





**YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ İLE  
GÖRÜNTÜ KÜMELEME VE GÖRÜNTÜ ERİŞİMİ**

**YÜKSEK LİSANS TEZİ**

**Ayşegül KARCILI**

**Matematik Mühendisliği Anabilim Dalı**

**Matematik Mühendisliği Programı**

**MAYIS 2018**



**YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ İLE  
GÖRÜNTÜ KÜMELEME VE GÖRÜNTÜ ERİŞİMİ**

**YÜKSEK LİSANS TEZİ**

**Ayşegül KARCILI  
(509131051)**

**Matematik Mühendisliği Anabilim Dalı**

**Matematik Mühendisliği Programı**

**Tez Danışmanı: Doç. Dr. Burcu TUNGA**

**MAYIS 2018**



İTÜ, Fen Bilimleri Enstitüsü'nün 509131051 numaralı Yüksek Lisans Öğrencisi Ayşegül KARCILI, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ İLE GÖRÜNTÜ KÜMELEME VE GÖRÜNTÜ ERİŞİMİ” başlıklı tezini aşağıdaki imzaları olan jüri önünde başarı ile sunmuştur.

**Tez Danışmanı :**      **Doç. Dr. Burcu TUNGA** .....  
İstanbul Teknik Üniversitesi

**Jüri Üyeleri :**        **Doç. Dr. Ahmet KIRIŞ** .....  
İstanbul Teknik Üniversitesi

**Dr. Öğr. Üyesi Murat DEMİRER** .....  
Üsküdar Üniversitesi

**Teslim Tarihi :**      **4 Nisan 2018**

**Savunma Tarihi :**    **8 Mayıs 2018**







*Annem, babam ve kardeşlerime...*



## ÖNSÖZ

Bu tez çalışmasını başarıyla tamamlamamda, özgün yöntemin tasarımından bilimsel çalışmanın yapılması ve geliştirilmesine kadar her aşamada, tüm zorluklara rağmen yüreklendiren ve zaman ayırarak geri bildirimlerle destekleyen, danışman hocam Doç. Dr. Burcu Tunga'ya çok teşekkür ederim.

Savunma sınavı jüri üyesi hocalarım Doç Dr. Ahmet Kırış ve Dr. Murat Demirer'e, tezime gösterdikleri ilgi için ayrıca teşekkür ederim.

Son olarak, bu zorlu süreçte verdiği moral ve motivasyon için ve tüm manevi katkıları için, anneme minnettarım.

Mayıs 2018

Ayşegül KARCILI



## İÇİNDEKİLER

### Sayfa

ÖNSÖZ .....	vii
İÇİNDEKİLER .....	ix
KISALTMALAR.....	xi
ÇİZELGE LİSTESİ.....	xiii
ŞEKİL LİSTESİ.....	xvii
ÖZET .....	xix
SUMMARY .....	xxi
<b>1. GİRİŞ.....</b>	<b>1</b>
1.1 Görüntü Erişimi.....	2
1.2 Kümeleme.....	3
1.3 Boyut İndirgeme .....	4
1.4 Özgün Çalışma .....	6
<b>2. YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ.....</b>	<b>9</b>
2.1 Yüksek Boyutlu Model Gösterilimi (YBMG) ile Fonksiyon Ayrıştırımı .....	9
2.2 Yüksek Boyutlu Model Gösterilimi ile Veri Ayrıştırımı .....	13
<b>3. MATEMATİKSEL YÖNTEMLER .....</b>	<b>17</b>
3.1 YBMG Yönteminin Görüntüler Üzerinde Uygulanışı .....	17
3.2 Kümeleme Yöntemleri .....	19
3.2.1 Eşit aralık yöntemi.....	20
3.2.2 K-ortalamlar yöntemi.....	21
3.2.3 Eşit üye sayısı yöntemi.....	23
3.3 Küme Sayısı Seçimi .....	24
3.4 Karesel Öklit Uzaklığı.....	26
3.5 Azalan Kümülatif Kazanç .....	27
<b>4. ÖZGÜN BİR YÖNTEM OLARAK YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ İLE GÖRÜNTÜ KÜMELEME VE GÖRÜNTÜ ERİŞİMİ ...</b>	<b>29</b>
4.1 Eşit Aralık Yöntemi ile $f_0$ Değerlerini Kümeleme.....	32
4.1.1 $f_1$ değerleri ile görüntü erişimi .....	32
4.1.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi.....	34
4.1.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi .....	35
4.2 K-ortalamlar Yöntemi ile $f_0$ Değerlerini Kümeleme.....	39
4.2.1 $f_1$ değerleri ile görüntü erişimi .....	39
4.2.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi.....	40
4.2.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi .....	40
4.3 Eşit Üye Sayısı Yöntemi ile $f_0$ Değerlerini Kümeleme .....	41
4.3.1 $f_1$ değerleri ile görüntü erişimi .....	45
4.3.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi.....	45

4.3.3 Tek deęişkenli yaklařtırım deęerleri ile görüntü erişimi .....	46
4.4 K-ortalamar Yöntemi ile $f_3$ Deęerlerini Kümeleme.....	47
4.4.1 $f_1$ deęerleri ile görüntü erişimi .....	47
4.4.2 Tek deęişkenli terimlerin uç uca eklenmesi ile görüntü erişimi .....	51
4.4.3 Tek deęişkenli yaklařtırım deęerleri ile görüntü erişimi .....	52
4.5 Eřit Aralık Yöntemi ile $f_3$ -RGB Deęerlerini Kümeleme .....	52
4.5.1 $f_1$ deęerleri ile görüntü erişimi .....	56
4.5.2 Tek deęişkenli terimlerin uç uca eklenmesi ile görüntü erişimi .....	56
4.5.3 Tek deęişkenli yaklařtırım deęerleri ile görüntü erişimi .....	57
<b>5. KARŐILAŐTIRMA .....</b>	<b>63</b>
5.1 YBMG Yönteminin Kümeleme Yapılması ve Kümeleme Yapılmaması Durumlarındaki Görüntü Eriřimi Performansının Karőılařtırılması .....	63
5.2 YBMG Yöntemi ile Temel Bileřen Analizi Yönteminin Görüntü Eriřimi Performansının Karőılařtırılması .....	69
<b>6. SONUÇ VE ÖNERİLER .....</b>	<b>81</b>
<b>KAYNAKLAR.....</b>	<b>83</b>
<b>ÖZGEÇMİŐ .....</b>	<b>87</b>

## **KISALTMALAR**

<b>YBMG</b>	: Yüksel Boyutlu Model Gösterilimi
<b>HDMR</b>	: High Dimensional Model Representation
<b>İTGE</b>	: İçerik Tabanlı Görüntü Erişimi
<b>CBIR</b>	: Content Based Image Retrieval
<b>TBA</b>	: Temel Bileşen Analizi
<b>PCA</b>	: Principal Component Analysis
<b>COIL</b>	: Columbia Object Image Library
<b>RGB</b>	: Red - Green - Blue
<b>AKK</b>	: Azalan Kümülatif Kazanç
<b>DCG</b>	: Discounted Cumulative Gain





## ÇİZELGE LİSTESİ

### Sayfa

<b>Çizelge 4.1</b>	: Eşit aralık yöntemiyle $f_0$ değerlerini kümeleme ve $f_1$ değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.....	34
<b>Çizelge 4.2</b>	: Eşit aralık yöntemiyle $f_0$ değerlerini kümeleme ve $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	34
<b>Çizelge 4.3</b>	: Eşit aralık yöntemiyle $f_0$ değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	35
<b>Çizelge 4.4</b>	: K-ortalamlar yöntemiyle $f_0$ değerlerini kümeleme ve $f_1$ değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.....	39
<b>Çizelge 4.5</b>	: K-ortalamlar yöntemiyle $f_0$ değerlerini kümeleme ve $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	40
<b>Çizelge 4.6</b>	: K-ortalamlar yöntemiyle $f_0$ değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	41
<b>Çizelge 4.7</b>	: Eşit üye sayısı yöntemiyle $f_0$ değerlerini kümeleme ve $f_1$ değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.....	45
<b>Çizelge 4.8</b>	: Eşit üye sayısı yöntemiyle $f_0$ değerlerini kümeleme ve $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	46
<b>Çizelge 4.9</b>	: Eşit üye sayısı yöntemiyle $f_0$ değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	46
<b>Çizelge 4.10</b>	: K-ortalamlar yöntemiyle $F_3$ değerlerini kümeleme ve $f_1$ değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.....	51
<b>Çizelge 4.11</b>	: K-ortalamlar yöntemiyle $f_3$ değerlerini kümeleme ve $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	51
<b>Çizelge 4.12</b>	: K-ortalamlar yöntemiyle $f_3$ değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	52
<b>Çizelge 4.13</b>	: Eşit aralık yöntemiyle $f_3$ -RGB değerlerini ayrı ayrı kümeleme ve $f_1$ değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	56

<b>Çizelge 4.14</b> :	Eşit aralık yöntemiyle $f_3$ -RGB değerlerini ayrı ayrı kümeleme ve $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	57
<b>Çizelge 4.15</b> :	Eşit aralık yöntemiyle $f_3$ -RGB değerlerini ayrı ayrı kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları. ....	57
<b>Çizelge 5.1</b> :	Kümeleme yapılmaksızın YBMG- $f_1$ değerleriyle görüntü erişimi deneyinin başarı oranları. ....	64
<b>Çizelge 5.2</b> :	Kümeleme yapılmaksızın $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi deneyinin başarı oranları. ....	65
<b>Çizelge 5.3</b> :	Kümeleme yapılmaksızın YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi deneyinin başarı oranları. ....	65
<b>Çizelge 5.4</b> :	Kümeleme basamağı bulunmayan çalışmalar ile $f_0$ değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1$ değerleriyle görüntü erişimi). ....	66
<b>Çizelge 5.5</b> :	Kümeleme basamağı bulunmayan çalışmalar ile $f_0$ değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi). ....	66
<b>Çizelge 5.6</b> :	Kümeleme basamağı bulunmayan çalışmalar ile $f_0$ değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması (YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi). ....	66
<b>Çizelge 5.7</b> :	Kümeleme basamağı bulunmayan çalışmalar ile $f_3$ değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1$ değerleriyle görüntü erişimi). ....	66
<b>Çizelge 5.8</b> :	Kümeleme basamağı bulunmayan çalışmalar ile $f_3$ değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi). ....	67
<b>Çizelge 5.9</b> :	Kümeleme basamağı bulunmayan çalışmalar ile $f_3$ değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması (YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi). ....	67
<b>Çizelge 5.10</b> :	YBMG verileriyle kümeleme yapılması durumunda ilgili kümeden görüntü erişimi deneylerinin çalışma süreleri (saniye)...	68
<b>Çizelge 5.11</b> :	Kümeleme yapılmaksızın doğrudan YBMG verileriyle görüntü erişimi deneylerinin çalışma süreleri (saniye). ....	68
<b>Çizelge 5.12</b> :	TBA-I yöntemiyle görüntü erişimi deneyinin başarı oranları. ....	74
<b>Çizelge 5.13</b> :	TBA-II yöntemiyle görüntü erişimi deneyinin başarı oranları. ....	75
<b>Çizelge 5.14</b> :	TBA-III yöntemiyle görüntü erişimi deneyinin başarı oranları. ....	76
<b>Çizelge 5.15</b> :	Kümeleme yapılmaksızın YBMG- $f_1$ değerleriyle görüntü erişimi ve TBA-I birinci temel bileşen değerleriyle görüntü erişimi karşılaştırması. ....	77
<b>Çizelge 5.16</b> :	Kümeleme yapılmaksızın $f_1, f_2$ ve $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi ve TBA-II birinci temel bileşen değerleriyle görüntü erişimi karşılaştırması. ....	77
<b>Çizelge 5.17</b> :	Kümeleme yapılmaksızın YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi ve TBA-III birinci temel bileşen değerleriyle görüntü erişimi karşılaştırması. ....	77

<b>Çizelge 5.18</b> : Çizelge 5.15'te başarı oranları karşılaştırılan deneylerin çalışma süreleri karşılaştırması (saniye).....	78
<b>Çizelge 5.19</b> : Çizelge 5.16'da başarı oranları karşılaştırılan deneylerin çalışma süreleri karşılaştırması (saniye).....	78
<b>Çizelge 5.20</b> : Çizelge 5.17'de başarı oranları karşılaştırılan deneylerin çalışma süreleri karşılaştırması (saniye).....	78





## ŞEKİL LİSTESİ

	<u>Sayfa</u>
<b>Şekil 3.1</b> : YBMG- $f_0$ değerleri kullanılarak eğitim görüntülerinin eşit aralık yöntemiyle kümelenmesi algoritması.....	21
<b>Şekil 3.2</b> : YBMG- $f_3$ değerleri kullanılarak eğitim görüntülerinin k-ortalamlar yöntemiyle kümelenmesi algoritması. ....	23
<b>Şekil 3.3</b> : Küme sayısı seçiminde dirsek metodu grafiği. ....	26
<b>Şekil 4.1</b> : COIL-100 veri tabanındaki bir objenin tüm görüntüleri (0, 5, 10, ..., 355 derece). ....	31
<b>Şekil 4.2</b> : COIL-100 veri tabanında bulunan tüm objelerin ilk görüntüleri (0 derece). ....	31
<b>Şekil 4.3</b> : YBMG- $f_1$ değerleri ile ilgili kümeden görüntü erişimi algoritması.	33
<b>Şekil 4.4</b> : Çizelge 4.1 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	36
<b>Şekil 4.5</b> : Çizelge 4.2 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	37
<b>Şekil 4.6</b> : Çizelge 4.3 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	38
<b>Şekil 4.7</b> : Çizelge 4.4 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	42
<b>Şekil 4.8</b> : Çizelge 4.5 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	43
<b>Şekil 4.9</b> : Çizelge 4.6 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	44
<b>Şekil 4.10</b> : Çizelge 4.7 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	48
<b>Şekil 4.11</b> : Çizelge 4.8 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	49
<b>Şekil 4.12</b> : Çizelge 4.9 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	50

<b>Şekil 4.13</b>	: Çizelge 4.10 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	53
<b>Şekil 4.14</b>	: Çizelge 4.11 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	54
<b>Şekil 4.15</b>	: Çizelge 4.12 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	55
<b>Şekil 4.16</b>	: Çizelge 4.13 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	59
<b>Şekil 4.17</b>	: Çizelge 4.14 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	60
<b>Şekil 4.18</b>	: Çizelge 4.15 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).....	61
<b>Şekil 5.1</b>	: Kümeleme yapılmaksızın YBMG- $f_1$ değerleriyle görüntü erişimi algoritması. ....	64
<b>Şekil 5.2</b>	: Temel Bileşen Analizi'nin (TBA) genel algoritması. ....	71
<b>Şekil 5.3</b>	: TBA-I yöntemiyle görüntü erişimi algoritması. ....	73

# YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ İLE GÖRÜNTÜ KÜMELEME VE GÖRÜNTÜ ERİŞİMİ

## ÖZET

Bu tez çalışmasında, ele alınan bir görüntünün benzerlerinin büyük bir veri tabanından bulunup getirilmesi anlamına gelen içerik tabanlı görüntü erişimi problemi ele alınmış ve bu konuda özgün bir yaklaşım ortaya konmuştur. Önerilen yaklaşım, görüntülerin içeriklerinin Yüksek Boyutlu Model Gösterilimi (YBMG) yöntemiyle analiz edilmesini içermektedir. YBMG yöntemi etkin bir veri bölüntüleme ve boyut indirgeme yöntemidir ve hesaplama karmaşıklığı düşük olduğundan oldukça hızlı çalışmaktadır. Büyük ölçekli veri tabanlarında görüntü erişimi problemlerine uygulandığında da yüksek bir başarı göstermiştir. Bu çalışmada ayrıca, algoritmayı daha da hızlandırmak amacıyla, veri tabanındaki görüntülerin öncelikle kümelenecek benzer görüntülerin bir arada olduğu küçük veri tabanlarının oluşturulması önerilmektedir. Kümelemenin ardından yapılan görüntü erişimi çalışmasında, ele alınan bir sorgu görüntüsü için en yakın küme tespit edilmekte ve bu görüntünün benzerleri yalnızca bu küme içinde aranmaktadır. YBMG yöntemi, hem ilk aşamada veri tabanı görüntüleri kümelendirirken hem de ikinci aşamada görüntü erişimi yapılırken kullanılmaktadır ve bu yöntemle görüntüleri temsil edecek düşük boyutlu veriler hesaplanmaktadır.

YBMG yöntemi, bir ayrıştırım yöntemidir. Verilen çok değişkenli bir fonksiyonun sabit terim, tek değişkenli terimler, iki değişkenli terimler ve bu şekilde giderek artan sayıda değişkene bağlı terimlerin toplamı olarak yeniden ifade edilmesini sağlamaktadır. Bu yöntemle aynı zamanda çok değişkenli veriler de daha az değişkenli bileşenlere ayrıştırılabilmektedir. Bu tez çalışmasında görüntüler, birer çok değişkenli veri olarak ele alınmış ve her birine YBMG ayrıştırımı uygulanmıştır. Görüntü ayrıştırımıyla elde edilen bileşenler sabit, vektör veya matris yapısındaki verilerdir.

Çalışmanın ilk basamağı olan kümeleme aşamasında, öncelikle veri tabanındaki her bir görüntüye YBMG ayrıştırımı uygulanarak düşük boyutlu veriler elde edilmiştir. Bu veriler orijinal çok boyutlu görüntünün yerine kullanılarak kümeleme yapılmıştır. Burada, belirlenen düşük boyutlu iki ayrı veri çeşidinin üç farklı kümeleme yöntemi ile kümelenebilir durumları üzerinde çalışılmıştır. Görüntü erişimi basamağına gelindiğinde ise ilk olarak, bir sorgu görüntüsü ele alınmış ve kümeleme basamağında elde edilen kümeleme ölçütleri kullanılarak bu görüntünün en yakın olduğu küme, yani hedef küme saptanmıştır. Ardından seçilen bir YBMG verisi hem sorgu görüntüsü hem de hedef kümede bulunan görüntüler için hesaplanmış ve bu veri kullanılarak görüntü erişimi yapılmıştır. Bu basamakta tercih edilen düşük boyutlu YBMG verileri, kümeleme basamağında tercih edilen verilere göre daha yüksek boyutludur ve görüntüleri daha iyi temsil etmektedir. Bu veriler esas alınarak hedef kümede bulunan görüntülerden, sorgu görüntüsüne karesel Öklit uzaklığı ölçüsüne göre en yakın olan görüntüler sırayla getirilmiştir. Getirilen görüntülerden belirlenen ilk bir kaç tanesinin, sorgu görüntüsü ile aynı objeye ait olup olmadıkları değerlendirilmiştir.

Değerlendirme çalışmasında, görüntülerin getirilme sıralarını da dikkate alan azalan kümülatif kazanç yöntemi kullanılmıştır. Elde edilen kazanç, ideal durumda elde edilecek olan kazanca oranlanmıştır. Önerilen iki aşamalı bu sistem, COIL-100 görüntü veri tabanı üzerinde yapılan deneysel çalışmalarla sınanmış ve oldukça yüksek başarı oranları elde edilmiştir. Önerilen yöntemlerden bazıları % 100 başarı oranı vermiştir.

Bu özgün yaklaşım, daha iyi bir değerlendirme yapabilmek amacıyla, kümeleme basamağının bulunmadığı çalışmalarla ve YBMG yöntemi yerine Temel Bileşen Analizi yönteminin kullanıldığı başka çalışmalarla karşılaştırılmıştır. Önerilen yaklaşımın başarı oranları ve özellikle de hesaplama süreleri bakımından daha avantajlı olduğu görülmüştür.





# IMAGE CLUSTERING AND IMAGE RETRIEVAL WITH HIGH DIMENSIONAL MODEL REPRESENTATION

## SUMMARY

Content based image retrieval means retrieving similars of a considered image from a large image database by means of mathematical methods and computer systems. This is a significant field of study in image processing because of increasing needs in a wide range of application areas such as commercial advertisements, criminal cases, security systems, artificial intelligence studies, etc. This thesis proposes a novel method in this field which contains analyzing the contents of images by High Dimensional Model Representation (HDMR) method. HDMR is an effective method in data partitioning and dimensionality reduction and it works fast since its computational complexity is considerably low. Therefore it has succeeded in image retrieval problem when it is used to reduce the dimension of images each of which is a high dimensional datum. In addition, to speed up the algorithm even more, the images in database are proposed to be clustered first to obtain small groups in which similar images are together. After clustering step, for a considered query image the closest cluster is determined and the similars of the query image is searched in that cluster. Both in the clustering and the retrieval steps the HDMR method is utilized.

The general HDMR expansion for a given multivariate function is as follows.

$$f(x_1, \dots, x_N) = f_0 + \sum_{i=1}^N f_i(x_i) + \sum_{\substack{i,j=1 \\ i < j}}^N f_{ij}(x_i, x_j) + \dots + f_{12\dots N}(x_1, \dots, x_N) \quad (1)$$

This is a finite sum and its terms are called constant term, univariate terms, bivariate terms respectively and so on. Each term is uniquely determined in the given order from the given multivariate function using formulas for HDMR terms. Due to the structure of HDMR, all the terms are orthogonal to each other and the lower variate terms which are calculated earlier carry higher amount of information of the original function. Thus this method is commonly used to approximate the given function by truncating after some terms.

This method for multivariate functions has also been adapted for multivariate data. Multivariate data is similarly decomposed and then approximated by its terms using formulas for HDMR data decomposition. In this thesis, an RGB image as a three-way array is considered as a trivariate datum and decomposed using HDMR formulas to reduce dimension, in other words, to obtain lower dimensional terms. These terms represent the original image very accurately owing to HDMR's powerful mathematical structure and are calculated very quickly owing to its low computational complexity.

When the method is applied on an RGB image, the constant, univariate and bivariate terms are obtained as constant, vectors and matrices respectively. The formulas for

these terms are given below. The trivariate term which is a three-way array corresponds to margin of error and calculated by subtracting all the previous terms from the original image.

$$\begin{aligned}
f_0 &= \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \sum_{k_3=1}^3 \frac{1}{n_1} \cdot \frac{1}{n_2} \cdot \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) \\
f_1(\xi_1^{(k_1)}) &= \sum_{k_2=1}^{n_2} \sum_{k_3=1}^3 \frac{1}{n_2} \cdot \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_0 \\
f_2(\xi_2^{(k_2)}) &= \sum_{k_1=1}^{n_1} \sum_{k_3=1}^3 \frac{1}{n_1} \cdot \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_0 \\
f_3(\xi_3^{(k_3)}) &= \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \frac{1}{n_1} \cdot \frac{1}{n_2} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_0 \\
f_{12}(\xi_1^{(k_1)}, \xi_2^{(k_2)}) &= \sum_{k_3=1}^3 \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_1(\xi_1^{(k_1)}) - f_2(\xi_2^{(k_2)}) - f_0 \\
f_{13}(\xi_1^{(k_1)}, \xi_3^{(k_3)}) &= \sum_{k_2=1}^{n_2} \frac{1}{n_2} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_1(\xi_1^{(k_1)}) - f_3(\xi_3^{(k_3)}) - f_0 \\
f_{23}(\xi_2^{(k_2)}, \xi_3^{(k_3)}) &= \sum_{k_1=1}^{n_1} \frac{1}{n_1} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_2(\xi_2^{(k_2)}) - f_3(\xi_3^{(k_3)}) - f_0 \quad (2)
\end{aligned}$$

In these formulas  $\xi_i^{(k_i)}$  represents  $k_i$  th domain point along the  $\xi_i$  th direction of the image array (or along the  $\xi_i$  th variable). Therefore  $f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)})$  is the related pixel value of the given image.  $n_i$  represents the number of discrete domain points along  $\xi_i$  th direction. Since the color direction has just three domain points corresponding to R, G and B layers,  $n_3$  equals 3.

In this thesis, a two-step image retrieval system is proposed. In the first step of the proposed methods,  $f_0$  or  $f_3$  term of HDMR is calculated for each image in the database and used for clustering the images. These terms, which are considerably low dimensional, have been selected to be used because this study proposes the clustering step to speed up the retrieval process by coping with the problem of large scale image database. Clustering is executed using three different clustering methods: equal interval size clustering, k-means clustering and equal number of elements clustering. Clustering studies via two different HDMR terms and using three different clustering methods are performed completely separately. In this regard an analysis-based study is followed. The clustering criteria obtained in these clustering studies are kept to be used in the second step.

When it comes to the retrieval step, a query image which does not exist in the database is taken for investigation. The closest cluster for that image is determined using the criteria obtained in the clustering step. Then a chosen HDMR data is calculated for both the query image and the images in the closest cluster to be used in the retrieval process. Using these data the closest images to the query image according to squared Euclidean distance measure are retrieved in order. The HDMR data which are used in this step have higher dimensions than the ones which are used in the clustering step

and thus represent the original image better. One of the representation terms used is  $f_1$  term. The vector obtained by attaching  $f_1$ ,  $f_2$  and  $f_3$  terms end-to-end is used as another tool. Lastly the univariate HDMR approximation which is the superposition of  $f_0$ ,  $f_1$ ,  $f_2$  and  $f_3$  is examined. Studies with these representation data are performed separately.

The retrieved images are evaluated in terms of whether or not they belong to the same object as the query image, taking into account of their order in the list. For this purpose the discounted cumulative gain method is used. This is a method of calculating the gain of each retrieved image by discounting with respect to the order number of that retrieved image in the result list. The gain value is calculated by accumulation throughout the retrieved images. In this study, the obtained gain value is divided by the value which will be obtained in the ideal case to give the evaluation results as success rates.

The proposed methods in this thesis were tested by experiments conducted on COIL-100 image database. This database includes 100 different objects and consists of 72 poses per object. Images of an object are at 5 degrees of rotation on a table around its own axis. For experiments of this thesis, randomly selected one image for each object was discarded from the database. These discarded 100 images were used as query images while the remaining 7100 images were used as the training set. Experiments were conducted on these separated image sets by implementing the algorithms on MATLAB. In these experiments the proposed methods gave significantly high success rates. In some studies, 100% success rate was obtained.

The novel approach proposed in this study was also subjected to comparative analysis. Since one of the novel aspects of this approach is clustering, the proposed methods was firstly compared with methods which omit the clustering step and perform image retrieval directly. With this comparison, it was seen that the proposed methods are better than the methods without clustering in terms of computation times and, in some cases, success rates. A second comparison was made in terms of the preferred dimensionality reduction method. PCA (Principal Component Analysis) method is used instead of HDMR method for this comparison. HDMR gave better success rates and computation times with respect to PCA.

The experiments and comparisons in this thesis have shown that using HDMR method for dimensionality reduction and clustering database images prior to retrieval are notably successful in image retrieval problem.



## 1. GİRİŞ

Günümüzde dijital görüntüler, hemen her alanda önemli bir yer teşkil etmektedir. Teknolojinin ilerlemesiyle görüntüleme cihazları hızla gelişmiş ve yaygınlaşmış, bu da görüntü sayısında devasa bir artış getirmiştir. Aynı zamanda internetin yaygınlaşması ve bilgisayar sistemlerinin kaydettiği ilerlemeler, görüntülerin kullanım amacını genişletmiş ve çeşitliliğini arttırmıştır. Artık görüntüler, ticari faaliyetlerden kriminal vakalara, güvenlik sistemlerinden yapay zeka çalışmalarına kadar hemen her alanda kullanılmaktadır [1]. İnsan yüzü tanıma, objeleri ayırt etme, radyoloji görüntülerinden karakteristik tanımlamalar yapma kullanım amaçlarından yalnızca birkaçıdır [2].

Kullanım alanlarının genişlemesi ve görüntü sayısındaki büyüme, görüntülerin verimli bir şekilde kullanılabilmesinin ve ihtiyaç duyulduğunda aranıp bulunabilmesinin önemini artırmıştır. Bu durum görüntü erişimi adı verilen bir alanın doğmasına neden olmuştur. Bunun yanı sıra, görüntülerin iyileştirilmesi, yeniden düzenlenmesi, boyutlarının değiştirilmesi gibi çalışmaları içeren pek çok görüntü işleme alanı geliştirilmiştir.

Bu tez çalışması, görüntü erişimi alanında özgün bir yöntem sunmayı amaçlamaktadır. Çalışmada, görüntü erişimi öncesinde görüntü kümeleme önerilmekte, böylece görüntü sayısının çok olması sorununa bir çözüm sunulmaktadır. Çalışmanın hem görüntü kümeleme, hem de görüntü erişimi basamaklarında Yüksek Boyutu Model Gösterilimi (YBMG, High Dimensional Model Representation, HDMR) yöntemi kullanılarak boyut indirgeme yapılmakta ve bu yöntem sayesinde hızlı ve etkin bir görüntü erişimi sistemi ortaya konmaktadır.

Çalışmanın bu giriş kısmında görüntü erişimi, kümeleme ve boyut indirgeme konuları ele alınacaktır. İkinci kısımda bu tezin dayanak noktası olan YBMG yönteminin teorisi detaylı olarak anlatılacaktır. Üçüncü kısımda, bu tez çalışmasında kullanılan matematiksel yapılar verilecektir. Örneğin, çalışmada kullanılan kümeleme yöntemleri, küme sayısı seçimi, deney sonuçlarının değerlendirilmesinde kullanılan azalan kümülatif kazanç yöntemi ele alınacaktır. Dördüncü kısım, özgün çalışmanın

detaylı olarak anlatılacağı ve deney sonuçlarının verileceği kısımdır. Beşinci kısımda özgün çalışma diğer çalışmalarla karşılaştırılacaktır. Altıncı kısımda ise sonuç ve öneriler yer alacaktır.

## 1.1 Görüntü Erişimi

Görüntü erişimine dönük çalışmalar temelde, metin tabanlı ve içerik tabanlı olmak üzere iki ayrı yaklaşımla ele alınmaktadır. Ayrıca bu iki yaklaşımın sentezini yapan çalışmalar da mevcuttur. Metin tabanlı görüntü erişimi yaklaşımında, görüntünün kendisi değil görüntüye iliştilirilmiş metinler kullanılarak arama yapılır [3]. Görüntülere metin iliştilirme işlemi ise insan gücüyle yapıldığından zahmetli ve zaman alıcı bir işittir. Ayrıca kişi bağımlı olduğundan objektif değildir. Görüntü ile alakalı olmayan veya rastgele isimler verilebilmekte, sistematik olmayan yöntemler kullanılabilirmekte, görüntülere metin iliştilirilmesi ihmal edilebilmektedir [2]. Veri tabanlarında, aynı görüntüler farklı etiketlerle birden fazla kez yer alabilmektedir. Metinler özenle düzenlense ve tanımlamalar olabildiğince iyi yapılırsa bile, kullanım amacına bağılı olarak görüntünün başka karakteristik özellikleri gerekli olabilmektedir.

İçerik Tabanlı Görüntü Erişimi (İTGE, Content Based Image Retrieval, CBIR) yaklaşımında ise görüntüye iliştilirilen metinler değil görüntü verisinin kendisi kullanılır. Genellikle görüntünün öznitelikleri çıkarılarak, bunları esas alan bir arama yapılır [4]. Şekil, renk ve doku kullanılan özniteliklerden bazılarıdır [5]. Bu yaklaşımı kullanan sistemlerde, görüntü erişimi bir görüntü seçilmesi ile başlar. Seçilen görüntünün ve veri tabanındaki diğer görüntülerin içerikleri analiz edilerek, seçilen görüntüye benzer başka görüntülere ulaşılmaya çalışılır [2]. İTGE sistemleri genellikle, öznitelik bulma, benzerlik ölçümü ve ilişki geri bildirim basamaklarını içerir [6, 7].

İçerik tabanlı görüntü erişimi, metin tabanlı yaklaşımın aksine görüntünün kendisini kullandığından kullanıcı bağımlı değildir ve objektiftir. Bu yüzden son yıllarda bu yaklaşımla yapılan çalışmalar ağırlık kazanmıştır. Bununla beraber, metin tabanlı ve içerik tabanlı yaklaşımların birleştirilmesiyle geliştirilen farklı yaklaşımlar da vardır [8]. Bunlardan biri olan sematik tabanlı görüntü erişimi yaklaşımında, veritabanındaki görüntüler önceden içerik tabanlı olarak analiz edilir, elde edilen metinsel bilgiler veritabanına kaydedilir ve görüntü erişimi için metin tabanlı bir arama yapılır [4].

Bu tez çalışmasında içerik tabanlı yaklaşım benimsenmiştir. Diğer bir deyişle, görüntülerin yalın olarak ele alındığı, görüntüye iliştilirilmiş anahtar kelimelerin kullanılmadığı bir sistem önerilmiştir. Ayrıca önerilen sistem, görüntülerin şekil, renk, doku gibi öznitelikleri çıkarılmaksızın, görüntü verisinin boyut indirgeme yöntemiyle doğrudan işlenmesini içermektedir. Benzerlik ölçümü ise, görüntülerden elde edilen düşük boyutlu verilerin, seçilen görüntüye olan uzaklıklarının karşılaştırılmasıyla yapılmıştır. Oluşturulan bu sistem dördüncü kısımda ayrıntılı olarak anlatılacaktır.

## 1.2 Kümeleme

Veri kümeleme, verileri kendi içinde olabildiğince benzer ve kendi aralarında olabildiğince farklı gruplara ayırma işlemidir. Bu gruplandırma işleminde, önceden belirlenmiş ortak özellikler esas alınmaz ve önceden belirlenmiş gruplar da bulunmaz. Belli bir uzaklık kıstasına göre birbirine yakın olan verilerin aynı kümeye atanmasını hedefleyen bir sistem oluşturulur. Bu yönüyle kümeleme, bir güdümsüz öğrenme yöntemidir [9]. Geliştirilmiş pek çok kümeleme yöntemi ve ara yöntemler vardır. K-ortalamlar (k-means) ve bulanık c-ortalamlar (fuzzy c-means) en bilinen kümeleme yöntemlerindedir [10]. Kümelemeden tamamen farklı olan sınıflandırma işlemi ise, verilerin önceden belirlenmiş olan sınıflara atanması işlemidir ve bir güdümlü öğrenmedir.

Kümeleme, günümüzde büyük verilerden anlamlı bilgiler elde etmek için yaygın olarak kullanılan bir yaklaşımdır. Örnekte tanıma, bilgi getirme, veri madenciliği, veri sıkıştırma gibi pek çok alanda kullanılmaktadır. Bu yaklaşım benzer verilerin bir arada olduğu küçük gruplar oluşturduğundan, veri tabanı ölçeğinin küçültülmesinde de kullanılmaktadır [11]. Bu tez çalışmasında kümeleme, büyük ölçekli bir veri tabanından küçük veri tabanları oluşturma ve böylece çalışmalarını kolaylaştırma amacıyla kullanılmıştır. Veri tabanındaki görüntülerin öncelikle kümelenecek, görüntü erişiminin yalnızca ilgili küme içerisinden yapılması önerilmiştir. Bununla algoritma karmaşıklığının düşürülmesi hedeflenmiştir. Çalışmalarda kullanılan kümeleme yöntemleri ve küme sayısı seçimi üçüncü bölümde verilecektir.

### 1.3 Boyut İndirgeme

İçerik tabanlı görüntü erişiminde görüntü verisinin ve görüntüden çıkarılan özniteliklerin yüksek boyutlu olması önemli bir sorundur. Buna boyutun laneti (curse of dimensionality) adı verilmektedir. Örneğin  $256 \times 256$  çözünürlüğünde gri tonlamalı bir görüntü, satırların uç uca eklenmesi sonucu, 65536 boyutunda bir vektörle çalışılmasını gerektirmektedir [12]. Bu durum, arama yapılan veri tabanının büyüklüğü, arama algoritmasının karmaşıklığı gibi etkenlerle bir araya geldiğinde, algoritma performansını daha da kötüleştirilmektedir. Bu yüzden, bir boyut indirgeme yöntemi kullanılarak boyutun azaltılması gerekmektedir.

Boyut indirgeme (dimensionality reduction), çok boyutlu bir verinin, daha iyi anlaşılabilmesi veya daha etkili kullanılabilmesi için, veri içeriğinden elde edilen düşük boyutlu verilerle temsil edilmesidir [12]. Boyut indirgeme işlemi güdümlü (supervised) veya güdümsüz (unsupervised) öğrenme yöntemleri ile yapılmaktadır [13]. Etiketlenmemiş yalın veriler üzerinde çalışılan güdümsüz yöntemlerin başında Temel Bileşen Analizi (TBA, Principal Component Analysis, PCA) gelmektedir [14]. Bunun yanı sıra KPCA, MDS, ISOMAP, LLE, LPP gibi güdümsüz yöntemler de geliştirilmiştir [7]. Güdümlü boyut indirgeme yöntemleri ise, başlıca FLDA ve LFDA'dır [7]. Güdümlü boyut indirgeme işleminde, ele alınan verilerin sınıf etiketi kullanılır ve aynı sınıftaki veriler birlikte analiz edilerek boyut indirgenir.

Yaygın olarak kullanılan bu boyut indirgeme yöntemlerinin yanı sıra, son yıllarda yapılan çalışmalarda YBMG yönteminin de bir boyut indirgeme yöntemi olarak ele alındığı görülmektedir [15]. Yüksek Boyutlu Model Gösterilimi yöntemi, çok değişkenli bir fonksiyonun daha az değişkenli fonksiyonların toplamı olarak ifade edilmesini sağlayan bir böl-yönet algoritmasıdır. İlk olarak, 1990 yılında Rus matematikçi I. M. Sobol tarafından temelleri atılmıştır [16]. 2000'lerin başında Princeton Üniversitesi'nden Prof. Herschel Rabitz tarafından geliştirilmiştir [17–19]. Son yıllarda Prof. Metin Demiralp ve grubu tarafından daha da geliştirilerek çeşitlendirilmiş ve farklı problemlere uygulanmıştır [20–24]. Bugün dünyada, farklı bilim insanları tarafından farklı problemler için geliştirilmeye ve uygulanmaya devam etmektedir [25,26].



YBMG yöntemi, ele alınan çok değişkenli bir fonksiyonu, sabit fonksiyon, değişkenlerden her birine ayrı ayrı bağlı olan tek değişkenli fonksiyonlar, değişkenlerin ikili kombinasyonlarına bağlı olan iki değişkenli fonksiyonlar, böylece giderek artan sayıda değişkene bağlı fonksiyonların toplamı şeklinde yazmayı sağlar. Eşsiz olarak elde edilen bu az değişkenli fonksiyonlar toplamı, verilen orijinal fonksiyona eşittir. YBMG yöntemiyle yalnızca analitik olarak ifade edilebilen fonksiyonlar değil, çok boyutlu veriler de ayrıştırılabilmektedir. Veri ayrıştırımında da, fonksiyon ayrıştırımına benzer şekilde, ele alınan çok değişkenli verinin daha az değişkenli verilerin toplamı olarak ifade edilmesi amaçlanır.

Elde edilen az değişkenli fonksiyonlardan (veya az değişkenli verilerden) daha az değişkene bağlı olanların toplanması ve diğerlerinin ihmal edilmesi ile orijinal fonksiyona bir yaklaştırım da yapılabilir. Yapılan çalışmalar, bu yaklaştırım yönteminin oldukça başarılı sonuçlar verdiğini göstermektedir [20, 21, 27–29]. Bu başarı, YBMG ayrıştırımında orijinal fonksiyonun yoğunluklu olarak daha az değişkenli fonksiyonlarla temsil edilmesinden ve görece daha çok değişkenli fonksiyonların ise yüksek frekanslı bilgileri taşımasından kaynaklanmaktadır.

Yaklaştırım yapmak için kullanılan ve istenen başarıyı yakalayan YBMG'den başka pek çok matematiksel yapı bulunmaktadır. Bu noktada YBMG yönteminin katkısı, aynı orandaki bir yaklaştırımı daha hızlı ve daha düşük maliyetli olarak yapmasıdır [15].

YBMG yöntemi uygun bir yaklaşımla, çok değişkenliliğin söz konusu olduğu hemen her alanda, problemleri yönetilebilir hale getirmek için kullanılabilir. Örneğin, çok değişkenli bir verinin interpolasyonu oldukça zahmetlidir. Bunun yerine veri YBMG yöntemiyle ayrıştırılıp, elde edilen az değişkenli verilerle interpolasyon yapılabilir. Böyle bir çalışmada, altı değişkenli bir veri en fazla iki değişkenli terimleri kullanılarak interpolate edilmiş ve ihmal edilebilir hata oranları elde edilmiştir [27].

YBMG yöntemini görüntüler üzerinde kullanmak da mümkündür [24, 29–31]. Bir çalışmada RGB formatındaki dijital görüntüler, ayrıştırıma tabi tutulup yeniden birleştirildiğinde orijinal görüntüye oldukça yakın görüntülerin elde edildiği gösterilmiştir [29]. Başka bir çalışmada görüntünün ayrıştırılması ve elde edilen verilerin yaklaştırıma tekrarlı olarak eklenmesiyle, görüntü kontrastında

artış sağlanmıştır [30]. Ayrıca YBMG yöntemi, görüntülerdeki eksik kısımların tamamlanması ve bozuk kısımların düzeltilmesi için içboyama amacıyla da kullanılmış ve başarılı sonuçlar elde edilmiştir [31,32].

YBMG veri ayrıştırımı (data decomposition), temelde bir veri bölüntüleme (data partitioning) yöntemidir. Bununla beraber, bu ayrıştırımla elde edilen az değişkenli veriler az sayıda değer içerdiğinden veri boyutunun indirgenmesi için de kullanılabilir. Bu tez çalışmasında da görüntülerin boyutu bu yöntemle indirgenmiştir. Çalışmada, YBMG yöntemiyle boyut indirgeme, güdümsüz öğrenme yaklaşımıyla ele alınmıştır. Yani veri tabanındaki görüntüler, içeriklerine dair etiketler taşımayan, etiketsiz tek bir grup olarak kabul edilmiş ve sadece görüntü verileri analiz edilerek tanınmıştır. Görüntülerden YBMG ayrıştırımıyla elde edilen düşük boyutlu veriler, hem görüntü kümelemede, hem de görüntü erişiminde kullanılmıştır.

#### **1.4 Özgün Çalışma**

Bu tez çalışmasında YBMG yöntemi kullanılarak görüntü erişimi yapılmaktadır. Çalışmanın özgün yanı, görüntü getirme işlemi öncesinde veri tabanındaki görüntülerin kümelenebilmesi, böylece veri tabanı ölçeği düşürülerek zaman kazancı sağlanmasıdır. Daha önce yapılan bir çalışmada [15] da YBMG ayrıştırımı görüntü erişimi probleminde kullanılmış, fakat bahsedilen çalışmada kümelemeye gidilmeksizin doğrudan görüntü erişimi yapılmıştır. Ayrıca görüntü erişimi işleminde benzerlik ölçümü vektörler arasındaki açı ölçümüyle yapılmıştır. Bunların yanı sıra, YBMG ayrıştırımından elde edilen sabit terim ile renk değişkenine bağlı tek değişkenli terim, çok düşük boyutlu olduklarından görüntü erişiminde başarısız bulunmuştur. Bu tez çalışmasında ise, görüntü erişimi öncesinde görüntü kümeleme yapılarak zaman kazancı sağlanmıştır, benzerlik ölçümü karesel Öklit uzaklığı ile yapılarak daha başarılı sonuçlar elde edilmiştir ve sabit terim ile renk değişkenine bağlı olan terim görüntü kümeleme amacıyla kullanılarak bu iki terimin de görüntüleri başarılı şekilde temsil ettiği gösterilmiştir. Ayrıca, kümeleme işleminin zaman kazancının yanı sıra başarı artışı da sağladığı gösterilmiştir. Bu tezde önerilen sistemi benimseyen ilk çalışma, bir konferans bildirisi [28] olarak sunulmuştur. Bu tez çalışmasında ise önerilen yöntem çeşitlendirilerek geliştirilmiş ve başarı oranlarında ciddi bir artış kaydedilmiştir.

Bu çalışmadaki algoritmalarda genel olarak şöyle bir yöntem benimsenmiştir: Üzerinde çalışılan COIL-100 görüntü veri tabanından [33] bazı görüntüler test amacıyla ayrılmış, geri kalan görüntüler eğitim görüntüleri olarak kullanılmıştır. Her bir eğitim görüntüsünün YBMG bileşenlerinden biri hesaplanmış ve düşük boyutlu bu verilerle kümeleme yapılmıştır. Test işlemine geçildiğinde ise, test görüntülerinden biri sorgu görüntüsü olarak ele alınmış ve ilk olarak bu görüntünün en yakın olduğu küme tespit edilmiştir. Bu hedef kümede bulunan görüntülerin, kümeleme aşamasında kullanılanlara göre daha fazla veri içeren başka bir YBMG bileşeni hesaplanmış, sorgu görüntüsünün de aynı bileşeni hesaplanarak aralarındaki uzaklık ölçülmüştür. En yakın veriler tespit edilip, karşılık gelen en yakın görüntüler getirilmiştir. Getirilen görüntülerin sorgu görüntüsündeki objeyle aynı objeye ait olup olmadığı değerlendirilmiştir. Çalışmanın ayrıntılı yapısı ve elde edilen başarı oranları dördüncü kısmında yer almaktadır.

Önerilen sistemin başarısı, karşılaştırmalı olarak da değerlendirilmiştir. İlk olarak özgün çalışma, kümelemenin söz konusu olmadığı çalışmalarla karşılaştırılmıştır. Bu karşılaştırma, kümeleme işlemi sayesinde hesaplama zamanından sağlanan kazancı ortaya koymaktadır. İkinci olarak ise, YBMG yönteminin görüntü erişimindeki başarısını ortaya koymak amacıyla, bu yöntem Temel Bileşen Analizi yöntemiyle karşılaştırılmıştır. Bu karşılaştırma ise, hesaplama zamanı ve başarı oranları açısından YBMG yönteminin daha avantajlı olduğu sonucunu vermiştir. Karşılaştırma çalışması beşinci kısımda anlatılmaktadır.

Önerilen yöntemlerin test edildiği deneyler, MATLAB [34] üzerinde programlanan algoritmalarla yapılmıştır. Deneylerde kullanılan bilgisayar, Intel(R) Core(TM) i5-6200U 2.30 GHz - 2.40 GHz işlemci ile 3.82 GB kullanılabilir bellek içermektedir ve 64-bit tabanlı işlemci ve işletim sistemi ile çalışmaktadır.



## 2. YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ

Bu kısımda, çalışmada önemli bir yer teşkil eden Yüksek Boyutlu Model Gösterilimi (YBMG, High Dimensional Model Representation, HDMR) yönteminin teorisi anlatılacak ve her bir terimi matematiksel olarak elde edilecektir. Öncelikle, analitik olarak ifade edilebilen fonksiyonlar için YBMG açılımı verilecektir. İkinci bölümde ise YBMG yönteminin, analitik olarak ifade edilemeyen fakat tanım kümesinin ayrık noktalarında aldığı değerler bilinen çok değişkenli fonksiyonlar, yani veriler üzerine uyarlanması anlatılacaktır.

### 2.1 Yüksek Boyutlu Model Gösterilimi (YBMG) ile Fonksiyon Ayrıştırımı

Verilen  $N$  değişkene bağlı bir  $f(x_1, \dots, x_N)$  fonksiyonu için genel YBMG açılımı aşağıda verildiği gibidir.

$$f(x_1, \dots, x_N) = f_0 + \sum_{i=1}^N f_i(x_i) + \sum_{\substack{i,j=1 \\ i < j}}^N f_{ij}(x_i, x_j) + \dots + f_{12\dots N}(x_1, \dots, x_N) \quad (2.1)$$

Bu açılım sonlu bir toplamdır. Açılımın ilk terimi olan  $f_0$ , bir sabit fonksiyondur.  $f_i(x_i)$  terimleri, tek değişkenli fonksiyonlardır ve  $\binom{N}{1}$  tanedir.  $f_{ij}(x_i, x_j)$  iki değişkenli fonksiyonları ise  $\binom{N}{2}$  tanedir. Benzer şekilde daha çok değişkenli fonksiyonlar da  $\binom{N}{r}$  kombinasyon katsayıları kadardır. Dolayısıyla toplam terim sayısı  $2^N$ 'dir. Son terim olan  $f_{12\dots N}(x_1, \dots, x_N)$  fonksiyonu, orijinal fonksiyon  $f(x_1, \dots, x_N)$  gibi tüm değişkenlere birden bağlı olan bir fonksiyondur. Verilen açılımda yer alan bu fonksiyonlara YBMG terimleri adı verilir.

YBMG terimlerini eşsiz olarak belirlemek için Sobol tarafından ortaya atılan ve aşağıda verilen yok etme koşulları kullanılmaktadır.

$$\int_{a_1}^{b_1} dx_1 \dots \int_{a_N}^{b_N} dx_N W_1(x_1) \cdots W_N(x_N) f_i(x_i) = 0, \quad i = 1, \dots, N \quad (2.2)$$

Yok etme koşulunda yer alan  $W_i(x_i)$  ifadeleri, değişkenlerden her birine bağlı olarak ayrı ayrı tanımlanan ağırlık fonksiyonlarıdır. Yok etme koşulları uygun bir iç çarpım tanımını altında, Prof. Demiralp tarafından kanıtlanan aşağıdaki diklik koşullarına karşılık gelmektedir.

$$(f_{i_1 i_2 \dots i_k}, f_{i_1 i_2 \dots i_l}) = 0, \quad i_1 i_2 \dots i_k \neq i_1 i_2 \dots i_l, \quad k, l = 1, \dots, N \quad (2.3)$$

$i_1 i_2 \dots i_k \neq i_1 i_2 \dots i_l$  şartından anlaşıldığı gibi diklik koşulu, genel YBMG açılımında yer alan herhangi iki terimin birbirine dik olması koşuludur. Diğer bir deyişle, YBMG açılımını bir dik ayrıştırımdır (orthogonal decomposition).

Görüldüğü gibi, diklik koşulu bir iç çarpım üzerinde tanımlanmaktadır. Öncelikle, YBMG açılımında orijinal fonksiyonun ve YBMG terimlerinin karesi integrallenebilir fonksiyonlar olduğu kabul edilmektedir. Ayrıca, kare integraller ve iç çarpım, ağırlık fonksiyonu altında, her bir bağımsız değişkenin belli bir aralığında tanımlanır ( $x_i \in [a_i, b_i]$ ). Bu şartlar altında bahsedilen iç çarpım, herhangi iki karesi integrallenebilir  $u(x_1, \dots, x_N)$  ve  $v(x_1, \dots, x_N)$  fonksiyonları için,

$$(u, v) = \int_{a_1}^{b_1} dx_1 W_1(x_1) \dots \int_{a_N}^{b_N} dx_N W_N(x_N) u(x_1, \dots, x_N) v(x_1, \dots, x_N) \quad (2.4)$$

şeklinde tanımlanır.

Ayrıca, YBMG terimlerinin belirlenmesini kolaylaştırmak amacıyla, her bir ağırlık fonksiyonu, bağlı olduğu değişkene göre ilgili aralıktaki integrali 1 olacak şekilde seçilmelidir.

$$\int_{a_i}^{b_i} dx_i W_i(x_i) = 1, \quad i = 1, \dots, N \quad (2.5)$$

Buna normalizasyon kriteri adı verilir. Buraya kadar anlatılmış olan ağırlık fonksiyonları ve diklik koşulu kullanılarak YBMG terimleri sırayla çıkarılabilir.

$f_0$  sabit terimini hesaplamak için, genel YBMG eşitliğinin (Denklem 2.1) her iki tarafı  $W_1(x_1) \dots W_N(x_N)$  ile çarpılır ve tüm değişkenlere göre ilgili aralıklarda integrallenir. Bu işlem sonucunda, eşitliğin sağ tarafında yer alan  $f_0$  hariç tüm terimler yok etme koşulundan dolayı yok olacaktır. Bunun sonucu olarak, eşitliğin solunda elde edilen ifade,  $f_0$  terimi olarak belirlenir:

$$f_0 = \int_{a_1}^{b_1} dx_1 \dots \int_{a_N}^{b_N} dx_N W_1(x_1) \cdots W_N(x_N) f(x_1, \dots, x_N) \quad (2.6)$$

Tek deęişkenli terimlerin yapısı da benzer şekilde elde edilir. Burada, herhangi bir  $f_i(x_i)$  fonksiyonunu elde etmek için, genel YBMG eşitliğinin her iki tarafı ağırlık fonksiyonları ile çarpılırken ve integralenirken, ilgili  $x_i$  deęişkenine baęlı ağırlık fonksiyonu ve integralleme işlemi dışlanır. Bu durumda, eşitliğin saę tarafında  $f_0$  fonksiyonu ve ilgili  $f_i(x_i)$  fonksiyonu kalacak, dięer tek deęişkenli terimler ile daha çok deęişkenli terimler yok olacaktır. Buradan da,  $f_0$  fonksiyonu eşitliğin dięer tarafına geçirilerek, ilgili  $f_i(x_i)$  terimi elde edilir. Bu durumda tek deęişkenli terimler kuralı aşıęıdaki gibidir.

$$f_i(x_i) = \int_{a_1}^{b_1} dx_1 \dots \int_{a_{i-1}}^{b_{i-1}} dx_{i-1} \int_{a_{i+1}}^{b_{i+1}} dx_{i+1} \dots \int_{a_N}^{b_N} dx_N W_1(x_1) \cdots W_{i-1}(x_{i-1}) \cdot W_{i+1}(x_{i+1}) \cdots W_N(x_N) f(x_1, \dots, x_N) - f_0, \quad i = 1, \dots, N \quad (2.7)$$

İki deęişkenli terimlerin yapısını elde etmek için de benzer bir yol izlenir. Ele alınan bir  $f_{ij}(x_i, x_j)$  iki deęişkenli terimini elde etmek için, genel YBMG eşitliğinin her iki tarafı,  $W(x_i)$  ve  $W(x_j)$  hariç dięer ağırlık fonksiyonları ile çarpılır ve  $x_i$  ve  $x_j$  hariç dięer deęişkenlere göre ilgili aralıklarda integralenir. Bu işlem sonucunda eşitliğin saę tarafında sabit fonksiyon, tek deęişkenli fonksiyonlardan ilgili  $f_i(x_i)$  ile  $f_j(x_j)$  fonksiyonları ve iki deęişkenli fonksiyonlardan sadece ilgili  $f_{ij}(x_i, x_j)$  fonksiyonu kalacaktır. Dięer terimler yok etme koşulundan dolayı yok olacaktır. Elde edilen eşitlikten  $f_{ij}(x_i, x_j)$  fonksiyonu çekilerek, bu fonksiyonun yapısı elde edilir. Bu durumda iki deęişkenli terimlerin genel yapısı aşıęıdaki gibidir.

$$f_{ij}(x_i, x_j) = \int_{a_1}^{b_1} dx_1 \dots \int_{a_{i-1}}^{b_{i-1}} dx_{i-1} \int_{a_{i+1}}^{b_{i+1}} dx_{i+1} \dots \int_{a_{j-1}}^{b_{j-1}} dx_{j-1} \int_{a_{j+1}}^{b_{j+1}} dx_{j+1} \dots \int_{a_N}^{b_N} dx_N \cdot W_1(x_1) \cdots W_{i-1}(x_{i-1}) W_{i+1}(x_{i+1}) \cdots W_{j-1}(x_{j-1}) W_{j+1}(x_{j+1}) \cdots W_N(x_N) f(x_1, \dots, x_N) - f_i(x_i) - f_j(x_j) - f_0, \quad i, j = 1, \dots, N \quad (2.8)$$

Üç, dört ve daha çok deęişkenli terimlerin yapısı da aynı mantık kullanılarak elde edilebilir. En sonda yer alan  $f_{12\dots N}(x_1, \dots, x_N)$  fonksiyonu, orijinal fonksiyon  $f(x_1, \dots, x_N)$  gibi tüm deęişkenlere birden baęlıdır. Bu son terimi elde edebilmek için, ilgili ağırlık fonksiyonları ve ilgili integraller dışlandıęında geriye hiç bir ağırlık

fonksiyonu ve hiç bir integral işlemi kalmamaktadır. Bu durumda, bu son fonksiyonun yapısını elde etmek için, yalnızca genel YBMG eşitliğinden bu fonksiyonun kendisi çekilmelidir. Yani bu son terim, diğer tüm terimler orijinal fonksiyondan çıkarılarak elde edilir.

Analitik olarak verilen bir  $f(x_1, \dots, x_N)$  fonksiyonunun YBMG dik ayrıştırım terimleri anlatılan yöntemlerle elde edilir. Bu sonlu terimler toplamı orijinal fonksiyona eşittir. Elde edilen az değişkenli fonksiyonlar, çok değişkenli orijinal fonksiyonun yapısının daha kolay anlaşılmasını sağlar. Bu yüzden pek çok farklı amaçla kullanılabilir. Bunun yanı sıra, elde edilen az değişkenli fonksiyonlar ile orijinal fonksiyona bir yaklaştırım yapılabilir. Eğer genel YBMG açılımı sabit fonksiyondan sonra kesilirse elde edilen yaklaştırıma sabit yaklaştırım adı verilir. Benzer şekilde, tek değişkenli fonksiyonlardan sonra kesilerek tek değişkenli yaklaştırım ve iki değişkenli fonksiyonlardan sonra kesilerek iki değişkenli yaklaştırım elde edilir. Bu YBMG yaklaştırım bağıntıları aşağıdaki gibi yazılabilir.

$$\begin{aligned}
s_0(x_1, \dots, x_N) &= f_0 \\
s_1(x_1, \dots, x_N) &= f_0 + \sum_{i=1}^N f_i(x_i) \\
s_2(x_1, \dots, x_N) &= f_0 + \sum_{i=1}^N f_i(x_i) + \sum_{\substack{i,j=1 \\ i < j}}^N f_{ij}(x_i, x_j) \\
&\vdots \\
s_k(x_1, \dots, x_N) &= s_{k-1}(x_1, \dots, x_N) + \sum_{\substack{i_1, \dots, i_k=1 \\ i_1 < \dots < i_k}}^N f_{i_1 \dots i_k}(x_{i_1}, \dots, x_{i_k}) \\
1 \leq k \leq N & \tag{2.9}
\end{aligned}$$

Bu bağıntılar kullanılarak, orijinal fonksiyona

$$f(x_1, \dots, x_N) \approx s_k(x_1, \dots, x_N) \tag{2.10}$$

şeklinde yaklaştırım yapılabilir.



## 2.2 Yüksek Boyutlu Model Gösterilimi ile Veri Ayrıştırımı

YBMG yöntemi ile analitik olarak verilen fonksiyonlar ayrıştırılabildiği gibi, analitik yapısı bilinmeyen bunun yerine  $x_1, \dots, x_N$  bağımsız değişkenlerinin ifade ettiği Öklit uzayının ayrık sonlu sayıda noktasında aldığı değerler bilinen fonksiyonlar, yani veriler de ayrıştırılabilir. Bu amaçla, her bir  $x_i$  değişkeni için, bir ayrık küme  $\mathcal{D}_i$  tanımlanır.

$$\mathcal{D}_i = \left\{ \xi_i^{(1)}, \dots, \xi_i^{(k_i)}, \dots, \xi_i^{(n_i)} \right\}, \quad i = 1, \dots, N \quad (2.11)$$

$x_1, \dots, x_N$  değişkenleri için tanımlanan bu  $\mathcal{D}_1, \dots, \mathcal{D}_N$  kümelerinin kartezyen çarpımı, orijinal verinin tanım kümesini verir. Yani, analitik yapısı verilmeyen bir fonksiyonun, üzerinde tanımlandığı sonlu sayıda ayrık nokta oluşturulmuş olur. Bu tanım kümesi

$$\mathcal{D} = \mathcal{D}_1 \times \mathcal{D}_2 \times \dots \times \mathcal{D}_N \quad (2.12)$$

veya daha açık olarak

$$\mathcal{D} = \{ \tau \mid \tau = (x_1, x_2, \dots, x_N), x_i \in \mathcal{D}_i, i = 1, 2, \dots, N \} \quad (2.13)$$

şeklinde ifade edilebilir. Oluşturulan bu kartezyen çarpım  $N$ 'lileri içerir ve  $f(x_1, \dots, x_N)$  fonksiyonunun, yani verinin tanım kümesidir. Veriyi oluşturan değerler, bu tanım kümesi üzerinde tanımlanmış bir fonksiyonun aldığı değerlerdir.

Ayrık noktalarda tanımlanan bir fonksiyon ile çalışıldığından, ağırlık fonksiyonlarının da bu yapıya uyarlanması gerekmektedir. Bunun için ağırlık fonksiyonları, Dirac delta fonksiyonlarının bir lineer kombinasyonu olarak tanımlanabilir:

$$W_i(x_i) = \sum_{k_i=1}^{n_i} \alpha_{k_i}^{(i)} \delta(x_i - \xi_i^{(k_i)}), \quad x_i \in [a_i, b_i], i = 1, \dots, N \quad (2.14)$$

Bu ağırlık fonksiyonlarının çarpımı, Dirac delta fonksiyonunun özelliklerinden dolayı, aşağıdaki gibi hesaplanır.

$$\prod_{i=1}^N \left( \sum_{k_i=1}^{n_i} \alpha_{k_i}^{(i)} \delta(x_i - \xi_i^{(k_i)}) \right) = \prod_{i=1}^N \alpha_{k_i}^{(i)}, \quad i = 1, \dots, N \quad (2.15)$$

Her bir ağırlık fonksiyonunun ilgili değişkene göre ilgili aralıkta integrali alındığında, yine Dirac delta fonksiyonunun özelliklerinden dolayı, aşağıdaki sonuca ulaşılır.

$$\int_{a_i}^{b_i} dx_i \sum_{k_i=1}^{n_i} \alpha_{k_i}^{(i)} \delta(x_i - \xi_i^{(k_i)}) = \sum_{k_i=1}^{n_i} \alpha_{k_i}^{(i)}, \quad i = 1, \dots, N \quad (2.16)$$

Normalizasyon kriterine göre, her bir ağırlık fonksiyonunun, bağlı olduğu değişkene göre ilgili aralıktaki integrali 1 olması gerektiğinden, yukarıdaki denklemde elde edilen ifade 1 olmalıdır.

$$\sum_{k_i=1}^{n_i} \alpha_{k_i}^{(i)} = 1, \quad i = 1, \dots, N \quad (2.17)$$

Veriler için uyarlanan bu ağırlık fonksiyonları kullanılarak YBMG veri ayrıştırımının terimleri belirlenebilir.

Yapısı Denklem 2.6'da verilen  $f_0$  fonksiyonu için, denklemde bulunan ağırlık fonksiyonları, veriler için uyarlanan ağırlık fonksiyonları ile değiştirildiğinde ve integral hesapları yapıldığında, sabit fonksiyonun yapısı

$$f_0 = \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \dots \sum_{k_N=1}^{n_N} \left( \prod_{i=1}^N \alpha_{k_i}^{(i)} \right) f(\xi_1^{(k_1)}, \dots, \xi_N^{(k_N)}) \quad (2.18)$$

şeklinde bulunur. Elde edilen  $f_0$  bir sabit sayıdır.

Benzer şekilde,  $f_i(x_i)$  tek değişkenli fonksiyonlarının kuralında (Denklem 2.7) bulunan ağırlık fonksiyonları da değiştirildiğinde ve integral hesapları yapıldığında,

$$f_i(\xi_i^{(k_i)}) = \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \dots \sum_{k_{i-1}=1}^{n_{i-1}} \sum_{k_{i+1}=1}^{n_{i+1}} \dots \sum_{k_N=1}^{n_N} \left( \prod_{\substack{m=1 \\ m \neq i}}^N \alpha_{k_m}^{(m)} \right) f(\xi_1^{(k_1)}, \dots, \xi_N^{(k_N)}) - f_0$$

$$; \xi_i^{(k_i)} \in \mathcal{D}_i; k_i = 1, \dots, n_i; i = 1, \dots, N \quad (2.19)$$

yapısı elde edilir. Veriler için elde edilen tek değişkenli terimlerin her biri bir vektör şeklindedir.  $f_i(x_i)$  vektörü, yani  $x_i$  değişkenine bağlı tek değişkenli fonksiyon,  $n_i$  tane değer içerir. Bu şekilde  $\binom{N}{1}$  tane vektör vardır.

İki deęişkenli terimlerin kuralını belirlemek için de, Denklem 2.8'de bulunan aęırlık fonksiyonları uyarlanan aęırlık fonksiyonlarıyla deęiştirilir ve gerekli integral hesapları yapılır. Bu işleme, veriler için olan iki deęişkenli terimlerin yapısı

$$f_{ij}(\xi_i^{(k_i)}, \xi_j^{(k_j)}) = \sum_{k_1=1}^{n_1} \dots \sum_{k_{i-1}=1}^{n_{i-1}} \sum_{k_{i+1}=1}^{n_{i+1}} \dots \sum_{k_{j-1}=1}^{n_{j-1}} \sum_{k_{j+1}=1}^{n_{j+1}} \dots \sum_{k_N=1}^{n_N} \left( \prod_{\substack{m=1 \\ m \neq i \\ m \neq j}}^N \alpha_{k_m}^{(m)} \right) f \left( \xi_1^{(k_1)}, \dots, \xi_N^{(k_N)} \right) - f_i \left( \xi_i^{(k_i)} \right) - f_j \left( \xi_j^{(k_j)} \right) - f_0$$

$$; \xi_i^{(k_i)} \in \mathcal{D}_i ; \xi_j^{(k_j)} \in \mathcal{D}_j ; k_i = 1, \dots, n_i ; k_j = 1, \dots, n_j ; i, j = 1, \dots, N \quad (2.20)$$

şeklinde bulunur. Buradan elde edilen iki deęişkenli terimlerin her biri bir matris şeklindedir.  $f_{ij}(x_i, x_j)$  matrisi, yani  $x_i$  ve  $x_j$  deęişkenlerine baęlı iki deęişkenli fonksiyon,  $n_i \cdot n_j$  tane deęer içerir. Dięer bir deyişle  $n_i \times n_j$  boyutunda bir matristir. Bu şekilde  $\binom{N}{2}$  tane matris vardır.

İkiden fazla deęişkenli terimler de benzer şekilde üretilebilir. Veri ayrıştırımında sabit fonksiyon sabit bir deęere, tek deęişkenli fonksiyonlar vektörlere, iki deęişkenli fonksiyonlar matrislere karşılık geldięi gibi, üç deęişkenli fonksiyonlar da üç boyutlu dizilere karşılık gelmektedir. Bu şekilde daha çok deęişkenli terimlerin de yapısını isimlendirmek mümkündür.

Analitik fonksiyonlarda olduęu gibi, veri ayrıştırımında da elde edilen YBMG terimleri toplanarak orijinal veri elde edilebilir veya terimlerin bir kısmı toplanarak orijinal veriye yaklaştırım yapılabilir. Veri ayrıştırımının terimlerinin toplanması söz konusu olduğundan, bu toplama işleminin yeniden yerleştirme yani süperpozisyon işleminin olacaktır. Dięer yandan, tek bir deęerden oluşan sabit terim, vektör yapısında olan farklı uzunluklardaki tek deęişkenli terimler vb. verilerin süperpoze edilmeksizin toplanması mümkün de deęildir.

Sabit yaklaştırım, fonksiyonun, tanım kümesinin tüm  $(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \dots, \xi_N^{(k_N)})$  noktalarında aldığı deęerlerin sabit terim olarak alınması ile elde edilir. Tek deęişkenli yaklaştırım, tanım kümesinin her bir  $(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \dots, \xi_N^{(k_N)})$  noktasındaki deęerin  $f_1(\xi_1^{(k_1)}) + f_2(\xi_2^{(k_2)}) + \dots + f_N(\xi_N^{(k_N)})$  şeklinde hesaplanması ve sonucun

sabit yaklařtırım ile toplanması ile elde edilir. İki deęiřkenli yaklařtırım ise, tanım kümesinin her bir  $(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \dots, \xi_N^{(k_N)})$  noktasındaki deęerin  $f_{12}(\xi_1^{(k_1)}, \xi_2^{(k_2)}) + f_{13}(\xi_1^{(k_1)}, \xi_3^{(k_3)}) + \dots + f_{(N-1)N}(\xi_{N-1}^{(k_{N-1})}, \xi_N^{(k_N)})$  řeklinde hesaplanması ve sonucun tek deęiřkenli yaklařtırım ile toplanması ile elde edilir. Benzer řekilde daha çok deęiřkenli yaklařtırmalar da elde edilebilir. YBMG veri ayrıřtırımının son terimi de, orijinal veriden  $N - 1$  deęiřkenli yaklařtırımın ıkarılmasıyla elde edilir. ( $k_i = 1, \dots, n_i ; i = 1, \dots, N$ )

Veri ayrıřtırımı terimleriyle yapılan yaklařtırmaların orijinal veriye olan yakınlıęı eřitli hata analizi yöntemleriyle ölçülebilir. Bu yöntemlerden biri olan baęıl hata analizi

$$\mathcal{N}_{s_i} = \frac{\|f_{orijinal} - f_{s_k}\|}{\|f_{orijinal}\|} \quad (2.21)$$

řeklinde uygulanabilir. Burada  $f_{orijinal}$  orijinal veriyi,  $f_{s_k}$  ise  $k$  deęiřkenli yaklařtırımı ifade etmektedir.

### 3. MATEMATİKSEL YÖNTEMLER

Bu kısımda tez çalışmasının alt yapısını oluşturan matematiksel yöntemler anlatılacaktır. Bu kısım beş bölümden oluşmaktadır. Öncelikle YBMG veri ayrıştırımının görüntü verisine uygulanışı ele alınacaktır. İkinci bölümde görüntü erişimini hızlandırmak amacıyla ileri sürülen kümeleme yaklaşımı anlatılacak ve çalışmada kullanılan kümeleme yöntemleri verilecektir. Sonraki bölümlerde küme sayısı seçimi, bir uzaklık ölçüsü olarak karesel Öklit uzaklığı ve bir değerlendirme yöntemi olarak azalan kümülatif kazanç yöntemi işlenecektir.

#### 3.1 YBMG Yönteminin Görüntüler Üzerinde Uygulanışı

Yüksek Boyutlu Model Gösterilimi (YBMG) yöntemiyle fonksiyon ve veri ayrıştırımı bir önceki kısımda anlatılmıştır. RGB formatındaki dijital görüntüler analitik fonksiyon yapısında değildir, ayrık değerlerden oluşan üç boyutlu dizi yapısına sahiptir. Bu yüzden YBMG yöntemi görüntüler üzerine uygulanırken, veri ayrıştırımı kuralları esas alınır.

RGB formatındaki bir görüntü ekranda, *düşey piksel sayısı*  $\times$  *yatay piksel sayısı* boyutunda dikdörtgensel bir alan olarak gösterilir. Her bir piksel ise sahip olduğu kırmızı(R) yeşil(G) ve mavi(B) yoğunluğunu belirten üç tane değer ile belirlenir. Bunlar, [0, 255] aralığında tamsayı değerleridir. Bu üç değer de 0 olması durumunda ilgili pikselde siyah renk gösterilir. Üç değer de 255 olması durumunda ise, her üç renk maksimum yoğunluktadır ve ilgili piksel beyaz olarak gösterilir.

YBMG ayrıştırımı görüntüler üzerine uygulanırken, görüntü verisi *düşey*, *yatay* ve *renk* adındaki üç değişkene bağlı bir fonksiyonun ayrık noktadaki değerleri olarak ele alınmıştır. Üç değişken olduğundan, YBMG ayrıştırımında bulunan  $N = 3$ 'tür ve dolayısıyla  $i = 1, 2, 3$  değerlerini alır. Kartezyen çarpım ile elde edilen tanım kümesinin üzerinde tanımlandığı kümelerin eleman sayıları ise  $n_1$  : *düşey piksel sayısı*,  $n_2$  : *yatay piksel sayısı* ve  $n_3$  : *renk sayısı*(= 3) şeklindedir.  $f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)})$  değerleri ise, görüntü verisini oluşturan değerlerdir ( $k_1 = 1, \dots, n_1$ ;  $k_2 = 1, \dots, n_2$ ;  $k_3 = 1, 2, 3$ ).

İlgili bölümde verildiği gibi, ayrıştırımda yer alan  $\alpha_{k_i}^{(i)}$  ağırlıkları aşağıdaki normalizasyon kriterini sağlamalıdır.

$$\sum_{k_i=1}^{n_i} \alpha_{k_i}^{(i)} = 1, \quad i = 1, 2, 3 \quad (3.1)$$

Bu durumda,  $\alpha_{k_i}^{(i)}$  ağırlıklarının

$$\alpha_{k_i}^{(i)} = \frac{1}{n_i}, \quad i = 1, 2, 3 \quad (3.2)$$

şeklinde homojen olarak seçilmesi hesaplamalarda kolaylık sağlayacaktır. Verilen ağırlıklar altında, görüntüler için YBMG veri ayrıştırımının sabit terimi, tek değişkenli ve iki değişkenli terimleri hesaplanabilir. Bir önceki kısımda Denklem 2.18, 2.19 ve 2.20'de verilen bağıntıların, yukarıda anlatıldığı gibi görüntülere uyarlanması ile,

$$\begin{aligned} f_0 &= \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \sum_{k_3=1}^3 \frac{1}{n_1} \cdot \frac{1}{n_2} \cdot \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) \\ f_1(\xi_1^{(k_1)}) &= \sum_{k_2=1}^{n_2} \sum_{k_3=1}^3 \frac{1}{n_2} \cdot \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_0 \\ f_2(\xi_2^{(k_2)}) &= \sum_{k_1=1}^{n_1} \sum_{k_3=1}^3 \frac{1}{n_1} \cdot \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_0 \\ f_3(\xi_3^{(k_3)}) &= \sum_{k_1=1}^{n_1} \sum_{k_2=1}^{n_2} \frac{1}{n_1} \cdot \frac{1}{n_2} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_0 \\ f_{12}(\xi_1^{(k_1)}, \xi_2^{(k_2)}) &= \sum_{k_3=1}^3 \frac{1}{3} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_1(\xi_1^{(k_1)}) - f_2(\xi_2^{(k_2)}) - f_0 \\ f_{13}(\xi_1^{(k_1)}, \xi_3^{(k_3)}) &= \sum_{k_2=1}^{n_2} \frac{1}{n_2} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_1(\xi_1^{(k_1)}) - f_3(\xi_3^{(k_3)}) - f_0 \\ f_{23}(\xi_2^{(k_2)}, \xi_3^{(k_3)}) &= \sum_{k_1=1}^{n_1} \frac{1}{n_1} \cdot f(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)}) - f_2(\xi_2^{(k_2)}) - f_3(\xi_3^{(k_3)}) - f_0 \end{aligned} \quad (3.3)$$

eşitlikleri elde edilir. Elde edilen bu terimlerin, yeniden yerleştirme yapılarak toplanmasıyla, yani terimlerin süperpozisyonu ile, orijinal görüntüye sabit yaklaştırım, tek değişkenli yaklaştırım ve iki değişkenli yaklaştırım yapılabilir. Görüntü üç değişkenli olarak ele alındığından en fazla iki değişkenli yaklaştırım yapılabilir. Üç değişkene birden bağlı olan  $f_{123}(\xi_1^{(k_1)}, \xi_2^{(k_2)}, \xi_3^{(k_3)})$  terimi ise, orijinal görüntü verisinden iki değişkenli yaklaştırım çıkarılarak elde edilir.

Bu tez çalışmasında YBMG veri ayrıştırımı, hem görüntü kümeleme, hem de ilgili küme içerisinde görüntü erişimi amacıyla kullanılmıştır. Görüntü kümelemede, tek bir değer içeren  $f_0$  fonksiyonu ve üç tane değer içeren  $f_3$  fonksiyonu kullanılmıştır. Görüntü kümeleme işlemi çok sayıda görüntü kullanılarak yapıldığından  $f_0$  ve  $f_3$  gibi oldukça düşük boyutlu verilerin kullanılması hesaplama karmaşıklığı açısından avantaj sağlamıştır. Görüntü erişimi için ise,  $n_1$  boyutlu  $f_1$  terimi ve görüntü ile aynı boyuta ( $n_1 \cdot n_2 \cdot 3$ ) sahip tek değişkenli yaklaşımlar tercih edilmiştir. Bunların yanı sıra  $f_1$ ,  $f_2$  ve  $f_3$  terimlerinin uç uca eklenmesiyle elde edilen  $n_1 + n_2 + 3$  boyutlu vektör de görüntü erişimi çalışmalarında kullanılmıştır. Bu veriler  $f_0$  ve  $f_3$ 'e göre daha yüksek boyutludur. Bu yüzden görüntüleri daha iyi temsil etmekte ve görüntü erişiminde yüksek başarı göstermektedirler.

### 3.2 Kümeleme Yöntemleri

Kümeleme makina öğrenmesi ve veri madenciliğinde yaygın olarak kullanılan bir yaklaşımdır. Bu yaklaşım, verilerden anlamlı bilgiler elde etmek veya verilerin başka amaçlarla işlenmesini kolaylaştırmak amacıyla kullanılmaktadır.

Kümeleme, ele alınan verileri kendi içinde olabildiğince benzer, kendi aralarında ise olabildiğince farklı gruplara ayırma işlemidir. Sınıflandırma işleminden farklıdır. Sınıflandırma işleminde ele alınan bir veri noktası daha önceden var olan sınıflardan birine atanır. Kümeleme işleminde ise veri seti, başlangıç grupları veya temel kabuller olmaksızın, belirlenen bir uzaklık kıstasına göre birbirine yakın olan verilerin bir arada toplandığı alt gruplara ayrılır.

Günümüzde, işlenmek istenen bir veri seti çok sayıda veri noktası içerebilmektedir. Bu durum, veri noktalarının çok boyutlu olması ve verinin işleneceği algoritmanın yüksek karmaşıklık düzeyine sahip olması gibi durumlar da eklenince, çalışmaların verimliliğini düşürmektedir. Hesaplama karmaşıklığını çok yükselten bu gibi durumlarda veri seti öncelikle kümelenecek ve sonra, veri setinin tamamı yerine tek bir küme üzerinde çalışılarak, verimlilik artırılabilir. Bu tez çalışmasında kümeleme bu amaçla kullanılmıştır. Önce veri tabanı görüntüleri kümelenecek, sonra ele alınan bir görüntünün benzerleri, bu görüntünün en yakın olduğu küme içerisinde getirilmiştir. Bu yaklaşım verimlilik bakımından kayda değer bir kazanç sağlamıştır.

Bu tez çalışmasında kümeleme işlemi eşit aralık yöntemiyle, k-ortalamlar yöntemiyle ve eşit üye sayısı yöntemiyle yapılmıştır. Bu kümeleme yöntemleri burada alt başlıklar halinde anlatılacaktır.

### 3.2.1 Eşit aralık yöntemi

Veri setinin tek boyutlu veri noktalarından oluşması durumunda, yani her bir veri noktasının tek bir değer olarak ifade edilmesi halinde, veri skalasını eşit parçalara ayırma yaklaşımı, en yalın kümeleme yaklaşımıdır. Bu durumda elde edilen eşit aralıklardan her biri, bir kümenin aralığı olacaktır.

Her biri tek bir değer içeren  $x_1, \dots, x_i, \dots, x_n$  verileri  $K$  kümeyle ayrılmak istendiğinde,  $\min(x_i)$  ile  $\max(x_i)$  arasında kalan aralık  $K$  eşit parçaya bölünerek her bir parça kümelerden birinin aralığı olarak belirlenir. Bunun için öncelikle,  $K$  tane küme için  $K + 1$  tane küme sınırı oluşturulur.  $b_1, \dots, b_j, \dots, b_{K+1}$  sınır değerlerini ifade etmek üzere, bu değerler aşağıdaki gibi hesaplanır.

$$b_j = \min(x_i) + (j - 1) \cdot \frac{\max(x_i) - \min(x_i)}{K}, \quad j = 1, \dots, K + 1; \quad i = 1, \dots, n \quad (3.4)$$

Eşit aralık yöntemiyle kümelemede, her bir verinin tek bir kümeyle atanması amaçlanır. Bu nedenle küme aralıklarını, ardışık sınır değerlerinin arası olarak ifade ederken yarı açık aralık kullanılır. Bu sayede, sınır değerlerinden birine eşit olan bir verinin komşu iki kümeden yalnız birine atanmasını sağlar. Soldan kapalı, sağdan açık aralık tercih edildiğinde,  $küme_j$  kümelerinin aralıkları

$$küme_j = [b_j, b_{j+1}), \quad j = 1, \dots, K \quad (3.5)$$

şeklinde ifade edilir. Burada  $küme_1$  ve  $küme_K$  aralıkları ayrıca değerlendirilmelidir. Küme aralıkları tespit edilip kümeleme yapıldıktan sonra veri seti dışından yeni bir veri noktası ele alınıp, yeniden kümelemeye gidilmeksizin, bu veri noktasının var olan kümelerden hangisine en yakın olduğunun araştırılması söz konusu ise, bu veri noktası veri setindeki noktaların tümünden daha küçük veya daha büyük olabileceğinden,  $küme_1$  ve  $küme_K$  aralıkları bu duruma uygun olarak daha geniş alınmalıdır. Bunun için  $küme_1$  soldan,  $küme_K$  ise sağdan sınırlandırılmamalıdır. Bir başka deyişle,  $b_1$  ve  $b_{K+1}$  aşağıdaki şekilde güncellenmelidir.



$$b_1 = -\infty \quad , \quad b_{K+1} = \infty \quad (3.6)$$

Buna uygun bir algoritma tasarımı,  $x_i$ 'nin alması mümkün olan en küçük değerden daha küçük bir değer  $b_1$  olarak alınması ve  $x_i$ 'nin alması mümkün olan en büyük değerden daha büyük bir değer  $b_{K+1}$  olarak alınması yöntemiyle yapılabilir. Sınır değerler tespit edildikten sonra tüm veriler Denklem 3.5'te verilen aralıklara göre kümelere atanarak kümeleme işlemi tamamlanır.

Bu tez çalışmasında eşit aralık yöntemi, eğitim görüntülerinin  $f_0$  değerlerinin kümeleneğinde kullanılmıştır. Ayrıca bu yöntemle,  $f_3$  vektörünün değerlerinden birinin seçilerek kümeleneği işlemi de yapılmıştır. Eşit aralık yönteminin algoritma tasarımına örnek olarak,  $f_0$  değerlerinin kümeleneği Şekil 3.1'de verilmiştir.

```

1 forall Eğitim görüntüleri do
2   | Eğitim görüntüsünü oku ve  $f_0$  değerini hesapla;
3   |  $f_0$  değerini sakla;
4 end
5 En büyük ve en küçük  $f_0$  değerlerini bul;
6 En büyük değer en küçük değerden farkını  $K$ 'ya bölerek aralık genişliğini
   hesapla;
7 for  $j$ , 2'den  $K$ 'ya kadar do
8   | En küçük  $f_0$  değerine, aralık genişliğinin  $j - 1$  katını ekleyerek  $j$ . sınır
   değeri bul;
9 end
10 1. sınır değeri  $-1$  olarak al;
11  $(K + 1)$ . sınır değeri 256 olarak al;
12 forall Eğitim görüntüleri do
13   | Eğitim görüntüsünün  $f_0$  değeri,  $j$ . sınır değerden büyük eşit ve  $(j + 1)$ . sınır
   değerden küçükse, bu görüntüyü  $j$ . kümeğe ata;
14 end
15 return Hesaplanan sınır değerlerini ve eğitim görüntülerinin hangi kümelere
   atandığı bilgisini geri döndür;

```

**Şekil 3.1** : YBMG- $f_0$  değerleri kullanılarak eğitim görüntülerinin eşit aralık yöntemiyle kümeleneği algoritması.

### 3.2.2 K-ortalamlar yöntemi

K-ortalamlar (k-means) yöntemi en yaygın kullanılan kümeleme yöntemlerinden biridir. İsminde bulunan  $K$ , küme sayısını temsil etmektedir. Bu yöntemin temel fikri, seçilmiş merkez değerleriyle başlanmak üzere, veri noktalarını merkezlere olan

yakınlığına göre birer kümeye atamak, ardından her kümenin küme merkezini yeniden hesaplamak ve bunu tekrarlayarak bir denge noktasına ulaşmaktır.

Her biri  $p$  tane değer içeren, yani her biri  $p$  boyutlu olan  $x_1, \dots, x_i, \dots, x_n$  verileri  $K$  kümeye ayrılmak istendiğinde öncelikle  $K$  tane başlangıç merkez noktası tespit edilir. Bu merkezler rastgele seçilebilir, *kmeans++* gibi olasılıksal algoritmalar ile belirlenebilir veya başka bir yöntemle elde edilebilir. Bu tez çalışmasında, olasılıksal bir seçim yerine iyi tanımlanmış başlangıç merkezlerinin kullanılması tercih edilmiştir. Bu merkezlerin  $l$ . değeri ( $l = 1, \dots, p$ ) eşit aralık yöntemine benzer bir mantıkla eşit aralıklarla belirlenmiştir:

$$\text{merkez}_{j;l} = \min(x_i(l)) + (2 \cdot j - 1) \cdot \frac{\max(x_i(l)) - \min(x_i(l))}{2 \cdot K}$$
$$j = 1, \dots, K ; l = 1, \dots, p ; i = 1, \dots, n \quad (3.7)$$

Burada  $n$  veri noktası sayısını,  $p$  veri noktalarının boyutunu,  $K$  ayrılmak istenen küme sayısını belirtmektedir.

Başlangıç merkez değerleri hesaplandıktan sonra veri setindeki her bir veri noktası, karesel Öklit uzaklığı ölçüsüne göre kendisine en yakın merkezin temsil ettiği kümeye atanır. Ardından bu atama işlemiyle elde edilen her bir küme için, kümede bulunan tüm veri noktalarının orta noktası bulunur. Bu orta noktalar kümelerin başlangıç merkezlerden farklı çıkabilir. Bulunan bu orta noktalar kümelerin yeni merkezleri olarak alınır. Tüm veriler, eski kümeleri dikkate alınmaksızın, bu yeni merkezlere göre yeniden kümelendir. Her atama işleminin sonunda küme merkezleri yenilenerek, ve ardından yeniden kümeleme yapılarak, merkezlerin artık değişmediği bir denge durumuna veya belirlenen maksimum tekrar sayısına ulaşıncaya kadar devam edilir. Elde edilen son kümeler, k-ortalamalar yönteminin sonucudur. Burada merkezlere uzaklığı ölçmede kullanılan karesel Öklit uzaklığı sonraki bölümlerde ele alınacaktır. Yaygın olarak kullanılan ölçü bu olmasına rağmen, farklı uzaklık ölçüleri kullanılarak da k-ortalamalar yönteminin uygulanması mümkündür.

Kümeleme, eşit aralık yönteminde aralıklar esas alınarak, k-ortalamalar yönteminde ise merkezlere yakınlık esas alınarak yapılmaktadır. Bununla birlikte, tek boyutlu veriler bu iki yöntemle kümelendirken, k-ortalamalar yöntemiyle yapılacak ilk basamak kümeleme işleminin, eşit aralık yöntemiyle aynı sonucu verdiği görülecektir. Çünkü

k-ortalamlar yöntemi için hesaplanan başlangıç merkezleri, eşit aralık yönteminde oluşturulan aralıkların orta noktalarıdır.

K-ortalamlar yöntemiyle bu tez çalışmasında  $f_0$  ve  $f_3$  değerleri kümelenmiştir. Örnek olarak,  $f_3$  değerlerinin kümeleme algoritması Şekil 3.2'de verilmiştir. Tek boyutlu  $f_0$  değerlerinin kümelenmesi bir önceki bölümde örneklendiğinden, burada üç boyutlu  $f_3$  değerlerinin kümeleme algoritmasının örnek verilmesi tercih edilmiştir. Çalışmalar sırasında k-ortalamlar yöntemiyle kümeleme işlemi, MATLAB programının kmeans() fonksiyonu kullanılarak yapılmıştır.

```
1 forall Eğitim görüntüleri do
2   | Eğitim görüntüsünü oku ve  $f_3$  vektörünü hesapla;
3   |  $f_3$  vektörünü sakla;
4 end
5 for  $l$ , 1'den 3'e kadar do
6   |  $f_3$  vektörlerinin  $l$ . sıradaki değerlerinin en küçük ve en büyüğünü bul;
7   | En büyük değerin ve en küçük değerden farkını  $2 \cdot K$ 'ya bölerek  $l$ . sıra için
   |   yarım aralık genişliğini hesapla;
8 end
9 for  $j$ , 1'den  $K$ 'ya kadar do
10  | for  $l$ , 1'den 3'e kadar do
11  |   | En küçük  $f_3(l)$  değerine,  $l$ 'ye ait yarım aralık genişliğinin  $2 \cdot j - 1$  katını
   |   | ekleyerek  $j$ . merkezin  $l$ . değerini bul;
12  | end
13 end
14 Hesaplanan başlangıç merkezlerini kullanarak eğitim görüntülerini kmeans()
   |   fonksiyonu ile kümele;
15 Elde edilen son merkezleri sakla;
16 return Son merkezleri ve eğitim görüntülerinin hangi kümelere atıldığı
   |   bilgisini geri döndür;
```

Şekil 3.2 : YBMG- $f_3$  değerleri kullanılarak eğitim görüntülerinin k-ortalamlar yöntemiyle kümelenmesi algoritması.

### 3.2.3 Eşit üye sayısı yöntemi

Temel kümeleme yaklaşımlarından biri de her kümeye eşit sayıda veri atanması yaklaşımıdır. Bu yöntemle tek boyutlu veri noktalarından oluşan bir veri seti kümelenmek istendiğinde, ilk olarak veriler küçükten büyüğe doğru sıralanır. Veri sayısı ve hedeflenen küme sayısı dikkate alınarak her kümeye kaç veri düşeceği (üye sayısı) tespit edilir. Sıralanmış veri setinde, üye sayısı ve katlarını ifade eden sıralarda bulunan veriler her bir küme için son sınır değeri olarak belirlenir. Böylece küme

aralıkları, soldan açık sağdan kapalı aralık olarak tespit edilir. Tüm veriler bu küme aralıklarına göre kümelere atanır.

Her biri tek boyutlu olan  $x_1, \dots, x_i, \dots, x_n$  verileri  $K$  kümeye ayrılmak istendiğinde, öncelikle veriler küçükten büyüğe doğru sıralanarak  $x'_i$  sıralanmış veri seti elde edilir.  $n$  tane veri noktası bulunduğundan kümelerin her birinin  $\frac{n}{K}$  üyeye sahip olması beklenir. Bu oranın bir tamsayı vermediği durumlarda, üye sayısının olabildiğince eşit olması hedeflenir. Bunun için her bir küme numarasına karşılık, *küme numarası*  $\cdot \frac{n}{K}$  değerleri hesaplanır ve en yakın tamsayıya yuvarlanır.  $x'_i$  veri setinde bu indislerde bulunan veriler kümelerin üst sınır değerleri olarak belirlenir.

$b_0, \dots, b_j, \dots, b_K$  sınır değerleri olmak üzere, bu değerler aşağıdaki gibi hesaplanır.

$$b_j = x'_{[j \cdot \frac{n}{K}]}, j = 1, \dots, K - 1 \quad (3.8)$$

$$b_0 = -\infty \quad (3.9)$$

$$b_K = \infty \quad (3.10)$$

Burada  $[\dots]$  operatörü, en yakın tamsayıya yuvarlama operatörüdür.  $b_0$  ve  $b_K$  sınır değerlerinin ayrıca değerlendirilmesinin nedeni, eşit aralık yönteminde olduğu gibi, belirlenen aralıkların, kümelenen veri seti dışından bir veri noktası için de küme ataması yapabilmelerini sağlamaktır. Bu sınır değerler kullanılarak küme aralıkları

$$küme_j = (b_{j-1}, b_j], j = 1, \dots, K \quad (3.11)$$

şeklinde elde edilir. Veri setindeki tüm veriler bu aralıklara göre kümelere atanarak kümeleme işlemi tamamlanır. Bu kümeleme yöntemi bu tez çalışmasında tek boyutlu  $f_0$  değerlerinin kümelenmesinde kullanılmıştır.

### 3.3 Küme Sayısı Seçimi

Anlatılan kümeleme yöntemlerinde de görüldüğü gibi kümeleme, temel kabuller olmaksızın yapılan bir gruplandırma işlemidir. Veri noktaları seçilen bir yöntem ve belirlenen bir uzaklık ölçüsüne göre kümelere atanır. Yöntem ve uzaklık ölçüsünün yanı sıra kümelemede önemli olan diğer bir etken, seçilen küme sayısıdır. Küme

sayısı seçimi için tamamen objektif bir kural yoktur, fakat bazı sistematik yaklaşımlar geliştirilmiştir.

Küme sayısı, ampirik kural olarak bilinen

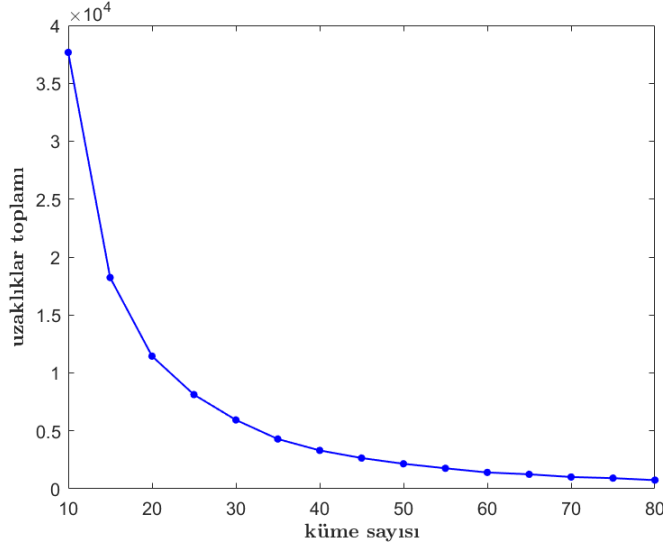
$$K \approx \sqrt{\frac{n}{2}} \quad (3.12)$$

kuralıyla seçilebilir. Burada  $n$  veri sayısını,  $K$  küme sayısını temsil etmektedir. Yapılan çalışmalarda daha başarılı bulunmuş küme sayılarının, bu kuralın verdiği sayıya yaklaşık olduğu tespit edilmiştir. Bu yüzden, ilk fikri edinmek için yaygın olarak kullanılmaktadır.

Bu tez çalışmasında deneyler, 7200 tane görüntü içeren COIL-100 veri tabanı üzerinde yapılmıştır. Bu görüntülerden 100 tanesi test amacıyla ayrılmış, 7100 tanesi ise eğitim görüntüsü olarak kullanılmıştır.  $n = 7100$  için ampirik kural ile hesaplanan küme sayısı  $K \approx 60$  çıkmaktadır. Fakat sadece veri sayısına bakmak ideal küme sayısına karar vermek için yeterli değildir; ideal küme sayısı verilerin kendisine göre de değişiklik gösterebilmektedir. Bu yüzden ilk fikir edinildikten sonra, küme sayısını çalışılan veri setine uygun seçmek amacıyla dirsek metodu adı verilen görsel bir yöntem kullanılır.

Dirsek metodu şu şekilde uygulanmaktadır: Belli aralıklarla seçilen farklı küme sayılarıyla veri seti kümelendir. Yapılan her bir kümelemenin sonunda tüm veri noktalarının kendi merkezlerine olan uzaklıklarının toplamı hesaplanır ve bu toplam değerleri kullanılarak değerlendirme yapılır. Küme sayısı  $K$  azaldıkça bu toplam giderek büyüyecektir. Bu, küme içi veri benzerliğinin giderek azalması anlamına gelmektedir. Yani küme sayısı az olduğundan çok farklı veriler aynı kümelere atanmaktadır, dolayısıyla merkezlerine olan uzaklıkları toplamı yüksek çıkmaktadır. ( $K = 1$  seçilmesi durumunda bahsedilen toplam en büyük değerine ulaşacaktır, bu ise kümeleme yapılmaması demektir.) Bu yüzden, küme içi veri benzerliğinin yükseltebilmek yani bahsedilen toplamı küçültebilmek için,  $K$  artırılmalıdır. Fakat küme sayısını çok yüksek seçmek kümeleme işlemi fazla seçici kılacak, çok benzer verileri de ayırmaya başlayacaktır. (Küme sayısı veri sayısına ulaştığında, yani  $K = n$  seçildiğinde bu toplam 0 değerine ulaşacaktır, bu ise anlamsızdır.) Öyleyse optimal küme sayısı, bahsedilen toplamın yeterince küçük olduğu, küme sayısını daha fazla artırmanın toplamı küçültmeye kayda değer bir katkı sağlamayacağı sayıdır.

Bu sayıyı belirleyebilmek için, farklı küme sayılarıyla yapılan kümeleme işlemleri için hesaplanan toplam değerleri bir grafikte incelenir ve görsel olarak bir dirsek noktası tespit edilmeye çalışılır. Bu tez çalışmasında kullanılan veri tabanı üzerinde dirsek metodu uygulandığında Şekil 3.3'te verilen grafik elde edilmiştir. Bu metod uygulanırken, ampirik kural ile elde edilen 60 sayısı civarında küme sayıları ile hesaplama yapılmıştır.



**Şekil 3.3** : Küme sayısı seçiminde dirsek metodu grafiği.

Bu grafikten optimal küme sayısının  $K \approx 30$  civarında olduğu tespit edilmiştir. Farklılıkları da ortaya koyabilmek açısından  $K = 20, 30, 40, 50, 60$  küme sayılarının kullanılmasına karar verilmiştir.

### 3.4 Karesel Öklit Uzaklığı

$a$  ve  $b$ ,  $n$  boyutlu Öklit uzayında herhangi iki nokta olmak üzere, karesel Öklit uzaklığı aşağıdaki gibi tanımlanmaktadır.

$$D(a, b) = \sum_{i=1}^n (a_i - b_i)^2 \quad (3.13)$$

Bu ölçü,

$$d(a, b) = \sqrt{\sum_{i=1}^n (a_i - b_i)^2} \quad (3.14)$$

şeklinde verilen Öklit uzaklığının karesine eşittir. İsmi de buradan almaktadır. Karesel Öklit uzaklığı ölçüsü, üçgen eşitsizliği özelliğini sağlamadığından bir metrik değildir. Bu yüzden bu ölçüm ile elde edilen sonuçlar genellikle bir uzaklık ölçüsü olarak kabul edilmez. Fakat uzaklıkların sadece karşılaştırma amacıyla kullanıldığı çalışmalarda metrik bir ölçüye ihtiyaç yoktur. Karesel Öklit uzaklığı ile de aynı karşılaştırma sonuçları elde edildiğinden, Öklit uzaklığı tanımındaki karekök alma işlemi, hesaplama karmaşıklığını azaltmak amacıyla ihmal edilir.

Karesel Öklit uzaklığı bu çalışmada görüntülerin aynı tip YBMG bileşenlerinin arasındaki uzaklığın aynı şekildeki diğer uzaklıklarla karşılaştırılmasında kullanılmıştır. Örneğin, seçilen bir görüntünün  $f_1$  vektörünün başka görüntülerin  $f_1$  vektörlerinden hangisine daha yakın olduğunu saptamak için aralarındaki karesel Öklit uzaklıkları karşılaştırılmıştır.

### 3.5 Azalan Kümülatif Kazanç

Azalan kümülatif kazanç (AKK, Discounted cumulative gain, DCG), bilgi getirme çalışmalarında kullanılan bir değerlendirme yöntemidir. Bu yöntem, getirilen bilgilerin isabetliliğini, getirilme sıralarını da dikkate alarak ölçer. Arama motorları ile dosya ve görüntü getirme algoritmalarında, algoritmanın başarısını ölçmek için yaygın olarak kullanılır. Temelde bir arama motorunun veya benzeri bir yapının isabetli bilgileri getirmesi istenir. Ayrıca getirilen isabetli bilgilerden en alakalı olanların sıralamada daha önce gelmesi daha yararlı bulunur. Azalan kümülatif kazanç, bu iki hedefin ne derece başarıldığının birlikte değerlendirilmesini sağlar. Getirilen bilgilerin ilk  $p$  tanesinin kazancı  $AKK_p$  olmak üzere azalan kümülatif kazanç

$$AKK_p = \sum_{i=1}^p \frac{rel_i}{\log_2(i+1)} \quad (3.15)$$

şeklinde tanımlanır. Burada  $rel_i$  getirilen bilginin isabetlilik puanıdır. Yani getirilen bilgilerin ne kadar yararlı olduğunu yargılayan bir kullanıcının verdiği puandır. Görüldüğü gibi azalan kümülatif kazanç, getirilen her bir bilginin isabetlilik puanının, getirilme sırası oranında azaltılarak toplanması anlamına gelmektedir. Azalan kümülatif kazanç için bir başka tanım

$$AKK_p = \sum_{i=1}^p \frac{2^{rel_i} - 1}{\log_2(i+1)} \quad (3.16)$$

şeklinde yapılmaktadır. Bu tanımın avantajı, getirilen bilgilerin isabetli bilgiler olmasını, getirilme sırasına göre daha fazla ön plana çıkarıyor olmasıdır. Fakat eğer isabetlilik ikilik sisteme göre yargılanıyorsa, yani sadece  $\{0, 1\}$  puanları kullanılıyorsa, bu iki tanım aynı sonucu verecektir.

Bu tez çalışmasında, azalan kümülatif kazanç ölçüsü, görüntü erişim deneylerinin başarısının değerlendirilmesinde kullanılmıştır. Örneğin, sorgu görüntüsü olarak alınan bir görüntüye, en yakın altı görüntü sırayla getirilmiş, bu görüntülerin sorgu görüntüsü ile aynı objeye ait olup olamaması ikilik sisteme göre puanlanmış ve bu puanlarla azalan kümülatif kazanç hesaplanmıştır. Başarıyı bir oran olarak ifade edebilmek için hesaplanan kazanç, ideal durumda elde edilecek kazanç oranlanmıştır. İdeal durum, getirilen görüntülerin hepsinin sorgu görüntüsü ile aynı objeye ait olduğu durumdur.

Bu kısımda, tez çalışmasının temelini oluşturan matematiksel yapılar ayrıntılı olarak ele alınmıştır. Bir sonraki kısımda önerilen yöntemler anlatılacak ve elde edilen deney sonuçları verilecektir.



#### 4. ÖZGÜN BİR YÖNTEM OLARAK YÜKSEK BOYUTLU MODEL GÖSTERİLİMİ İLE GÖRÜNTÜ KÜMELEME VE GÖRÜNTÜ ERİŞİMİ

Bu tez çalışmasında görüntü erişimi problemi ele alınmış ve çeşitli yöntemler önerilmiştir. Çalışmanın amacı, görüntü erişimini etkin ve hızlı bir şekilde yapmaktır. Görüntü erişimi problemlerinde ise, görüntülerin çok boyutlu veriler olması ve görüntü veri tabanlarının çok fazla sayıda görüntü içermesi gibi hesaplama sürelerini ve başarı oranlarını olumsuz etkileyen faktörler söz konusudur. Bu sorunlarla baş edebilmek amacıyla bu çalışmada, boyut indirgeme ve kümeleme yaklaşımları kullanılmıştır.

Görüntü erişimi çalışmalarında veri tabanındaki her bir görüntü, bir veri noktası anlamına gelmektedir. Önceki kısımlarda belirtildiği gibi RGB formatındaki renkli bir görüntü  $düşey\ piksel\ sayısı(n_1) \cdot yatay\ piksel\ sayısı(n_2) \cdot renk\ sayısı(3)$  tane tamsayıdan oluşmaktadır. Dolayısıyla görüntüler  $n_1 \cdot n_2 \cdot 3$  boyutlu veri noktalarıdır. Bu boyut görüntü erişimi çalışması yapılabilmesi için oldukça fazladır. Bu tez çalışmasında, bu probleme bir çözüm olarak, Yüksek Boyutlu Model Gösterilimi (YBMG) yöntemiyle boyut indirgeme yapılması önerilmektedir.

Görüntü erişimi çalışmalarında diğer bir konu olan veri tabanı ölçeğinin çok büyük olması sorununa ise kümeleme yaklaşımı önerilmiştir. Görüntülerin öncelikle kümelenmesi ile, büyük veri tabanının küçük veri tabanlarına ayrılması amaçlanmıştır. Kümeleme işleminden sonra gelen görüntü erişimi basamağında ise, ele alınan sorgu görüntüsüne en yakın küme tespit edilerek, bu görüntünün benzerleri sadece bu küme içerisinde aranmıştır. Kümeleme işleminde görüntüleri temsil etmek amacıyla kullanılan veriler, yine YGMG ayrıştırımından elde edilen düşük boyutlu veriler arasından seçilmiştir. Yani YBMG ayrıştırımı, hem kümeleme aşamasında hem de görüntü erişimi aşamasında kullanılmaktadır.

Çalışmada kümeleme ve ilgili kümeden görüntü erişimi yaklaşımını içeren çok sayıda yöntem önerilmiştir. Yöntemlerde kümeleme işlemi, daha düşük boyutlu olmaları sebebiyle, YBMG açılımının sabit terimi olan  $f_0$  veya tek değişkenli terimlerinden biri olan  $f_3$  terimi kullanılarak yapılmıştır. Bu veriler esas alınarak görüntüler eşit

aralık yöntemi, k-ortalamlar (k-means) yöntemi veya eşit üye sayısı yöntemiyle küçük kümelere ayrılmıştır.

Kümeleme basamağı tamamlandıktan sonra ele alınan bir sorgu görüntüsünün aynı YBMG bileşeni ( $f_0$  veya  $f_3$ ) hesaplanmış ve bu değer en yakın olduğu küme (hedef küme) tespit edilmiştir. Bu küme, görüntü erişimi çalışmasının yapılacağı küçük bir veri tabanı olarak ele alınmıştır.

Görüntü erişimi basamağında görüntüleri temsil etmede kullanılmak üzere, üç farklı YBMG verisi önerilmiştir. Bunlardan ilki  $n_1$  sayıda değer içeren  $f_1$  bileşenidir. İkincisi,  $f_1$ ,  $f_2$  ve  $f_3$  terimlerinin uç uca eklenmesiyle elde edilen vektördür ve  $n_1 + n_2 + 3$  tane değer içermektedir. Üçüncüsü ise YBMG tek değişkenli yaklaşım değerleridir. Tek değişkenli yaklaşım,  $f_1$ ,  $f_2$  ve  $f_3$  bileşenlerinin süperpozisyonu ile elde edilen üç boyutlu bir dizidir ve  $n_1 \cdot n_2 \cdot 3$  tane değer içermektedir. Dolayısıyla görüntü ile aynı boyuta sahiptir ve boyut indirgeme amacıyla kullanılamaz. Fakat çok boyutlu olarak da görüntü erişimi aşamasında, görüntüleri temsil etmek amacıyla tercih edilebilir.

Bu üç çeşit veriden seçilen bir tanesi, sorgu amacıyla ele alınan bir görüntü ve bu görüntünün hedef kümesindeki tüm görüntüler için hesaplanmıştır. Bu veriler esas alınarak, sorgu görüntüsü ile hedef kümede yer alan her bir görüntü arasındaki karesel Öklit uzaklığı bulunmuştur. Bulunan uzaklıklara göre sorgu görüntüsüne en yakın olan görüntüler sırayla getirilmiştir. Getirilen görüntülerin, özellikle de ilk sıralarda getirilenlerin, sorgu görüntüsü ile aynı objeye ait olması görüntü erişimi çalışmasının hedefidir.

Çalışmada önerilen yöntemler, COIL-100 görüntü veri tabanı üzerinde yapılan deneysel çalışmalarla sınanmıştır. Bu veri tabanı, 100 objenin her birinin 72 farklı görüntüsünü içermektedir ve toplam 7200 tane görüntüden oluşmaktadır. Görüntüler, her objenin kendi ekseni etrafında 5'er derece döndürülerek fotoğraflanmasıyla elde edilmiştir. Veri tabanından bir objenin tüm görüntüleri Şekil 4.1'de örnek olarak verilmiştir. Veri tabanında bulunan tüm objeler ise, ilk görüntüleri alınarak Şekil 4.2'de örneklenmiştir.

Bu tez çalışmasındaki deneyleri uygulanmak amacıyla, COIL-100 görüntü veri tabanındaki görüntüler, eğitim görüntüleri ve test görüntüleri olarak iki gruba



ayrılmıştır. Her bir objenin 72 adet görüntüsünden rastgele seçim yöntemiyle seçilen bir tanesi test amacıyla ayrılmış, diğerleri eğitim görüntüsü olarak kullanılmıştır. Bu ayrımın sonucunda 100 test görüntüsü ve 7100 eğitim görüntüsü elde edilmiştir.

Bölüm 3.3'te açıklandığı gibi, COIL-100 görüntü veri tabanı üzerinde önerilen yöntemlerin test edilmesinde, eğitim görüntülerinin 20,30,40,50 ve 60 kümeye ayrılması durumları incelenmiştir. Bu küme sayılarıyla yapılan kümeleme işlemleri birbirinden bağımsız çalışmalar olarak ele alınmıştır.

Görüntü erişimi deneyleri, getirilen görüntülerden ilk 6, 2 veya 1 tanesi dikkate alınarak değerlendirilmiştir. 6, 2 ve 1 sayıları, etkili bir değerlendirme sağlayacağı düşünülen subjektif olarak seçilmiştir. Değerlendirmede dikkate alınan her bir görüntüye sorgu görüntüsüyle aynı objeye ait olması durumunda 1, farklı objeye ait olması durumunda 0 puan verilmiş ve getirme sırası esas alınarak azalan kümülatif kazanç hesaplanmıştır. Elde edilen kazanç, tümünün 1 puan alacağı ideal durumda elde edilecek olan kazanç oranlanmıştır. Bu çalışmada verilecek olan başarı oranları, 100 adet sorgu görüntüsünün hepsi için yapılan deneylerden elde edilen oranların ortalaması alınarak hesaplanmıştır ve yüzdeler olarak ifade edilmiştir.

Önerilen çalışmalarda kullanılan matematiksel yapılar önceki kısımda detaylı olarak ele alınmıştır. Bu kısımda, önerilen kümeleme süreçleri ayrı bölümlerde ve görüntü erişim süreçleri ise bu bölümlerin alt başlıklarında incelenecek ve deneylerden elde edilen başarı oranları da bu alt başlıklarda verilecektir.

#### **4.1 Eşit Aralık Yöntemi ile $f_0$ Değerlerini Kümeleme**

Eğitim görüntüleri,  $f_0$  değerleri esas alınarak eşit aralık yöntemiyle kümelenecektir. Kümelenecek olan eğitim görüntüleri çok sayıda olduğundan, hesaplama karmaşıklığını azaltmak için  $f_0$  gibi tek değer içeren bir veri kullanılması tercih edilmiştir. Bu kümeleme işleminin algoritma tasarımı önceki kısımda eşit aralık yöntemi anlatılırken örnek olarak verilmiştir (Şekil 3.1).

##### **4.1.1 $f_1$ değerleri ile görüntü erişimi**

Kümeleme işlemi tamamlanıp görüntü erişimi basamağına gelindiğinde, ele alınan bir sorgu görüntüsüne en yakın görüntüleri ilgili kümeden getirmek için kullanılacak

düşük boyutlu veri olarak, bu çalışmada, bir tek değişkenli YBMG bileşeni olan  $f_1$  verisi seçilmiştir.  $f_1$  bir vektör formatındadır ve  $n_1$  tane değer içermektedir.  $f_0$  verisine göre daha fazla bilgi içerdiğinden görüntüyü daha başarılı bir şekilde temsil etmektedir. Bu yüzden, görüntü erişimi basamağında kullanılması tercih edilmiştir. Şekil 4.3'te bu görüntü erişimi çalışması özetlenmiştir. Bu algorithmada yer alan  $S$ , en yakın kaç tane görüntünün getirilmek istendiğidir.

```

1 Sorgu görüntüsünü oku ve  $f_0$  değerini hesapla;
2 Kümeleme basamağında hesaplanan sınır değerlerini kullanarak hedef kümeyi
  belirle;
3 Sorgu görüntüsünün  $f_1$  değerini hesapla;
4 forall Hedef kümedeki eğitim görüntüleri do
5   | Eğitim görüntüsünü oku,  $f_1$  vektörünü hesapla;
6   | Eğitim görüntüsünün  $f_1$  vektörü ile sorgu görüntüsünün  $f_1$  vektörü
   | arasındaki karesel Öklit uzaklığını hesapla;
7   | Uzaklık bilgisini sakla;
8 end
9 Uzaklıkları küçükten büyüğe sırala;
10 En küçük  $S$  tane uzaklığı sıralı olarak al;
11 forall En küçük  $S$  tane uzaklık do
12   | Uzaklığın hangi eğitim görüntüsünden elde edildiğini bul;
13   | if Bu eğitim görüntüsü ile sorgu görüntüsü aynı objeye aitse then
14     | En küçük kaçınıcı uzaklık olduğuna göre azalan kümülatif kazanç
     | katkısını hesapla ve kazanca ekle;
15   | end
16 end
17 Hesaplanan kazancı ideal durum kazancına oranla;
18 return Kazanç oranını ve hesaplama süresini geri döndür;
```

**Şekil 4.3** : YBMG- $f_1$  değerleri ile ilgili kümeden görüntü erişimi algoritması.

Bu deney, sorgu görüntüsü olarak ayrılan tüm görüntüler üzerinde tekrarlanmış ve elde edilen sonuçların aritmetik ortalaması alınmıştır. Ayrıca kümeleme işlemi farklı küme sayıları ile yapıldığından, her bir kümeleme işlemi için de görüntü erişimi çalışmaları tekrarlanmış ve başarı oranları hesaplanmıştır. En yakın 6, 2 ve 1 görüntü getirilmesi durumlarında, azalan kümülatif kazanç yöntemiyle hesaplanan başarı oranları Çizelge 4.1'de verilmiştir.

Çizelge 4.1'de görüldüğü gibi bu yöntemle kayda değer sonuçlar elde edilmektedir. Ayrıca, küme sayısı azaldıkça sonuçlar iyileşmektedir. Bunun nedeni, küme sayısı azaldıkça her bir kümeye düşen görüntü sayısının artmasıdır. Kümelerdeki görüntü sayısının artması, aynı kümeye düşen aynı objeye ait görüntü sayılarının artmasına

**Çizelge 4.1** : Eşit aralık yöntemiyle  $f_0$  değerlerini kümeleme ve  $f_1$  değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 87.1	% 81.3	% 75.4	% 72.1	% 68.6
2	% 92.9	% 90.3	% 85.5	% 81.1	% 80.8
1	% 96.0	% 93.0	% 89.0	% 85.0	% 82.0

ve dolayısıyla görüntü erişimi deneyinde doğru görüntülere ulaşılma ihtimalinin de artmasına neden olmaktadır. Fakat diğer yandan kümelerdeki görüntü sayısı arttıkça, görüntü erişimi çalışmasının hesaplama zamanı bir miktar artacaktır. Çünkü, görüntü erişimi  $f_1$  gibi,  $f_0$ 'a kıyasla çok daha fazla değer içeren bir bileşen ile yapılmaktadır ve kümelerdeki görüntü sayısının artması,  $f_1$  bileşenleri karşılaştırılacak veri sayısının artması anlamına gelmektedir.

#### 4.1.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi

Görüntü erişimi aşamasında,  $f_1$  vektörü yerine, vektör yapısında olan YBMG tek değişkenli verilerinin uç uca eklenmesiyle elde edilen vektör de kullanılmıştır.  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle elde edilen bu vektör  $n_1 + n_2 + 3$  tane değere sahiptir ve görüntüyü daha ayrıntılı bir şekilde temsil etmektedir. Fakat sadece  $f_1$ 'in kullanılması durumuna kıyasla daha fazla veri içerdiğinden, çalışmanın hesaplama süresinin artmasına neden olmaktadır. Bu çalışmanın algoritma yapısı,  $f_1$  verisi yerine  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesi yazılması kaydıyla, Şekil 4.3'te verilen algoritmaya benzer olarak tasarlanmıştır. Elde edilen başarı oranları Çizelge 4.2'de verilmiştir.

**Çizelge 4.2** : Eşit aralık yöntemiyle  $f_0$  değerlerini kümeleme ve  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 84.3	% 78.3	% 72.8	% 69.7	% 67.1
2	% 94.7	% 89.5	% 83.9	% 81.2	% 79.8
1	% 97.0	% 93.0	% 87.0	% 87.0	% 81.0

Elde edilen başarı oranlarının bir önceki çalışmaya göre yükselmesi beklenirken bazı durumlarda bir miktar düştüğü görülmektedir. Görüntüye ait daha fazla bilgi taşınmasına rağmen başarı oranlarının düşmesi durumunun, her biri görüntü verisinin farklı açılardan kullanılmasıyla hesaplanan  $f_1$ ,  $f_2$ ,  $f_3$  terimlerin uç uca eklenmesinin bir sonucu olarak, karesel Öklit uzaklığı hesaplanırken  $f_1$ 'lerin,  $f_2$ 'lerin ve  $f_3$ 'lerin

farkı birlikte toplandığından  $f_1, f_2, f_3$  terimlerinin birbirine gürültü etkisi yapmasından kaynaklandığı düşünülmektedir.

#### 4.1.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi

Burada, görüntü erişimi için YBMG tek değişkenli yaklaştırım değerleri kullanılmıştır. Tek değişkenli yaklaştırım,  $f_1, f_2$  ve  $f_3$ 'ün süperpozisyonu ve hesaplanan tüm değerlere  $f_0$ 'ın eklenmesi ile elde edilir.  $n_1 \cdot n_2 \cdot 3$  tane değer içerir. Orijinal görüntü ile aynı boyuta sahip olduğundan boyut indirgeme avantajı sağlamaz. Fakat görüntü erişimindeki başarısını test etmek ve böylece  $f_1, f_2$  ve  $f_3$  vektörlerinin görüntüyü temsil etme yeteneğini anlamak için çalışmaya dahil edilmiştir. Tek değişkenli yaklaştırım değerleri kullanılarak yapılan görüntü erişimi çalışmasının algoritma yapısı,  $f_1$  verisi yerine tek değişkenli yaklaştırım yazılması kaydıyla, Şekil 4.3'te verilen algoritmaya benzer şekilde oluşturulmuştur. Elde edilen başarı oranları Çizelge 4.3'te verilmiştir.

**Çizelge 4.3** : Eşit aralık yöntemiyle  $f_0$  değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 88.4	% 82.1	% 76.9	% 73.5	% 71.2
2	% 96.1	% 91.7	% 87.5	% 85.2	% 83.1
1	% 98.0	% 94.0	% 89.0	% 91.0	% 87.0

Bu çalışmanın başarısının, önceki iki çalışmaya göre yükseldiği görülmektedir. Buradan  $f_1, f_2$  ve  $f_3$  terimlerinin birlikte, görüntüyü başarılı bir şekilde temsil edebildiği sonucuna ulaşılır. Tek değişkenli yaklaştırım verisi çok boyutlu olduğundan karesel Öklit uzaklığı hesaplanıp karşılaştırılacak veri büyüktür ve bu durum ek zaman maliyeti getirmektedir.

Buraya kadar  $f_0$  değerlerinin eşit aralık yöntemi ile kümeleneceği durumunda, yapılan farklı görüntü erişimi çalışmalarının başarı oranları incelenmiştir. Bu görüntü erişim çalışmalarıyla getirilen görüntülere örnekler, Şekil 4.4, 4.5 ve 4.6'da verilmiştir. Bu örnekler, görüntülerin 20 kümeye ayrılması durumunda yapılan görüntü erişim deneylerinden seçilmiştir. Bu kısım boyunca diğer çalışma sonuçlarına da örnek resimler verilirken, 20 küme sayısı esas alınacaktır.



**Şekil 4.4 :** Çizelge 4.1 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).





**Şekil 4.5 :** Çizelge 4.2 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



**Şekil 4.6 :** Çizelge 4.3 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).

## 4.2 K-ortalamlar Yöntemi ile $f_0$ Değerlerini Kümeleme

Bu bölümdeki çalışmalar için eğitim görüntüleri,  $f_0$  değerleri esas alınarak k-ortalamlar (k-means) yöntemiyle kümelendi. Bölüm 3.2.2’de anlatıldığı gibi k-ortalamlar yönteminin ilk basamağında yapılan kümeleme işlemi, eşit aralık yöntemi ile aynı kümeleme değildir. Ardından k-ortalamlar algoritması işletilerek küme merkezleri ve dolayısıyla oluşturulan kümeler değişmiştir. Bu yöntemle 20, 30, 40, 50 ve 60 küme sayıları için ayrı ayrı kümeleme işlemleri yapılmıştır. Farklı verilerle görüntü erişim çalışmaları yapılırken bu kümeleme sonuçları üzerinde ayrı ayrı çalışılmıştır.

### 4.2.1 $f_1$ değerleri ile görüntü erişimi

Kümeleme basamağı tamamlandıktan sonra yapılan görüntü erişimi çalışmasında, ele alınan bir sorgu görüntüsünün hedef kümesinin belirlenmesinde k-ortalamlar yöntemiyle elde edilen son merkezler kullanılmıştır. Hedef küme içerisinde karesel Öklit uzaklığına göre sorgu görüntüsüne en yakın görüntülerin getirilmesi için, görüntülerin  $f_1$  vektörleri esas alınmıştır. Sorgu görüntüsüne en yakın görüntüler sıralı olarak getirilmiş ve sorgu görüntüsüyle aynı objeye ait olup olmadıkları değerlendirilmiştir. Bu çalışmalarla elde edilen başarı oranları Çizelge 4.4’te verilmiştir.

**Çizelge 4.4** : K-ortalamlar yöntemiyle  $f_0$  değerlerini kümeleme ve  $f_1$  değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 84.9	% 80.0	% 76.8	% 73.2	% 66.0
2	% 92.1	% 88.1	% 87.3	% 83.3	% 77.7
1	% 94.0	% 90.0	% 90.0	% 86.0	% 82.0

Bu çalışma sonuçları, görüntü erişimi için yine  $f_1$  değerlerinin kullanıldığı, fakat kümelemenin k-ortalamlar yerine eşit aralık yöntemiyle yapıldığı önceki çalışmayla karşılaştırıldığında, başarı oranlarında az bir düşüş görülmektedir. Başka bir deyişle eşit aralık yöntemi k-ortalamlar yöntemine göre daha iyi bir performans göstermiştir. K-ortalamlar yöntemi ilk kümeleme basamağında, eşit aralık yöntemi ile aynı atamaları yapmasına rağmen, sonraki basamaklarında yer alan merkezlerin yenilenmesi prensibi başarı düşüşüne neden olmuştur. Bu durumun, dar bir skalada değerler alan çok sayıda verinin (7100 tane  $f_1$  verisi, [0,255] aralığında) bazı

noktalarda çok yoğunlaşması, k-ortalamlar yönteminin ise merkezleri verilerin yoğun olduğu noktalara doğru taşınması ve bunun da fazla seçici bir kümelemeye, yani aynı objeye ait görüntülerin aynı kümeye atanmasında sapmalara neden olmasından kaynaklandığı düşünülmektedir. Eşit aralık yöntemi ise yapısı gereği yoğunluk ve seyreklikten etkilenmemektedir.

#### 4.2.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi

Bölüm 4.1.2'deki çalışmaya benzer şekilde bu çalışmada da, görüntü erişimi amacıyla  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle elde edilen vektör kullanılmış, fakat bu çalışma  $f_0$  değerlerinin k-ortalamlar yöntemiyle kümeleneşinden elde edilen kümeler üzerinde yapılmıştır. Deneylerden elde edilen başarı oranları Çizelge 4.5'te verilmiştir.

**Çizelge 4.5** : K-ortalamlar yöntemiyle  $f_0$  değerlerini kümeleme ve  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 82.9	% 79.0	% 74.2	% 70.8	% 64.8
2	% 93.7	% 88.5	% 86.7	% 82.4	% 76.6
1	% 96.0	% 92.0	% 91.0	% 87.0	% 82.0

Bu çalışma sonuçları ile, görüntü erişimi için aynı değerlerin kullanıldığı, fakat kümelemenin eşit aralık yöntemiyle yapıldığı çalışma sonuçları karşılaştırmasında, başarının bu çalışmada bir miktar düşüş göstermesi beklenmiştir. Bir önceki çalışmada olduğu gibi K-ortalamlar yönteminin, merkezleri taşınmasından dolayı kümeleme işleminde sapmalara neden olacağı düşünülmüştür. Bu durum kısmen gerçekleşmiş olsa da, beklenenin aksine 40 ve 50 küme sayısı için başarıyı bir miktar artırdığı gözlenmiştir. Bu sonuç, kesin ve objektif bir ideal küme sayısının olmadığı, optimal küme sayısının kullanılan kümeleme yöntemi ve elde edilen kümelerin nasıl kullanıldığı gibi pek çok etkene bağlı olarak değişebildiği kanısını doğrulamaktadır.

#### 4.2.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi

$f_0$  değerlerinin k-ortalamlar yöntemiyle kümeleneşinin üzerine yapılan bu son görüntü erişimi çalışmasında ise, tek değişkenli yaklaştırım değerleri esas alınarak ilgili kümeden görüntü erişimi yapılmıştır. Deney sonuçları Çizelge 4.6'da verildiği gibidir.

**Çizelge 4.6 :** K-ortalamlar yöntemiyle  $f_0$  değerlerini kümeleme ve tek değişkenli yaklaşımların değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 87.0	% 81.9	% 78.2	% 74.9	% 68.9
2	% 94.7	% 90.1	% 89.1	% 87.3	% 80.8
1	% 97.0	% 94.0	% 93.0	% 90.0	% 87.0

Bu çalışma sonuçları, görüntü erişiminin yine tek değişkenli yaklaşımların değerleri ile yapıldığı, fakat kümelemenin eşit aralık yöntemiyle yapıldığı önceki çalışma sonuçlarıyla karşılaştırıldığında beklendiği gibi 40 ve 50 küme sayısı için başarıyı biraz yükselttiği, diğer küme sayıları için biraz düşürdüğü gözlemlenmektedir. Diğer yandan tek değişkenli yaklaşımların değerleri ile görüntü erişimi yapıldığı için bu çalışma,  $f_0$  değerlerinin yine k-ortalamlar yöntemiyle kümelendiği, fakat görüntü erişiminin farklı verilerle yapıldığı yukarıdaki iki çalışmadan daha iyi sonuçlar vermektedir.

$f_0$  değerlerinin k-ortalamlar yöntemi ile kümelmesi ve ardından yapılan farklı görüntü erişimi çalışmalarıyla getirilen görüntülere örnekler, Şekil 4.7, 4.8 ve 4.9'da verilmiştir.

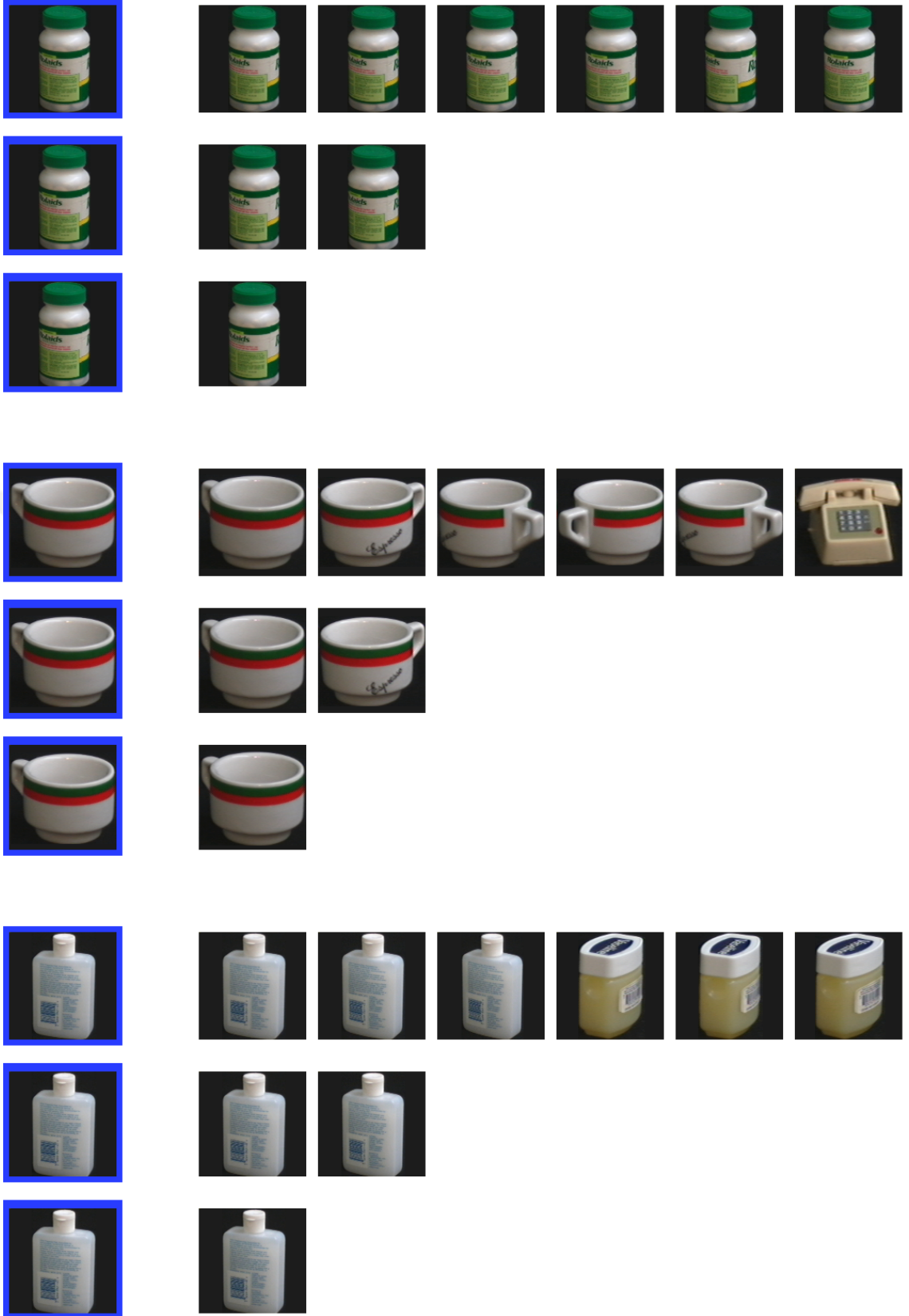
### 4.3 Eşit Üye Sayısı Yöntemi ile $f_0$ Değerlerini Kümeleme

Bu bölümdeki görüntü erişimi çalışmaları, eğitim görüntülerinin  $f_0$  değerleri esas alınarak eşit üye sayısı yöntemine göre kümelmesiyle elde edilen kümeler kullanılarak yapılmıştır. Buradaki çalışmalardan elde edilen başarı oranları, bu kümeleme yönteminin kümeleme başarısını gösterecektir.

Eğitim görüntüleri 7100 tane olduğundan tüm kümelere tamamen eşit sayıda görüntü atanmadığı durumlarda olabildiğince eşit atama yapabilmek için Bölüm 3.2.3'te anlatılan yuvarlama işleminden yararlanılmıştır. Örneğin, 30 kümeye ayırma çalışmasında,  $7100/30 = 236.7$  olduğundan, bu oran ve katları en yakın tamsayıya yuvarlanarak küme sınır değerleri belirlenmiş ve bu sayede kümelere sırasıyla 237, 236, 237, 237, 236, 237, ... tane görüntü atanmıştır.



**Şekil 4.7** : Çizelge 4.4 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



**Şekil 4.8 :** Çizelge 4.5 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



**Şekil 4.9** : Çizelge 4.6 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



### 4.3.1 $f_1$ değerleri ile görüntü erişimi

Eşit üye sayısı ile kümeleme yönteminin üzerine yapılan görüntü erişimi çalışmalarında da ilk olarak  $f_1$  değerleri kullanılmıştır. Sorgu görüntüsünün en yakın olduğu küme, kümeleme aşamasında elde edilen sınır değerlerine bakılarak belirlenmiş ve ardından sorgu görüntüsünün  $f_1$  vektörüne karesel Öklit uzaklığı ölçüsüne göre en yakın  $f_1$  vektörüne sahip görüntüler getirilmiştir. Getirilen görüntüler doğru objeye ait olup olmamasına ve getirilme sırasına göre azalan kümülatif kazanç yöntemiyle değerlendirilmiş, elde edilen başarı oranları Çizelge 4.7’de verilmiştir.

**Çizelge 4.7** : Eşit üye sayısı yöntemiyle  $f_0$  değerlerini kümeleme ve  $f_1$  değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 81.4	% 75.9	% 71.2	% 67.3	% 63.6
2	% 90.5	% 83.9	% 81.9	% 79.7	% 75.9
1	% 92.0	% 87.0	% 85.0	% 82.0	% 79.0

Bu çalışmada,  $f_1$  değerleri ile görüntü erişimi yapılan diğer çalışmalardan daha düşük sonuçlar elde edildiği görülmektedir. Hatta sonuçlar  $f_1$ ,  $f_2$  ve  $f_3$  vektörlerinin uç uca eklenmesiyle yapılan görüntü erişimi çalışmalarından da düşük çıkmıştır. Buradan, eşit üye sayısı yönteminin görüntüleri kümeleme başarısının önceki kümeleme yöntemlerine göre düşük olduğu anlaşılmaktadır. Bunun nedeni, k-ortalamar yönteminde etkili olan sapma probleminin burada daha belirgin olarak ortaya çıkıyor olmasıdır. Çok sayıda veri, dar bir skalada değerler almaktadır (7100 tane  $f_1$  verisi,  $[0, 255]$  aralığında) ve eşit üye sayısı yöntemi verilerin yoğunlaştığı yerlerde k-ortalamar yönteminden de daha fazla seçici davranarak aynı objeye ait görüntüleri farklı kümelere atamaktadır. Bu özelliğin, bu kümeleme yöntemiyle yapılan diğer görüntü erişimi çalışmalarında da başarıyı düşürmesi beklenmektedir.

### 4.3.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi

Bu çalışmada eşit üye sayısı yöntemiyle kümelenen eğitim görüntüleri üzerinde,  $f_1$ ,  $f_2$  ve  $f_3$ ’ün uç uca eklenmesiyle elde edilen vektör kullanılarak görüntü erişimi yapılmıştır. Deneylerden elde edilen başarı oranları Çizelge 4.8’de verilmiştir.

**Çizelge 4.8 :** Eşit üye sayısı yöntemiyle  $f_0$  değerlerini kümeleme ve  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 78.4	% 74.8	% 69.3	% 65.9	% 63.2
2	% 88.3	% 86.1	% 80.7	% 78.4	% 74.4
1	% 91.0	% 90.0	% 83.0	% 83.0	% 81.0

Bu çalışmada sonuçların,  $f_1$  verisi ile görüntü erişimi yapılan bir önceki çalışmaya göre, beklendiği gibi, genel olarak bir miktar düştüğü görülmektedir. Ancak getirilen görüntülerde yalnız birincisinin değerlendirildiği nadir durumlarda başarı bir miktar yükselmiştir. Diğer yandan bu çalışma, görüntü erişiminin buradaki gibi  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle yapıldığı, fakat kümelemenin eşit üye sayısı yöntemi yerine diğer yöntemlerle yapıldığı çalışmalarla karşılaştırıldığında, bu çalışmanın başarı oranlarının daha düşük olduğu gözlenmektedir. Bu durum bu tezdeki çalışmalar için, eşit üye sayısının kümeleme başarısının diğer kümeleme yöntemlerinden düşük olduğu çıkarımını güçlendirmektedir.

#### 4.3.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi

Eşit üye sayısı yöntemiyle  $f_0$  değerlerinin kümeleneceği basamağının üzerine yapılan bu son görüntü erişim çalışmasında, tek değişkenli yaklaştırım değerleri kullanılmıştır. Hesaplanan başarı oranları Çizelge 4.9'da verilmiştir.

**Çizelge 4.9 :** Eşit üye sayısı yöntemiyle  $f_0$  değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 83.2	% 78.7	% 73.7	% 69.3	% 66.7
2	% 90.9	% 89.7	% 84.4	% 81.8	% 78.0
1	% 94.0	% 94.0	% 89.0	% 88.0	% 85.0

Görüntü erişimi tek değişkenli yaklaştırım değerleri ile yapıldığından bu çalışma, görüntü erişiminde diğer verilerin kullanıldığı önceki çalışmalara kıyasla daha başarılı sonuçlar vermiştir. Fakat görüntü erişimi için aynı verinin kullanıldığı, kümelemenin ise farklı yöntemlerle yapıldığı çalışmalarla karşılaştırıldığında eşit üye sayısı yönteminden dolayı daha başarısız sonuçlar elde edilmiştir.

$f_0$  değerlerinin eşit üye sayısı yöntemi ile kümeleneceği ve ardından yapılan farklı görüntü erişimi çalışmalarıyla getirilen görüntülere örnekler, Şekil 4.10, 4.11 ve 4.12'de verilmiştir.

Buraya kadar verilen çalışmalarda, en başarılı kümelemenin eşit aralık yöntemi ile yapıldığı görülmüştür. K-ortalamlar yöntemiyle kümeleme yapıldığında da oldukça başarılı sonuçlar elde edilmiştir. Eşit üye sayısı yöntemi ise, diğer iki kümeleme yöntemine göre daha seçicidir ve dolayısıyla daha başarısız bir kümeleme yapmaktadır. Bu yüzden eşit üye sayısı yöntemi sonraki çalışmalarda terk edilmiştir. İlerleyen bölümlerde öncelikle  $f_3$  değerlerinin k-ortalamlar yöntemiyle kümeleneceği, ardından üç değer içeren  $f_3$  vektörünün her bir değeri ayrı ayrı ele alınarak eşit aralık yöntemiyle kümeleneceği çalışması yapılacaktır.

#### **4.4 K-ortalamlar Yöntemi ile $f_3$ Değerlerini Kümeleme**

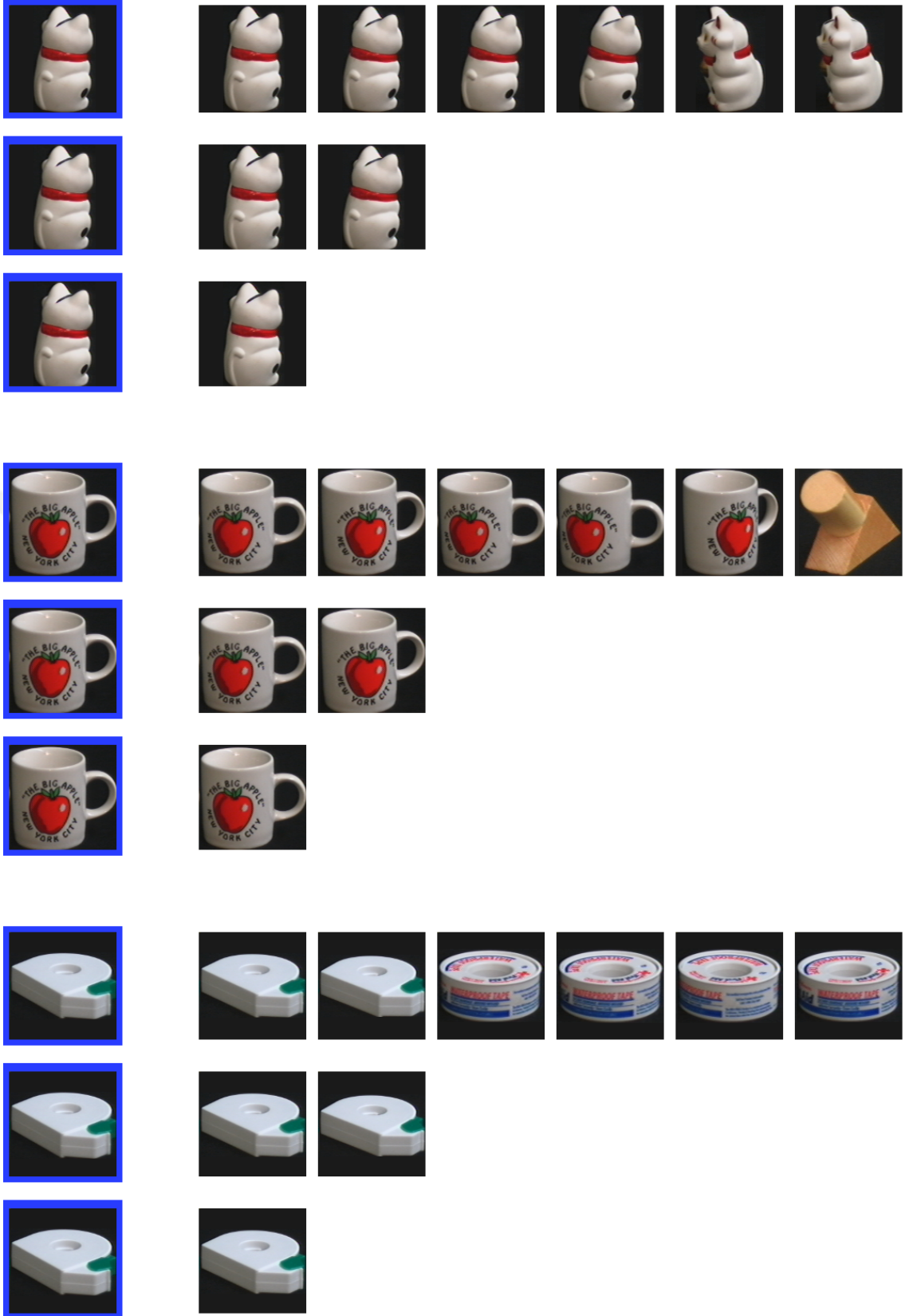
Bir üç boyutlu dizi olan görüntü verisine YBMG ayrıştırımı uygulandığında elde edilen  $f_3$  terimi üç tane değer içermektedir. Bu değerler, sırayla kırmızı(R), yeşil(G) ve mavi(B) renk matrislerinin taşıdığı değerlerin aritmetik ortalamalarıdır. Bu bölümde, bu  $f_3$  üçlülere k-ortalamlar yöntemiyle kümeleneceği. K-ortalamlar yönteminin uygulanabilmesi için ihtiyaç duyulan başlangıç merkez değerleri,  $f_3$  üçlülerinin R, G ve B matrislerine karşılık gelen değerleri ayrı ayrı değerlendirilerek, Denklem 3.7'deki gibi hesaplanmıştır.  $f_3$  üçlülerden oluştuğu için bu denklemde  $p = 3$ 'tür ve dolayısıyla  $l = 1, 2, 3$  değerlerini almaktadır. Burada, başlangıç merkezleri bu kurala göre hesaplanarak,  $f_3$  vektörleri k-ortalamlar yöntemiyle kümeleneceği. Bu kümeleme çalışmasının bütüncül algoritması Şekil 3.2'de, k-ortalamlar yönteminin algoritma tasarımına örnek olarak verilmiştir. Bu algoritmayla 20, 30, 40, 50 ve 60 küme sayıları için ayrı ayrı kümeleme çalışmaları yapılmıştır. Kümelemenin ardından yapılan görüntü erişim çalışmaları aşağıda sırayla verilecektir.

##### **4.4.1 $f_1$ değerleri ile görüntü erişimi**

Bu çalışmada öncelikle, k-ortalamlar yöntemiyle yapılan kümeleme işleminde ulaşılan son merkez değerleri esas alınarak, ele alınan bir sorgu görüntüsünün hangi kümeye daha yakın olduğu tespit edilmiştir. Ardından sorgu görüntüsü ile, bu sorgu görüntüsünün en yakın olduğu kümede bulunan eğitim görüntülerinin  $f_1$  değerleri



**Şekil 4.10** : Çizelge 4.7 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



Şekil 4.11 : Çizelge 4.8 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



**Şekil 4.12 :** Çizelge 4.9 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).

arasındaki karesel Öklit uzaklığı hesaplanmış ve en düşük uzaklıkları veren görüntüler getirilmiştir. Bu görüntü erişim deneylerinden elde edilen başarı yüzdeleri Çizelge 4.10'da verilmiştir.

**Çizelge 4.10** : K-ortalamlar yöntemiyle F3 değerlerini kümeleme ve  $f_1$  değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 94.5	% 95.1	% 94.1	% 95.7	% 94.5
2	% 96.8	% 98.5	% 98.1	% 98.8	% 98.6
1	% 98.0	% 100.0	% 100.0	% 100.0	% 99.0

Bu çalışmanın önceki çalışmaların hepsinden daha başarılı sonuçlar verdiği ve başarı oranlarındaki artışın oldukça büyük olduğu görülmektedir. Ayrıca bu başarı oranları nicel olarak da tatmin edici boyuttadır. Başarı oranlarındaki bu yükselmenin sebebi kümeleme işleminde kullanılan  $f_3$  vektörüdür. Bu vektörün, görüntünün boyutunu çok fazla indirgemesine rağmen, kümeleme işlemi sırasında görüntüleri başarılı bir şekilde temsil ettiği anlaşılmaktadır. Ayrıca  $f_3$  verisi ile kümeleme yapılan bu çalışmanın diğer bir avantajı, küme sayısının artmasının başarı oranlarında bir düşüşe neden olmamasıdır. Yani, kümelemede kullanılan  $f_3$  verisinin görüntüleri temsil başarısı, k-ortalamlar yönteminin  $f_0$  verisi üzerinde uygulandığı çalışmada görülen yüksek küme sayısının olumsuz etkisini ortadan kaldırmıştır.

#### 4.4.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi

$f_3$  vektörlerini k-ortalamlar yöntemiyle kümeleme çalışmasının sonuçları üzerinde yapılan bu ikinci görüntü erişimi çalışmasında,  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle elde edilen vektör kullanılmıştır. Bu vektör esas alınarak yapılan görüntü erişimi deneylerinin sonuçları Çizelge 4.11'de verilmiştir.

**Çizelge 4.11** : K-ortalamlar yöntemiyle  $f_3$  değerlerini kümeleme ve  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 94.4	% 94.8	% 93.4	% 94.9	% 94.7
2	% 98.6	% 99.0	% 98.6	% 99.0	% 99.2
1	% 99.0	% 99.0	% 99.0	% 99.0	% 100.0

Bu çalışmadan elde edilen sonuçlar  $f_1$  değerleri ile görüntü erişimi yapılan bir önceki çalışma sonuçları ile karşılaştırıldığında kayda değer bir düşüş görülmemiştir. Tam

tersine, küme sayısının 60 olduğu durumda bir yükseliş gözlenmiştir. Kümelemede  $f_3$  değerlerinin kullanılmasının getirdiği küme sayısının yüksek olmasına karşı dayanıklılık avantajı, daha güçlü olarak burada da görülmektedir. Küme sayısının büyük seçilebilmesi ise, bu tez çalışmasına algoritmaları hızlandırması için dahil edilen kümeleme yaklaşımının verimliliğini artırmaktadır. Küme sayısı büyük seçildiğinde kümelerdeki görüntü sayısı azaldığından görüntü erişimi işlemi hızlanmaktadır.

#### 4.4.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi

Bu görüntü erişim çalışmasında, ele alınan bir sorgu görüntüsünün, kendi hedef kümesindeki görüntülere olan uzaklıklarının tespitinde tek değişkenli yaklaştırım değerleri esas alınmıştır. Yapılan deneysel çalışmalardan elde edilen başarı oranları Çizelge 4.12'deki gibidir.

**Çizelge 4.12** : K-ortalamlar yöntemiyle  $f_3$  değerlerini kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

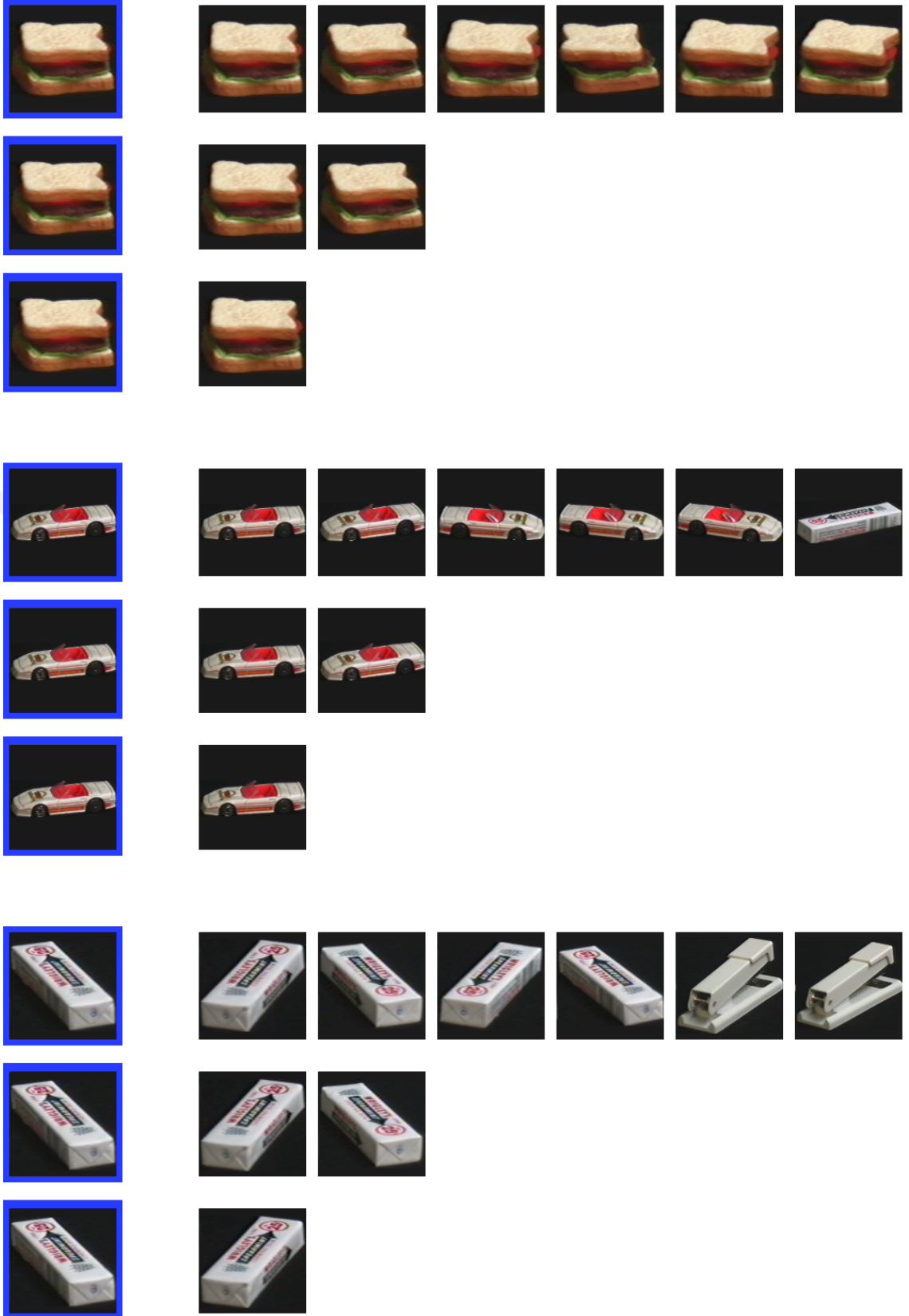
	20 küme	30 küme	40 küme	50 küme	60 küme
6	% 96.2	% 96.5	% 94.4	% 96.3	% 95.5
2	% 98.6	% 99.6	% 99.2	% 99.0	% 99.6
1	% 99.0	% 100.0	% 100.0	% 99.0	% 100.0

Bu çalışmada başarı oranlarının, aynı kümeleme sonuçlarının kullanıldığı önceki iki görüntü erişimi çalışmasına göre, beklendiği gibi bir miktar yükseldiği görülmektedir.  $f_3$  değerlerinin k-ortalamlar yöntemi ile kümeleneceği durumunda, yapılan farklı görüntü erişimi çalışmalarının örnek görüntüleri, Şekil 4.13, 4.14 ve 4.15'te verilmiştir.

#### 4.5 Eşit Aralık Yöntemi ile $f_3$ -RGB Değerlerini Kümeleme

Bu kümeleme çalışmasında, görüntülerin  $f_3$  üçlülerinin her bir değeri ayrı ayrı ele alınarak eşit aralık yöntemiyle kümelenecektir. Dolayısıyla bu kümeleme çalışması, yalnızca 20, 30, 40, 50 ve 60 küme sayıları için toplam beş kez değil,  $f_3$  vektörünün R, G ve B değerleri de birbirinden ayrı olarak ele alındığından toplam 15 kez yapılmıştır. Bu yüzden bu kümeleme işleminin sonuçları üzerinde çalışılan her bir görüntü erişimi çalışması da 15 ayrı durum için yapılacaktır.





Şekil 4.13 : Çizelge 4.10 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



Şekil 4.14 : Çizelge 4.11 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



Şekil 4.15 : Çizelge 4.12 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).

#### 4.5.1 $f_1$ değerleri ile görüntü erişimi

Önceki çalışmalara benzer şekilde, bu kümeleme yönteminin sonuçları kullanılarak yapılan ilk görüntü erişim çalışmasında da görüntülerin  $f_1$  değerleri esas alınmıştır. 15 ayrı kümeleme çalışması üzerinde yapılan bu görüntü erişim çalışmalarının başarı oranları Çizelge 4.13'te verilmiştir.

**Çizelge 4.13** : Eşit aralık yöntemiyle  $f_3$ -RGB değerlerini ayrı ayrı kümeleme ve  $f_1$  değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

		20 küme	30 küme	40 küme	50 küme	60 küme
R	6	% 93.7	% 93.2	% 92.1	% 89.4	% 88.3
	2	% 98.8	% 97.1	% 97.1	% 95.2	% 94.5
	1	% 100.0	% 99.0	% 99.0	% 96.0	% 96.0
G	6	% 94.6	% 93.7	% 92.4	% 92.7	% 91.1
	2	% 98.2	% 96.8	% 97.2	% 97.6	% 95.7
	1	% 99.0	% 98.0	% 98.0	% 98.0	% 98.0
B	6	% 91.3	% 90.7	% 88.9	% 88.1	% 86.3
	2	% 95.8	% 95.6	% 93.4	% 92.2	% 92.3
	1	% 97.0	% 96.0	% 93.0	% 93.0	% 95.0

Eğitim görüntülerini kümelemede iyi bir performans gösterdiği daha önce de tespit edilen  $f_3$  teriminin, R, G ve B değerleri ayrı ayrı kullanıldığında da benzer bir başarıyı yakaladığı görülmektedir.  $f_3$  vektörünün tek bir değeri kullanılarak kümeleme yapılmasına rağmen başarılı sonuçlar vermesinin, görüntünün renk özneliğini yansıttığından kaynaklandığı düşünülmektedir. Nitekim  $f_0$  terimi de tek değer içermektedir fakat buradaki çalışmanın başarısı Bölüm 4.1.1'de elde edilen başarının oldukça üzerindedir.

#### 4.5.2 Tek değişkenli terimlerin uç uca eklenmesi ile görüntü erişimi

$f_3$ -RGB değerlerinin eşit aralık yöntemiyle ayrı ayrı kümeleneşinin üzerine yapılan bu ikinci görüntü erişim çalışmasında  $f_1, f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle elde edilen vektör kullanılmıştır. Yapılan deneylerden elde edilen başarı oranları Çizelge 4.14'te verilmiştir.

Bu çalışma sonuçları bir önceki çalışmayla karşılaştırıldığında net bir başarı farkı ortaya çıkmamaktadır. Bazı durumlarda başarıda küçük bir düşüş, bazı durumlarda

**Çizelge 4.14** : Eşit aralık yöntemiyle  $f_3$ -RGB değerlerini ayrı ayrı kümeleme ve  $f_1, f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

		20 küme	30 küme	40 küme	50 küme	60 küme
R	6	% 92.8	% 92.8	% 91.5	% 89.3	% 87.0
	2	% 97.8	% 97.8	% 97.5	% 95.6	% 94.5
	1	% 99.0	% 99.0	% 99.0	% 96.0	% 96.0
G	6	% 94.2	% 93.3	% 92.1	% 91.9	% 91.2
	2	% 100.0	% 99.0	% 97.8	% 98.2	% 96.1
	1	% 100.0	% 99.0	% 99.0	% 99.0	% 98.0
B	6	% 92.4	% 91.6	% 90.0	% 89.3	% 87.1
	2	% 97.8	% 97.5	% 95.5	% 94.5	% 91.1
	1	% 99.0	% 99.0	% 97.0	% 96.0	% 95.0

küçük bir yükseliş gözlemlenmekte, bazı durumlarda da sabit kalmaktadır. Bunun, oldukça yüksek bir başarı yakalandıktan sonra geriye kalan küçük hataların, verilerin ayırt edici özellikleri sebebiyle oluşmasından çok, rastgele oluşmasından kaynaklandığı düşünülmektedir.

#### 4.5.3 Tek değişkenli yaklaştırım değerleri ile görüntü erişimi

Bu son çalışmada görüntü erişimi, tek değişkenli yaklaştırım değerleri esas alınarak yapılmıştır. 100 sorgu görüntüsü üzerinde yapılan deneylerle getirilen görüntülerin azalan kümülatif kazanç yöntemiyle değerlendirilmesiyle elde edilen başarı yüzdeleri Çizelge 4.15'te verilmiştir.

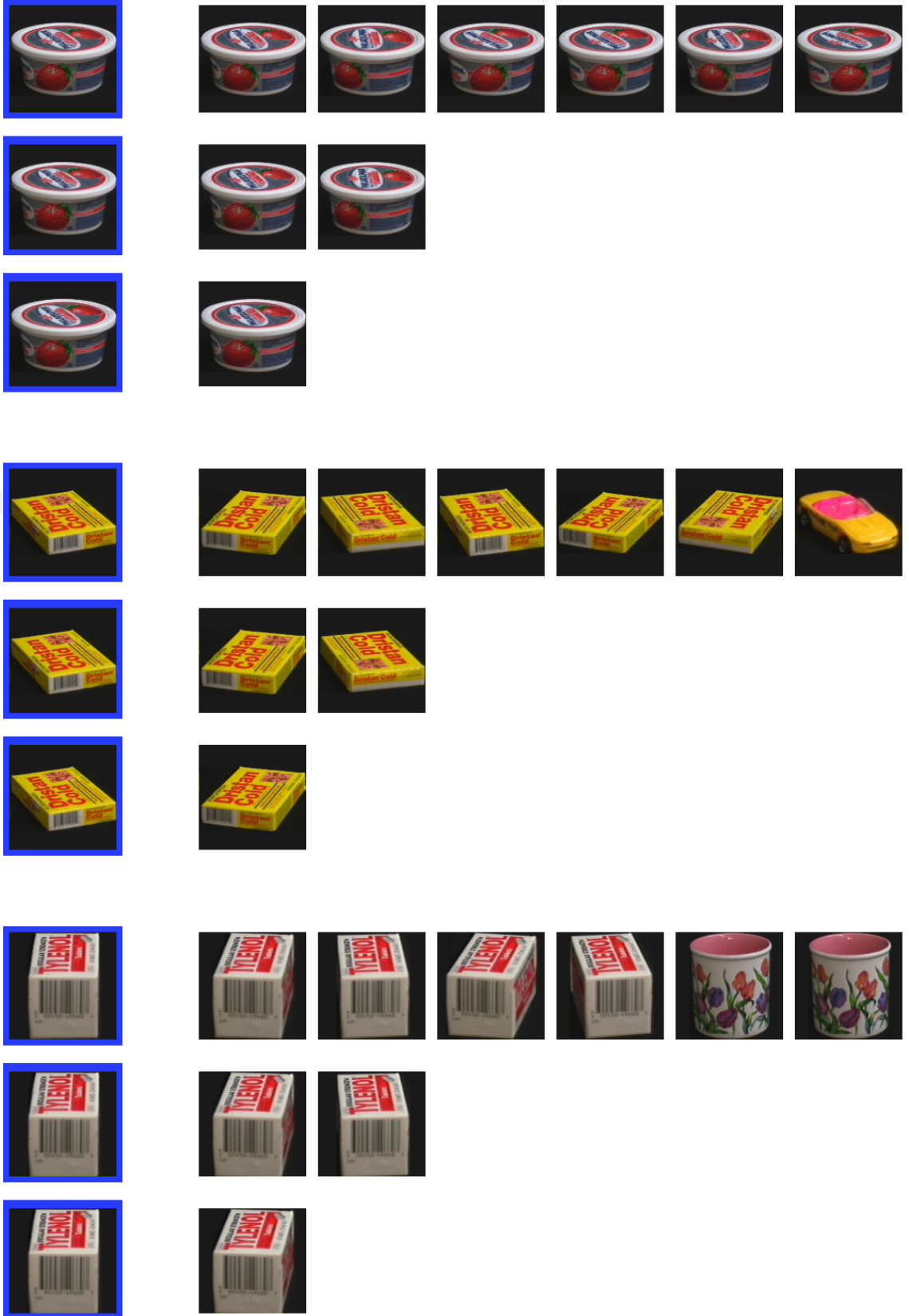
**Çizelge 4.15** : Eşit aralık yöntemiyle  $f_3$ -RGB değerlerini ayrı ayrı kümeleme ve tek değişkenli yaklaştırım değerleriyle ilgili kümeden görüntü erişimi deneyinin başarı oranları.

		20 küme	30 küme	40 küme	50 küme	60 küme
R	6	% 94.5	% 94.5	% 92.2	% 90.6	% 87.9
	2	% 98.8	% 98.1	% 97.3	% 95.8	% 94.6
	1	% 100.0	% 100.0	% 100.0	% 97.0	% 95.0
G	6	% 95.4	% 94.5	% 93.3	% 93.7	% 91.6
	2	% 100.0	% 99.2	% 98.8	% 99.2	% 96.1
	1	% 100.0	% 100.0	% 100.0	% 100.0	% 98.0
B	6	% 94.1	% 93.2	% 90.9	% 89.6	% 87.7
	2	% 97.8	% 98.2	% 97.2	% 94.5	% 93.7
	1	% 99.0	% 99.0	% 98.0	% 96.0	% 96.0

Bu çalışma sonuçlarının, beklendiği gibi, aynı kümeleme işlemi üzerinde yapılan önceki iki görüntü erişimi çalışmasının sonuçlarından daha iyi çıktığı görülmektedir. Bu çalışmayla birlikte daha belirginleşmiştir ki özniteliklerden yeşil renk (G), COIL-100 veri tabanındaki görüntüleri diğer renklere göre biraz daha iyi temsil etmektedir. Sonuç olarak,  $f_3$  verisinin R, G ve B değerlerinin ayrı ayrı ele alınmasıyla yapılan kümeleme çalışmasının üzerine yapılan bu görüntü erişim çalışmalarında oldukça yüksek başarı oranları elde edilmiştir.  $f_3$  vektörünün tek bir değeri esas alındığında,  $f_0$  verisinde olduğu gibi tek boyutlu bir veri ile çalışılıyor olmasına rağmen, renk özneliğinin de kısmen temsil edilebilmesi başarı oranlarını önemli oranda artırmıştır.

$f_3$  vektörünün her bir değerinin ayrı ayrı eşit aralık yöntemi ile kümelmesi ve farklı verilerle görüntü erişimi yapılması çalışmalarına örnek görüntüler, Şekil 4.16, 4.17 ve 4.18'de verilmiştir.

Bu tezde önerilen özgün yöntemlerin anlatımı, sonuçları ve yorumlanması burada son bulmaktadır. Ulaşılan sonuçlar, eşit aralık ve k-ortalamar kümeleme yöntemlerinin, eşit üye sayısı yöntemine göre daha yüksek bir başarı gösterdiği,  $f_3$  terimiyle  $f_0$  teriminden daha iyi kümeleme yapılabildiği ve  $f_1$  teriminin görüntü erişimi amacıyla boyut indirgemedede oldukça başarılı olduğudur. Bu kısımda anlatılmış olan bu özgün yöntemlerin genel başarısını ortaya koyabilmek için ise, önerilen yöntemlerin başka yöntemlerle karşılaştırılması gerekmektedir. Bu karşılaştırmalar sonraki kısımda yer alacaktır.



Şekil 4.16 : Çizelge 4.13 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



Şekil 4.17 : Çizelge 4.14 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).





Şekil 4.18 : Çizelge 4.15 deney sonuçlarından farklı başarı derecelerinde 3 örnek. Solda sorgu görüntüsü, sağda getirilen görüntüler (6, 2 veya 1 tane).



## 5. KARŞILAŞTIRMA

Önceki kısımlarda anlatıldığı gibi bu tez çalışmasında, YBMG veri ayrıştırımı kullanılarak görüntü erişimi öncesi görüntü kümeleme ve ardından ilgili kümeden görüntü erişimi yapılmasını kapsayan çeşitli yöntemler önerilmiş ve bu yöntemler deneysel çalışmalarla sınanmıştır. Elde edilen sonuçlar kayda değer bir başarı ortaya koymuştur. Bu kısımda ise, yöntemlerin başarısı ve hesaplama zamanları farklı yöntemlerle karşılaştırılacaktır. Böylece, öne sürülen yöntemler karşılaştırmalı olarak da değerlendirilmiş olacaktır.

İlk olarak görüntü erişimi öncesi görüntüleri kümelemenin sağladığı avantajı ölçmek amacıyla, önerilen yöntemler kümeleme basamağının bulunmadığı yöntemlerle karşılaştırılacaktır. Karşılaştırma yapılan yöntemlerde de YBMG ayrıştırımı kullanılacak, tek fark kümeleme yapılmaması olacaktır. Bu karşılaştırma, başarı oranlarındaki farkı da ortaya koyacak fakat en önemli karşılaştırma, hesaplama süreleri açısından yapılacaktır. İkinci bölümde ise YBMG ayrıştırımının görüntü erişimi performansını değerlendirebilmek için, bu ayrıştırım yöntemi ile Temel Bileşen Analizi yöntemi (TBA, Principal Component Analysis, PCA) karşılaştırılacaktır. Bu karşılaştırma, hem hesaplama zamanları hem başarı oranları açısından önemlidir.

Önceki kısımlarda olduğu gibi burada da, üç boyutlu dizi olan RGB formatındaki bir görüntünün *düşey piksel sayısı*,  $n_1$  ile ve *yatay piksel sayısı*,  $n_2$  ile temsil edilmektedir. *renk sayısı* ise 3'tür. Karşılaştırma amacıyla öne sürülen buradaki yöntemlerde de deneysel çalışmalar, özgün yöntemlerde olduğu gibi COIL-100 görüntü veri tabanı üzerinde yapılmıştır. Bu veri tabanındaki görüntüler için,  $n_1 = 128$  ve  $n_2 = 128$ 'dir.

### 5.1 YBMG Yönteminin Kümeleme Yapılması ve Kümeleme Yapılmaması Durumlarındaki Görüntü Erişimi Performansının Karşılaştırılması

Bu tez çalışmasında temelde, YBMG yönteminin görüntü erişimi için kullanılması incelenmiştir. Çalışmanın özgün yanı, görüntü erişimi öncesinde veri tabanındaki görüntülerin kümelenebilmesidir. Böylece, görüntü erişimi sadece ilgili küme içinden

yapılmakta ve bu durum zaman kazancı sağlamaktadır. Bu kazancı ölçmek için önerilen yöntemler, görüntü kümeleme basamağının atlanıp doğrudan görüntü erişimi çalışmasının yapıldığı başka çalışmalar ile karşılaştırılacaktır. Sağlıklı bir karşılaştırma olması için diğer tüm değişkenler sabit tutulmuştur. Örneğin, önerilen yöntemlerde olduğu gibi, kümeleme yapılmayan buradaki yöntemlerde de, YBMG ayrıştırımıyla elde edilen veriler kullanılmıştır. Kümeleme yapılmayan birinci yöntemin tasarımı Şekil 5.1'deki gibidir.

```

1 Sorgu görüntüsünü oku ve  $f_1$  vektörünü hesapla;
2 forall Eğitim görüntüleri do
3   | Eğitim görüntüsünü oku,  $f_1$  vektörünü hesapla;
4   | Eğitim görüntüsünün  $f_1$  vektörü ile test görüntüsünün  $f_1$  vektörü arasındaki
   | karesel Öklit uzaklığını hesapla;
5   | Uzaklık bilgisini sakla;
6 end
7 Uzaklıkları küçükten büyüğe sırala;
8 En küçük  $S$  tane uzaklığı sıralı olarak al;
9 forall En küçük  $S$  tane uzaklık do
10  | Uzaklığın hangi eğitim görüntüsünden elde edildiğini bul;
11  | if Bu eğitim görüntüsü ile sorgu görüntüsü aynı objeye aitse then
12  |   | En küçük kaçınıcı uzaklık olduğuna göre azalan kümülatif kazanca
   |   | katkısını hesapla ve kazanca ekle;
13  | end
14 end
15 return Kazancı ve hesaplama süresini geri döndür;

```

**Şekil 5.1** : Kümeleme yapılmaksızın YBMG- $f_1$  değerleriyle görüntü erişimi algoritması.

Algoritmada bulunan  $S$ , en yakın kaç tane görüntünün getirilmek istendiğidir. Bu çalışmada, özgün çalışmada olduğu gibi en yakın 6, 2 ve 1 tane görüntü getirilmesi durumları ele alınmış ve bu durumlar için başarı oranları hesaplanmıştır. Sonuçlar Çizelge 5.1'de verildiği gibidir.

**Çizelge 5.1** : Kümeleme yapılmaksızın YBMG- $f_1$  değerleriyle görüntü erişimi deneyinin başarı oranları.

	Başarı oranı
6	% 90.8
2	% 94.1
1	% 96.0

Bu tez çalışmasında önerilen özgün yöntemlerde görüntü erişimi çalışmaları sadece  $f_1$  değerleri ile yapılmamış, tek değişkenli terimlerin uç uca eklenmesiyle elde

edilen vektör ve ayrıca tek değişkenli yaklaştırım verileri de incelenmiştir. Tam bir karşılaştırma ortaya koyabilmek için, bu verilerle de kümeleme basamağının bulunmadığı görüntü erişimi çalışmaları yapılmıştır. Bu yöntemlerin algoritması, görüntü erişimde kullanılan verilerin değiştirilmesi kaydıyla, yukarıdaki algoritmaya benzer şekilde tasarlanmıştır. Elde edilen başarı oranları Çizelge 5.2 ve 5.3'te verilmiştir.

**Çizelge 5.2 :** Kümeleme yapılmaksızın  $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi deneyinin başarı oranları.

Başarı oranı	
6	% 93.1
2	% 98.6
1	% 99.0

**Çizelge 5.3 :** Kümeleme yapılmaksızın YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi deneyinin başarı oranları.

Başarı oranı	
6	% 96.0
2	% 98.6
1	% 99.0

Bu çizelgelerde görüldüğü gibi YBMG verileri kullanılarak yapılan görüntü erişimi çalışmaları, kümeleme yapılmadığında da oldukça başarılı sonuçlar vermektedir. Bu başarı oranları kümeleme yapılan yöntemlerin başarı oranları ile karşılaştırıldığında daha başarılı ve daha başarısız olduğu iki farklı durum ile karşılaşılmaktadır.

Kümelemenin YBMG- $f_0$  değerleri kullanılarak yapıldığı çalışmalarda, kümeleme yapılmayan buradaki çalışmalardan daha düşük sonuçlar elde edilmiştir. Küme sayısı arttıkça başarı oranlarındaki kayıp da giderek arttığından, en iyi değerleri veren 20 küme sayısı esas alınarak nicel karşılaştırma yapılması tercih edilmiştir. Görüntü erişimin  $f_1$  değerleri ile yapıldığı çalışmalar Çizelge 5.4'te karşılaştırılmıştır. Görüntü erişimin tek değişkenli terimlerin uç uca eklenmesi ile yapıldığı çalışmaların karşılaştırması Çizelge 5.5'te ve görüntü erişimin tek değişkenli yaklaştırım değerleri ile yapıldığı çalışmaların karşılaştırması ise Çizelge 5.6'da verilmiştir.

Bu karşılaştırmalardan görüntü kümeleme işleminin bir miktar başarı kaybına neden olduğu anlaşılmaktadır. Ancak hesaplama zamanları karşılaştırmasına gelindiğinde kümeleme işleminin bu çalışmalarda yaklaşık 14 kat kazanç sağladığı görülecektir.

**Çizelge 5.4 :** Kümeleme basamağı bulunmayan çalışmalar ile  $f_0$  değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1$  değerleriyle görüntü erişimi).

	Kümeleme yok	Eşit aralık	K-ortalamlar	Eşit üye sayısı
6	% 90.8	% 87.1	% 84.9	% 81.4
2	% 94.1	% 92.9	% 92.1	% 90.5
1	% 96.0	% 96.0	% 94.0	% 92.0

**Çizelge 5.5 :** Kümeleme basamağı bulunmayan çalışmalar ile  $f_0$  değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi).

	Kümeleme yok	Eşit aralık	K-ortalamlar	Eşit üye sayısı
6	% 93.1	% 84.3	% 82.9	% 78.4
2	% 98.6	% 94.7	% 93.7	% 88.3
1	% 99.0	% 97.0	% 96.0	% 91.0

**Çizelge 5.6 :** Kümeleme basamağı bulunmayan çalışmalar ile  $f_0$  değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması (YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi).

	Kümeleme yok	Eşit aralık	K-ortalamlar	Eşit üye sayısı
6	% 96.0	% 88.4	% 87.0	% 83.2
2	% 98.6	% 96.1	% 94.7	% 90.9
1	% 99.0	% 98.0	% 97.0	% 94.0

Diğer yandan kümeleme her zaman başarı kaybına neden olmamaktadır. Kümeleme işleminin  $f_3$  değerleri ile yapıldığı çalışmalar, kümeleme yapılmayan buradaki çalışmadan daha başarılı sonuçlar vermektedir. Yani bu çalışmalarda kümeleme yaklaşımı, düşük hesaplama zamanı avantajının yanı sıra başarı oranlarını da yükseltmiştir. Çizelge 5.7, 5.8 ve 5.9'da başarı oranları nicel olarak karşılaştırılmıştır. Hesaplama zamanları karşılaştırıldığında ise kümeleme işleminin ortalama 5-7 kat kazanç sağladığı görülecektir.

**Çizelge 5.7 :** Kümeleme basamağı bulunmayan çalışmalar ile  $f_3$  değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1$  değerleriyle görüntü erişimi).

	Küm. yok	K-ortala.	EşitAra-R	EşitAra-G	EşitAra-B
6	% 90.8	% 94.5	% 93.7	% 94.6	% 91.3
2	% 94.1	% 96.8	% 98.8	% 98.2	% 95.8
1	% 96.0	% 98.0	% 100.0	% 99.0	% 97.0

Bu çizelgelerden, görüntü kümeleme işleminin başarı oranlarında genel olarak bir artış getirdiği görülmektedir. Aslen hesaplama zamanından kazanç sağlamak için yapılan

**Çizelge 5.8 :** Kümeleme basamağı bulunmayan çalışmalar ile  $f_3$  değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması ( $f_1$ ,  $f_2$  ve  $f_3$ 'ün uç uca eklenmesiyle görüntü erişimi).

	Küm. yok	K-ortala.	EşitAra-R	EşitAra-G	EşitAra-B
6	% 93.1	% 94.4	% 92.8	% 94.2	% 92.4
2	% 98.6	% 98.6	% 97.8	% 100.0	% 97.8
1	% 99.0	% 99.0	% 99.0	% 100.0	% 99.0

**Çizelge 5.9 :** Kümeleme basamağı bulunmayan çalışmalar ile  $f_3$  değerlerinin 20 kümeye ayrıldığı çalışmalar karşılaştırması (YBMG tek değişkenli yaklaştırım değerleriyle görüntü erişimi).

	Küm. yok	K-ortala.	EşitAra-R	EşitAra-G	EşitAra-B
6	% 96.0	% 96.2	% 94.5	% 95.4	% 94.1
2	% 98.6	% 98.6	% 98.8	% 100.0	% 97.8
1	% 99.0	% 99.0	% 100.0	% 100.0	% 99.0

kümeleme müdahalesinin beklenenin aksine başarıyı da yükseltmesinin nedeninin, YBMG- $f_0$ 'a kıyasla daha fazla bilgi içeren ve görüntünün renk özneliğini de yansıtan YBMG- $f_3$  gibi bir veri ile kümeleme yapılıyor olması ve dolayısıyla yapılan iyi bir kümeleme işleminin veri tabanındaki gürültüyü azaltması olduğu düşünülmektedir.

Önerilen yöntemler ile kümeleme basamağının atlandığı buradaki yöntemler, başarı oranları açısından ayrıntılı olarak karşılaştırılmış bulunmaktadır. Fakat daha önemli bir karşılaştırma konusu algoritmaların çalışma süreleridir. Çünkü kümeleme işlemi -temelde- zaman kazancı sağlamak için önerilmiştir. Nicel bir karşılaştırma yapabilmek ve zaman kazancını ortaya koyabilmek için, algoritmaların çalışma süreleri saniye cinsinden incelenmiştir. Önerilen yöntemlerin hesaplama süreleri Çizelge 5.10'da, kümeleme yapılmaksızın görüntü erişimi yöntemlerinin hesaplama süreleri ise Çizelge 5.11'de verilmiştir.

Burada saniye cinsinden verilen zaman değerleri tek bir görüntü için yapılan görüntü erişimi deneyinin süresini ifade etmektedir. Bu değerler, 100 sorgu görüntüsünün her biri ile yapılan deneylerin süre ortalaması alınarak elde edilmiştir. Başarı oranlarının, getirilen görüntülerden 6, 2 ve 1 tanesinin dikkate alınması durumuna göre verilmiş olması, hesaplama sürelerini değiştiren bir etken değildir, bu yüzden her bir çalışma için tek bir saniye değeri verilmiştir.  $f_3$  vektörünün R, G ve B değerleri ayrı ayrı ele alınarak kümeleme ve ardından görüntü erişimi yapılan çalışmalar için ise, bu üçünün süre ortalamaları verilmiştir.

**Çizelge 5.10 :** YBMG verileriyle kümeleme yapılması durumunda ilgili kümeden görüntü erişimi deneylerinin çalışma süreleri (saniye).

Bölüm	20 küme	30 küme	40 küme	50 küme	60 küme
4.1.1	1.34	0.93	0.74	0.61	0.44
4.1.2	2.17	1.04	0.71	0.57	0.45
4.1.3	2.42	1.57	1.13	0.89	0.75
4.2.1	0.90	0.71	0.59	0.55	0.49
4.2.2	3.29	1.14	0.73	0.59	0.46
4.2.3	1.97	1.45	1.06	0.87	0.71
4.3.1	0.80	0.57	0.39	0.30	0.25
4.3.2	0.95	0.62	0.49	0.39	0.33
4.3.3	1.52	0.94	0.69	0.56	0.46
4.4.1	2.12	1.51	1.18	1.18	1.12
4.4.2	3.12	1.90	1.49	1.48	1.39
4.4.3	5.11	3.48	2.79	2.82	2.46
4.5.1	2.01	1.36	1.06	0.91	0.75
4.5.2	2.86	2.05	1.55	1.21	0.94
4.5.3	2.85	2.87	2.82	2.18	2.36

**Çizelge 5.11 :** Kümeleme yapılmaksızın doğrudan YBMG verileriyle görüntü erişimi deneylerinin çalışma süreleri (saniye).

	Hesaplama zamanları
$f_1$ değerleri ile erişim	13.97
Tek değişkenli terimler uç uca	16.86
Tek değişkenli yaklaştırım	27.42

Çizelge 5.10 ve 5.11’de verilen süreler karşılaştırıldığında genel olarak 5-14 kat gibi kayda değer bir kazanç dikkati çekmektedir. Bu kazanç, kümeleme yaklaşımının algoritma karmaşıklığını azaltma konusundaki yüksek performansını göstermektedir.

Bu çalışmalarda kümeleme yaklaşımı, veri tabanı ölçeğini hedef almaktadır; büyük ölçekli veri tabanını küçük veri tabanlarına ayırarak algoritma karmaşıklığını düşürmektedir. Bu noktada, kümelemenin etkin şekilde yapılması önem kazanmaktadır. Zamandan kazanç sağlanırken başarı olabildiğince korunmalıdır. Kümeleme yönteminin seçimi, küme sayısının tespiti, kümelenecek verinin boyutu ve esas veriyi temsil etmedeki başarısı gibi faktörler yöntemin başarısını koruma konusunda etkili olmaktadır. Bu konularda yapılan tercihler, öne sürülen özgün yöntemlerde etkin bir başarı ortaya koymuştur.

Kümeleme temelde veri tabanı ölçeğini hedef almakla birlikte, aynı zamanda algoritma karmaşıklığını artıran diğer faktörlerin de etkisini azaltmaktadır. Örneğin her biri bir veri noktası olan görüntülerin boyutunun artması veya görüntü erişiminde esas



alınacak verinin boyutunun artması hesaplama sürelerini artırmakta, bu durumda kümeleme zaman kazancı sağlamada daha fazla önem arz etmektedir.

Bu bölümde kümeleme yapılması ile yapılmaması durumları için YBMG ayrıştırımının görüntü erişimi performansı incelenmiştir. Bir sonraki bölümde ise YBMG ve Temel Bileşen Analizi yöntemlerinin görüntü erişimi performansları karşılaştırılacaktır.

## **5.2 YBMG Yöntemi ile Temel Bileşen Analizi Yönteminin Görüntü Erişimi Performansının Karşılaştırılması**

Bu tez çalışması en genel haliyle, YBMG yönteminin görüntü erişimi problemlerinde kullanılması fikrini konu almaktadır. Önceki bölümde çalışmanın kümeleme basamağı üzerinde durulmuş ve yöntemin performansında kümelemenin önemi vurgulanmış olsa da, bu çalışmanın temel motivasyon kaynağı boyut indirgeme amacıyla kullanılan YBMG yöntemidir. Çünkü görüntüler gibi yüksek boyutlu veriler ile çalışılırken verinin işlendiği matematiksel yöntemler boyut problemi ile baş etmede kritik öneme sahiptir ve oluşturulan sistemlerin dayanak noktalarıdır.

Bu tez çalışmasında boyut indirgeme YBMG yöntemiyle yapılmıştır. Bu yöntem pek çok matematiksel yapı kullanılarak elde edilse de, veriler üzerine uygulanışı temel matematiksel operasyonlara dayandığından diğer boyut indirgeme yöntemlerine göre daha düşük algoritma karmaşıklığına sahiptir. Bu özelliği, YBMG'nin zaman performansını yükseltmektedir. Diğer yandan matematiksel yapısının güçlü olması, orijinal veriyi başarılı bir şekilde temsil etmesini sağlamaktadır. Dolayısıyla problemlere etkin çözümler sunmakta ve sistemlerin sonuçlarını iyileştirmektedir.

YBMG yönteminin başarısını ve zaman performansını ortaya koyabilmek için bu yöntem, sıkça kullanılan ve boyut indirgemedede en etkin yöntemlerden biri olarak kabul edilen Temel Bileşen Analizi (TBA) yöntemi ile karşılaştırılacaktır.

TBA yöntemi, çok değişkenli bir veri setinden mümkün olan en büyük varyanslara sahip ve birbirine dik yeni değişkenler oluşturan bir yöntemdir. Bir vektör uzayında ifade edilmiş olan veri seti için, yeni baz vektörler oluşturulur ve böylece yeni bir vektör uzayı elde edilir. Bu bir ortogonal dönüşüm işlemidir. Temel bileşenler şu kuralı sağlamalıdır: Birinci temel bileşen veri setinin en büyük varyansa sahip olduğu

yönü temsil etmelidir ve sonraki her temel bileşen de kendinden öncekilere dik olmak koşuluyla mümkün olan en büyük varyansa sahip olmalıdır. Bileşenler varyanslarına göre sıralandığından en temel bileşenler ilk sıralarda olanlardır. Birinci temel bileşen verilerin en belirgin şekilde farklılaştığı yönü veren bileşendir.

Bileşenler tespit edildikten sonra veri seti bu bileşenlerle yeniden ifade edilebilir. Yani veri noktaları hesaplanan baz vektörler cinsinden ifade edilerek yeni vektör uzayına geçirilebilir. Fakat bu durumda boyut indirgenmemiş olacaktır. Verinin boyutunu düşürebilmek için birinci temel bileşenden başlayarak temel bileşenlerin sadece bir kısmı alınmalıdır. Kaç tane temel bileşen seçileceğine karar vermek için seçilen temel bileşenlere karşılık gelen özdeğerler toplamının, tüm özdeğerler toplamına oranı kullanılabilir. Daha fazla temel bileşen alınarak bu oran artırıldıkça, seçilen temel bileşenlerin verileri temsil etme başarısı artacak, fakat boyut indirgeme oranı azalacaktır. Boyutu önemli oranda indirgeyebilmek için, olabildiğince az sayıda temel bileşen seçilmesi tercih edilir. Kullanılacak temel bileşenler seçildikten sonra bu bileşenler kullanılarak orijinal veri noktalarına yaklaştırım yapılabilir. Bu yaklaştırım, her bir veri noktasının seçilen temel bileşenler üzerine izdüşümü hesaplanarak yapılır. Elde edilen izdüşüm katsayıları ele alınan orijinal veri noktasını bir miktar kayıpla temsil ederler. Sadece seçilen baz vektörler üzerine izdüşüm katsayıları hesaplandığından, verinin değişken sayısından daha az sayıda katsayı elde edilmiş, orijinal verinin boyutu düşürülmüş olur. Temel Bileşen Analizinin matematiksel yapısı Şekil 5.2’de verilmiştir.

Bu algoritmanın 1. basamağında elde edilen veri matrisinin her satırı bir veri noktasına, her sütunu ise bir değişkene karşılık gelmektedir. 3. basamakta, veriler ağırlık merkezi orijine gelecek şekilde taşınmış yani merkezleştirilmiş olur. 4. basamakta elde edilen kovaryans matrisi her bir değişkenin kendisi ve diğer değişkenlerle olan ilişkisini yani varyans ve kovaryans değerlerini taşımaktadır. Kovaryans matrisinin 5. basamakta hesaplanan özvektörleri birbirine diktir ve yeni vektör uzayının baz vektörleridir. Özdeğerler ise bu vektörlerin varyansını verir. Sonraki algoritma basamakları ise önceki paragraflarda açıklanmış bulunmaktadır.

TBA yöntemi literatürde görüntü erişimi amacıyla kullanılmış olan bir yöntemdir. Bu uygulama yapılırken yaygın olarak kullanılan yaklaşım şu şekildedir: İlk olarak bir başlangıç görüntü veri tabanı eğitim görüntüleri olarak alınır. Her bir görüntü

```

1 Veri noktalarını satır vektörler olarak al ve alt alta birleştirerek bir veri matrisi
  oluştur;
2 Veri matrisinin her bir sütununun ortalamasını bul;
3 Veri matrisinin her bir elemanından bulunduğu sütuna ait ortalamayı çıkar;
4 Elde edilen matrisin devriği ile matrisin kendisini çarparak kovaryans matrisini
  bul;
5 Kovaryans matrisinin özdeğer ve özvektörlerini hesapla;
6 Özdeğerleri büyükten küçüğe sırala;
7 Bu sıralamayı kullanarak, özdeğerlere karşılık gelen özvektörleri sırala;
8 Bu özvektörlerden ilk  $k$  tanesini temel bileşenler olarak al;
9 if Verilerin düşük boyutta yeniden ifade edilmesi gerekiyorsa then
10   Her bir veri noktasının temel bileşenler üzerine izdüşüm katsayılarını bul;
11   if Verilerin düşük boyuttaki temsilleri ile orijinal verileri karşılaştırmak
     gerekiyorsa then
12     Her bir veri noktası için katsayıları kullanarak temel bileşenlerin lineer
       kombinasyonunu hesapla;
13     Hesaplanan temsili veri ile orijinal veriyi karşılaştır;
14   end
15 end

```

**Şekil 5.2** : Temel Bileşen Analizi'nin (TBA) genel algoritması.

verisi bir satır vektör haline getirilir. Bu işlem, örneğin, gri tonlamalı bir görüntü için satırları yan yana eklenerek yapılabilir. Elde edilen görüntü vektörleri alt alta birleştirilerek bir veri matrisi oluşturulur. Bu matrise TBA yöntemi uygulanarak özvektörleri hesaplanır ve bunların en büyük özdeğerlere sahip olan bir kısmı temel bileşenler olarak alınır. Bu aşamada seçilen temel bileşenler, veri tabanında bulunan tüm görüntüler için baz görüntüler olarak değerlendirilir. Tüm görüntü vektörlerinin bu baz görüntü vektörleri üzerine izdüşüm katsayıları hesaplanır. Elde edilen bu katsayılar, görüntüleri temsil eder. Özvektörlerin tamamı değil bir kısmı temel bileşenler olarak alındığından, ele alınan bir görüntüyü temsil eden katsayılar, orijinal görüntü vektöründen daha az değer içerir, dolayısıyla boyut indirgenmiş olur. Buraya kadar elde edilmiş olan temel bileşenler, yani baz görüntüler ve ayrıca her bir görüntü için hesaplanmış olan izdüşüm katsayıları saklanır. Görüntü erişimi aşamasında ise veri tabanı dışından bir sorgu görüntüsü alınır ve bu görüntünün aynı baz görüntüler üzerine izdüşümü hesaplanır. Elde edilen temsili katsayılar, eğitim görüntülerinden elde edilen katsayılarla karşılaştırılır, sorgu görüntüsüne en yakın katsayılar sahip görüntüler getirilir [14]. Bu yöntem bilgisayarda uygulanırken veri tabanı ölçeği çok büyük olduğunda, yani veri tabanındaki eğitim görüntülerinin sayısı çok fazla olduğunda bellek yetersizliği sorunu ortaya çıkmaktadır. Bunun nedeni, her bir

görüntünün bir satır vektörü haline getirilip, tüm görüntülerden elde edilen satır vektörlerin birleştirilmesi ve ortaya çıkan devasa büyüklükteki matrisin özdeğer ve özvektörlerinin hesaplanmak istenmesidir.

TBA yöntemi ile bu tez çalışmasında kullanılan YBMG yöntemi, görüntü erişimindeki performansları açısından karşılaştırılacaktır. Önerilen çalışmalarda YBMG yöntemiyle görüntülerin boyutunun indirgenmesi işlemi, görüntüler ayrı ayrı ele alınarak yapıldığından, TBA yöntemiyle boyut indirgeme yapılırken de görüntüler ayrı ayrı ele alınmalıdır. TBA yöntemi literatürde bulunan ve yukarıda anlatılmış olan yaklaşımla görüntü erişimine uygulandığında, her bir görüntüye ayrı ayrı değil, tüm görüntülere bir bütün halinde uygulanmış olmaktadır. Çünkü her bir görüntü bir satır vektörü haline getirilip alt alta birleştirilmekte ve elde edilen büyük matrise TBA yöntemi uygulanmaktadır. Burada ise, YBMG yöntemi ile karşılaştırma yapılacağı için, YBMG yönteminde yapıldığı gibi görüntülere ayrı ayrı TBA yöntemi uygulanmak istendiğinden, her bir görüntünün ayrı bir matris olarak ele alınması ve TBA yönteminin bu matrislere ayrı ayrı uygulanması tercih edilmiştir. Bununla beraber, COIL-100 veri tabanındaki görüntülerle, yine de, literatürdeki gibi çalışmak tercih edildiğinde bellek yetersizliği sorunu ortaya çıkmakta ve bu çalışma mümkün olamamaktadır.

RGB formatındaki tek bir görüntü ele alınıp TBA yöntemi uygulanmak istendiğinde, öncelikle bir üç boyutlu dizi formatında olan görüntü verisinin, bir matris haline getirilmesi gerekmektedir. Bu işlem RGB formatındaki görüntü gri tonlamalı görüntüye çevrilerek yapılabilir.  $n_1 \times n_2 \times 3$  boyutundaki bir görüntüden elde edilen gri tonlamalı görüntü,  $n_1 \times n_2$  boyutunda bir matristir. Bu matrise TBA yöntemi uygulandığında düşük boyutlu bir veri elde edilecek ve orijinal görüntü bu veri ile temsil edilebilecektir. Tek bir görüntüye TBA yöntemi uygulandığından, gri tonlamalı görüntünün satırları veri noktaları olarak, sütunları ise değişkenler olarak alınmış olacaktır. Bu durumda temel bileşenlerden her biri görüntünün tamamını temsil ederken, bir satır vektörün temel bileşenler üzerine izdüşüm katsayıları ise görüntünün yalnızca o satırını temsil edecektir. İzdüşüm katsayıları görüntünün satırlarını ayrı ayrı temsil ettiğinden, görüntüyü temsil edecek veri olarak kullanılması uygun değildir. Bu durumda orijinal görüntüyü bütüncül olarak temsil edecek düşük boyutlu veri olarak, temel bileşenlerden birinin, yani birinci temel bileşenin seçilmesi daha uygundur.

Burada, veri matrisi olarak alınan gri tonlamalı görüntü  $n_1 \times n_2$  boyutunda bir matris olduğundan, elde edilecek birinci temel bileşen  $n_2$  tane değer içerecek, başka bir deyişle  $n_2$  boyutlu olacaktır.

TBA yöntemiyle görüntü erişimi çalışması yapılırken görüntü kümeleme işlemi yapılmayacaktır. Çünkü TBA yöntemi görüntülere ayrı ayrı uygulandığında,  $f_0$  veya YBMG- $f_3$  düzeyinde düşük boyutlu bir veri elde edilmemektedir. Dolayısıyla TBA yöntemiyle yapılan görüntü erişimi çalışması, Bölüm 5.1’de verilen, kümeleme basamağının bulunmadığı görüntü erişimi çalışmasıyla karşılaştırılacaktır. Böylece karşılaştırılacak olan yöntemler kümeleme yapılmaması yönüyle aynı olduklarından yapılacak karşılaştırma, doğrudan kullanılan boyut indirgeme yöntemlerinin karşılaştırması, yani YBMG - TBA karşılaştırması olacaktır.

Görüntülere ayrı ayrı TBA uygulandığı, gri tonlamalı görüntüye çevirme yöntemiyle görüntü verisinin matris haline getirildiği, TBA ile elde edilen birinci temel bileşenin görüntüyü temsil ettiği bu sisteme, bu tez çalışmasında TBA-I adı verilmiştir. Bu yöntemin algoritma yapısı Şekil 5.3’teki gibidir.

```
1 Sorgu görüntüsünü oku ve gri tonlamalı görüntüye dönüştür;  
2 Bu görüntü matrisine TBA yöntemi uygula ve birinci temel bileşeni al;  
3 forall Eğitim görüntüleri do  
4   Eğitim görüntüsünü oku ve gri tonlamalı görüntüye dönüştür;  
5   Bu görüntü matrisine TBA yöntemi uygula ve birinci temel bileşeni al;  
6   Eğitim görüntüsünün birinci temel bileşeni ile sorgu görüntüsünün birinci  
   temel bileşeni arasındaki karesel Öklit uzaklığını hesapla;  
7   Uzaklık bilgisini sakla;  
8 end  
9 Uzaklıkları küçükten büyüğe sırala;  
10 En küçük  $S$  tane uzaklığı sıralı olarak al;  
11 forall En küçük  $S$  tane uzaklık do  
12   Uzaklığın hangi eğitim görüntüsünden elde edildiğini bul;  
13   if Bu eğitim görüntüsü ile test görüntüsü aynı objeye aitse then  
14     En küçük kaçınıcı uzaklık olduğuna göre azalan kümülatif kazanca  
     katkısını hesapla ve kazanca ekle;  
15   end  
16 end  
17 return Kazancı ve hesaplama süresini geri döndür;
```

Şekil 5.3 : TBA-I yöntemiyle görüntü erişimi algoritması.

Algoritmada bulunan  $S$ , en yakın kaç tane görüntünün getirilmek istendiğidir. YBMG yönteminin kullanıldığı çalışmalarda olduğu gibi bu çalışmada da en yakın 6, 2 ve

1 tane görüntü getirilmesi durumları ele alınmış ve bu durumlar için başarı oranları hesaplanmıştır. Sonuçlar Çizelge 5.12’de verilmiştir.

**Çizelge 5.12** : TBA-I yöntemiyle görüntü erişimi deneyinin başarı oranları.

Başarı oranı	
6	% 70.8
2	% 85.3
1	% 88.0

Sonuçları verilen bu yöntem tasarlanırken görüntüler RGB formatından gri tonlamalı görüntü formatına çevrilmiştir. Fakat RGB formatındaki görüntüyü ifade eden üç boyutlu diziyi matrise çevirme işlemi farklı şekillerde de yapılabilir. Örneğin, RGB formatındaki görüntünün R G ve B renk matrisleri uç uca eklenerek de matris elde edilebilir.

Renk matrislerinin uç uca eklenmesi işlemi, yan yana veya alt alta eklenmesi gibi iki farklı şekilde yapılabilir. Alt alta eklenmesi durumunda elde edilen matrisin satır sayısı artırılmış olacaktır. Satır sayısının artırılması veri noktası sayısının artırılması anlamına gelmektedir. Değişken sayısı anlamına gelen sütun sayısı değişmeyecektir. Dolayısıyla bu matrise TBA uygulanmasıyla elde edilecek olan birinci temel bileşenin boyutu, gri tonlamalı görüntü matrisine TBA uygulanmasıyla elde edilen birinci temel bileşenin boyutu ile aynı olacaktır. Bu durumda, matrise çevirme şeklinin değiştirilmesinin başarıya katkı sağlaması beklenmez. Tam tersine, tek bir piksele ait renk değerleri birbirinden uzak noktalara taşındığından başarıyı düşürme etkisi yapacaktır.

Renk matrislerinin yan yana eklenmesi durumunda ise, değişken sayısı olarak yorumlanan sütun sayısı üç katına çıkarılmış olur. Böylece TBA ile elde edilecek birinci temel bileşenin boyutu da aynı oranda artacaktır. Bu durum algoritma karmaşıklığını yükseltse de, başarıya önemli bir katkı sağlayacaktır. Bu yüzden bu çalışmada, görüntünün renk matrislerinin yan yana eklenmesiyle veri matrisi elde edilmesi tercih edilmiştir. Bu veri matrisi  $n_1 \times (3 \cdot n_2)$  boyutunda bir matris olduğundan, birinci temel bileşen  $3 \cdot n_2$  tane değer içerecek, yani  $3 \cdot n_2$  boyutlu bir vektör olacaktır. Bu yöntem TBA-II adı verilmiştir ve algoritma yapısı, görüntülerin gri tonlamaya dönüştürülmesi yerine renk matrislerinin yan yana eklenmesi, yazılması

kaydıyla Şekil 5.3'te verilen algoritmaya benzer şekilde oluşturulmuştur. Elde edilen başarı oranları ise Çizelge 5.13'te verilmiştir.

**Çizelge 5.13** : TBA-II yöntemiyle görüntü erişimi deneyinin başarı oranları.

	Başarı oranı
6	% 90.4
2	% 96.8
1	% 98.0

Çizelge 5.13'te görüldüğü gibi, görüntülerin renk matrislerinin yan yana eklenmesi durumunda TBA yöntemi görüntü erişiminde oldukça başarılı sonuçlar vermektedir. Yan yana ekleme durumunda elde edilen matrisin, gri tonlamalı görüntüye çevirme durumuna kıyasla satır sayısında bir değişiklik olmazken sütun sayısı üç kat artmıştır. Bu yüzden algoritma çalışma süresinde önemli bir artış beklenir. Zaman karşılaştırmalarına gelindiğinde, gri tonlamalı görüntüye çevirme durumunda da oldukça yüksek olan hesaplama süresinin, yan yana ekleme durumunda daha da yükseldiği görülecektir.

Üç boyutlu dizi yapısında olan RGB formatındaki bir görüntüyü matrise çevirmek için ele alınabilecek üçüncü bir metot ise görüntünün piksel noktalarının birer değişken olarak, bu piksel noktalarını oluşturan R, G ve B değerlerinin ise birer veri noktası olarak değerlendirilmesi ve bu şekilde veri matrisi elde edilmesi metodudur. Bunun için görüntünün renk matrisleri ayrı ayrı ele alınır. Her bir renk matrisinin sütunları alt alta eklenerek veya satırları yan yana eklenerek bir vektör oluşturulur. Her bir renk için ayrı ayrı oluşturulan bu vektörler birleştirilerek bir matris elde edilir ve bu matrise TBA yöntemi uygulanır. Renk matrislerinin satırlarının yan yana eklenmesi ve sütunlarının alt alta eklenmesi arasında kayda değer bir fark yoktur, çünkü her iki durumda da oluşturulacak vektör  $n_1 \cdot n_2$  uzunluğuna sahiptir ve aynı değerleri sadece yerleri değişik olarak taşımaktadır. Bu çalışmada sütunların alt alta eklenmesi tercih edilmiştir ve elde edilen sütun vektörü, devriği alınarak satır vektöre çevrilmiştir. Bu şekilde her bir renk için ayrı ayrı oluşturulan satır vektörleri, birer veri noktası olarak değerlendirilip alt alta birleştirilmiştir. Elde edilen veri matrisi  $3 \times (n_1 \cdot n_2)$  boyutundadır. Bu matrise TBA yöntemi uygulanmış ve yine birinci temel bileşen, görüntüyü temsil edecek düşük boyutlu veri olarak seçilmiştir. Bu çalışmada birinci temel bileşen  $n_1 \cdot n_2$  tene değer içermektedir. Bu yöntem TBA-III adı verilmiş ve algoritma yapısı, görüntülerin matrise çevrilme şeklinin değiştirilmesi kaydıyla, Şekil

5.3'te verilen algoritmaya benzer şekilde oluşturulmuştur. Bu çalışmadan elde edilen başarı oranları ise Çizelge 5.14'te verilmiştir.

**Çizelge 5.14** : TBA-III yöntemiyle görüntü erişimi deneyinin başarı oranları.

Başarı oranı	
6	% 92.8
2	% 98.5
1	% 100.0

Çizelgede görüldüğü gibi, görüntülerin renk matrislerinden vektör oluşturulması durumunda TBA yönteminin görüntü erişimindeki başarısı daha da yükselmiştir. Bu şekilde elde edilen veri matrisinin, diğer durumlara kıyasla sütun sayısı kat kat artmış, satır sayısı ise sadece 3 olmuştur. Eleman sayısı bakımından ise, gri tonlamalı görüntüye çevirmeyle elde edilen matrisin eleman sayısının üç katı kadar, renk matrislerinin yan yana eklenmesiyle elde edilen matrisin eleman sayısına ise eşittir. Bu durumun, gri tonlamalı görüntü durumundaki sürenin bir miktar üzerinde bir hesaplama süresi getirdiği görülecektir.

Buraya kadar TBA yöntemiyle görüntü erişimi, üç farklı yöntemle incelenmiş bulunmaktadır. Bu yöntemler, kümeleme yapılmaksızın YBMG verileriyle görüntü erişimi yöntemleriyle karşılaştırılacaktır. TBA-I yöntemiyle elde edilen birinci temel bileşen  $n_2$  boyutlu olduğundan bu yöntem,  $n_1$  boyutlu olan  $f_1$  değerlerinin kullanıldığı YBMG çalışmasıyla karşılaştırılacaktır. TBA-II yöntemiyle elde edilen birinci temel bileşen  $3 \cdot n_2$  boyutlu olduğundan bu yöntemin,  $n_1 + n_2 + 3$  boyutlu olan, YBMG tek değişkenli verilerin uç uca eklenmesi ile elde edilen vektörün kullanıldığı çalışma ile karşılaştırılması tercih edilmiştir. TBA-III yöntemiyle elde edilen birinci temel bileşen  $n_1 \cdot n_2$  boyutlu olduğundan  $n_1 \cdot n_2 \cdot 3$  boyutlu olan YBMG tek değişkenli yaklaşım değerlerinin kullanıldığı çalışma ile karşılaştırılacaktır. İkili olarak karşılaştırılacak yöntemlerde kullanılan temsili verinin yakın boyutlarda olması, boyutu benzer oranda indirgemek için ve boyut indirgindikten sonraki benzerlik ölçümü için gerekli olan hesaplama zamanının etkin bir şekilde karşılaştırılmasını sağlayacaktır. Daha önemlisi, boyutun benzer oranda indirgenmesi durumu için, görüntülerin temsil edilme başarıları karşılaştırılmış olacaktır. (Gerçekte YBMG tek değişkenli yaklaşım verisi boyutu indirgememektedir fakat TBA-III yönteminin birinci temel bileşeni ile boyut uyumluluğu söz konusu olduğundan YBMG - TBA kıyaslaması için karşılaştırma



yapılmıştır.) Bahsedilen karşılaştırmalar, başarı oranları açısından Çizelge 5.15, 5.16 ve 5.17’de nicel olarak verilmiştir.

**Çizelge 5.15 :** Kümeleme yapılmaksızın YBMG- $f_1$  değerleriyle görüntü erişimi ve TBA-I birinci temel bileşen değerleriyle görüntü erişimi karşılaştırması.

	YBMG - $f_1$ vektörü	TBA gri tonlamalı
6	% 90.8	% 70.8
2	% 94.1	% 85.3
1	% 96.0	% 88.0

**Çizelge 5.16 :** Kümeleme yapılmaksızın  $f_1$ ,  $f_2$  ve  $f_3$ ’ün uç uca eklenmesiyle görüntü erişimi ve TBA-II birinci temel bileşen değerleriyle görüntü erişimi karşılaştırması.

	YBMG tek deęiş. uç uca	TBA renk matr. yan yana
6	% 93.1	% 90.4
2	% 98.6	% 96.8
1	% 99.0	% 98.0

**Çizelge 5.17 :** Kümeleme yapılmaksızın YBMG tek deęişkenli yaklaşım değerleriyle görüntü erişimi ve TBA-III birinci temel bileşen değerleriyle görüntü erişimi karşılaştırması.

	YBMG tek deęiş. yaklaşt.	TBA renk vektörleri
6	% 96.0	% 92.8
2	% 98.6	% 98.5
1	% 99.0	% 100.0

Bu karşılaştırma çizelgeleri incelendiğinde, TBA yöntemi gri tonlamalı görüntüye çevirme durumunda düşük sonuçlar verirken, renk matrislerinin yan yana eklendiği veya renk matrislerinden vektör oluşturulduğu durumlarda, başarılı fakat YBMG yönteminden bir miktar düşük sonuçlar vermiştir.

Bir boyut indirgeme yönteminin performansı değerlendirilirken başarı oranlarının yanı sıra algoritmanın hesaplama karmaşıklığına da bakılması gerekir. Aynı başarıyı veya yakın başarı oranlarını veren iki yöntem arasında hesaplama zamanları açısından bazen kat kat fark olabilmektedir. Böyle durumlarda maliyeti düşürmek adına algoritma karmaşıklığı düşük olan yöntemler tercih edilmektedir. Bunun için başarı oranlarını karşılaştırdığımız bu yöntemlerin hesaplama süreleri de Çizelge 5.18, 5.19 ve 5.20’de karşılaştırılmıştır.

**Çizelge 5.18** : Çizelge 5.15'te başarı oranları karşılaştırılan deneylerin çalışma süreleri karşılaştırması (saniye).

YBMG - $f_1$ vektörü	TBA gri tonlamalı
13.97	35.5

**Çizelge 5.19** : Çizelge 5.16'da başarı oranları karşılaştırılan deneylerin çalışma süreleri karşılaştırması (saniye).

YBMG tek deęiş. uç uca	TBA renk matr. yan yana
16.86	62.01

**Çizelge 5.20** : Çizelge 5.17'de başarı oranları karşılaştırılan deneylerin çalışma süreleri karşılaştırması (saniye).

YBMG tek deęiş. yaklaşt.	TBA renk vektörleri
27.42	39.36

Saniye cinsinden verilen bu deęerler, 100 farklı sorgu görüntüsünün her biri ile yapılan görüntü erişim deneylerinin süre ortalaması alınarak hesaplanmıştır ve dolayısıyla tek bir görüntünün görüntü erişiminin süresini temsil etmektedir.

YBMG yönteminin TBA yöntemine kıyasla önemli oranda zaman kazancı sağladığı görülmektedir. Çizelge 5.18 karşılaştırmasında bu oran 2.5 kattır. Buradaki TBA yönteminin, gri tonlamalı görüntüye çevrilen yani boyutu peşinen indirgenen görüntüler üzerine uygulandığı da hesaba katılmalıdır. Çizelge 5.19 karşılaştırmasında zaman kazancı 3.6 kata çıkmıştır. Çizelge 5.20 karşılaştırmasında ise süreler yakın olmasına rağmen YBMG yöntemi yine de zaman kazancı sağlamaktadır. Burada, kullanılan YBMG verisinin üç kat daha fazla deęer içeriyor olmasına rağmen yine de zaman kazancı sağlamanın nedeni, YBMG yönteminin TBA yöntemine göre oldukça hızlı çalışmasıdır. Bu karşılaştırmalarla YBMG yönteminin hesaplama zamanı performansının TBA yöntemine kıyasla belirgin oranda yüksek olduğu sonucuna ulaşılır.

Başarı oranları ve hesaplama süreleri karşılaştırması birlikte deęerlendirildiğinde YBMG ayrıştırımının tercih edilebilecek önemli bir boyut indirgeme yöntemi olduğu anlaşılmaktadır.

Boyut indirgeme yöntemleri veri noktalarının boyutunu hedef almaktadır. Boyutu düşürerek sistemlerin algoritma karmaşıklığını azaltmakta, yani verimliliği arttırmaktadırlar. Bununla birlikte, elde edilen düşük boyutlu verinin orijinal

veriyi temsil kabiliyetinin de olabildiğince yüksek olması gerekmektedir. Ayrıca, boyut indirgeme temelde veri noktalarının boyutu ile ilgili olsa da verilerle ilgili diğer faktörlere bağlı olarak da etkinliği değişmektedir. Örneğin veri tabanının içerdiği veri sayısı arttıkça boyut indirgeme yöntemleri arasındaki maliyet farkı daha belirgin olacaktır. Bu noktalarda YBMG yönteminin kayda değer bir başarı gösterdiği görülmektedir.

Özetle, bu bölümde YBMG ve TBA boyut indirgeme yöntemleri karşılaştırılmış ve YBMG yönteminin daha avantajlı olduğu sonucuna ulaşılmıştır.





## 6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, görüntü erişimi problemi ele alınmış ve Yüksek Boyutlu Model Gösterilimi (YBMG) yöntemi ile boyut indirgeme yapılan bir sistem önerilmiştir. Ayrıca görüntü erişimi öncesi görüntü kümeleme yapılarak görüntü erişimi sürecinin hızlandırılması hedeflenmiştir. Bu bakış açısıyla önerilen yöntemler COIL-100 görüntü veri tabanı üzerinde yapılan deneysel çalışmalarla test edilmiş ve elde edilen sonuçlar ilgili kısım içinde verilmiştir. Önerilen yöntemler ayrıca karşılaştırma çalışmalarıyla da analiz edilmiştir.

Yapılan deneysel çalışmalarla görüntü kümeleme basamağında, eğitim görüntülerinin kümelenmesinde kullanılan YBMG- $f_3$  verisinin YBMG- $f_0$  verisine göre daha başarılı olduğu görülmüştür. Bunun nedeni  $f_3$  verisinin görece daha yüksek boyutlu olması ve aynı zamanda görüntülerin renk özneteliğini de yansıtmasıdır. Bununla beraber, daha yüksek boyutlu olarak kullanılmadığı, tek bir değeri ile kümeleme yapıldığı çalışmalarda da, görüntülerin renk özneteliğini yansıtması sayesinde  $f_0$  verisinden daha başarılı çıkmıştır. Ayrıca,  $f_0$  verisi beklendiği gibi küme sayısının daha yüksek olduğu çalışmalarda başarı kaybına neden olurken,  $f_3$  verisi bu duruma karşı oldukça dayanıklıdır. Küme sayısının 60 olarak seçilmesi durumunda dahi bazı erişim çalışmalarında % 100 başarı oranı vermiştir.

Kullanılan kümeleme yöntemleri karşılaştırıldığında, eşit aralık ve k-ortalamlar yöntemleri daha başarılı bulunmuştur. Eşit üye sayısı yöntemi ise verilerin çok yoğunlaştığı bölgelerde fazla seçici davranarak, aynı objeye ait görüntülerin aynı kümeye atanmasında sapmalara neden olmaktadır.

Görüntü erişimi aşamasına gelindiğinde ise, bu aşamada kullanılan verilerden YBMG- $f_1$  verisinin, başarı ve hız açısından oldukça verimli olduğu sonucuna ulaşılmıştır.  $f_1$ ,  $f_2$  ve  $f_3$  vektörlerinin uç uca eklenmesi ile elde edilen vektör ise genel olarak  $f_1$  verisi kadar başarılı sonuçlar vermese de küme sayısının çok olduğu durumlarda tercih edilebilir. Tek değişkenli yaklaştırım değerleri ile görüntü erişimi yapılması durumu ise, boyut indirgeme yapılmadığı için ekonomik olmamasına

rağmen, sabit ve tek deęişkenli terimlerin görüntüleri temsil etmedeki yüksek başarısını göstermesi bakımından kayda değerdir.

Tez çalışmasında ortaya atılan yöntemlerin değerlendirilmesi için yapılan karşılaştırma çalışmaları iki ayrı başlıkta incelenmiştir. Önerilen yöntemlerle kümeleme basamağının ihmal edildiği yöntemin karşılaştırmasında  $f_0$  değerleri ile kümeleme yapılan çalışmalar, kümeleme yapılmayan çalışmaların gösterdiği başarıyı yakalayamamış, fakat  $f_3$  değerleri ile kümeleme yapılan çalışmalar, kümeleme yapılmayan çalışmalardan daha başarılı sonuçlar vermiştir. Her iki durumda da kümeleme işlemi, hedeflendiği gibi, kat kat zaman kazancı sağlamıştır.

YBMG yönteminin boyut indirgeme ve görüntüleri temsil etme başarısını ölçmek için yapılan yöntem karşılaştırması çalışmasında ise Temel Bileşen Analizi (TBA) kullanılmıştır. Bu karşılaştırmayla, YBMG yönteminin TBA yöntemine göre daha başarılı sonuçlar verdiği ve aynı zamanda zaman kazancı sağladığı görülmüştür.

Tüm bu sonuçlar birlikte değerlendirildiğinde YBMG yönteminin boyut indirgeme amacıyla kullanılacak önemli bir ayrıştırım yöntemi olduğu anlaşılmaktadır. Bu yöntem hem verileri temsil etme başarısı hem de hesaplama verimliliği konusunda oldukça avantajlıdır. Ayrıca, görüntü erişimi öncesi kümeleme yaklaşımının hesaplama verimliliğini artırdığı sonucuna ulaşılmaktadır.

Bundan sonra yapılacak çalışmalar için, bu tezde geliştirilen yaklaşımın görüntülerin öznitelikleri çıkarılarak bunlar üzerinde uygulanması önerilebilir. Görüntü verisi yerine bu veriden çıkarılan öznitelikler üzerinde çalışıldığında daha başarılı sonuçlar elde edilebilir. Ayrıca, kümeleme yöntemi olarak bulanık c-ortalamlar (fuzzy c-means) yöntemi de denenebilir. Bunların yanı sıra, COIL-100 görüntü veri tabanı dışında başka görüntü veri tabanları üzerinde de deneyler yapılarak yöntemlerin başarısı test edilebilir.

## KAYNAKLAR

- [1] **Braveen, M. ve Dhavachelvan, P.** (2009). Evaluation of Content Based Image Retrieval Systems Based on Color Feature, *International Journal of Recent Trends in Engineering*, 1(2), 57–62.
- [2] **Karthikeyan, T., Manikandaprabhu, P. ve Nithya, S.** (2014). A Survey on Text and Content Based Image Retrieval System for Image Mining, *International Journal of Engineering Research and Technology (IJERT)*, 3(3), 509–512.
- [3] **Yan, L. ve Ma, Z.** (2011). *Intelligent Multimedia Databases and Information Retrieval: Advancing Applications and Technologies*, IGI Global.
- [4] **Alkhawlani, M., Elmogy, M. ve El-Bakry, H.** (2015). Text-based, Content-based, and Semantic-based Image Retrievals: A Survey, *International Journal of Computer and Information Technology*, 4(1), 58–66.
- [5] **Kavitha, H. ve Sudhamani, M.V.** (2014). Object Based Image Retrieval from Database Using Combined Features, *2014 Fifth International Conference on Signal and Image Processing*, s.161–165.
- [6] **Datta, R., Joshi, D., Li, J. ve Wang, J.Z.** (2008). Image Retrieval: Ideas, Influences, and Trends of the New Age, *ACM Computing Surveys*, 40(2), 5:1–5:60.
- [7] **Zhuo, L., Cheng, B. ve Zhang, J.** (2014). A comparative study of dimensionality reduction methods for large-scale image retrieval, *Neurocomputing*, 141(Supplement C), 202–210.
- [8] **Ogle, V.E. ve Stonebraker, M.** (1995). Chabot: Retrieval from a Relational Database of Images, *Computer*, 28(9), 40–48.
- [9] **Oktar, Y. ve Türkan, M.** (2018). A review of sparsity-based clustering methods, *Signal Processing*, 148, 20–30.
- [10] **Cluster analysis** (t.y.). *Wikipedia*. Erişim: 15 Kasım 2017, [https://en.wikipedia.org/wiki/Cluster\\_analysis](https://en.wikipedia.org/wiki/Cluster_analysis).
- [11] **Benkrama, S., Zaoui, L. ve Charrier, C.** (2014). Clustering and dimensionality reduction for image retrieval in high-dimensional spaces, *Journal of Theoretical and Applied Computer Science*, 8(3), 37–50.
- [12] **Wang, J.** (2012). *Geometric Structure of High-Dimensional Data and Dimensionality Reduction*, Higher Education Press.
- [13] **Zhong, J., Wang, L.N., Ling, X. ve Dong, J.** (2016). An overview on data representation learning: From traditional feature learning to recent deep learning, *The Journal of Finance and Data Science*, 2(4), 265–278.

- [14] **Turk, M. ve Pentland, A.** (1991). Eigenfaces for recognition, *Journal of Cognitive Neuroscience*, 3(1), 71–86.
- [15] **Önder Özütemiz** (2015). *Yüksek Boyutlu Model Gösterilimi Yöntemi İle Büyük Ölçekli Veri Tabanlarında Görüntü Erişimi*, (Yüksek lisans tezi). İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [16] **Sobol', I.M.** (1990). Sensitivity Estimates for Nonlinear Mathematical Models, *Matematicheskoe Modelirovanie*, 2(1), 112–118, Rusça, İngilizce Çev: MMCE, (1993) 1(4), 407–414.
- [17] **Rabitz, H. ve Alış, Ö.F.** (1999). General foundations of high-dimensional model representations, *Journal of Mathematical Chemistry*, 25, 197–233.
- [18] **Rabitz, H. ve Alış, Ö.F.** (2001). Efficient implementation of high dimensional model representations, *Journal of Mathematical Chemistry*, 29(2), 127–142.
- [19] **Li, G., Rosenthal, C. ve Rabitz, H.** (2001). High Dimensional Model Representations, *The Journal of Physical Chemistry A*, 105(33), 7765–7777.
- [20] **Demiralp, M.** (2003). High dimensional model representation and its application varieties, *Mathematical Research*, 9, 146–159.
- [21] **Tunga, M.A. ve Demiralp, M.** (2009). Computational complexity investigations for high-dimensional model representation algorithms used in multivariate interpolation problems. In N. Mastorakis, J. Sakellaris (eds.), *Advances in Numerical Methods*, (pp.15–29). Springer US.
- [22] **Tunga, B. ve Demiralp, M.** (2010). The influence of the support functions on the quality of enhanced multivariate product representation, *The Journal of Mathematical Chemistry*, 48, 827–840.
- [23] **Tunga, B. ve Demiralp, M.** (2009). Constancy maximization based weight optimization in high dimensional model representation, *Numerical Algorithms*, 52(3), 435–459.
- [24] **Demiralp, E.** (2009). Applications of High Dimensional Model Representations to Computer Vision, *WSEAS Transactions on Mathematics*, 8(4), 184–192.
- [25] **Yuan, Q. ve Liang, D.** (2011). A New Multiple Sub-domain Rs-hdmr Method and Its Application to Tropospheric Alkane Photochemistry Model, *The International Journal of Numerical Analysis and Modeling - Series B*, 2(1), 73–90.
- [26] **Mehta, P.M., Kubicek, M., Minisci, E. ve Vasile, M.** (2017). Sensitivity analysis and probabilistic re-entry modeling for debris using high dimensional model representation based uncertainty treatment, *Advances in Space Research*, 59(1), 193–211.
- [27] **Tunga, M.A. ve Demiralp, M.** (2008). A new approach for data partitioning through high dimensional model representation, *International Journal of Computer Mathematics*, 85(12), 1779–1792.
- [28] **Karcılı, A. ve Tunga, B.** (2017). High dimensional model representation (HDMR) with clustering for image retrieval, *American Institute of Physics Conference Proceedings (AIP)*, cilt1798, s.1–8.



- [29] **Altın, E. ve Tunga, B.** (2014). High Dimensional Model Representation in Image Processing, *International Conference on Computational and Mathematical Methods in Science and Engineering (ICMMSE)*, s.55–64.
- [30] **Tunga, B. ve Koçanaoğulları, A.** (2017). Digital image decomposition and contrast enhancement using high-dimensional model representation, *Signal, Image and Video Processing*.
- [31] **Karaca, E. ve Tunga, M.A.** (2016). A method for inpainting rectangular missing regions using High Dimensional Model Representation and Lagrange interpolation, *2016 24th Signal Processing and Communication Application Conference (SIU)*, s.1349–1352.
- [32] **Karaca, E. ve Tunga, M.A.** (2018). An interpolation-based texture and pattern preserving algorithm for inpainting color images, *Expert Systems with Applications*, 91, 223–234.
- [33] **Columbia Object Image Library (COIL-100)**. (1996). Erişim: 01 Eylül 2017, <http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php>.
- [34] **MATLAB and Statistics Toolbox** (Release 2017a) [Bilgisayar yazılımı]. The Mathworks, Inc.



## ÖZGEÇMİŞ

**Ad Soyad:** Ayşegül KARCILI

**Doğum Tarihi ve Yeri:** 10.04.1989, Antalya

**E-Posta:** aysegulkarcili@outlook.com



### ÖĞRENİM DURUMU:

- **Yüksek Lisans:** 2018, İstanbul Teknik Üniversitesi, Matematik Mühendisliği
- **Lisans:** 2013, Boğaziçi Üniversitesi, Matematik Öğretmenliği
- **Orta Öğretim:** 2007, Denizli Lütfi Ege Anadolu Öğretmen Lisesi

### YÜKSEK LİSANS TEZİNDEN TÜRETİLEN YAYINLAR VE SUNUMLAR:

- A. Karcılı ve B. Tunga, "High dimensional model representation (HDMR) with clustering for image retrieval", ICNPAA 2016 World Congress: International Conference in Nonlinear Problems in Aviation and Aerospace (July 4-8, 2016 La Rochelle, France), AIP Conference Proceedings, 1798, 020072, 2017.