. *CVPR*, 1:1424–1429, 2000.
- [19] T.-S. Lee. A complete russel’s method for the transportation problem. *Society for Industrial and Applied Mathematics*, 28(4):547–549, 1986.
- [20] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *IEEE Conference on Computer Vision and Pattern Recognition CVPR’03*, pages 409–415, 2003.

- [21] D. Marr. *Vision*. W.H. Freeman, 1982.
- [22] D. Marr. *Vision : A computational investigation into the human representation and processing of visual information*. Freeman, San Francisco, CA/, 1982.
- [23] J. Matousek. On embedding trees into uniformly convex banach spaces, 1999.
- [24] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.
- [25] M. Pelillo, K. Siddiqi, and S. W. Zucker. Matching hierarchical structures using association graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:1105–1120, 1998.
- [26] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover’s distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.
- [27] L. G. Shapiro and R. M. Haralick. Structural descriptions and inexact matching. Technical report, Virginia Polytechnic Institute, Department of Computer Science, 1979.
- [28] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching, 1998.
- [29] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. *Int. J. Comput. Vision*, 35:13–32, November 1999.
- [30] S. Thorpe, D. Fize, and C. Marlot. Speed of processing in the human visual system. *Nature*, 381:520 – 522, 1996.
- [31] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. Image parsing: Unifying segmentation, detection, and recognition, 2005.
- [32] Y.-K. Wang, K.-C. Fan, and J.-T. Horng. Genetic-based search for error-correcting graph isomorphism. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 27(4):588–597, 1997.

- [33] P. Winston, T. Binford, B. Katz, and M. Lowry. Learning physical description from functional descriptions, examples, and precedents. In *Proceedings, AAAI*, pages 433–439, Palo Alto, CA, August 1983.
- [34] W. L. Winston. *Operations Research Applications and Algorithms*. Duxbury Resource Center, 1987.
- [35] B. Yao, X. Yang, and S.-C. Zhu. Introduction to a large-scale general purpose ground truth database: methodology, annotation tool and benchmarks. In *Proceedings of the 6th international conference on Energy minimization methods in computer vision and pattern recognition, EMMCVPR'07*, pages 169–183, Berlin, Heidelberg, 2007. Springer-Verlag.
- [36] S.-C. Zhu and D. Mumford. A stochastic grammar of images. *Foundations and Trends in Computer Graphics and Vision*, 2:259–362, January 2006.
- [37] S. C. Zhu, Y. N. Wu, and D. Mumford. Minimax entropy principle and its application to texture modeling, 1997.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : AKİMALİEV, Marlen
Uyruğu : Kırgızistan
Doğum tarihi ve yeri : 02.06.1983 Baetov
Medeni hali : Evli
Telefon : +90 555 701 22 83
e-mail : makimaliev@.etu.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Y. Lisans	TOBB Ekonomi ve Teknoloji Üniversitesi	2011
Lisans	Kırgızistan-Türkiye Manas Üniversitesi	2004

İş Deneyimi

Yıl	Yer	Görev
May '12 - Ara '12	TOBB ETÜ, Bilgisayar Müh. Bölümü	Bölüm Asistanı
Oca '11 - Mar '12	TOBB ETÜ, TÜBİTAK Kariyer Projesi	Araştırma Görevlisi
Tem '09 - May '10	SEKOM İletişim	Video Tekn. Mühendisi
Kas '08 - May '09	IRIS Bilgi Teknolojileri Ltd. Şti.	Programcı
Eki '07 - Oca '08	Fenikeli Ltd.	Programcı

Yabancı Dil

İngilizce (Çok iyi)
Türkçe (Çok iyi)
Rusça (Çok iyi)

Yayınlar

M. Akimaliev and M. Fatih Demirci. Forming shape abstractions using optimal

solutions. IEEE 20th Signal Processing and Communications Applications Conference (SIU), Ölüdeniz, Fethiye, Turkey, April 18-20, 2012.

M. Akimaliev and M. Fatih Demirci. Shape Abstraction through Multiple Optimal Solutions. 7th International Symposium on Visual Computing (ISVC11) Las Vegas, Nevada, September 26-28, 2011. (Lecture Notes in Computer Science, 2011, Volume 6939/2011, pp588-596)