KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ
ANABİLİM DALI

DOKTORA TEZİ

MEVKİSEL VE ANLAMSAL GÖRECELİ NİTELİKLER YARDIMIYLA GÖRÜNTÜ TANIMA

EMRAH ERGÜL

KOCAELİ 2016

KOCAELİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ

ANABİLİM DALI

DOKTORA TEZİ

MEVKİSEL VE ANLAMSAL GÖRECELİ NİTELİKLER YARDIMIYLA GÖRÜNTÜ TANIMA

EMRAH ERGÜL

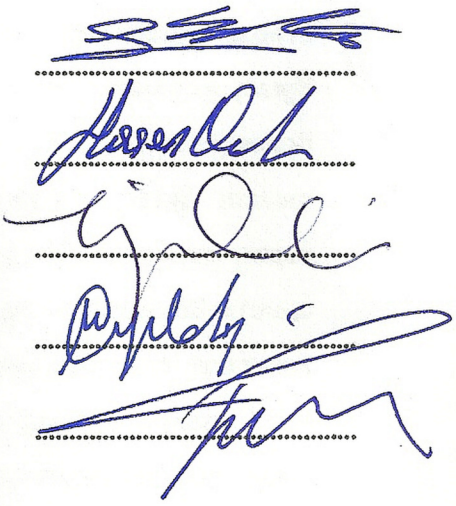
Prof.Dr. Sarp ERTÜRK
Danışman, Kocaeli Üniv.

Prof.Dr. Hasan OCAK
Jüri Üyesi, Kocaeli Üniv.

Doç.Dr. M. Kemal GÜLLÜ
Jüri Üyesi, Kocaeli Üniv.

Yrd.Doç.Dr. Murat YILDIZ
Jüri Üyesi, Sakarya Üniv.

Yrd.Doç.Dr. Tolga ÖNEL
Jüri Üyesi, Deniz Harp Okulu



Tezin Savunulduğu Tarih: 08.02.2016

ÖNSÖZ VE TEŞEKKÜR

Bilgisayar teknolojisinin gelişmesiyle birlikte; günümüzde çok büyük miktarda görüntü verisini kısa zamanda üretme, saklama ve işleme yeteneğine sahibiz. Ayrıca kurduğumuz çeşitli ölçekteki sosyal ağlar yardımıyla veriye istediğimiz yerden istediğimiz zaman ulaşabiliyoruz. Bunun bir parçası olarak özellikle son on yılda video görüntüleri ve imgelerin üretilme hızı inanılmaz boyutlara ulaşmış ve uygulamaları çok çeşitli askeri ve endüstriyel sahalara yayılmıştır.

Görüntü verisi üzerinde tatmin edici bir başarı yüzdesiyle, denetimsiz veya düşük denetimle sahne/nesne tanıma/sınıflandırma işlemini otomatik şekilde yapabilen bir bilgi sistemi karşılanması gereken önemli bir ihtiyaçtır. Böyle bir yeteneğe sahip sistem özellikle ortam araması, çevre algılaması, robot uygulamaları, uydu görüntüleme, hastalık tanıma, istihbarat toplama vb. pek çok karar destek uygulamasında kullanılabilecektir.

Görüntü tanıma problemine yönelik literatür çalışmalarının temelde yerel veya bütünsel düşük seviye öznitelik vektörlerinin çıkartılması, orta veya ileri seviye imge temsili ve sınıflandırıcı metodlarının uygulanma usulleri bakımından ayrıştığı görülmüştür. Bu kapsamda İmgenin nasıl temsil edileceğinin belirlenmesinin başarı performansını etkileyen en önemli husus olduğu tespit edilmiştir.

Bu tez çalışmasında, görüntü tanıma probleminin çözümüne yönelik iki özgün yöntem önerilmiştir. İlk aşamada görüntü tanıma probleminde sıkça kullanılan görsel kelimeler sözlüğünü denetimsiz ve hiyerarşik şekilde elde ederek ham veriyi öncelikle düşük seviye ve boyutta yeni bir öznitelik uzayına taşımak amacıyla derin sinir ağı yapısı tasarlanmıştır. Ayrıca ikinci aşamada, öğrenci-öğretmen etkileşimli bir yapı içerisinde, başlangıçta denetimsiz olarak oluşturulan kategori tabanlı göreceli nitelik uzaylarını sınıf modellerine uygun şekilde ve eş zamanlı güncelleyecek özgün bir görüntü tanıma algoritması geliştirilmiştir.

Çalışmalarım süresince bana her türlü desteği sunan ve tezim ile ilgili yaptığım faaliyetlerin tüm aşamalarında yanımda olan tez danışmanım Prof. Dr. Sarp ERTÜRK'e, ikinci tez danışmanım Doç. Dr. Nafiz ARICA'ya, tez izleme komitesi üyesi hocalarım Prof. Dr. Hasan OCAK ve Doç. Dr. M. Kemal GÜLLÜ'ye sonsuz teşekkürlerimi sunarım.

Diğer taraftan, TÜBİTAK'ın 4 Ocak 2013 gün ve B.14.2.TBT.0.06.01-214-83 sayılı kararı sonucunda bu tez çalışması 2214-Yurt Dışı Araştırma Burs Programı kapsamında desteklenmiştir. Bu kapsamda, Haziran 2013 tarihinden itibaren bir yıl süre ile doktora tez çalışmalarımı University of Illinois at Urbana-Champaign/ABD'de yürütmemi sağlayan Türkiye'deki bilimsel çalışmaların öncüsü TÜBİTAK kurumuna, ABD'de Beckman Enstitüsü Yapay Zeka ve Bilgisayarla

Görme laboratuvarı yöneticisi Prof. Dr. Narendra AHUJA'ya sonsuz teşekkür ve saygılarımı sunarım.

Son olarak, bugünlere gelmemde büyük emekleri olan sevgili annem ve babam ile yoğun çalışmalarım süresince hiç bir fedakarlıktan kaçınmayan değerli eşim Serpil'e verdikleri destek için çok teşekkür ederim.

Şubat - 2016 Emrah ERGÜL

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR i
İÇİNDEKİLER iii
ŞEKİLLER DİZİNİ v
TABLOLAR DİZİNİ vii
SİMGELER VE KISALTMALAR DİZİNİ viii
ÖZET x
ABSTRACT xi
GİRİŞ 1
1. GENEL BİLGİLER 9
1.1. Yapay Sinir Ağları (YSA) 11
1.2. Çok Katmanlı YSA Yardımıyla Denetimsiz Öznitelik Çıkartımı 14
1.3. Kendine Kodlayıcılar (Auto Encoders - AEs) 15
1.3.1. Gürültü giderici kendine kodlayıcılar (Denoising AEs) 16
1.3.2. Sönümlendirmeli kendine kodlayıcılar (Droput AEs) 17
1.3.3. Seyretmeli kendine kodlayıcılar (Sparcity AEs) 17
1.4. Kısıtlanmış Boltzmann Makineleri (Restricted Boltzmann Machines) 18
1.5. AEs ve RBMs Ağ Yapılarının Çok Katmanlı Mimarilere Dönüşümü 19
1.6. Yapay Sinir Ağı Mimarisinde Öbekleme 20
1.6.1. Basit rekabetçi öğrenme ağı (BRÖA) 22
1.6.2. Self organizing map (SOM) ağı 24
1.6.3. Learning vector quantization (LVQ) ağı 25
1.6.4. Neural gas (NG) ağı 26
1.6.5. Diğer sinir ağı tabanlı öbekleme algoritmaları 28
1.7. Görüntü Tanımada Öznitelik Vektörlerinin Kullanımı 29
1.8. Niteliklerin Orta Seviye Veri Temsilinde Kullanımı 31
1.8.1. Nitelik türleri konusunda literatür taraması 34
1.8.2. Nitelik çıkartımı konusunda literatür taraması 35
1.8.3. Nitelik uygulamaları konusunda literatür taraması 39
2. KAPSAM VE ÖNERİLEN YÖNTEMLER 42
2.1. Melez Sinir Ağları Mimarisinde Öbekleme 44
2.1.1. Çalışmanın literatüre özgün katkısı 46
2.1.2. Prototip kodlama ağı 47
2.1.3. Öbeklerin ayarlanmasında destek vektörleri makinesi ağı 51
2.1.4. PKA ve DVMA mimarileri ile melez ağ yapısında öbekleme 55
2.2. Derin Ağ Yapısında Hiyerarşik Görüntü Temsili 58
2.2.1. Çalışmanın literatüre özgün katkısı 59
2.2.2. Önerilen çok katmanlı ağ mimarisi 60
2.2.3. K-ortalamalar algoritmasının ağ yapısında taklidi 61
2.2.4. Görüntü verisinin hiyerarşik GKS uzaylarında temsili 63
2.3. Denetimsiz Göreceli Nitelik Çıkarımı 66
2.3.1. Denetimsiz göreceli nitelik öğrenme ve imge sınıflandırma 68
2.4. Göreceli Nitelikler Yardımıyla Görsel Tanımada Artırımlı Öğrenme 71

2.4.1. Çalışmanın literatüre özgün katkısı 74
2.4.2. Başlangıç ayarları hakkında ön bilgiler....75
2.4.3. Görsel tanımada öğrenci – öğretmen etkileşimi 78
2.4.4. Göreceli nitelikler uzayının yeniden ayarlanması 82
2.4.5. Sınıf modellerin güncellemesinde ağırlıklı adayların seçimi 83
2.4.6. Sınıf modellerin güncellenmesi 86
3. DENEYSEL TASARIM VE BULGULAR 90
3.1. Melez Sinir Ağları Mimarisinde Öbekleme Deneysel Tasarımı....91
3.1.1. Görsel öbekleme analiz sonuçları 92
3.1.2. Analitik performans analiz sonuçları 100
3.2. Derin Ağ Yapısında Hiyerarşik Görüntü Temsili Deneysel Tasarım 109
3.2.1. Deneysel bulgular 112
3.3. Denetimsiz Göreceli Nitelik Çıkarımı Deneysel Tasarımı 115
3.3.1. Deneysel bulgular 117
3.4. Görsel Tanımada Artırımlı Öğrenme Deneysel Tasarımı 119
3.4.1. Deneysel bulgular 122
4. SONUÇLAR VE ÖNERİLER....129
4.1. Elde Edilen Sonuçlar....130
4.1.1. Melez sinir ağları mimarisinde öbekleme 130
4.1.2. Derin ağ yapısında hiyerarşik görüntü temsili....133
4.1.3. Denetimsiz göreceli nitelik modellemesi 134
4.1.4. Göreceli nitelikler ile görüntü tanıma....135
4.2. Öneriler....137
KAYNAKLAR....140
KİŞİSEL YAYIN VE ESERLER....148
ÖZGEÇMİŞ....149

ŞEKİLLER DİZİNİ

Şekil 1.1. Görüntü tanımaya yönelik süreç sahaları 10
Şekil 1.2. Hesaplama elementi olan bir nöronun yapısı.....12
Şekil 1.3. Çok katmanlı YSA yapısı 12
Şekil 1.4. Kendine kodlayıcı ağ yapısı.....15
Şekil 1.5. Gürültü giderici kendine kodlayıcı ağ yapısı.....17
Şekil 1.6. Kısıtlanmış boltzmann makineleri ağ yapısı.....19
Şekil 1.7. Basit rekabetçi öğrenme ağı yapısı 22
Şekil 1.8. SOM yapısı 24
Şekil 1.9. Görüntü verisinin temsili 30
Şekil 1.10. İkili nitelik kullanımı.....32
Şekil 1.11. Göreceli nitelik kullanımı.....33
Şekil 1.12. İkili veya reel sayılı nitelik-sınıf matrisi 34
Şekil 1.13. Nitelik tabanlı görüntü tanıma üzerine literatür özeti 34
Şekil 2.1. Öbekleme için önerilen çok katmanlı kompozit ağ mimarisi.....45
Şekil 2.2. Ptototip kodlayıcı ağ mimarisi.....48
Şekil 2.3. DVM sınıflandırıcısı ve destek vektörleri 51
Şekil 2.4. Öbekleri ince ayarlayan DVMA mimarisi.....52
Şekil 2.5. Açgözlü katman tabanlı melez ağ mimarisine ait pseudo-code.....56
Şekil 2.6. Açgözlü katman tabanlı kompozit ağ mimarisi akış diyagramı.....57
Şekil 2.7. Hiyerarşik GKS uzayında görüntü verisinin temsili 58
Şekil 2.8. Tek katmanlı ağ mimarisinde görsel kelime çıkartımı.....63
Şekil 2.9. İlk katmandaki GKS vektör çıkarımı 64
Şekil 2.10. Mevki bilgisinin dahil edildiği ikinci katman yapısı.....65
Şekil 2.11. Hiyerarşik GKS temsili için önerilen üç ara katmanlı ağ yapısı.....66
Şekil 2.12. Sınıf tabanlı göreceli nitelikler ile modelleme.....68
Şekil 2.13. Rastgele nitelik öğrenmede sınıf tabanlı ikili başlangıç durumu.....69
Şekil 2.14. Sınıflar arası göreceli ilişkilerin öğrenilmesi algoritması 71
Şekil 2.15. Önerilen yöntemin temel esasları.....72
Şekil 2.16. Göreceli niteliklerle öğrenci merkezli öğrenme yöntemi.....74
Şekil 2.17. Öğrenci merkezli öğrenmede başlangıç ayarları.....76
Şekil 2.18. Öğrencinin Uset'ten maksimum entropi ile örnek seçimi.....79
Şekil 2.19. Görüntü tanıma sisteminde öğrenci-öğretmen etkileşimi.....80
Şekil 2.20. Ağırlıklandırma şeması içerisinde aday örneklerin Uset'ten seçilmesi.....85
Şekil 2.21. Önerilen görüntü tanıma algoritmasının akış diyagramı 89
Şekil 3.1. Değişken karmaşıklığa sahip iki boyutlu sentetik veri setleri 93
Şekil 3.2. Görsel analizlerde kullanılan S_1 ve aggregation veri setleri.....93
Şekil 3.3. WTA ağ mimarilerinin S_1 seti üzerinde öbekleme sonuçları 94

Şekil 3.4. NFB ağ mimarilerinin S_1 seti üzerinde öbekleme sonuçları.....96
Şekil 3.5. S_1 seti üzerinde yetersiz PKA girdileri ile ilklendirilen DVMA öbekleme sonuçları.....97
Şekil 3.6. Aggregation seti üzerinde yetersiz PKA girdileri ile ilklendirilen DVMA öbekleme sonuçları.....98
Şekil 3.7. S_1 seti üzerinde ardışık PKA yapısı ara katman dönüşümleri 100
Şekil 3.8. S_2 seti üzerinde yinelemeli hata trend plotlaması 101
Şekil 3.9. Wine-red seti üzerinde yinelemeli hata trend plotlaması 102
Şekil 3.10. Wine-white seti üzerinde yinelemeli hata trend plotlaması.....102
Şekil 3.11. S_3 seti üzerinde öbekleme algoritmalarının Silhouette analizi 106
Şekil 3.12. Wine-red seti üzerinde öbekleme algoritmalarının Silhouette analizi.....107
Şekil 3.13. Wine-white seti üzerinde öbekleme algoritmalarının Silhouette analizi.....107
Şekil 3.14. CIFAR-10 veriseti.....110
Şekil 3.15. Çok katmanlı ağ mimarisi deneysel tasarımı.....112
Şekil 3.16. Farklı boyutlarda öznitelik vektörleriyle performans analizi 114
Şekil 3.17. Denetimsiz göreceli nitelik çıkarımı verisetleri özeti 116
Şekil 3.18. Denetimsiz göreceli nitelik çıkarımı deneysel tasarım detayı 116
Şekil 3.19. OSR veriseti üzerinde performans karşılaştırmaları 118
Şekil 3.20. PubFig veriseti üzerinde performans karşılaştırmaları.....118
Şekil 3.21. Görsel tanımada artırımlı öğrenme için kullanılan verisetleri.....120
Şekil 3.22. Değişen miktarda nitelik ile başarı performansı 123
Şekil 3.23. Verisetleri üzerinde yineleme oranına göre performans analizi 125
Şekil 3.24. Verisetleri üzerinde karşılaştırmalı ROC analizleri 126

TABLOLAR DİZİNİ

Tablo 3.1. Deney tasarımında kullanılan veri setlerinin özet tablosu.....91
Tablo 3.2. Sentetik verisetlerine ait MSE ve zamanlama analiz sonuçları.....103
Tablo 3.3. UCI verisetlerine ait MSE ve zamanlama analiz sonuçları.....103
Tablo 3.4. Bütünlük ve ayrışım temelli validasyon analiz sonuçları.....105
Tablo 3.5. Silhouette değerleri üzerinden karşılaştırmalı analiz sonuçları.....108
Tablo 3.6. Sentetik verisetlerinde başarı ve saflık analiz sonuçları.....109
Tablo 3.7. UCI verisetlerinde başarı ve saflık analiz sonuçları.....109
Tablo 3.8. Piramid formunda vektörlerinin sınıflandırma analiz sonuçları.....113
Tablo 3.9. CIFAR-10 test veriseti üzerinde sınıflandırma sonuçları.....113
Tablo 3.10. Yama boyutlarının başarı oranına etkisi.....114
Tablo 3.11. Algoritmaların verisetleri üzerindeki performansları.....128

SİMGELER VE KISALTMALAR DİZİNİ

$h_{jw}(t)$: Gaussian Komşuluk Fonksiyonu
$h(r,t)$: Neural Gas Sıralama Çarpanı
$\eta(t)$: Öğrenme Katsayısı
L_2 : İkinci Düzey Vektör Normu
$\sigma(t)$: Gaussian Ölçekleme Faktörü
μ : Kronecker Delta Parametresi
W : Yapay Sinir Ağları Ağırlık Parametreleri

Kısaltmalar

AEs : Auto-Encoders (Kendine Kodlayıcılar)
ANN : Artificial Neural Networks (Yapay Sinir Ağları)
ART : Adaptive Resonance Theory (Uyumlanabilir Rezonans Teorisi)
BINs : Binary Discriminants (İkili Ayırtaçlar)
BoW : Bag-of-Words (Görsel Kelimeler)
BRÖA : Basit Rekabetçi Öğrenme Ağı (Simple Competitive Learning)
C4.5 : C4.5 Karar Ağacı (C4.5 Decision Tree)
CNNs : Convolutional Neural Networks (Konvolüsyon Tabanlı Sinir Ağları)
DAEs : Denoising Auto Encoders (Gürültü Giderici Kendine Kodlayıcılar)
DAP : Direct Attribute Prediction (Doğrudan Nitelik Tahmini)
DBNs : Deep Belief Networks (Derin Varsayım Ağları)
DN : Doğru Negatif (True Negative)
DP : Doğru Pozitif (True Positive)
DVM : Destek Vektör Makineleri
DVMA : Destek Vektör Makineleri Ağı
EM : Expectation-Maximization (Beklenti – Maksimize etme)
FC : Fuzzy Clustering (Bulanık Öbekleme)
FFNN : Feed Forward Neural Networks (İleri Beslemeli Sinir Ağları)
FLD : Fisher Linear Discriminant (Fisher Doğrusal Ayırtaç)
GKF : Gaussian Kernel Fonksiyonu
GKS : Görsel Kelimeler Sözlüğü (Bag-of-Words -BoW)
HOG : Histogram of Oriented Gradients (Yönlü Gradyen Histogramı)
KL : Kullback-Leibler Divergence (Ayrışması)
KNN : K-Nearest Neighbor (K En Yakın Komşu)
LVQ : Learning Vector Quantization Network (Vektör Kuantizasyon Ağı)
ME : Maximum Entropy (Maksimum Belirsizlik)
MSE : Mean Squared Error (Ortalama Karesel Hata)
NFB : Neighborhood Function Based (Komşuluk Fonksiyon Tabanlı)
NG : Neural Gas Network (Sinir Gaz Ağı)
OGTA : Olasılıksal Gizli Tema Analizi (Probabilistic Latent Semantic Analysis)
OKH : Ortalama Karesel Hata

OSR : Outdoor Scene Recognition (Dış Mekan Tanıma)
PCA : Principal Component Analysis (Ana Bileşke Analizi)
PE : Prototype Encoding (Prototip Kodlayıcı)
PHOG : Pyramid Histogram of Oriented Gradients (Yönlü Piramid Histogram Gradyenleri)
PKA : Prototip Kodlama Ağı
PLSA : Probabilistic Latent Semantic Analysis (Olasılıksal Gizli Tema Analizi)
PubFig : Public Figure Face (Tanınmış İnsan Yüzü)
RBF : Radial Basis Funciton (Radyal Tabanlı Fonksiyon)
RBMs : Restricted Boltzmann Machines (Kısıtlanmış Boltzmann Makineleri)
ROC : Receiver Operating Characteristic (Alıcı Çalışma Karakteristiği)
SAEs : Stacked Auto Encoders (Yığıt Kendine Kodlayıcılar)
SAT : Supervised Attribute (Denetimli Nitelik)
SCL : Simple Competitive Learning (Basit Rekabetçi Öğrenme Ağı)
SIFT : Scale Invariant Feature Transform (Ölçek Bağımsız Nitelik Dönüşümü)
SOM : Self-Organizing Map (Kendinden Düzenlemeli Haritalama)
SSAC : Semi Supervised Attribute Classification (Yarı Denetimli Nitelik Sınıflaması)
SURF : Speeded Up Robust Features (Hızlandırılmış Gürbüz Öznitelikler)
SVM : Support Vector Machines (Destek Vektör Makineleri)
UAT : Unsupervised Attribute (Denetimsiz Nitelik)
UWC : Uset Weighting Cube (U Veriseti Ağırlık Küpü)
WTA : Winner-Takes-All (Kazanan Hepsini Alır)
YN : Yanlış Negatif (False Negative –FN)
YP : Yanlış Pozitif (False Positive –FP)
YSA : Yapay Sinir Ağları (Artificial Neural Networks)

MEVKİSEL VE ANLAMSAL GÖRECELİ NİTELİKLER YARDIMIYLA GÖRÜNTÜ TANIMA

ÖZET

Bu tezin amacı, görüntü verisinin içeriğine uygun olarak tanımlanmasını sağlayacak yeni bir makine öğrenme algoritması geliştirmektir. Önerilen algoritma; ilk aşamada görüntü tanıma probleminde sıkça kullanılan görsel kelimeler sözlüğünü denetimsiz şekilde elde ederek ham veriyi öncelikle düşük seviye ve boyutta yeni bir öznitelik uzayına taşımak amacıyla derin sinir ağı yapısı kullanmıştır. Auto-Encoder gibi denetimsiz öğrenmeyi sağlayan bir yapay sinir ağı mimarisinde, ara katmanlarda yer alan düğümlere ait ağırlık vektörleri öbekleme algoritması içerisinde prototip vektörlerini temsil etmiştir. Elde edilen prototip vektörleri görsel kelimeler sözlüğünü oluşturmuş ve bir benzerlik metriği kullanılmak suretiyle görüntü verisi düşük seviye ve boyutta öznitelik vektörleri ile ifade edilmiştir.

Ayrıca ikinci aşamada, öğrenci-öğretmen etkileşimli bir yapı içerisinde başlangıçta denetimsiz olarak oluşturulan kategori tabanlı göreceli nitelik uzaylarını sınıf modellerine uygun şekilde ve eş zamanlı güncelleyecek özgün bir yöntem geliştirilmiştir. Daha sonra, hedef kategoriler yarı denetimli olarak etkileşimli ve arttırımlı bir yapıda, görsel kelimeler ve göreceli niteliklerle birlikte öğrenilerek görüntü sınıfları modellenmiştir.

Tez çalışması neticesinde, bir görüntü içeriğindeki nesne ve nitelikleri dinamik ve çok kriterli ortamda etkin bir şekilde tanıyan sınıflandırma sistemi geliştirilmiştir. Göreceli nitelikler yardımıyla daha geniş bir ifade özgürlüğüne kavuşulacağından, insan-makine etkileşiminde takviyeli öğrenmenin görüntü analizine yansımaları ayrıca değerlendirilmiştir. Görüntü tanımaya/sınıflandırmaya yönelik bulunacak çözüm önerilerinin uygulama sahası olarak yüksek çözünürlüklü uydu görüntülerinin, kızılötesi görüntülerin, medikal görüntülerin, istihbarat maksatlı kullanılan hava araçlarından anlık olarak iletilen görüntülerin ve sualtı ses sinyallerinin analizi kapsamında da kullanılabileceği değerlendirilmektedir.

Anahtar Kelimeler: Arttırımlı Öğrenme, Denetimsiz Öğrenme, Göreceli Nitelikler, Görüntü Tanıma, Yapay Sinir Ağları.

VISUAL RECOGNITION VIA SPATIALLY AND SEMANTIC RELATIVE ATTRIBUTES

ABSTRACT

The aim of this thesis is to develop a new machine learning algorithm which leads to the recognition of visual data relevant to the semantic content. The proposed method implements a multi-layer neural network architecture in the first stage to achieve a dictionary of 'Visual Words' in an unsupervised manner which is often used in visual recognition problems. The visual words are used mainly to convey raw visual data into a new feature space that is low level and provides a comperatively reduced representation. The weight vectors which belong to neurons in the hidden layers of a neural netwok alike Auto-Encoder that help unsupervised feature extraction, actually represent prototype vectors for a clustering algorithm. The achieved prototype vectors constitute a dictionary of visual words, and the visual data are represented with low level and reduced feature vectors using a similarity metric.

Additionaly in the second step, a new learning method is developed to update category based relative attributes space, initialized with an unsupervised way, according to and concurrent with the class models in a structure of student-teacher interaction. Thereafter, the visual classes are modeled as the target categories are learnt incrementally by means of visual words and relative attributes in a semi-supervised and interactive structure.

At the end of this thesis, a visual classification system is developed by means of recognizing the objects and the attributes in its contents effectively at multi-criteria environment. Not only can we analyze the scene/object categories that are included in the training stage, but also new classes which are introduced into the system at the test stage can be recognized by the learnt relative attributes. Additionally, the effects of reinforced learning to the vision processing are evaluated at the human-machine interaction since much more freedom of expression is achieved by means of relative attributes. We think that the proposals for solution of visual recognition/classification problems can indeed be used in the interest areas of high resolution satellite images, infra-red visions, medical images, unmanned aerial vehicle surveilance images and underwater acoustic signal analyzing.

Keywords: Incremental Learning, Unsupervised Learning, Relative Attributes, Visual Recognition, Artificial Neural Networks.

GİRİŞ

Bilgisayar teknolojisinin gelişmesiyle birlikte; günümüzde çok büyük miktarda veriyi kısa zamanda üretme, saklama ve işleme yeteneğine sahibiz. Ayrıca kurduğumuz çeşitli ölçekteki iletişim ağları yardımıyla veriye istediğimiz yerden istediğimiz zaman ulaşabiliyoruz. Bunun bir parçası olarak özellikle son on yılda video görüntüleri ve imgelerin üretilme hızı inanılmaz boyutlara ulaşmış ve uygulamaları çok çeşitli askeri ve endüstriyel sahalara yayılmıştır. Yüksek çözünürlüklü görüntüleme sistemlerine sahip uyduların, arama ve takip platformlarının, kameraların, fotoğraf makinelerinin, cep telefonlarının, taşınabilir kişisel bilgisayarların gelişmesi, yüksek hafıza ve işlemci kapasiteli bilgi işlem sistemlerinin ve makinelerin ortaya çıkmasıyla birlikte; insanlar çeşitli formatlarda görüntüyü her an üretebilme ve paylaşabilme yeteneğine kavuşmuştur. Örneğin, istatistiksel veriye dayanarak Flickr adlı resim ve video paylaşım sitesinin 2005 Haziran ayında sadece 1 milyon kayıtlı kullanıcısı ve 19,5 milyon imge kapasitesi mevcutken [1]; sadece 2 yıl sonra 5 milyon müşteri ve 250 milyon imge kapasitesine [2], 2015 yılında ise 112 milyon kayıtlı kullanıcı, 10 milyar imge kapasitesi ve günlük ortalama 1 milyon imge paylaşım seviyesine ulaştığı tespit edilmiştir [3]. Bu durum bilgi toplumunun gelişmesi bakımından ilk bakışta cesaret verici gibi gözükse de; çok büyük miktardaki verinin bilgiye dönüştürülmesi kapsamında önümüzde acilen çözülmesi gereken ciddi problemler mevcuttur. Çünkü verinin üretilme hızına paralel olarak aynı veya yakın süratte verinin yönetilmesi ve karar desteği maksadıyla bilgiye dönüştürülmesi prensibi benimsenmezse, veri öbekleri içerisinde yolumuzu kaybedeceğimiz ve veri çöplüğünde boğulacağımız açıktır.

Her ne kadar insanlar internette anlamsal filtreleme ve arama maksadıyla imgelere/videolara gittikçe artan oranlarda etiket bilgisi yerleştirse de; halen çoğu görüntü verisi istenilen seviyede etiketlenememiştir. Etiket bilgisi insanların bakış açılarına göre göreceli olarak yerleştirildiğinden, ortak bir bilişsel yöntem saptanamadığından, çoğu zaman etiket bilgisine rağmen hedeflenen görüntü verisinin elde edilmesi mümkün olmamaktadır. Bu yüzden görüntü verisinde mümkün

olduğunca otomatik tanıma, etiketleme ve arama tekniklerinin geliştirilmesine ihtiyaç duyulmaktadır. Görüntü verisinin anlamsal içerik temelli ifade edilmesi ile birlikte, büyük miktarda verinin makul şekilde organize edilmesi, saklanması ve çıkarımlar yolu ile bilginin elde edilmesi mümkün olacaktır.

Görüntü verisi üzerinde tatmin edici bir başarı yüzdesiyle, denetimsiz veya düşük denetimle sahne/nesne tanıma/sınıflandırma işlemini otomatik şekilde yapabilen bir bilgi sistemi karşılanması gereken önemli bir ihtiyaçtır. Böyle bir yeteneğe sahip sistem özellikle ortam araması, çevre algılaması, robot uygulamaları, uydu görüntüleme, hastalık tanıma, istihbarat toplama vb. pek çok karar destek uygulamasında kullanılabilecektir. Özellikle sahne hakkında elde edilecek bilgi görüntü analizini daha akıllı hale getirecektir. Bilgisayarın görüntü verisindeki nitelik kategorisini ayırt etmesiyle birlikte imgede yer alan nesnelerin birbirleriyle olan anlamsal ilişkileri de çözülebilecektir. Bu ilişkiler yardımıyla sahnenin aktörleri konumunda tespit edilen nesnelerin faaliyetleri 'Etkinlik Tanıma' kapsamında algılanabilecek ve mevcut durumdan bir sonraki hedeflenen duruma erişebilmek için yapılması gereken harekete karar verilebilecektir.

Görüntü verisinin analiz edilerek içerik bilgisine göre anlamlandırılması problemi daima ilgi çeken bir çalışma alanı olmuştur. Son zamanlarda çok kategorili görüntü sınıflandırma probleminin çözülmesi maksadıyla değişik algoritmalar ve test veri setlerinin yayımlanması [4-11] görüntü analizine ne denli dikkat çekildiğini göstermektedir. Büyük boyutlara ulaşan veri kümeleri incelendiğinde problemin ne kadar zor olduğu, hatta bazen insanı bile karar vermede tereddüde düşürecek hedef kategorilerin bulunduğu görülebilir. Aynı sınıfa ait örneklerin birbirinden çok farklı niteliklere sahip olduğu, farklı sınıftan örneklerin ise benzer niteliklere sahip olduğu gözlemlenebilir. Hedef kategoriler arasındaki temel farklılık ise verinin bütünsel içerik bilgisinden kaynaklanmaktadır. Dolayısıyla bu alandaki en önemli problem verinin temsil edileceği düşük/orta seviye öznitelik uzaylarını oluşturmak ve hedef sınıfları bu uzaya uygun şekilde modellemektir.

Çok kategorili görüntü tanıma problemine yönelik test veri setleri 2000'li yılların başında 8-15 sınıf üzerinde yoğunlaşırken, günümüzde üzerinde çalışılan sınıf sayısı binlere ulaşmış ve beklentiler artmıştır. Bunun nedeni analitik/istatistiki yöntemlerle

elde edilen desen, histogram, spektrum ve süzgeçleme cevaplarının düşük seviye öznitelik vektörleri olarak doğrudan görüntü tanımada kullanılması yerine artık denetimsiz, hiyerarşik ve anlamsal içerikli daha ayırt edici betimleyicilerin eğitim ve validasyon algoritmalarıyla eş zamanlı öğrenilmesidir. Tüm bu sayılan hususlar göz önünde bulundurularak ölçeklenebilir, optimum denetimli, istenilen başarı yüzdesini sağlayacak ölçüde etkin ve hafıza/işlem yükü az olan bir görüntü tanıma metodolojisinin problemin tanımını oluşturduğunu söyleyebiliriz.

Görüntünün tanımlanmasında genel olarak 5 alt problem sahasına el atılması gerekmektedir. Bunlardan ilki verinin bir ön işlemeye tabi tutulmasıdır. Morfolojik işlemler, istenen band geçişlerini yapmak için filtreleme, Fourier/Laplace dönüşümleri sonrasında elde edilen yüksek frekanslı bölümlerin silinmesi, normalizasyon ve enterpolasyon görüntü işlemede ön plana çıkan yöntemlerdir. İkinci problem sahası verinin gösteriminin nasıl olacağının belirlenmesidir. Verinin ham haliyle doğrudan algoritmalarda kullanılması veri kaybı endişesiyle başlangıçta iyi bir fikir gibi görülse de; hafıza ve işlem yükü açılarından istenmeyen sonuçlara gebedir. Bunun yanında anlamsal olarak aynı sınıfı ifade etmesine rağmen ham veri bulunduğu ortama uyum sağlamak için bir takım dönüşümlere tabi tutulmuştur. Bu yüzden örüntü analizi algoritmalarında görüntü verisinin rotasyon, bakış açısı, ölçekleme, deformasyon, örtüşme, yer değiştirme ve aydınlatma gibi bir takım değişmelere karşı gürbüz olmasını sağlayacak düşük seviye boyut indirgeme yöntemleriyle dönüşümü sağlanır. Öznitelik çıkartımı ve alt küme seçimi kapsamında renk, desen, yönelim, histogram ve frekans spektrumu analizi başlıca kullanılan yöntemlerdir.

Üçüncü problem sahası düşük seviye veri gösteriminin görüntü tanıma gibi karmaşık problemleri çözmeye yetmemesidir. Düşük seviye öznitelik vektörleri ham veriye dayandığından sadece görsel ölçümlere dayanan istatistikleri sunmaktadır. Oysa verinin ardında yatan anlamsal gösterim çoğu zaman daha mühimdir. Burada maksat denetimli veya denetimsiz bir şekilde konsept veya nitelikler yolu ile imgeyi orta seviyelerde temsile çalışmaktır. Bu kapsamda Görsel Kelimeler Sözlüğü (GKS) [12], Olasılıksal Gizli Tema Analizi (OGTA) [6, 13] ve Göreceli Nitelikler [7, 14] ön plana çıkan çalışmalardır.

Dördüncü problem sahası ise sınıflandırmaya ve veri kümesine uygun hipotezlerin ortaya konulduğu öğrenme safhasıdır. Öğrenmek hipotez uzayımıza uygun şekilde eğitim veri kümesi üzerinde kurmaya çalıştığımız modelin parametrelerini optimize etmektir. Öğrenmede elde edilecek hatanın en aza indirgenmesi test aşamasında da benzer başarının elde edileceği anlamına gelmemektedir. Bu problemler temelde parametrik [15], yarı parametrik [12] ve parametrik olmayan [16] yöntemler kullanılarak çözülür. Son problem sahası ise öğrenilen modellerin test edilmesi ve sonuçların analiz edilmesidir. Bir problemin çözümünde sadece tek bir yol olamayacağı ve bir çözümün tüm uygulamalarda geçerli olamayacağı prensibinden hareketle literatürde görüntü tanımaya yönelik pek çok metod önerilmiştir. Bu metodların başarı yüzdeleri ele alınan örnek verisinin doğasına göre değişiklik gösterebilmektedir. Bu nedenle araştırmalarda algoritmaların mukayesesi ve performans değerlendirmesi ayrıca önemlidir. Diğer taraftan nesne/sahne sınıflandırma gibi karmaşık problemlerin çözümünde çoğu zaman izlenen yol tek bir algoritma ile karmaşık örnek uzayının modellenmesi yerine, örnek uzayının daha sade alt uzaylara taşınması, bu yeni uzaylarda basit algoritmalar yardımıyla alt problemlerin çözülmesi ve sonuçların birleştirilmesidir. Bu nedenle algoritmaların ikili veya çoklu mukayeselerinin yapılarak birbirlerini destekleyecek şekilde melez bir yapı içerisinde birlikte kullanılması teşvik edilen bir husustur.

Neticede, çoğu zaman verinin yaratılışı sürecinin altında yatan işleyişinin detaylarına sahip değilizdir. Ancak şunu da iyi biliriz ki; ortam gürültülü olsa da hiçbir şey tamamen rastgele üretilmez. Verinin içinde üretildiği dünyanın verinin yaratılışına etki eden parametrelerini matematiksel olarak ifade ederek, tam olmasa da kabul edilebilir kayıplarla veriyi ifade edebiliriz. Bu yüzden biz veriyi önce depolar ve gürültüyü azaltır, sonra veriyi anlamamızı sağlayan belirgin örüntüleri analizler yolu ile keşfederiz. Neticede bu örüntülerden modeller öğrenir ve gelecekteki durumları tahmin etmek için öğrendiğimiz bilgileri sistemlerimizde kullanabiliriz.

Bu teoriden hareketle bahse konu tezin amacı; görüntü verisinin içeriğine uygun olarak tanımlanmasını sağlayacak yeni bir makine öğrenme algoritması geliştirmektir. Önerilen algoritma; ilk kısımda görüntü tanıma probleminde sıkça kullanılan GKS'yi denetimsiz şekilde elde ederek ham veriyi öncelikle düşük seviye ve boyutta yeni bir öznitelik uzayına taşımak amacıyla derin sinir ağı yapısı

kullanmaktadır. Ayrıca ikinci kısımda, öğrenci-öğretmen etkileşimli bir yapı içerisinde başlangıçta denetimsiz (rastgele) olarak oluşturulan kategori tabanlı göreceli nitelik uzaylarını sınıf modellerine uygun şekilde ve eş zamanlı güncelleyecek özgün bir yöntem sunmaktadır. Daha sonra, hedef kategoriler yarı denetimli olarak interaktif ve arttırımlı bir yapıda göreceli niteliklerle birlikte öğrenilerek görüntü sınıfları modellenmektedir. Bu temel amaç doğrultusunda şu alt hedefler belirlenmiştir: Veri bütünlüğünü sağlayan bir görsel kelimeler sözlüğünü denetimsiz şekilde elde ederek ham veriyi öncelikle düşük seviye ve boyutta yeni bir öznitelik uzayına taşımak (verinin temsili), Öğrenci-öğretmen etkileşimli bir yapı içerisinde başlangıçta denetimsiz olarak oluşturulan kategori tabanlı göreceli nitelik uzaylarını sınıf modellerine uygun şekilde ve eş zamanlı güncellemek (verinin anlamlandırılması), Hedef kategorilerini interaktif ve arttırımlı bir yapıda yarı denetimli bir mekanizda, görsel kelimeler ve göreceli niteliklere uygun şekilde öğrenmek (sınıf modelleme), Görüntü verisinin tanımlanmasında başarı performansını yükseltmek için düşük ve orta seviye örüntülerin görüntü üzerindeki mevki bilgisini kullanmak (uzaysal bilgi).

Tez kapsamında geliştirilecek sistemin katma değeri hususunda öğrenilmiş nitelikler çeşitli uygulamalarda farklı koşullanmalarla ele alınabilir. Nitelikler tıpkı hedef kategoriler gibi insanların zihinlerinde anlamlandırılabilen ortak kavramlardır ve görüntüleri betimlemekte kullanılırlar. Bu kapsamda niteliklerden sadece öğrenildikleri eğitim veri kümelerinde yer alan hedef kategorileri değil, eğitimde kullanılmayan diğer sınıfları da tarif etmede faydalanılabilir. Dolayısı ile tez kapsamında öğrenilecek niteliklerin kategoriler arasında paylaşılacak ortak bir sözlük oluşturabileceği değerlendirilmektedir. Bu amaç ve alt hedeflere uygun şekilde tezin kapsamı ise şu şekilde belirlenmiştir: Çok katmanlı bir yapay sinir ağında öbekleme algoritması geliştirilerek ara katmanlardan elde edilen örüntülerin görsel kelimeler sözlüğünü oluşturması, Görüntü verisinin bütününden elde edilecek görsel kelimeler sözlüğünün mevkisel düzlemde farklı bölgelerde düşük seviye öznitelik vektörlerine dönüştürülmesi, Hedef kategoriler ile eş zamanlı göreceli niteliklerin öğrenilmesi, İnteraktif öğrenme yapısı içerisinde öğrenci ve öğretmen makinelerin göreceli nitelikler üzerinden birbirleri ile karşılıklı soru-cevap şeklindeki etkilişimi.

Tezin gerçeklenmesiyle elde edilecek nihai görüntü tanıma algoritmasında: Klasik yarı denetimli öğrenmeden farklı olarak bir örneğin yinelemeler esnasında birden çok kez kullanılması ve bu sayede öğrenmede örnek sayısının sanal olarak arttırılması (yanlılık azaltımı), Sınıf modellerinin öğrenilmesine en çok katkıyı sağlayacak şekilde ve sırada örneklerin sisteme sokulması, Her bir iterasyonda öğrenmeyi geliştirecek doğrultuda göreceli niteliklerin güncellenmesi, Güncellenen göreceli niteliklere uygun olarak yeni uzayda sınıf modellerin güncellenmesi (değişinti azaltımı) öngörülmektedir.

Tezin yan ürünü olarak elde edilecek özgün bir "Görsel Nitelik Sözlüğü" ile ilgili olarak; eğitimde kullanılmamış ancak nitelikleri bilinen kategorilerin tanınması (zero-shot learning) [17, 18], ne eğitimde kullanılmış ne de nitelikleri bilinen yeni sınıfların teşhisi (unfamiliar class detection) [19], görüntüdeki anormalliklerin tespiti (hedef kategoride olması gereken bir niteliğin bulunmayışı veya tam tersi) (anomaly detection) [20] ve karmaşık bir görüntü içeriğinin sözel olarak tanımlanması [21-23] gibi uygulama alanlarında kullanılabileceği değerlendirilmektedir.

Buraya kadar bahsedilen amaç ve kapsama uygun olarak tez çalışması üç aşamada yapılmıştır. Birinci aşamada, verinin temsil edildiği uzayın modellemede en büyük etken olduğu prensibinden hareketle, denetimsiz öznitelik vektörlerinin eldesinde çok katmanlı Yapay Sinir Ağı (YSA) mimarileri üzerinde durulmuştur. Bu kapsamda öncelikle denetimsiz öznitelik çıkartımı için geliştirilen ve 'Derin Öğrenme' adı ile anılan çok katmanlı YSA mimarileri üzerine literatür taraması yapılmıştır. Müteakiben veriyi orta seviyede temsil etmek üzere GKS'nün esasını oluşturan görsel kelime vektörlerinin öbekleme yoluyla elde edilmesine çalışılmıştır. Öbeklemede kullanılan diğer YSA mimarileri incelenmiş ve tezin ilk ürünü olarak iki farklı ve birbirini tamamlayıcı ağ mimarisi tasarlanmıştır: Prototip Kodlama Ağı (PKA) ve Destek Vektör Makineleri Ağı (DVMA).

Tezin ikinci aşamasında orta seviye görüntü verisinin temsilinde diğer bir popüler yaklaşım olan niteliklerin öğrenilmesi üzerinde durulmuştur. Literatürdeki nitelik tabanlı görüntü tanıma çalışmaları genel olarak ikili (bir örnekte niteliğin var olup olmadığı) ve göreceli (bir örnekte niteliğin ne kadar var olduğu) olarak tasniflenmiş ve çalışmalar göreceli nitelik uzaylarının denetimsiz oluşturulması üzerine

yoğunlaştırılmıştır. Tezin bu aşamasında imge kategorileri (sınıfları) ile eş zamanlı öğrenilen rastgele göreceli niteliklerin modellenmesi tasarlanmıştır. İnteraktif bir öğrenme yapısı içerisinde öğrenci ve öğretmen makinelerin göreceli nitelikler üzerinden birbirleri ile karşılıklı soru-cevap şeklindeki etkilişimini sağlayan tasarım neticesinde, göreceli nitelikler ve sınıflandırıcılar yarı denetimli bir mekanizmada öğrenilmiştir.

Üçüncü aşamada ise geliştirilen yöntemlerin literatürdeki öbekleme ve diğer görüntü tanıma algoritmalarıyla mukayeseli bir şekilde analizleri yapılmış ve sonuçları değerlendirilmiştir. Öbekleme ağ mimarileri olarak tasarlanan PKA ve DVMA'nın yanlızca öbekleme hususundaki performans analizleri yapılmamış; aynı zamanda GKS oluşturulmak suretiyle imge sınıflandırmadaki genel performansı da incelenmiştir. İlaveten GKS'nün mevkisel düzlemde farklı bölgelerde düşük seviye öznitelik vektörlerine dönüştürülmesi için gerekli tasarım da oluşturulmuştur. Diğer bir taraftan; görsel nitelik uzaylarının göreceli ve rastgele oluşturulmasında, görüntü ayırtaç fonksiyonlarının nitelik uzayları ile eş zamanlı ve yarı denetimli modellenmesinde genel performansı etkileyecek kontrol parametrelerinin optimizasyonu yapılmış ve sonuçları değerlendirilmiştir.

Tezin ilerleyen bölümleri şu şekilde oluşturulmuştur: Bölüm 1'de, denetimsiz öznitelik çıkartımı kapsamında derin öğrenme, derin öğrenmenin çatısını oluşturan çok katmanlı YSA mimarileri, ağ mimarilerinin öbeklemede kullanılması, orta seviye görüntü verisi temsilinde nitelik uzayları, görsel nitelik uzaylarının oluşturulması ve kullanım sahaları ile ilgili genel bilgiler ele alınmıştır.

Bölüm 2'de öncelikle tez çalışmasında elde edilen ve öbekleme ağ mimarileri olarak tanımlanabilecek PKA ve DVMA'nın tanıtımı yapılmıştır. Bahse konu mimarilerin gerek öbeklemede gerekse görüntü tanımada mevki bilgisini de içerecek şekilde uygulanabilirliği modellenmiştir. Daha sonra veriyi orta seviyede temsil eden göreceli öznitelik uzaylarının denetimsiz oluşturulması ve görüntü kategorilerinin bahse konu göreceli nitelikler üzerinden eş zamanlı ve interaktif bir şekilde nasıl oluşturulacağı ele alınmıştır.

Bölüm 3'te, PKA ve DVMA'nın düşük seviyede öbekleme ve GKS içerisinde üst seviye imge sınıflandırmada testler sonucundaki sınıflandırma başarımları verilmiştir. Aynı şekilde göreceli niteliklerin denetimsiz çıkartımı ve sınıf kategorilerinin orta seviye görsel temsil uzayları ile eş zamanlı ve yarı denetimli öğrenilmesi sonucu performans analizleri ortaya konmuştur. Deneylerde; bahse konu önerilen metodların literatürdeki diğer benzer çalışmalar ile mukayeseli sonuçları verilmiş, referans olarak gösterilen bir çok veri seti üzerinde görsel ve analitik test yöntemleri uygulanmıştır.

Bölüm 4'te, tez çalışması ile elde edilen sonuçlar ve öneriler sunulmuştur.

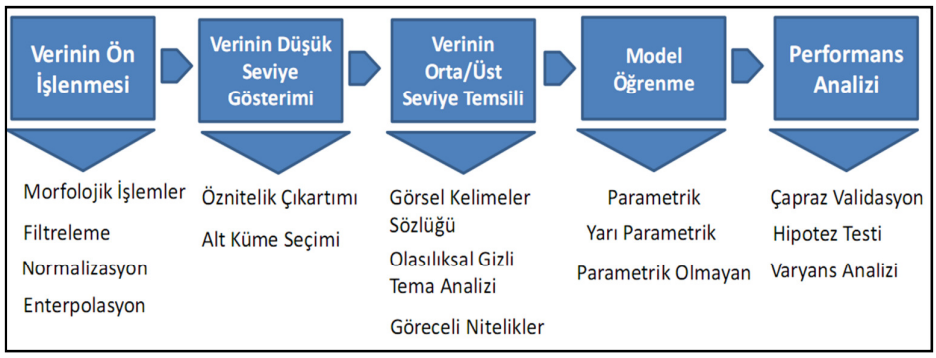
1. GENEL BİLGİLER

Bir öğrenme algoritmasının temel hedefi genellemedir. Genelleme, eğitim safhasında öğrenilen veri modellerinin test aşamasında sistem tarafından daha önceden görülmemiş örnekler üzerinde tatmin edici görev perfomansına ulaşmasıdır. Diğer taraftan modelleme, sistemin giriş sinyali ile arzu edilen çıkışların göreve yönelik olarak ilişkilendirildiği bir fonksiyondur. Başarılmaya çalışılan görev endüstriyel veya bilimsel alanlarda çeşitlilik gösterse de; veri işleme açısından sınıflandırma, yakınsama, gruplama, tespit, seçim veya sıralama türlerinden birisi olabilir. Modelleme kapsamında üretilen fonksiyon ise eğitimde kullanılan veri üzerine oluşturulmuş ayırt edici veya bağlanım sağlayıcı hipotezdir. Bu noktada elde edilecek hipotez fonksiyonunun yanlılık-değişinti ikilemi (bias-variance dilemma)'ne karşı gürbüz olması beklenmektedir [15]. Dolayısı ile öğrenen yapı eğitim verisine ne aşırı uydurmaya neden olacak şekilde karmaşık; ne de yetersiz kalacak şekilde basit olmalıdır.

Geliştirilen modelin başarım performansını etkileyen en önemli faktör göreve yönelik olarak seçilen eğitim verisidir. Eğitim verisinin gerçek dünyayı yansıtacak şekilde çok miktarda ve çeşitlilikte olması beklenir. Çeşitlilik verinin miktarı anlamına gelmemekte olup gelecekte karşılaşılabilecek her türlü durumu yansıtabilme yetisidir. Veri doğası gereği içerisinde gürültü barındıracağı için çeşitlilik sağlanırken verinin ön işlemeye tabi tutulması veya doğrulamasının yapılması da ayrıca önemlidir. İlaveten, eğitim esnasında arzu edilen çıkışların bu küme ile birlikte belirlenmesi gerekeceğinden; eğitim veri setinin oluşturulması anlamına gelen örnekleme iş gücü ve zaman açılarından maliyetlidir.

Veriye ait bir başka husus giriş sinyali olarak verinin nasıl kullanılacağıdır. Ham veri ön işlemeye tabi tutulsa dahi algılayıcılardan temin edilerek sayısal ortama aktarıldığında çoğunlukla yüksek boyutludur. Örneğin, 200x200 ebatlarında 3 renk kanalına sahip bir imge piksel seviyesinde 120.000, 200 Hz. örneklemeli 3 dakikalık bir ses sinyali ise tek kanalda 36.000 boyutlu bir vektöre karşılık gelmektedir. Bu kadar yüksek boyutlu veriyi modelleyecek bir yapının karmaşıklığı, eğitimde

kullanılan örnek sayısının da fazlalığı dikkate alınırsa genellikle yönetilemez. Ayrıca, bahse konu veri boyutlarının çoğu hedefe yönelik olarak anlamsız olup tek başlarına ya da mevcut ilk halleri ile bir anlam ifade etmez. Bu nedenle verinin daha anlamlı, çok daha az boyutlu düşük ve orta seviyelerde öznitelik uzaylarına taşınması gerekmektedir. Görüntü tanımaya yönelik buraya kadar bahsedilen süreç sahaları Şekil 1.1'de gösterilmiştir. Sonuç olarak; modellemede aşırı uydurmaya karşı çok örnekle eğitimin, ayırt edici veri temsil uzayının ve maliyet etkinliği sağlamak üzere mümkün olduğunca az denetimin altının çizilmesi gerekmektedir.



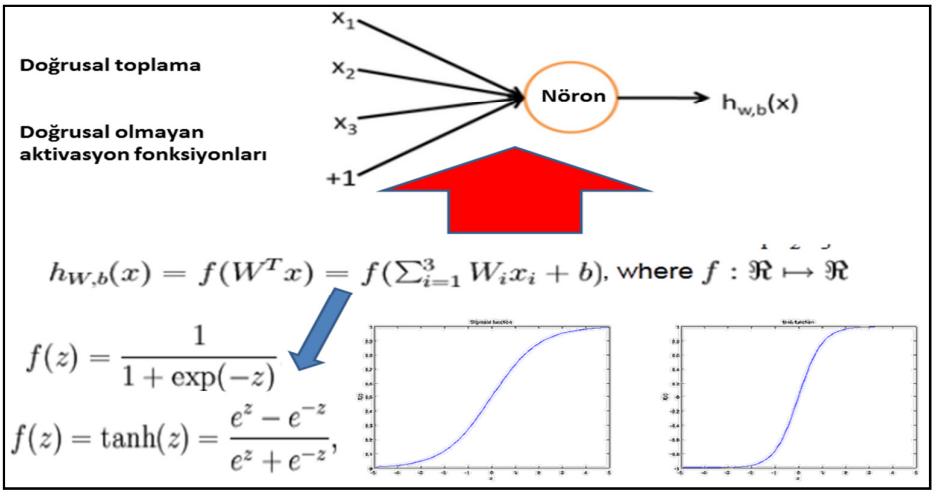
Şekil 1.1. Görüntü tanımaya yönelik süreç sahaları

Tezin ilerleyen bölümlerinde özellikle görüntü verisinin daha ayırt edici temsil uzaylarında tanımlanması üzerinde durulacaktır. Temsil uzaylarının öğrenimi ham veriden daha kullanışlı bilginin çıkartımını sağlamak üzere dönüşümlerin veya istatistiksel modeller için ardıl olasılık dağılımlarının tespit edilmesi anlamına gelmektedir [24]. Denetimsiz bir yapı içerisinde büyük miktarda etiketsiz veri kullanılmak suretiyle, öğrenme algoritmaları istatistiki değerlerden ziyade örneklere dayanan bir forma sokulabilir. İlaveten, giriş verisinin seçildiği uzay hedef öznitelik uzayı ile örtüşmeyebilir. Bu noktada çok katmanlı yapay sinir ağları raftaki en başarılı öznitelik çıkartımı metodu olarak ön planan çıkarılabilir. Görüntü tanımada kullanılan Scale Invariant Feature Transform (SIFT) [25], Speeded Up Robust Features (SURF) [26], Pyramid Histogram of Oriented Gradients (PHOG) [27] veya Gist [10] gibi belirli kalıplar (süzgeçler) veya istatikler üzerine oluşturulmuş veri tanımlayıcıların aksine YSA her bir katmanında daha kullanışlı öznitelikleri herhangi bir ilave iş gücü olmaksızın elde edebilmektedir.

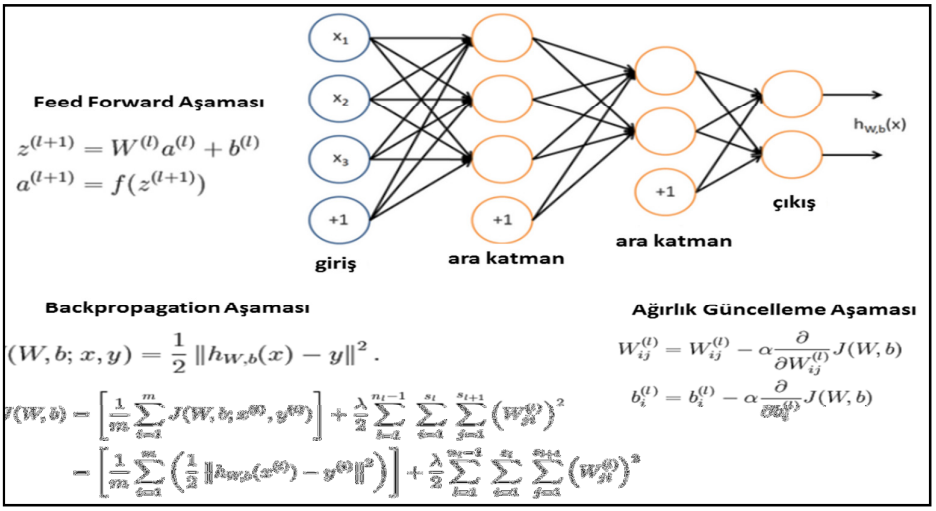
Tez çalışmasında elde edilecek nihai ürün olan görüntü tanıma algoritması içerisinde üç alt hedef bulunmaktadır: Çok katmanlı bir yapay sinir ağı mimarisinde öbekleme metodu geliştirerek görsel kelimeler sözlüğünü denetimsiz elde etmek ve bu sözlüğü düşük seviye öznitelik uzayı olarak kullanmak, hedef görüntü kategorileri ile eş zamanlı olarak sınıf tabanlı göreceli nitelikleri denetimsiz öğrenmek ve bu nitelikleri orta seviye öznitelik uzayı olarak kullanmak, öğrenci-öğretmen etkileşimli bir yapı içerisinde göreceli nitelikleri cevaplarda açıklama olarak kullanmak suretiyle görüntü sınıflarını yarı denetimli modellemek. Bu kapsamda öncelikle literatürde yapay sinir ağlarının öbeklemede nasıl kullanıldığına değinilecek, müteakiben niteliklerin görüntü tanımada kullanılış usülleri tartışılacaktır.

1.1. Yapay Sinir Ağları (YSA)

YSA teriminin ilk kullanımı 1940'lara kadar gitmektedir. Kısaca bu çok katmanlı yapıları, en güçlü öğrenen biyonik makine olarak tasfir edebileceğimiz beyini taklit etmeye çalışan derin ağ yapısı olarak tanımlayabiliriz. Beyin üzerine yapılan araştırmalarda bilişsel kabiliyet ile ilişkilendirilen neo-korteks tabakanın katmanlı ve hiyerarşik olduğu tespit edilmiştir [28]. Bu tabaka algılayıcılardan gelen ses ve görüntü gibi sinyalleri sahip oldukları örüntüler temelinde yeni temsil uzaylarında gözlemlere dönüştürür. Dönüşümlerde nöron adı verilen hesaplama elementleri kullanılır. Bu elementler katmanlar halinde birbirlerine kenetlenerek sahip oldukları doğrusal olmayan fonksiyonlar yardımıyla giriş sinyallerinden gizli örüntüleri ortaya çıkarır. Görüntü tanıma açısından gizli örüntüler aydınlatma, ölçekleme, döndürme ve kaydırma gibi doğal dönüşümlere karşı değişmezlik sağlayan; YSA içerisinde derinlere inildikçe daha ayırt edici bir hal alan veri temsil uzaylarıdır. Basit bir hesaplama elementi Şekil 1.2'de, YSA yapısı Şekil 1.3'te sunulmuştur.



Şekil 1.2. Hesaplama elementi olan bir nöronun yapısı



Şekil 1.3. Çok katmanlı YSA yapısı

YSA yapısı incelendiğinde her zaman bir giriş ve çıkış katmanı olduğu, eldeki problemin karmaşıklığına göre farklı miktarda ara katmanların olduğu görülecektir. Her katman değişik miktarda nöron sayısına sahip olmakla birlikte; ilk katman giriş verisinin boyutu oranında, çıkış katmanı ise arzu edilen çıkışlar kadar nöron ile ilklendirilir. Nöronlar arası bağlantı aktivasyon tipine bağlı olup tümsel veya bölgesel olabilir. Nöron aktivasyonları çoğunlukla 'sigmoid' veya 'arctanjant' gibi doğrusal olmayan ve belirli bir aralıkta çıkış verecek fonksiyonlar arasından seçilir. Katmanlar arası bağlantıları temsil eden ağırlık vektörleri yeni temsil uzayının temel fonksiyonlarıdır. Gizli uzayın çok boyutlu düzlemlerine dik biçimde uzanan bu ağırlık vektörleri bir önceki katman çıkışları ile nokta çarpımı işlemine tabi tutulduğu

için, ara katmanlardaki her bir nöron çıkışı aslında giriş örneğinin temsil edilen düzleme olan mesafesini belirler. Bu mesafenin belirlenmesinde referans alınan merkez nokta, yani düzlem ile ağırlık vektörünün keşisim noktası ise yanlılık (bias) parametresi ile ayarlanır. Ağırlık vektörünü dik kesen düzlemin uzayı ikiye böldüğü varsayılırsa; bir nöron daha yüksek değerle tetiklendikçe bahse konu örnek genel uzayın o ekseninde pozitif tarafa doğru kayıyor anlamına gelmektedir.

Klasik YSA yapısı denetimli öğrenmeye güzel bir örnektir. Özellikle sınıflandırma ve regresyon görevlerinde literatürde sıkça kullanılmaktadır. Giriş örneklerinin arzu edilen çıkış sinyalleri ile birlikte verildiği bir ortamda başlangıçta rastgele ilklendirilen ağırlık vektörleri ileri besleme – geri yayılım yöntemi ve gradyan azaltımı algoritması yardımıyla optimize edilir. Dolayısı ile YSA ara katmanları yeni temsil uzayları yaratırken, aynı zamanda son ara ve çıkış katmanı yardımıyla modelleme yapılabilmektedir. YSA yapısının oluşturulmasında ve eğitiminde kullanıcı tarafından belirlenmesi gereken en temel parametereler ara katman ve her bir katmandaki nöron sayıları ile gradyan azaltımı algoritmasında kullanılan öğrenme katsayısıdır.

Bu yapıdaki temel hipotez katmanlar ilerledikçe derinlerde daha soyut ve hiyerarşik temsil uzaylarının öğrenileceğidir. Hesaplama elementleri olan nöronların doğrusal olmayan fonksiyonlardan üretildiği varsayıldığında teorik olarak çok kaymanlı bir YSA ile her türden karmaşık problemin çözümüne yönelik model üretilebilir [29]. İçerisinde çok karmaşık problemleri barındıran ve tezimizin esasını teşkil eden görüntü tanıma için YSA yapıları bu nedenle oldukça kullanışlıdır. Çok katmanlı bir yapıda imgenin modellenmesi tüme varım yöntemiyle gerçeklenebilir. Örneğin piksel seviyesindeki imge örneklerini modelleyen bir YSA yapısında ilk ara katmandaki nöronlar pikselleri gruplayarak kenarları, ikinci katmandaki nöronlar kenarları gruplayarak konturları [30], sonraki katman nöronları ise nesne şekillerinin bir parçasını hatta nesnelerin kendisini [31] temsil edebilir. Sonuç olarak, her bir ara katman bir önceki katmandaki aktivasyon çıkışlarını ağırlık vektörleri ile yeni bir temsil uzayına taşıdığı ve bir takım doğrusal olmayan dönüşüm fonksiyonları kullandığı için çok katmanlı YSA yapıları güçlü bir veri temsil gücüne sahiptir.

1.2. Çok Katmanlı YSA Yardımıyla Denetimsiz Öznitelik Çıkartımı

YSA'nın yukarıda bahsedilen kabiliyetlerine ve özellikle çok katmanlı yapıların teorik olarak her türlü fonksiyonu yakınsayabilmesine karşın YSA iki önemli kısıtı nedeniyle 2006 yılında Hinton ve diğ. [32] tarafından önerilen 'fırsatçı katman tabanlı denetimsiz ön eğitim' yöntemine kadar kendisine geniş uygulama alanları yaratamamıştır.

İlkin, geleneksel ileri besleme – geri yayılım yöntemini kullanan çok katmanlı ağlar denetimli öğrenme yapısına sahip olup amaç fonksiyonları giriş verisinin arzu edilen çıkış (etiket) sinyallerine de ihtiyaç duymaktadır. Ancak, bilindiği üzere etiketlenmiş sinyaller genellikle seyrektir ve denetimin kalitesi etiketlemeyi sağlayan insan/makine deneklerin deneyimleri ile doğru orantılıdır. Kısaca denetim iş gücü bakımından çoğunlukla maliyetli ve uygulamaya özgü bir süreçtir. Derin ağ yapısının karmaşık yapısı göz önüne alındığında yetersiz miktarda etiketlenmiş veri ile oluşturulan bir eğitim aşırı uydurma ile sonuçlanacaktır.

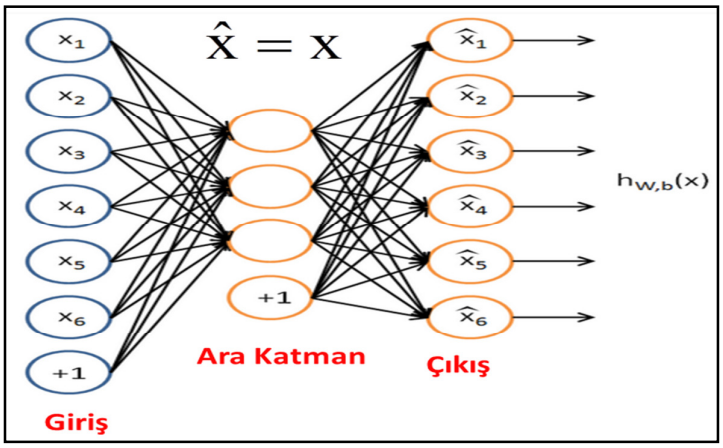
İkinci husus, çok katmanlı YSA yapısı tek seferde eğitildiğinde yüksek karmaşıklıkta ve konveks olmayan bir optimizasyon problemi ile karşı karşıya kalınmasıdır. Bu çözümü genelde kötü lokal optima noktasına taşımaktadır. Çünkü hesaplama elementlerinde sigmoid gibi doğrusal olmayan fonksiyonlar ardışık katmanlarda üst üste kullanılmakta, ağırlık vektörleri rastgele ilklendirilmekte ve düzenlileştirme esnasında normları azaltılmaya çalışılmaktadır. Sonuç olarak yenilemeli bir algoritmada hata vektörünün ters yönünde çıkışa yakın katmandaki ağırlık vektör değişimleri arzu edilen seviyede olsa da; ilk katmanlardaki ağırlık değişimleri oldukça azalmaktadır.

Denetim ve kötü lokal optima problemlerinin üstesinden gelmek için fırsatçı katman tabanlı denetimsiz ön eğitim metodu geliştirilmiştir [32, 33]. Bu metodolojide ana fikir etiketlenmemiş veriyi kullanarak denetimsiz bir yapı içerisinde çok katmanlı YSA mimarisini katman katman öğrenmektir. Başlangıçta ağ yapısının yalnızca ilk katmanı alınarak eğitim gerçekleştirilir. İlk katman eğitiminin tamamlanmasını müteakip ikinci katman sisteme dahil edilir ve ilk katman ağırlık vektörleri sabit tutularak güncel yapı yeniden eğitilir. Bu işleme kalan ara katmanlar birer birer dahil edilmek suretiyle devam edilir. Sonunda, çok katmanlı ağ mimarisi güncellenmiş

ağırlık vektörleriyle birlikte, bütün haliyle, klasik YSA yaklaşımındaki gibi son kez eğitilir. Denetimsiz yapıda çok miktarda etiketsiz verinin sisteme dahil edilmesi neticesinde, bu metod yardımıyla daha gürbüz gizli örüntüler ortaya çıkarılabilmektedir. YSA'da denetimsiz öğrenmenin temel yapı taşları olarak literatüre sunulan Auto-Encoders (AEs) ve Restricted Boltzmann Machines (RBMs) [34] yapılarına aşağıda kısaca değinilmiştir.

1.3. Kendine Kodlayıcılar (Auto Encoders - AEs)

Bu algoritmada elimizde yalnızca bir takım etiketsiz eğitim verisinin bulunduğu varsayılmıştır [35]. Eğitim veri seti $X=\{x^{(1)},x^{(2)},x^{(3)},...,x^{(i)}\}, x^{(i)} \in \Re^n$; verildiği takdirde ileri besleme – geri yayılım yöntemi ile öğrenen kendine kodlayıcı ağ yapısı arzu edilen çıkış değerlerini giriş sinyallerine ($y^{(i)} = x^{(i)}$) eşitlemektedir. Bu durumda AEs için amaç fonksiyonu $h_{W,b}(x^{(i)}) \approx x^{(i)}$ olacağı için diğer bir deyişle özdeşlik fonksiyonuna yakınsanmaya çalışılmaktadır. AEs temel yapısı sadece bir ara katman ihtiva etmekte olup Şekil 1.4'te gösterilmiştir.



Şekil 1.4. Kendine kodlayıcı ağ yapısı

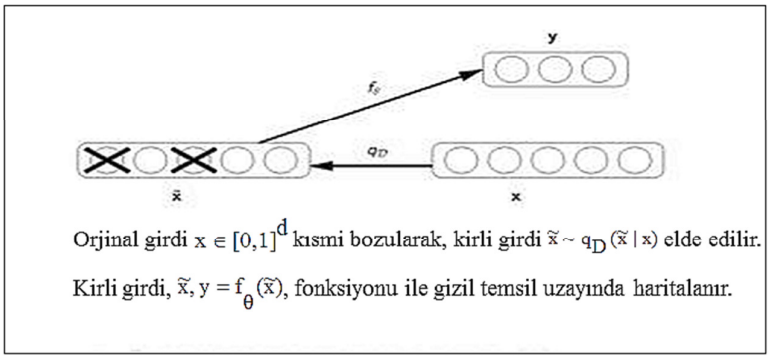
Şekil 1.4'te görüleceği üzere; AEs öncelikle giriş verisini, $x^{(i)} \in \Re^n$, alarak kodlamayı sağlayan ağırlık vektörleri yardımıyla gizli uzayda, $h^{(i)} \in \Re^k$, temsil eder. Burada k orjinal giriş uzay boyutu olan n'ye göre genellikle daha küçük (k << n) olup hesaplama elementleri gerekirci bir fonksiyon olan sigmoid veya hiperbolik tanjant olabilmektedir. Gizli uzayda temsil edilen bu 'kod' çıkış tarafındaki ağırlık vektörleri aracılığı ile giriş sinyalinin kendisine çözümlenir. Bu iki ağırlık matrisi bazı

durumlarda birbirlerine eşitlenir, $W_{çöz} = W_{kodla}^{T}$, ki buna kısıtlamalı ağırlık yönetimi adı verilir. Her ne kadar zorunlu olmasa da, bu durumda optimize edilmeye çalışılan parametre sayısı azaltılmış olur. AEs ağırlık vektörleri başlangıçta rastgele ve 0 değerine yakın ufak değerlerde ilklendirilir. Klasik YSA yaklaşımı ile paralel bir şekilde gradyen azaltımı algoritması yardımıyla yinelemeli olarak model eğitilir. Model eğitiminin denetimsiz bir şekilde tamamlanmasını müteakip giriş verisi kodlama ağırlık vektörleri yardımıyla yeni uzaya taşınarak müteakip işlemler burada gerçeklenir.

Her ne kadar denetimsiz öğrenmeyi sağlayan AEs mimarisi etkileyici gözükse de; bu yapının ciddi bir problemi mevcuttur. Eğer ara kartmandaki nöron sayısı giriş verisinin boyutuna yakın veya daha fazla olursa ($k \geq n$) algoritma sadece özdeşlik fonksiyonu oluşturmaktadır. Bu durumda ayırt edici nitelik uzayının ara katman çıkışlarında oluşması beklenemez. Diğer taraftan giriş verisinin orjinal uzaydaki boyutunun düşük olduğu karmaşık bir problemin çözümü için verinin çoğu zaman daha büyük boyutlu uzaylara taşınması arzu edilmektedir. Dolayısı ile bahse konu problemin çözümüne yönelik olarak literatürde üç temel yaklaşım benimsenmiştir.

1.3.1. Gürültü giderici kendine kodlayıcılar (Denoising AEs)

Problemin çözümüne yönelik bir çözüm, modelin eğitimi aşamasında giriş verisini bir miktar gürültü, v, ile bozmak ve çıkışa yine orjinal giriş sinyalini yerleştirmektir. Burada parametre v bozulma oranının yüzdesel değeridir. Bu yapıya Gürültü Giderici Kendine Kodlayıcı (DAEs) adı verilmiştir [36]. Giriş verisi, $x^{(i)}$, giriş verisinin karakteristik özelliklerine uygun bir dağılıma sahip gürültü dağılımı yardımıyla kirletilerek değiştirilir ve gürültülü veri, ỹ, ağ yapısına sokulur. Görüntü verisi için siyah beyaz imgelerde binom gürültüsü (bazı giriş piksel değerlerinin sıfıra çekilmesi), renkli imgelerde ilintisiz Gauss gürültüsü eklenmesi popüler yöntemlerdir. Böylece DAEs amaç fonksiyonu özdeşlik fonksiyonunu öğrenmek yerine sinyal gürültüsünü arındıracak şekilde verideki gizli örüntüleri tespit etmek şekline sokulmuştur. DAEs yapısının yığıt veya konvolüsyon şeklinde karakter tanıma ve nesne sınıflandırmaya yönelik çalışmalarda [37] başarılı performanslar sergilediği görülmüştür. DAEs yapısı Şekil 1.5'te gösterilmiştir.



Şekil 1.5. Gürültü giderici kendine kodlayıcı ağ yapısı [36]

1.3.2. Sönümlendirmeli kendine kodlayıcılar (Droput AEs)

AEs algoritmasında özdeşlik fonksiyonu oluşumunu engelleyici diğer bir yaklaşım giriş sinyalini gürültü ile bozmak yerine ara katmandaki nöron çıkışlarını her yenilemede farklı olacak şekilde bastırmak olmuştur [38]. Sönümlendirmeli kendine kodlayıcı adı verilen bu metodda belirli miktarda, v, nöron çıkışı maskelenerek sıfıra çekilir. Her yenilemede hangi nöron çıkışlarının pasif hale getirileceği ise belirli bir olasılıksal dağılıma (normal ve düzgün dağılım gibi) uyularak tespit edilir. Belirli bir dağılım koşullanmasında ara katmandaki nöron sayısı kadar rastgele üretilen değerler içerisinden olasılığı % 30'un (v parametresi) altındaki nöronların çıkışları pasif yapılır. Buradaki yaklaşım tıpkı DAEs gibi özdeşlik fonksiyonundan uzak ve gürültüye karşı gürbüz yeni temsil uzayları elde etmek olmuştur.

1.3.3. Seyretmeli kendine kodlayıcılar (Sparcity AEs)

Son olarak, AEs yapısında ara katman nöron çıkışlarını kısıtlamak üzere amaç fonksiyonuna seyreltme düzeltmesi ilave edilmiştir [35, 39, 40]. Modelin çıkışta ürettiği hataya eklenen seyreltme cezası nöronların ortalama ne kadar aktif durumda kaldıkları ile ilişkilendirilmiştir. Nöron aktivasyonları ikili (0 veya 1) değer üretecek şekilde tasarlandığı takdirde, seyrekme cezası tıpkı sönümlendirmeli AEs gibi her yinelemede sadece belirli miktarda nöronun aktif kalmasına indirgenmiştir. Nöron çıkışlarının reel sayılar üretmesi durumunda ise ara katmandaki her bir nöronun ortalama aktivasyon değeri belirli bir sabit sayı ile kısıtlanmıştır. Bu kapsamda ara katman nöronlarının mümkün olduğunca pasif halde kalmaları arzu edilerek bahse konu sabit sayı aktivasyon değerine göre minimum değere yakın (sigmoid fonksiyonu için 0,05 ve tanjant hiperbolik için -0,95 gibi) tutulmuştur. Nöronların

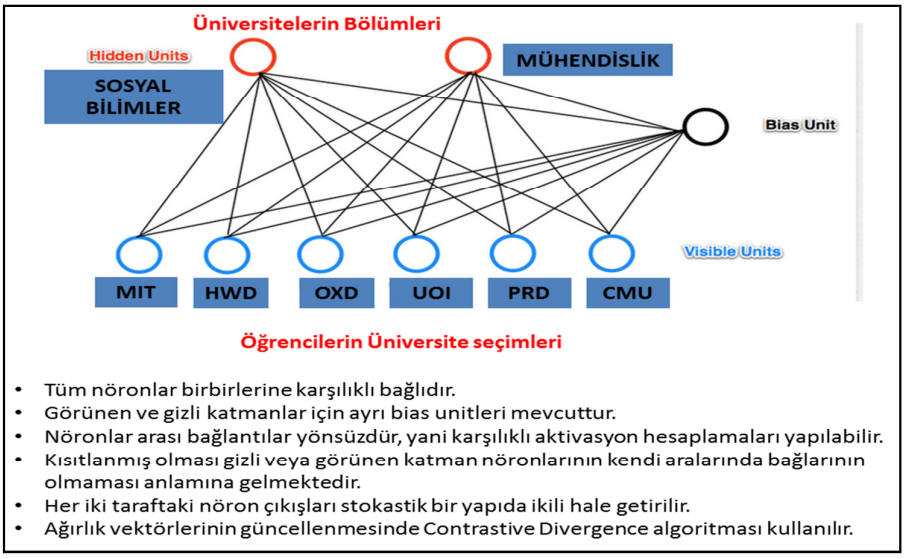
ortalama çıkış değerleri ile arzu edilen asgari çıkış kısıtı arasındaki hatayı amaç fonksiyonuna hata olarak yansıtan türevi alınabilir bu ilave fonksiyon Kullback-Leibler (KL) ayrışması olarak adlandırılmıştır.

1.4. Kısıtlanmış Boltzmann Makineleri (Restricted Boltzmann Machines)

RBMs stokastik bir yapay sinir ağı yapısında tasarlanmıştır [32, 36]. YSA mimarisi açısından iki katmana sahiptir. Her iki katmandaki aktivasyon çıkışları stokastik bir süreç içerisinde ikili (0 veya 1) duruma çevrilmektedir. İlk katmanda gözlenen nöron elementleri bulunmaktadır. Buradaki nöron çıkışları ilk giriş sinyallerini ağ yapısına sokmaktan sorumludur. Diğer katman ise ara katman olarak adlandırılır ve buradaki nöron çıkışları gözlenen değerler ile ağırlık vektörlerinin skaler çarpımları sonucu elde edilmektedir. Ayrıca, her iki katmanda da ayrı yanlılık (bias) elementleri bulunmaktadır. Her zaman aktif konumda tutulan bu elementler farklı giriş sinyalleri ile ara katman çıktı faktörleri arasındaki düzenlemeyi yapmaktadır.

Ağırlık vektörleri klasik YSA mimarisinden farklı olarak yönsüzdür. Dolayısı ile simetrik ağırlık vektörleri aracılığıyla ileri beslemede ara katmandaki ikili çıkışlar bulunabileceği gibi geri beslemede giriş katmanındaki çıkışlar da hesaplanabilmektedir. İki katmandan oluşan bu basit ancak etkin yapıyı optimize etmek üzere, ağırlık vektörleri 'Contrastive Divergence' adı verilen bir algoritma ile güncellenmiştir. Hesaplamalarda kolaylık olması açısından ara veya gözlenen katmandaki nöronların birbirleri ile bağlantıları koparılmıştır ve 'kısıtlanmış' ifadesi buradan gelmektedir. RBMs yapısı Şekil 1.6'da gösterilmiştir.

AEs mimarisi ile RBMs yapısı karşılaştırıldığında; denetimsiz öznitelik öğrenimi maksadıyla RBMs stokastik koşullu olasılık fonksiyonları kullanmaktadır. İki katmandan oluşan yapı içerisinde yönsüz ağırlık vektörlerini simetrik biçimde kullanmak suretiyle; her iki taraftaki ardışık çıkışlar arasındaki farklar yardımıyla ağırlık vektörlerini rastgelelik temelinde güncellemektedir. Sonuç itibariyle, AEs yapısı gerekirci fonksiyonları ile aşırı tamlanmış durumda özdeşlik modeli üretirken RBMs rastgele güncelleme sayesinde gürültüye veya aykırı örneklere karşı daha gürbüzdür.



Şekil 1.6. Kısıtlanmış boltzmann makineleri ağ yapısı

1.5. AEs ve RBMs Ağ Yapılarının Çok Katmanlı Mimarilere Dönüşümü

Yukarıda bahsedildiği üzere; derin ağ mimarileri katmanlar arası geçişlerde ilerlendikçe daha soyut ve nitelikli temsil uzaylarının eldesinde kullanılan, birden fazla ara katmana sahip ve denetimsiz öznitelik öğrenmeyi teşvik eden yapılardır. Bunun nedeni her ara katmanın kendisinden önceki katman aktivasyonlarını kullanması ve belirli bir amaç fonksiyonu doğrultusunda hata oranını azaltmayı hedeflemesidir.

Bu kapsamda AEs ve RBMs temel yapı taşlarını ele alarak yığıt şeklinde bunların üst üste eklenmesiyle çok katmanlı denetimsiz öğrenme mimarileri ortaya çıkarılmıştır. Literatürde yer alan Yığıt Kendine Kodlayıcılar (Stacked Auto Encoders -SAEs) AEs ağ yapılarının, Derin Varsayım Ağları (Deep Belief Networks -DBNs) ise RBMs ağ yapılarının ardışık olarak kullanılmasıyla elde edilmiş popüler yaklaşımlardır [34, 36]. Bölüm 1.2'de ayrıntıları verilen 'fırsatçı katman tabanlı denetimsiz ön eğitim' metodu her iki çok katmanlı mimarinin eğitilmesinde referans teşkil etmiştir. Örneğin, SAEs yapısını eğitmek üzere öncelikle bir AE ağı giriş verisi yardımıyla eğitilmiştir. Müteakiben, giriş verisi eğitilen kodlama vektörleri yardımıyla birinci AE'nin ara katmanına taşınmış ve böylece ikinci AE ağına giriş sinyalleri elde edilmiştir. Bütün AE'lerin müstakil eğitimlerinin tamamlanmasının ardından, tüm AE yapıları birleştirilmek ve kodlayıcı vektörleri daha önceden eğitilenler ile

ilklendirilmek suretiyle bütün yapı son kez ince ayarlanmıştır. Anlatılan bu süreç DBNs mimarisi için RBMs ağ yapıları üzerinde de geçerlidir.

AEs ve RBMs yapılarından ayrı olarak, literatürde sıkça bahsedilen ve Bölüm 3'teki deneysel sonuç karşılaştırmalarında kullanılan Konvolüsyon Tabanlı Sinir Ağları (Convolutional Neural Networks –CNNs)'dan da kısaca bahsetmek gerekir. CNNs yapısı konvolüsyon ve alt örnekleme/havuzlama isimli iki farklı katmanın birbiri sıra yer değiştirerek yığıt haline getirilmesiyle oluşturulmuştur. Konvolüsyon katmanı klasik YSA'daki ara katmandan farklı olarak görüntü verisi üzerinde konvolüsyon işlemini gerçekleyen nöronlara sahiptir. Dolayısı ile ara katman nöron aktivasyonları skaler vektör çapım sonuçları değil, iki boyutlu konvolüsyon işlem sonuçlarıdır. Bu çıkışlara 'Nitelik Haritaları' denilmektedir ve alt örnekleme/havuzlama katmanının görevi nitelik haritalarının ölçeklerini indirgemektir. Ağırlık vektörlerinin güncellemesi aynı şekilde ileri besleme – geri yayılım metodu ile yapılmakla birlikte hata değer dağılımında konvolüsyon işleminin türevinden faydalanılmıştır.

Denetimsiz nitelik çıkarımı ile sınıflandırma veya yakınsama odaklı görevleri çok katmanlı ağ yapılarında birleştirmek üzere; çıkışa bir ayırtaç veya uydurma (softmax, Destek Vektör Makineleri gibi) katmanı eklenmektedir. Bu şekilde tasarlanmış derin ağ mimarileri nesne tanıma [42], yüz tespiti [31] ve faaliyet tespiti [43] gibi bilgisayarla görme problemlerinde uygulanmış ve başarılı sonuçlar elde edilmiştir.

1.6. Yapay Sinir Ağı Mimarisinde Öbekleme

Veri öbekleme aynı öbekteki örneklerin diğer öbeklerdeki örnekler ile belirli bir metrik sistem üzerinde kıyaslandığında; aynı öbekteki örneklerin birbirlerine daha benzer olması gerektiği prensibine dayanan ve etiketsiz veri örneklerinin gruplandırılması problemine çözüm arayan metodlarının ortak adıdır [47,48]. Temel hedef girdi verisinin atanacağı öbeklere aidiyet olasılığını maksimize eden, bu atanmaları kodlayan parametreleri temsil eden ve verinin altında yatan gizli örüntüleri bulmaktır. Öbekleme metodları bir üst hedef olan sınıflandırma ve regresyon probleminin çözümü için bir ön işlem olarak kullanılabilir. Özellikle öznitelik çıkartma, vektör kuantizasyon, imge bölütleme, boyut indirgeme ve fonksiyon yakınsama gibi pek çok maksatlar için düşünülebilir.

Her ne kadar literatürdeki öbekleme teknikleri çok çeşitli görünse de; göreve yönelik uygulamalar göz önüne alındığında öbekleme algoritmalarını 'hiyerarşik' ve 'rekabetçi' olarak iki ana kategoride toplamak mümkündür. Hiyerarşik öbekleme algoritmalarında maksat tüme varım veya tümden gelen yöntemleri kullanarak hiyerarşik öbekleme paternleri oluşturmaktır. Diğer taraftan, rekabetçi öbekleme metodlarında ise tek seferde birbirinden bağımsız öbekler elde edilmeye çalışılmaktadır.

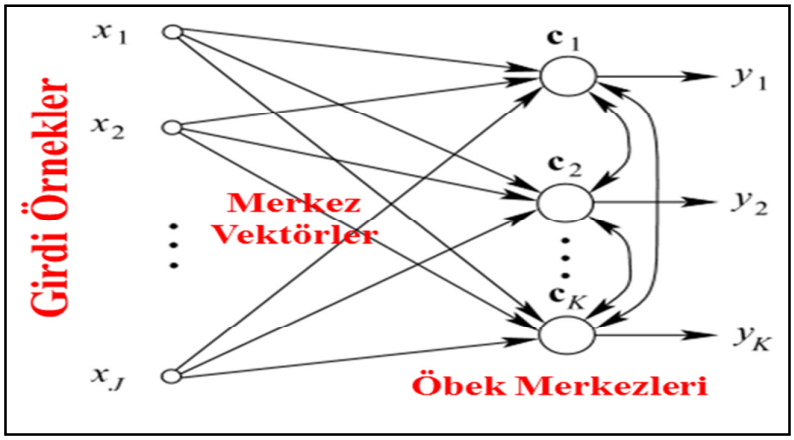
Ancak her iki türde veriyi modeleyen parametrelerin optimizasyonu önemlidir. Burada bahsedilen parametreler her bir öbeği modelleyen ve öbeğe özgü gizli paternleri temsil eden değişkenler olmakla birlikte; seçilen bir metrik sistemde hesaplanan veriye has temel istatistiki bilgilerdir. Ele alınan istatistik türüne has olmak üzere, örnekler arasındaki benzerlik (ya da benzemeklik) hesaplamaları bir nitelik uzayı üzerinde yapılır; ya da benzer şekilde ardıl olasılık dağılımlarındaki yoğunluk kestirimi için örneklerin öbeklere ait olma olasılıkları tespit edilebilir. Son aşamada ise amaç fonksiyonu olarak adlandırılan bir ilişkilendirme kriteri kullanılarak öbekleme problemi için optimum çözümler üretilir.

Graf ve ağaç temelindeki yapıların yanında, yapay sinir ağları eldeki göreve has uygulamalarda kullanılabilecek herhangi bir öğrenme algoritmasını esnek yapısı itibariyle kolayca simule edebilme özelliğine sahiptir. Bu nedenle hiyerarşik veya rekabetçi öbekleme metodlarının koşulmasında haklı bir üne sahiptir. Yapay sinir ağları, verinin orjinal nitelik uzayı yerine örtük bir şekilde daha derinlerdeki katmanlarda yeni nitelik uzaylarında temsil edilebilmesine imkan sağlamaktadır. Bu mimari her ne kadar başlangıçta denetimli öğrenme için tasarlanmış olsa da; literatürdeki son çalışmalar bu ağların denetimsiz öğrenmedeki kabiliyetini ortaya çıkartmıştır. AEs ve RBMs yapay sinir ağları içerisinde denetimsiz örnek uzaylarının öğrenilmesinde temel yapı taşları olarak başvurulabilir. Bu yapılar, türleri ve uygulamaları [29, 31, 36, 38, 39]'da detayları ile incelenmiştir. Bu metodları etkin kılan şey, örneklerin temsil edileceği yeni uzayların denetimsiz olarak üretilmesi esnasında büyük miktarlarda etiketsiz veriyi kullanmaları nedeniyle aşırı uydurmadan, istenildiği ölçüde karmaşık mimari oluşturulmasına imkan vermeleri nedeniyle de düşük uydurmadan kaçınabilmeleridir.

Literatürdeki öbekleme odaklı klasik yapay sinir ağı mimarileri incelendiğinde, esas itibariyle giriş ve çıkış olmak üzere iki katmanlı oldukları ve çıkış katmanındaki nöronların birbirlerini kısıtlatıcı bir takım dahili bağlantılara sahip oldukları görülmektedir. Tipik olarak çıkış katmanındaki her bir nöronun bir öbeği temsil ettiği düşünülerek giriş-çıkış katmanları arasındaki ağırlık parametreleri doğrudan prototip/sentroid vektörleri olarak kullanılmaktadır. Çıkış katmanındaki kısıtlamalar basit bir ifadeyle öbek güncelleme prosedürünün karakteristik özelliklerini belirlemektedir. Her bir giriş örneği için ya belirlenmiş bir alıcı alandaki (receptive field) öbekleri temsil eden bütün sentroidler veya en yakın öbeği temsil eden tek bir prototip vektörü belirli bir ağırlıklandırma şeması dahilinde güncellenmektedir. Genel olarak öbekleme problemleri analitik çözümlere sahip olmadığından örnekleri tahmin edilen öbeklere yenilemeli bir yapıda atamayı formülize eden maliyet fonksiyonundaki toplam hata gradyen tabanlı algoritmalar ile minimize edilmektedir. Tezin aşağıdaki bölümlerinde literatürde sıkça başvurulan ve Bölüm 3'teki deneysel sonuç karşılaştırmalarında kullanılan öbekleme ağ mimarileri incelenmiştir.

1.6.1. Basit rekabetçi öğrenme ağı (BRÖA)

Öbeklemeye yönelik ilk sinir ağı yapıları Basit Rekabetçi Öğrenme Ağı (BRÖA) (Simple Competitive Learning Network – SCL) mimarisiyle başlamıştır [48, 49]. BRÖA en yakın komşuluk paradigmasını uygulayan iki katmanlı ve tam bağlantılı bir ağdır. BRÖA yapısı Şekil 1.7'de sunulmuştur.



Şekil 1.7. Basit rekabetçi öğrenme ağı yapısı

Bu mimaride giriş verisi $X=\{x^{(i)} | \ i = 1, 2, 3, \ldots, N\}$, $x^{(i)} \in R^d$, çıkış katmanında öbekleri temsil eden sonlu sayıdaki prototip vektörüne $C=\{c^{(j)} | \ j = 1, 2, 3, \ldots, K\}$, $K << N$ ve $c^{(j)} \in R^d$, yakınsanmaktadır. BRÖA'nın amaç fonksiyonu Ortalama Karesel Hata (Mean Squared Error -MSE)'yı minimize etmektir;

$$E = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}\frac{1}{2}\mu_{ij}\left\|x^{(i)} - c^{(j)}\right\|^2; \ \mu_{ij} \in \{0,1\} \tag{1.1}$$

Burada μ_{ji} Kronecker Delta parametresi olarak adlandırılır. Her bir örnek veri , $x^{(i)}$, için L_2 norm metrik sistemde en yakın (kazanan) prototip vektörü, $c^{(j)}$, üzerinde 1 diğer merkez vektörleri için ise 0 değeri alır. Bahse konu en yakın prototip kısıtlamasına Kazanan Hepsini Alır (Winner-Takes-All WTA) işlemi adı verilir. Sonuçta, Kronecker Delta parametresi her bir örnek için diğer prototip vektörlerinin güncellenmesini engeller; sadece kazanan prototip vektörü , $c^{(w)}$, güncellenir. Girdi ve çıktı katmanları arasındaki ağırlık vektörleri esas itibariyle ptototip vektörlerini, C, temsil etmektedir. Amaç fonksiyonu gradyen azaltma metodu ile optimize edilmektedir. Böylece, BRÖA prototip vektörlerini yinelemeli olarak şu şekilde güncellemektedir;

$$\forall x ; c^{(w)} = c^{\left(\text{argmin}_{j=1,2,3,\ldots,K}\left\|x^{(i)} - c^{(j)}\right\|\right)} \tag{1.2}$$

$$\frac{\partial E}{\partial c^{(w)}} = -\left[x^{(i)} - c^{(w)}(t)\right] \tag{1.3}$$

$$c^{(w)}(t+1) = c^{(w)}(t) + \eta(t)\left[x^{(i)} - c^{(w)}(t)\right] \tag{1.4}$$

$$c^{(j)}(t+1) = c^{(j)}(t) \ ; \ j \neq w \tag{1.5}$$

Burada $\eta(t)$ pozitif küçük öğrenme katsayısını ifade eder ve genellikle zamanla azalan bir fonksiyon üzerinden gösterilir. Literatürde bir kısım çalışmalarda öğrenme katsayısı tüm öbekler için eşit ve $\eta(t) = \eta_0(1 - t/T)$, $\eta_0 \in (0,1]$; t yineleme sırası, T yineleme sınırı ile ifade edilmektedir. Eğer $\eta(t)$ her bir öbek için farklı olacak şekilde $\eta_j = 1/N_j$; N_j her bir yineleme esnasında j nci öbeğe atanmış örnek sayısı, seçilirse öbek içi varyansın minimize edildiği [50]'de gösterilmiştir.

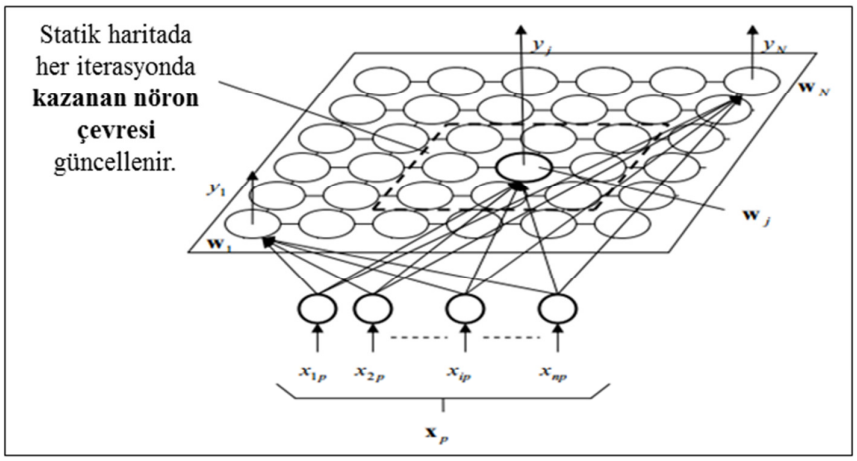
Tıpkı diğer yapay sinir ağı mimarilerinde olduğu gibi BRÖA'da da tüm veri seti elde mevcut ise bir anda (batch) işleme sokulabilir veya öğrenme çevrim içi modda veri

geldikçe birbirini izleyen artışlar halinde gerçekleşebilir. Öbek sayısı kullanıcı tarafından önceden belirlenir ve prototip (sentroid) vektörleri ağdaki ağırlık vektörleri olarak rastgele ilklendirilir. BRÖA algoritmasının dezavantajlarından biri öbekleme performansının rastgele ilklendirilen prototip vektörlerine yüksek derecede bağlı olmasıdır. Çünkü her bir veri örneği yalnızca kazanan prototip vektörünün güncellenmesine neden olmaktadır ve verinin giriş katmanından giriş sırası da çevrim içi öğrenmede öbeklerinin güncellenmesine ayrıca etki eden unsurdur.

1.6.2. Self organizing map (SOM) ağı

Diğer popüler öbekleme ağı Kohonen ya da daha çok tanınan ismiyle SOM (Self-Organizing Map) mimarisidir. SOM [51-53] esas itibariyle BRÖA'ya benzer bir ağ yapısına ve amaç fonksiyonuna sahiptir. Ancak BRÖA'dan farklı olarak SOM uyarma veya komşuluk adı verilen bir fonksiyon aracılığı ile her bir örneğin kazanan (en yakın) prototipi merkezde olmak üzere belirli bir alandaki tüm prototip vektörlerini aynı anda güncellemektedir. Basit SOM yapısı Şekil 1.8'de gösterilmiştir.

İlk olarak belirgin bir grid konfigürasyonuna sahip (bir – iki –çok boyutlu) sanal bir yüzey oluşturulur ve prototip vektörler bu grid yüzeydeki kesişim noktalarına konuşlandırılır. Bu prototip vektörleri, C, başlangıçta rastgele oluşturulur ve girdi-çıktı katmanlarındaki ağırlık vektörlerini temsil etmektedir [54]. Oluşturulan sabit grid konfigürasyonu, çıkış katmanında mevcut neronlar arasında önceden belirlenmiş bir metrik sistem üzerindeki, Euclidean gibi, bağlantıları yansıtmaktadır.



Şekil 1.8. SOM yapısı [54]

Bir örnek girdi için en yakın prototip vektörü, $c^{(w)}$, seçildikten sonra bu vektörün grid yapıdaki belirli bir komşuluğunda bulunan diğer prototipler şu şekilde güncellenir;

$$c^{(j)}(t+1) = c^{(j)}(t) + \eta(t) h_{(jw)}(t)\left[x^{(i)} - c^{(j)}(t)\right] \tag{1.6}$$

$$h_{jw}(t) = h_0 e^{\frac{-\left\|c^{(j)} - c^{(w)}\right\|^2}{\sigma^2(t)}} \; ; \forall x \tag{1.7}$$

$$\sigma(t) = \sigma_0 e \frac{-t}{T} \; ; \; \sigma_0 \in (0,1] \tag{1.8}$$

Burada $h_{jw}(t)$ Gaussian komşuluk fonksiyonudur ve kazanan prototip vektörüne, $c^{(w)}$, olan mesafe arttıkça azalan bir ağırlık katsayısını ifade etmektedir. $\sigma(t)$ Gaussian fonksiyonundaki ölçekleme faktörünü temsil eder. Uyarılan komşuluk alanının çapına etki eden bir unsurdur. $\eta(t)$ zamanla azalan öğrenme oranıdır ve h_0 (0,1] aralığındaki başlangıç komşuluk sabitidir. Dikkat edilmesi gereken unsur SOM mimarisindeki grid yüzeydeki komşuluğu temsil eden $h_{jw}(t)$ fonksiyonu yerine Kronecker Delta parametresi kullanıldığı takdirde, SOM'un BRÖA'ya indirgenebileceğidir [48]. Bu uyarlamada, grid yüzeydeki topoloji korunmak üzere, bir örnek üzerinden kazanan prototipin yakın çevresindeki diğer vektörler uyarıcı bağlantılar aracılığıyla ve belirli bir ağırlıklandırma üzerinden hep birlikte güncellenirken daha uzakta bulunan kalan prototip vektörler ise kısıtlayıcı bağlantılar varmış gibi, ancak komşuluk fonksiyondaki ağırlıklar yüzünden güncellenmemektedir. Dikkat edilirse SOM'daki öbek güncelleme prosedürü, BRÖA'nın aksine, belirli bir amaç fonksiyonundan elde edilen hata oranının minimize edilmesi ile yürütülmemektedir. Bu nedenle SOM'da yakınsama (convergence) garanti edilemez ve verinin girdi sırası da ayrıca önemlidir.

1.6.3. Learning vector quantization (LVQ) ağı

SOM ve BRÖA'da daha önceden bahsedilen dezavantajlar ışığında, Vektör Kuantizasyon Ağı (Learning Vector Quantization – LVQ) daha önceden denetimsiz şekilde elde edilen öbekleri denetimli bir şekilde ince ayarlamak maksadıyla [53]'te önerilmiştir. LVQ temelinde denetimli öğrenme odaklıdır. Zira daha önceden elde edilmiş prototip vektörleri ve veri-öbek atanmalarını verinin ardıl yoğunluk fonksiyonu olarak ele alır ve BRÖA'nın amaç fonksiyonu Denklem (1.1)'i şu şekilde minimize etmeye çalışmaktadır;

$$c^{(w)}(t+1) = c^{(w)}(t) - (-1)^{y_w^{(i)}} \eta(t)\left[x^{(i)} - c^{(w)}(t)\right];\ y_w^{(i)} \in \{0,1\} \quad (1.9)$$

$$c^{(j)}(t+1) = c^{(j)}(t)\ ;\ j \neq w \quad (1.10)$$

Burada artık girdi verisinin $P = \{ x^{(i)}, y^{(i)}\}$; $x^{(i)} \in R^d$ ve $y^{(i)} \in \{0,1\}^K$, patern çiftlerinden oluştuğu görülmektedir. İkili vektöler, $y^{(i)}$, daha önceden elde edilmiş ve öbekleme prosedürü boyunca sabit tutulacak olan başlangıç veri-öbek atanmalarını temsil eder. Yine $\eta(t)$ $(0 < \eta(t) < 1)$ zamanla azalan öğrenme oranını ifade eder. Denklem (1.9) ve (1.10)'da yer alan formüllere yakından bakılacak olursa, LVQ'nun bir örnek, $x^{(i)}$, için yalnızca en yakın prototip vektörün güncellenmesi açısından BRÖA'ya benzediği görülecektir. Ancak Denklem (1.9)'da yer alan negatiflik operatörü örnek verinin ağ içerisinde yanlış öbeğe atanması durumunda kazanan prototip vektörün örnekten açasıya zıt yönde güncellenmesine sebep olmaktadır. Bunun amacı Bayesian karar yüzeyinde prototip vektörlerinin etrafındaki yoğunluğu azaltmak olmakla birlikte maalesef aynı zamanda referans vektörlerdeki sapmaları (divergence) tetiklemektedir [53].

Karşılaştırma olması açısından, LVQ ve bu tezde önerilen DVMA mimarilerinin her ikisi de daha önceden oluşturulmuş başlangıç öbeklerini yeniden ayarlamak üzere denetimli öbekleme maksadıyla kullanılmaktadır. Diğer taraftan, DVMA kendi içerisinde oluşturulan yeni gizli uzayda veri-öbek atamalarını yinelemeli olarak güncellemeye devam eder ve maksimum marjin amaç fonksiyonunu kullanır. Bunlar ikisini birbirinden ayıran DVMA'nın temel özelliğidir. Bu nedenle, Bölüm 3'teki deneylerde DVMA'nın öbekleme performansını dikkate değer bir şekilde geliştirdiği gözlenmiştir.

1.6.4. Neural gas (NG) ağı

Şimdiye kadar BRÖA ve LVQ'nun Kazanan-Hepsini-Alır yapısında sırasıyla denetimsiz ve denetimli şekilde sadece kazanan prototip vektörleri güncellediği görülmüştür. Diğer taraftan SOM, her bir örnek girdi için en yakın olarak seçilen prototip vektörüne göre belirli bir komşuluk fonksiyonu ile ağırlıklandırmak suretiyle kazanan vektörün belirli bir yakınlıktaki tüm prototip vektörleri aynı anda güncellemektedir. Her ne kadar bu komşuluk ilişkisindeki öbeklerin güncellenmesi prosedürü ilham verici olsa da; SOM öbekleri temsil eden prototipler arasında daha önceden belirlenmiş grid yapıda statik komşuluk ilişkileri tanımlamaktadır.

Oluşturulan grid konfigürasyon yapısı ilerleyen öbekleme prosedürü içerisinde vektörler arası mevkisel ve göreceli topolojiyi koruyacak yönde ve orjinal nitelik uzayında sabit tutulmaktadır.

Bu bakış açısını değiştirmek üzere [48, 55]'te Neural Gas (NG) ağ mimarisi önerilmiştir. Soft-max kuralı standard k-means algoritmasını genişletecek biçimde ve SOM'daki komşuluğa dayalı öbeklemeye benzer yapıda NG'de uygulanmaktadır. Soft-max adaptasyonu bir örnek için sadece kazanan prototip vektörünü, $c^{(w)}$, değil; ilaveten kazananın yakınındaki diğer prototip vektörlerinin de yeniden yapılandırımasını sağlar. Bu konudaki diğer bazı çalışmalar güncellemeyi sağlayan Soft-max kuralı yerine Maximum Entropy (ME) algoritmasını [56] da işletmiştir. Ancak önemli olan unsur, NG'nin dinamik bir komşuluk ilişkisine dayalı sıralama düşüncesini geliştirmek suretiyle her bir örnek girdi için prototip vektörlerini yinelemeler ilerledikçe farklı ağırlıklarda güncellenmesidir.

Başlangıçta, her bir veri girdisi, $x^{(i)}$, için tüm prototip vektörlerine, C, komşuluk sıralaması Euclidean mesafesi kullanılmak suretiyle tespit edilir. Müteakiben her bir prototip vektörüne, $c^{(j)}$, örnek girdiye olan mesafesine bağlı olarak bir tamsayı sıra indisi atanır. r_{ji} = 0,1,2, …, K-1, K kullanıcı tarafından önceden tanımlanmış öbek sayısı ve 0 kazanan prototip vektörünü, $c^{(w)}$, ifade etmektedir. Bu kapsamda gradyen azaltım tabanlı öbek güncelleme kuralı şu şekildedir;

$$c^{(j)}(t+1) = c^{(j)}(t) + \eta(t)h(r_{ji}, t)\left[x^{(i)} - c^{(j)}(t)\right] \tag{1.11}$$

$$h(r_{ji}, t) = e^{\frac{-r_{ji}}{\rho^{(t)}}} \tag{1.12}$$

$$\eta(t) = \eta_0\left(\frac{\eta_f}{\eta_0}\right)^{\frac{t}{T}}, \ \rho(t) = \rho_0\left(\frac{\rho_f}{\rho_0}\right)^{\frac{t}{T}} \tag{1.13}$$

Burada h(r,t) sıralama çarpanını temsil eder. Bu fonksiyon, prototiplerin örneklere olan mesafelerini esas alan sıra indisi, r_{ji}, ve karakteristik azalan sabiti, $\rho(t)$, parametre olarak kullanır. Dolayısı ile belirgin bir örnek, $x^{(i)}$, için her bir prototip vektörü sıralama indisi ölçütünde güncellenmektedir. Daha önceki algoritmalarda karşılaştığımız $\eta(t)$ gibi, azalan sabit $\rho(t)$ de zamana dayalı azalan bir parametredir. η_0 ve ρ_0 başlangıç, η_f ve ρ_f ise son sınır azalma parametreleridir.

NG basit şekliyle komşuluk fonksiyonunda SOM gibi prototip vektörler arasındaki mesafeleri doğrudan kullanmak yerine, örneklere göre sıralamayı belirten indisleri kullanmaktadır. Prototip vektörler her bir yinelemede değiştiği için örneklere olan göreceli sıralama indisleri de haliyle değişecektir. Bu şekil bir dinamik sıralama, r_{ji}, ve öbekleme şekil kipleyicisi, ρ, sayesinde NGN'nin diğer algoritmalara göre daha düşük hatalara sahip olduğu ve lokal minimalara karşı daha gürbüz olduğu [48]'de gözlenmiştir. Zira bütün prototip vektörleri her durumda değişen bölgesel alanlarda sıra indisleri ölçütünde yinelemeli olarak her bir örnek üzerinden güncellenmektedir.

1.6.5. Diğer sinir ağı tabanlı öbekleme algoritmaları

Literatürdeki diğer popüler sinir ağı tabanlı öbekleme algoritmaları Adaptive Resonance Theory (ART) [57] ve Fuzzy Clustering (FC) ağlarıdır [58]. ART ağının altında yer alan temel fikir, her iki yönde ve birbirinden farklı olacak şekilde ağırlık vektörleri yardımıyla birbirine tam bağlı girdi, F_1, ve çıktı, F_2, katmanlarından oluşan bir iki katmanlı tekrarlı (recurrent) yapısı oluşturmaktır. İleri yönlü besleme ağırlıkları, $w^{(i)}$, kısa dönem hafızayı temsil etmektedir; ki bir girdi örneğine göre çıkış katmanındaki en yakın prototip vektörü, $c^{(w)}$, bu ağırlıklar sayesinde elde edilir. Geri yönlü ağırlık vektörleri, $c^{(j)}$, ise uzun dönem hafızayı temsil eder ve aslında prototip vektörleri, C, burada muhafaza edilmektedir. Öbek güncelleme işlemi esnasında girdi örüntüleri ile en yakın prototip vektörleri arasındaki yakınlık ve uyumluluk kıstaslarını ölçen iki test icra edilmektedir. Her iki yöndeki ağırlık vektörlerini yinelemeli olarak güncellemek maksadıyla lineer olmayan difransiyel fonksiyonlar kullanılmaktadır. Her ne kadar ART mimarisi çabuk değişen girdi sekanslarına göre gürbüz öbekler elde edebilse de; kullanılan girdi sinyali ve ağırlık vektörleri ikilidir, girdi örüntülerinin sırasına göre duyarlıdır ve işlem karmaşıklığı difrensiyel denklemler nedeniyle yüksektir.

Diğer taraftan FC ağı SCL mimarisi ile benzer yapıdadır. Ancak FC veri setini bulanık mantık prensibine göre birbiriyle örtüşen öbeklere ayırmaktadır. Dolayısı ile her bir girdi örneği belirli bir üyelik fonksiyonu, $f(x^{(i)},c^{(j)})$, altında belirli bir olasılıkla birden çok öbeğe atanabilmektedir. Her bir örneğin tüm öbeklere üyeliği Fuzzy Set Teorisine göre olasılıklara bağlandığı için üyeliklerin toplamı bire eşittir. Bu kapsamda sadece prototip vektörleri değil, aynı zamanda üyelik değerleri de yinelemeli bir şekilde güncellenmektedir. Bu kapsamda, öncelikle üyelik

derecelerinin soft-max kuralı ile güncellendiği ve sonrasında ise öbek içi örneklerin bulanık ortalamaları alınarak prototip vektörlerinin güncellendiği bir sıralı değişken optimizasyon prosedürü uygulanmaktadır [48]. Üyelik değerleri ve prototip vektörleri başlangıçta rastgele belirlenmektedir. FC mimarisi yüksek hesaplama karmaşıklığı ve hafıza tabanlı olması sebebiyle özellikle büyük ölçekli veri setleri için tercih edilmemektedir.

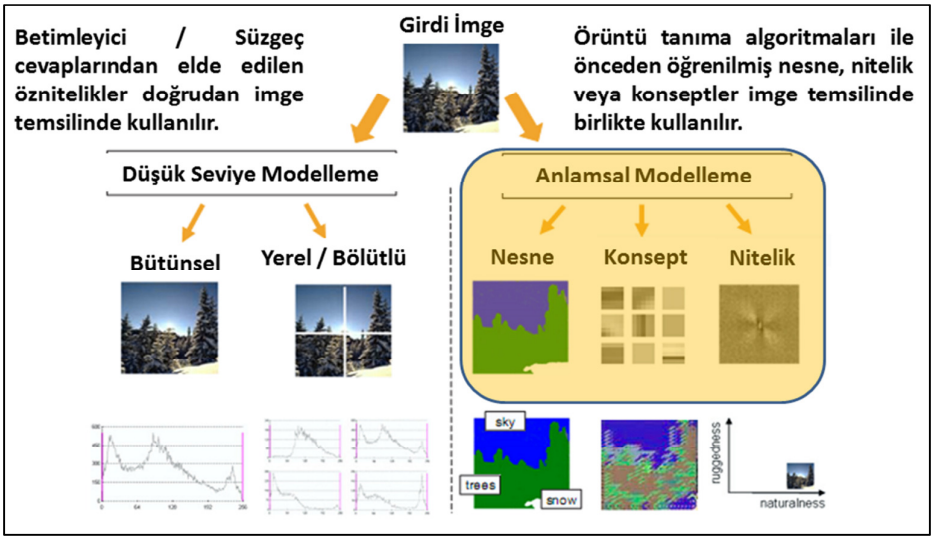
Öbekleme probleminin çözümünde kullanılan DVM tabanlı yaklaşımlar incelendiğinde, literatürde iki farklı yöntem göze çarpmaktadır. Birinci kategoride yer alan yaklaşımlarda orjinal nitelik uzayında öbek olarak kullanılmak üzere küresel bölgeleri tespit etmek için maksimum marjin amaç fonksiyonu bu yönde adapte edilmektedir [59, 60]. Lagrangian metodu destek vektörlerini öbekler arasındaki varsayımsal sınırlar olacak şekilde amaç fonksiyonlarına eklemektedir. Destek vektörlerinin bulunmasını müteakip denetimsiz bir şekilde öbekleme etiketlerini atamak maksadıyla graf tabanlı algoritmalar yardımıyla konpenentler elde edilir. İkinci tür DVM yaklaşımlarda ise ana öbekleme algoritmasını tamamlayıcı olacak şekilde DVM'nin bütünleyici olarak kullanılması prensibi benimsenmiştir. Örneğin [61]'de yazılı dökümanları konu odaklı öbeklere ayırmak için k-ortalamalar metodu en üst seviyede kullanılmıştır. Bu esnada diğer ilişkili verisetleri kullanılarak DVM'leri ile ayırtaç fonksiyonları oluşturulmuş ve k-ortalamalar metodundan elde edilen denetim bilgisi ile bu fonksiyonlar birleştirilerek her bir öbek üzerinde koşullanmalar oluşturulmuştur. Dolayısı ile farklı bir ortamda (örneğin düşük seviyede 'kelimeler') ancak aynı ana veri seti üzerinde DVM fonksiyonları yardımıyla oluşturulan bu koşullanmalar beklenti aşamasında veri-öbek atamaları üzerindeki karar aşamasında kullanılmıştır. Bu tezde önerilen DVMA öbekleme mimarisi yukarıda bahsedilen metodlardan tamemen farklıdır. Zira önerilen DVMA ana algortima olarak yeni gizli uzayda öbekleri bir sinir ağı içerisinde güncellemektedir ve basit maksimum marjin amaç fonksiyonu yardımıyla daha önceden elde edilen başlangıç öbeklerini ince ayarlanmak maksadıyla kullanılmaktadır.

1.7. Görüntü Tanımada Öznitelik Vektörlerinin Kullanımı

Görüntü tanıma problemine yönelik literatür çalışmalarının temelde yerel veya bütünsel düşük seviye öznitelik vektörlerinin çıkartılması, orta veya ileri seviye imge

temsili ve sınıflandırıcı metotlarının uygulanma usulleri bakımından ayrıştığı gözlenmektedir. İmgenin nasıl temsil edileceğinin belirlenmesinin sınıflandırma başarı performansını etkileyen en önemli husus olduğu aşikardır. Zira sınıflandırma işlemi temsil uzayında yapılacağından, uygulamaya dönük en ayırt edici betimleyicilerin kullanılması gereklidir. İmge temsilinin ise temelde yerel veya bütünsel olarak imge üzerinden çıkartılan düşük seviye öznitelik betimleyicileri ile anlamsal modelleme olarak ikiye ayrıldığı söylenebilir. Bu temel husus Şekil 1.9'da sunulmuştur.

Düşük seviye öznitelik vektörleri yardımıyla görüntü verisinin modellenmesi problemi imge ve video analizi alanlarında yıllardır araştırılmaktadır [63]. Öncü çalışmalarda renk, desen, yönelim, moment, şekil ve frekans öznitelikleri imgelerin bir takım konsept kategorilere (iç mekan, dış mekan, şehir, orman gibi) atanmasında önceleri doğrudan kullanılmıştır. Düşük seviye öznitelik vektörleri bu tezin asli öznesi konumunda olmadığından bu bölüm geçilmiştir. Diğer taraftan düşük ve yüksek seviye (sınıf kategorileri ile etiketleme) imge betimleme çalışmalarındaki boşluğu doldurmak maksadıyla bir sonraki aşamada görüntünün anlamsal orta seviyede nitelik/konsept modellenmesi önerilmiştir. Dolayısıyla görüntü tanıma alanındaki en belirleyici faktörün veri prototiplerinin modellenmesi olduğu ve verinin bütününü betimleyecek şekilde içerik tasvirinin nasıl yapılacağı sorusunun günümüz çalışmalarının temelini teşkil ettiği tespit edilmiştir.



Şekil 1.9. Görüntü verisinin temsili [62]

1.8. Niteliklerin Orta Seviye Veri Temsilinde Kullanımı

Düşük seviye öznitelik vektörlerini (GIST, SIFT, renk/yönelim histogramları, spektral analizler, süzgeç cevapları, görsel kelimeler sözlüğü, Fourier/Wavelet/Hough/Laplace dönüşümleri, kuantize edilmiş piksel değerleri gibi) doğrudan sınıf etiketlerine haritalamak yerine orta seviyede görsel nitelikler günümüzde bilgisayarla görme uygulamalarında oldukça populer olan araçlardır [14, 17, 18, 64-68]. Nitelikler görsel uzayda insanlar tarafından ortak bir konsept içerisinde algılanabilen varlıklardır. Sadece makineler tarafından anlaşılan ve hesaplamalarda kullanılan düşük seviye öznitelik vektörleri ile üst seviyede hedef/karar sınıf etiketleri arasındaki orta seviyede temsil uzaylarının temel fonksiyonlarını (basis functions) oluştururlar. Ortak paylaşım platformlarını teşkil ettiğinden, sınıflar arasındaki keskin sınırlar aşılarak veya yumuşatılarak, daha çok örnek ile ayırt edici temsil uzayları öğrenilebilmektedir.

Nitelikler, üst seviyedeki kategorilerin tanınmasında insanlar tarafından gözlemlenebilen ipuçları ve sınıfların ayırt edilmesinde öğrenilen özellikler olarak ele alındığında, bilgisayarla görme problemlerinin pek çoğunda yaygın olarak kullanılmaktadır. Daha önce bahsedildiği üzere; niteliklerin en önemli özellikleri insanlar tarafından anlamlandırılabilir ve sınıflar arasında paylaşılabilir olmalarıdır. Bu nedenle klasik sınıflandırma veya tanıma problemlerinden farklı olarak; nitelik değerleri bilinen ancak öğrenme aşamasında eğitim verisetinde yer almayan yeni sınıfların tespit edilmesinde (Zero Shot Learning – Sıfır Test verisi ile Öğrenme) [69, 70], sınıf etiketleri belirtilmeden görüntünün tanımlanmasında [14, 71] ve hatta nitelik değerleri dahi bilinmeyen yeni türlerin belirlenmesinde (Unfamiliar Object Detection – Bilinmeyen Nesne Tespiti) [19, 72] nitelikler kullanılabilir. Yüz tanıma, nesne sınıflandırma, sahne anlamlandırma, bir video görüntüsündeki aykırı görsellikleri tespit etmek, olay tespiti (Event Detection) gibi daha pek çok uygulama alanı niteliklerin ilgi alanına girmektedir ve detayları ilerleyen bölümlerde verilmiştir.

Niteliklerin düşük seviye öznitelik vektörleri yerine görsel tanımada orta seviye yeni temsil uzayları olarak kullanılmasının bir takım nedenleri bulunmaktadır. Öncelikle düşük seviye öznitelik vektörleri bir takım istatistikler kullanılarak eğitim setindeki örnekleri bireysel olarak işleyen ve el işçiliğine dayanan yöntemlerle

hesaplanmaktadır. El işçiliği ile kasıt örneklerin daha önceden belirlenmiş prototip vektörleri, süzgeçler ve parametreleri sabit bir takım dönüştürücülere verdiği cevapların öznitelik vektörlerinde depolanmasıdır. Oysa nitelikler eğitim sınıflarındaki örneklerin farklı konseptler içerisinde kümelenmesi ve bu kümeler içerisindeki ilişkilerin öğrenilmesi prensibine dayanmaktadır.

Bu noktada belirtilmesi gereken diğer bir husus, nitelik öğrenmenin öbeklemeden veya boyut indirgeme tekniklerinden farklı olduğudur [70]. Öbeklemede amaç örnekleri prototip vektörler merkezinde gruplamak iken; nitelikler tıpkı bir sınıflandırıcı gibi yeni uzayda örnekleri birbirlerinden ayıran bir ayırtaç ya da üzerinde bulunduğu imgedeki gücü ölçen bir sıralama fonksiyonu işlevini görmektedir. Boyut indirgemede ise yeni ve daha düşük boyutlu projeksiyon uzayları elde edilmektedir. Ancak bu uzayın temel vektörlerini yine sınıf etiketleri belirlemektedir. Oysa nitelik uzaylarında örnekler sınıf etiketlerini bir kenara bırakarak belirli konseptler dahilinde birbirinden ayrıştırılmaya veya güçleri dahilinde sıralanmaya çalışılmaktadır. Bu noktada nitelik uzayları basit projeksiyon hesaplamalarından ziyade dönüşüm fonksiyonlarını tanımlamaktadır.

Nitelikler, tek bir kelime yardımıyla dile getirilebileceği gibi ikili veya göreceli karşılaştırmalar ile ve doğrudan bir kelime ile ifade edilmeyecek şekilde oluşturulabilir. ‘A ve B sınıflarında var; ancak C sınıfında yok’ cümlesi ikili isimsiz nitelik (Şekil 1.10), ‘A sınıfında B sınıfına gore daha belirgin, C sınıfı ile benzer şekilde’ göreceli isimsiz niteliklere (Şekil 1.11) örnek cümlelerdir. Burada niteliğe anlaşılabilir tekil bir kelime atanmamış, ancak sınıflar üzerinde nitelikler ayırt edici bir şekilde tarif edilmiştir.



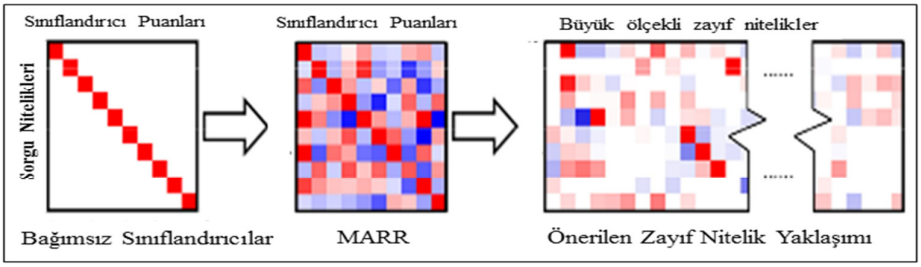
Şekil 1.10. İkili nitelik kullanımı [64]



Şekil 1.11. Göreceli nitelik kullanımı [14]

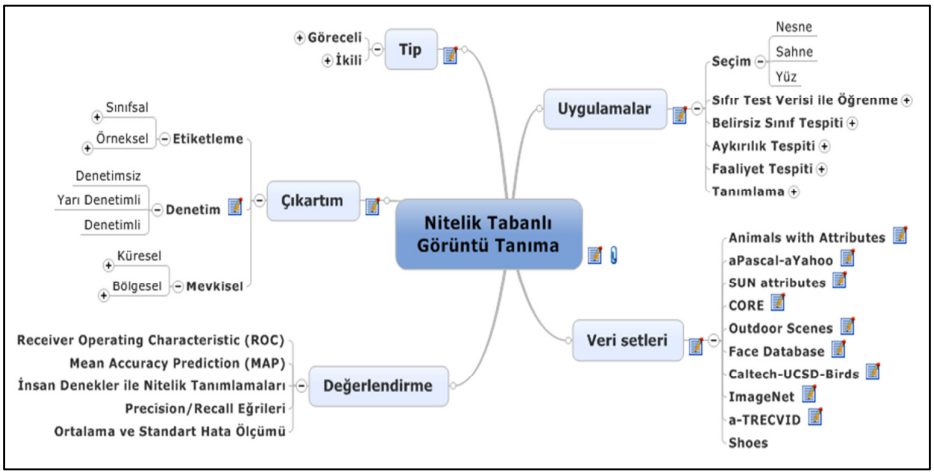
Diğer bir taraftan nitelikler mevki bilgileri, kalıtım veya kapsama açısından hiyerarşik bir yapı içerisinde birbirleri ile ilişkilendirilebilir [73, 74]. Bütünselden ziyade lokal nitelikler özellikle mevkisel sıralama bilgisinin önemli olduğu zamanlarda dikkate alınmıştır [72, 75, 76]. Ancak lokal niteliklerin öğrenimi bölütleme, etiket bilgisini çevreleyici kutularla sınırlama gibi ekstra yoğun etiketleme faaliyetlerine dönüşebilir. Dolayısı ile nitelik öğrenilmesi asıl hedef olarak algılanmamalıdır. Ana hedef tezimizinde öznesi olarak görüntü tanımak ve nitelikleri ayırt edici bir şekilde kullanmaktır.

Niteliklerin öğrenilmesi esasında bir nitelik-sınıf eş bulunma matrisinin oluşturulması anlamına gelmektedir. Şekil 1.12'de bu matrise göre sınıf ile nitelikler arasındaki ilişki tanımlanmaktadır. Bu tanımlama 0-1 şeklinde ikili (bulunup bulunmama) olarak adlandırılabileceği gibi, Bu ilişki matrisi niteliğin sınıf üzerindeki göreceli gücünü açıklayacak şekilde reel değerlerden veya sınıf sıralamalarından oluşabilir. Bu durum imgenin bütünü veya mevkisel olarak belirli kesitleri ele alınarak lokal şekilde tesis edilebilir. İlişki matrisi tam denetim ile (uzman bir heyet tarafından önceden) oluşturulabileceği gibi rastgele [77] veya başlangıç kombinasyonel [72] de oluşturulabilir.



Şekil 1.12. İkili veya reel sayılı nitelik-sınıf matrisi [73]

Literatürdeki nitelik tabanlı bilgisayarla görme problemleri incelendiğinde niteliklerin; çeşitleri ve çıkarılış yöntemleri, üzerinde koşuldukları veri setleri ve deneylerde kullanılan performans ölçütlerine göre özetlenebileceği değerlendirilmektedir. Şekil 1.13'te literatür taraması özetlenmiştir.



Şekil 1.13. Nitelik tabanlı görüntü tanıma üzerine literatür özeti

1.8.1. Nitelik türleri konusunda literatür taraması

Göreceli nitelik kavramı ilk kez Parikh ve Grauman [14] tarafından ortaya atılmıştır. Nitelikler üzerinde sınıfların göreceli sıralamalarını belirlemek suretiyle elde edilen göreceli nitelikler, ikili eşleniklerine göre daha zengin temsil uzayları ve ayırtaç fonksiyonları oluşturmuştur. Newton metodunun temel alındığı SVM benzeri bir algoritma sıralama fonksiyonu oluşturmak maksadıyla kullanılmıştır. Bu sayede sınıfları ayırt edecek marjinleri maksimize etmek yerine sınıfların sıralamarını maksimize edecek bir ağırlık vektörü elde edilmiştir. 'Sınıf A nitelik 1 üzerinde sınıf B'ye göre daha güçlüdür, sınıf C ile benzerdir' şeklinde cümleler ile sınıflar arası

ilişkiler belirlenmiştir. Nitelikler isimlendirilmek ve sınıflar önceden sıralanmak suretiyle göreceli niteliklerin öğrenilmesinde denetimli bir yapı uygulanmıştır.

Sharma ve diğ. [67] çalışmalarında insanları ve yaptıkları eylemleri tanımlamak için niteliklerden yararlanmış; ayrıca poz tahminlemesi ile uğraşmak yerine ayırt edici parça şablonlarını nitelik olarak öğrenmiştir. İlaveten, insanlar ile içerikteki nesneler arasındaki ilişki tiplerini modellemektense insanların görünüşlerini modellemeyi amaçlamış ve bu modele 'Expanded Parts Model' adını koymuştur. Eğitim setindeki imgelerde yer alan insanlar daha önce denetim olarak kutucuklara alınmış, bu kutulardan mevkisel olarak ayırt edici parça şablonları öğrenilmiştir. Nitelik/eylem ikili etiketleri ile etiketlenmiş bu eğitim setinden elde edilen mevkisel parça şablonları, test aşamasında verilen bir imgenin puanlanmasında kümülatif olarak kullanılmıştır.

Tamara ve diğ. [75] internet arama motorlarından nitelikleri aratmak suretiyle görev alanına uygun nitelikleri elde edilen gürültülü imgeler üzerinde eğitmeyi amaçlamıştır. Dolayısı ile eğitim esnasında nitelik etiketleme yapılmamış ve denetimden kaçınılmıştır. Bu nedenle elde edilen imgeler üzerinde niteliklerin görsel olarak tesbiti, doğruluğu veya bütünlüğü garanti edilememiştir. Görsel nitelikler, lokal veya bütünsel, renk, desen veya şekil temelinde ayrı ayrı öğrenilmiştir. Eğitim esnasında, bir imgenin üzerinde niteliklerin görsel olarak hangi oranda tespit edileceğini belirleyecek bir puanlama fonksiyonu öğrenilmiştir.

Biswas ve Parikh nesne/yüz tanımanın göreceli nitelikler üzerinden interaktif bir yapıda oluşturulmasını önermiştir [78]. Daha önceki yarı denetimli çalışmalarını [69] seçilen negative örnekleri ağırlıklandırma, makine tarafından insana verilecek sorgu imgesinin seçimi ve niteliklerin güncellenmesi yönünde geliştirmiştir.

1.8.2. Nitelik çıkartımı konusunda literatür taraması

Ferrari ve Zisserman [64] zayıf bir denetim mekanizması içerisinde nesne tanımaya yönelik olarak nesnenin ikili nitelikleri içerip içermediği ve içeriyorsa hangi bölgelerde bahse konu niteliğe sahip olduğunu belirleyecek olasılıksal türetici bir model sunmuştur. Nitelikler renk ve desenden oluşan oldukça basit görsel varlıklar olarak ele alınmıştır. Bir bölütleme metodu yardımıyla görünüş açısından sıklıkla

karşımıza çıkan ve imgeler arasında ortak paylaşım sağlayacak bölgeler ön plan olarak seçilmiştir. Şekil, renk ve desenden oluşan ön planda tek bölgeden elde edilen nitelikler basit, yan yana olmak kaydıyla, iki bölge sıkça birlikte bulunuyorsa (çizgili, noktalı, damalı gibi) bunlara da kompleks nitelikler denilerek birlikte modellenmiştir.

Lampert ve diğ. [17, 18] özellikle nesne tanımada Zero-Shot öğrenme metodunu ortaya atmıştır. Denetimli bir şekilde daha önceden verilmiş sınıf tabanlı nitelik-sınıf ikili matrisine göre bir takım sınıfları eğitimde kullanmamışlar; test aşamasında kullanılmayan sınıfları öğrenilen nitelikler üzerinden tahminlemeye çalışmışlardır. Dolayısı ile eğitim setinde görünen sınıflar ile kullanılmayan sınıflar arasında nitelikler üzerinden bir bağlantı kurulmuştur. Bu bağlamda 'doğrudan' ve 'dolaylı' nitelik tahmini isimli iki metod önerilmiş ve koşullu olasılıklar (Bayesian) kullanılmıştır.

Farhadi vd. [65] niteliklerin birbirleriyle mevkisel ilişkilerini göz önüne alan parça temelli nesne tanıma problemi ile uğraşmıştır. Tanımlamanın ötesinde bir sahnedeki nesnelerin mevki bilgilerini nitelikleri aracılığı ile tespit etmek hedeflenmiştir. Her bir sınıfı ayrı ayrı öğrenmektense, öncelikle 'hayvanlar' ve 'vasıtalar' şeklinde iki geniş küme oluşturulmuş; nitelikler ise temel, üst kademe kategori ve parça detektörü şeklinde hiyerarlik olarak modellenmiştir. Modellemede yönlü grafik metodu olasılıksal bir çerçevede kullanılmıştır. Başka bir çalışmalarında [66] nesne tanımlama ve özellikle nitelik öğrenme ile ilgili metodlar üzerinde yoğunlaşmışlardır. Denetimli nitelikler için eğitim seti oluşturulurken internetten elde edilen nesne imgeleri kutucuklara alınmış ve parça tabanlı, şekil, desen, materyal olarak bir takım nitelikler denetimli olarak öğrenilmiştir. İlaveten; denetimsiz nitelik öğrenme kapsamında nitelikler sınıfların ikili karşılaştırmaları ile elde edilmiştir. Son olarak; ikili niteliklerin öğrenilmesinde bütün farklı sınıfların düşük seviye uzayda bir arada kullanılmasının çok değişinti yaratacağı ve gerçek anlamda niteliğin öğrenilemeyeceği vurgulanmıştır. Bunun yerine, öncelikle her sınıf kendi içerisinde niteliği ortaya çıkaran öznitelik indisleri için kullanılmış; son aşamada ise sınıflar seçilen ortak öznitelik indisleri üzerinde birleştirilerek nitelikler öğrenilmiştir.

Akata ve diğ. [68] nitelik tabanlı sınıflandırma problemini çözmek amacıyla orta seviyedeki nitelik uzaylarına hem düşük seviye öznitelik vektörleri hem de üst seviye sınıf etiketleri tarafından aynı anda yaklaşacak kompozit bir altyapı tesis etmiştir. Yani esasında imge gömme (bir imgeyi tanımlama) yerine nitelikler üzerinden kategori gömme (bir kategoriyi konsept olarak tanımlama) üzerine yoğunlaşılmıştır. Bu sayede sınıflandırıcılar ile birlikte nitelik uzayları da eş zamanlı oluşturulmuştur. Algoritmada nitelikler yan bilgi olarak sınıf etiketlerinin modellenmesi kapsamında öğrenilirken her bir imgenin sınıf etiketleri ile uyumluluğunu belirleyen olasılık temelli parametric bir yapı da ayrıca öğrenilmiştir.

Russakovsky ve Fei-Fei [79] ImageNet adı verilen çok büyük ölçekli bir veri seti üzerinde görünüm tabanlı nitelikleri öğrenecek bir çerçeve metod önermiştir. Sınıflar arası bağlantılar ve hiyerarşik ilişkiler veri seti bünyesinde ayrıca verildiği için ortak alan kabul edilen niteliklerin öğrenilebilirliği artırılmıştır. İkili SVM metodu hiyerarşik imgeleri eğitimde kullanarak niteliklerin öğrenilmi için koşulmuştur. Nesne tanıma, zero-shot sınıflandırma ve nesne sınıfları arasındaki ilişkilerin nitelikler üzerinden ortaya çıkarılması uygulamaları makalede tecrübe edilmiştir.

Rastegari ve diğ. [71] her bir niteliği ayrı öğrenmek yerine daha ayırt edici ve daha kolay ögrenilecek olmak kaydıyla eldeki nitelikleri birlikte öğrenmeye odaklanmıştır. Makalelerinde ikili uzayda bir öğrenilebilirlik fonksiyonu oluşturulmuş, bu fonksiyon içerisinde sınıf içi bütünlük ve sınıflar arası marjin değişintisi parameter olarak alınmıştır. Neticede öğrenilebilirlik fonksiyonu çıktısı pozitif çıkan nitelikler ikili biçimde birleştirilmiş ve birleşik olarak yeni bir nitelik öğrenilmiştir.

Ma ve diğ. [70] sınıf temelli göreceli nitelikleri denetimsiz bir şekilde öğrenecek yeni bir yaklaşım önermiştir. Bu çalışma [14]'dekinin aksine sınıflar arası göreceli ilişkilerin önceden elle oluşturulmasından farklı bir yaklaşımdır. Önerilerin başlangıcı sınıfları birbirlerine göre ikili olarak 'daha çok' göreceliliğine göre kıyaslamaktır. Yani nitelik adedi sınıf sayısının ikili kombinasyonudur. Başlangıçta sadece iki sınıfı daha çok şeklinde kısıtlayıp göreceli niteliği öğrendikten sonra; kalan sınıfları nitelik eğitimine fırsatçı bir yaklaşımla ve entropi temelinde teker teker eklenmiştir.

Daha önceden elle belirlenmiş ikili/göreceli nitelik etiket ve sınıf sıralamaları yerine Karayel ve Arıca [77] denetimsiz nitelik öğreniminde temelde [14, 66, 76] ile benzer bir yaklaşım sergilemiştir. Fakat bu çalışmada, nitelikler tamamen rastgele öğrenilmiştir. İkili nitelik öğreniminde, her bir nitelik için sınıflara rastgele 0 veya 1 verilirken, göreceli nitelik öğreniminde sınıflar rastgele sıralanmıştır. Tek şart ise daha önceden oluşturulan sıralamadan farklı bir sıralama elde edilmesidir. Böylece sınıf sayısının ikili nitelik olarak ikinin üsteli, görecelilikte ise faktöriyeli oranında daha geniş bir temsil uzayı elde edilmiştir.

Wang ve Mori [80] bir eğitim seti vasıtasıyla nitelikler arasındaki bağlantıyı ortaya koyacak bir yönsüz graf modeli önermiş ve bunu sınıflandırmada kullanmıştır. Diğer çalışmalardan ayrılan belirgin özelliği, nesne sınıflandırma ve nitelik tahminini tek bir çatı altında toplamaktır. Zira nesneler ile nitelikleri arasında yakın ilişki konsepleri vardır. Eğitim setindeki nesne sınıf ve nitelik etiketleri denetim kapsamında verilmiştir. Netice verilen nitelik garfına göre bir imgenin en iyi yakınsamasını sağlayan nitelik konfigürasyonu elde edilmeye çalışılmıştır. Nesneler ve nitelikleri bütünsel olarak ve sınıf temelinde nitelik-nitelik ve nitelik-sınıf ilişkilerini ortaya koyacak şekilde graf parametreleri optimize edilmiştir.

Yu ve diğ. [73] problem çözümüne değişik bir boyut getirmiş; az sayıdaki denetimli nitelik ile sınıfların arasına orta seviyede çok boyutlu (6000 kadar) bir temsil uzayı daha katmış ve buna zayıf nitelik uzayı adı vermişlerdir. Zayıf nitelikler, daha önceden bazı uygulamalardan elde edilen ve elimizdeki sınıflar ile doğrudan bağlantısı da olmayan diğer sınıflandırıcılar, orta seviye otomatik olarak sisteme ilave iş gücü getirmeyecek şekilde elde edilen neticelerdir. Sonuçta nitelik – zayıf nitelik arasındaki ilişki matrisi yarı denetimli graf algoritması ile tespit edilmiş, her bir ana niteliğin hangi zayıf nitelikleri tetiklediği tespit edilmiştir. Böylece az sayıda ana nitelikle temsil yerine bunlara karşılık gelen çok sayıda zayıf nitelik ile temsilin performansı artırdığı görülmüştür.

Chen ve diğ. [81] regresyon model öğrenimi için kümülatif nitelik temsilini önermiştir [19]. Yaş ve kalabalık kestirimi gibi regresyon problemlerinin çözümünde skaler değerlerin birbirleri ile yakın ilişki içerisinde olduğu varsayılmıştır. Öyle ki, temsil uzayında birbirine yakın değerlerin orjinal uzaydaki koordinat sisteminde de

komşu olmaları beklenmiştir. Böylece nitelik uzayında da yakın yaş gruplarının birbirine yakınsaması sağlanmıştır. İki katmanlı regresyon algoritmasında ilk katmanda düşük seviye öznitelik vektörleri nitelik uzayına çekilirken, son katmanlı nitelik uzayı tek bir yaş değer kestirimi yapacak şekilde yapay sinir ağı kullanılmıştır.

İmgelerin sınıf etiketlerinin verildiği bir ortamda, sınıflar arasındaki nitelikleri otomatik olarak öğrenen bir algoritma Yu ve diğ. [82] tarafından formalize edilmiştir. Bu bağlamda daha önceki çalışmalarda yer alan rastgele sınıf bölünmelerinden farklı olarak kategori ayrılabilirlik ve nitelik öğrenilebilirlik fonksiyonları türetilmiştir. Sınıf seviyesindeki nitelikler isimlendirilmemekle birlikte bilinen sınıflarla ilişkilendirilmiştir. Sınıflar arası mesafeler bir metrik üzerinde bulunarak öncelikle sınıf benzerlik matrisi, S, oluşturulmuştur. Müteakiben, S matrisinin Laplace dönüşümü yapılarak en yüksek Eigen değere sahip vektör ilk nitelik olarak alınmıştır. Bu işlem fırsatçı bir şekilde her yinelemede bulunan nitelik vektörleri ile S matrisi yeniden dönüştürülerek bir sonraki nitelik ağırlık vektörü hesaplanmıştır. Nitelik sayısı ise kullanıcı tarafından belirlenmiştir.

1.8.3. Nitelik uygulamaları konusunda literatür taraması

Wah ve Belongie [19] çalışmalarında belirsiz sınıfların (unfamiliar classes) görsel uzayda tespitine yönelik önerilerde bulunmuştur. Bu tanımsız sınıflar ne eğitimde gözlenen ne de nitelik dağılımları belli olan sınıflardır. Esasında her sınıfın nitelik uzayında kendine özgü bir kimliğinin olduğu vurgusu yapılmış, test aşamasında öğrenilmiş olan nitelikler ışığında daha önceden belirlenen sınıf-nitelik dağılımlarından farklı bir iz bulunduğu takdirde, bu imge belirsiz sınıfa atanmıştır.

Saleh ve diğ. [20] çalışmalarında aykırılık tespit sistemi önermiş; öncelikle sınıf seviyesinde tipiklik fonksiyonunu sınıfın sahip olması gereken görsel nitelikler üzerinden modellemiştir. Daha sonra ise her bir nesne sınıfı için ortaya konan birden çok prototip örnek üzerinden önemli değişintileri hesaplayacak metrik bir yapı tesis edilmiştir. Buradaki temel düşünce sınıflar ve nitelikleri üzerinde tipik örneklerin benzer dağılım davranışları göstereceğidir. Bu korelasyonlar sayesinde anomaliler için sebep sonuç ilişkileri çıkartılmıştır. Sınıf-nitelik arasındaki tipiklik fonksiyonu belirlenirken Bayesian koşullu olasılık denklemlerinden faydalanılmıştır.

Wang ve diğ. [74] dış ortam sahne sınıflandırmasında nitelikleri kullanmış; isim ve sıfat çiftlerinden elde edilen nitelik sözcüklerinin imge örneklerindeki gerçek mevkileri yerine bütünsel etiket bilgileri verilerek zayıf bir denetimle öğrenme algoritması önerilmiştir. Sahnelerin parçalardan oluştuğu ve bu parçaların da hiyerarşik bir ilişkiler yumağına ait oldukları prensibinden hareketle, imge bölgeleri nitelikler ile ilişkilendirilmeye çalışılmıştır. 'Hierarchical Space Tiling' metodu yardımıyla, isim nitelikleri sahne parçalarına ve isim niteliklerinin görselliği ise sıfat nitelikleri ile hiyeraşik olarak ilişkilendirilmiştir. Bu sayede sadece sahnede nitelik tespiti değil, aynı zamanda mevkisel tespit de yapılabilmiştir.

Kumar ve diğ. [76] nitelik öğrenme konusunda çalışmalarında yeni bir tartışma başlatmış; yüz tanımada bazı belirgin insan yüzleri dikkate alınarak diğer eğitim yüz imgelerini bahse konu referans yüzlere göre tanımlamaya çalışmışlardır. Bu niteliklere 'similes' adı verilmiştir. Bu tanımdan hareketle, yüzdeki belirgin özellikler referans yüzlere göre tanımlanmıştır. Mesela; alnı Barack Obama gibi açık, gözleri Jennifer Lopez gibi. Benzerlik yanında saç rengi, yaş ve ırk gibi ikili nitelikler de destekleyici olarak kullanılmıştır. Bir yüz tanıma algoritması ile elde edilen yüz bölgesi, elle bölütlenmiş ortak alt alanlara ayrılmış ve yüz alt bölgeleri yine elle etiketlenmiştir. Her bir alt bölge taşıdığı nitelik özelliğine göre değişik düşük seviye öznitelik vektörleri ile ifade edilmiştir.

Li ve diğ. [83] video verisinden insan aktivitelerini nitelikler yoluyla tespit etmek üzerine çalışmalar yürütmüştür. Kısaca her video görüntüsü davranışları temsil edecek şekilde birbirlerinin üstüne bindirmeli segmentlere bölünmüş ve her segmentte çerçeveler bulunmuştur. Bir video için tek bir nitelik puanlaması yapmak yerine, segmentleri elde eden kayan pencere yardımıyla video zaman vektöründe bahse konu niteliğin izi haline getirilmiştir.

Videodaki nitelik kullanımına yönelik bir başka örnek de Ma ve diğ. [84] tarafından verilebilir. Daha önceden elde edilmiş etiketli ve çok basit nitelikler ihtiva eden referans video parçacıkları diğer videolardaki karmaşık faaliyetleri tespit etmek maksadıyla kullanılmıştır. Karmaşık faaliyeti tanımlamaya yarayan basit ve temel nitelikler insanlar, nesneler, sahneler veya olaylar olabilir. Bahse konu çalışmada Hilbert uzayında bu nitelik sınıflandırıcılar ile olay detektörleri arasında bir ilişki

matrisi oluşturulmuştur. Multi-level Collaborative Regression modeli ile karmaşık faaliyete ilişki matrisi üzerinden bir skor verilmeye çalışılmıştır.

Chen ve diğ. [85] yüz tanımlayıcılarının nitelikler üzerinden eğitiminde çok farklı bir yöntem belirlemiştir. Nitelik olarak yüz sahibine ait isimler ele alınmış; yani aynı isme sahip insanların yüzünün ortak bir sınıf oluşturabileceği değerlendirilmiştir. Nitelikler isimlerin ikili olarak eşleştirilmesi ile oluşturulan ayırtaç fonksiyonlarından üretilmiş ve yaş sınıflandırıcıları öğrenilmiştir. Eğitim setindeki örnek yüz imgeleri ise denetimsiz bir şekilde internet arama motorlarından elde edilmiştir. Deneyler sonucunda ise insanlara isimlerin tamamen rastgele verilmediği; bunun yerine cinsiyet, sosyo-ekonomik durum, etnik köken gibi dayanakların bulunduğu tespit edilmiştir.

İnsan tasfirinde görsel niteliklerin kullanımı Sadovnik ve diğ. [86] tarafından yapılmıştır. Özellikle kalabalık içerisinde bir insanı diğerlerinden ayırt edecek şekilde nitelikler üzerinden tarif etmenin önemi üzerinde durulmuştur. Model amaç fonksiyonunda eldeki pek çok nitelikten en az miktarda niteliği kullanarak dinleyiciye tarif edilen insanı en doğru şekilde göstermek hedeflenmiştir. Bunu yapabilmek için, esasında bünyesindeki iki alt problem aynı anda çözebilecek bir olasılıksal metod önerilmiştir. Nitelik sınıflandırıcılardaki belirsizliği hesaplayan nitelik seçimi ile niteliklerin birlikte kullanılmasıyla hedef insanın doğru tanımlaması olasılıksal olarak dönüşümlü hesaplanmıştır. İlaveten, kalabalıktaki bir insanı tanımlarken sadece o insandaki nitelikler değil, daha ayırt edici olmak kaydıyla mevkisel olarak hemen yanındaki diğer insanların özellikleri de ayrıca kullanılmıştır.

Choi ve diğ. [87] nesne sınıflandırma üzerinde durmuş, ImageNet'ten çekilen bir takım örnekler yarı denetimli olarak nitelikler üzerinden öğrenilmeye çalışılmıştır. Bu kapsamda oluşturulan etiketsiz bir havuz içerisinden yeni bir örneğin nitelikler üzerinden sınıf tahminlemesi ile ayırt edici nitelik temsil uzaylarının güncellenmesi ardışık olarak yürütülmüştür. Çalışmada sınıfa ve örneğe özgü şeklinde iki ayrı nitelik türü kullanılmıştır. Sınıfa özgü nitelikler ile eğitimde kullanılan sınıfların sahip olduğu en yaygın niteliklere sahip örnekler seçilirken; bir sınıfa genel özellikleri itibariyle atanması zor aykırı örnekler seçilmiştir.

2. KAPSAM VE ÖNERİLEN YÖNTEMLER

Bu bölümde, esas itibariyle tez çalışmasında önerilen denetimsiz düşük/orta seviye veri temsil uzaylarının oluşturulması ve görüntü tanımada nitelik uzayları ile eş zamanlı ve artırımlı görüntü tanıma yöntemleri üzerinde durulmuştur.

Düşük seviye veri temsil uzaylarının eldesine zemin hazırlama maksadıyla, tezin ilk kısmında iki aşamalı ve çok katmanlı yapar sinir ağı mimarisine dayanan bir öbekleme algoritması önerilmiştir. Algoritmanın temelinde, Kendine Kodlayıcı (Auto-Encoder) tipine benzer bir ağ yapısı ile denetimsiz olarak yeni bir öznitelik (temsil) uzayının öğrenilebilmesi ve Destek Vektörleri Makinesi Ağı (DVMA) yapısı aracılığı ile başlangıç öbekleme koşullarına ince ayarlama yapılması prensiplerinin birleştirilmesi yatmaktadır. İlk aşamadaki sinir ağı Prototip Kodlama Ağı (PKA) olarak adlandırılmaktadır. Bu ağda verinin yeniden yapılandırma hatası denetimsiz olarak minimize edilmiştir. İkinci seviye ağ olan DVMA'da ise ilk ağdan elde edilen öbekleme çıktıları denetim olarak kullanılmak suretiyle, öbeklerin sınırları arasındaki ayırdım maksimize edilmeye çalışılmıştır. Her iki seviye yapay sinir ağında oluşturulan yeni gizli uzayda Self Organizing Map (SOM -Kendinden Düzenlemeli Haritalama) metoduna benzer bir yapı kullanılarak topolojiyi koruyan bir şekilde öbekler ardışık olarak güncellenmiştir. Öbek kaydırma işlemi, çok katmanlı bir yapay sinir ağı mimarisinde her iki ağın kodlama katmanlarının dönüşümlü olarak birbirlerini izleyecek şekilde kullanılmaları ve ortaya çıkan gizli uzayların öbek güncellemede işlenmeleri suretiyle gerçeklenmiştir. Önerilen öbekleme algoritması deneysel ortamda birbirlerinden farklı örüntülerde, değişik miktarda örnek ve öbeğin bulunduğu iki ayrı popüler veritabanından elde edilen veri setleri üzerinde tecrübe edilmiştir. Önerdiğimiz çok katmanlı sinir ağı mimarisindeki öbekleme metodu literatürde mevcut benzer öbekleme algoritmaları ile analitik ve görsel farklı sonuçlar üzerinde karşılaştırıldığında; daha iyi sonuçlar elde edildiği gözlenmiştir.

Bir önceki kısımda geliştirilen yapay sinir ağlarından elde edilen öbeklerin Görsel Kelimeler Sözlüğü (GKS) vektörüne dönüştürüldüğü ikinci kısımda; görüntü tanıma için yeni bir yöntem önerilmiştir. Temelinde, bir görüntü verisini hiyerarşik GKS

temsil uzaylarına taşıyan ve her katmanda K-ortalamalar (K-means) algoritmasını taklit eden çok katmanlı bir ağ mimarisi tasarlanmıştır. Bu kapsamda her katmandaki girdi uzaylarında GKS vektörlerini oluşturan görsel kelimeler PKA öbekleme yapısında elde edilmiştir. Expectation-Maximization (EM – Beklenti-Maksimize etme) algoritmasını kullanmak suretiyle, katmanlar arası geçişte veriyi daha anlamlı temsil uzaylarına taşıyacak konvolüsyonel işlemler yapılmıştır. Beklenti aşamasında mevcut görsel kelimeler sabit tutularak ağ parametreleri optimize edilirken; güncellenmiş ağ yapısı içerisinde öbekler arası mesafeyi maksimize edecek şekilde görsel kelimeler kaydırılmıştır. İlaveten, derin ağın ikinci katmanında imgelerin çeyrek dilimlerinde farklı ağlar içerisinde yeni görsel kelimeler öğrenilerek GKS temsil uzayına mevki bilgisi de dahil edilmiştir. Önerilen yöntem görüntü tanımada populer CIFAR-10 isimli geniş ölçekli veri seti üzerinde literatürdeki benzer yaklaşımlar ile kıyaslanmıştır.

Orta seviyede görüntü verisi temsilinin incelendiği üçüncü kısımda, temel olarak nitelik uzaylarının denetimsiz oluşturulması incelenmiştir. Bilindiği üzere; imge sınıflandırmada niteliklerin öğrenilmesi aşamasında denetimin kalitesi insan deneklerin tecrübesi ile doğru orantılı ve emek yoğun bir iştir. Ayrıca, görüntü verisindeki yüksek sınıf içi ve düşük sınıflar arası değişinti miktarları sınıflandırmada niteliklerin kategorik (ikili) kullanımını çoğunlukla yetersiz kılmaktadır. Bahsedilen kısıtlamaları ortadan kaldırarak ölçeklenebilir, düşük maliyetli ve kabul edilebilir başarımda imge sınıflandırması amacıyla; bu kısımda imge sınıflarından göreceli nitelikleri denetimsiz olarak çıkaran yeni bir yöntem önerilmiştir. Önerilen yaklaşım, literatürde ön plana çıkan diğer nitelik tabanlı çalışmalarla orta ölçekli veri kümeleri üzerinde ve aynı deney şartlarında test edilerek daha başarılı sonuçlar elde edilmiştir.

Denetimsiz orta seviye veri temsil uzayı çıkartımı ile eş zamanlı görüntü modellemenin birleştirildiği son kısımda; üst seviye imge sınıfları arasındaki karşılaştırmalı ilişkileri açığa vuran göreceli niteliklerin makine-makine etkileşimlerinde geri besleme olarak kullanıldığı ve artırımlı öğrenme tabanlı bir görsel tanıma algoritması önerilmiştir. Tıpkı gerçek dünyadaki öğrenme prosedüründe olduğu gibi, makinelerden biri elindeki başlangıçta sınırlı sayıdaki sınıf etiketli veri ile ‘Öğrenci’ (S) rolünü üstlenerek zaman içerisinde sorgulama

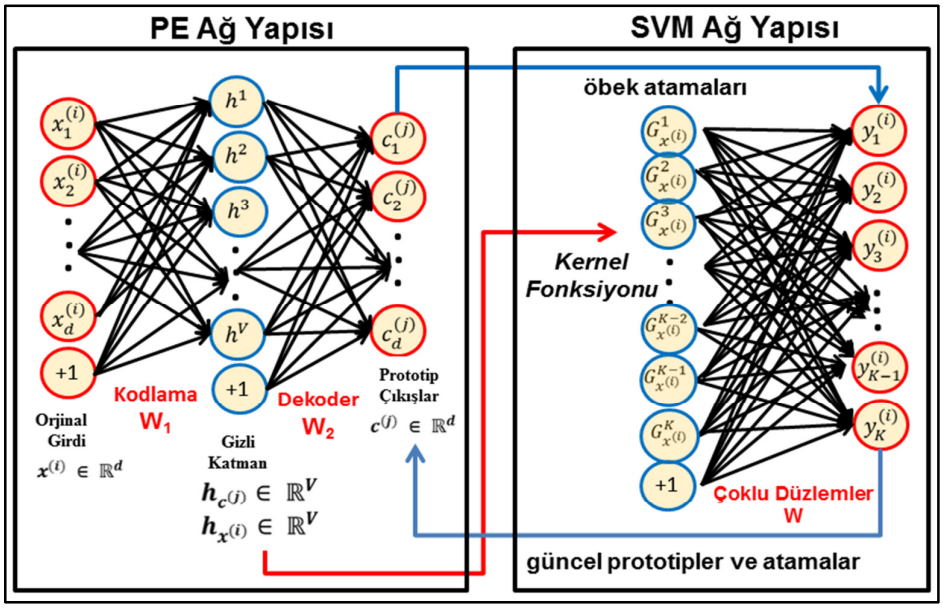
yoluyla görev sahası öğrenilmeye çalışılmıştır. Diğer makine ise çok daha geniş bilgiye sahip bir şekilde 'Öğretmen' (T) rolünü üstlenerek S'ye imge sınıflarını öğrenmede rehberlik yapmıştır. T göreceli nitelikleri imge sınıflarını rastgele sıralamak suretiyle denetimsiz bir yapıda ilklendirmiş ve bahse konu nitelikler S ile iletişim kanalı olarak kullanılmıştır. Diğer taraftan, S görsel kategorileri orta seviye nitelik uzayında ve başlangıçta kendisine sınırlı sayıda verilen etiketlenmiş eğitim seti ile görüntü kategorilerini modellemiştir. S ilk olarak etiketlenmemiş verinin oluşturduğu geniş bir havuz içerisinden en faydalı örneği seçerek tahmin ettiği sınıf etiketi ile bir sorgu üzerinden bu örneği T'ye iletmek suretiyle etkileşim başlatılmıştır. Yarı denetimli öğrenmeden farklı olarak, T havuz verisinin etiket bilgisine haiz olduğu için, sadece S'nin tahmininin doğru veya yanlış olduğunu bildirmemiş, aynı zamanda her iki durumda da sınıflar üzerinden karşılaştırmalı bir nitelik tabanlı açıklama ile S'ye cevap vermiştir. Ancak tahminin yanlış olması durumunda seçilen örneğin gerçek sınıf etiket bilgisi açığa çıkarılmamıştır. Böylece S tarafından yanlış tahmin edilen örnek yeniden etiketsiz veri havuzuna atılarak eğitim verisi sanal olarak arttırılmıştır. Sınıf modellerinin güncellenmesinde, göreceli nitelik uzayı üzerinde oluşturulacak bir takım koşullanmalar sayesinde etiketsiz veri içerisinden bazı adaylar seçilerek S'nin elindeki etiketli veri ile birleştirilmiştir. Bu dönüşümlü güncellemeye S tarafından etiketsiz veri havuzu boşaltılıncaya (doğru şekilde etiketlenene) kadar devam edilmiştir. Önerilen algoritma literatürde yaygın bir şekilde kullanılan yüz (PubFig) ve doğal sahne (OSR) veri setlerinde sınıflandırma maksatlı olarak test edilmiş ve deneylerde diğer benzer yaklaşımlara göre dikkate değer bir performans artışı gözlenmiştir.

2.1. Melez Sinir Ağları Mimarisinde Öbekleme

Tez çalışmasının bu kısmında, yeni bir nitelik uzayının veriyi temsil etmek maksadıyla denetimsiz bir şekilde öğrenildiği yapı ile DVM'nin maksimum ayırtaç özelliğinin gücünü kullanarak öbeklemeyi daha ince ayarda yeniden güncelleyen yapının kompozit bir şekilde kullanıldığı çok katmanlı sinir ağı mimarisi önerilmektedir. Önerilen yapı basit şekliyle Şekil 2.1'de gösterilmiştir ve iki parçadan oluşmaktadır.

İlk aşama, girdi verisinin en yakın prototip vektörlerine haritalanmasını sağlayan üç katmanlı bir ağdan oluşmaktadır. Bu ağda girdi ile gizli katman arasında bulunan

kodlama ağırlık vektörleri başlangıçta rastgele seçilmiş olan prototip vektörlerinin ikili şekilde ilişkilendirilmesi neticesinde oluşturulan ve Voronoi diyagramlarındaki benzer yapıda çok boyutlu düzlemler şeklinde ilklendirilmektedir. Bu işlem sayesinde aynı zamanda gizli katmanda yer alan neron sayısı da otomatik bir biçimde tespit edilmektedir. İlaveten; sinir ağının öğrenilmesi aşamasında her bir örneğin gizli katman aktivasyon çıkışları amaç fonksiyonu içerisinde bahse konu örneğe en yakın prototip vektör aktivasyonları ile sınırlandırılmaktadır. Örnek girdiler ise orjinal nitelik uzayı yerine yinelemeli olarak elde edilen gizli uzayda en yakın prototip vektöre atanmaktadır. Bu ağ mimarisine Prototip Kodlama Ağı (PKA) adı verilmektedir ve detayları Bölüm 2.1.1'de verilmiştir.



Şekil 2.1. Öbekleme için önerilen çok katmanlı kompozit ağ mimarisi

İkinci aşamada yer alan ağ yapısı ise PKA'dan elde edilen öbekleri başlangıç koşulları olarak ele alır ve bu öbekleri güncelleme amacındadır. PKA'nın bulduğu nihai veri-öbek atamaları bu ağda başlangıç öbek etiketleri olarak ve orjinal veriye ait gizli katman çıkışları ise yeni girdi sinyalleri olarak kullanılır. Bu kapsamda son aşama DVM konseptini amaç fonksiyonu olarak işletir. Bu ağ geçerli prototip vektörlerini kernel fonksiyonu içerisinde kerteriz noktaları olarak kullanır ve öbek sınırları arasındaki ayırdımı maksimize etmeyi amaçlar. Neticede, Destek Vektörleri Makine Ağı (DVMA) olarak adlandırılan ağda öbeklerin daha ayırt edici şekilde ayrıştırılması hedeflenir ve detaylar Bölüm 2.1.2'de yer almıştır.

Her iki ağ mimarisinde kendilerine özgü gizli nitelik uzaylarında SOM'a benzer bir yaklaşımla öbekler yinelemeli olarak güncellenir. Bu noktada PKA'da gizli uzay gizli katman çıktıları ile temsil edilirken DVMA'da gizli uzay çıkış katmanında temsil edilmektedir. Son olarak önerilen her iki ağ mimarisindeki parametreler açgözlü katman tabanlı öğrenme prensibinde bütünsel olarak güncellenmektedir ve detayları Bölüm 2.1.3'te verilmiştir.

2.1.1. Çalışmanın literatüre özgün katkısı

Bu tezin YSA ile öbekleme kapsamında literatüre katkısını iki şekilde açıklamak mümkündür. Öncelikle, yapay sinir ağı mimarisinde öbekleme odaklı yeni bir yaklaşım olarak üç katmanlı Prototip Kodlama Ağı (PKA) önerilmektedir. Klasik iki katmanlı öbekleme yaklaşımının aksine, PKA giriş katmanındaki her bir örneği çıkış katmanında daha önceden atanmış prototip vektörüne atamakta, gizli katmanda oluşturulan gizli nitelik uzayını ise öbekleri güncellemek amacıyla kullanmaktadır. Oysa iki katmanlı öbekleme mimarisinde çıkış katmanındaki nöronlar öbek merkezlerini temsil etmiş ve aradaki ağırlık vektörleri ise sendroid vektörlerini oluşturmuştur.

Diğer yönden, PKA giriş verisini çıkış katmanında kendisine haritalayan AEs yapısından açık biçimde farklıdır. Kendine kodlayıcının yapısı itibariyle çıkış katmanındaki varyans pratik olarak artacaktır. İlaveten, gizli katmandaki neron sayısı giriş katmanındaki boyuttan fazla olduğu takdirde, veri çıkışta kendine kodlandığı için, özdeşlik fonksiyonu öğrenilebilir. Diğer bir taraftan, PKA'nın giriş-gizli katmanlar arasında yer alan kodlama ağırlıkları Voronoi diyagramına benzer ve öbekleri ikili olarak birbirinden ayıracak şekilde ilklendirilmektedir. Bu tip bir ilklendirme sayesinde sadece PKA'nın kodlama ağırlıkları için rastgele seçimden daha iyi bir başlangıç noktası değil, aynı zamanda rastgele ilklendirilen öbek merkez vektörlerine uygun olarak dinamik bir şekilde gizli katmandaki neron sayısı da tespit edilecektir.

İkinci olarak, DVMA'nın öbeklemede ince ayar günceleme olarak kullanıldığı bir fırsatçı katman tabanlı öbekleme mimarisi önerilmektedir. Bahse konu kompozit ağ mimarisinde, PKA'nın optimizasyonu sonrasında girdi verisi, orjinal nitelik uzayından ziyade, PKA'nın gizli katman çıkışından elde edilen yeni nitelik uzayında

temsil edilerek DVMA'ya girdi olarak beslenmektedir. PKA'dan elde edilen veri-öbek atamaları başlangıç denetimi mahiyetinde DVMA'da kullanılmaktadır.

Son olarak öbekleme algoritmasından elde edilen güncel prototip vektörleri her yinelemede Gaussian Kernel Fonksiyonu (GKF)'nda kerteriz noktaları olarak kullanılmak suretiyle geleneksel DVM öğrenme yapısındaki karmaşıklık azaltılmıştır. Zira geleneksel DVM metodolojisinde her bir girdi örneği kernel fornksiyonlarında kullanıldığından karmaşıklığı O(N) iken önerilen yöntemde karmaşıklık O(K) (N: örnek sayısı, K: öbek sayısı ve N >> K)'ya düşmektedir.

2.1.2. Prototip kodlama ağı

PKA temelinde kendine kodlayıcı (Auto-Encoder - AE) [36,39] ile benzer bir düşüncenin ürünüdür. AE'de girdi sinyali öncelikle arada yer alan bir gizli katmanda üretilen gizli nitelik uzayına taşınmış ve çıkış katmanında ise yine girdi sinyalinin kendisine haritalanmıştır. Ancak önerilen PKA'da, AE'nin aksine, yinelemeli olarak her bir girdi sinyali çıkışta daha önce atanmış olduğu öbeği temsil eden prototip vektörü ile eşleştirilmektedir. Dolayısı ile her bir öbek rastgele ilklendirilmiş olan bir prototip vektörü ile temsil edilmektedir. Metrik sistem olarak Euclidean tıpkı k-means [15] yapısındaki gibi en yakın komşuluğun tespitinde kullanılmaktadır.

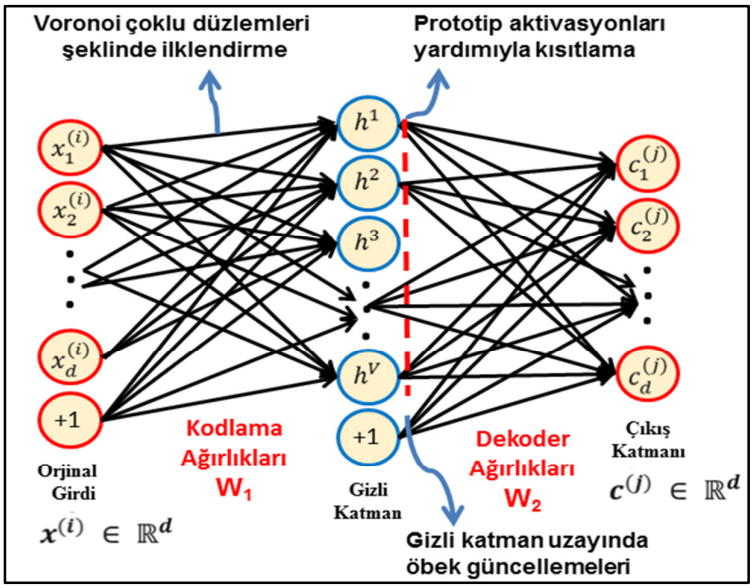
Rastgele seçilen prototip vektörleri arasından Euclidean mesafesine göre her bir örnek için en yakınındakinin seçilmesi ile elde edilen örnek-prototip çiftleri X = $\{x^{(i)},c^{(j)}\}$, $x^{(i)}$ ve $c^{(j)} \in R^d$; i=1,2,3,…, N; j=1,2,3,…, K , K kullanıcı tarafından önceden belirlenmiş öbek sayısı ve N sistemde bulunan örnek girdi sayısı olmak üzere; amaç fonksiyonu şu şekildedir;

$$\forall x\,;c^{(j)} = c^{\left(\text{argmin}_{k=1,2,3,\ldots,K}\left\|x^{(i)}-c^{(k)}\right\|\right)} \tag{2.1}$$

$$J_X(W,b) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\left\|h_{W,b}(x^{(i)}) - c^{(j)}\right\|^2 + \tag{2.2}$$
$$\beta\sum_{v=1}^{V}\left(h^{v}_{c^{(j)}}\log\frac{h^{v}_{c^{(j)}}}{h^{v}_{x^{(i)}}} + (1-h^{v}_{c^{(j)}})\log\frac{1-h^{v}_{c^{(j)}}}{1-h^{v}_{x^{(i)}}}\right) + \frac{\lambda}{2}\sum_{W_1,W_2}\|W\|^2$$

Amaç fonksiyonu, $J_X(W,b)$, basitçe üç temel elemana sahiptir. Bunlar sırasıyla Ortalama Karesel Hata (OKH), ayrışma ve ağırlık düzenlemedir. OKH ağın

mimarisine uygun tahminleri, $h_{W,b}(x)$, ile çıkış katmanında arzu edilen çıkış sinyalleri (her bir örnek için en yakın prototip vektörü), $c^{(j)}$, arasındaki farkı cezalandıran bir fonksiyon terimidir. Ayrışma terimi ise ağ içerisinde yinelemeli olarak gizli katmanda üretilen gizli nitelik uzayında girdi sinyali ile en yakınlarındaki prototip vektör projeksiyonlarının benzer olmasını zorlayan bir fonksiyondur. Bu girdi sinyalin gizli katman çıkışlarının, h, entropi tabanlı bir formül olan Kullback-Leibler divergence, $KL(h_c \parallel h_x)$ ile kısıtlanması prensibine dayanır. Neticede orjinal nitelik uzayı yerine herbir örneği gizli nitelik uzayında en yakınındaki öbeğe atamaya çalıştığımızdan, hedef nitelik uzayının çatısını teşkil eden ve kodlama ağırlıklarını temsil eden baz vektörlerin amaca uygun güncellenmesi için amaç fonksiyonuna bu şekilde bir yakınsama terimi eklenmesi akla uygun gelmektedir. Ağırlık düzenleme terimi, diğer bir taraftan, sistemin aşırı doyuma (over-fitting) ulaşmasını, yani eğer ağ parametreleri çok ve sistem karmaşık ise bazı ağırlık parametrelerinin sönümlendirilmesini sağlamaktadır. β ve λ ise KL ayrışması ve ağırlık düzenleme terimlerinde yer alan denge sabitleridir ve belirli doğrulamalarla önceden ayarlanmaları gerekmektedir.



Şekil 2.2. Ptototip kodlayıcı ağ mimarisi

Önerilen PKA mimarisindeki diğer önemli bir nokta, giriş-gizli katman arasında yer alan kodlama ağırlık vektörlerinin ilklendirilmesidir (Şekil 2.2'deki W_1 ağırlıkları). Bahse konu kodlama ağırlıkları Voronoi diyagramına [48, 55] benzer yapıda nitelik

uzayını çok boyutlu düzlemler ile nitelik uzayını sınırlı sayıda bölgeye bölütlemek amacıyla düzlemlerin ikili kullanılması şeklinde ilklendirilmektedir. Voronoi düzlemleri öbek merkezlerini çekirdek noktalar olarak tesis ederek oluşturulmaktadır. Prototip vektörlerinin başlangıçta rastgele seçilmesinin ardından, her bir düzlem ikili olarak tüm prototipleri birleştiren doğruların tam ortasından dik bir şekilde geçen düzlemi temsil eden vektörler olarak bulunur. Ancak öbek sayısının ikili kombinasyonları şeklinde hesaplanan bu düzlemlerin toplamı ağ mimarisinde işlemsel karmaşıklığın üssel olarak artmasına sebep olur. Zira düzlemler esas itibariyle gizli katmandaki nöronları ifade etmektedir ve bu şekildeki ikili kombinasyonlarının çoğunun nitelik uzayını K bölgeye ayırmak için gereksiz olduğu aşikardır.

Dolayısıyla şu şekilde bir yol izlenmiştir: Sırası ile seçilen her bir prototip vektör için kalan prototip vektörler seçilene göre Euclidian mesafesinde azdan çoğa doğru sıralanır. Sonra seçilen prototip ile en yakınındaki prototip vektör arasına ikili eşleme yapısına uygun ve yukarıda bahsedildiği şekilde bir düzlem oluşturulur. Seçilen prototip vektör ile en yakınındaki diğer prototipi ayırmak üzere elde edilen bu düzlem seçilen prototip için minimum ayırtaç marjini olarak kabul edildiğinden, eğer bu hesaplanan düzleme göre seçilen prototip ile diğer kalan bir prototip daha fazla bir mesafede ayırt edilebiliyorsa o halde bu ikili arasında yeni bir düzlem oluşturma ihtiyacı kalmamıştır. Bu prosedür ikili eşleştirilen prototip çiftleri kalmayıncaya kadar devam ettirilir. Yapılan deneylerde bu açgözlü yaklaşımın düzlem sayısını dramatik bir biçimde düşürdüğü gözlenmiştir.

Dikkate değer diğer bir husus ise, bu sayede sadece kodlama ağırlıkları daha mantıklı bir şekilde ilklendirilmemiş; aynı zamanda gizli katmandaki nöron sayısı da rastgele seçilmiş prototip vektörlerinin birbirlerine olan göreceli mevkileri baz alınarak dinamik bir yapıda tespit edilmiştir. Gizli-çıkış katmanları arasındaki kod çözücü ağırlıklar (W_2) ise tıpkı prototip vektörleri gibi rastgele ilklendirilmiştir. Doğrusal olmayan fonksiyonlar yardımıyla denklemler oluşturmak üzere, gerek gizli gerekse çıkış katmanlarında aktivasyon fonksiyonu olarak sigmoid (lojistik) fonksiyonları kullanılmıştır;

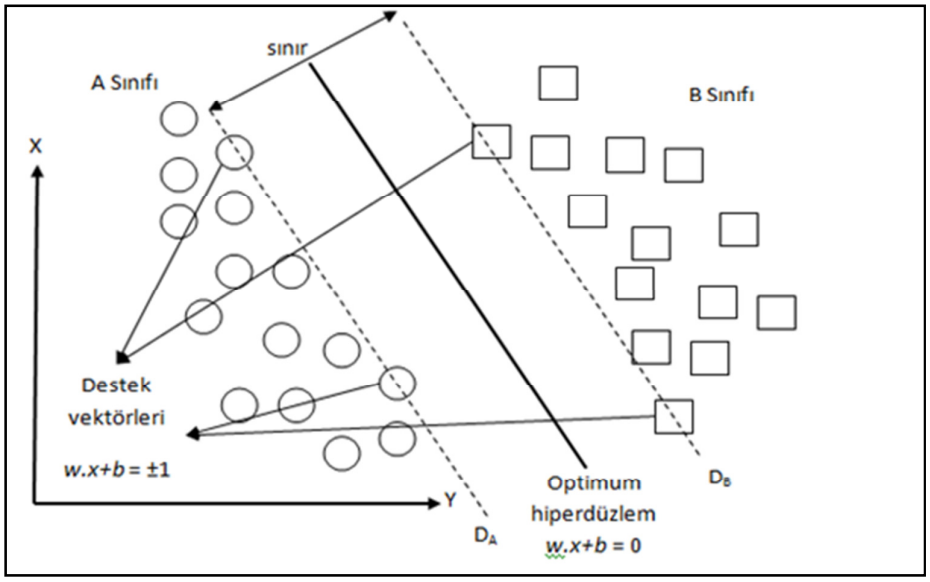
$$\text{Sigmoid}(t) = \frac{1}{1+e^{-t}} \tag{2.3}$$

PKA içerisinde mevcut parametrelerin optimizasyonunda gradyen azaltım algoritması koşulmuştur. Kısmi hata türevleri geleneksel ileri besleme -geri yayılım metodu yardımıyla hesaplanmaktadır. Böylece ağ yapısındaki ağırlık ve yanlılık parametreleri mini paketler halinde ve girdi verisi – öbek atanmaları ile gizli katman aktivasyon kısıtlamaları sabit tutulmak suretiyle güncellenir. Öbeklenecek tüm veri setinin mini paketler halinde sistemden bir geçişinin ardından (epok), bu defa prototip vektörleri güncellenir. Prototip vektörlerinin güncellenmesinde Denklem (1.6-8)'de yer alan SOM ağ mimarisi esas alınır. Ancak girdi örnekleri ile prototip vektörleri öncelikle gizli katman çıkışları kullanılarak yeni gizli nitelik uzayına taşınır ve her bir örnek için en yakın (kazanan) prototip vektör seçimi, $c^{(w)}$, ile Gaussian fonksiyonundan elde edilen komşuluk ağırlık çarpanları, $h_{jw}(t)$, yine bu gizli uzayda hesaplanır. Diğer taraftan prototip vektörlerini güncelleme prosedürü ise yine orjinal nitelik uzayında icra edilir. Çünkü giriş katmanında yinelemeli olarak aynı girdi örnekleri kullanılmaktadır ve neticede çıkış katmanındaki en yakın güncellenmiş ptototip vektörleri de orjinal nitelik uzayında olmak zorundadır.

Ayrıca not edilmesi gereken diğer bir husus, prototip vektörlerinin güncellenmesi neticesinde sonraki yinelemede yer alacak veri-öbek atanması ile gizli katman kısıtlama değerlerinin de değişecek olmasıdır. Bu noktada, PKA'nın kodlama ağırlıkları öbekler arasındaki sınırları temsil eden düzlemler olarak tahmin edilmekte ve her bir örnek için gizli katman çıkışları ise bahse konu öbek sınırlarına pozitif ve negatif yönlerdeki mesafeleri oluşturmaktadır. Girdi örnekleri gizli katman çıkışlarında daha yüksek boyutlu nitelik vektörleri olarak temsil edileceğinden, bu yeni uzayda daha iyi ayırt edilebileceği düşünülebilir. Son olarak; ağ parametreleri (ağırlık ve yanlılık vektörleri) ile prototiplerin güncelleme prosedürü yinelemeli olarak öbeklerde artık herhangi bir değişiklik olmayıncaya (hata değeri belirli bir eşik değerinin altına düşmüştür) veya yineleme sınırına ulaşıncaya kadar devam ettirilir.

2.1.3. Öbeklerin ayarlanmasında destek vektörleri makinesi ağı

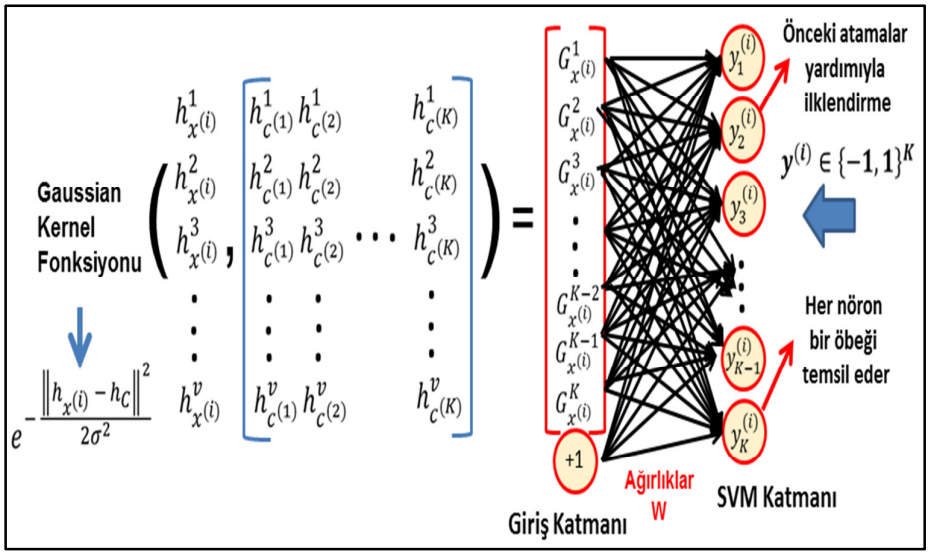
DVM nitelik uzayını doğrusal bir şekilde pozitif ve negatif olarak iki kategoriye sınıflandırmak için kullanılan güçlü bir denetimli öğrenme aracıdır. Amacı pozitif ve negatif taraflar arasındaki ayırdımı maksimize edecek şekilde bir düzlem oluşturmaktır. Bu düzleme dik olan ağırlık vektörü girdi uzayı boyutunda doğrusal ayırtaç fonksiyonu görevini üstlenmektedir [15, 45]. DVM'daki aralık ile her iki taraftaki birbirlerine yakınlığı belirli bir şekilde kısıtlanmış olan örneklerin, yani destek vektörlerinin arasındaki boşluk kasdedilmektedir. Ana fikir, destek vektörleri ile arasındaki toplam mesafeyi minimum seviyede tutacak bir düzlemi optimize etmektir. Bahse konu mesafe ise destek vektörleri ile tespit edilecek düzleme dik olan bir ağırlık parametre vektörünün nokta çarpım (dot product) operatörü yardımıyla doğrusal çarpılması şeklinde hesaplanmaktadır. Nokta çarpım operatörü esasında düzleme dik olan ağırlık vektörü üzerinde örneklerin artı ve eksi yönlerde işaretlenmiş yansımalarıdır. İlaveten, orjinal nitelik uzayındaki merkezden çıkan ağırlık vektörünün düzlemi dik olarak kestiği ve yeni nitelik uzayındaki orjini temsilen yanlılık terimi de sisteme ilave edilir. DVM'nin eğitilmesinden sonra, optimize edilmiş ağırlık parametre vektörü ile yeni bir örneğin skaler çarpımından elde edilen sonuç bu yeni örneğin pozitif veya negatif olarak sınıflandırılmasında kullanılmaktadır. Klasik DVM yapısı Şekil 2.3'te verilmiştir.



Şekil 2.3. DVM sınıflandırıcısı ve destek vektörleri

Aslına bakacak olursak; DVM doğrusal ayırtaç fonksiyonudur ve yalnız doğrusal olmayan şekilde kategorilendirilebilecek veri setlerini doğası gereği tatmin edici bir başarı yüzdesiyle malesef ele alamaz. Tam da bu noktada bir takım kernel fonksiyonları (Gaussian, polynomial, chi-square, histogram intersection gibi) literatürde tanıtılmıştır ve bunların amacı doğrusal olmayan DVM sınıflandırıcıları tesis etmektir. Doğadaki pek çok uygulama doğrusal olmayan çözümler içerdiği için DVM gerçek anlamda popülaritesini kernel fonksiyonlarına boçludur. DVM esasında kernel bakış açısında yine doğrusallık özelliğini muhafaza etmektedir; fakat girdi verisi bahsedilen doğrusal olmayan kernel fonksiyonları ile DVM'den önce işleme sokulduğu için DVM işlemlerine doğrusal olmayan yeni nitelik uzayında devam etmekte. Haliyle DVM sonuçları da doğrusal olmayan çözümler doğurmaktadır. Bu nedenle, DVM ayırtaç fonksiyonunun kendisi karmaşık hale getirilmektense daha yüksek boyutlu ve karmaşık bir yeni nitelik uzayında sınıflandırma problemi çözülmektedir.

Önerilen çok katmanlı sinir ağı mimarisinin bu ikinci aşamasında, daha önceden elde edilen öbekleri mihenk alarak bunları daha da geliştirmek (ince ayar) için, çok öbekli problem çözümünü sağlayacak bire-karşı-hepsi (one-vs-all) DVM topolojisine sahip yeni bir ağ mimarisi tasarlanmıştır. DVMA'nın iki katmanlı denetimli öbekleme sağlayan bu yapısı Şekil 2.4'te gösterilmiştir.



Şekil 2.4. Öbekleri ince ayarlayan DVMA mimarisi

Girdi verisi X = $\{x^{(i)}\}$ ve başlangıç öbek prototip vektörleri C = $\{c^{(j)}\}$ verildiği takdirde; Gaussian Kernel Fonksiyonu (GKF) yardımıyla oluşturulan yeni nitelik uzayında, $K(x^{(i)},C)$, DVMA'ya girecek yeni veri sinyalleri öncelikle şu şekilde hesaplanır;

$$K(x^{(i)},c) = e^{\frac{-\left\|x^{(i)} - c\right\|}{2\sigma^2}} ;\ c \in C \tag{2.4}$$

Burada σ örneklerin prototip vektörleri ile komşuluk ilişkilerini tanzim eden ölçekleme parametresidir. Dolayısıyla her bir örnek girdi orjinal nitelik uzayı yerine GKF cevaplarının oluşturduğu doğrusal olmayan yeni bir nitelik uzayında temsil edilmektedir. Bu noktada prototip vektörleri orjinal nitelik uzayındaki kerteriz noktalarını temsil etmektedir. Ayrıca dikkat edilmesi gereken diğer bir husus, DVMA'ya girecek yeni girdi sinyallerinin boyutlarının öbek sayısı, K, ile ifade edilmesidir. DVM denetimli bir öğrenme algoritması olduğu için, daha önceki bir uygulamadan (PKA gibi) elde edilecek veri-öbek atamaları DVMA mimarisinin çıkış katmanında başlangıç öbek etiketlerini oluşturacaktır. Bu şartlarda DVMA'daki kısıtlanmayan amaç fonksiyonu şu şekilde ifade edilir;

$$J_{G_x,y}(W,b) = P\frac{1}{N}\sum_{i=1}^{N}\max(1 - w^T G_{x^{(i)}} y^{(i)},0)^2 + \frac{1}{2}\sum\left\|W\right\|^2 ;\ y^{(i)} \in \{-1,1\}^K \tag{2.5}$$

Burada P beklenen öncül öbeklerinin aksine çıkışta marjin kısıtlamasına uymayarak yanlış öbeklere atanan girdi örneklerini cezalandıran denge sabitini temsil etmektedir. $G_x^{(i)}$ DVMA'ya girecek her bir örneğin Denklem (2.3)'deki GKF çıkışını, yeni nitelik uzayındaki projeksiyonunu belirler. W matrisi ise matematiksel ifadeyi basitleştirmek adına yanlılık parametrelerini de içerisine alan, K adet öbeği birbirinden ayırt eden düzlemlere dik ve tıpkı PKA'daki kod çözücü (W2) ağırlık vektörleri gibi başlangıçta rastgele seçilmiş bulunan parametre matrisidir. Daha önce bahsedildiği üzere, DVM basitçe örnek uzayını iki parçaya ayırır. DVM yaklaşımını çok öbekli problem çözümüne genellemek için bire-karşı-hepsi yaklaşımı benimsenmiştir. K-öbek probleminde, bir öbek için diğer tüm öbeklerde yer alan örnekler negatif tarafı temsil edecek şekilde birbirinden bağımsız K adet doğrusal DVM öğrenilmiştir [45]. Sonuçta her bir girdi için çıkış sinyali, $y^{(i)}$, bu örneğin daha önceden atanmış olduğu öbeği temsil eden çıkış nöronu için 1 ve kalan diğer tüm

öbekler için ise -1'ler ile ifade edilmektedir. Stokastik gradyen azaltım algoritması ise amaç fonksiyonunun çözümünde şu şekilde kullanılır;

$$h_W(G_x) = W^T G_x \tag{2.6}$$

$$\delta w = \frac{\delta J(W,b)}{\delta h_W} = -2Py \max(1 - h_W(G_x)y, 0) \tag{2.7}$$

$$\Delta w = \alpha\left(\frac{1}{N}\sum(G_x^T \delta w) + w\right) \tag{2.8}$$

$$w_{yeni} = w_{eski} - \Delta w \tag{2.9}$$

Burada δw her bir girdi sinyalinin geri propagasyondan elde edilen kısmı türevini, Δw ise girdi – çıktı katmanları arasındaki W parametre ağırlık vektörüne uygulanacak olan ortalama hata düzeltme vektörünü ifade etmektedir. Bu noktada L_2 norm düzenlemesinden gelen ilave hata düzeltmesinin (Δw içerisinde w) yanlılık terimlerine etki etmediği unutulmamalıdır. Ayrıca α sistem öğrenme oranı ve h_w ağ çıkışını temsil eden hipotez fonksiyonu, yani beklentidir. DVMA'da uygulanan tahminleme prosedürü aslında soft-max kuralı ile büyük oranda benzerlik göstermektedir. Ancak hata türevlerinin bütünleşik bir yapıda matris çarpımı ile tespit edildiği soft-max kuralının aksine, DVMA çıkış nöronları hata değerlerini birbirinde bağımsız bir şekilde bireysel olarak geriye yaymıştır. Bu da matematiksel ifade ile W matrisinin bütünsel kullanımı yerine sütunlarını temsil eden öbek ağırlık vektörlerinin, $w^{(j)}$, hata türev hesaplamalarında ayrı ayrı kullanılmasına denktir.

DVMA'nın oluşturulmasını müteakip ağırlık parametreleri (W) yinelemeli bir şekilde ve girdi verisinin mini paketlere ayrılarak hata türevlerinin bu paketlerde hesaplanması ile optimize edilmiştir. Öncelikle geçerli veri-öbek atamaları DVMA'nın çıkış katmanında kullanılmak suretiyle ve maksimum marjinli amaç fonksiyonu içerisinde öbekleri birbirinden ayıran düzlemler optimize edilmektedir. Tüm verinin (epok) düzlem optimizasyonunda kullanılmasının ardından PKA'daki yapıya benzer şekilde öbek güncelleme prosedürü icra edilir. Şöyle ki; DVMA'nın çıkış katmanındaki aktivasyonlar yeni nitelik uzayı olarak kullanılır. Çok katmanlı ağ mimarisinin birinci aşaması olan PKA'daki gizli katman çıkışları da benzer şekilde kullanılmıştı. Ardından SOM Denklem (1.3)'deki öbek güncelleme şeması hem

prototip vektörlerinin hem de örnek-öbek atamalarının yeni nitelik uzayında güncellenmesini sağlamaktadır. Örnek-öbek atamaları güncellendikten sonra ise istenilen çıkış sinyalini temsil eden ve her bir örnek için ikili (1 ve -1'lerden oluşan) oluşturulan denetim vektörü, $y^{(i)}$, yeniden tanımlanır. Ayrıca dikkat edilmesi gereken husus, GKF'da kerteriz noktalarını temsil eden prototip vektörleri güncellendiği için kernel çıkışları, dolayısı ile DVMA girişleri de yeniden hesaplanır. Neticede her ne kadar DVMA'ya giren veri orjinal haliyle korunsa da; her epok sonrasında nitelik uzayı ve girdi sinyali değişmektedir. DVMA'nın sonlandırılma koşulları PKA ile aynıdır.

2.1.4. PKA ve DVMA mimarileri ile melez ağ yapısında öbekleme

Şimdiye kadar rekabetçi öbekleme kategorisinde iki farklı sinir ağı mimarisi sunulmuştur ve bunlar birbirlerinden farklı alt amaçlara hizmet etmektedir. PKA'nın alt hedefi öbeklemede verinin daha ayırt edici bir nitelik uzayında temsil edilmesini sağlamak maksadıyla denetimsiz bir şekilde yeni gizli nitelik uzayı öğrenmektir. Bu nedenle prototip vektörleri ve veri-öbek atamaları bu yeni nitelik uzayında yinelemeli bir biçimde güncellenmiştir. Diğer taraftan DVMA mimarisinde, daha önceden tesis edilmiş bulunan öbekler arasındaki marjin maksimize edilecek şekilde ve geçerli örnek-öbek atamaları öncül denetim olarak sisteme dahil edilerek önceki öbekleme yeniden dizayn edilmiştir. Aynı zamanda DVMA'nın çıkış katman aktivasyonları da öbek güncelleme prosedüründe yeni nitelik uzayı olarak kullanılmıştır. Dikkat edilirse, her iki ağda yer alan bütün katmanlardaki nöron sayıları da serbest parametre olmaktan çıkarılmış ve otomatik olarak hesaplanmıştır. Bu aşamada, bahse konu iki ağ mimarisi açgözlü katman tabanlı öbekleme mantığıyla bütünsel olarak birlikte kullanılmış ve bu çok katmanlı ağ mimarisine ilişkin pseudo-code Şekil 2.5'te sunulmuştur.

Bu çok katmanlı ağın başında PKA olduğu için; ilkin PKA'nın kodlama ağırlıklarını (W_1) temsil eden parametreler, Voronoi diyagramına benzer bir yapıda başlangıçta rastgele seçilmiş olan prototip vektörlerinin ikili eşleştirilmesi yoluyla ve detayları Bölüm 2.1.1'de belirtildiği şekilde ilklendirilir. Yeni nitelik uzayında prototiplerin ve aynı zamanda ağ parametrelerinin güncellenmesi kapsamında, PKA'da esasında Beklenti-Maksimizasyon (Expectation-Maximization) algoritması koşulmaktadır. Beklenti aşamasında prototip vektörleri ve haliyle örnek-öbek atamaları ve gizli

katman çıkış kısıtlamaları sabit tutularak bir epok içerisinde ağ ağırlık parametreleri (W_1 ve W_2) güncellenir. Maksimizasyon aşamasında ise yeni nitelik uzayının baz vektörlerini temsil eden PKA'ya ait kodlama ağırlıkları (W_1) sabit tutularak öbeklerin yeni nitelik uzayında güncellenmesi icra edilir. Bu alternatif döngü daha önceden belirlenen bir kıstasa ulaşılıncaya kadar devam ettirilir.

Algoritma 1 Öbekleme için önerilen melez yapay sinir ağı mimarisi

Girdi:
Öbeklenecek Veri: $X = \{\boldsymbol{x}^{(i)}\}, \boldsymbol{x}^{(i)} \in R^d$; i=1,2,3,..., N
Rastgele ilklendirilen prototip vektörleri: $C = \{\boldsymbol{c}^{(j)}\}, \boldsymbol{c}^{(j)} \in R^d$; j=1,2,3,..., K
Çıktı:
Nihai veri-öbek atamaları: $D_c = \{\boldsymbol{x}^{(i)}, \boldsymbol{c}^{(j)}\}$

1. **BAŞLA:** PE ağ yapısı
2. Ağ parametrelerini, kodlayıcı ve dekoder ağırlıkları ilklendir: $\mathbf{W_1}$ & $\mathbf{W_2}$
   2.1. $\mathbf{W_1}$ ağırlıklarını Voronoi çoklu düzlemleri ile ilklendir (Bölüm 2.1.1)
   ($\mathbf{W_1}$, #NeuronsOfHiddenLayer) ← *InitW1(C)*
   2.2. $\mathbf{W_2}$ ağırlıklarını rastgele ilklendir
3. D_c değerlerini gizli katman çıkışlarında hesapla (gizil uzay)
4. **HER** epok için
5. $\mathbf{W_1}$ & $\mathbf{W_2}$ ağırlıklarını D_c değerlerini sabit tutarak Denklem (2.2) ile güncelle
6. *C* ve D_c değerlerini $\mathbf{W_1}$ & $\mathbf{W_2}$ ağırlıklarını sabit tutarak gizil uzayda Denklem (1.6-7-8) ile güncelle
   BİTİR: HER
7. **BİTİR:** PE ağ yapısı

8. **BAŞLA:** SVM ağ yapısı
9. Ağ parametrelerini, **W**, rastgele ilklendir
10. **HER** epok için
11. PE ağ yapısında *X* ve *C* vektörlerinin gizli katman çıkış değerlerini hesapla
12. Gaussian Kernel Fonksiyonu ile SVM ağ yapısının giriş sinyallerini elde et (Denklem 2.4)
13. **W** ağırlıklarını D_c değerlerini sabit tutarak Denklem (2.5-9) ile güncelle
14. *C* ve D_c değerlerini **W** ağırlıklarını sabit tutarak gizil uzayda Denklem (1.6-7-8) ile güncelle
15. **END:** HER
16. **END:** SVM ağ yapısı

Şekil 2.5. Açgözlü katman tabanlı melez ağ mimarisine ait pseudo-code

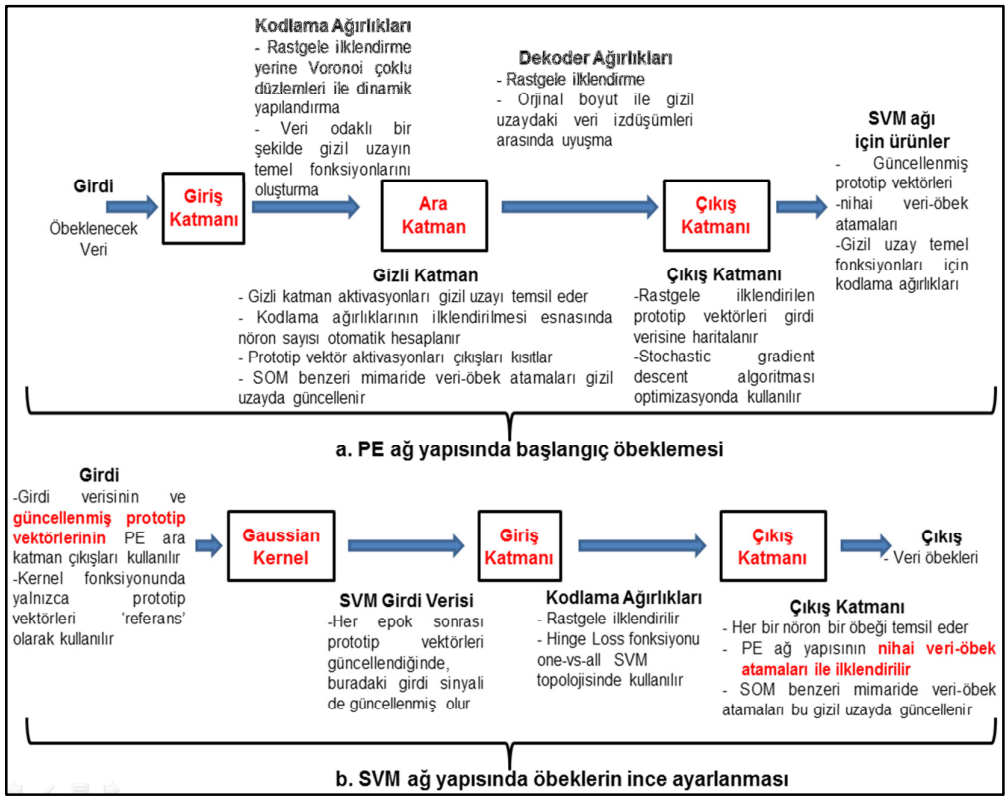
Çok katmanlı kompozit ağ mimarisinin ikinci kısmı olan DVMA'da aynı girdi verisinin orjinal nitelik uzayında tekrar giriş yapmasına izin verilmez. Bunun yerine girdi verisi, $x^{(i)}$, ile PKA tarafından finalde bulunan prototip vektörlerinin, $c^{(j)}$, PKA gizli katman çıkışındaki yeni nitelik uzayı projeksiyonları şu şekilde hesaplanır;

$$H(\varphi) = \mathrm{Sigm}(W_1^T \varphi) = \frac{1}{1 + e^{-W_1^T \varphi}} \tag{2.10}$$

$$H_x = \{h(x^{(i)}) = h_{x^{(i)}} \mid i = 1,2,3,...,N;\ h_x \in R^v\} \tag{2.11}$$

$$H_c = \{h(c^{(j)}) = h_{c^{(j)}} \mid j = 1,2,3,...,K;\ h_c \in R^v\} \tag{2.12}$$

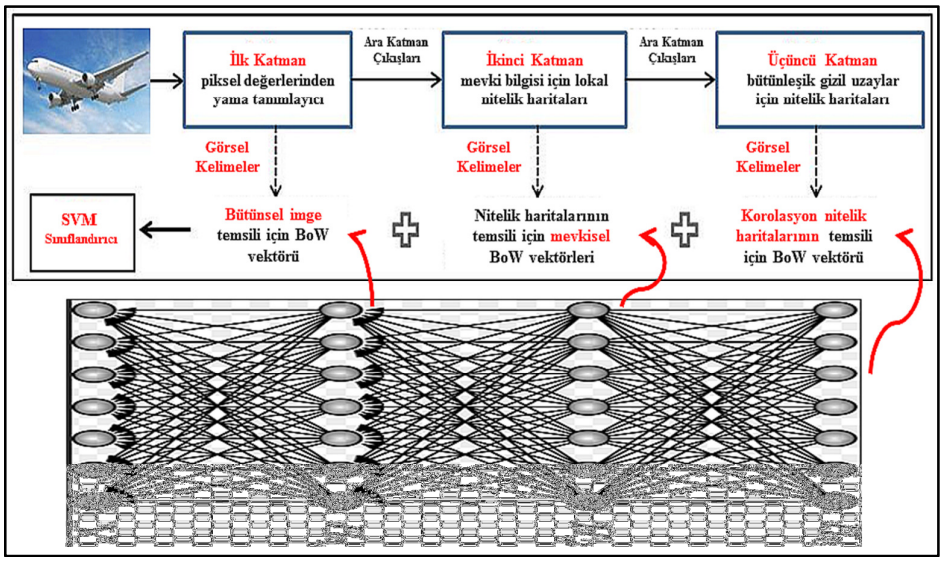
PKA tarafından elde edilen nihai veri-öbek atamaları da DVMA'da başlangıç denetim bilgisi olarak ayrıca işlenir. İlaveten; DVMA için aktif girdi sinyali, doğrusal olmayan yeni bir uzaya taşımak üzere, geçerli prototip vektörlerinin kerteriz noktaları olarak kullanılacağı GKF çıkışında elde edilir. Bölüm 2.1.2'de anlatıldığı şekliyle DVMA içerisinde yinelemeli olarak ve maksimum marjin amaç fonksiyonu kullanılarak öbek ince ayar güncelleme prosedürü koşulur. Her bir epok sonucunda bire-karşı-hepsi DVM prensibiyle prototip vektörlerinin ve örnek-öbek atamalarının yeni nitelik uzayında güncellenmesinin ardından, yeni atamalar DVMA çıkış katmanında her bir örnek için oluşturulan beklenen atama vektörlerinin, $y^{(i)}$, güncellenmesinde kullanılır. Prototip vektörleri güncellendiği için PKA'nın gizli katmanındaki prototip çıkış vektörleri, H_c, yeniden hesaplanır. Bu da örneklerin GKF çıkışlarını etkilen bir unsurdur. PKA parametreleri daha önceden sonlandırıldığı için güncellenmez; sadece GKF hesaplamalarında yeni uzaya geçiş aşamasında kullanılır. Bu döngü bir döngü sınırına ulaşılıncaya veya hata değeri eşik değerinin altına indirgeninceye kadar tekrarlanır. Melez ağ yapısına ilişkin detaylı akış diyagramı Şekil 2.6'da sunulmuştur.



Şekil 2.6. Açgözlü katman tabanlı kompozit ağ mimarisi akış diyagramı

2.2. Derin Ağ Yapısında Hiyerarşik Görüntü Temsili

Tezin bu bölümünde, imge sınıflandırma için derin ağ mimarisine dayalı bir denetimsiz öznitelik çıkartım algoritmasını tasarlanmıştır. Temelinde çok katmanlı ağ yapısında büyük miktarda etiketsiz veri ile hiyerarşik öznitelik öğrenimi modellenmiştir. Literatürde yer alan benzer çalışmalarda [32, 37, 39, 42] yeni veri temsil uzayı olarak ya son gizli katman çıkışları ya da bütün katman çıkışları birleşik bir vektör üzerinde kullanılmıştır. Ancak, çok katmanlı bir sinir ağında her bir katmanın aynı girdi uzayını diğer öznitelik uzaylarına taşıyan farklı bir bakış açısını yansıttığı öngörülmektedir. Bu nedenle katman çıkışlarını birleştirme prensibinin performans arttırımı için tek başına yeterli olmayacağı düşünülmektedir.



Şekil 2.7. Hiyerarşik GKS uzayında görüntü verisinin temsili

İlaveten, denetimli sınıflandırma aşamasında yalnızca son gizli katman çıkışları ele alındığında önceki katmanlardan gelen tamamlayıcı çabaların da göz ardı edildiği varsayılmaktadır. Varsayımlara karşılık bu tezde yumuşak ağırlıklandırmalı Görsel Kelimeler Sözlüğü (GKS) [8, 39, 88] imge sınıflandırma için önerilmiştir. Yinelemeli olarak çok katmanlı ağın her bir katmanında görsel kelimeleri öğrenmek maksadıyla k-ortalamalar algoritması taklit edilmiştir. Son olarak, mevkisel yerleşim bilgisini sisteme eklemlemek için ikinci gizli katmanda her bir imge birbiriyle örtüşmeyen çeyrek dilimlere bölünerek modelleme bu bölgelerde ayrı ayrı

öğrenilmiştir. Neticede görüntü verisi GKS uzayında piramid yapısı içerisinde temsil edilmiştir. Bahse konu hiyerarşik modelleme Şekil 2.7'de sunulmuştur.

2.2.1. Çalışmanın literatüre özgün katkısı

Literatür çalışmaları dikkatle incelendiği takdirde; denetimsiz öznitelikleri temsil eden ara katman çıkışlarının ya doğrudan K en yakın komşu (kNN) [15], Karar Ağaçları (C4.5 Karar Ağacı gibi) [44], Destek Vektör Makineleri (SVM) [45] gibi başka bir öğrenme modelinde kullanıldığı, ya da denetimli bir ağ yapısının ilklendirilmesinde kullanıldığı görülmüştür [30]. Alternatif olarak ara katman çıkışlarının orjinal giriş sinyali ile normalize edilerek birleştirilmesi sonucunda elde edilen daha büyük boyutlu verinin bir sonraki katmana taşındığı katman tabanlı yöntemler de kullanılmıştır [46].

Her ne kadar veriyi yeni temsil uzaylarında göstermek maksadıyla gizli katman aktivasyon değerlerinin kullanılması ilk bakışta akla yatkın gözükse de; aktivasyon değerleri kullanılan hesaplama fonksiyonları nedeniyle çok kısıtlı bir aralığa sıkıştırılmaktadır (sigmoid fonksiyonu için [0,1]). Aynı zamanda ara katmandaki nöronlar yeni temsil uzayını bölen düzlemlere dik uzanan vektörleri temsil ettiği için, aktivasyon değerleri esasında örnek girdilerin bu düzlemlere olan mesafelerini vermektedir. Girdi sinyali içerisinde yüksek varyans ve ara katmanlarda kısıtlı nöron sayısı bulunduğu farz edildiğinde; görüntü tanıma modeli veriye tatmin edici bir seviyede yakınsayamayacak ve bu durum düşük uydurma ile neticelenecektir.

Diğer taraftan, sistemi karmaşık bir yapıya taşımak için ara katmanlara daha çok nöron eklendiği takdirde etkin örüntülerin tespit edilmesi yerine sadece birbirinin benzeri işlevsiz örüntüler elde edilebilir ve nitekim aşırı uydurma problemi ile karşılaşılabilir. Her iki problem sahasından kaçınmak maksadıyla nöron sayısının dikkatlice seçilmesi, özdeşlik fonksiyonundan uzaklamak için ise giriş örneklerinin veya aktivasyon çıkış değerlerinin rastgele gürültülerle bozulması/pasif hale getirilmesi gerekecektir.

Tezin literatüre katacağı özgün değerlerden birisi olmak üzere; bahsedilen tüm hususları gidermek maksadıyla derin ağ yapısının eğitimi aşamasında görüntü verisinden hiyerarşik bir yapıda görsel kelimeleri (kod kelimeleri) çıkaracak yeni bir

yaklaşım önerilmiştir. Elde edilen kod kelimeleri Görsel Kelimeler Sözlüğü (GKS)'ne dönüştürülmüş ve bir görüntü verisi ara katman çıkış değerleri yerine GKS uzayında tanımlanmıştır. K-ortalamalar (K-means) algoritmasının taklit edildiği bu çok katmanlı ağ mimarisi içerisinde, AEs yapısından farklı olarak, çıkışlarda giriş sinyalinin kendisi yerine görsel kelimeleri temsil eden öbek merkez vektörleri kullanılmıştır. İlaveten, Lazebnik ve diğ. [8] tarafından önerilmiş yöntemin bir benzeri ağ yapısına eklenerek mevkisel bilgi de modelde kullanılmış; imgenin çeyrek bölgelerinden elde edilen GKS vektörleri piramid hiyerarşisi ile birleştrilerek veri temsilinde yeni bir yaklaşım sunulmuştur. Netice itibariyle; verinin temsil edildiği yeni uzayın boyutlarının ara katman nöron sayısından bağımsız kılınması sonucunda aşırı uydurma probleminin çözümü, çıkışlarda öbek merkez vektörleri kullanılarak özdeşlik fonksiyonunu ortadan kaldırma ve sisteme mevki bilgisinin eklenmesi hedeflenmiştir.

2.2.2. Önerilen çok katmanlı ağ mimarisi

Önerilen üç gizli katmanlı sinir ağında fırsatçı katman tabanlı öğrenme yapısıyla her bir katmanda ardışık olarak GKS temsil uzayları oluşturulmaktadır. Bir önceki katmanda çıkartılan daha düşük seviye öznitelik vektörleri ağın sonraki katmanlarına taşınmakta ve buradaki ağırlık parametreleri önceki ağırlıklar sabit tutularak optimize edilmektedir. Esas itibariyle, aynı prosedür katman katman tekrarlanarak GKS temsil uzayının temelini oluşturan görsel kelimeler (kod kelimeler) çıkarılmaktadır.

Çok katmanlı ağ yapısının ilk ara katmanını öğrenmek üzere; verilen eğitim imgelerinden, $X=\{x^{(1)}, x^{(2)}, x^{(3)}, ..., x^{(i)}\}$, $x^{(i)} \in R^{nxlxd}$; n ve l imge ebatları, d renk kanalları, çok fazla miktarda yamalar rastgele seçilerek ilk girdi verisi oluşturulmuştur. Her bir girdi M boyutlu (M = w x w x d) piksel yoğunluk değerlerine sahip vektörler ile temsil edilmiştir. Fakat bu yoğunluk değerleri gürültü ve yama içi korelasyona sahip olduğu için doğrudan kullanılmamıştır. Karşıtlık normalizasyonu için sıfır ortalama – bir standard sapma gürültü azaltımı, korelasyon giderici olarak beyazlatma (whitening) [39] metodu aynı değişintiyi sağlamak üzere uygulanmıştır. Tek katmanlı klasik ağın z miktarda yama ile öğrenilmesi için öngörülen maliyet fonksiyonu şu şekilde tanımlanır;

$$J(W,b) = \frac{1}{z}\sum_{j=1}^{z}\frac{1}{2}\left\|h_{W,b}(p^{(j)}) - y^{(j)}\right\|^2 + \frac{\lambda}{2}\sum\left\|W\right\|^2 \quad (2.13)$$

Burada $h_{W,b}(p)=f(W^T p + b)$, $f : R \rightarrow R$, örnek bir yama girdi örüntüsü, p, için sistemin tahminlemesini üreten hipotez fonksiyonunu, W ve b genlikleri azaltılmaya çalışılan model parametrelerini, y hedef değerleri ve λ dengeleyici katsayıyı temsil etmektedir. Dikkat edilirse, ara katman aktivasyon fonksiyonları, $f(\cdot)$, doğrusal olmadığı için hipotez fonksiyonu da doğrusal ve konveks olmayan bir yapıya dönüşmüş; sistem parametreleri de sıfıra yakın ve rastgele ilklendirilmiştir. Özetlemek gerekirse; aşırı uyumdan kaçınmak maksadıyla ağırlık vektörlerinin yüksek genlikleri cezalandırılırken ortalama karesel hataların toplamı en aza indirgenmiştir. Bu fonksiyon türevlenebilir olduğu için gradyen tabanlı algoritmalar optimum parametreleri elde etmek için kullanılmıştır. Geri yayılım metodu ise hata türevleri ağ boyunca dağıtmak için uygulanmıştır.

Önceki bölümlerde bahsedildiği gibi, AE sinir ağları giriş ve çıkış değerlerini eşitlemek suretiyle, $y^{(j)} = p^{(j)}$, denetimsiz bir yapıda öznitelik çıkarımı için kullanılmaktadır. Ancak, önerilen yöntemde çıkış hedef değerleri görsel kelimelere, $c^{(i)}$, haritalanarak AE yapısından farklı bir yapı tasarlanmıştır. K-ortalamalar algoritması ağ içerisinde taklit edilerek özdeşlik fonksiyonu yerine bir nevi kod kelimeleri öğrenilmiş ve detayları müteakip bölümde anlatılmıştır.

2.2.3. K-ortalamalar algoritmasının ağ yapısında taklidi

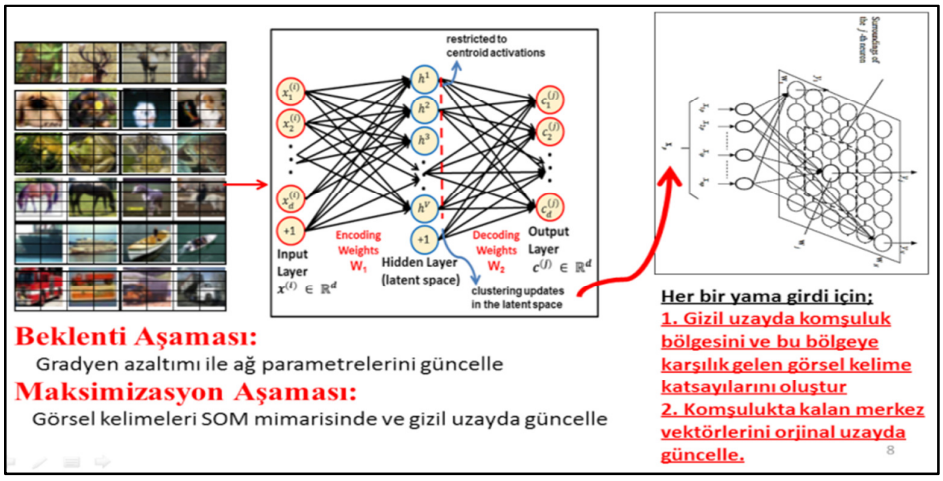
K-ortalamalar algoritması kodlama-kod çözücü ve yeniden yapılandırma açılarından sinir ağ mimarisine oldukça benzerlik gösterdiği için bu tezde k-ortalamalar metodu ile ağ yapısının optimize edilerek görsel kelimeler elde edilebileceği varsayılmıştır. Önceki gizli katman çıkışlarının daha derin katmanlara iletilerek sınıflandırma görevine yönelik GKS temsil vektörlerinin hiyerarşik yapıda ve daha ayırt edici yeni uzaylarda çıkartılabileceği öngörülmüştür. Bu kapsamda öncelikle k adet yama örneği rastgele seçilerek başlangıç merkez vektörleri, $c^{(i)}$, ilklendirilmiştir. Daha sonra Euclid uzayında her bir girdi yaması en yakın merkez vektör ile eşleştirilerek ağ yapısının çıkış hedef değerleri, $y^{(j)}=c^{(i)}$, elde edilmiştir. İleri besleme ve geri yayılım algoritması Denklem (2.13) üzerinde işletilerek sistem parametreleri mini paketler halinde girdi yamaları ile optimize edilmiştir. Burada dikkat edilmesi

gereken husus, tüm eğitim verisinin kullanıldığı bir epok bitene kadar görsel kelime vektörleri, $c^{(i)}$, ile örnek-öbek eşleştirmesinin sabit tutulması ve sadece ağırlık vektörlerinin güncellenmesidir. Her bir epok sonrasında ise bu defa ağırlık vektörleri sabit tutulmuş; hem örnek hem de merkez vektörler ara katman çıkışlarına taşınarak ara katmanda merkez vektörler güncellenmiştir. Beklenti – Maksimize etme (Expectation – Maximization) metodu şu şekilde gerçeklenmektedir;

$$\begin{aligned}
&\text{Beklenti}: (W,b)\,\text{güncelle}, c^{(i)}\,\text{sabit tut}\\
&\{W_1, b_1, W_2, b_2\} = \text{argmin}_{W,b} \frac{1}{2z}\sum_{t=1}^{z}\left\|h_{W,b}(p^{(t)}) - c^{(i)}\right\|^2 + \frac{\lambda}{2}\sum W^2
\end{aligned} \tag{2.14}$$

$$\begin{aligned}
&\text{Maksimize}: c^{(i)}\text{'yi gizil uzayda güncelle}, (W_1, b_1)\,\text{sabit tut}\\
&f_p(p^{(t)}; W_{1,}, b_1) = W_1^T p^{(t)} + b_1 ; f_c(c^{(i)}; W_{1,}, b_1) = W_1^T c^{(i)} + b_1\\
&\forall p^{(t)} \in P; q_i^t \leftarrow \begin{Bmatrix} 1 \ , & \left\|f_p - f_c\right\| = \min_i \left\|f_p - f_{c^{(i)}}\right\| \\ 0 \ , & \text{diger} \end{Bmatrix}\\
&\forall c^{(i)}, i = 1,2,3,...,k; \; c^{(i)} = \frac{\sum_t q_i^t p^{(t)}}{\sum_t q_i^t}
\end{aligned} \tag{2.15}$$

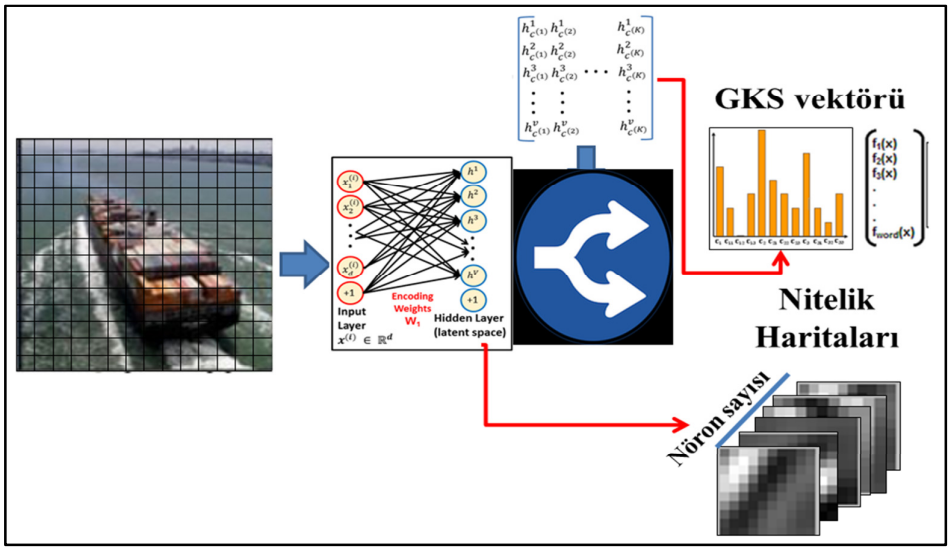
Denklem (2.14) ile sistem parametreleri (W,b)'nin güncellenmesinin ardından, öncelikle hem girdi, $p^{(t)}$, hem de merkez, $c^{(i)}$, vektörleri gizli uzaya taşınmıştır. Burada gizli uzay ile kasıt ara katmandaki nöron aktivasyon çıkış değerleridir. Denklem (2.15)'de ise eşleştirme gizli uzayda gerçeklenmiş ve örnek-öbek eşleştirmeleri en yakınlık prensibine uygun olarak güncellenmiş, öbekler arası mesafe arttırılmıştır. Dikkat edilirse, burada kullanılan ağ yapısı Bölüm 2.1.1'deki Prototip Kodlama Ağı (PKA) ile aynıdır. Merkez vektör kaydırmaları aynı şekilde SOM mimarisinde yürütülmüştür. Ağ parametreleri (ağırlık ve yanlılık vektörleri) ile merkezlerin güncelleme prosedürü yinelemeli olarak öbeklerde artık herhangi bir değişiklik olmayıncaya (hata değeri belirli bir eşik değerinin altına düşmüştür) veya yineleme sınırına ulaşıncaya kadar devam ettirilmiştir. Tek katmanda görsel kelime çıkartımı Şekil 2.8'de sunulmuştur.



Şekil 2.8. Tek katmanlı ağ mimarisinde görsel kelime çıkartımı

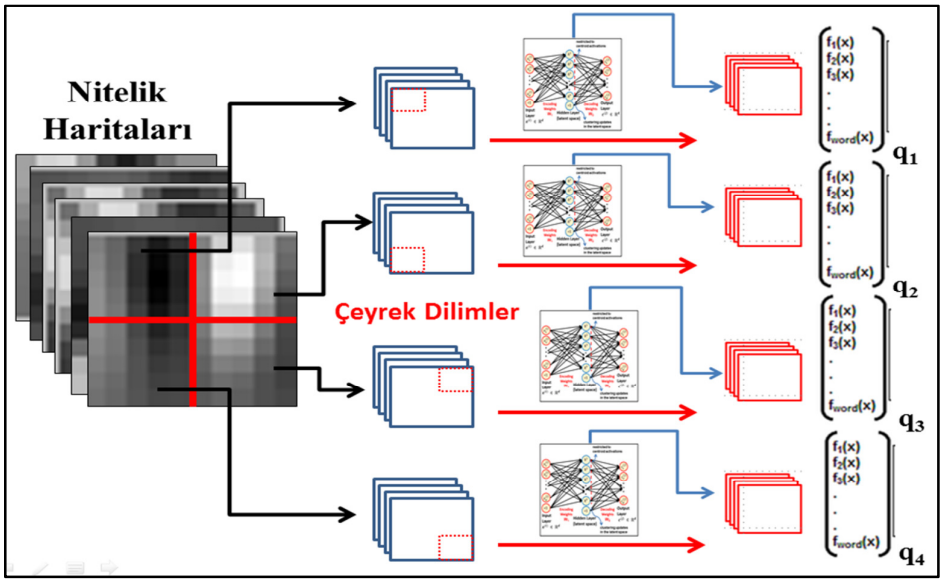
2.2.4. Görüntü verisinin hiyerarşik GKS uzaylarında temsili

Çok katmanlı ağ yapısının ardışık modellenmesinde eldeki mevcut bir katmanın öğrenimi tamamlandığında, artık verilen bir imge girdisi için ayarlanmış merkez vektörleri, $c^{(i)}$, yardımıyla tek bir GKS temsil vektörünün oluşturulması aşamasına geçilmektedir. Bu kapsamda öncelikle bir imgeden w x w ebatlarında ve birer piksel aralıklarla konvolüsyon işlemindeki gibi yamalar çıkartılmıştır. Bu girdiler çok katmanlı ağımızın ilk aşamasının girdi uzayını temsil etmektedir. Bölüm 2.2.1'de detayları verilen görüntü önişleme tekniklerinin ardından, yama girdilerin ilk gizli katman aktivasyon çıkış değerleri elde edilmiştir. Bu değerler ağımızın ikinci katmanı için kullanılacak yeni girdileri oluşturmaktadır. Esasında bu işlem görüntü verisindeki algı alanlarının (receptive fields) lokal örüntülerinin konvolüsyonel bir teknikle ortaya konulması anlamı taşır. Dolayısı ile ara katman çıkışları artık vektörler ile değil, nitelik haritaları ile ifade edilmektedir. İlaveten, GKS temsil vektörlerini doğrusal olmayan bir yapıda elde etmek için Coates ve diğ. [39] tarafından önerilen yumuşak oylama algoritması kullanılmıştır. Böylece bir imgeden alınan bütün yamalar ve önceki tek katmanlı ağın ürünü olan merkez vektörler kullanılarak bir imge için yeni bir temsil uzayı olan GKS vektörü hesaplanmıştır. İlk katmanda GKS çıkarımı Şekil 2.9'da gösterilmiştir.



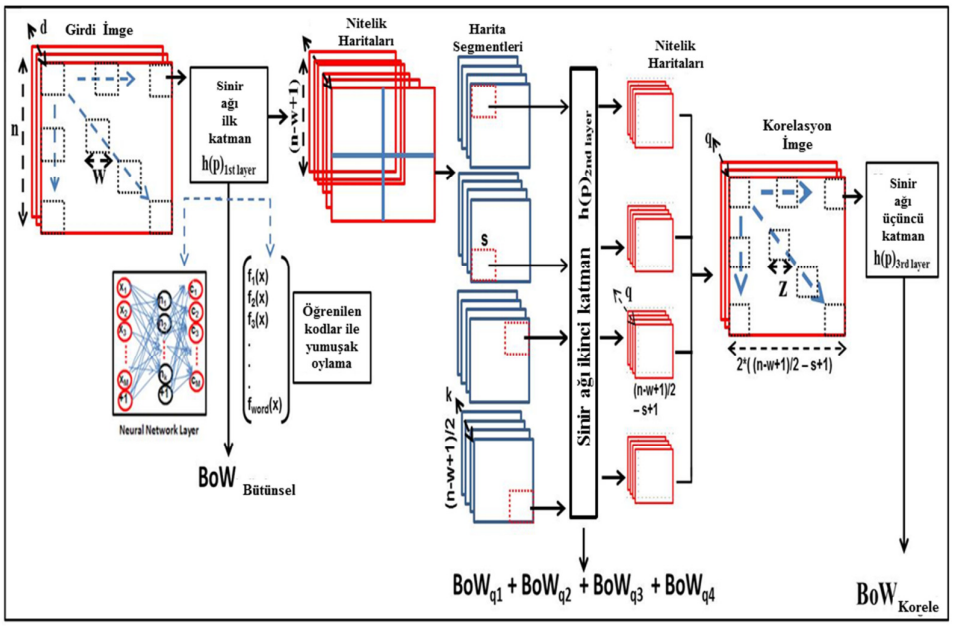
Şekil 2.9. İlk katmandaki GKS vektör çıkarımı

Çok katmanlı ağın ilk gizli katmanında imgelerin yama betimleyicilerden küresel nitelik haritalarının elde edilmesinin ardından, ağın ikinci kısmında yine denetimsiz bir yapıda olmak üzere ancak bu defa lokal nitelikleri oluşturacak şekilde bir ağ mimarisi tasarlanmıştır. Detay olarak; ilk gizli katman çıkışlarındaki nitelik haritaları mevkisel düzlemde birbiriyle örtüşmeyen çeyrek dilimlere bölünmüş ve her bir dilimde ayrı birer sinir ağı öğrenilmiştir. Burada ağın derinlerine ilerledikçe daha soyut nitelik haritalarının çıkarılacağı, bu haritalar yardımıyla birbiriyle ilintili, hiyerarşik ve yeni GKS temsil vektörlerinin elde edilebileceği ve kaba bölütleme ile mevki bilgisinin de sisteme dahil edileceği varsayılmıştır. Esasında ikinci katmanda da ilk katmandaki prosedürler takip edilmiştir. Her bir çeyrek dilimde ayrı olmak üzere; eğitim veri setindeki imgelerin nitelik haritalarından çok sayıda rastgele yama seçilerek girdi sinyali elde edilmiştir. Nihayetinde bu yama haritaları ikinci gizli uzayda o dilime özgü merkez vektörler ile modellenerek yeni GKS vektörleri ve nitelik haritaları tesis edilmiştir. Dikkat edilmesi gereken husus, merkez vektörlerin ilk katmandakinin aksine burada harita yamalarını temsil etmesidir. İkinci aşamadaki mevki bilgisi odaklı ağ mimarisi Şekil 2.10'da sunulmuştur.



Şekil 2.10. Mevki bilgisinin dahil edildiği ikinci katman yapısı

Çok katmanlı ağın üçüncü aşamasına geçilmeden önce ikinci gizli katmandan elde edilen çeyrek dilim nitelik haritaları birleştirilmiş ve bu yapı tümleşik olarak üçüncü aşama girdisi olarak sisteme dahil edilmiştir. Burada temel varsayım çeyrek dilimlerden ayrı ayrı elde edilen nitelik haritalarının korelasyona tabi tutulmasıdır. Bununla birlikte; tümleşik yapıda elde edilecek nitelik haritalarından çıkartılacak yeni GKS temsil vektörlerinin başarı performansını arttıracağı öngörülmüştür. Ayrıca dikkat edilmesi gereken diğer bir husus, nitelik haritalarının konvolüsyon işlemi ile elde edilmesi sonucunda her katmanda ebatların küçülmesi ve haritaların derinliğinin ise önceki ara katmanda bulunan nöron sayısına eşit olmasıdır. Son olarak; görüntü ön işlemede ilk katmanda karşıtlık normalizasyonu ve beyazlatma (whitening) uygulanırken, ikinci ve üçüncü katmanlarda yalnızca beyazlatma uygulanmıştır. Mevki bilgisini ihtiva eden ve görüntü verisinin temsili için hiyerarşik yapıda GKS (Bag-of-Words –BoW) vektörlerinin çıkarımını sağlayan üç gizli katmanlı ağ mimarisi Şekil 2.11'de gösterilmiştir.



Şekil 2.11. Hiyerarşik GKS temsili için önerilen üç gizli katmanlı ağ yapısı

Her üç katmandan elde edilen GKS vektörleri sınıflandırmada yeni öznitelik uzaylarını temsil edecek şekilde ayrı ayrı, çiftler halinde veya birleşik halde kullanılabilir. Bu şekildeki piramid biçimini andıran yapı ilk olarak Lazebnik ve diğ. [8] tarafından önerilmiş ve performans artışına neden olduğu görülmüştür. Üç gizli katmanlı ağ yapısından hiyerarşik biçimde ardışık elde edilen GKS vektörleri sınıf etiketleri ile birlikte verilmiş eğitim imgeleri üzerinde standart ayırtaç fonksiyonlarını oluşturmak için kullanılmıştır. Bu kapsamda deneylerde L_2 normunda Karar Vektörleri Makinesi (KVM) metodu koşulmuştur. KVM'ye ait cezalandırma sabiti (C)'ni ayarlamak için çapraz-doğrulama uygulanmıştır. Deney ortamı ve sonuçlarına yönelik detaylar Bölüm 3'te verilmiştir.

2.3. Denetimsiz Göreceli Nitelik Çıkarımı

Görüntü verisinin analiz edilerek içerik bilgisine göre anlamlı kategorilere atanması son yıllarda ilgi çeken bir çalışma alanı olmuştur. İmge sınıflandırma problemi imgelerin işlenerek anlamlı kategoriler ile etiketlenmeleri olarak tanımlanabilir. İmge modellemesine yönelik oluşturulacak prototiplerin esasını, içeriğindeki nitelikler teşkil etmektedir.

İnsanların zihinlerinde anlamlı bir noktada buluştukları kavramlar şeklinde tanımlanabilecek niteliklerin tespiti sınıflandırma probleminin zorluk derecesine göre

denetimli veya denetimsiz, uygulaması ise kategorik (var-yok şeklinde ikili) veya göreceli olabilmektedir. Niteliklerin öğrenilmesi aşamasında denetimin kalitesi deneklerin bilgi ve tecrübesi ile doğru orantılı, imge sınıflarının sayısı arttıkça ölçeklenebilirliği azalan, emek yoğun bir iştir. Aynı zamanda denetim aşamasında kullanılan niteliklerin sezgisel olması objektif bir denetimi zorlaştıracak, somut olsa bile etkisi düşük olan niteliklerin kullanılması imge sınıflandırma performansını düşürecektir.

Tezin bu kısmında, denetimli nitelik çıkarma işlemindeki problemleri azaltmak amacıyla göreceli niteliklerin denetimsiz bir şekilde bulunmasına yönelik bir yöntem önerilmiştir. Göreceli nitelikler, imge sınıflarının oluşturduğu belirli koşullanmalar ile göreceli (benzerdir veya büyüktür gibi) ve örnek tabanlı mesafe fonksiyonları marifetiyle yinelemeli bir şekilde denetimsiz olarak öğrenilmiştir. Göreceli niteliklerin anlamlandırılması Şekil 2.12'de gösterilmiştir.

Literatürde benzer bir yaklaşım, Ma ve diğ. tarafından [70] yakın zamanda yayımlanmıştır. Ancak bu çalışmada önerilen yöntemde farklı olarak; nitelik öğrenmeye büyüktür koşullanmasına ilave olarak benzerlik koşullanması da eklenmiş ve sınıflar arası göreceli ilişkilerin belirlenmesinde entropi kriteri yerine yeni bir algortima önerilmiştir. Ayrıca sınıflandırma aşamasında, nitelikler belirlenirken nitelik vektörleriyle birlikte ara aşamalarda tespit edilen ağırlık vektörleri de kullanılmıştır.



Şekil 2.12. Sınıf tabanlı göreceli nitelikler ile modelleme

2.3.1. Denetimsiz göreceli nitelik öğrenme ve imge sınıflandırma

Eğitim aşamasında elimizde R^n boyutlu $\{x_i\}$ düşük seviye öznitelik vektörleri ile ifade edilen $I=\{i\}$ imgeler ve bu imgelere ait $\{c_a\}$ sınıf etiketleri bulunduğunu varsayalım. Başlangıç aşamasında nitelikler üzerinden imge sınıflarının birbirleri ile büyüktür veya benzerdir şeklindeki göreceli ilişkileri belirtilmemiş olup bu ilişkiler sıralama fonksiyonları yardımıyla denetimsiz öğrenilmektedir. Göreceli ilişkilerin denetimsiz kurulması maksadıyla nitelik uzayı $A=\{a_m\}$ imge sınıflarının ikili kombinasyonları kadar genişletilmektedir. Bu nedenle N sınıflık bir eğitim veri seti için üretilen nitelik sayısı $N(N-1)/2$ 'dir. Başlangıçta üretilen rastgeke ikili nitelik öğrenimi Şekil 2.13'te sunulmuştur.

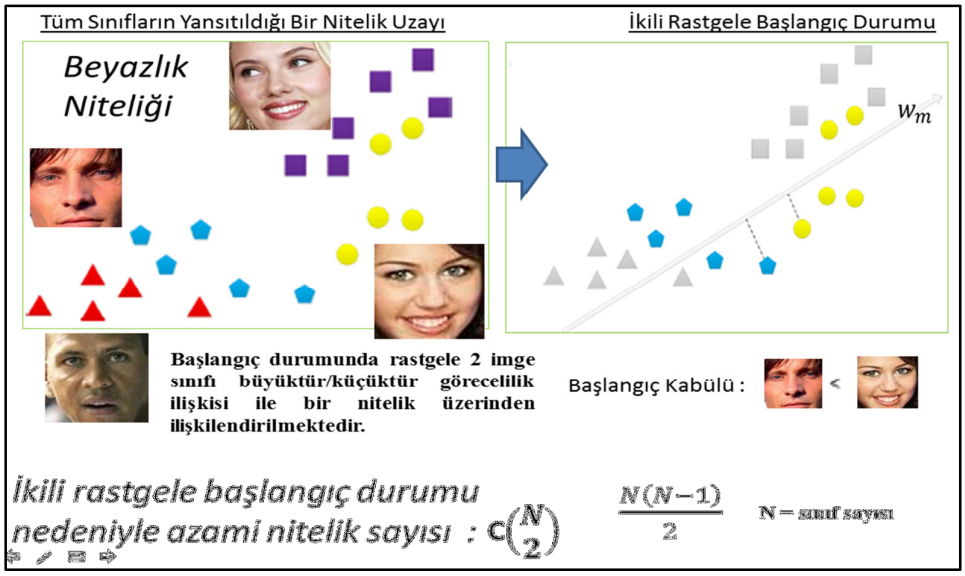
Göreceli bir niteliğin öğrenilmesine ilk etapta iki imge sınıfı arasında büyüktür ($c_a > c_b$) koşullanması kullanılarak başlanmıştır. Bir a_m niteliğinin göreceli öğrenilmesi aşamasında sınıflar arası sıralamayı sağlayan doğrusal skor fonksiyonu ve kısıtlamaları şu şekildedir;

$$r_m(x_i) = w_m^T x_i \tag{2.16}$$

$$\forall (i,j) \in O_m : w_m^T x_i^a > w_m^T x_j^b ; i \in c_a, j \in c_b, c_a > c_b \tag{2.17}$$

$$\forall(i,j)\in S_m : w_m^T x_i^a \approx w_m^T x_j^b ; i\in c_a, j\in c_b, c_a \approx c_b \quad (2.18)$$

Burada $r_m(\cdot)$, a_m niteliğinin kullanıldığı doğrusal skor fonksiyonunu, w_m imge sınıfları arasında göreceli öğrenilen niteliğin ağırlık vektörünü, O_m ve S_m nitelik üzerinden sırasıyla büyüktür ve benzerdir şeklinde göreceli ilişkilerin kurulduğu ikili sınıf setlerini ifade etmektedir.



Şekil 2.13. Rastgele nitelik öğrenmede sınıf tabanlı ikili başlangıç durumu

Nitelik uzayını temsil eden w_m ağırlık vektörleri DVM'ye benzer bir algoritma ile optimize edilmiştir. Ancak [70]'den farklı olarak büyüktür yanında benzerdir koşullanmasının da kullanılması neticesinde elde edilen ikinci derece kayıp fonksiyonunun yer aldığı optimizasyon problemi şu şekilde geliştirilmiştir;

$$\text{argmin}_{w_m}\left(\frac{1}{2}\left\|w_m^T\right\|_{L_2}^2 + C\left(\sum \varepsilon_{ij}^2 + \sum \gamma_{ij}^2\right)\right) \quad (2.19)$$

$$\text{st.} \quad w_m^T(x_i - x_j) \geq 1 - \varepsilon_{ij} \;\; ; \;\; \forall(i,j)\in O_m$$

$$| w_m^T(x_i - x_j) | \leq \gamma_{ij} \;\; ; \;\; \forall(i,j)\in S_m$$

$$\varepsilon_{ij} \geq 0 ; \gamma_{ij} \geq 0$$

Burada C ikili göreceli kısıtlamaları (O_m ve S_m) sağlayarak sıralama marjinini azami yapmayı amaçlayan denge sabitini, ε ve γ ise negatif olmayan küçük değişkenleri ifade etmektedir. İmge sınıflarının nitelik uzayında denetimsiz şekilde

ilişkilendirilmesinde karşımıza iki problem sahası çıkmaktadır: Birincisi, nitelik öğrenmede kalan imge sınıflarından hangisinin sıradaki yinelemede kullanılacağıdır. İkincisi ise seçilen aday sınıfın öğrenmede kullanılmış diğer sınıflar ile nasıl ilişkilendirileceğidir. Henüz kullanılmamış imge sınıflarından hangisinin sıradaki yineleme için en uygunu olduğuna karar verilirken aşağıdaki mesafe fonksiyonu kullanılmıştır;

$$C_{aday} = \text{argmin}_{C_{unused}} \sum_{Cused=1}^{i} (\sum_{k=1}^{j} (P_k - M_{Cused}))^2 \tag{2.20}$$

$$P_k = \{p \mid p = w_m^T x_k^{unused} \;;\; k \in C_{unused}\} \tag{2.21}$$

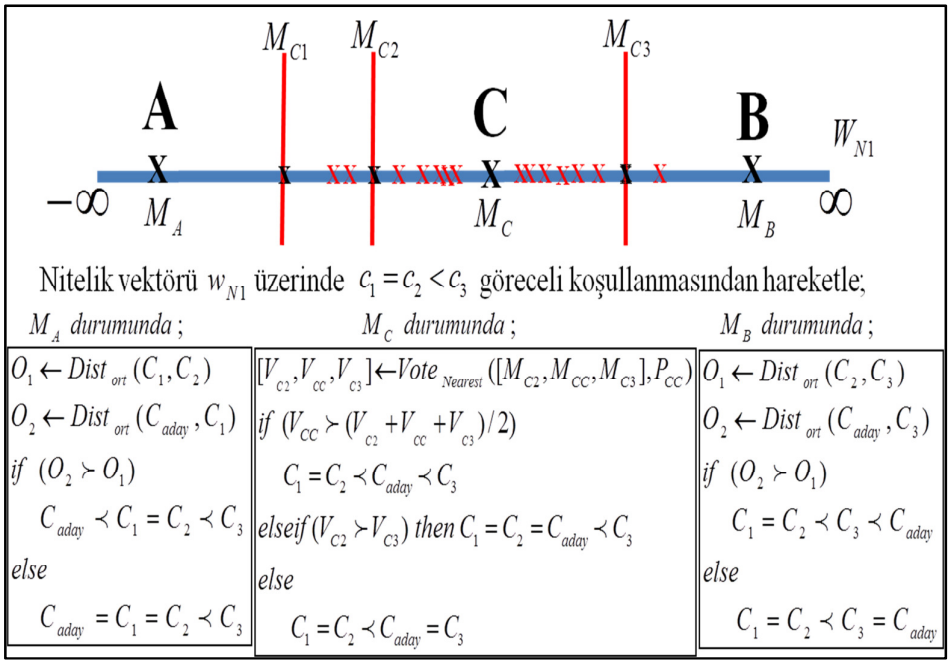
$$M_{C_{used}} = \text{median}(\{p \mid p = w_m^T x_i^{used} \;;\; i \in C_{used}\}) \tag{2.22}$$

Burada P_k kullanılmamış imge sınıflarında yer alan örneklerin göreceli nitelik vektörü üzerindeki skor değerlerini, M_c ise kullanılan imge sınıflarının medyan noktalarını ifade etmektedir. Neticede kullanılan imge sınıflarının orta değerlerine toplamda en yakın örneklere sahip imge sınıfı bir sonraki yinelemede kullanılacaktır. Dikkat edilirse örneklerin medyan noktalara olan mesafelerinin mutlak değerlerinin yanında yönleri de (solda (-) veya sağda (+)) bu fonksiyonda önem kazanmıştır.

Diğer önemli bir husus olarak; gerek aday sınıfın bulunmasında gerekse öğrenilen nitelik üzerinde göreceli ilişkilerin sağlanmasında, kullanılmış sınıflardaki sadece sağlam örnekler işleme sokulmuştur. 'Sağlam' ve 'Çürük' örnekleri tespit etmek için o sınıf ile w_m nitelik vektörü üzerinde göreceli sıralamadaki en yakın diğer sınıflar arasındaki ortalama mesafeler hesaplanmıştır. Görecelilik ilişkisine göre büyüktür koşullanmasında her iki sınıf arasında ortalama mesafeden daha az, benzerdir koşullanmasında ise daha fazla mesafe bulunan örnek çiftleri her iki taraftan da çürük olarak ayrılmış; kalan sağlamlar ile işlemlere devam edilmiştir.

Bulunan aday sınıfın eğitilen nitelik üzerinde diğer kullanılmış sınıflar ile göreceli ilişkilerini belirlemek maksadıyla, [70]'deki entropi hesabından farklı olarak Şekil 2.14'deki algoritma uygulanmıştır. Aday sınıfın bulunması ve bu sınıfın öğrenilen nitelik üzerinde diğer kullanılmış sınıflar ile ilişkisinin belirlenmesi işlemlerine bütün sınıflar kullanılıncaya kadar yinelemeli olarak devam edilmiştir. Her yinelemede kullanılmış sınıflardaki sağlam örnekler o safhada tespit edilen nitelik ara ağırlık

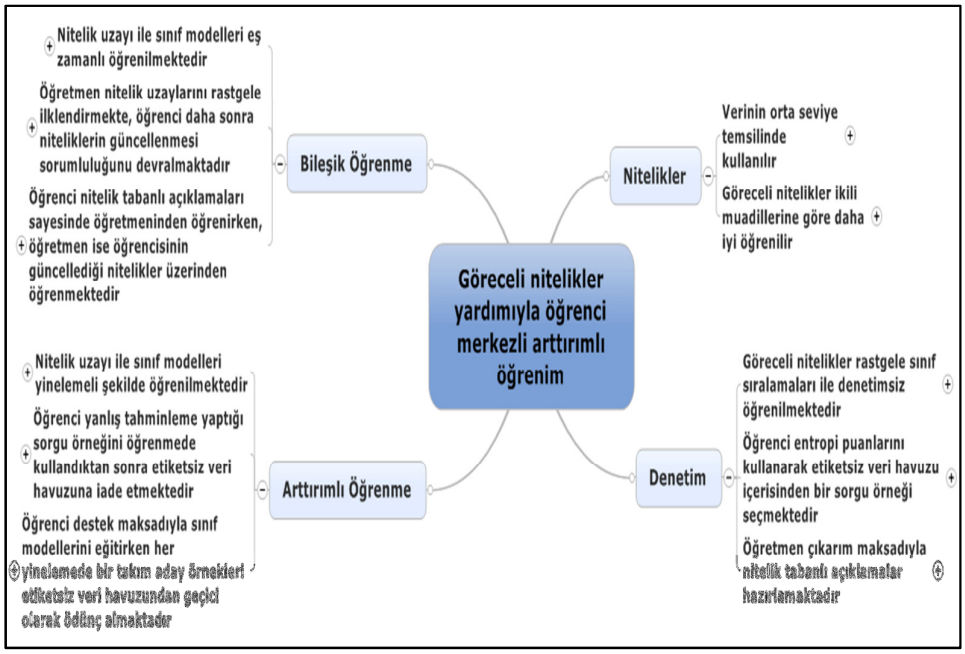
vektörlerine skaler yansıtılmış; çürükler ise bir önceki uzayda aynen bırakılarak eğitime devam edilmiştir.



Şekil 2.14. Sınıflar arası göreceli ilişkilerin öğrenilmesi algoritması

2.4. Göreceli Nitelikler Yardımıyla Görsel Tanımada Artırımlı Öğrenme

Tez çalışmasının son kısmında, esas itibariyle göreceli nitelikler makine-makine etkileşiminde iletişim kanalı olarak kullanılmış; orta seviyede temsil uzayını oluşturan nitelikler ve imge sınıf modelleri artırımlı şekilde eş zamanlı öğrenilmiştir. Gerçek hayattaki öğrenme prosedürüne uygun olarak, bir makine 'Öğrenci' (S) rolünü üstlenirken diğer makine 'Öğretmen' (T) gibi davranacaktır. Başlangıç koşullarında S'in elinde sınıf etiket bilgisi verilmiş sınırlı sayıda eğitim verisi (Sset) olduğu varsayılmış ve Sset üzerinden S hem nitelikleri hem de sınıfları modellemiştir. Önerilen yöntemin temel esasları Şekil 2.15'te sunulmuştur.



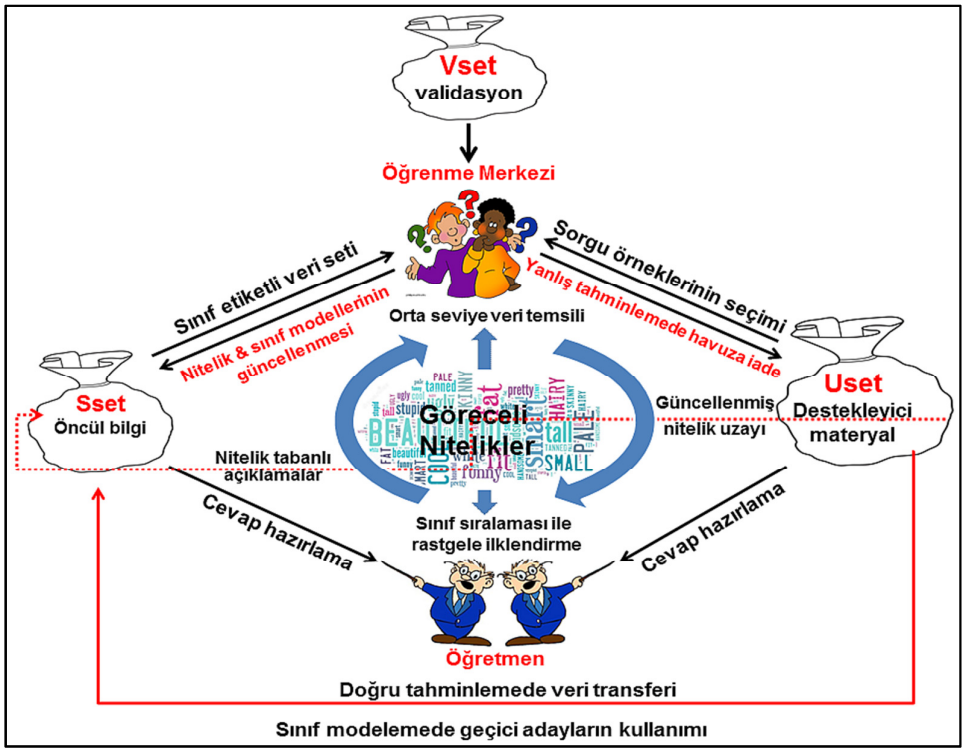
Şekil 2.15. Önerilen yöntemin temel esasları

İlaveten, T'nin elinde ise sınıf etiketlerini sadece kendisinin bildiği başka bir veriseti (Uset) bulunacak; S açısından Uset'in etiketleri görülemeyecektir. Nitelikler sınıflar temelinde göreceli ilişkileri belirleyeceğinden, T nitelikleri sınıfları rastgele sıralamak suretiyle ilklendirmiş ve bu sıralamalar sabit tutulmuştur. Böylece nitelik-sınıf matrisi [77]'ye benzer yapıda denetimsiz bir şekilde oluşturulmuştur. İkili nitelikler yerine göreceli niteliklerin kullanımının temel sebepleri: Temsil uzayının faktöriyel oranında genişletilmesi ve öğrenilmeleri açısından görecelilik kavramının insan doğasına çok daha uygun olmasıdır. Zira bir niteliğin imge üzerinde var olup olmadığını kesin biçimde belirlemek yerine bahse konu imgenin başka bir imge ile o nitelik üzerinden mukayese edilmesi ve 'daha çok/az' veya 'benzer' şekilde öğrenime sokulması nitelik öğrenmede daha iyi bir genelleme sağlamaktadır [14]. Sset üzerinden sınırlı miktardaki örnek ile düşük seviye bütünsel öznitelik vektörleri ve T tarafından ilklendirilen rastgele nitelik-sınıf göreceli sıralama matrisi kullanılarak, S öncelikle nitelik uzaylarını ve nitelik ağırlık vektörleri üzerinden de sınıf modellerini başlangıç koşullarında oluşturmuştur.

Müteakiben, S mevcut sınıf modellerini kullanarak kendisine etiketsiz görünen Uset içerisinden kendi işine en çok yarayan (maksimum entropi) örneği (x_q) seçerek tahmin ettiği sınıf etiketi ($C_{prediction}$) ile birlikte cevaplaması maksadıyla T'ye iletir. T

Uset içerisindeki örneklerin gerçek sınıf etiketlerini bildiği için bu tahmini doğru bir şekilde yanıtlayabilir. Ancak, yarı denetimli öğrenmeden farklı olarak, T sorguyu sadece 'evet' veya 'hayır, x_q aslında $C_{ground\text{-}truth}$ sınıfındandır' şeklinde gerçek sınıf etiketi ile birlikte yanıtlamaz. Bunun yerine T, S-T arasındaki iletişim kanalı olan nitelikler üzerinden bir açıklama cümlesi ile S'i cevaplar. Dikkat edilirse, bu uygulamanın iki ana etkisi olacaktır. Birincisi, her bir iterasyonda göreceli nitelik uzayları S tarafından güncellendiği için T'nin müteakip dönem cevapları da S'in güncellemeleri ile şekillenecek; bir nevi S de T'yi neyi ne kadar öğrendiğini geri beslemiş olacaktır. Yani T'ye nitelik uzaylarını anladığı biçimde öğretecektir. Diğer bir husus ise, S tarafından seçilen örneğin yanlış tahminlenmesi durumudur. S, x_q için yanlış sınıf tahmininde bulunduğunda örneğin gerçek sınıf etiketini $C_{ground\text{-}truth}$ vermeyecek; sadece tahminin neden yanlış olduğunu nitelikler üzerinden mukayeseli açıklayacaktır.

Böylece, seçilen örnek yeniden Uset içerisine atılabilecek ve diğer yinelemelerde öğrenmeden defaten kullanılabilecektir. Eğitim seti sanal olarak arttırılmış olacaktır. S, T'den gelen nitelik tabanlı açıklamalar ışığında öncelikle göreceli nitelik uzaylarını ve daha sonra aynı iterasyon içerisinde sınıf modellerini güncelleyecektir. Dikkat edilirse, T S'nin zihninde yer alan sınıf modellerine doğrudan müdahale etmemekte; ancak nitelikler üzerinden sistemi iyileştirici açıklamalarını yaymaktadır. Böylece öğrenci merkezli bir öğrenme metodu uygulanmaktadır. Sınıf modellerin güncellenmesinde diğer bir husus, Uset'ten ilave aday örneklerin de belirli ağırlıklandırma ile seçilmesi ve modelde iteratif olarak geçici olarak kullanılmasıdır. Böylece sınıf modellerin eğitiminde her yinelemede Uset'ten alınan farklı örnekler değişik ağırlıkları ile aday sınıflarına atanmak suretiyle sınıf model güncellemeleri yapılmaktadır. S-T arasındaki yinelemeli olarak yürütülen nitelik uzayları ile sınıf modellerinin S tarafından eş zamanlı güncellendiği önerilen artırımlı öğrenme algoritması Uset'teki bütün örneklerin S tarafından doğru sınıflar ile etiketlenmesi ile sonlandırılmıştır. Bahse konu kapsamlı öğrenme metodu Şekil 2.16'da gösterilmiştir.



Şekil 2.16. Göreceli niteliklerle öğrenci merkezli öğrenme yöntemi

2.4.1. Çalışmanın literatüre özgün katkısı

Bu tezin nitelikler üzerinden görüntü tanıma literatürüne üç ana katkısı olacağı öngörülmektedir. Birincisi, nitelik-sınıf ilişki matrisinin rastgele oluşturulması ile nitelik öğrenmede ilave iş gücü kaybının önlenmesidir. Göreceli nitelikler temsil uzayını faktöriyel oranında genişlettiği için rastgele seçilebilecek sınıf sıralamaları da çok fazladır ve böylece kelimeler ile ifade edilemeyen ancak öznellikten uzak ve ayırt edici nitelikler öğrenilmektedir. Deneylerde rastgele üretilen niteliklerin denetim sonucu öğrenilen niteliklere göre sayıca üstünlüğünden dolayı performans kapsamında üstünlüğü olduğu görülmüştür.

İkinci önemli husus, yarı denetimli yapıdan farklı olarak öğrenci makine tarafından nitelik uzaylarının ve sınıf modellerinin eş zamanlı öğrenilmesidir. Daha önceki çalışmalar temelde önce niteliklerin öğrenilmesi ve sonrasında nitelikerden bağımsız olarak sınıf modellerinin oluşturulması prensibine dayanmaktadır. Bu noktada, niteliklerin ne kadar iyi seviyede öğrenildiği asıl hedef olan sınıfların öğrenilmesi kapsamında doğrudan önemli değildir. Çok iyi öğrenilen nitelik uzayları sınıfları

ayırt edici özellik taşımayabilir. Oysa nitelikler ile birlikte sınıfların birlikte öğrenildiği önerilen yapıda iki unsur birbirini control etmekte ve desteklemektedir.

Son önemli nokta ise öğrenci-öğretmen makine etkileşimi olarak artırımlı öğrenen yeni bir metodoloji sunulmasıdır. Diğer benzer literature çalışmaları insan-makine etkileşimi içerisinde bir yapı kurmuşlar ve insanın verdiği cevaplara göre yarı denetimli bir öğrenim sunmuşladır [69,78]. Bu yapıda insan karşısındaki makinenin hangi seviyede bilgi sahibi olduğunu yok saymakta ve kendi bilgi ve tecrübesi ile makineyi yönlendirmeye çalışmaktadır. Bu durumda geri beslemeler çoğu zaman öğrenen makineyi arzu edilen öğrenim seviyesine uzun zamanda getirebilmekte ve bu emek yoğun bir iş halini almaktadır. Oysa, önerilen yapıda öğretmen makine cevaplarını tamamen öğrenci makinenin bakış açısına uygun hazırlamakta ve öğrenciyi bilgi seviyesine uygun ve onun anlayacağı şekilde açıklamalarıyla yönlendirmektedir. Böylece, daha az örnek kümesi ve zamanda öğrenmede daha çok mesafe katedilebilmektedir.

2.4.2. Başlangıç ayarları hakkında ön bilgiler

Tezin bu kısmında, artırımlı bir öğrenme prosedürü içerisinde göreceli niteliklerle eş zamanlı olarak göreve yönelik görsel kategorilerin modellenmesi amaçlanmıştır. Öğrenci (S) – Öğretmen (T) makineleri arasındaki çift yönlü etkileşimde iletişim kanalı olarak göreceli nitelikler kullanılmıştır. Bunu gerçekleştirmek üzere, eldeki büyük veri seti, $X = \{x^{(i)},y^{(i)}\}$; $x^{(i)} \in R^d$, $y^{(i)}=\{1,2,\ldots,K\}$, x düşük seviye öznitelik vektörü, y örneğe ait sınıf etiketi ve $i=\{1,2,\ldots,N\}$, öncelikle rastgele olarak ve test setinden ayrı olarak üç alt kümeye ayrılmıştır: Öğrenci seti (Sset), Öğretmen seti (Uset) ve Validasyon seti (Vset).

Sset T tarafından S'ye verilen ve sınıf etiket bilgisine sahip çok az sayıdaki örneğin oluşturduğu set olarak düşünülebilir. S bu alt seti başlangıç sınıf modellerini oluşturmak ve göreceli nitelikleri güncellemek maksadıyla kullanmıştır. Diğer taraftan, S T ile ortak platformda paylaşılan Uset'in etiket bilgisine haiz değildir ve Uset artırımlı öğrenmede bir destek kümesi olarak kullanılmıştır. Beklendiği üzere Uset Sset'ten daha geniş bir kümeyi ifade eder, çünkü Uset T'nin görev alanına yönelik ilave kullanılan ve T'nin sınıf etiketlerini bildiği eğitim materyalini oluşturur. S Uset'i sorgulama esnasında T'ye tahminini yayarken her yinelemede bir

örnek seçiminde ve sınıf modellerini güncelleme esnasında bazı aday örnekleri de ağırlıklı bir şekilde sisteme katmak maksadıyla kullanmıştır.



Şekil 2.17. Öğrenci merkezli öğrenmede başlangıç ayarları

Seçilen adayların gerçek sınıf etiketlerine sahip olmamakla birlikte, bu adaylar seçilen örnek ve T tarafından verilen açıklama cümlesine göre mevcut sınıf modelleri kullanılarak yine tahminleme yöntemiyle en uygun sınıflara geçici olarak atanacaktır. T ise Uset'in içerisindeki örneklerin kategori etiketlerini ve göreceli nitelikleri öğrenmede kullanılan sınıf sıralamalarını bildiği için; S tarafından yayılan tahminin neden doğru ya da yanlış olduğunu belirten nitelik tabanlı cevaplarını hazırlamak maksadıyla Uset'i kullanmıştır. Bu noktada, literatürdeki diğer çalışmalar ve özellikle yarı denetimli öğrenmeden farklı olarak, S makinesinin öğrendiği sınıf modelleri üzerinde T'nin doğrudan bir etkisi bulunmamaktadır. Hatta T S'nin sınıf modellerini nasıl oluşturduğu ile ilgilenmemekte, sadece nitelik uzayı üzerinden öğrenmeye dolaylı müdahale etmektedir. Dikkat edilirse T'nin gelecek dönem cevapları da S'nin nitelik algısı ile şekillenmekte, bir nevi T de S'den nitelikler üzerinden bir şeyler öğrenmektedir. Çünkü S sınıf modelleri ile eş zamanlı olarak göreceli nitelik uzaylarını da güncellemekten sorumlu tutulmuştur. İlaveten, Vset ise S tarafından yinelemeli güncellemeler esnasında sistemin serbest parametrelerini optimize etmek maksadıyla artırımlı öğrenme içerisinde kullanılmıştır. Algoritmanın başlangıç ayarları Şekil 2.17'de sunulmuştur.

Önerilen yöntemdeki bir diğer uygulama detayı S-T makineleri arasındaki iletiş kanalını oluşturan göreceli nitelikler uzayının nasıl elde edildiğidir. Kategoriler üzerinde var olup olmadığı şekildeki ikili kullanımı yerine [65, 66, 68], bu çalışmada daha iyi ayırtaçlığı sağlamak üzere uzayı faktöriyel oranında genişleten göreceli nitelikler [14] kullanılmıştır. Böylelikle nitelik öğrenmede kullanılan sınıflara ait imgelerin düşük seviye öznitelik vektörleri çiftler şeklinde alınarak büyüktür/küçüktür kıstasları ile nitelik ağırlık vektörü üzerinde skaler değerlerde sıralama yaratacak biçimde dönüşüme tabi tutulmaktadır.

Her ne kadar bahse konu sınıf sıralamalarını denetimli yöntemlerle elde etmek [14, 69] akla yatkın görünse de, nitelik sayısını kısıtlayacak şekilde emek-yoğun bir faaliyet olması ve genellemeden uzaklaştıracak öznel değerlendirmeler içermesi nedenleriyle etkin bir yöntem olmadığı değerlendirilmektedir. Bu yüzden literatüre denetimsiz nitelik öğrenme teknikleri [66, 70, 89] sunulmuştur. Bu yöntemlerde özellikle nitelik sayısı görece emeksiz biçimde çok arttırıldığı için başarı performansında da dikkate değer artışlar gözlenmiştir. Bu tezde niteliklerin öğrenilmesinde denetimsiz bir yol izlenmiştir. T Sset'i kullanarak göreceli nitelikleri ranking SVMs [14] yöntemi ile ilklendirmiş, her bir nitelik için sınıf sıralamaları ise rastgele elde edilmiştir [77]. T ilklendirmeden sonra başlangıç nitelik ağırlık vektörlerini, yöntemini ve sınıf sıralamalarını S ile paylaşmakta, bundan sonraki nitelik güncelleme sorumluluğunu S'ye devretmektedir. Rastgele sınıf sıralamaları tüm sınıfların katılımı ile üretilmesine ragmen, bazı sınıfların rastgele kısıtlanarak parçalı sıralamalar ile nitelik öğrenme de mümkündür. Rastgelelikte tek şart ise daha önceden oluşturulan sıralamaların birbirinden bir seviyede (en az 4 pozisyonun farklı olması gibi) farklı olmasıdır.

İlaveten, deneylerde daha detaylı analizler yapabilmek maksadıyla veri setleri ile birlikte verilen denetimli sınıf sıralamaları da bulunan rastgele sınıf sıralamaları ile birlikte kullanılmıştır. Gözlemlediğimiz kadarı ile T nitelik tabanlı açıklamalarında denetimsiz öğrenilen niteliklere daha çok başvurabilmekte, hatta uzunca bir süre (50 iterasyon kadar) denetimli nitelikleri hiç kullanmayabilmektedir. Dolayısı ile denetimli şekilde dikkatli bir biçimde elde edilen bu isimli niteliklerin, düşük kullanım oranları nedeni ile isimsiz rastgele niteliklere göre daha ayırt edici olmadıkları değerlendirilmiştir. Bu gözlem esasında [70]'deki bulguları

desteklemektedir. Bu nedenle deneylerde periyodik olarak (mesela 30-50 iterasyonda bir) bir niteliğin sistemden uzaklaştırılması sonucu azaltılmış nitelik uzayının performansa etkisi de ayrıca incelenmiştir.

Rastgele ilklendirmeden sonra, S nitelik uzayının güncellenmesi görevini devralmakta, ancak rastgele sınıf sıralamaları eğitim boyunca sürekli sabit tutulmaktadır. Nitelik güncellemeleri sonrasında T'den gelen cevaplara göre sınıf model güncelemeleri de eş zamanlı biçimde tetiklenmektedir. T tarafından niteliklerin rastgele ilklendirilmesi ve eğitim setlerinin (Sset, Uset ve Vset) ayrılması sonrası sıra S tarafından Sset kullanılmak suretiyle sınıf modellerinin oluşturulmasına gelmektedir. S sınıf modellerini oluştururken RBF (Radial Basis Funciton -Radyal Tabanlı Fonksiyon) kernel tabanlı bire-karşı-hepsi SVM [90] yöntemini kullanmıştır. Sistem serbest parametrelerini optimize etmek maksadıyla S grid aramasını 10-katlı çapraz validasyon algoritması ile Vset üzerinde yapmaktadır. Özetlemek gerekirse, düşük seviye öznitelik vektörleri (GIST gibi), $x^{(i)}$, öncelikle nitelik uzayına, $A = \{a^{(i)}\} \epsilon R^m$, taşınmış; sonra kategori modelleme göreceli nitelik uzayında gerçekleştirilmiştir. Dikkat edilmesi gereken diğer bir husus ise sınıf modellemede SVM girdi olarak kullanılan kernel sonuçlarının 0 ortalama ve 1 standart değişinti olacak şekilde normalize edilmesidir.

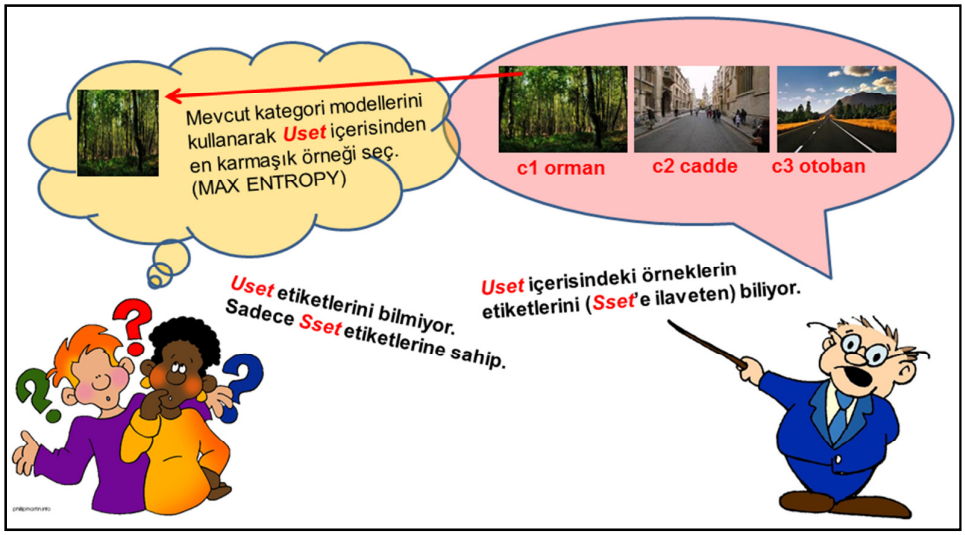
2.4.3. Görsel tanımada öğrenci – öğretmen etkileşimi

Bu çalışmada önerilen artırımlı öğrenme, S-T etkileşimleri çerçevesinde nitelikler uzayı ile sınıf modellerinin dönüşümlü ve eş zamanlı biçimde aşamalı olarak güncellenmesi anlamına gelmektedir. Bu dönüşümlü güncelleme S'nin Uset'teki bütün örnekleri T'nin nitelikler üzerinden vermiş olduğu cevaplar ışığında doğru bir şekilde etiketlenmesine, yani etiketsiz Uset havuzunu boşaltana kadar devam edecektir. Detay olarak her bir yineleme içerisinde öncelikle S Uset içerisinden bir örneği, x^q, tahminleme maksadıyla seçmiş; bu seçimde S'nin işine en çok yarayacak örneğin belirlenmesi işlemi maksimum entropi, H, [69] yöntemine dayandırılmıştır;

$$H = -\sum_{y} p(x^{(i)})_y \log_2(p(x^{(i)})_y) \tag{2.23}$$

Burada $p(x^{(i)})_y$ Uset içerisindeki bir örneğin, $x^{(i)}$, her bir sınıf için, y, SVM çıkışlarının olasılığa dönüştürülmüş sonucudur. Neticede en çok entropiye

(belirsizliğe) sahip örnek seçilerek, mevcut sınıf modellerine göre en yüksek olasılığa sahip kategori etiketi ile birlikte sorgulanmak makadıyla T'ye iletilecektir: 'Bence x^q 1 nulu sınıfa ait, siz ne dersiniz?' Bu husus Şekil 2.18'de sunulmuştur.

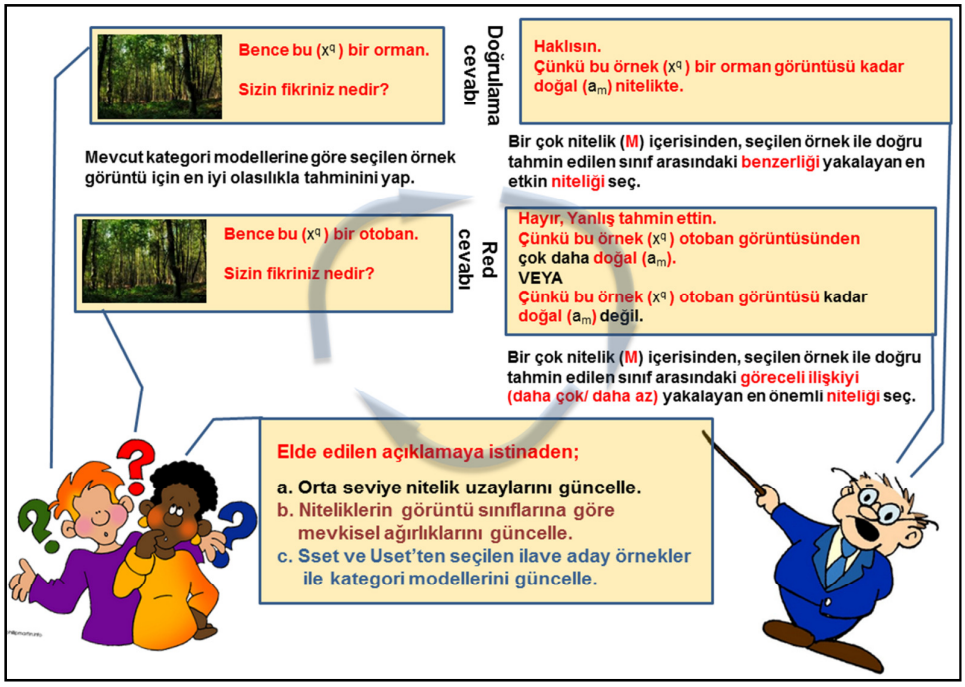


Şekil 2.18. Öğrencinin Uset'ten maksimum entropi ile örnek seçimi

Ayrıca, S en şüpheli yerine mevcut modellerine göre en iyi örneği de arada bir (mesela her 10 iterasyonda bir) seçerek T'ye iletmiştir. Çünkü [69, 78]'den farklı olarak, T sadece yanlış tahmin durumunda değil ayrıca doğru tahmin anında da 'benzerlik' ilişkisi kurarak nitelik tabanlı açıklamalar yapmaktadır. Daha önce bahsedildiği üzere, yanlış tahmin edilen örnek güncellemelerin ardından Uset'e yeniden döndürülmekte ve eğitim seti sanal olarak genişletilmektedir. Bu noktada S'in tahminleme algoritmasını geliştirmek üzere iki yeni imkan daha sunulmuştur. İlk olarak, S Uset'ten seçilen örneklerin sık aralıklarla yeniden seçilmesini önleme maksadıyla tedbir almıştır. Zira aynı örneğin ard arda seçilmesi ve yanlış tahminlenmesi sonucu elde edilecek geri beslemelerden olumlu bir iyileştirme yapılması beklenmemektedir. Bu nedenle Usetten her bir örneğe aynı zamanda bir sayaç atanmış ve yanlış tahmin edilen örneklerin sayaçları üçe seçilerek Uset'e geri döndürülmüştür. Her iterasyonda seçilmeyen diğer örneklerin sayaçları bir azaltılmış ve sayacı 0 olmayan örneklerin seçilmesine müsade edilmemiştir. İkinci olarak, güncel Uset içerisindeki örneklerin seçilme tarihçelerinin kaydı tutulmaktadır. Böylelikle, daha önceden yanlış tahminlerle seçilerek geri döndürülmüş örneklerin

yine yanlış sınıflar ile tahminlenmesinin önüne geçilmiştir. Doğru tahminlerde de T'den açıklamalar geldiği için niteliklerin güncellenmesinin önü açılmıştır.

S tarafından bir sorgu içerisinde tahminin yayınlanmasının ardından sıra T tarafından sorgunun nitelikler üzerinden yanıtlanması sürecine gelmiştir. Bu durum Şekil 2.19'da açıklanmıştır.



Şekil 2.19. Görüntü tanıma sisteminde öğrenci-öğretmen etkileşimi

Nitelikler ortak platformda bulunmakta ve sınıfların görece rastgele sıralanması ile oluşturulmuştur. Bu sıralama bütün öğrenme boyunca sabit kalmakla birlikte; T hem Uset hem de Sset'teki gerçek sınıf etiketlerini bilmektedir. Dolayısı ile T bu şartlarda sorguyu sadece kabul ya da red etmenin ötesinde bir nitelik seçerek en iyi olasılıkla ve sorgu örneği ile tahminlenen sınıfı nitelik üzerinde karşılaştırmalı olarak açıklamaya çalışacaktır. Yarı denetimli öğrenmenin aksine [87, 91], S tamamlayıcı nitelik uzayından etkilenmekte ve sınıf modellerini bu göreceli nitelikler üzerinden şekillendirmektedir. Dolayısı ile sorgu niteliği, x^q, eğer tahmini yanlış ise yine Uset'e geri iade edilmiş, cevap doğru olduğunda ise zaten örneğin gerçek sınıfı ortaya çıktığından Sset'e aktarılmıştır. T'nin nitelik tabanlı açıklamaları şu örnekler ile çeşitlendirilebilir: 'Evet, x^q sınıf 1 kadar doğal (a_m).', veya 'Hayır, x^q sınıf 1'den çok daha doğal', veya 'Hayır, x^q sınıf 1 kadar doğal değil'. Dikkat edilirse, benzerlik

terimi doğru tahminlerde karşılaştırma terimi olarak kullanılırken daha çok/az kıyaslamaları yanlış tahminlerde kullanılmakta sorgu örneği x^q'nun gerçek sınıf etiketi yanlış tahminlerde belirtilmemektedir. Bu kapsamda T'nin sorumluluğu en iyi nitelik-karşılaştırma terimi çiftini açıklama cümlesi oluşturmak maksadıyla seçmektir. Yanlış tahminleme durumundaki en iyi çift seçiminde öncelikle veri setleri Sset ve Uset her bir nitelik eksenine $r_a(x^{(i)}) = (x^{(i)})^T w_a$ doğrusal denklemi ile yansıtılmıştır. Buradaki w_a, nitelik a_m için oluşturulmuş ağırlık vektörüdür. Müteakiben, örnekler puanları r_a ölçütünde tıpkı [78]'deki gibi artan yönde sıralanmış ve şu oy verme işlemi uygulanmıştır;

$$\text{vote}(x^q; a_m, x^{(j)}) = \sum_j \text{indnum}(x^q; a_m) - \text{indnum}(x^{(j)}; a_m) \;;\; x^{(j)} \in C_{\text{predicted}} \tag{2.24}$$

Buradaki indnum(·) bahse konu nitelik uzayında oluşturulan sıralı bir örneğin sıra indis numarasını ifade etmektedir. Her ne kadar artan sıralama modu 'daha çok' kıyaslamasını temsil etse de, 'daha az' koşullanması için elde edilecek oylar 'daha çok' oylarının sadece negatif değerlerini almak suretiyle hesaplanmıştır. Oylama işlemindeki varsayım şudur: Eğer sorgu örneği x^q yanlış tahmin edilen sınıfa, $C_{\text{predicted}}$, ait değil de başka bir sınıfa, C_{true}, ait ise ve a_m niteliği üzerinde $C_{\text{true}} >$ (veya <) $C_{\text{predicted}}$ koşullanması varsa; o halde bu koşullanmanın elde edeceği oy oranının pozitif ve yüksek olması beklenir. Tersine, eğer tahminleme doğru ise oy oranının mutlak değerde olabildiğince minimum olması umulur. Böylece oylama esnasında tahmin hatalı ise T'nin 2M seçeneği (daha çok veya az için M'er adet nitelik seçeneği) eğer tahmin doğru ise M adet seçeneği (çünkü koşullanma sadece 'benzerlik' yönündedir) bulunmaktadır.

Ayrıca, S yanlış tahminde bulunduğu esnada T'nin en iyi açıklamasını yaratacak şekilde iki farklı bakış açısı ele alınmıştır. S bakış açısından, T Denklem (2.24)'de sadece Sset ve x^q 'yu kullanır ve 2M adet olasılık hesaplar. Oyların olasılıklara çevrilmesi işleminde öncelikle oylar minimum 0 noktasına doğrusal biçimde kaydırılır ve toplam oy oranına bölünür. T'nin bakış açısında ise, Uset ve Sset birleştirilerek yeni bir entegre veriseti üzerinde çalışılır. Diğer bir önemli husus, x^q'nun gerçek sınıf etiket bilgisine sahip aynı sınıftaki C_{true} diğer örneklerle yer değiştirmesi ve oyların ortalamasının alınmasıdır. Çünkü T aslında x^q 'nun hangi sınıfa ait olduğunu bilmektedir. Beklenti şu şekildedir: x^q gerçekte C_{true} sınıfını

temsil etmektedir ve bu sınıftaki tüm örneklerin de aynı açıklayıcı cevabı genelleme bakımından desteklemesi beklenir. Böylece her bir örnek $x^{(t)} \in C_{true}$ (Sset+Uset) için oylama ayrı ayrı yapılır ve olasılıklar ortalama oylar üzerinde gerçekleştirilir. En iyi açıklamayı seçmek için ise her iki bakış açısındaki 2M olasılık değerleri birbirleri ile eleman temelinde çarpılmak süretiyle en yüksek olasılıksal nitelik ve koşulu seçilir.

Yani bakış açılarındaki olasılıkların birbirinden bağımsız olduğu düşünülmektedir. Sadece bulunan olasılık değerleri üzerinden açıklama cümlesi oluşturma prensibinden hareket edilirse; T'nin S'yi yanlış yönlendirebileceği düşünülmüştür. Örneğin, 'Hayır, x^q sınıf 1 kadar doğal (a_m) değil' açıklaması eğer gerçek sıralama 'doğallık niteliğinde C_{true} > class 1' yönünde ise ters düşebilir. T nitelikler üzerinde sınıf sıralamalarını x^q'nun gerçek sınıf etiketi ile birlikte bildiği için, açıklama cümlesi eğer gerçeklerle çelişiyorsa bir sonraki en yüksek olasılıklı nitelik-kıyaslama cümlesi ile değiştirilir. Son olarak tahmin doğru olduğunda böyle bir ilave hesaplamaya ihtiyaç bulunmamaktadır, çünkü zaten ele alınan tek kıyaslama terimi benzerlik üzerinedir. Dolayısı ile M adet nitelik oyu üzerinden mutlak değerde en düşük oyu alan nitelik açıklamasına ilave edilir (Evet cevap doğru çünkü x^q sınıf 1'e doğallık (a_m) olarak çok benzer. Ya da x^q sınıf 1 kadar doğal olduğu için bu sınıfa atayabilirsin gibi).

2.4.4. Göreceli nitelikler uzayının yeniden ayarlanması

Temelde, öğrenci-öğretmen etkileşimi bir sorgu-cevap çifti üretmekte ve her iterasyonda bu çift sayesinde dönüşümlü olarak önce nitelik uzayı sonra da sınıf modellerin parametreleri güncellenmektedir. Her ne kadar T göreceli nitelikler uzayını Bölüm 2.4.2'de bahsedildiği gibi sınıfları rastgele sıralamak suretiyle ilklendirse de; S T'den gelen cevaplara uygun şekilde nitelikleri güncelleme görevini devralır. Ancak güncelleme esnasında sınıf sıralamaları değiştirilmez. Böylece T açıklamalarını hazırladığı yeni nitelik uzaylarını S'den öğrenirken S de sınıf modellerini T'den öğrenmiş olur. S aslında sadece Sset'i etiket bilgisi ile birlikte bilmektedir ve nitelikler üzerindeki sınıf sıralamalarını T'den öğrenmiştir. S sorgu imgesini ve T'den gelen koşullu nitelik tabanlı cevaba istinaden seçilen ilave örnek çiftlerini, nitelikleri güncellemek için kullanmıştır.

T'nin S tahminini kabul ettiğini varsayarsak, 'Evet, x^q tıpkı sınıf 1 kadar doğal' gibi, x^q artık Sset'e eklenip Uset'ten çıkartılabilir; çünkü artık x^q 'nun gerçek sınıf etiketi belli olmuştur. Nitekim, bütün nitelik uzayları x^q ve ait olduğu sınıf bilgisi kullanılarak yeni çiftler oluşturmak suretiyle güncellenebilir. T tahmini reddettiği anda durum biraz ilginç hale gelir; zira x^q 'nun gerçek etiket bilgisi cevap esnasında verilmeyecek ve dolayısı ile x^q Sset'e eklenmeyecektir. Ancak S yine de cevaplamada seçilen niteliği güncelleyecek ilave çiftleri kendi algısal çıkarımları ile bulabilir. Örneğin, 'Hayır, x^q sınıf 1'e göre daha doğal' cevaplamasına göre x^q 'nun tahmin edilen sınıfta olmadığı ve bunun nedeninin ise çok daha doğal olduğu bildirilmiştir. Böylece yeni sıralama çiftleri $O^m_{new} = \{(x^q, x^{(j)})\}$, burada $x^q > x^{(j)}$, doğallık niteliğini güncellemek maksadıyla sisteme eklenebilir. Yani esasında x^q sınıf 1 'deki tüm imgelerden , $\forall x^{(j)} \in C_1$, daha doğaldır.

İlaveten, S önceki sınıf sıralama bilgilerine göre daha da fazla çiftler oluşturabilir. Sınıf tabanlı sıralamanın a_m üzerinde C_3, C_2, C_1, C_4 şeklinde olduğunu varsayalım. Dolayısı ile bir önceki örnek açıklama bu sıralama üzerinden doğallık niteliği için verilirse, x^q sorgu örneğinin aslında $\forall x^{(j)} \in C_1 + C_4$, yani iki sınıf için de geçerli olacağı görülecektir. Aynı düşünce daha az koşulu için ancak ters sırada geçerlidir. Böylelikle, tek bir sorguda çok daha fazla eğitim çifti göreceli nitelik güncellemede kullanılmış ve görsel tanımada daha ayırt edici nitelikler elde edilmiştir. Göreceli nitelikler elde edilen ilave çiftler ile Bölüm 2.3.1'de belirtmiş Denklem (2.16–18)'i tanımlayan Newton metodu sayesinde güncellenmiştir. Ayrıca dikkat edilmesi gereken diğer bir husus, yanlış tahminlemenin yapıldığı her iterasyonda yeni eklenen örnek çiftlerin bir sonraki iterasyonda silinmesi gerektiğidir. Çünkü x^q'nun gerçek etiket bilgisi henüz belirsizdir. Bunun yanında, daha önceden belirlenmiş olan nitelik ağırlık vektörleri, $w_{a,}$ iterasyonlar esnasında bir üstteki yinelemeye devredilmiş ve güncellemeler mevcut durum üzerinden yapılmış, yani yeniden rastgele ilklendirme yapılmamıştır. Ancak güncellemeler esnasında bütün örnek çiftleri değil, mevcut duruma göre kısıtlamaları ihlal eden çiftler güncellemede yer almıştır.

2.4.5. Sınıf modellerin güncellemesinde ağırlıklı adayların seçimi

S'nin artırımlı öğrenmede nitelikler uzayını ayarlamasının ardından, diğer bir sorumluluğu ise sınıf modellerini aynı yineleme içerisinde güncellemektir.

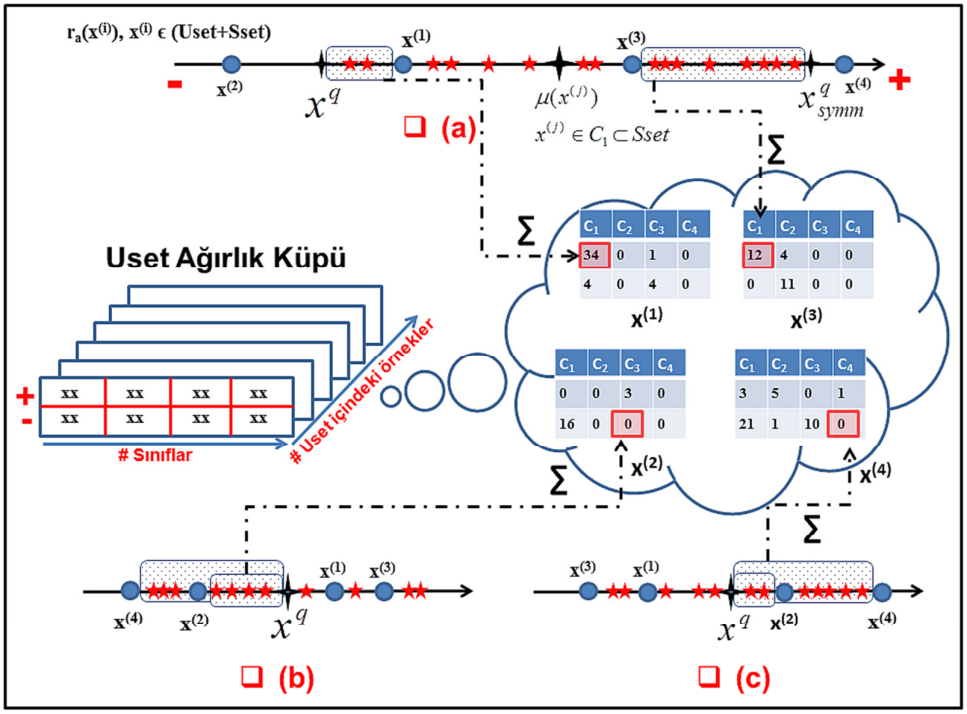
Geleneksel yarı denetimli öğrenme yönteminde, etiketsiz örnek havuzundan mevcut sınıf modellerine göre yüksek güvenilirlikle tahmin edilen en iyi n adet örnek tahmin edildikleri sınıf etiketleri ile birlikte başlangıçta etiketlenmiş kısıtlı sayıdaki örnek kümesine kalıcı olarak atanmaktadır. Müteakip sınıf model güncellemesi ise bu genişletilmiş kümede koşulmaktadır.

Fakat bu yaklaşım önerilen algortmada geçersiz kılınmıştır, çünkü nitelik tabanlı açıklamalar S'ye rehberlik yapmak için kullanılmış, doğrudan sınıf modellemede kullanılmamıştır. Tıpkı nitelik uzayının güncellenmesinde olduğu gibi, S T'nin cevaplarından bir takım çıkarımlar yapmak üzere Uset'ten bir takım aday örnekleri sınıf modelleri eğitmek için geçici olarak seçmiş ve modellemede ağırlandırmak maksadıyla işlem yapmıştır.

Kabul cevabında S'in tahmin ettiği sınıf yönünde, red cevabında ise S'nin tahmin sınıfından farklı bir sınıf üzerinde olmak üzere, Uset'ten bir takım örnekler seçilecek ve bu seçim cevaplamada kullanılan nitelikten bağımsız yapılacaktır. Bununla birlikte, seçilen örnekler sınıf modellerine göre farklı şekilde ağırlıklandırılmıştır. Bunun için S bir ağırlık kübü (UWC -Uset Weighting Cube) oluşturmuş, bu kübün bir bacağında Uset'teki örnekler, diğer ekseninde sınıflar ve son bacağında ise pozitif ve negative taraflar yer almıştır. Dolayısı ile kübün elementleri uygun adaylara göre kümülatif ve iteratif şekilde güncellenmiştir. Ağırlık kübünün oluşturulma esasları Şekil 2.20'de verilmiştir.

Detay olarak, aday örneklerin, $X_{candidates} = \{x^{(p)}\}_{p=1,2,\ldots,U}$, $x^{(p)} \in$ Uset, belirlenmesi maksadıyla T tarafından verilen cevap açıklaması için seçilen nitelik tabanında, a_m, öncelikle bütün örneklerin, $r_a(x^{(i)}) = (x^{(i)})^T w_a$, $x^{(i)} \in$ (Uset+Sset) iz düşürülmesi sağlanır. Müteakiben, örnekler sıralanır ve sıralı listedeki Uset örnekleri x^q ile göreceli mevkilerine göre yine T cevabındaki koşullanmaya uyularak değerlendirilir. Örneğin S x^q sorgu örneğini C_1 sınıfı için doğru tahmin etmişse, açıklamadaki koşul benzerliktir. Dolayısı ile seçilen örneklerin C_1 sınıfında değerlendirilmesi için adayların a_m nitelik uzayında C_1 sınıfına x^q 'dan daha benzer olmaları beklenir. Bu örnek için öncelikle Sset'te yer alan C_1 sınıfındaki orta (medyan) imge tespit edilir ($\mu(x^{(j)})$, bakınız Şekil 2.20/a.). Sonra bu imge merkez olmak üzere ve sınırları x^q tarafından belirlenecek şekilde simetrik bir ilgi sahası çizilir. Uset içerisinde yer alan,

daha önceden yapılan sorgularda T tarafından C_1 sınıfı için red edilmeyen ve bu bölgeye sıralamada iz düşümü gerçekleşen örnekler mevcut yinelemede C_1 sınıfı için pozitif aday olarak gösterilir.



Şekil 2.20. Ağırlıklandırma şeması içerisinde aday örneklerin Uset'ten seçilmesi. $\{x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}\} \in$ Uset T'den gelen açıklamaya göre seçilmekte ve ağırlıklandırılmaktadır: a) x^q sınıf 1'e a_m niteliği üzerinde çok benzemekte, b) x^q sınıf 3 kadar a_m değil c) x^q sınıf 4'ten daha a_m

İlaveten, yanlış tahminler kapsamında seçilen sınıf için negatif adaylar da benzer bir yaklaşımla belirlenebilir. Ancak bu noktada koşullanmalar daha çok veya daha az olduğu için referans nokta olan x^q'ya göre açasıya bölgeler işaretlenir. Daha az koşullanmasında x^q 'nun solunda kalan bölge, daha çok koşullanmasında ise x^q'nun sağında kalan bölge artık yeni ilgi alanıdır ve orta nokta belirlemeye ihtiyaç kalmamıştır. Daha az koşullanması için (x^q nitelik a_m üzerinde C_1 sınıfına göre daha az gibi); Uset'te yer alan örneklerden x^q'dan da az olanların C_1 sınıfından olmaması gerektiği aşikardır. Benzer durum daha çok koşullanması için ama aksi yönde gerçekleştirilir (Bakınız Şekil 2.20/b ve c).

Tahminlenen sınıfa, $C_{predicted}$, göre pozitif veya negative yönde seçilmiş aday Uset örneklerinin, $x^{(p)}$, belirlenmesinin ardından S bu örneklerin kümülatif ağırlıklarını şu şekilde güncellemiştir;

$$W_{UB}(x^{(p)}; a_m, x^q) = \sum_{iter=1}^{UB} |\, indnum(x^q; a_m) - indnum(x^{(p)}; a_m) \,| \,(1 - e^{\frac{-iter}{\sigma}}) \qquad (2.25)$$

Burada UB şu ana kadar ki iterasyon sayısını (kümülatiflik) ve σ ise ölçek faktörünü ifade eder. Formüldeki toplama işlemi her bir iterasyonda T tarafından seçilmiş bulunan nitelikten, a_m, bağımsız olmak üzere kümülatif ağırlıklandırmayı ifade eder. Dolayısı ile ağırlıklandırma işlemi sıralı liste içerisinde esasen T tarafından kullanılan koşullanma yönünde (daha çok için sağ, daha az için sol ve benzerlik için merkez imge yönünde) x^q ile seçilmiş aday arasındaki imge sayısıdır. Ayrıca dikkat edilirse, zaman içerisinde iterasyonlardaki ağırlık birikimleri artan bir oranda etki etmektedir. Çünkü iterasyonlar ilerledikçe nitelikler uzayının daha stabil hale geldiği ve sınıf modelleri ile uyumlu olduğu, dolayısı ile ilerleyen yinelemelerde seçilen adayların önceki iterasyonlara göre daha sağlıklı olacağı öngörülmüştür.

Bölüm 2.4.2'den hatırlanacağı üzere, S Uset içerisindeki mevcut örneklerin daha önceki seçim kayıtlarını tutmaktadır. Yani daha önceden S tarafından seçilmiş ancak yanlış tahminleme nedeni ile tekrar Uset içerisine geri atılmış örneklerin kesinlikle hangi sınıflardan olmayacaklarına ait bilgi de S'in elinde ayrıca mevcuttur. Kesinlikle $C_{predicted}$ sınıfından olmadığı bilinen bu Uset örneklerinin negatif yöndeki ağırlıkları ise her iterasyonda maksimize edilmektedir, çünkü S bunların yanlış $C_{predicted}$ sınıf(lar)ından olmadığından emindir. Neticede, S Uset içerisinden bazı örnekleri aday olarak belirlemiş, T'den gelen cevaplara istinaden bunları kümülatif olarak ağırlıklandırmış ve bir sonraki bölümde açıklanacağı üzere sınıf modellerinin güncellemesinde geçici olarak kullanmıştır. Her iterasyonda seçilen adaylar ve ağırlıkları değişmiş, bir sonraki iterasyona aktarılmamıştır.

2.4.6. Sınıf modellerin güncellenmesi

Şu ana kadar, S nitelikler uzayını güncellemiş ve Sset'i sadece o iterasyona özel biçimde geçici şekilde genişletmek adına; Uset'ten bazı örnekleri ilave olarak belirlemiş ve ağırlıklarını kümülatif şekilde güncellemiştir. Sıra yine S tarafından sınıf modellerinin nitelikler uzayında güncellenmesine gelmiştir ve bu işlem bire-

karşı-hepsi L_2 normunda RBF kernel tabanlı SVM aracılığı ile gerçeklenmiştir [90]. Yarı denetimli öğrenmeden farklı olarak, Sset'e geçici eklenmiş Uset aday örnekleri T'nin nitelik tabanlı açıklamaları ile seçilmiştir. Bu aday örnekler ise Sset'teki etiket bilgileri S tarafından kesin bilinen örneklerden ayrıdır. Çünkü S hala aday örneklerin gerçek sınıflarını bilmemekte, sadece T'den gelen cevaplara ve mevcut nitelikler uzayına göre sadece tahmin edebilmektedir. Bu nedenle, aday örnekler yinelemeli şekilde ve kümülatif olarak ağırlıklandırılmıştır. Ağırlıklar esasen seçilen nitelik eksenindeki sıralı listede x^q ile aralarındaki imge adedidir. Sset'teki örneklerin sınıf etiketleri S tarafından bilindiği için SVM ağırlıkları tıpkı [78]'deki gibi 1'dir. Bu nedenle, Aday örneklerin ağırlıkları sisteme katılmadan önce her sınıf ve pozitif/negatif yönde bağımsız şekilde [0,1] arasına normalize edilir. Normalizasyon esnasında doğru tahminlemede, aday ağırlık toplamları 1, yanlış tahminde ise maksimum ağırlık 1 olacak şekilde önlem alınır. Zira yanlış tahminde ilgili sınıf için kesin yanlış olduğu bilinen Uset örnekleri aday liste içerisine katıldığı için, bunların negatif ağırlıkları da doğal olarak 1 alınmıştır.

Temelde, bu noktada cevaplanması gereken soru çok sınıflı imge tanıma çerçevesinde her bir aday $x^{(p)}$ imgesi için iterasyonlarda geçici olarak hangi sınıf etiketi verilmelidir. Bu sorunun cevabı yine T'nin nitelik tabanlı açıklamasında gizlenmiştir. Hatırlanacağı üzere aday örnekler x^q ile $C_{predicted}$ arasındaki koşullu ilişkiyi yansıtacak şekilde seçilmiştir. Dolayısı ile aday örneklerin de sınıf seçiminde sorgu örneği ile aynı kaderi paylaşması düşünülebilir. S tarafından x^q doğru bir şekilde tahminlendiğinde aday örnekler de normalize edilmiş kümülatif ağırlıkları oranında sisteme $C_{predicted}$ sınıfı ile etiketlenerek sokulur.

Oysa T'nin S tahminini red etmesi durumunda ortaya karmaşık bir durum çıkar; çünkü bu durumda bilinen tek şey tıpkı x^q gibi şeçilen aday imgelerin de $C_{predicted}$ sınıfından olmayacağıdır. Dolayısı ile SVM sınıflandırmada tam olarak aday sınıfların hangi sınıflar ile etiketlenmesi gerektiği doğrudan verilmemiş, sadece hangi sınıftan olmadıkları tahminlenmiştir. Bu yüzden aday örneklere sınıf atamaları yapılırken a_m üzerindeki bilinen sınıf sıralamaları kullanılmıştır. Örneğin, T 'Hayır, çünkü x^q sınıf 3'e atanmak için yeteri kadar a_m değil' şeklinde açıklama yapmış ve o anda sınıf sıralaması ise a_m üzerinde '$C_2 > C_3 > C_4 > C_1$' şeklinde olsun. Bu durumda

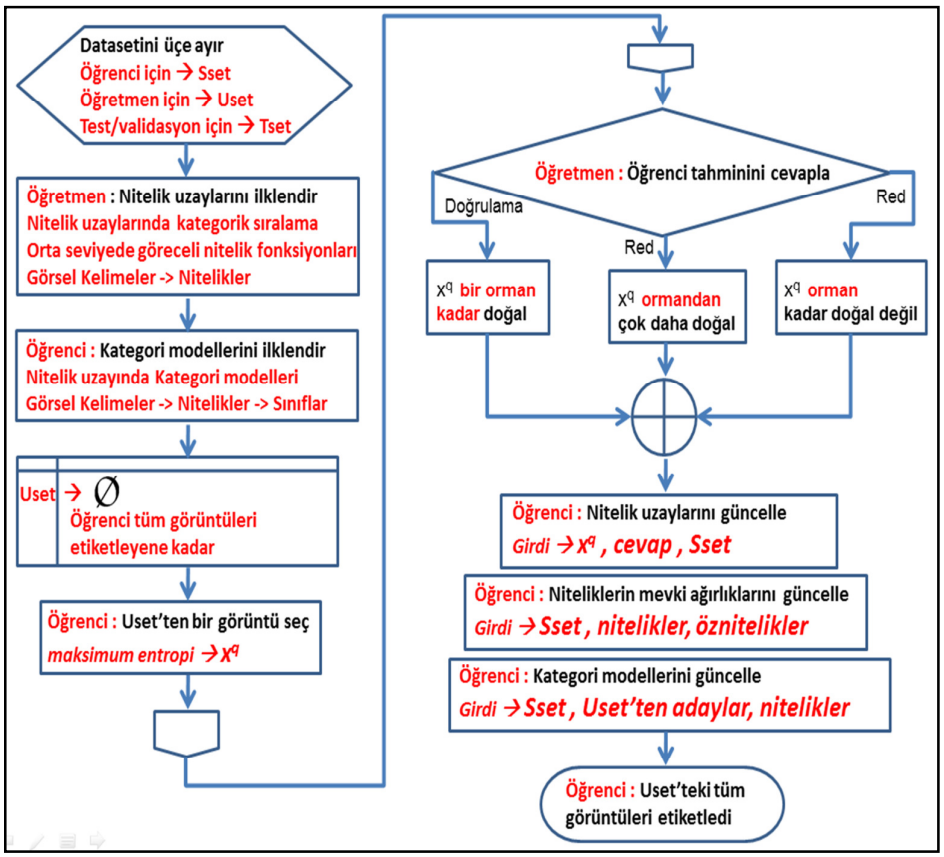
S tıpkı x^q gibi aday tüm örneklerin C_4 veya C_1'den olabileceği çıkarımında bulunabilir.

Karar sınıfını belirlemek tek seçenek olduğunda problem çözümü elbette çok kolaydır. Mesela C_3 ile C_4 sıralamada yer değiştirdiğinde tüm aday örnekler doğrudan C_1 sınıfı ile etiketlenerek kümülatif ağırlıkları doğrultusunda (Uset ağırlık küpündeki $C_{predicted}$ negatif ağırlığı) sisteme eklenebilir. Ancak birden çok sınıf olasılığı varsa S aday örnekler $x^{(p)}$ için en iyi tahminini yapmak zorundadır. Şöyle ki; her bir aday için, eğer olası sınıflarda herhangi bir ağırlıklandırma önceden yapılmadıysa mevcut sınıf modellerini kullanarak en iyi olanı seçer. Aksi halde, pozitif ve negetif yönlerdeki ağırlıklarının farkına göre en çok oy alan sınıf aday örnek için seçilir. Bu durumda sınıf ağırlıkları ise negatif-$C_{predicted}$ ağırlığının aday sınıf sayısına bölünmesi ile tespit edilmiştir.

S'in aday örnekleri Uset içerisinden belirleyip uygun şekilde ağırlıklandırması ve sınıf etiketleri ile etiketlemesinin ardından, bu adaylar Sset'e geçici olarak eklenir ve oluşturulan yeni eğitim seti SVM algoritmasında koşularak sınıf modelleri güncellenir. Dikkat edilmesi gereken bir diğer husus ise tıpkı nitelikler uzayının güncellenmesinde olduğu gibi, sınıf model parametreleri de bir önceki iterasyonda kaldığı noktadan devam ettirilmiş ve böylece artırımlı öğrenme hedefine ulaşılmıştır. Sistem serbest parametreleri (kernel ve SVM) 10-katlı çapraz validasyon yöntemi ve grid arama metodu ile Vset üzerinde optimize edilmiştir. Ayrıca, Vset üzerindeki validasyonlarda bir önceki başarı performansının altında bir değer alındığı takdirde system parametreleri güncellenmemiş, bir önceki eski haline döndürülerek müteakip iterasyona geçilmiştir. Burada sezgi şudur: Bazen T'nin açıklamaları kendi içerisinde tutarlı olsa da S tarafından anlaşılmayabilir veya farklı yorumlanabilir. Böyle bir durumda S'nin öğrenme eğrisi istenen çizgiden uzaklaşabilir. Dolayısı ile eskisinden daha kötü bir seviyede devam etmektense arada bir S seviyesi hakkında uyarılmalı ve gerekirse T'nin açıklamalarına güncellemeler ile cevap vermemelidir. Yani her açıklamada sistem güncelleme ısrarında olunmamalıdır. Böylelikle artırımlı öğrenmedeki dengesizliklerden kaçınılması hedeflenmiştir.

Sonuç olarak; S sınıf modellerini bir adım önce ayarladığı nitelikler uzayında güncellemiştir. Bu güncellemede Sset'e ilaveten T'nin açıklama cevaplamalarından

yola çıkılarak elde edilen aday örnekler kümülatif ağırlıkları ile birlikte kullanılmıştır. Bahse konu dönüşümlü güncelleme metodu S'nin Uset'i boşaltması, yani Uset'teki her örneği T'den gelen cevaplar doğrultusunda nihayetinde doğru sınıflar ile etiketlesine kadar devam ettirilmiştir. Deneylerde görülmüştür ki, sanal eğitim seti artırımı sayesinde yarı denetimli öğrenmeye oranla iterasyon sayısı % 70 oranında artmakta ve başarı yüzdesi ise dikkate değer seviyelere yükseltilmektedir. Göreceli niteliklerle öğrenci merkezli arttırımlı öğrenme algoritmasının akış diyagramı Şekil 2.21'de verilmiştir.



Şekil 2.21. Önerilen görüntü tanıma algoritmasının akış diyagramı

3. DENEYSEL TASARIM VE BULGULAR

Tez çalışması kapsamında Bölüm 1'de denetimsiz düşük seviye öznitelik ve orta seviye nitelik çıkartımı ile görüntü tanıma probleminin çözümünde kullanılan modellere yönelik genel literatür taraması yapılmıştır. Bölüm 2'de ise görüntü verisinin temsili ve görüntünün tanınması bakımından tez çalışmasının hedefine uygun olarak dört farklı, ancak birbirini tamamlayıcı algoritma önerilmiş ve detayları açıklanmıştır.

Tez çalışmasının bu bölümünde, detayları Bölüm 2'de verilen tez ürünlerinin deneysel tasarımları anlatılmış; literatürde yer alan benzer çalışmalarla karşılaştırmalı olarak analiz sonuçları ortaya çıkarılmış ve önemli bulgular tartışılmıştır.

Bu kapsamda; öncelikle Bölüm 3.1'de öbekleme probleminin çözümünü optimize etmek maksadıyla birbirinden farklı amaç fonksiyonları kullanan, birbirini tamamlayıcı özelliklerde ve rakabetçi öbekleme metodu olarak tanımlanabilecek çok katmanlı yapay sinir ağı mimarisi analiz edilmiştir. Prototip Kodlama Ağı (PKA)'nı kullanarak üç ara katmanlı sinir ağı mimarisinde hiyerarşik Görsel Kelimeler Sözlüğü (GKS)'nü oluşturan ve mevki bilgisini de dahil etmek suretiyle piramid şeklinde bulunan GKS vektörlerini görüntü tanımada kullanan ağ yapısının analizleri Bölüm 3.2'de verilmiştir.

Bölüm 3.3'te denetimsiz göreceli nitelik çıkarımının deneysel tasarımı ve bulguları ele alınmştır. Düşük seviye veri temsilinden farklı olarak; orta seviyede görüntünün göreceli nitelikler yoluyla modellendiği bu bölümde özellikle göreceli niteliklerin ikili niteliklere göre üstün olup olmadıkları tartışılmıştır. Müteakiben nitelik öğrenmede denetimsizliğin denetimli yapılara göre performansı analiz edilmiştir.

Son olarak Bölüm 3.4'te öğrenci (S) – öğretmen (T) etkileşimini taklit eden ve öğrenci merkezli görsel tanımada göreceli nitelikler yoluyla arttırımlı öğrenmenin etkileri gerek görsel gerekse analitik yöntemlerle ortaya konmuştur.

3.1. Melez Sinir Ağları Mimarisinde Öbekleme Deneysel Tasarımı

Bölüm 2.1'de detayları verilmiş ve bu tezin ilk ürünleri olarak önerilen öbekleme amaçlı sinir ağı mimarileri, PKA ve DVMA (SVM ağı), literatürde sıkça kullanılan benzer çalışmalardan SCL, LVQ, SOM ve NG ile iki farklı tipte performans karşılaştırmasına tabi tutulmuştur: Görsel ve Analitik. Bu maksatlarla 9 farklı veri seti kullanılmış ve verisetlerinin özeti Tablo 3.1'de sunulmuştur. Dikkat edilirse, bu verisetleri değişik miktarda örnek, öbek ve vektör boyutlarında seçilmiş; seçim esnasında internet ortamındaki birbirinden farklı canlı veri depolarından (UCI machine learning, Speech and Image Processing) yararlanılmıştır. Böylece deney ortamında bu kadar değişken veri girişleriyle algoritmaların mukayeseli olarak nasıl davrandıkları incelenebilmiştir. Esas itibariyle, daha basit seviyedeki iki boyutlu sentetik veri setleri görsel analizlerde kullanılırken, öbeklemede popular olan UCI Machine Learning deposundan alınan gerçek veri setleri ise sayısal performans analizlerinde kullanılmıştır.

Tablo 3.1. Deney tasarımında kullanılan veri setlerinin özet tablosu

	Sentetik Verisetleri					**UCI Verisetleri**			
	S_1	**S_2**	**S_3**	**S_4**	**Aggre-gation set**	**Wine_ red**	**Wine_ white**	**letter recognition**	**clave vectors**
Örnek Sayısı	5000				788	1599	4898	20000	10800
Öbek Sayısı	15				7	6	7	26	4
Vektör Boyutu	2				2	11	11	16	16
Referans Kaynak	[92, 93]				[92, 94]	[95, 96]		[98]	[97]

Ayrıca, genel bir uygulama olarak öbekleme öncesinde ham verinin gürültü ve korelasyondan arındırılmasını sağlamak üzere ön işleme tabi tutulması dikkate değerdir. Bu çalışmada, ön işleme olarak beyazlatma metodu (whiteninig) [39] uygulanmış; nitelik vektör elemanlarının varyansları eşitlenmiş ve nitelik değerleri [0,1] aralığına çekilerek veri normalizasyonu icra edilmiştir.

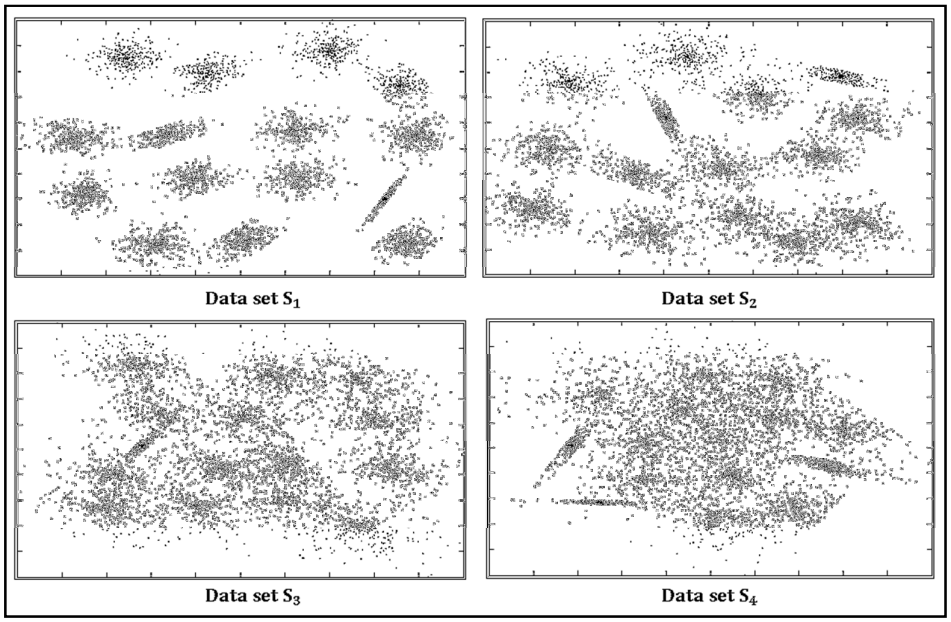
Önerilen ağ mimarileri ile karşılaştırılacak olan diğer öbekleme sinir ağları orjinal makalelerine [48, 52, 53, 55] bağlı kalınarak ve Bölüm 1.7'de yer alan denklemler

kullanılarak MATLAB [99] ortamında kodlanmıştır. Bütün algoritmalar farklı sayıda serbest parametreye sahip oldukları için, öbeklemenin haricinde tutulan bir validasyon seti üzerinde bütün metodların serbest parametreleri belirli aralıklarda grid arama yöntemi ile optimize edilmiş ve öbekleme işlemleri bu optimum değer ile yürütülmüştür. Denetimli öbeklemenin icra edildiği LVQ ve DVMA algoritmaları hariç olmak üzere; diğer tüm öbekleme yöntemlerinde rastgele ilklendirilen aynı prototip vektörleri ile öbekleme başlangıcı yapılmıştır. Bu arada, LVQ ağ mimari başlangıcında SCL prototip ve atama sonuçları, DVMA ağ mimarisinde ise önerildiği şekilde PKA sonuçları başlangıçta kullanılarak kompozit bir yapı oluşturulmuştur.

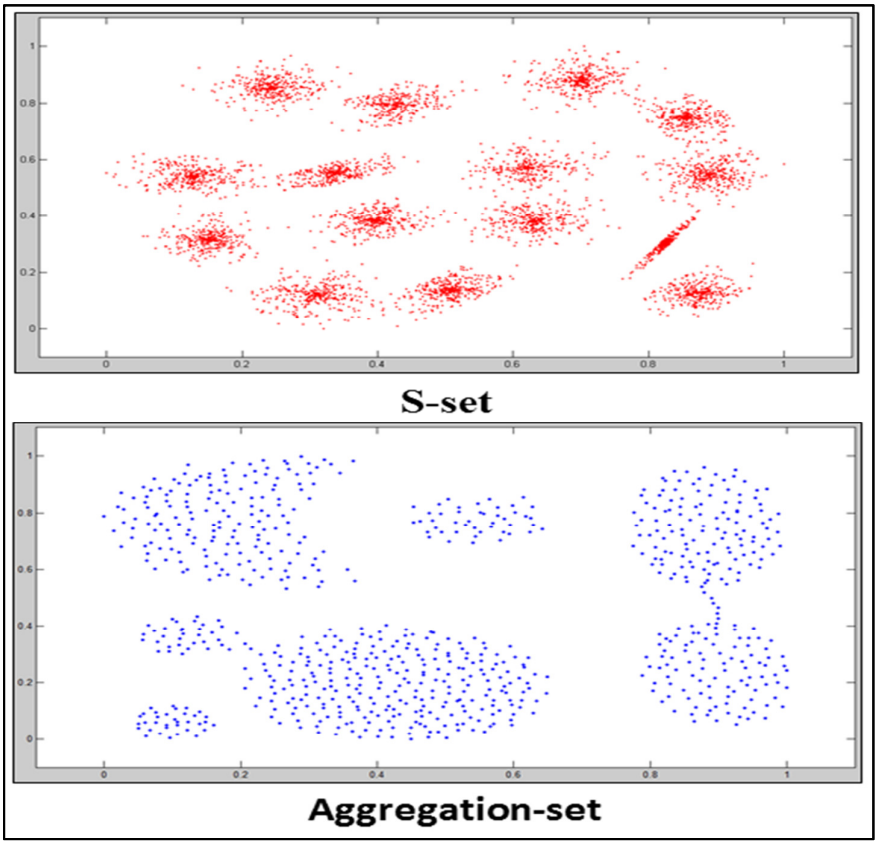
Öbekleme deneyleri aynı veri setlerinde ve koşullarda 50 kez tekrarlanmıştır. Görsel sonuçlar ile dengeli bir karşılaştırma yapabilmek maksadıyla analitik sonuçlarda tüm veri setleri için ortalama ve standart sapma değerleri de ayrıca hesaplanmıştır. Görsel sonuçların mukayesesi bakımından, her bir öbekleme ağ metodu için en iyi sonuçlar kayıt altına alınarak görsel ve analitik düzlemde öbekleme sonuçları karşılaştırılmıştır.

3.1.1. Görsel öbekleme analiz sonuçları

Görsel deneylerde kullanılan sentetik veri setleri, S_1-S_4, ve Aggregation set orjinal öznitelik uzaylarında sırasıyla Şekil 3.1 ve 3.2'de plotlanmıştır. Öbekleme karakteristikleri olan şekil, büyüklük, görece birbirlerine yakınlık ve dağılım ölçütleri iki boyutlu mevkisel dağılımda değişkenlik arz ettiğinden her birinin kendine has zorluk derecesi mevcuttur. Sentetik veri setleri incelendiğinde; S_1 ve S_2 veri setlerinin diğer veri setlerine oranla daha bütünsel yapıda olduğu ve öbeklerin görsel uzayda kolayca ayırt edilebildiği görülmektedir. Bu kapsamda daha iyi görsel çıkarımlar elde edebilmek adına, öbekleme algoritmalarının görsel sonuçları sadece S_1 veri seti üzerinde verilmiştir. Ayrıca, diğer veri setleri üzerinde yapılan ve burada detayları verilmeyen görsel sonuçların, S_1 veri seti üzerindeki sonuçlar ile benzerlik gösterdiği ve bu görsel sonuçların sonraki alt kısım olan analitik analizlerden elde edilen sayısal sonuçlar ile de uyumlu olduğu ayrıca görülmüştür. Diğer taraftan Aggregation-set ise görsel uzayda kolayca ayırt edilebildiği için sentetik veri setlerine oranla daha bütünleşik ve basit bir yapıdadır.

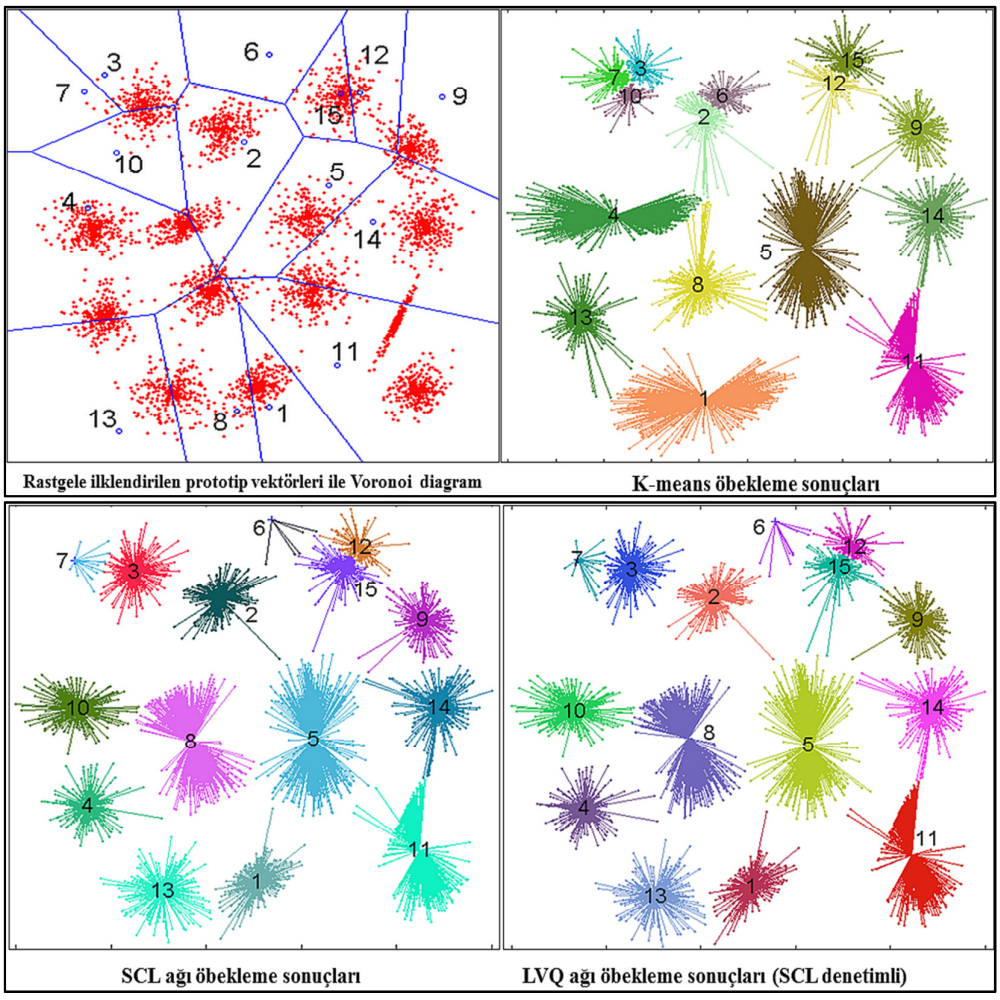


Şekil 3.1. Değişken karmaşıklığa sahip iki boyutlu sentetik veri setleri



Şekil 3.2. Görsel analizlerde kullanılan S_1 ve aggregation veri setleri

Daha önce de belirtildiği üzere, bütün öbekleme algoritmalarında rastgele ilklendirilen aynı prototip vektörleri başlangıç parametresi olarak metodlara tanıtılmış ve hepsi yinelemeli yapılar oldukları için sonlandırma kriteri 100 döngü olarak belirlenmiştir. Karşılaştırmada kullanılan öbekleme algoritmaları Kazanan Hepsini alır (Winner-Takes-All (WTA)) Tabanlı ve Komşuluk Fonksiyon Tabanlı (Neighborhood Function Based (NFB)) şeklinde iki gruba ayrılmıştır. Bu nedenle algoritmaların görsel sonuçları bu kategorilendirme esas alınmak suretiyle sırası ile Şekil 3.3 ve Şekil 3.4'te verilmiştir.



Şekil 3.3. WTA ağ mimarilerinin S_1 seti üzerinde öbekleme sonuçları

Şekil 3.3'te öncelikle başlangıç koşullarının durumunu açık biçimde analiz edebilmek için rastgele seçilmiş prototip vektörleri ile veri setlerinin Voronoi teselasyon diyagramı üzerindeki görselliği yansıtılmıştır. K-means (K-ortalamalar)

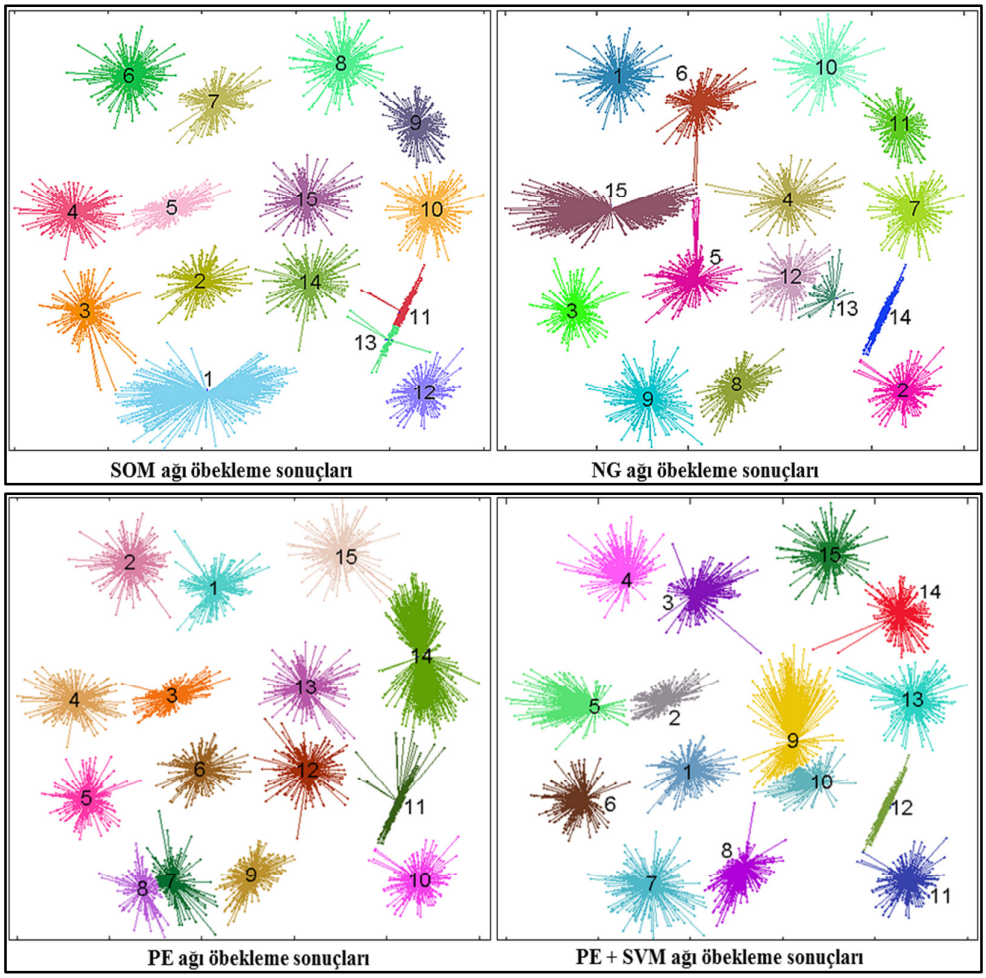
algoritması da WTA katerorisinde popüler bir öbekleme metodu olduğu için görsel sonuçları ayrıca verilmiştir.

Genel olarak, WTA ağ mimarilerinin NFB öbekleme metodlarına göre görsel açıdan çok daha kötü sonuçlar ürettiği görülmüştür. Bunun nedeni WTA metodlarında her bir öbek için yalnızca bir prototip vektörün (metrik sistemde örneğe en yakın/kazanan) güncellenmesi olarak izah edilebilir. WTA ağları başlangıç prototip vektörlerine yüksek derecede bağlı oldukları için yinelemeler müddetince bu prototip vektörlerin göreceli mevkilerini, yani topolojik ilişkilerini (mevkisel konfigürasyon) muhafaza ettikleri görülmüştür. WTA metodları NFB algoritmalarına kıyasla nihai öbek sonuçların çok daha hızlı ancak daha büyük yeniden yapılandırma hata oranları ile ulaşmıştır. Çünkü veri-öbek atamalarının az sayıda yineleme miktarından sonra pek fazla değişmediği gözlenmektedir.

Diğer taraftan SCL ve LVQ ağları klasik k-means metoduna göre daha iyi öbekleme performansı sergilemiştir. K-means algoritmasındaki eşit ağırlıklı basit ortalama alma yöntemi yerine, belirli bir öğrenme oranı ile örneklerin öbek güncellemeye atandığı prototip vektörüne olan mesafesi ölçüsünde katkı sağlamasının performansı arttırdığı değerlendirilmiştir. LVQ ağı denetimli öbekleme yöntemi olduğundan SCL tarafından üretilen prototip vektörleri ve atamalar başlangıç denetim bilgisi olarak LVQ'e verilmiştir. Ancak SCL ile kıyaslandığında LVQ'in dikkate değer bir performans artışı sergilemediği tespit edilmiştir. Bunun temel nedenlerinin LVQ ağ döngüsü esnasında SCL'den denetim maksadıyla temin edilen örnek-öbek atamalarının, $y^{(i)}$, sürekli sabit tutulması ve böylece LVQ'in bir taraftan SCL'i simule etmeye zorlanması olduğu düşünülmektedir.

NFB ağ mimarilerinin Şekil 3.4'te görüleceği üzere WTA metodları ile karşılaştırıldığında daha anlamlı görsel sonuçlar ürettiği görülmüştür. Bunun temel nedeni ise kullandıkları komşuluk fonksiyonlarıdır. Genel olarak her bir örnek, prototip vektörleri üzerinde komşuluk dereceleri oranında güncelleme etkisi yaratmıştır. Bu yaklaşım örnek uzayının daha etkin bir biçimde kullanılmasını sağlamış ve başlangıç prototiplerine olan bağımlılığı büyük oranda azaltmıştır. Ayrıca; hepsinde öbek güncelleme prosedürü içerisinde grandyen azaltım algoritması

koşulduğu için prototiplerin görece mevkisel konfigürasyonunda değişiklikler gözlenmiştir.

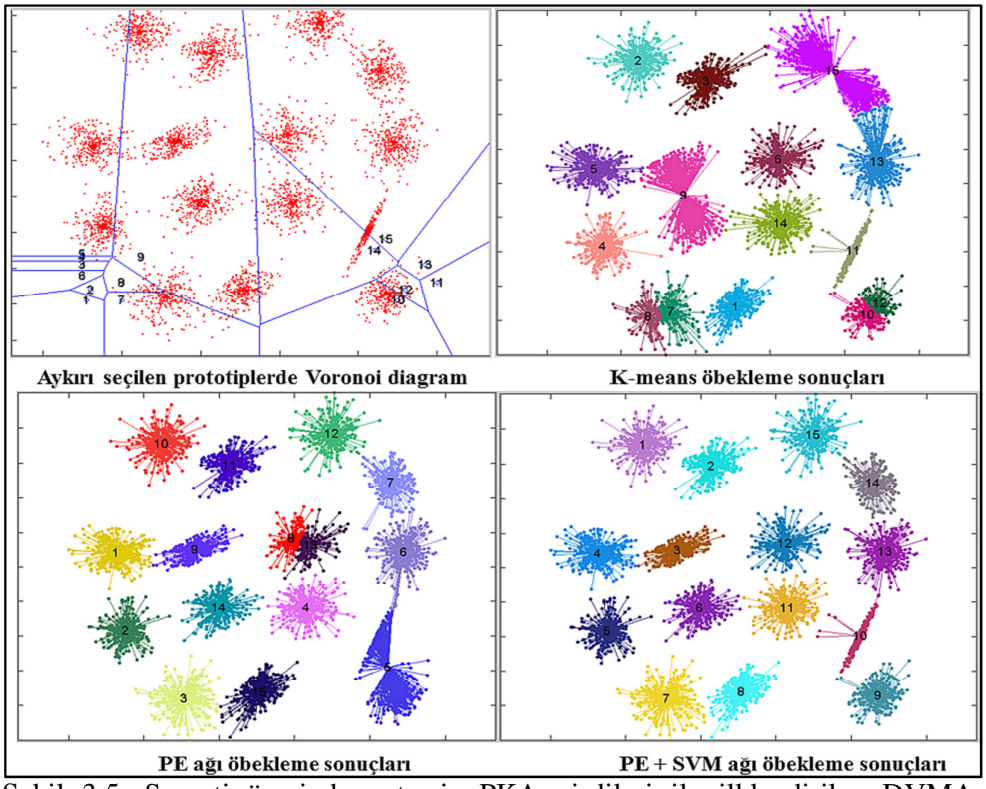


Şekil 3.4. NFB ağ mimarilerinin S_1 seti üzerinde öbekleme sonuçları

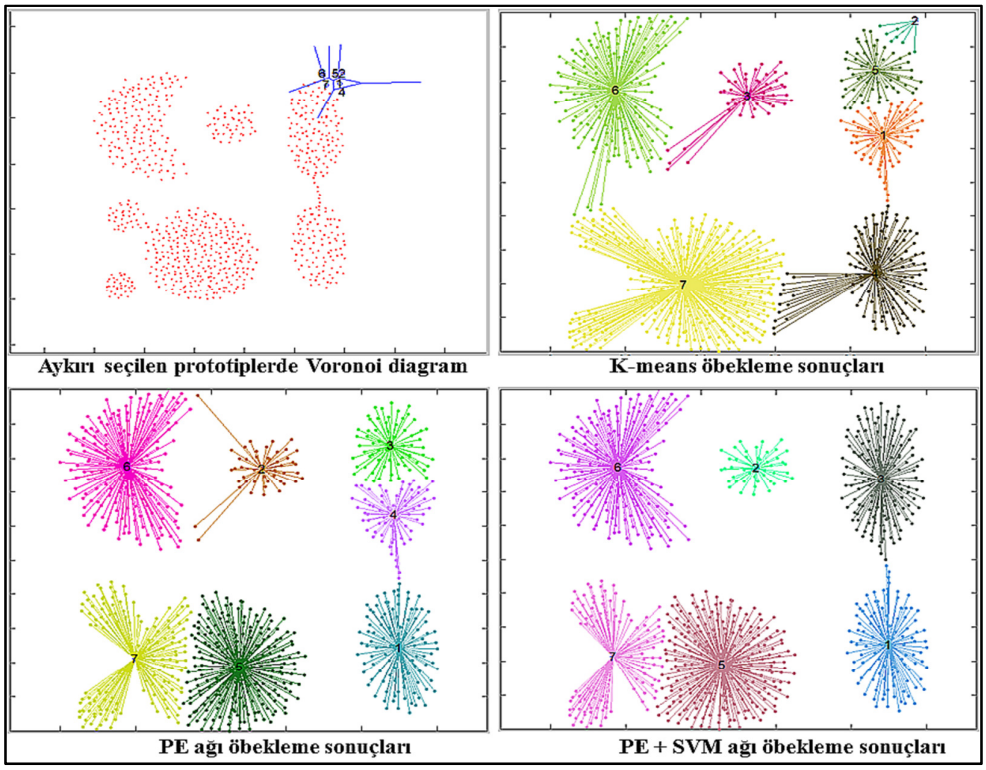
SOM, NG ve ilk ağ olarak önerdiğimiz PKA (Prototype Encoding – PE) mimarisinin görsel açıdan birbirleriyle kıyaslanabilecek benzer sonuçlar ürettiği tespit edilmiştir. Ancak PKA'dan almış olduğu başlangıç denetim bilgisi ile PKA öbeklerini maksimum marjin amaç fonksiyonu içerisinde ince ayar güncellemeye devam eden DVMA mimarisinin (Şekil 3.4'deki PE + SVM) öbekleme performansını kayda değer bir şekilde arttırdığı görülmüştür. VQL'in aksine, DVMA mimarisi yinelemeli olarak yeni gizli nitelik uzayında öbekleri güncellemeye devam etmiş ve başlangıç örnek-öbek atamalarını da dinamik bir şekilde değiştirmiştir. Dikkat edilirse, önerilen PKA ve DVMA mimarilerinin her ikisinin de ağırlık parametrelerini öbek güncelleme işlemi esnasında oluşturulan ve kendilerine ait yeni nitelik uzaylarında

icra ettikleri görülecektir. Bu nedenle, çok daha fazla etiketsiz verinin bulunduğu geniş ölçekli veri setleri ile çalışıldığı takdirde daha az bayaslanmış gizli örüntülerin elde edileceği ve öbekleme performansının da artacağı varsayılmaktadır.

Görsel analizde müteakip işlem olarak, DVMA ağına ardışık olarak beslenen PKA öbekleme sonuçlarının tatmin edici seviyelerde olmadığı varsayılmıştır. Böylece yetersiz bir öbekleme denetimi ile ilklendirilen bir DVMA yapısının bu durumu ne ölçüde telafi edebileceği deneysel olarak araştırılmıştır. Deney ortamını sağlamak üzere, öncelikle PKA yapısında giriş katmanına giren veriye Gauss gürültüsü eklenmiştir. İkinci olarak; yine PKA yapısında öbek merkezleri öznitelik uzayının aykırı bölgelerinde olmak üzere rastgele ilklendirilmiştir. Böylece PKA ağının kasıtlı bozulmuş bu ilklendirme koşullarında vasat bir performans sergileyeceği öngörülmüştür. Bahse konu yetersiz öbeklemelere DVMA yapısının tepkisinin ne şekilde olduğu S_1 ve Aggregation setler üzerinde Şekil 3.5 ve 3.6'da sunulmuştur.



Şekil 3.5. S_1 seti üzerinde yetersiz PKA girdileri ile ilklendirilen DVMA öbekleme sonuçları



Şekil 3.6. Aggregation seti üzerinde yetersiz PKA girdileri ile ilklendirilen DVMA öbekleme sonuçları

Şekil 3.5'te S_1 veri seti üzerinde öbek merkezleri kasıtlı olarak iki farklı uçta (sağ ve sol alt köşelerde) ve birbirlerine son derece yakın bir şekilde ilklendirilmiştir. Beklendiği üzere PKA mimarisi k-ortalamalar algoritmasına göre çok daha iyi bir performans sergilemiş ve çoğu öbeği tespit edebilmiştir. Diğer taraftan DVMA (PE+SVM) melez yapısı her ne kadar büyük ölçüde PKA yapısı ile aynı öbekleme konfigürasyonunu oluştursa da; daha iyi bir performans elde etmiştir. Öyle ki; PKA'dan faklı şekilde 10 ve 12 nolu öbekler (sağ alt çerçevede) DVMA mimarisinde başarıyla ayırt edilebilmiştir.

Benzer deneysel tasarım Aggregation-set üzerinde Şekil 3.6'da görüleceği üzere bu defa bütün başlangıç öbek merkezleri sağ üst köşede ve örnek uzayının dışında konuşlandırılmıştır. Şaşırtıcı bir şekilde; PKA yapısında 3 ve 4 nolu merkez olarak tespit edilen iki öbek DVMA melez yapısında 3 nolu öbek üzerinde tek grupta birleştirilmiştir. Dikkati çeken diğer bir husus ise DVMA tarafından tespit edilen bu öbeğin gerçek durum (ground truth) ile örtüşmesi ve yedi olan orjinal öbek sayısının otomatik biçimde altıya düşürülmesidir. Nihayetinde; DVMA melez ağının PKA

yapısındaki öbekleme sonuçlarına sıkı bir şekilde bağlı olmadığı, bu duyarsızlık nedeniyle PKA öbekleme konfigurasyonunu zaman zaman değiştirebileceği sonucuna varılmıştır. İlaveten, yinelemeli öbek güncelleme prosedürü gereğince DVMA mimarisi daha iyi öbekleme sonuçları elde edebilmek adına ilklendirmeden gelen bazı öbekleri feda edebilmektedir.

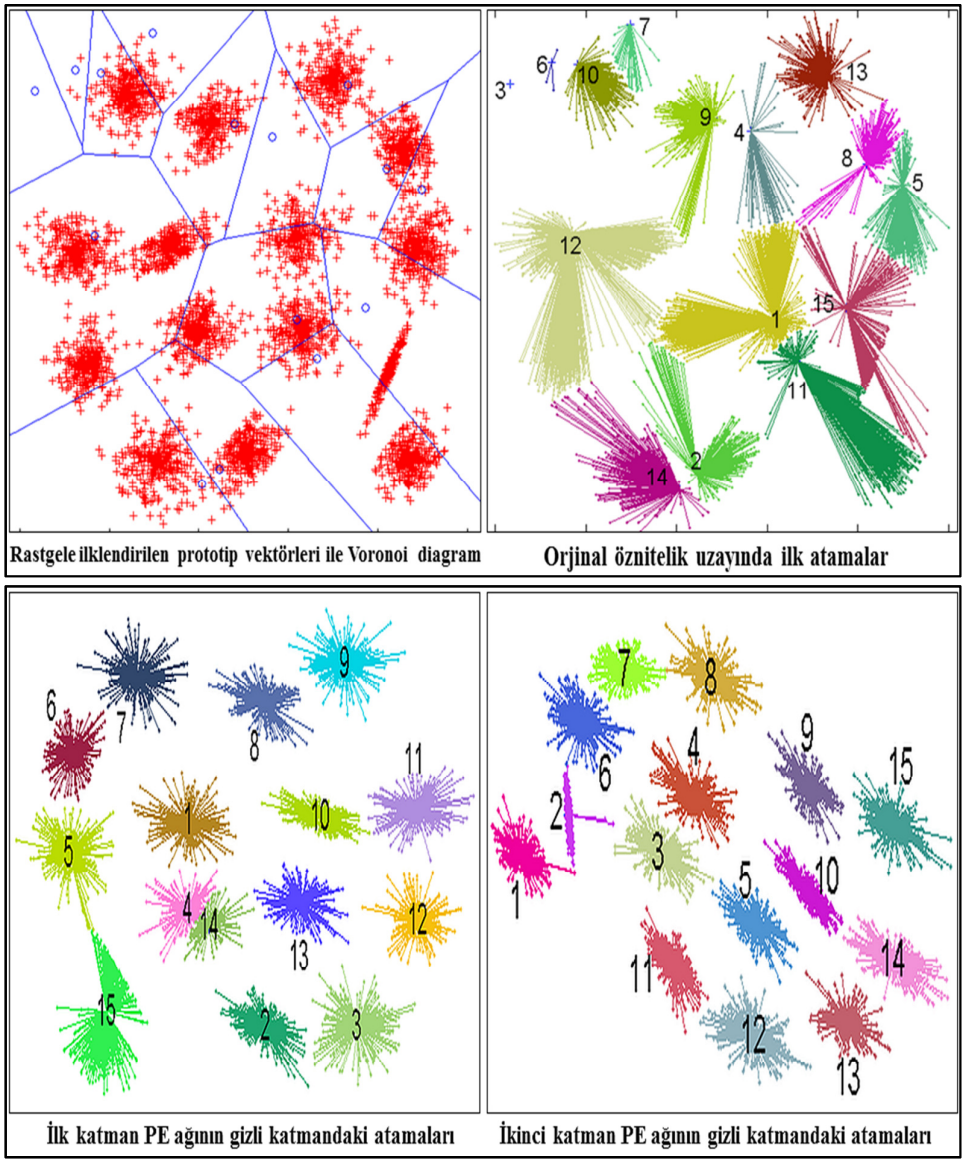
Son olarak; önerilen PKA mimarisinin ara katman aktivasyon çıkışları görsel olarak analiz edilmiştir. Böylece gizli katmanda öğrenilen örüntülerin girdi verisinin orjinal uzayına nasıl bir dönüşüm uyguladığı görülmüştür. Ayrıca; tek ara katmanlı PKA mimaride geliştirilen öbekleme sonuçlarının birden çok katmanlı ve ara katmanlarda farklı miktarda nöron olmak üzere yığıt şekline dönüştürülmesi ele alınmıştır. Bu nedenle, iki PKA yapısı ardışık olarak birleştirilmiş; ilk PKA'nın öğrenilmesiyle elde edilen ara katman aktivasyon çıkışları ikinci PKA yapısına girdi verisi olarak sunulmuştur. Dikkat edilirse; ilk PKA orjinal öznitelik uzayını, ikincisi ise ilk ara katman çıkışlarını kullanacağı için doğal olarak ara katmanlardaki nöron sayıları da değişken olacaktır.

Detay olarak, S_1-set ve rastgele ilklendirilen prototip vektörleri daha önceki görsel sonuçlar ile karşılaştırılmak üzere kullanılmıştır. İlk PKA ağırlık vektörleri optimize edildikten sonra, bu ağın kodlayıcı tarafı kullanılmak üzere hem girdi verisi hem de başlangıçta rastgele seçilen prototip vektörleri yeni ve daha büyük boyutlu uzaya taşınmıştır. Müteakiben ikinci PKA yapısı bu yeni girdi uzayı ile benzer koşullarda koşulmuştur. Her iki PKA'nın gizli katmanlarından elde edilen değerlerin ilk iki ana bileşeni Principal Component Analysis (PCA) yöntemiyle seçilmiş ve uzay dönüşümünü gösteren görsel sonuçlar Şekil 3.7'de sunulmuştur.

Şekil 3.7'de her zamanki gibi öncelikle girdi verisi ve rastgele ilklendirmeye göre ilk örnek-öbek eşleştirmesi orjinal öznitelik uzayında plotlanmıştır. Başlangıç koşullarında 3 nolu öbeğe hiç bir örnek üye olarak atanmamış ve öbek merkezleri mevkisel olarak belirli alanlarda özellikle gruplanmıştır. İlk PKA mimarisinin ardından öbekler arası mesafenin arttırılacağı şekilde orjinal öznitelik uzayının ayırt edici biçimde dönüştürüldüğü gözlenmiştir.

Benzer husus ikinci PKA yapısı için de geçerlidir. İlkinden ayrıcalıklı olarak; ikinci PKA yapısında ilk PKA'daki 4 ve 14 nolu öbeklerin 3 nolu öbekte birleştirildiği, ilk

PKA'daki 15 nolu öbeğin ise ayrıştırılarak 2 nolu yeni bir öbeğin keşfedildiği gözlenmiştir. Bu ise veri kümesindeki gerçek durumu yansıtmaktadır. Dolayısı ile PKA mimarilerinin yığıtlar şeklinde birbirine eklenmesiyle ara katmanlarda kodlayıcı vektörel dönüşümler sonucu daha ayırt edici öznitelik uzaylarının elde edildiği ve öbekleme performansının arttığı sonucuna varılmıştır.

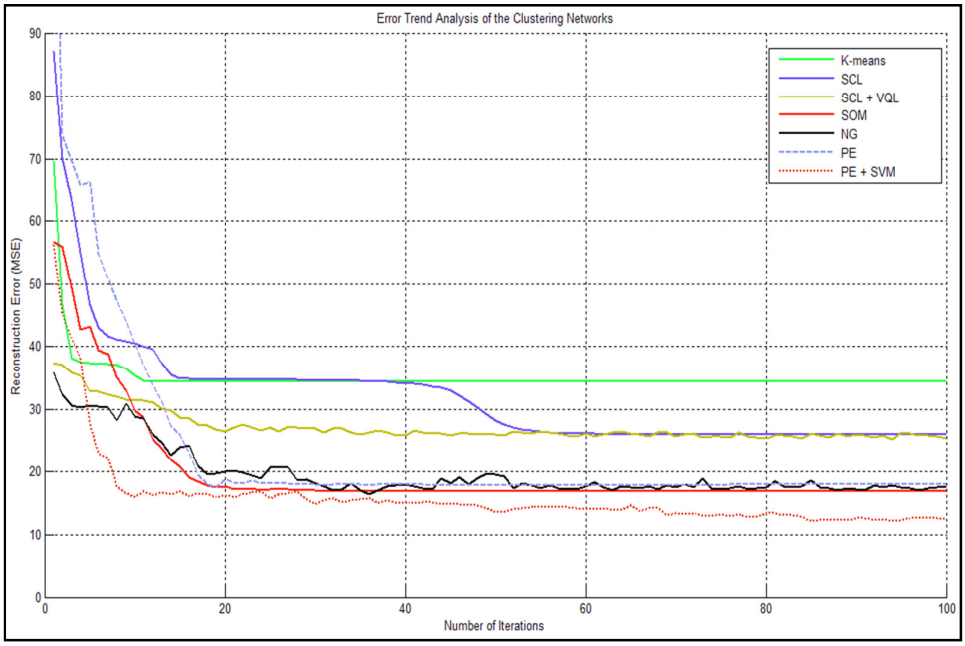


Şekil 3.7. S_1 seti üzerinde ardışık PKA yapısı gizli katman dönüşümleri

3.1.2. Analitik performans analiz sonuçları

Bu bölümde öncelikle öbekleme ağlarının Ortalama Karesel Hatalar (Mean Square Error -MSE)'daki eğilimi S_2 sentetik veri seti ile wine setleri üzerinde analiz edilmiş

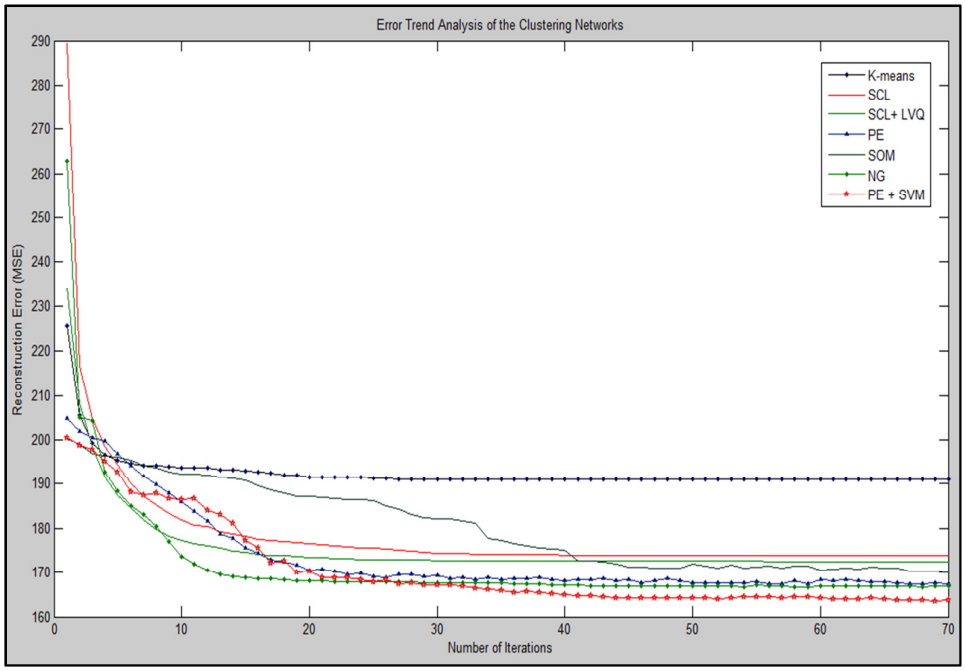
ve sonuçları Şekil 3.8, 3.9 ve 3.10'da gösterilmiştir. Daha sonra Tablo 3.2 ve 3.3'te deneysel sonuçlarda bütünlük sağlamak maksadıyla bütün veri setlerine ait MSE ve zaman analiz sonuçları verilmiştir. Deneylerde tüm veri setleri için aynı rastgele ilklendirilen prototip vektörleri kullanılmış; yineleme sınırı 100 olarak tespit edilmiştir.



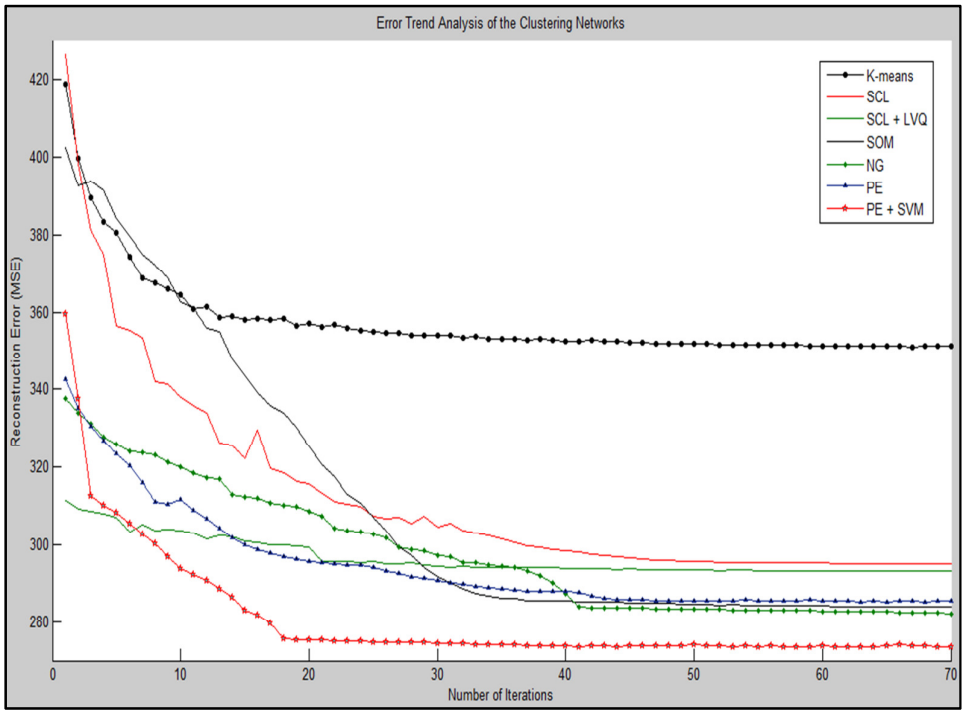
Şekil 3.8. S_2 seti üzerinde yinelemeli hata trend plotlaması

Beklendiği gibi, k-means algoritması içlerinde en hızlı doyuma ulaşan ancak en kötü hata oranına sahip metod olarak göze çarpmaktadır. Her ne kadar SCL ağından elde ettiği öncül bilgiyi (örnek-öbek atamaları ve prototip vektörleri) denetim maksadıyla kullanan LVQ ağ mimarisi başlangıç döngülerinde SCL'den daha düşük hata oranlarına sahip olsa da; son yinelemelerde SCL'e göre LVQ'nun az daha düşük hata trendine sahip olduğu gözlenmiştir. WTA ağlarından ayrı olarak, SOM, NG ve PKA ağ algoritmalarının komşuluk fonksiyonlarını kullanmaları nedeni ile benzer ve WTA metodlarından çok daha iyi performanslar sergileri ortaya çıkmıştır. Diğer taraftan DVMA (PE+SVM) ile elde edilen öbek iyileştirmesi sonucunda ise en iyi performans neticesi elde edilmiştir. Performansta gözlemlenen kayda değer artışın PKA başlangıç denetimi ve elbette yinelemeler müddetince öbek güncellemelerinin gizli nitelik uzaylarında icra edilmesinden kaynaklandığı değerlendirilmiştir. NFB ağ

mimarilerinde hata trendi esnasında gözüken osilasyonun stokastik gradyen azaltım algoritmasının bir sonucu olduğu varsayılmaktadır.



Şekil 3.9. Wine-red seti üzerinde yinelemeli hata trend plotlaması



Şekil 3.10. Wine-white seti üzerinde yinelemeli hata trend plotlaması

Tablo 3.2. Sentetik verisetlerine ait MSE ve zamanlama analiz sonuçları

Algoritmalar	Sentetik Verisetleri							
	S_1		S_2		S_3		S_4	
	MSE	**Time**	**MSE**	**Time**	**MSE**	**Time**	**MSE**	**Time**
K-means	28,71 ±12,44	13,32 ±0,46	26,70 ±4,86	14,79 ±0,71	29,99 ±3,15	14,83 ±0,76	31,18 ±1,19	14,36 ±0,76
SCL	21,28 ±7,11	23,44 ±0,62	21,82 ±4,91	24,59 ±0,41	25,71 ±1,60	26,41 ±0,49	26,29 ±2,07	16,55 ±0,53
SCL+LVQ	20,24 ±6,20	89,77 ±4,58	20,80 ±3,66	93,33 ±0,26	24,93 ±1,38	83,63 ±0,22	23,12 ±1,11	84,10 ±0,25
SOM	12,27 ±2,69	97,35 ±5,76	18,46 ±2,89	90,42 ±2,32	22,70 ±1,04	92,64 ±1,48	20,19 ±0,99	90,81 ±0,20
NG	14,01 ±1,09	53,93 ±0,79	17,93 ±1,44	55,15 ±0,90	23,98 ±0,63	58,25 ±0,92	20,95 ±0,36	56,49 ±1,52
PE	**12,46 ±3,12**	**68,98 ±1,87**	**19,19 ±1,31**	**66,76 ±0,76**	**22,83 ±1,16**	**76,15 ±0,52**	**20,41 ±0,58**	**65,43 ±1,50**
PE+SVM	**10,32 ±1,44**	**122,69 ±6,86**	**16,86 ±2,45**	**128,69 ±6,41**	**17,71 ±1,50**	**137,16 ±4,18**	**18,34 ±1,31**	**134,82 ±4,65**

Tablo 3.3. UCI verisetlerine ait MSE ve zamanlama analiz sonuçları

Algoritmalar	UCI Verisetleri							
	wine_red		**wine_white**		**letter_recognition**		**clave_vectors**	
	MSE	**Time**	**MSE**	**Time**	**MSE**	**Time**	**MSE**	**Time**
K-means	276,38 ±4,41	7,36 ±0,06	408,57 ±12,39	18,26 ±0,61	3306,28 ±44,89	61,49 ±0,46	41297 ±201	31,05 ±0,56
SCL	234,21 ±4,62	7,26 ±0,13	346,85 ±19,96	21,14 ±0,88	3172,38 ±67,24	64,61 ±0,57	39812 ±329	36,14 ±0,76
SCL+LVQ	221,18 ±4,08	24,21 ±1,89	328,52 ±12,16	43,90 ±0,30	3033,88 ±63,51	180,06 ±18,87	39270 ±24	64,16 ±0,42
SOM	180,43 ±3,24	26,41 ±1,48	303,40 ±6,50	53,44 ±0,29	2865,60 ±45,35	213,77 ±5,35	37010 ±159	109,49 ±0,53
NG	179,84 ±3,46	21,27 ±0,65	291,95 ±2,56	42,14 ±0,11	2845,62 ±35,86	170,11 ±3,26	37208 ±124	73,67 ±1,19
PE	**176,56 ±0,73**	**24,41 ±1,87**	**296,99 ±6,11**	**60,34 ±0,52**	**2847,76 ±32,82**	**204,53 ±33,29**	**37364 ±134**	**93,53 ±1,85**
PE+SVM	**168,50 ±1,19**	**44,73 ±1,93**	**284,24 ±1,85**	**119,61 ±0,51**	**2746,91 ±12,74**	**316,70 ±37,98**	**36350 ±87**	**144,94 ±2,83**

Tablo 3.2 ve 3.3'te bütün veri setlerine ait MSE (Euclid metrik sisteminde) ve zaman (saniye cinsinden) analizler sonuçları sunulmuştur. Deneyler 50 kez tekrarlandığı için ortalama ve standart sapma değerleri ayrıca hesaplanmıştır. Genel olarak; DVMA (PE+SVM) yapısı bütün durumlarda en iyi performansı sergilemiştir. Diğer

taraftan; önerilen melez ağ mimarisinin veri öbeklemede en yakın rakibine göre iki katı kadar bir sürede işlemleri tamamladığı gözlenmiştir. Bunun nedeninin iki ağ yapısının ardışık olarak koşulması ve ağırlık parametrelerinin yinelemelerde mini paketler halinde güncellenmesi olduğu düşünülmektedir. Ancak önerilen her iki ağ yapısının zamanlama olarak bireysel açıdan benzerleri ile mukayese edilebilir oldukları ve önceliğin öbekleme performansına verildiği durumlarda birlikte kullanılabilecekleri önerilmiştir.

Öbek içi bütünlük (Compactness) ve öbekler arası ayrışım (Separation) ölçütleri literatürde öbekleme validasyonu için kullanılan iki önemli terimdir [48]. Öbekleme validasyon değeri şu şekilde hesaplanır;

$$\text{Validity} = \frac{1}{K}\sum_{j=1}^{K} \max_{j\neq m}\{\frac{d_{wcs}(c^{(j)}) + d_{wcs}(c^{(m)})}{d_{bcs}(c^{(j)}, c^{(m)})}\} \tag{3.1}$$

$$d_{wcs}(c^{(j)}) = \frac{\sum_{i=1}^{P} \left\| x^{(i)} - c^{(j)} \right\|}{P} \;;\; \forall x^{(i)} \in C_j \tag{3.2}$$

$$d_{bcs}(c^{(j)}, c^{(m)}) = \left\| c^{(j)} - c^{(m)} \right\|_{L_2} \tag{3.3}$$

Burada öbek içi saçılım, d_{wcs}, bir öbeği temsil eden prototip (sentroid) vektör, $c^{(j)}$, ile ilgili öbeğe atanmış bulunan örnek verisinin, $x^{(i)}$, arasındaki ortalama mesafedir. Öbekler arası ayrışım miktarı , d_{BCS}, ise; öbek j ve m için, basitçe öbekleri temsil eden prototipler arasındaki mesafe olarak hesaplanmaktadır. Bu kapsamda en iyi öbekleme metodunun öbek validasyon değerini minimize eden algoritma olacağı aşikardır. Tablo 3.4'te öbekleme ağ algoritmalarının bütün veri setlerindeki validasyon sonuçları karşılaştırmalı olarak sergilenmiştir. Her ne kadar önerilen PKA (PE) yapısı SOM ve NG ağları ile benzerlik gösterse de; melez çok katmanlı ağın (PE+SVM) tüm veri setlerinde en düşük validasyon değerine ulaştığı ve bu durumun daha önceki görsel ve analitik sonuçları desteklediği tespit edilmiştir. Ayrıca; örnek sayısının ve öznitelik boyutunun arttıkça NFB ağlarının WTA mimarilerine göre çok daha iyi öbekleme başarısı gösterdiği görülmüştür.

Tablo 3.4. Bütünlük ve ayrışım temelli validasyon analiz sonuçları

Algoritmalar	Sentetik Verisetleri				UCI Verisetleri			
	S_1	**S_2**	**S_3**	**S_4**	**wine_ red**	**wine_ white**	**letter recog-nition**	**clave vectors**
K-means	0,85 ±0,09	0,63 ±0,09	0,85 ±0,04	0,82 ±0,03	2,86 ±0,12	2,88 ±0,33	2,95 ±0,17	7,84 ±0,10
SCL	0,61 ±0,12	0,62 ±0,08	0,78 ±0,03	0,76 ±0,04	2,85 ±0,08	2,73 ±0,08	2,77 ±0,04	6,83 ±0,13
SCL+LVQ	0,60 ±0,11	0,62 ±0,05	0,76 ±0,04	0,73 ±0,03	2,84 ±0,07	2,71 ±0,07	2,74 ±0,04	6,81 ±0,11
SOM	0,42 ±0,06	0,55 ±0,06	0,73 ±0,05	0,71 ±0,04	2,07 ±0,31	1,94 ±0,03	1,70 ±0,02	3,72 ±0,16
NG	0,41 ±0,06	0,52 ±0,10	0,74 ±0,04	0,72 ±0,02	2,04 ±0,18	1,90 ±0,09	1,72 ±0,03	3,48 ±0,13
PE	**0,41 ±0,05**	**0,55 ±0,04**	**0,74 ±0,03**	**0,71 ±0,03**	**2,04 ±0,11**	**1,96 ±0,16**	**1,71 ±0,04**	**3,52 ±0,18**
PE+SVM	**0,39 ±0,06**	**0,49 ±0,03**	**0,70 ±0,03**	**0,68 ±0,02**	**1,96 ±0,05**	**1,67 ±0,11**	**1,68 ±0,03**	**3,22 ±0,14**

Silhouette metriği öbekleme problemlerinin çözümünde kullanılan uygulamaların performans doğrulamasını ve anlaşılırlığını gözlemleyen ve literatürde sıkça başvurulan diğer bir metoddur [100]. Daha önceden öbeklenmiş (yani örnek-öbek ilişkisi belli olan) bir veri seti için, her bir örneğin atandığı öbeğe ve en yakın komşuluğundaki bir diğer öbeğe göre uyumluluk oranını tespit eder. Silhouette, $S(\cdot)$, uyum oranı şu şekilde hesaplanır;

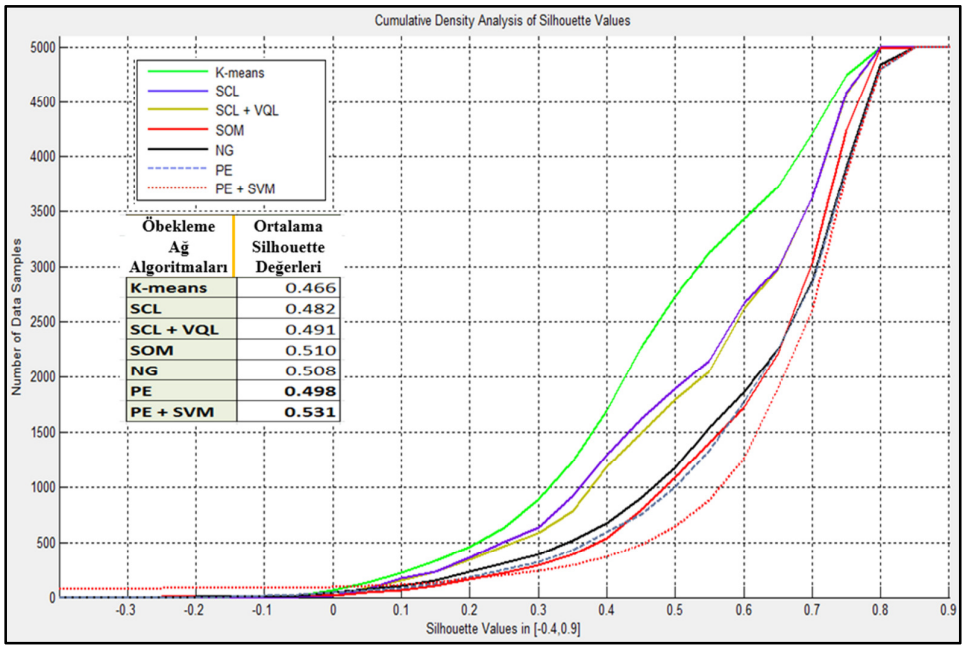
$$a(x^{(i)}) = \frac{\sum_{n=1}^{N} \left\| x^{(i)} - x^{(n)} \right\|}{N} \; ; \; \forall x^{(i)} \in C_j \tag{3.4}$$

$$b(x^{(i)}) = \frac{\sum_{m=1}^{M} \left\| x^{(i)} - x^{(m)} \right\|}{M} \; ; \; \forall x^{(i)} \in C_j, \forall x^{(m)} \in C_z \tag{3.5}$$

$$C_z = c^{\left(\text{argmin}_{j=1,2,3,\ldots,K} \left\| x^{(i)} - c^{(j)} \right\|\right)} \tag{3.6}$$

$$S(x^{(i)}) = \begin{cases} 1 - \dfrac{a(x^{(i)})}{b(x^{(i)})} & ; \; a(x^{(i)}) < b(x^{(i)}) \\ 0 & ; \; a(x^{(i)}) = b(x^{(i)}) \\ \dfrac{b(x^{(i)})}{a(x^{(i)})} - 1 & ; \; a(x^{(i)}) > b(x^{(i)}) \end{cases} \tag{3.7}$$

Burada $a(\cdot)$ bir örneğin, $x^{(i)}$, atanmış olduğu bir öbek ile ne derece uyumlu olduğunu (ne kadar düşükse o kadar iyi uyum) gösterir. Diğer taraftan $b(\cdot)$, bir örneğin, $x^{(i)}$, atanmış olduğu öbek haricindeki komşu en yakın öbek ile olan ortalama benzemezlik miktarını (ne kadar fazlaysa o kadar çok benzemez) verir. Yukarıdaki tanımlardan hareketle $S(\cdot)$ uyum oranının (-1,1) aralığında ölçüldüğü ve en iyi öbekleme algoritmasının bu oranı maksimize etmeye çabaladığı değerlendirilmiştir.

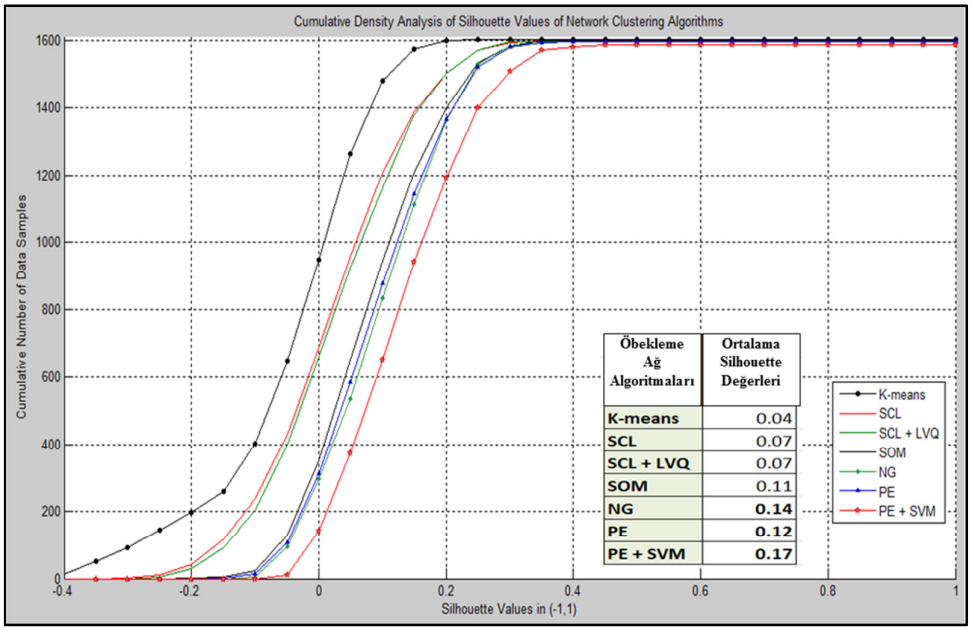


Öbekleme Ağ Algoritmaları	Ortalama Silhouette Değerleri
K-means	0.466
SCL	0.482
SCL + VQL	0.491
SOM	0.510
NG	0.508
PE	0.498
PE + SVM	0.531

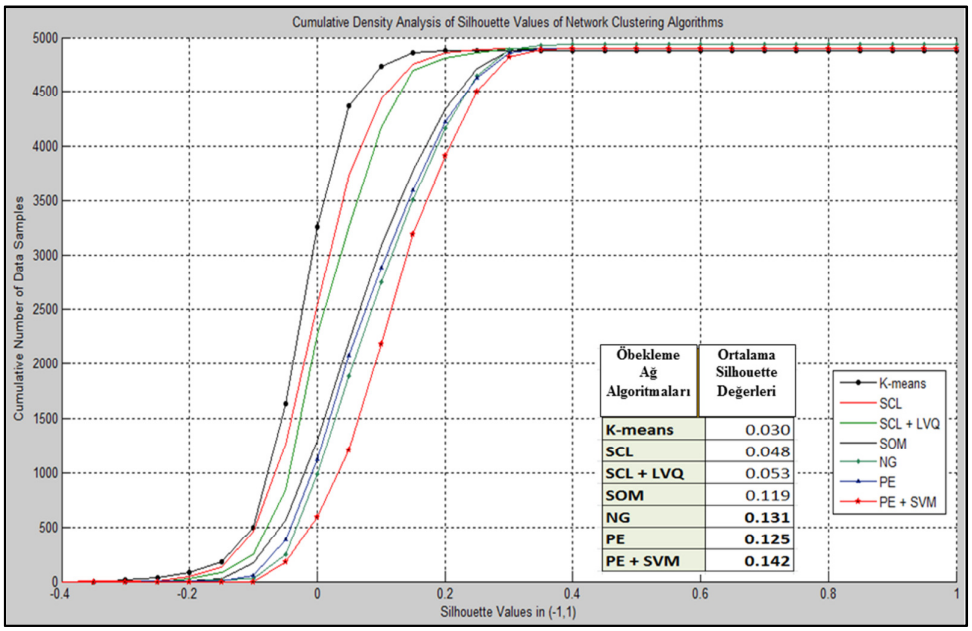
Şekil 3.11. S_3 seti üzerinde öbekleme algoritmalarının Silhouette analizi

Silhouette değerleri üzerinden öbekleme ağ algoritmalarının karşılaştırması Şekil 3.11, 3.12 ve 3.13'te verilmiştir. Her bir örnek kendi $S(\cdot)$ değerine sahip olacağından, algoritmanın genel öbekleme performansını ölçmek için öncelikle örnek veri noktalarının elde ettiği bireysel $S(\cdot)$ miktarları 0,2 genişliğindeki aralıklardan oluşan bir histograma dökülmüştür. Müteakiben, daha anlaşılır çıkarımlar elde edebilmek için her bir metodun sahip olduğu histogram kümülatif yoğunlukları yansıtacak şekilde yeniden dizayn edilmiştir. Bu kapsamda kümülatif yoğunluk plotlamasında

bir algoritmanın ne kadar geç yükselişe geçerse o kadar iyi öbekleme performansına sahip olacağı aşikardır.



Şekil 3.12. Wine-red seti üzerinde öbekleme algoritmalarının Silhouette analizi



Şekil 3.13. Wine-white seti üzerinde öbekleme algoritmalarının Silhouette analizi

Şimdiye kadar yapılan tüm performans karşılaştırmalarında olduğu gibi, DVMA (PE+SVM) ağ mimarisi Silhouette metrik sistemde de en iyi performansı elde

etmiştir. Çünkü bu kompozit algoritmanın kümülatif yoğunluk plotlaması en son yükselişe geçmiş ve ortalama Silhouette değeri en yüksek metod yine önerilen bu melez metoddur. Daha önceki karşılaştırmalı görsel ve analitik performans yorumları diğer algoritmalar için aynen geçerlidir. Tablo 3.5'te ise tüm veri setlerine ait Silhouette değerleri ortalama ve standart standart sapmaları ile birlikte verilmiştir.

Tablo 3.5. Silhouette değerleri üzerinden karşılaştırmalı analiz sonuçları

Algoritmalar	**Sentetik Verisetleri**				**UCI Verisetleri**			
	S_1	**S_2**	**S_3**	**S_4**	**wine_ red**	**wine_ white**	**letter recog-nition**	**clave vectors**
K-means	0,61 ±0,04	0,56 ±0,04	0,44 ±0,01	0,41 ±0,03	0,04 ±0,01	0,06 ±0,01	0,09 ±0,01	0,03 ±0,01
SCL	0,62 ±0,04	0,56 ±0,04	0,45 ±0,01	0,42 ±0,03	0,07 ±0,01	0,07 ±0,01	0,10 ±0,01	0,04 ±0,01
SCL+LVQ	0,62 ±0,03	0,57 ±0,04	0,46 ±0,01	0,42 ±0,03	0,07 ±0,01	0,07 ±0,01	0,10 ±0,01	0,04 ±0,01
SOM	0,70 ±0,02	0,59 ±0,02	0,49 ±0,02	0,43 ±0,01	0,11 ±0,01	0,12 ±0,01	0,14 ±0,01	0,06 ±0,01
NG	0,69 ±0,02	0,62 ±0,01	0,51 ±0,01	0,43 ±0,01	0,14 ±0,01	0,12 ±0,01	0,14 ±0,01	0,05 ±0,01
PE	**0,69 ±0,02**	**0,61 ±0,02**	**0,50 ±0,02**	**0,43 ±0,01**	**0,12 ±0,01**	**0,11 ±0,01**	**0,14 ±0,01**	**0,05 ±0,01**
PE+SVM	**0,72 ±0,02**	**0,62 ±0,02**	**0,52 ±0,02**	**0,44 ±0,02**	**0,17 ±0,01**	**0,14 ±0,01**	**0,15 ±0,01**	**0,07 ±0,01**

Son olarak, ağ mimarilerinin bütün veri setleri üzerindeki öbekleme performansları başarı ve saflık metrik ölçüsünde analiz edilmiş ve analiz sonuçları Tablo 3.6 ve 3.7'de yüzdelik (%) ifadelerle gösterilmiştir. Veri örneklerinin gerçek öbek etiketleri verildiği takdirde, başarı yüzdesi öncelikle verinin algoritma içerisindeki öbek atanmaları ile gerçek öbek etiketleri arasındaki en iyi kombinasyon bulunmak suretiyle hesaplanır. O halde başarı, Doğru Pozitif (True Positive) miktarının örnek sayısına oranıdır. Saflık oranına gelince; her bir öbek başlangıçta o öbek içerisinde en çok örnek sayısına sahip gerçek etiketin bilgisi ile etiketlenir (Majority Voting). Sonrasında ise saflık başarı ile aynı şartlarda hesaplanır. Dikkat edilirse gerçek etiket – öbek çiftleri bire-çok ilişkisine sahip olabilir. Yani eğer bir gerçek etiket sayısı birden çok öbekte en yüksek çıkarsa, her iki öbek de aynı etiket ile etiketlenebilir.

Her ne kadar PKA (PE) ağ mimarisi SOM ve NG algoritmalarına kıyasla tüm veri setlerinde genelde biraz daha düşük başarı ve saflık oranlarına sahip olsa da; DVMA (PE+SVM) kompozit ağ mimarisi PKA'dan elde ettiği öbekleri gizli nitelik uzayında güncellediği için diğer algoritmalardan daha iyi performanslar sergilemiştir. Başarı ve saflık sonuçları daha önceki görsel ve istatistiki öbekleme neticelerini desteklemektedir.

Tablo 3.6. Sentetik verisetlerinde başarı ve saflık analiz sonuçları

Algoritmalar	**Sentetik Verisetleri**							
	S_1		**S_2**		**S_3**		**S_4**	
	acc.	**pur.**	**acc.**	**pur.**	**acc.**	**pur.**	**acc.**	**pur.**
K-means	81,20 ±9,42	93,47 ±1,18	81,90 ±9,18	92,33 ±0,98	66,99 ±6,21	71,14 ±1,26	65,40 ±1,43	67,46 ±1,43
SCL	84,56 ±9,03	93,25 ±1,10	82,77 ±9,96	92,56 ±1,18	68,99 ±6,67	72,90 ±1,72	66,52 ±2,93	68,33 ±1,93
SCL+LVQ	84,79 ±9,16	93,49 ±1,24	83,56 ±9,01	92,63 ±1,11	69,95 ±6,72	74,83 ±1,33	67,94 ±2,04	69,22 ±1,46
SOM	91,61 ±4,38	95,20 ±0,25	90,03 ±5,77	94,12 ±1,39	77,70 ±5,74	81,00 ±1,18	74,07 ±2,81	78,48 ±1,03
NG	93,74 ±3,63	96,00 ±0,12	92,65 ±3,47	94,89 ±0,55	78,69 ±6,42	81,77 ±1,19	75,12 ±1,04	78,44 ±1,04
PE	**91,85 ±3,56**	**94,91 ±0,08**	**91,20 ±2,04**	**94,15 ±1,00**	**77,65 ±5,50**	**81,38 ±1,42**	**73,94 ±2,83**	**78,12 ±0,68**
PE+SVM	**95,73 ±1,18**	**96,73 ±0,45**	**93,86 ±2,95**	**94,73 ±0,46**	**79,66 ±3,32**	**83,26 ±0,96**	**77,84 ±2,27**	**80,29 ±1,02**

Tablo 3.7. UCI verisetlerinde başarı ve saflık analiz sonuçları

Algoritmalar	**UCI Verisetleri**							
	wine_red		**wine_white**		**letter_recognition**		**clave_vectors**	
	acc.	**pur.**	**acc.**	**pur.**	**acc.**	**pur.**	**acc.**	**pur.**
K-means	15,50 ±3,87	29,57 ±2,17	15,68 ±2,20	20,79 ±1,52	19,50 ±1,57	27,15 ±1,10	37,07 ±4,63	38,39 ±3,39
SCL	17,55 ±4,28	32,26 ±1,77	16,08 ±2,24	22,30 ±2,96	22,57 ±1,40	29,50 ±0,66	39,59 ±5,51	43,98 ±4,80
SCL+LVQ	17,70 ±3,86	33,36 ±2,42	16,83 ±2,10	23,36 ±1,17	22,94 ±1,69	30,43 ±1,13	39,47 ±5,50	43,46 ±3,96
SOM	20,72 ±1,45	46,50 ±4,49	20,05 ±1,26	25,92 ±1,03	24,93 ±1,49	33,32 ±0,82	42,82 ±3,35	44,90 ±2,67
NG	21,19 ±2,65	46,70 ±3,36	21,11 ±2,48	26,28 ±1,47	25,89 ±1,43	35,15 ±0,61	47,23 ±4,36	47,78 ±4,17
PE	**22,14 ±2,06**	**46,92 ±4,39**	**21,96 ±2,55**	**26,34 ±1,69**	**26,88 ±1,56**	**36,29 ±1,23**	**46,43 ±3,97**	**47,58 ±4,45**
PE+SVM	**23,31 ±2,72**	**48,12 ±3,84**	**23,40 ±1,80**	**29,84 ±1,26**	**27,50 ±1,57**	**37,13 ±1,10**	**47,44 ±3,63**	**48,07 ±3,68**

3.2. Derin Ağ Yapısında Hiyerarşik Görüntü Temsili Deneysel Tasarım

Bölüm 2.2'de detayları verilmiş ve bu tezin ilk ürünlerinden PKA yapısını esas alarak mevki bilgisinin de sisteme tabi tutulduğu çok katmanlı ağ mimarisi, nesne tanımaya yönelik literatürde sıkça kullanılan CIFAR-10 isimli geniş ölçekli veriseti

üzerinde tecrübe edilmiştir. Bu verisetine ilişkin detaylar [101]'de verilmiştir. Eğitim maksadıyla kullanılan kısmı özellikle belirlenerek beş eşit parçaya ayrılmış ve 32x32 piksel ebatlarında toplamda 50.000 renkli imgeden oluşmaktadır. Test veriseti ise 10.000 imgeden oluşmaktadır. 10 nesne kategorisinden toplamda 60.000 renkli imgenin yer aldığı deneyler de amaç; öğrenilen sınıf modellerini eğitimlerde veya optimizasyonda kullanılmayan test veriseti üzerinde kullanarak her birini ilgili nesne kategorisine atamaktır. Verisetine ilişkin özet bilgi Şekil 3.14'te verilmiştir.



Şekil 3.14. CIFAR-10 veriseti

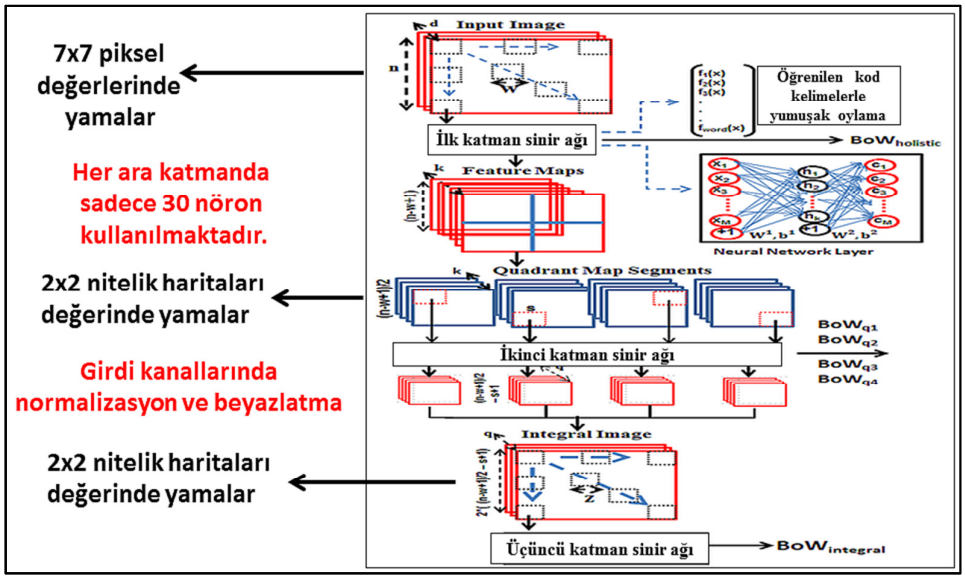
Denetimsiz nitelik çıkartımı için 3 ara katmanlı bir ağ mimarisi oluşturulmuştur. Görsel kelimeler (kodlar) fırsatçı katman tabanlı öğrenme metodu ile her bir katmanda ardışık olarak bulunurken aynı zamanda ağ parametreleri optimize edilmiştir. Tespit edilen kelimeler görüntü verisini hiyerarşik GKS öznitelik vektörleri ile temsil etmek maksadıyla kullanılmış; öğrenilen katmanlar ise girdi verisini bir sonraki katmanlara taşıyacak nitelik haritalarına dönüştürmüştür.

Detay olarak; ilk katman için 7x7, ikinci ve üçüncü katmanlar için ise 2x2 piksel boyutlarında ve 1 piksel aralıklarla elde edilen yamalar girdi verisini oluşturmuştur. Her bir katman için rastgele seçilen çok miktarda yama girdisiyle (yaklaşık 500.000) ağ parametreleri optimize edilmiştir. İlk katman sonrası ikinci ve üçüncü katmanlarda yama ebatları iki önemli nedenden ötürü azaltılmıştır. Birincisi, ilk ara katmanın ardından diğer katmanlara bir önceki ara katmandaki nöron sayısını ifade

eden h derinliğinde nitelik haritaları girdi olarak verilmiştir. Bu noktada ebatları arttırmak ağın karmaşıklığını arttırmak anlamına gelecektir. İkincisi, ayırt edici bir GKS temsili oluşturmak maksadıyla girdi olarak her katmanda yeterince çok miktarda yama örneğinin eldesi hedeflenmiştir. Bu da yama ebatlarının azaltılması ile mümkündür. Ağ karmaşıklığı (bias) ile örnek çokluğunu (variance) dengelemek üzere Quoc ve diğ. [31] tarafından önerildiği şekilde, yama pencerelerindeki her bir aktivasyon kanalının ortalama değeri de girdi olarak sisteme dahil edilmiştir. Bunun yanı sıra, ara katmanlarda hiperpolik tanjant fonksiyonu kullanılmış; çıkış katmanında ise aktivasyon fonksiyonu kullanılmamıştır (doğrusal).

Diğer bir uygulama detayı ise; literatürde yer alan benzer çalışmalarda [32, 37, 38, 39, 43] örnekler için yeni öznitelik uzayları oluşturulmasında ara katman aktivasyon çıkışları doğrudan kullanılırken, bu tezde sınıflandırma için GKS temsil vektörleri kullanılmıştır. Aktivasyon değerlerini kullanmak sisteme ilave hesap yükü getiren ara katman nöron sayılarının arttırılmasına sebep olmuştur. Önerilen yaklaşımda ise öznitelik temsil uzayının boyutu ağ yapısından bağımsız kılınarak sadece öğrenilen görsel kelime sayısı ile ilişkilendirilmiştir. Önerilen yaklaşımın en önemli avantajlarından birisi budur.

Literatürdeki benzer çalışmalarla önerilen metodu mukayese edebilmek adına; tıpkı [39]'daki deneylerde gibi görsel kelime sayısı sırasıyla 100, 200, 400, 800, 1200, ve 1600 olarak ayarlanmıştır. Ara katmanlardaki nöron sayısı çapraz validasyon yöntemiyle ve girdi vektörlerinin boyutundan çok daha düşük bir şekilde 30 olarak hesaplanmıştır. Son olarak; görüntü ön işleme kapsamında yalnızca ilk katmanda karşıtlık normalizasyonu ve tüm katmanlarda beyazlatma (whitening) [39] uygulanmıştır. Bahse konu çok katmanlı ağ mimarisine ait deneysel tasarım Şekil 3.15'te sunulmuştur.



Şekil 3.15. Çok katmanlı ağ mimarisi deneysel tasarımı

3.2.1. Deneysel bulgular

Bulguların ilk kısmında hiyerarşik GKS temsil vektörlerinin sınıflandırma başarısı analiz edilerek Tablo 3.8'de gösterilmiştir. Deneyler 30 defa tekrarlanmış ve ortalama yüzdesel başarı not edilmiştir. Tablo performansları Lazebnik ve diğ. [8] tarafından önerilen piramid formunda tek katman ve ardışık birden çok katmanın birleştirilmesiyle hesaplanmıştır. Piramid formu önceki ve mevcut katmanlardan elde edilen GKS vektörlerinin uç uca eklenmesiyle oluşturulmuştur. Tablo 3.8 incelendiğinde, ağ yapısında derinlere inildikçe daha iyi sonuçlar alındığı görülmüştür. En iyi tek katman başarısı ikinci katmanda % 72,16 ile elde edilmiştir. Dolayısı ile imgelerin çeyrek dilimlere bölünerek sisteme mevki bilgisinin de eklenmesinin performansı arttırdığı sonucuna varılmıştır. Bunun yanı sıra, GKS vektörleri birleştirilerek daha ayırt edici temsil uzayları üretilmiştir. Hiyerarşik GKS temsilinin piramid formunda oluşturulmasıyla sınıflandırmada yaklaşık % 11'lik bir başarı artışı gözlenmiştir. İlaveten, Coates ve diğ. [39] tarafından önerilen benzer çalışmadan da % 2 oranında daha iyi bir sınıflandırma başarısı elde edilmiştir.

Ardından, denetimsiz öznitelik öğrenme yaklaşımı kapsamında literatürde yer alan benzer çalışmalar bu tezde önerilen metod ile karşılaştırmalı olarak Tablo 3.9'da gösterilmiştir. Bu algoritmalarda denetimsiz nitelik öğrenmeye yönelik Bölüm 1'de temel yapı taşları olarak bahsedilen AEs, RBMs ve CNNs gibi ağ mimarileri

kullanılmıştır. Tablo 3.9'da görüleceği gibi, önerilen metod k-means [39] hariç diğerlerinden çok daha basit yapıda olmasına rağmen, en az % 1'lik bir performans artışı sergilemiştir. Buradan maliyet etkin bir şekilde bütün katmanlardan elde edilen özniteliklerin birleştirilmesiyle daha başarılı olunacağı sonucuna varılmıştır.

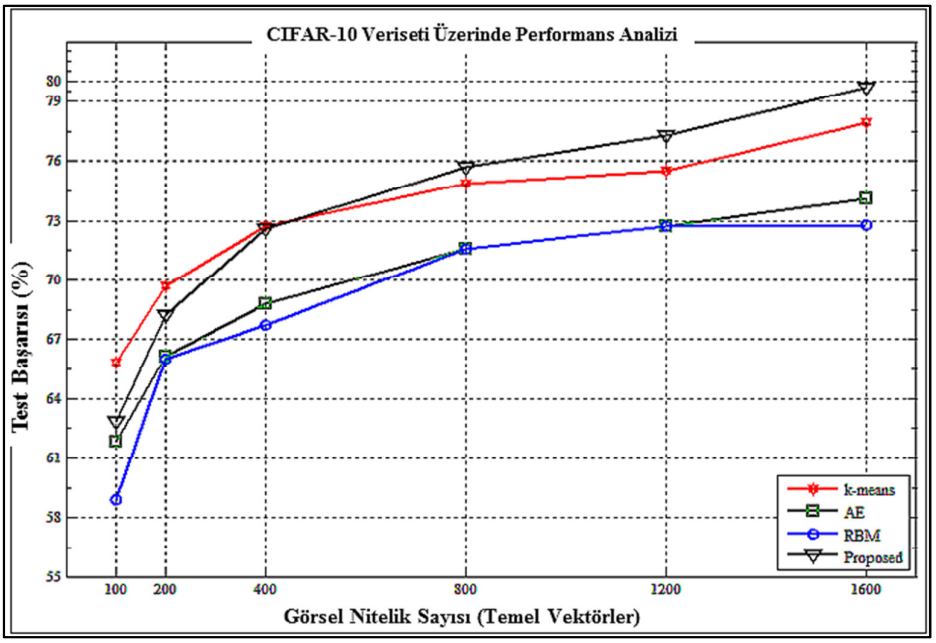
Tablo 3.8. Piramid formunda vektörlerinin sınıflandırma analiz sonuçları

	GKS Piramidi Kelime Sayısı = 1600	
Katman	**Tek Katman (%)**	**Piramid Katmanlar (%)**
1	68,24	---
2	72,16	77,45
3	70,17	**79,72**

Tablo 3.9. CIFAR-10 test veriseti üzerinde sınıflandırma sonuçları

Algoritmalar	**Başarı (%)**
3-way Factorized RBM [103]	65,3
Convolutional RBM [102]	78,9
Sparse Auto-encoder [39]	73,4
Sparse RBM [39]	72,4
Triangle k-means [39]	77,9
Mean-Covariance RBM [46]	71,0
Convolutional Neural Network[37]	77,5
Concolutional Auto-Encoder [37]	78,2
Önerilen Hiyerarşik GKS	**79,7**

Müteakiben, Şekil 3.16'da değişen öznitelik boyutuna göre bazı çalışmaların performans karşılaştırmaları verilmiştir. Öznitelik vektörünün boyutu k-means ve önerilen yaklaşımda görsel kelime sayısına karşılık gelirken, AEs ve RBMs yapılarında gizli katmanlardaki nöron miktarı ile belirlenmiştir. Beklendiği gibi öğrenilen öznitelik boyutu arttıkça bütün algoritmaların başarı oranlarının yükseldiği gözlenmiştir. Şekil 3.16'ya göre bu tezde önerilen yaklaşım, her durumda tek katman AEs ve RBMs metodlarından daha iyi sonuçlar elde etmiştir. Diğer taraftan k-means metodu [39] 400 nitelik boyutuna kadar önerilen algoritmadan daha başarılı olsa da; o noktadan sonra önerilen yöntem üstün gelmiştir. Bu kapsamda basit yapılar içerisinde hiyerarşik öznitelik temsilinin başarı oranını yükselttiği sonucuna varılmıştır.



Şekil 3.16. Farklı boyutlarda öznitelik vektörleriyle performans analizi

Son olarak; ağ yapısına beslenen yamaların ebatlarının genel sınıflandırma yüzdesine etkileri incelenmiş ve sonuçları Tablo 3.10'da sunulmuştur. Nitelik haritalarının giriş yaptığı ikinci ve üçüncü katmanlardaki yama ebatları 2x2 piksel olarak sabitlenirken, piksel yoğunluk değerlerinin girdi verisi olarak kullanıldığı ilk katman yama ebatları değerlendirme maksadıyla değiştirilmiştir. Ayrıca, yama kaydırma aralığı daha önceki gibi 1 piksel, görsel kelime sayısı ise 1600 olarak ayarlanmıştır. Tablo 3.10 incelendiğinde, Coates ve diğ. [39] tarafından da belirtildiği gibi yama ebatlarının sınıflandırma genel başarısına ciddi bir katkısı olmadığı gözlenmiştir. Bunun yerine, kaydırma aralık değerinin ve görsel kelime sayısının daha etkin olduğu sonucuna varılmıştır. Çünkü kaydırma değeri küçüldükçe girdi örnek miktarının artacağı; kelime sayısı arttıkça daha ayırt edici uzayların elde edileceği düşünülmektedir.

Tablo 3.10. Yama boyutlarının başarı oranına etkisi

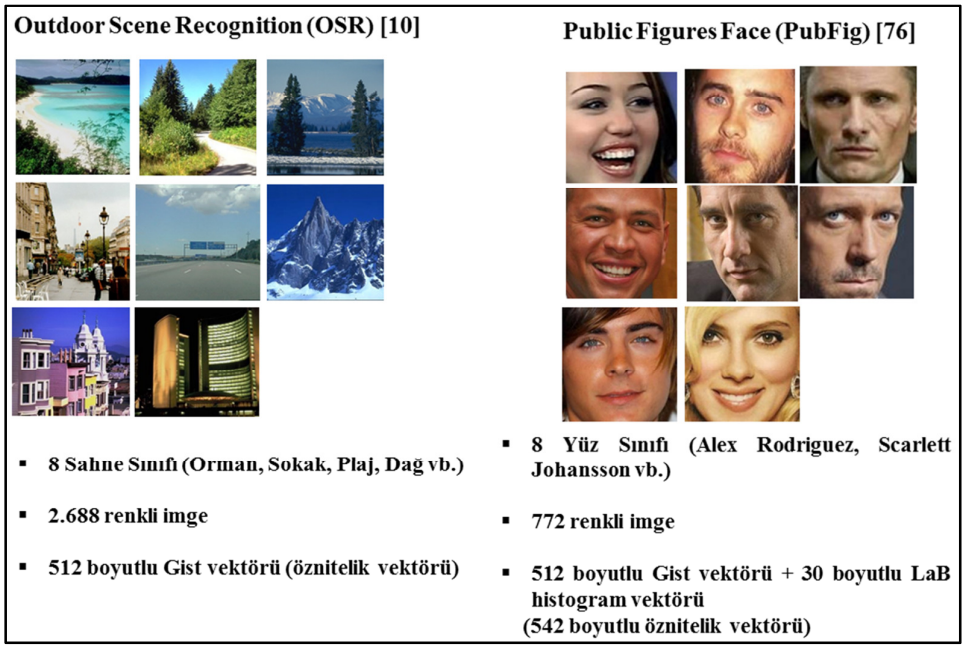
Yama Boyutu	**Başarı (%)**
3	77,6
5	78,9
7	**79,7**
9	79,1
12	78,6

3.3. Denetimsiz Göreceli Nitelik Çıkarımı Deneysel Tasarımı

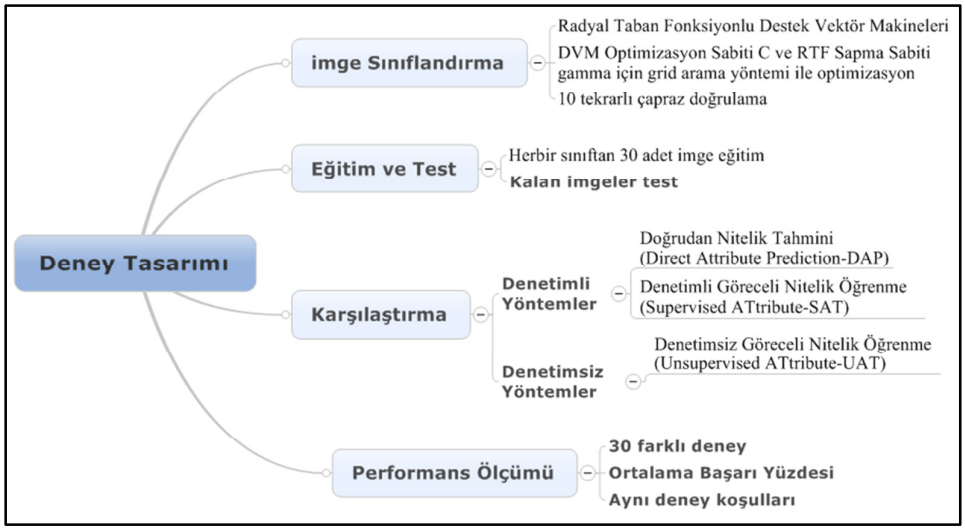
Bölüm 2.3'te detayları verilmiş ve daha önceki bölümlerden farklı olarak, görüntü verisini düşük seviye öznitelik vektörleri (GKS dahil) yerine göreceli nitelik uzaylarında temsile yönelik önerilen yönteme ilişkin deneysel tasarım ve bulgular bu bölümde sunulmuştur.

Hatırlanacağı üzere; insanların zihinlerinde anlamlı bir noktada buluştukları kavramlar şeklinde tanımlanabilecek niteliklerin tespiti sınıflandırma probleminin zorluk derecesine göre denetimli veya denetimsiz, uygulaması ise kategorik (var-yok şeklinde ikili) veya göreceli olabilmektedir. İmge sınıflandırmada niteliklerin öğrenilmesi aşamasında denetimin kalitesi deneklerin tecrübesi ile doğru orantılı ve oldukça emek yoğun bir iştir. Ayrıca, görüntü verisindeki yüksek sınıf içi ve düşük sınıflar arası değişinti miktarları sınıflandırmada niteliklerin kategorik kullanımını çoğunlukla yetersiz kılmaktadır. Bu kapsamda, denetimli nitelik çıkarma işlemindeki problemleri azaltmak amacıyla göreceli niteliklerin denetimsiz bir şekilde bulunmasına yönelik bir yöntem önerilmiştir.

Önerilen yaklaşım, benzerlik ve son zamanlarda geliştirilen başarılı çalışmalar olmaları nedeni ile denetimli Doğrudan Nitelik Tahmini (Direct Attribute Prediction-DAP) [18] ve Göreceli Nitelik Öğrenme (Supervised ATtribute-SAT) Yöntemleri [14] ve denetimsiz Göreceli Nitelik Öğrenme (Unsupervised ATtribute–UAT) Yöntemi [70] ile karşılaştırılmıştır. Bahse konu çalışmalarda yer alan veri setleri ve düşük seviye öznitelik vektörleri aynen kullanılmıştır. Performans karşılaştırmasında deneyler tüm algoritmalar için 30 defa tekrarlanarak sınıfların başarı yüzdelerinin ortalaması esas alınmıştır. İmge sınıflandırması iki veri seti üzerinde yapılmıştır. 2688 imge ve 8 sınıftan oluşan Outdoor Scene Recognition (OSR) veri seti için 'Gist' adı verilen 512 boyutlu düşük seviye öznitelik vektörü kullanılmıştır. 800 imge ve 8 sınıftan oluşan Public Figure Face (PubFig) veri setinde ise Gist'e ilave olarak 30 boyutlu 'LaB' renk histogramı kullanılmıştır. Verisetlerine ilişkin özet Şekil 3.17'de verilmiştir.



Şekil 3.17. Denetimsiz göreceli nitelik çıkarımı verisetleri özeti



Şekil 3.18. Denetimsiz göreceli nitelik çıkarımı deneysel tasarım detayı

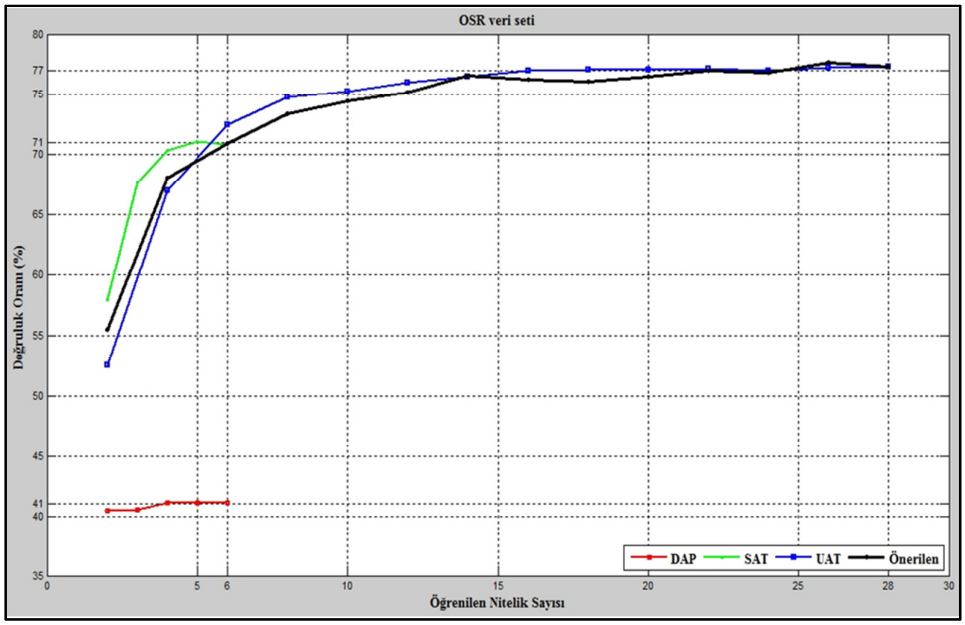
Göreceli niteliklerin öğrenilmesinden sonraki aşama olan imge sınıflandırmada, Radyal Taban Fonksiyonlu (Radial Basis Function-RBF) DVM [90] kullanılmıştır. DVM ve RBF'in doğasında yer alan sabitler (C cezalandırma sabiti ile σ ölçek faktörü) grid arama yöntemi ile optimize edilmiş; 10 tekrarlı çapraz doğrulama uygulanmıştır. Her iki veri seti için bir imge sınıfından 30 adet imge eğitim için ayrılmış; kalanı ise test aşamasında kullanılmıştır. Deney tasarımına yönelik detaylar Şekil 3.18'de verilmiştir.

3.3.1. Deneysel bulgular

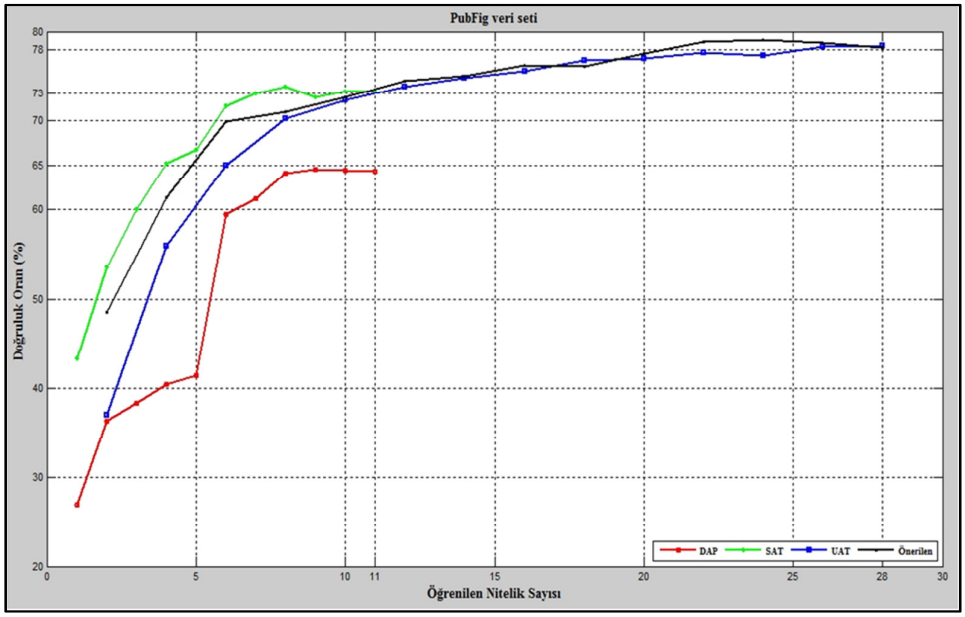
Önerilen çalışmanın DAP, SAT ve UAT yöntemleri ile performans karşılaştırmaları OSR ve PubFig verisetleri için sırasıyla Şekil 3.19 ve 3.20'de gösterilmiştir. DAP ve SAT yöntemlerinde nitelikler denetimli öğrenildiğinden, deneylerde OSR veri seti için 6, PubFig veri seti için 11 adet, daha önceden belirlenmiş nitelik kullanılmıştır [14]. UAT ve önerilen yaklaşımlarında ise, 28 adet göreceli nitelik denetimsiz olarak öğrenilmiştir. Ayrıca; performans karşılaştırmalarında öğrenilen nitelik sayısına uygun nitelik betimlemeleri ile 30 farklı deney yapılarak, ortalama sınıflandırma başarı değerleri kullanılmıştır.

Şekil 3.19 ve 3.20'yi detaylı bir şekilde inceleyecek olursak; niteliklerin kategorik (var-yok şeklinde ikili) öğrenildiği DAP yöntemindeki başarı oranının niteliklerin göreceli öğrenildiği diğer çalışmalara oranla oldukça düşük seviyelerde kaldığı görülmüştür. Diğer bir husus niteliklerin denetimsiz öğrenildiği (UAT ve Önerilen) çalışmalardaki başarı oranının denetimli çalışmalardan (SAT ve DAP) daha yüksek olmasıdır. Deneklerin tecrübesi ile doğru orantılı olduğu için öznel, emek yoğun olması sebebiyle de maliyeti yüksek olan denetimli çalışmalarda nitelik sayısında (OSR'de 6, PubFig'de 11) kısıtlama olmasına karşın; denetimsiz yaklaşımlarda az maliyetle nitelik uzayı genişletilerek başarı oranları etkili biçimde yükseltilmiştir.

Denetimsiz nitelik öğrenme çalışmaları karşılaştırıldığında; az sayıda (sadece 5-10) nitelik kullanıldığında önerilen yaklaşımın başarı oranı UAT yönetimini geçmiş, nitelik sayısı arttıkça UAT önerilen yaklaşıma zaman zaman üstün gelse de 20 nitelikten sonra her iki yaklaşım paralel bir performans sergilemiştir. Bu mukayesedeki dalgalanmanın nedeninin imge sınıflandırmada kullanılan nitelik sayısı düştükçe kombinasyonların artması ve bu kombinasyonlardan en iyisi yerine sadece 10 tanesinin rastgele seçilerek başarı oranının ortalama kabul edilmesi olduğu değerlendirilmiştir.



Şekil 3.19. OSR veriseti üzerinde performans karşılaştırmaları



Şekil 3.20. PubFig veriseti üzerinde performans karşılaştırmaları

Önerilen yaklaşımda önemli bir katkı olarak ayrıca [70]'den farklı biçimde göreceli niteliklerin örnek tabanlı öğrenilmesinden hareketle, önerilen yaklaşımdaki her yineleme aşamasında hesaplanan ara ağırlık vektörlerinin de sınıflandırmaya dahil edilmesidir. Bu sayede nitelik uzayı 196'ya (28 nitelik için 7 yinelemedeki ara değerler) kadar çıkartılmıştır. Bu şekilde ilave az bir hesap yükü ile her iki veri seti

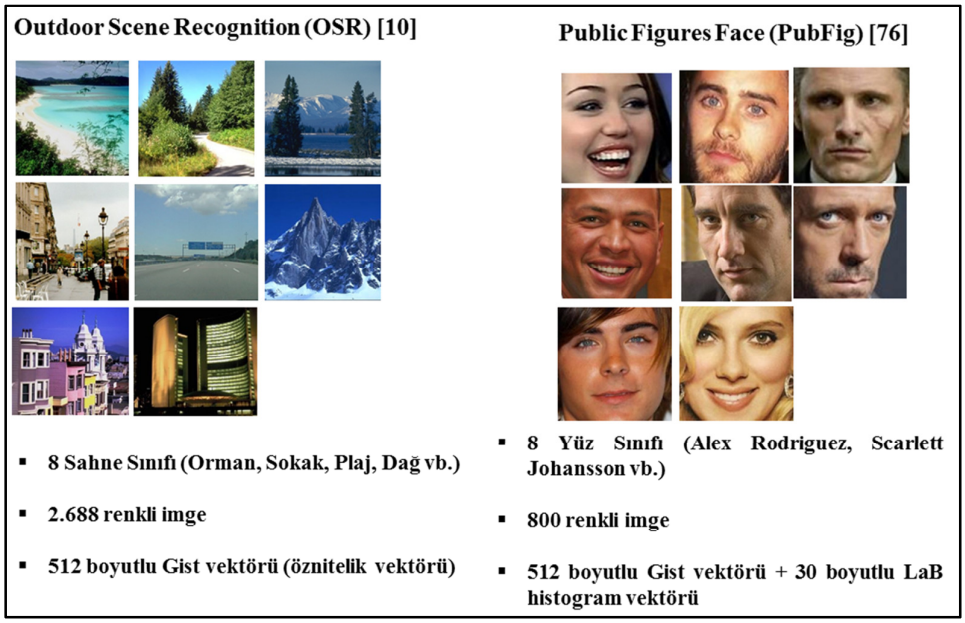
için sınıflandırma başarı oranında yaklaşık % 2 artış (OSR ~ % 79, PubFig ~ % 81) elde edilmiştir.

3.4. Görsel Tanımada Artırımlı Öğrenme Deneysel Tasarımı

Son olarak, Bölüm 2.4'te detayları verilmiş ve Bölüm 2.3'te önerilen denetimsiz göreceli nitelik çıkarımını tamamlayıcı olmak üzere; göreceli nitelik uzaylarının görüntü sınıf modelleri ile eş zamanlı ve arttırımlı öğrenildiği algoritma analiz edilmiştir. Hatırlanacağı gibi, bu sistemde gerçek hayata uygun olarak tasarlanan bir öğrenci-öğretmen makine etkileşimi ile görüntü tanıma yapısı hedeflenmiş ve bu yapının çatısını ikili arasındaki iletişim kanalı olarak adlandırılabilecek nitelikler oluşturmuştur.

Önerilen metodu diğer nitelik tabanlı çalışmalar ile birlikte değerlendirebilmek maksadıyla literatürde sıkça kullanılan 2 farklı veriseti deneylerde kullanılmıştır: Outdoor Scene Recognition (OSR) [14, 69, 70] içerisinde 2688 renkli imge ve 8 sahne sınıfı, ve Public Figure Face (PUBFIG) [14, 77, 78] içerisinde 800 renkli imge ve 8 yüz sınıfı mevcuttur. Ayrıca, veri setleri ile birlikte verilen düşük seviye öznitelik vektörleri de (512 boyutlu Gist ve PUBFIG için ilaveten 30 boyutlu (toplamda 542) LaB renk histogramları) aynen kullanılmıştır. Deneyler esnasında metodların mukayesesini, genellenebilirliğini ve tekrarlanabilirliğini sağlamak maksadıyla, tecrübeler otuzar kere tekrarlanmış ve sayısal değerler bu tekrarların ortalaması dikkate alınarak hesaplanmıştır. Deneylerde kullanılan verisetlerine ait özet Şekil 3.21'de sunulmuştur.

Her tekrarda sınıfların her biri için Sset'te 15, Uset'te ise 25'er adet örnek imge bulundurulmuş; kalanı test set olarak ayrılmıştır. Bu eğitim/test set ayrışmaları her tekrarda yenilenmiştir. Sistem serbest parametrelerinin (SVM için C sabiti ve RBF kernel için σ ölçekleme faktörü) grid arama metodu ve 10-katlı çapraz validasyon yöntemi ile optimize edildiği Vset ise Sset+Uset'ten rastgele seçilen imgelere belirli oranlarda gürültü eklenmesi sonucunda oluşturulmuştur. Sınıf modellerinin hesaplanmasında LibSVM [90] algoritması kullanılmış, SVM'i doğrusallıktan kurtarmak için ise RBF kernel fonksiyonu uygulanmıştır.



Şekil 3.21. Görsel tanımada artırımlı öğrenme için kullanılan verisetleri

Nitelikler üzerinden yöntem karşılaştırmalarını sağlamak üzere her çalışmada göreceli nitelikler kullanılmış ve bu göreceli niteliklerin hesaplanmasında Newton metodu [14] standart olarak koşulmuştur. Dolayısı ile ikili nitelikler ile göreceli niteliklerin karşılaştırılması ayrıca yapılmamıştır. Zira daha önceki çalışmalarda [14, 69, 70, 77, 78] temsil uzayını çok daha fazla genişlettiği ve öğreniminin daha kolay olması nedenleriyle göreceli niteliklerin ikili niteliklere üstünlük sağladığı zaten gösterilmiştir.

Göreceli niteliklerin eldesinde düşük seviye öznitelik vektörlerine ilaveten veri setleri ile birlikte verilen ve daha önce denetimli bir şekilde belirlenmiş sınıf tabanlı sıralamalar (OSR için 6, PUBFIG için 11 adet isimlendirilmiş [14]) tecrübelerde kullanılnıştır. Denetimsiz göreceli nitelikler ise tamamen rastgele sınıf sıralamaları ile elde edilmiş olup tek koşul daha önceki sınıf sıralamaları ile belirli oranda farklı olmasıdır. Rastgele nitelik üretmedeki kasıt ilave iş gücü olmaksızın denetimlerin öğrenilmesidir. Esasında bu çalışmanın esas hedefi nitelikleri en iyi şekilde öğrenmek değil, görsel sınıflandırmadaki başarıyı arttırmaktır. Dolayısı ile en iyi sınıf sıralamasını yakalamak yerine, birbirlerinden belirli oranda farklı ancak isimlendirilmemiş nitelikleri ayırt edici şekilde sınıflandırmada kullanmak suretiyle başarının yükselip yükselmediğini gözlemlemek deneylerin çatısını oluşturmuştur. Bununla birlikte, göreceli nitelikler rastgele üretildiğinde ortaya sınıf sayısının

faktöriyeli çarpanında ihtimal çıktığından, en iyi sıralamayı bulmak diğer bir çalışmanın hedefi olabilir.

Önerilen algoritmanın karşılaştırılması, özü itibariyle iki çatı nitelik tabanlı görüntü tanıma metodu ile birlikte yapılmıştır: Doğrudan Nitelik Tahminlemesi (DAP – Direct Attribute Prediction) ve Yarı Denetimli Nitelik Sınıflaması (SSAC – Semi Supervised Attribute Classification). DAP yöntemi iki seviyeli bir öğrenme yaklaşımı olup nitelikler uzayı ile sınıf modellerinin birbirinden bağımsız öğrenilmesi prensibine dayanmaktadır. Bu kapsamda deneylerde Sset ve Uset eğitim seti olarak birleştirilerek birlikte kullanılmıştır. Birleşik veri seti üzerinde öncelikle göreceli nitelikler (denetimli ve denetimsiz) eğitilmiş ve nitelikler uzayına taşınan eğitim seti bu defa yeni temsil uzayında kategorik modellenmiştir.

SSAC ise önerilen algoritma ile DAP arasındaki geçiş metodolojisini temsil etmektedir. Bu yöntemde tıpkı önerilen yaklaşım gibi Sset, Uset ve test seti ayrı amaçlarla kullanılmıştır. Ancak, DAP gibi nitelikler uzayının eğitiminde Sset+Uset birleşik olarak kullanılmış ve sınıfların modellenmesi esnasında nitelik ağırlık vektörleri devamlı sabit tutulmuş; güncellenmemiştir. S başlangıçta Sset'i ve nitelikler uzayını kullanarak sınıf modellerini ilklendirmiş ve her yinelemede kendisinin sınıf etiketlerini bilmediği Uset içerisinden bir imgeyi seçmiştir. Seçim esnasında en güvenilir (minimum entropi ile maksimum olasılıklı) olan örneği belirlemiş ve Sset içerisine tahmin ettiği sınıf etiketi ile birlikte bu örneği atmış, Uset'ten eksiltme yapmıştır.

Bu noktada SSAC önerilen yaklaşımdan ayrılmaktadır. Çünkü önerilen yaklaşımda mevcut T makinesi S'i nitelik tabanlı açıklamaları ile yönlendirmekte ve sorgulanan örnekler üzerinden nitelik uzayları ile sınıf modelerin eş zamanlı güncellenmesini sağlamaktadır. Oysa SSAC metodunda S sadece tahminlerine güvenmekte ve tahminlediği sınıf etiketleri ile Sset'i her yinelemede genişletmektedir. Sorgu imgesinin Sset'e atılmasını müteakip S sınıf modellerini güncellenmekte ve bu iteratif yaklaşım Uset'in boşaltılmasına kadar devam etmektedir.

Son olarak, deneylerdeki tekrarlarda bahse konu üç ana yöntemin adil şekilde karşılaştırılması için eğitim/test setleri, üretilen rastgele göreceli nitelikler, göreceli nitelik eğitiminde kullanılan Newton sıralama fonksiyonu, sınıf modellemede

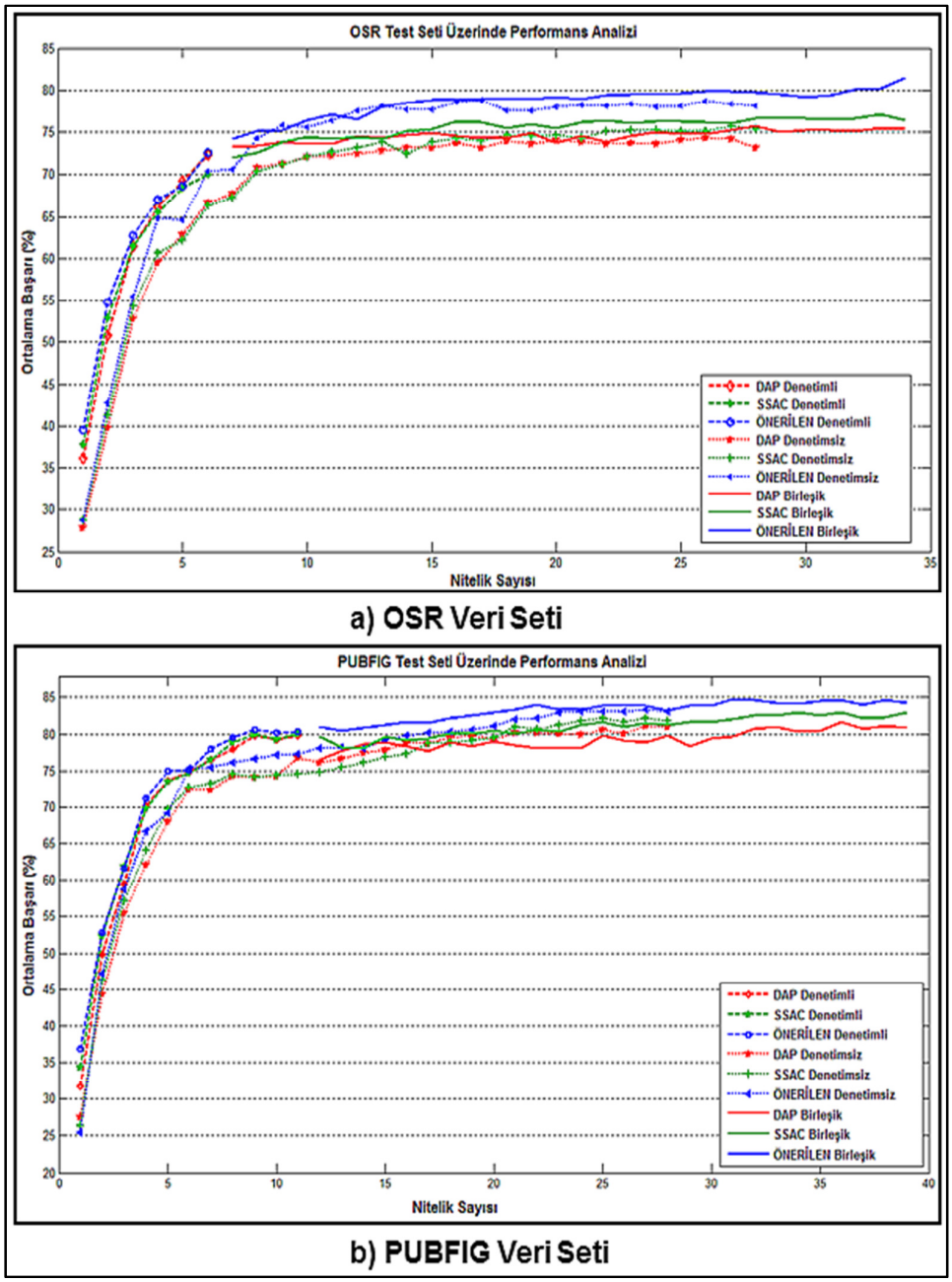
kullanılan SVM algoritması ve girid arama ile çapraz validasyon aynı şekilde kullanılmıştır. SVM fonksiyonu girişi öncesinde RBF kernel çıkışları 0 ortalama ve 1 standart değişinti ölçütünde normalize edilmiştir.

3.4.1. Deneysel bulgular

Çok sınıflı görüntü tanıma probleminin çözümünde başlangıç sorularımız: Acaba görsel temsilde göreceli nitelikler orta kademe temsil uzayı olarak ne kadar etkin, denetimsiz elde edilen göreceli nitelikler emek yoğun bir şekilde elde edilen denetimli niteliklere göre ne derece faydalı, ve genelleme açısından yarı denetimli öğrenme (SSAC) veya direk niteliklerle sınıflandırmaya (DAP) göre sınıf modellerinin nitelik uzaylarıyla eş zamanlı ve artırımlı öğrenildiği bir makine-makine etkileşim ortamında başarı performansı nasıl değişmektedir.

Daha önceden bahsedildiği gibi, deneyler 30'ar tekrar ve her üç yöntem için aynı koşullar (göreceli nitelikler ve eğitim/test veri setleri) üzerine kurgulanmış olup plotlamada tekrarlardan elde edilen ortalama değerler yüzdesel olarak sergilenmiştir. Ayrıca her üç yöntem için de veri setleri ile birlikte sunulan [14] denetimli göreceli nitelikler, rastgele elde edilen nitelikler [77] ve her ikisinin birleşimden çıkan toplu nitelikler ayrı ayrı değerlendirilmiştir.

Bu kapsamda OSR ve PUBFIG veri setleri için değişik nitelik sayılarına göre ortalama başarı yüzdelerinin plotlaması Şekil 3.22'de gösterilmiştir. Sadece denetimli nitelikler üzerinden algoritmaların karşılaştırması kapsamında; her ne kadar önerilen metod ile SSAC yöntemi DAP'a göre daha iyi bir başlangıç yapsa da; neticede her üç yöntem de benzer bir performans sergilemiş ve OSR'de maksimum % 73, PUBFIG'de % 79 civarında doğruluğa erişmiştir. Bunun nedenlerinin denetim sonucu elde edilen nitelik sayısının çok sınırlı (OSR'de 6, PUBFIG'de 11 adet) ve göreceli nitelikler üzerindeki sınıf sıralamalarının birbirlerine çok yakın olduğu değerlendirilmiştir.



Şekil 3.22. Değişen miktarda nitelik ile başarı performansı: a) OSR, b) PUBFIG

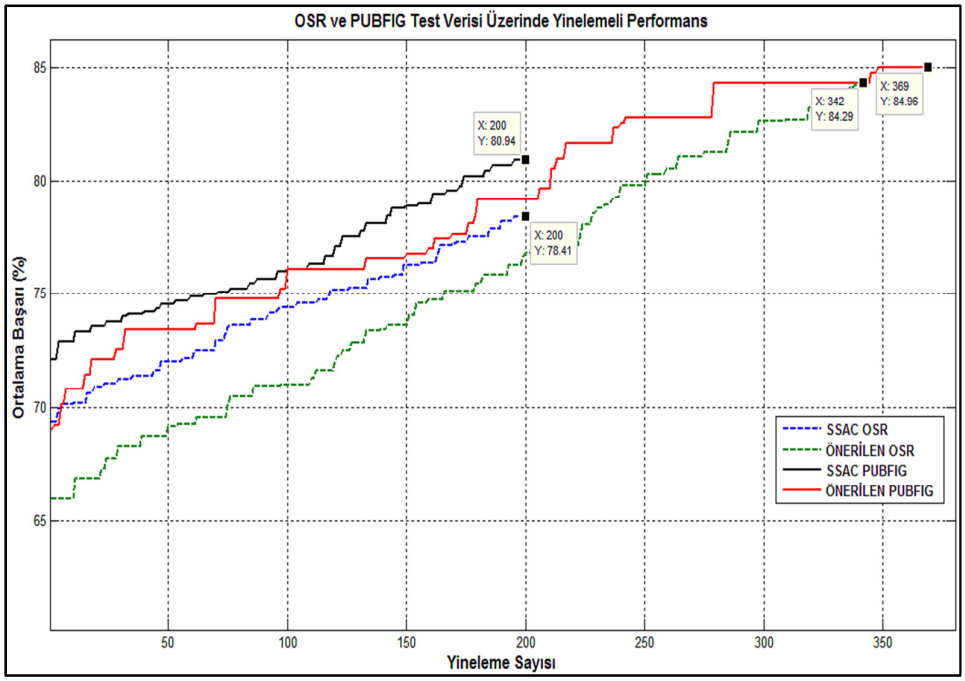
Denetimsiz nitelikler ile sınıflandırma kapsamında elde edilen sonuçların diğer benzer literatür çalışmaları ile mukayese edilebilmesi maksadıyla 28 adet rastgele sınıf sıralamalı nitelik öğrenilmiştir. Oysa sınıf sayısının faktöriyeli oranında türetilebilirdi. DAP ve SSAC metodlarının başlangıçta denetimli niteliklere göre başarısı çok düşük olsa da, belirli bir seviyeden sonra (OSR için 11-12, PUBFIG için 18-20) performanslarının az da olsa arttığı gözlenmiştir. Ortalama olarak DAP

denetimsizde % 1-2 civarında, SSAC'ta ise % 3-4 civarında artış sağlamıştır ki bu da denetimsiz elde edilmesine rağmen sayı üstünlüğünden dolayı rastgele niteliklerin daha iyi sonuçlar sergilediği şeklinde yorumlanabilir. Önerilen metodun ise hepsinden açık ara bir performans gösterdiği ve denetimlilere göre 28 denetimsiz nitelik ile % 6-8'lik bir artış sağladığı görülmüştür.

Benzer sonuç denetimli ve denetimsiz tüm niteliklerin kullanıldığı (34 adet) deneylerde de tespit edilmiştir. Bütün nitelikler bir arada kullanıldığında ayrı ayrı olduklarından çok daha iyi başarı yüzdeleri ile karşılaşılmış ve en az % 2'lik bir artışla OSR'de % 82, PUBFIG'de ise % 85 seviyelerine çıkıldığı gözlemlenmiştir. Denetimsiz nitelikler ile yapılan deneyler sonucunda plotlarda oluşan dalgalanma niteliklerin rastgele sıralanma ile elde edilmesinden ve en iyi nitelik seçimi gibi optimizasyon yöntemlerinin kullanılmayışından kaynaklanmaktadır. Sonuç olarak; denetimsiz niteliklerin temelde sayıca üstün ve ayırt edici olmaları nedeni ile denetimli niteliklere göre daha iyi sonuçlar elde ettiği, yarı denetimde görüldüğü üzere tüm örnekleri tek seferde kullanmak yerine artırımlı bir şekilde sınıf modelleri öğrenilmesinin başarı performansını artıracağı ve nitelik uzayının da benzer şekilde artırımlı ve sınıf modelleri ile eş zamanlı güncellenmesinin görüntü tanıma için yerinde bir karar olduğu tespit edilmiştir.

SSAC ile önerilen metodun yinelemeli bir şekilde sınıf modellerini öğrenmesi benzerliğinden hareketle, her iki yöntemin iterasyonlara göre başarı performansı OSR ve PUBFIG verisetleri üzerinde Şekil 3.23'te gösterilmiştir. Daha önceden bahsedildiği üzere SSAC metodunda başlangıçta göreceli nitelikler Sset+Uset veri setinde öğrenilmiş ve sınıf modellerinin öğrenilmesi müddetince sabit tutulmuştur. Oysa önerilen algoritmada makine-makine etkileşimi esnasında T'den gelen nitelik tabanlı açıklamalara istinaden S hem nitelikler uzayını hem de sınıf modellerini eş zamanlı öğrenmiştir. Plotlama esnasında yöntemler için ortak olarak aynı nitelikler (OSR için 34, PUBFIG için 39) ve veri setleri üzerindeki performansları kullanılmıştır. SSAC metodunda Uset'ten her yinelemede 1 adet örnek, $x^{(q)}$, alınarak tahminlenen sınıf etiketi ile Sset'e aktarım yapıldığı için, iterasyon sayısı Uset başlangıç örnek sayısı ile (8 sınıftan 25'er adet olmak üzere toplamda 200 adet) sınırlanmıştır. Oysa önerilen algoritmada S tarafından yanlış tahminleme

yapıldığında (ve her 20 iterasyonda bir Sset'teki yanlış tahminlenen bir örnek) Uset'e geri iade edildiği için iterasyon sayısı değişkendir.

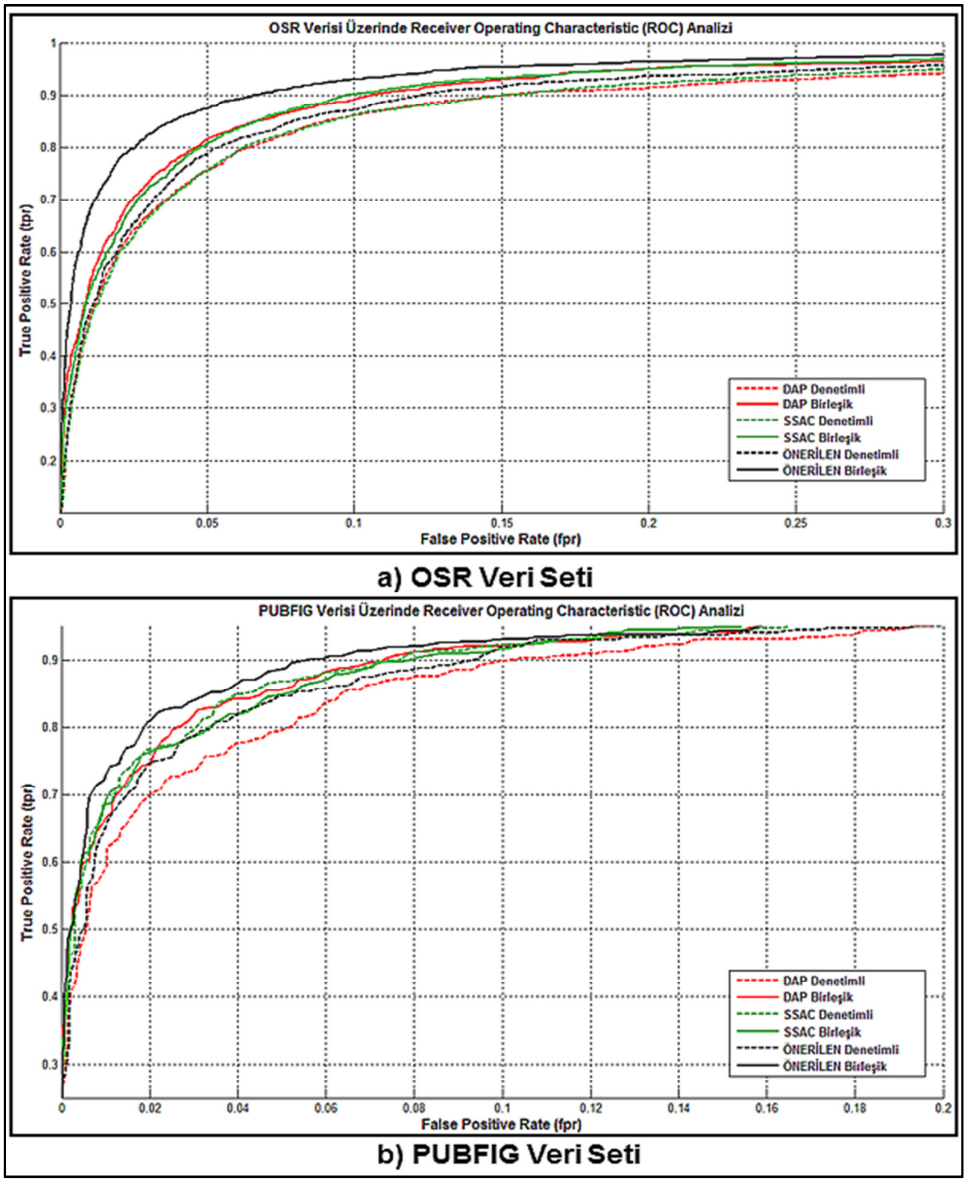


Şekil 3.23. Verisetleri üzerinde yineleme oranına göre performans analizi

Şekil 3.23 incelendiğinde, SSAC'ın % 2-4 civarında daha iyi bir başlangıç yaptığı gözlenmiştir. Bunun nedeninin nitelik uzayının başlangıçta Sset+Uset kullanılarak daha geniş bir veri seti üzerinde oluşturulmasından kaynaklandığı değerlendirilmiştir. Ayrıca SSAC yönteminin iterasyonlar ilerlerken elde ettiği başarı nisbeten daha dengeli bir eğilim izlemiştir. Çünkü nitelikler uzayı geniş bir veri setinde öğrenilmiş ve sınıf modellerinin öğrenilmesi müddetince sabit tutulmuştur. Diğer taraftan önerilen metod her ne kadar kötü bir başlangıç yapsa da, ki bunun nedeni niteliklerin S tarafından sadece Sset üzerinde ilklendirilmesidir, neticede çok daha iyi bir performans sergilemiştir. Örneğin, önerilen yöntem PUBFIG veriseti üzerinde 230. iterasyon civarında SSAC'ı yakalamış ve yaklaşık % 4 civarında fazla başarı elde etmiştir.

Önerilen yöntemin plotlarındaki dalgalanma ise niteliklerin de sınıf modelleri ile birlikte öğrenilmesi, yani devamlı nitelik uzaylarının değişken olmasıdır. İlaveten, hatırlanacağı üzere sınıf modellerinin öğrenilmesi esnasında S yinelemelerde geçici olarak Uset'ten bazı aday örnekleri kullanmıştır. S kendi tahminlerine göre aday

örnekleri etiketlediği için başlangıç koşullarında Sset'in sınırlı sayıdaki örnekleri ile aday örnekler birleştirildiğinde, bahse konu adaylar başarı oranında dalgalanmaya sebep olmuştur. Ancak bu dalgalanma sonradan daha aza indirgenmiştir. Bunun nedeninin zamanla Sset'in genişlemesi ve nitelikler uzayının da dengeli hal alması olduğu değerlendirilmiştir.



Şekil 3.24. Verisetleri üzerinde karşılaştırmalı ROC analizleri: a) OSR, b) PUBFIG

Receiver Operating Characteristic (ROC) eğrileri literatürde sıkça kullanılan ve sınıflandırıcıların başarı performanslarını ölçen bir plotlama yöntemidir. Bu noktada

belirli eşik değerlerine göre Yanlış Pozitif (YP) (False Pozitif –FP) ile Doğru Pozitif (DP) (True Positive –TP) oranlarının plotlaması yapılır. Bir sınıflandırıcının eğrisi ne kadar erken ve dik bir şekilde yükselirse o sınıflandırıcının başarısının daha iyi olduğu anlaşılır.

Her üç algoritmanın elde ettiği sınıflandırıcıların tutarlılık analizini yapmak maksadıyla, denetimli ve tüm nitelikler birlikte kullanılarak oluşturulmuş ROC plotları Şekil 3.24'te sunulmuştur. Tüm nitelikler birlikte kullanıldığında başarı oranında dikkate değer bir artış yakalanmıştır. Denetimsiz olarak elde edilen niteliklerin denetimli olanlara ilavesinin daha ayırt edici temsil uzayları oluşturduğu gözlenmiştir. Bu tespit önceki bulgularla uyuşmaktadır. Ayrıca, önerilen metodun her iki veriseti üzerinde de sırasıyla SSAC ve DAP yaklaşımlarına göre daha başarılı olduğu sonucuna varılmıştır.

Son olarak, önerilen metod aynı veri setlerinde literatürdeki diğer göreceli nitelik tabanlı yaklaşımlarla ortalama başarı yüzdeleri üzerinden mükayese edilmiş ve sonuçlar Tablo 3.11'de gösterilmiştir. Deneyler 30'ar defa tekrarlanmış ve ortalama değerler not edilmiştir. Tabloda yer alan Binary (BINs), Principal Component Analysis (PCA) ve Fisher Linear Discriminant (FLD) yöntemleri esas itibariyle boyut indirgemede kullanılan referans metodlar olup doğrudan nitelikler ile bağlantılı değildir. Ancak bahse konu algoritmalardan elde edilen temel vektörler yeni öznitelik uzaylarının tasarımında kullanıldığı için, deneylerde referans olarak alınmıştır. Diğer yöntemler ise görüntü verisini orta seviyede temsile yarayan ve denetimli/denetimsiz biçimde çıkartılan göreceli niteliklerin kullanıldığı metodlardır.

Sonuçlar incelendiğinde; [70]'ten farklı olarak niteliklerin tamamen rastgele elde edilmesine ve [16]'dan farklı olarak herhangi bir nitelik seçimi yapılmamasına rağmen, önerilen metodun en az yaklaşık % 3 oranında diğerlerinden daha iyi olduğu gözlenmiştir. Bu kayda değer artışın göreceli nitelik uzayları ile sınıf modellerinin eş zamanlı öğrenilmesine ve örnek sayısının sanal olarak arttırılmasına dayandığı sonucuna varılmıştır.

Özellikle nitelik tabanlı sınıflandırıcıların referans metodlara göre daha iyi olduğu gözlenmiştir. Dolayısı ile niteliklerin sadece boyut indirgemediği, aynı zamanda orta seviyede daha ayırt edici bir temsil uzayı oluşturduğu söylenebilir. Diğer taraftan,

deneylerde denetimsiz elde edilen niteliklerin denetimli öğrenilenlere göre daha iyi performans elde ettiği görülmüştür. Sonuç olarak, denetimsiz elde edilen niteliklerin sayısa fazlalığı ve sınıf tabanlı nitelik öğrenildiği için sınıf sıralamalarının ayırt edici olması durumunda ilave iş gücüne ihtiyaç duyulmaksızın daha ayırt edici öznitelik uzaylarının elde edileceği ve görüntü tanımada başarının artacağı öngörülmüştür.

Tablo 3.11. Algoritmaların verisetleri üzerindeki performansları

Algoritmalar	**Nitelik Sayısı**	**OSR Başarı (%)**	**PUBFIG Başarı (%)**
BINs [70]	28-28	76,05	72,32
PCA [70]	34-39	71,46	64,28
FLD [70]	28-28	63,10	60,71
SAT [14]	6-11	72,82	74,40
UAT [70]	28-28	76,57	78,60
SAT+UAT [70]	34-39	77,88	78,83
Rastgele [77]	34-39	78,64	81,25
Önerilen SAT	6-11	72,56	77,21
Önerilen UAT	28-28	78,08	81,90
Önerilen SAT+UAT	34-39	**81,51**	**84,05**

4. SONUÇLAR VE ÖNERİLER

Bu tezin amacı, görüntü verisinin içeriğine uygun olarak tanımlanmasını sağlayacak yeni bir makine öğrenme algoritması geliştirmektir. Görüntü tanıma problemine yönelik literatür çalışmalarının temelde yerel veya bütünsel düşük seviye öznitelik vektörlerinin çıkartılması, orta veya ileri seviye imge temsili ve sınıflandırıcı metodlarının uygulanma usulleri bakımından ayrıştığı görülmüştür. Bu kapsamda İmgenin nasıl temsil edileceğinin belirlenmesinin sınıflandırma başarı performansını etkileyen en önemli husus olduğu aşikardır. Sınıflandırma işlemi temsil uzayında yapılacağından, uygulamaya dönük en ayırt edici betimleyicilerin kullanılması gereklidir. İmge temsilinin ise temelde yerel veya bütünsel olarak imge üzerinden çıkartılan düşük seviye öznitelik betimleyicileri ile anlamsal modelleme olarak ikiye ayrıldığı tespit edilmiştir. Tüm bu sayılan hususlar göz önünde bulundurularak ölçeklenebilir, optimum denetimli, istenilen başarı yüzdesini sağlayacak ölçüde etkin ve hafıza/işlem yükü az olan bir görüntü tanıma algoritmasının literatürdeki tüm çalışmaların hedefini oluşturduğu söylenebilir.

Tez çalışması neticesinde, bir görüntü içeriğindeki nesne ve nitelikleri dinamik ve çok kriterli ortamda etkin bir şekilde tanıyan sınıflandırma sistemi geliştirilmiştir. Sadece eğitim aşamasında karşılaşılan sahne/nesne kategorilerinin analiz edilmesi yerine, eğitim aşamasında gözlemlenmeyen ve sadece test aşamasında sisteme sokulacak yeni kategorilerin de öğrenilecek göreceli nitelikler yardımı ile tanınması sağlanmıştır. Göreceli nitelikler yardımıyla daha geniş bir ifade özgürlüğüne kavuşulacağından insan-makine etkileşiminde takviyeli öğrenmenin görüntü analizine yansımaları ayrıca değerlendirilmiştir.

Görüntü tanımaya/sınıflandırmaya yönelik tez kapsamında bulunan çözüm önerilerinin uygulama sahası olarak yüksek çözünürlüklü uydu görüntülerinin, kızılötesi görüntülerin, medikal görüntülerin, istihbarat maksatlı kullanılan hava araçlarından anlık olarak iletilen görüntülerin ve sualtı ses sinyallerinin analizi kapsamında da kullanılabileceği değerlendirilmektedir.

4.1. Elde Edilen Sonuçlar

Tez kapsamında görüntü tanıma probleminin çözümünde temelde iki özgün yöntem sunulmuştur. İlk aşamada görüntü tanıma probleminde sıkça kullanılan görsel kelimeler sözlüğünü denetimsiz ve hiyerarşik şekilde elde ederek ham veriyi öncelikle düşük seviye ve boyutta yeni bir öznitelik uzayına taşımak amacıyla derin sinir ağı yapısı kullanmıştır. Hiyerarşik görsel kelimeler uzayını destekleyerek görsel kelimeleri elde etmek amacıyla önerilen öbekleme tabanlı melez ağ mimarisi Bölüm 2.1'de anlatılmıştır.

Ayrıca ikinci aşamada, öğrenci-öğretmen etkileşimli bir yapı içerisinde başlangıçta denetimsiz (rastgele) olarak oluşturulan kategori tabanlı göreceli nitelik uzaylarını sınıf modellerine uygun şekilde ve eş zamanlı güncelleyecek özgün bir yöntem geliştirilmiştir. Göreceli niteliklerin denetimsiz çıkarımına ilişkin tez kapsamında yapılan bir çalışma Bölüm 2.3'te ayrıca sunulmuştur. Daha sonra, hedef kategoriler yarı denetimli olarak etkileşimli ve arttırımlı bir yapıda göreceli niteliklerle birlikte öğrenilerek görüntü sınıfları modellenmiştir.

4.1.1. Melez sinir ağları mimarisinde öbekleme

Tez çalışmasının bu kısmında, öbekleme probleminin çözümüne esas teşkil etmek üzere birbirini tamamlayan iki ağ mimarisinin birlikte kullanıldığı kompozit çok katmanlı bir sinir ağı mimarisi önerilmiştir. PKA olarak adlandırılan ilk aşamadaki üç katmanlı ağ yapısında, giriş verisi yinelemeli olarak çıkış katmanında en yakın prototip vektörü ile haritalanmaya çalışılmıştır. Ağ ağırlık parametreleri optimize edilirken PKA'nın gizli katman çıkışında denetimsiz bir şekilde yeni bir öznitelik uzayı elde edilmiştir. Orjinal giriş uzayının kullanıldığı literatürdeki diğer ağ tabanlı öbekleme metodlarının [48, 51, 53, 55] aksine, öbek güncelleme işlemi bahse konu yeni öznitelik uzayında yürütülmüştür. Bununla birlikte; PKA'nın kodlama ağırlıklarını (W_1) temsil eden ve önerilen yöntemin doğası gereği esasında öbekler arasındaki sınırları oluşturan düzlemler, rastgele seçilmek yerine Voronoi diyagramlarındaki gibi prototip vektörlerin ikili eşleştirilmesi suretiyle daha uygun şekilde ilklendirilmiştir. Bu sayede gizli katmanda yer alan nöron sayısı da rastgele başlatılan prototip vektörlerine göre dinamik olarak belirlenmiştir. İlaveten, her bir örnek için sadece kazanan (en yakın) prototip vektörünü güncellemektense SOM gibi

Gaussian komşuluk fonksiyonu kullanılarak kazanan prototipin belirli bir komşuluğundaki tüm prototip vektörleri mesafeleri doğrultusunda bir ağırlık ölçütünde birlikte güncellenmiştir. Bu hususun deneysel sonuçlardaki performansı pozitif yönde etkilediği gözlenmiştir.

Çok katmanlı sinir ağı mimarisinin ikinci aşamasında yer alan ve iki katmandan oluşan DVMA mimarisi PKA'dan elde ettiği öncül öbekleme bilgisini kullanarak mevcut öbekleri iyileştirici yönde güncellemeye çalışmıştır. Bu kapsamda öbekler arasındaki ayırdımı maksimize etmek maksadıyla amaç fonksiyonunda ikinci derece Hinge Loss fonksiyonu kısıtsız bir şekilde kullanılmıştır. Tıpkı PKA gibi, DVMA da öbek güncelleme prosedürünü çıkış katmanındaki aktivasyonların temsil ettiği yeni öznitelik uzayında yinelemeli bir şekilde icra etmiştir. DVMA'daki diğer önemli bir uygulama detayı kernel fonksiyonunda görülmektedir. DVM'nin doğrusal ayırt edicilik doğasını karmaşık hale getirmektense, kernel fonksiyonları DVM'ye girdi sinyalini doğrusal olmayan denklemleri sayesinde ilk elden daha karmaşık hale getirmiştir. Böylelik doğrusal olmayan sınıflama problemlerinin DVM ile çözümüne büyük katkı sağlamıştır. Bu düşünce önerilen ağ mimarilerine de uygun olup Gaussian Kernel Fonksiyonunda prototip vektörleri kerteriz noktaları olarak kullanılmıştır. Böylece bütün girdi örneklerinin kernele kerteriz olarak sokulduğu geleneksel uygulamadan kaçınılarak nitelik boyutlarının örnek sayısına bağlı olarak çoğalması, değişintinin artması ve hesaplama maliyetinin yükselmesi önlenmiştir. Ayrıca orjinal girdi uzayı yerine verinin PKA ağındaki gizli katman aktivasyon değerleri GKF'ye ardışık biçimde sokularak her iki ağ mimarisinin uyum içerisinde çalışması sağlanmıştır.

Önerilen çok katmanlı melez ağ mimarisi literatürde yer alan öbekleme tabanlı benzer ağ yapılarıyla farklı değerlendirme ölçütlerinde karşılaştırılmıştır. Her bir performans spektrumunda deneysel sonuçları analiz etmek üzere, detayları Bölüm 3.1'de verilen internet ortamındaki birbirinden farklı veri depolarından (UCI machine learning, Speech and Image Processing) toplamda dokuz verisetinden yararlanılmıştır. Sentetik iki boyutlu verisetleri temelde görsel analizlerde kullanılırken; gerçek dünya UCI verisetleri analitik çıkarımlarda kullanılmıştır. Deneylerden elde edilen hem görsel hem de analitik sonuçlar PKA+DVMA ağ

mimarisinin aynı deney koşullarında diğer algoritmalardan daha başarılı olduğunu göstermiştir.

İlaveten; melez yapının tüm katmanlarındaki nöron sayıları otomatik olarak tespit edilmiş ve ağırlık parametreleri (PKA'daki W_1 hariç) rastgele seçilmiştir. Bu kapsamda önerilen mimarinin parametrik karmaşıklığının benzeri olan SOM ve NG algoritmaları ile aynı olduğu değerlendirilmiştir. Diğer taraftan; önerilen melez ağ mimarisinin veri öbeklemede en yakın rakibine göre iki katı kadar bir sürede işlemleri tamamladığı gözlenmiştir. Bunun nedeninin iki ağ yapısının ardışık olarak koşulması ve ağırlık parametrelerinin yinelemelerde mini paketler halinde güncellenmesi olduğu düşünülmektedir. Ancak önerilen her iki ağ yapısının zamanlama olarak bireysel açıdan benzerleri ile mukayese edilebilir oldukları ve önceliğin öbekleme performansına verildiği durumlarda birlikte kullanılabilecekleri önerilmiştir.

Ayrıca dikkat edilmesi gereken husus PKA ve DVMA ağ yapısında öbeklerin yeni öznitelik uzaylarında güncellenmesidir. Özellikle PKA başlangıç ağında gizli öznitelik uzayı denetimsiz bir şekilde oluşturulmuş ve bunun öbekleme kalitesine doğrudan etkisinin olduğu görülmüştür. Bu nedenle, daha büyük veri setleri kullanıldığı takdirde yanlılık etkisinin azaltılacağı ve daha kaliteli örüntüler tespit edilerek öbekleme performansının arttırılabileceği varsayılmaktadır.

Görsel analiz bölümünün bir işlem maddesinde DVMA ağına ardışık olarak beslenen PKA öbekleme sonuçlarının tatmin edici seviyelerde olmadığı varsayılmıştır. Böylece yetersiz bir öbekleme denetimi ile ilklendirilen bir DVMA yapısının bu durumu ne ölçüde telafi edebileceği deneysel olarak araştırılmıştır. Deney ortamını sağlamak üzere, öncelikle PKA yapısında giriş katmanına giren veriye Gauss gürültüsü eklenmiştir. İkinci olarak; yine PKA yapısında öbek merkezleri öznitelik uzayının aykırı bölgelerinde olmak üzere rastgele ilklendirilmiştir. Deneysel analizlerde DVMA ağ yapısının PKA ağındaki öbekleme sonuçlarına katı bir şekilde bağımlı olmadığı, PKA öbekleme konfigürasyonunu değiştirebileceği gözlenmiştir. İlaveten, yinelemeli öbek güncelleme prosedürü gereğince DVMA mimarisinin daha iyi öbekleme sonuçları elde edebilmek adına ilklendirmeden gelen bazı öbekleri feda edebildiği görülmüştür.

Son olarak; önerilen PKA mimarisinin ara katman aktivasyon çıkışları görsel olarak analiz edilmiştir. Böylece gizli katmanda öğrenilen örüntülerin girdi öznitelik uzayına nasıl bir dönüşüm uyguladığı görülmüştür. Ayrıca; tek ara katmanlı PKA mimaride geliştirilen öbekleme sonuçlarının birden çok katmanlı ve ara katmanlarda farklı miktarda nöron olmak üzere yığıt şekline dönüştürülmesi ele alınmıştır. Bu nedenle, iki PKA yapısı ardışık olarak birleştirilmiş; ilk PKA'nın öğrenilmesiyle elde edilen ara katman aktivasyon çıkışları ikinci PKA yapısına girdi verisi olarak sunulmuştur. Görsel analizlerde PKA mimarilerinin yığıtlar şeklinde birbirine eklenmesiyle ara katmanlarda kodlayıcı vektörel dönüşümler sonucu daha ayırt edici öznitelik uzaylarının elde edildiği ve öbekleme performansının arttığı sonucuna varılmıştır.

İleriye dönük ağ tabanlı öbekleme çalışması açısından, çok katmanlı sinir ağı içerisinde verilen etiketsiz bir veri seti için optimum öbek sayısının bulunması ve rastgele seçmek yerine başlangıç prototip vektörlerinin daha rasyonel bir şekilde ilklendirilmesi hedeflendirilmektedir. Yapay sinir ağları herhangi bir amaç fonksiyonunu simule edebilecek esnek bir konfigürasyon yapısına sokulabilir. Benzer şekilde, ara katmanlarda hiyerarşik yeni öznitelik uzayları elde edilebilir ve fırsatçı katman tabanlı öğrenim yaklaşımı ile her bir katman farklı bir amaç doğrultusunda görevlendirilebilir. Son olarak çok katmanlı ağ yapılarının karmaşık öbekleme problemlerini denetimsiz bir şekilde ve birbirlerine bağlanmış zincir halkaları şeklindeki görev tanımlarına istinaden yüksek başarı performansı ile çözebileceği önerilmektedir.

4.1.2. Derin ağ yapısında hiyerarşik görüntü temsili

Tezin bu kısmında, fırsatçı katman tabanlı öğrenme algoritması ile çok katmanlı yapay sinir ağı mimarisinde düşük seviye denetimsiz öznitelik çıkartımı yardımıyla görüntü tanıma incelenmiştir. Üç ara katmanlı bir ağ yapısı hiyerarşik GKS temsil uzaylarının oluşturulmasında görsel kelimeleri elde etmek için modellenmiştir. Görsel kelimeleri bulmak maksadıyla k-ortalamalar algoritması ağ içerisinde taklit edilmiştir. Beklenti-Maksimize etme aşamalarında yinelemeli bir şekilde ve sırasıyla mevcut katman ağ parametreleri ve görsel kelimeler güncellemiş; güncelleme esnasında büyük miktarda etiketsiz yama örnekleri sisteme beslenmiştir. Yama girdiler ilk ara katman için ham piksel yoğunluk değerleri ile ifade edilirken, ikinci

ve üçüncü katmanlarda daha soyut nitelik haritaları ile betimlenmiştir. Fırsatçı katman tabanlı öğrenme metoduna göre, sonraki katmanların optimizasyonu önceki katman parametrelerinin sabitlenmesiyle gerçeklenmiştir. İlaveten sınıflandırma aşamasında literatürdeki diğer çalışmaların aksine [37, 46, 102, 103], gizli katman aktivasyon çıkışları doğrudan kullanılmamış; öğrenilen görsel kelimeler GKS öznitelik uzaylarına dönüştürülerek sınıflandırma burada modellenmiştir.

Bu bağlamda iki temel avantaj elde edilmiştir. İlk olarak, öznitelik vektör boyutu ile ara katmanlardaki nöron sayısı arasındaki ilişki kırılmıştır. Böylece önerilen yaklaşım sayesinde basit ağ mimarilerinde daha çok boyutlu öznitelik vektörleri etkin bir biçimde tesis edilebilecektir. İkincisi, ağ yapısında elde edilen görsel kelimeler asıl sınıflandırmanın yapıldığı GKS uzaylarını oluşturmak üzere kullanıldığından, GKS öznitelik uzayları aktivasyon çıkışlarına göre daha az kısıtlanmıştır. Ayrıca; geleneksel GKS temsiline ağ mimarisi dahilinde mevki bilgisi eklenmiştir. Mevki bilgisi ikinci ara katman düzeyinde nitelik haritalarının çeyrek dilimlere bölünmesi ve her dilimde ayrı bir ağ yapısının işletilmesiyle elde edilmiştir. Deneylerde, önerilen hiyerarşik GKS algoritmasının aynı verisetleri üzerinde literatürdeki karmaşık yaklaşımlardan daha başarılı olduğu gözlenmiştir.

Tezin bu önerisi kapsamında geleceğe dönük olarak; derin ağ mimarisi içerisinde daha ayırt edici görsel kelimeler elde edilerek görüntü verisinin hiyerarşik bölütlenmesi, fırsatçı katman tabanlı öğrenme prosedürü ile katman ağ parametrelerinin birbirinde bağımsız optimizasyonu sonrasında bütünleşik bir ince ayarlama metodu geliştirilmesi önerilmektedir.

4.1.3. Denetimsiz göreceli nitelik modellemesi

Tezin bu kısmında denetimsiz bir şekilde öğrenilen göreceli nitelikler yardımıyla imge sınıflandırması ele alınmıştır. Göreceli nitelik uzayını temsil eden ağırlık vektörleri, sınıflar arası büyüktür veya benzerdir koşullanmaları kullanılarak DVM benzeri bir algoritma ile belirlenmiştir.

Deneyler sonucunda, niteliklerin denetimli öğrenilmesi aşamasında elde edilen başarı oranının iş yükü ile doğru orantılı ve denetimin sınıf/nitelik sayısına göre ölçeklenebilir olmadığı, denetimsiz yaklaşımlarda düşük işlem maliyeti ile daha iyi

sonuçlara ulaşılabildiği görülmüştür. Denetimsiz şekilde niteliklerin öğrenildiği mevcut çalışmadan [70] farklı olarak; büyüktür yanında benzerdir koşullanması da sınıflar arası göreceli ilişkilerde kullanılmış, yinelemeli yapıda entropi kriteri yerine Bölüm 2.3'te önerilen mesafe hesaplamalarına dayanan yeni bir algoritma önerilmiştir. Son olarak nitelik öğrenmede sınıf imgeleri örnek tabanlı bir yaklaşımla sağlam-çürük şeklinde ayrıştırılmış; daha anlamlı hale getirilen ara nitelik vektörlerinin de imge sınıflandırmada kullanılması neticesinde sınıflandırma performansı yükseltilmiştir.

Tezin bu önerisi kapsamında geleceğe dönük olarak; nitelikler arası göreceli ilişkilerin de (A sınıfı için birinci niteliğin ikinciye göre daha önemli veya benzer önemde olması gibi) belirlenmesi ve örnek tabanlı nitelik öğrenmenin geliştirilmesi sonucunda başarımın arttırılabileceği önerilmiştir.

4.1.4. Göreceli nitelikler ile görüntü tanıma

Tezin son kısmında temelde düşük seviye öznitelikler yerine orta seviyede nitelik tabanlı görsel tanıma üzerinde durulmuştur. Niteliklerin gerek makine gerekse insan tarafından algılanabilen anlamsal kavramlar olduğu ve bu nedenle insan-makine etkileşiminde orta seviye temsil uzayının oluşturulması esnasında son zamanlarda sıkça başvurulan bir çatı yapı olduğu görülmüştür.

Tezin konusuna uygun olarak göreceli niteliklerin denetimsiz ve artırımlı öğrenimi kapsamında, makine-makine etkileşimi temelinde bir algoritma önerilmiştir. Gerçek hayata uygun olarak, öğrenci ve öğretmen arasındaki etkileşimden yola çıkılarak öğrenci merkezli bir öğrenme yöntemi geliştirilmiştir. Öğrenci (S) ve öğretmen (T) arasındaki iletişim kanalı göreceli nitelikler üzerinden kurulmuştur.

Başlangıç koşullarında S'in elinde sınıf etiket bilgisi verilmiş sınırlı sayıda eğitim verisi (Sset) olduğu varsayılmış ve Sset üzerinden S hem nitelikleri hem de sınıfları modellemiştir. İlaveten, T'nin elinde sınıf etiketlerini sadece kendisinin bildiği başka bir veriseti (Uset) bulunmuş; S açısından Uset'in etiketleri gizli kalmıştır. Nitelikler sınıflar temelinde göreceli ilişkileri belirleyeceğinden, T nitelikleri sınıfları rastgele sıralamak suretiyle ilklendirmiş ve bu sıralamalar modelleme boyunca sabit

tutulmuştur. Böylece nitelik-sınıf matrisi [77]'ye benzer yapıda denetimsiz bir şekilde oluşturulmuştur.

İkili nitelikler yerine göreceli niteliklerin kullanımının temel sebepleri: Temsil uzayının faktöriyel oranında genişletilmesi ve öğrenilmeleri açısından görecelilik kavramının insan doğasına çok daha uygun olmasıdır. Zira bir niteliğin imge üzerinde var olup olmadığını kesin biçimde belirlemek yerine bahse konu imgenin başka bir imge ile o nitelik üzerinden mukayese edilmesi ve 'daha çok/az' veya 'benzer' şekilde öğrenime sokulması nitelik öğrenmede daha iyi bir genelleme sağlamaktadır [14].

Sset üzerinden sınırlı miktardaki örnek ile düşük seviye bütünsel öznitelik vektörleri ve T tarafından ilklendirilen rastgele nitelik-sınıf göreceli sıralama matrisi kullanılarak, S öncelikle nitelik uzaylarını ve nitelik ağırlık vektörleri üzerinden de sınıf modellerini başlangıç koşullarında oluşturmuştur. Müteakiben, S mevcut sınıf modellerini kullanarak kendisine etiketsiz görünen Uset içerisinden kendi işine en çok yarayan (maksimum entropi) örneği, $x^{(q)}$, seçerek tahmin ettiği sınıf etiketi ($C_{prediction}$) ile birlikte cevaplaması maksadıyla T'ye iletmiştir. T Uset içerisindeki örneklerin gerçek sınıf etiketlerini bildiği için bu tahmini doğru bir şekilde yanıtlayabilir. Ancak, yarı denetimli öğrenmeden farklı olarak, T sorguyu sadece 'evet' veya 'hayır, $x^{(q)}$ aslında $C_{ground\text{-}truth}$ sınıfındandır' şeklinde gerçek sınıf etiketi ile birlikte yanıtlamamıştır. Bunun yerine, T S-T arasındaki iletişim kanalı olan nitelikler üzerinden bir açıklama cümlesi ile S'yi cevaplamıştır.

Dikkat edilirse, bu uygulamanın iki ana etkisi olmuştur. Birincisi, her bir iterasyonda göreceli nitelik uzayları S tarafından güncellendiği için T'nin müteakip dönem cevapları da S'in güncellemeleri ile şekillenmiş; bir nevi S de T'yi neyi ne kadar öğrendiğini geri beslemiştir. Yani T'ye nitelik uzaylarını anladığı biçimde öğretmiştir. Diğer bir husus ise, S tarafından seçilen örneğin yanlış tahminlenmesi durumudur. S $x^{(q)}$ için yanlış sınıf tahmininde bulunduğunda, T örneğin gerçek sınıf etiketini $C_{ground\text{-}truth}$ vermemiş; sadece tahminin neden yanlış olduğunu nitelikler üzerinden mukayeseli açıklamıştır. Böylece, seçilen örnek yeniden Uset içerisine atılabilmiş ve diğer yinelemelerde defaten kullanılabilmiştir. Bu şekilde eğitim seti sanal olarak arttırılmıştır.

S T'den gelen nitelik tabanlı açıklamalar ışığında öncelikle göreceli nitelik uzaylarını ve daha sonra aynı iterasyon içerisinde sınıf modellerini güncellemiştir. Dikkat edilirse, T S'nin zihninde yer alan sınıf modellerine doğrudan müdahale etmemiş; ancak nitelikler üzerinden sistemi iyileştirici açıklamalarını yaymıştır. Böylece öğrenci merkezli bir öğrenme metodu uygulanmıştır. Sınıf modellerin güncellenmesinde diğer bir husus, Uset'ten ilave aday örneklerin de belirli ağırlıklandırma ile seçilmesi ve modelde iteratif olarak geçici olarak kullanılmasıdır. Böylece sınıf modellerin eğitiminde her yinelemede Uset'ten alınan farklı örnekler değişik ağırlıkları ile aday sınıflarına atanmak suretiyle sınıf model güncellemeleri yapılmıştır. S-T arasındaki yinelemeli olarak yürütülen nitelik uzayları ile sınıf modellerinin S tarafından eş zamanlı güncellendiği önerilen artırımlı öğrenme algoritması, Uset'teki bütün örneklerin S tarafından doğru sınıflar ile etiketlenmesi ile sonlandırılmıştır.

Deneylerde önerilen nitelik tabanlı yaklaşım, DAP ve SSAC gibi çatı metodlar ve literatürde yer alan benzer çalışmalar [14, 70, 77] ile iki popüler orta ölçekli veriseti, OSR ve PUBFIG, üzerinde karşılaştırılmıştır. Sınıflandırma bu çalışmanın odak noktası olduğu için yineleme miktarı, nitelik sayısı, ortalama başarı oranı ve ROC eğrileri çoklu deneylerde analiz temalarını oluşturmuştur. Özetle, önerilen algoritma diğerlerine göre dikkate değer biçimde daha iyi sonuçlar almıştır. İlave iş gücü olmaksızın rastgele sınıf sıralamaları ile elde edilmelerine rağmen, göreceli niteliklerin sınıf modelleri ile eş zamanlı ve arttırımlı öğrenildiğinde görüntü tanıma performansını yükselttiği tespit edilmiştir.

4.2. Öneriler

Tez çalışmasının temelde iki ana platformda kullanılması önerilmektedir. Birincisi düşük seviye öznitelik uzayının denetimsiz ve çok katmanlı bir sinir ağı mimarisi içerisinde veri kümeleme yoluyla oluşturulmasıdır. Literatürde yer alan ikili/göreceli nitelik öğrenme çalışmaları incelendiğinde, düşük seviye girdi öznitelik uzaylarının GIST, SIFT, HOG gibi el ile istatistiki oluşturulan süzgeç cevapları ve renk, yönelim gibi histogramlardan üretildiği görülmektedir. Fakat bu tezde düşük seviye öznitelik uzaylarının taban vektörleri denetimsiz bir şekilde doğrudan eğitim veri kümesindeki örneklerden ve çok katmanlı bir yapay sinir ağı mimarisi içerisinde öğrenilmiştir. Müteakiben görsel kelimeler sözlüğü olarak adlandırılan bahse konu taban vektörleri

bir benzerlik metriği yardımıyla örnekleri düşük seviyede yeni bir öznitelik uzayında temsil etmekte kullanılmıştır. Örneklerin bizzat kendisinden elde edilen bu denetimsiz dönüşüm sayesinde orta seviyede nitelik uzayına geçişte kategori içi değişintiyi azaltacak, kategoriler arası değişintiyi ise arttıracak yönde ayırt ediciliği desteklediği görülmüştür. Bununla birlikte; yapay sinir ağı mimarisi içerisinde oluşturulan kümeleme algoritmasının literatürde özgün bir yer elde edeceği ve bu yeni yönteme sıkça başvurulacağı değerlendirilmektedir.

Tezin diğer ana unsur ise görüntü tanımanın öğrenen-öğreten çift taraflı bir mekanizma içerisinde, öğrenci merkezli ve arttırımlı bir şekilde gerçekleştirilmesidir. Literatürde yer alan nitelik tabanlı görüntü tanıma çalışmaları incelendiğinde, orta seviye öznitelik uzayının taban vektörlerini temsil eden nitelik ağırlık vektörlerinin sınıf modellerininden tamamen bağımsız ve öğrenmenin daha ilk aşamasında elde edildiği görülmektedir. Oysa kategori modellerinin nitelik uzayındaki gizli örüntülerden oluşturulduğu düşünülürse; hedef kategoriler ile niteliklerin bir bütünün ayrılmaz parçaları olduğu varsayılmaktadır. Bu tezde ise nitelikler ile sınıf modellerin eş zamanlı olarak öğrenilmeleri amaçlanmıştır. Görüntü tanımada sınıflar ile niteliklerin değişimli şekilde aynı zamanda öğrenilmeleri kavramsal olarak literatüre özgün bir değer katacaktır.

Son olarak, nitelik tabanlı imge sınıflandırılmasında niteliklerin denetimsiz bir şekilde öğrenilmesi büyük önem arz etmektedir. Bu kapsamda müteakip çalışmalarda, bu tezde önerilen öbekleme algoritmasının kullanılması önerilmektedir. Görsel uzayda düşük seviye öznitelik vektörlerinin denetimsiz bir şekilde öbeklenmesi ile oluşan kümeler görsel nitelikleri temsil edebilir. Bu kapsamda öbeği temsil edecek olan prototip vektörü nitelik ağırlık vektörünü oluşturabilir. Bahse konu prototip vektörü kullanılmak suretiyle belirli bir metrik sistemdeki mesafe hesaplamaları sonucu elde edilecek skorlar da göreceli olarak bir imgenin bahse konu niteliğe ne derece sahip olduğunu gösterebilir. Böylece düşük seviye öznitelik uzayından bir üst kademe nitelik uzayına geçilmiş ve ana hedef olan sınıflandırıcı ayırtaç fonksiyonları bu nitelik uzayında elde edilebilir.

İlaveten; bu çalışmada çok katmanlı yapay sinir ağlarının esnek yapısı itibariyle her türlü amaç fonksiyonunu simüle edebileceği ve geleneksel sinir ağı yaklaşımının

aksine denetimsiz bir şekilde öğrenme algoritmalarının ağ üzerinde koşulabileceği öğrenilmiştir. Netice itibariyle; nitelik tabanlı imge sınıflandırma probleminin çözümünde çok katmanlı yapay sinir ağı modelinin kullanılabileceği ve her bir katmanının hiyerarşik bir şekilde sınıflandırmada ayrı bir fonksiyon ile görevlendirilmesi önerilmektedir.

KAYNAKLAR

[1] Li J., Wang J. Z., Real-time Computerized Annotation of Pictures, *Proc. Of the 14th ACM Multimedia Conference*, Santa Barbara, CA, USA, 23-27 October 2006.

[2] Ames M., Naaman M., Why We Tag: Motivations for Annotation in Mobile and Online Media, *Proc. SIGCHI Conf. on Human Factors in Computing Systems*, San Jose, CA, USA**,** 30 April-3 May 2007.

[3] http://expandedramblings.com/index.php/flickr-stats/, (Ziyaret tarihi: 24 Kasım 2015).

[4] Ariadna Q., Antonio T., Recognizing Indoor Scenes, MIT, http://web.mit.edu/torralba/www/indoor.html, (Ziyaret tarihi: 24 Kasım 2015).

[5] Deng J., Dong W., Socher R., Li L., Li K., Fei-Fei L., A Large-scale Hierarchical Image Database, ImageNet, http://www.image-net.org/, (Ziyaret tarihi: 24 Kasım 2015).

[6] Fei-Fei L., Perona P., A Bayesian Hierarchical Model for Learning Natural Scene Categories, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, San Diego, CA, USA, 20-26 April 2005.

[7] Genevieve P., James H., Discovering, Annotating and Recognizing Scene Attributes, SUN Attribute Database, http://groups.csail.mit.edu/vision/SUN/, (Ziyaret tarihi: 24 Kasım 2015).

[8] Lazebnik S., Schmid C., Ponce J., Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, New York, NY, USA, 17-22 June 2006.

[9] Sivic J., Russell B. C., Efros A., Zisserman A., Freeman W., Discovering of Objects and Their Location in Images, *IEEE ICCV*, Beijing, China, 17-20 October 2005.

[10] Oliva A., Torralba A., Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope, *Int'l J. Computer Vision*, 2001, **42**(3), 145-175.

[11] Vogel J., Schiele B., Semantic Modeling of Natural Scenes for Content-Based Image Retrieval, *Int'l J. Computer Vision*, 2007, **72**(2), 133-157.

[12] Bosch A., Zisserman A., Munoz X., Scene Classification Using a Hybrid Generative/Discriminative Approach, *IEEE Transections on PAMI*, 2008, **30**(4), 712-727.

[13] Hofmann T., Unsupervised Learning by Probabilistic Latent Semantic Analysis, *IEEE J. Machine Learning*, 2001, **41**(2), 177-196.

[14] Parikh D., Grauman K., Relative Attributes, *International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 6-13 November 2011.

[15] Alpaydın E., *Introduction to Machine Learning*, 3rd ed., The MIT Press, London, 2004.

[16] Tuytelaars T, Christoph H., Matthew B., Wray B., Unsupervised Object Discovery: A Comparison, *Int'l J. Comput Vision*, 2010, **88**(2), 284–302.

[17] Lampert C. H., Nickisch H., Harmeling S., Attribute-Based Classification for Zero-Shot Visual Object Categorization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014, **36**(3), 453-465.

[18] Lampert C. H., Nickisch H., Harmeling S., Learning to Detect Unseen Object Classes by Between-Class Attribute Transfer, *CVPR*, Miami Beach, FL, USA, 20-25 June 2009.

[19] Wah C., Belongie S., Attribute-Based Detection of Unfamiliar Classes with Humans in the Loop, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[20] Saleh B., Farhadi A., Elgammal A., Object-Centric Anomaly Detection by Attribute-Based Reasoning, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[21] Sadovnik A., Gallagher A., Chen T., It's Not Polite to Point: Describing People with Uncertain Attributes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[22] Li W., Yu Q., Sawhney H., Vasconcelos N., Recognizing Activities via Bag of Words for Attribute Dynamics, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[23] Ma Z., Yang Y., Xu Z., Sebe N., Yan S., Hauptmann A.G., Complex Event Detection via Multi-Source Video Attributes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[24] Bengio Y., Courville A., Vincent P., Representation Learning: A Review and New Per-spectives. *PAMI*, 2013, **35**(8), 1798-1828.

[25] Lowe D., Distinctive Image Features From Scale Invariant Keypoints, *Int'l J. Computer Vision*, 2004, **60**(2), 91-110.

[26] Bay H., Ess A., Tuytelaars T., Gool L.C., SURF: Speeded Up Robust Features, *Computer Vision and Image Understanding (CVIU)*, 2008, **110**(3), 346-359.

[27] Bosch A., Zisserman A., Munoz X., Representing Shape with a Spatial Pyramid Kernel, *ACM International Conference on Image and Video Retrieval*, Amsterdam, Netherlands, 9-11 September 2007.

[28] Arel I., Rose D. C., Karnowski T. P., Deep Machine Learning: A New Frontier in Artificial Intelligence Research, *IEEE Computational Intelligence Magazine*, 2010, **5**(4), 13-18.

[29] Hinton G. E., A Practical Guide to Training Restricted Boltzmann Machine, *Department of Computer Science University of Toronto*, UTML TR 2010 003, 3-12, 2010.

[30] http:/ufldl.stanford.edu / tutorial/ supervised/ MultiLayer Neural Networks, (Ziyaret tarihi: 24 Kasım 2015).

[31] Quoc L., Ranzato M., Monga R., Devin M., Chen K., Corrado G., Dean J., Andrew N. G., Building High-level Features Using Large Scale Unsupervised Learning, *International Conference in Machine Learning*, Edinburgh, Scotland, 26 June-1 July 2012.

[32] Hinton G. E., Osindero S., Teh Y. W., A Fast Learning Algorithm for Deep Belief Nets, *Neural Computation*, 2006, **18**(7), 1527-1554.

[33] Bengio Y., Lamblin P., Popovici D., Larochelle H., Greedy Layer- wise Training of Deep Networks, *Advances in Neural Information Processing Systems (NIPS)*, 2007, **19**(1), 153-160.

[34] Bengio Y., Learning Deep Architectures for AI, *Foundations and Trends in Machine Learning*, 2009, **2**(1), 1–127.

[35] http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/, (Ziyaret tarihi: 24 Kasım 2015).

[36] Rasmus B. P., Prediction as a Candidate for Learning Deep Hierarchical Models of Data, MS Thesis, Technical University of Denmark, DTU Informatics, 2012.

[37] Masci J., Meier U., Ciresan C., Schmidhuber J., Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction, *Proceedings of the 21th International Conference on Artificial Neural Networks*, Berlin, Germany, 14-17 June 2011.

[38] Srivastava N., Improving Neural Networks with Dropout, MS Thesis, Department of Computer Science, University of Toronto, 2013.

[39] Coates A., Lee H., Andrew Y. N., An Analysis of Single-layer Networks in Unsupervised Feature Learning, *14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Ft. Lauderdale, FL, USA, 11-13 April 2011.

[40] Raina R., Battle A., Honglak L., Packer B., Andrew Y. N., Self-taught Learning: Transfer Learning from Unlabeled Data, *In Proceedings of the 24th Int'l Conf. on Machine Learning (ICML)*, Corvallis, OR, USA, 20-24 June 2007.

[41] Gong Y., Wang L., Guo R., Lazebnik S., Multi- Scale Orderless Pooling of Deep Convolutional Activation Features, *Proceedings of the European Conference on Computer Vision (ECCV)*, Zurich, Switzerland, 6-12 September 2014.

[42] Krizhevsky A., Hinton G. E., Using very Deep Auto Encoders for Content based Image Retrieval, *19th European Symposium on Artificial Neural Networks (ESANN)*, Bruges, Belgium, 27-29 April 2011.

[43] Yang Y., Shah M., Complex Events Detection Using Data-driven Concepts, *ECCV*, Floransa, Italy, 7-13 October 2012.

[44] Şeker S. E., C 4.5 Ağacı (C 4.5 Tree), Bilgisayar Kavramları, http://bilgisayarkavramlari.sadievrenseker.com/2012/11/13/c4-5-agaci-c4-5-tree/ (Ziyaret Tarihi: 24 Kasım 2015).

[45] Cortes C., Vapnik. V., Support Vector Networks, Editor: Saitta L., *Machine Learning*, Kluwer Academic Publishers, Boston, 273-297, 1995.

[46] Ranzato M., Hinton G. E., Modeling Pixel Means and Covariances Using Factorized Third-order Boltzmann Machines. *23th CVPR*, San Francisco, CA, USA, 13-18 June 2010.

[47] Stekh Y., Sardieh M. E., Lobur M., Neural Network Based Clustering Algorithm, *International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH)*, Lviv, Ukraine, 22-24 April 2009.

[48] Du K. L., Clustering: A Neural Network Approach, *J. Neural Networks*, 2010, **23**(1), 89-107.

[49] Sum J. P. F., Leung C. S., Tam P. K. S., Young G. H., Kan W. K., Chan L. W., Analysis for a Class of Winner-Take-All Model, *IEEE Transactions on Neural Networks*, 1999, **10**(1), 64-71.

[50] Yair E., Zeger K., Gersho A., Competitive Learning and Soft Competition for Vector Quantizer Design, *IEEE Transactions on Signal Processing*, 1992, **40**(2), 294-309.

[51] Mishra M., Behera H. S., Kohonen Self Organizing Map with Modified K means Clustering for High Dimensional Data Sets, *International Journal of Applied Information Systems (IJAIS)*, 2012, **2**(3), 1-16.

[52] Kohonen T., *Self Organizing Maps*, 1st ed., Springer-Verlag, New York, 1997.

[53] Kohonen T., The Self-Organizing Map, *The Proceedings of IEEE*, 1990, **78**(9), 1464-1480.

[54] Dostál P., Pokorný P., Cluster Analysis and Neural Network, *Mezinárodní Konference Technical Computing*, Prague, Czech Republic, 11 November 2008.

[55] Martinetz T. M., Berkovich S. G., Schulten K. J., Neural-gas Network for Vector Quantization and its Application to Time-series Predictions, *IEEE Transactions on Neural Networks*, 1993, **4**(4), 558-569.

[56] Rose K., Gurewitz F., Fox G., Statistical Mechanics and Phase Transitions in Clustering, *Physical Rev. Lett.*, 1990, **65**(8), 945-948.

[57] Carpenter G. A., Grossberg S., A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine, *Computer Vision Graphics and Image Processing*, 1987, **37**(1), 54-115.

[58] Bezdek J., *Pattern Recognition With Fuzzy Objective Function Algorithms*, 1st ed., Plenum Press, New York, 1981.

[59] Ben-Hur A., Horn D., Siegelmann H. T., Vapnik V., Support Vector Clustering, *Journal of Machine Learning Research*, 2001, **2**(1), 125-137.

[60] Lee J., Lee D., Dynamic Characterization of Cluster Structures for Robust and Inductive Support Vector Clustering, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006, **28**(11), 1869-1874.

[61] Bolelli L., Ertekin S., Zhou D., Giles L. C., K-SVMeans: A Hybrid Clustering Algorithm for Multi-type Interrelated Datasets, *IEEE/WIC/ACM International Conference on Web Intelligence*, Silicon Valley, USA, 2-5 November 2007.

[62] Bosch A., Xavier M., Marti R., A Review: Which is the Best Way to Organize/Classify Images by Content, *Image and Vision Computing*, 2007, **25**(6), 778–791.

[63] Smeulders A. W., Worring M., Santini S., Gupta A., Jain R., Content-Based Image Retrieval at the End of the Early Years, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, **22**(12), 1349–1380.

[64] Ferrari V., Zisserman A., Learning Visual Attributes, *Advances in Neural Information Processing Systems (NIPS)*, 2007, **19**(1), 205-224.

[65] Farhadi A., Endres I., Hoiem D., Attribute-Centric Recognition for Cross-Category Generalization, *23th CVPR*, San Francisco, CA, USA, 13-18 June 2010.

[66] Farhadi A., Endres I., Hoiem D., Forsyth D., Describing Objects by Their Attributes, *CVPR*, Miami Beach, FL, USA, 20-25 June 2009.

[67] Sharma G., Jurie F., Schmid C., Expanded Parts Model for Human Attribute and Action Recognition in Still Images, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[68] Akata Z., Perronnin F., Harchaoui Z., Schmid C., Label Embedding for Attribute Based Classification, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[69] Parkash A., Parikh D., Attributes for Classifier Feedback, *ECCV*, Florance, Italy, 7-13 October 2012.

[70] Ma S., Sclaroff S., Cinbis N. I., Unsupervised Learning of Discriminative Relative Visual Attributes, *ECCV*, Florance, Italy, 7-13 October 2012.

[71] Rastegari M., Diba A., Parikh D., Farhadi A., Multi-attribute Queries: To Merge or Not to Merge, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[72] Wang Y., Mori G., A Discriminative Latent Model of Object Classes and Attributes, *The 11th European Conference on Computer Vision (ECCV)*, Crete Island, Grece, 5-11 September 2010.

[73] Yu F. X., Ji R., Tsai M., Ye G., Chang S., Weak Attributes for Large-scale Image Retrieval, *CVPR*, Rhode Island, USA, 16-21 June 2012.

[74] Wang S., Joo J., Wang Y., Zhu S. C., Weakly Supervised Learning for Attribute Localization in Outdoor Scenes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[75] Tamara L. B., Alexander C. B., Jonathan S., Automatic Attribute Discovery and Characterization From Noisy Web Data, *The 11th European Conference on Computer Vision (ECCV)*, Crete Island, Grece, 5-11 September 2010.

[76] Kumar N., Berg A. C., Belhumeur P. N., Nayar S. K., Attribute and Simile Classifiers for Face Verification, *ICCV*, Kyoto, Japan, 27 September-4 October 2009.

[77] Karayel M., Arica N., Random Attributes for Image Classification, *IEEE 21th Conference on Signal Processing and Communications Applications (SIU)*, Girne, TRNC, 24-26 April 2013.

[78] Biswas A., Parikh D., Simultaneous Active Learning of Classifiers & Attributes via Relative Feedback, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[79] Russakovsky O., Fei-Fei L., Attribute Learning in Large-scale Datasets, *The 11th European Conference on Computer Vision (ECCV)*, Crete Island, Grece, 5-11 September 2010.

[80] Wang Y., Mori G., A Discriminative Latent Model of Object Classes and Attributes, *The 11th European Conference on Computer Vision (ECCV)*, Crete Island, Grece, 5-11 September 2010.

[81] Chen K., Gong S., Xiang T., Loy C. C., Cumulative Attribute Space for Age and Crowd Density Estimation, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[82] Yu F. X., Cao L., Feris R. S., Smith J. R., Chang S., Designing Category-level Attributes for Discriminative Visual Recognition, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[83] Li W., Yu Q., Sawhney H., Vasconcelos N., Recognizing Activities via Bag of Words for Attribute Dynamics, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[84] Ma Z., Yang Y., Xu Z., Sebe N., Yan S., Hauptmann A. G., Complex Event Detection via Multi-source Video Attributes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[85] Chen H., Gallagher A., Girod B., What's in a Name: First Names as Facial Attributes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[86] Sadovnik A., Gallagher A., Chen T., It's not Polite to Point: Describing People with Uncertain Attributes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[87] Choi J., Rastegari M., Farhadi A., Davis L. S., Adding Unlabeled Samples to Categories by Learned Attributes, *CVPR*, Portland, OR, USA, 23-28 June 2013.

[88] Ergül E., Arıca N., Scene Classification Using Spatial Pyramid of Latent Topics, *IEEE 20th International Conference on Pattern Recognition (ICPR)*, Istanbul, Turkey, 23-26 August 2010.

[89] Ergül E., Ertürk S., Arıca N., Unsupervised Relative Attribute Extraction, *IEEE 21th Conference on Signal Processing and Communications Applications (SIU)*, Girne, TRNC, 24-26 April 2013.

[90] Chang C. C., Lin C. J., LIBSVM : A Library for Support Vector Machines, *J. of ACM Transactions on Intelligent Systems and Technology (TIST)*, 2011, **2**(3), 1-27.

[91] Shrivastava A., Singh S. and Gupta A., Constrained Semi-supervised Learning Using Attributes and Comparative Attributes, *ECCV*, Florance, Italy, 7-13 October 2012.

[92] http://cs.joensuu.fi/sipu/datasets/, (Ziyaret tarihi: 24 Kasım 2015).

[93] Fränti P., Virmajoki O., Iterative Shrinking Method for Clustering Problems, *IEEE J. Pattern Recognition*, 2006, **39**(5), 761-765.

[94] Gionis A., Mannila H., Tsaparas P., Clustering Aggregation, *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2007, **1**(1), 1-30.

[95] Cortez P., Cerdeira A., Almeida F., Matos T., Reis J., Modeling Wine Preferences by Data Mining from Physicochemical Properties, *In Decision Support Systems Elsevier*, 2009, **47**(4), 547-553.

[96] http://archive.ics.uci.edu/ml/datasets/Wine+Quality, (Ziyaret tarihi: 24 Kasım 2015).

[97] http://archive.ics.uci.edu/ml/datasets/Frm-Teac_Clv-Directio_Classification/, (Ziyaret tarihi: 24 Kasım 2015).

[98] http://archive.ics.uci.edu/ml/datasets/Letter+Recognition, (Ziyaret tarihi: 24 Kasım 2015).

[99] MathWorks, *MATLAB and Statistics Toolbox Versiyon 2013b*, The MathWorks Inc., Massachusetts, 2013.

[100] Scarpiniti M., Neural Networks and Cluster Analysis, University of Rome, http://ispac.ing.uniroma1.it/scarpiniti/files/NNs/Less5.pdf, (Ziyaret tarihi: 24 Kasım 2015).

[101] http://www.cs.toronto.edu/~kriz/cifar.html, (Ziyaret tarihi: 24 Kasım 2015).

[102] Krizhevky A., Convolutional Deep Belief Networks on CIFAR-10, *University of Toronto*, TR-CS-001, 1-9, 2010.

[103] Ranzato M., Krizhevsky A., Hinton G. E., Factored 3-way Restricted Boltzmann Machines for Modeling Natural Images, *13th Int'l Conf. on Artificial Intelligence and Statistics (ASTATS)*, Geranium Island, Italy, 13-15 May 2010.

.

KİŞİSEL YAYIN VE ESERLER

[1] **Ergül E.**, Scene Classification Using Spatial Pyramid of Latent Topics, Yüksek Lisans Tezi, Deniz Harp Okulu, Deniz Bilimleri ve Mühendisliği Enstitüsü, 2009, 275514.

[2] **Ergül E.**, Arıca N., Scene Classification Using Cascaded Probabilistic Latent Semantic Analysis, *Journal of Naval Science and Engineering*, 2009, **5**(2), 1-19.

[3] **Ergül E.**, Arıca N., Scene Classification Using Spatial Pyramid of Latent Topics, *IEEE 20th International Conference on Pattern Recognition (ICPR)*, Istanbul, Turkey, 23-26 August 2010.

[4] Çiftçi C., **Ergül E.**, Arıca N., Scene Classification Using Saliency Based Segmentation, *IEEE 19th Conference on Signal Processing and Communications Applications (SIU)*, Antalya, Turkey, 20-22 April 2011.

[5] Çiftçi C., **Ergül E.**, Arıca N., Feature Selection Using Filter Banks in Scene Classification, *IEEE 19th Conference on Signal Processing and Communications Applications (SIU)*, Antalya, Turkey, 20-22 April 2011.

[6] **Ergül E.**, Ertürk S., Arıca N., Unsupervised Relative Attribute Extraction, *IEEE 21th Conference on Signal Processing and Communications Applications (SIU)*, Girne, TRNC, 24-26 April 2013.

[7] Aydın S., **Ergül E.**, Arıca N., Tan O., Complexity and Coherence Analysis on EEG of Patients with Obsessive Compulsive Disorder, *ACTA Physiologica*, 2014, **21**(1), 151-157.

[8] Aydın S., Arıca N., **Ergül E.**, Tan O., Classification of Obsessive Compulsive Disorder by EEG Complexity and Hemispheric Dependency Measurements, *International Journal of Neural Systems*, 2015, **25**(3), 155-171.

[9] **Ergül E.**, Ertürk S., Arıca N., Hierarchical Image Representation Using Deep Network, *18th International Conference on Image Analysis and Processing (ICIAP)*, Genova, Italy, 7-11 September 2015.

[10] **Ergül E.**, Arıca N., Ahuja N., Ertürk S., Clustering Through Hybrid Network Architecture with Support Vectors, *IEEE Transactions on Neural Networks and Learning Systems*, DOI:10.1109/TNNLS-2015-P-5659.R1.

ÖZGEÇMİŞ

1979 yılında Afyonkarahisar'da doğdu. İlk ve orta öğrenimini Ankara'da tamamladı. Lise öğrenimini 1993-97 yılları arasında İstanbul Deniz Lisesi'nde, üniversite öğrenimini 1997-2001 yılları arasında Deniz Harp Okulu Kontrol ve Bilgisayar Mühendisliği bölümünde tamamlayarak 2001 yılında subay olarak mezun oldu. 2009 yılında Deniz Harp Okulu Deniz Bilimleri ve Mühendisliği Enstitüsü Bilgisayar Mühendisliği bölümünde Yüksek Lisans öğrenimini tamamladı. 2001 yılından bugüne Deniz Kuvvetleri Komutanlığı'nda elektronik, lojistik ve bilgisayar yazılımları ile ilgili görevlerde bulunmuştur. Bayan Serpil Ergül ile evli olup İngilizce bilmektedir.