

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DERİN ÖĞRENME İLE GÖRÜNTÜLERDE KARARLI ÖZNİTELİK
EŞLEME TEKNİKLERİNİN GELİŞTİRİLMESİ**

DOKTORA TEZİ
Muhammet Fatih AYDOĞDU

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Prof. Dr. M. Fatih DEMİRCİ

KASIM 2024







TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

Muhammet Fatih AYDOĞDU



ÖZET

Doktora Tezi

DERİN ÖĞRENME İLE GÖRÜNTÜLERDE KARARLI ÖZNETELİK EŞLEME TEKNİKLERİNİN GELİŞTİRİLMESİ

Muhammet Fatih AYDOĞDU

TOBB Ekonomi ve Teknoloji Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Prof.Dr. M. Fatih DEMİRCİ

Tarih: KASIM 2024

Bilgisayarlı görü teknikleri görüntüler üzerinde tespit edilen öznitelik noktalarından yaygın bir şekilde yararlanmaktadırlar. Bu öznitelik noktaları kullanılarak görüntü çiftleri arasında yararlı tutarlılıklar tespit edilebilir. Benzerlikler kullanılarak görüntü eşleştirme, nesne tanıma, görüntü dikişleme, görüntü mozaïği oluşturma ve nesne takibi gibi birçok uygulama için başarılı ürünler elde edilebilir. Görüntü çiftleri üzerinde tespit edilen öznitelik noktalarının eşleştirilmesi sırasında noktaların öznitelik uzayında birbirlerine göre uzaklıkları temel alınır. Öznitelik uzayında birbirlerine en yakın olan özniteliklerin yakınlıkları eğer yeterince özgünse bu öznitelikler varsayılan eşleşmeler olarak kabul edilir. Ancak bu varsayılan eşleşmeler çoğu zaman hatalı eşleşmeleri tümüyle saf dışı bırakamaz. Bunun için literatürde tekrarlamalı algoritmalarından faydalananak en çok sayıda varsayılan eşleşmeyi içerecek şekilde bir geometrik tutarlılık elde edilmeye çalışılır. Elde edilen geometrik tutarlılık sayesinde varsayılan eşleşmelerde bulunan hatalı eşleşmeler elenir. Bu yöntemdeki temel sorun tüm görüntü çiftlerinde başarıya götürecek bir tekrarlamalı sayısının elde edilmesinin pratikteki imkânsızlığıdır. Derin öğrenme yöntemlerinin literatürde birçok problemde alternatiflerine göre daha etkin sonuçlar elde etmesinden sonra çoğu bilgisayarlı görü ve görüntü işleme probleminde olduđu gibi öznitelik eşleştirme problemi için de derin öğrenme ile eğitilmiş yapay sinir ağıları kullanan çözümler literatüre yerleşmişlerdir.

Bu tezde, kararlı öznitelik eşleme yapan derin öğrenme ağıları ile bağılı kamera pozisyonu tahmini problemine çözümler geliştirilmiştir. Öncelikli olarak literatürdeki çalışmaların hepsine temel oluşturan $n-n$ 'lik çatı incelenmiştir. $n-n$ 'lik bu çatının varsayılan

eşlemelerdeki özniteliklerin koordinatlarından oluşan bir küme tipi girdi üzerinde çalıştığı gözlemlenmiştir. n - n 'lik çatıya ait girdiyi işleyen literatürdeki çalışmaların varsayılan öznitelik eşleşmelerinde genel bağlam ve yerel bağlam çıkarırken karşılaştığı zorluklar incelenmiştir. n - n 'lik çatıya alternatif olarak 1 - 1 'lik alternatif bir çatı oluşturulmuştur. n - n 'lik çatıda her bir yığın örneğinde tek bir görüntü çiftine ait veri bulunurken öne sürülen 1 - 1 'lik çatıda her bir yığında sadece tek bir görüntü çiftine ait veri bulunmaktadır. 1 - 1 'lik çatıdaki görüntü çiftine ait her bir varsayılan eşleşme için özel bir bağlam kanalının kullanılmasına imkân sağlanmaktadır. Dahası bu bağlam kanalındaki girdi satırların her bir varsayılan eşleşme için özel olarak sıralanabilmesi mümkündür. Tez kapsamında 1 - 1 'lik çatı kullanan çok sayıda ve farklı tipte ağ katmanları içeren yapay sinir ağları oluşturulmuştur. Oluşturulan 1 - 1 'lik yapay sinir ağları ve literatürdeki n - n 'lik başarılı sinir ağları Tensor İşlem Birimleri üzerinde eğitilmişlerdir. Eğitimlerde mimarilere ait hesapsal grafiklerdeki parametreler güncellenirken birden fazla kayıp fonksiyonunun birleşiminden oluşan bileşke kayıp fonksiyonundan faydalanılmıştır. Başarım metriği olarak minimum ortalama hassasiyet ölçütü temel alınmıştır. Elde edilen sonuçlara göre 1 - 1 'lik çatı için oluşturulan yapay sinir ağları literatürdeki n - n 'lik yapay sinir ağlarının biri hariç tümünün başarımlarını %30'a varan farklar ile geride bırakmıştır.

Ayrıca tezin asıl konusu üzerinde çalışmaya başlamadan önce derin öğrenme kullanarak yapay sinir ağlarının eğitilmesine aşinalığı arttırmak için bir durum çalışması yapılmıştır. Bunun için portre fotoğrafları üzerinden yaş sınıfı tahmini yapan yapay sinir ağları geliştirilmiştir. Kullanılan veri setindeki portreler 6 sınıfa ayrılmıştır. Literatürdeki 6 katmanlı bir yapay sinir ağının 18 ve 34 katmanlı artık sinir ağlarına göre daha başarılı olduğu gözlemlenmiştir. Kullanılan artık sinir ağlarının 6 sınıflı yaş tahmini problemi için aşırı öğrenmeye sebep olacak kadar derin olduğu veya veri setinin yeterince zengin olmadığı sonucuna varılmıştır.

Anahtar Kelimeler: Derin öğrenme, Bağlı kamera pozisyonu tahmini, Tensor işleme uniteleri, Öznitelik eşleştirme, Derin öğrenme ile yaş sınıfı tahmini.

ABSTRACT

Doctor of Philosophy

DEVELOPMENT OF ROBUST FEATURE MATCHING TECHNIQUES IN IMAGES USING DEEP LEARNING

Muhammet Fatih AYDOĞDU

TOBB University of Economics and Technology
Institute of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof.Dr. M. Fatih DEMİRCİ

Date: SEPTEMBER 2024

Computer vision techniques widely utilize feature points detected in images. Using these feature points, useful consistencies can be detected between pairs of images. By leveraging these similarities, successful results can be achieved for various applications such as image matching, object recognition, image stitching, creating image mosaics, and object tracking. When matching feature points detected in pairs of images, the distances between the points in the feature space are used as the basis. If the proximities of the features closest to each other in the feature space are sufficiently distinctive, these features are considered putative matches. However, these putative matches often cannot entirely exclude incorrect matches. To address this, iterative algorithms are used in the literature to achieve geometric consistency that includes the maximum number of putative matches. Thanks to the geometric consistency obtained, incorrect matches in the putative matches are eliminated. The main issue with this method is the practical impossibility of finding a repetition number that guarantees success for all image pairs. Following the effective results achieved by deep learning methods over alternatives in many computer vision problems in the literature, solutions that use artificial neural networks trained using deep learning have become prevalent for the feature matching problem, as they have in most computer vision and image processing problems.

In this thesis, solutions to the problem of relative camera pose estimation have been developed using deep learning networks that perform stable feature matching. First and foremost, the n -to- n framework, which forms the basis of all studies in the literature, was examined. This n -to- n framework was observed to operate on a set-type input

consisting of the coordinates of features in putative matches. The challenges faced by studies in the literature using this framework in deriving global and local contexts from putative feature matches were analyzed. As an alternative to the n -to- n framework, a *one-to-one* framework was proposed. In the n -to- n framework, each batch contains data belonging to a single image pair, whereas, in the proposed *one-to-one* framework, each batch also includes data belonging to only a single image pair. The *one-to-one* framework allows the use of a dedicated context channel for each putative match of the image pair. Moreover, it is possible to specifically sort the input rows in this context channel for each putative match. Within the scope of the thesis, a variety of fundamental artificial neural networks with different architectures using the *one-to-one* framework were generated. The *one-to-one* artificial neural networks developed, along with the successful n -to- n ones in the literature, were trained on Tensor Processing Units (TPUs). Multiple loss functions were utilized during the training. The minimum average precision was used as the performance metric. According to the results, the artificial neural networks designed for the *one-to-one* framework outperformed all but one of the n -to- n neural networks from the literature by up to 30%.

Additionally, before working on the main topic of the thesis, a case study was conducted to increase familiarity with deep neural networks. For this purpose, artificial neural networks were developed to estimate age categories using portrait photographs. The portraits in the dataset were divided into six classes. It was observed that a 6-layer artificial neural network from the literature performed better than the 18 and 34-layer residual neural networks. It was concluded that the 18-layer and 34-layer residual networks might have been unnecessarily deep for the 6-class age estimation problem, leading to overfitting, or that the dataset lacked sufficient diversity to demonstrate the advantages of deeper networks.

Keywords: Deep learning, Relative camera pose estimation, Feature matching, Tensor procesing units, Age class estimation using deep learning.

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren hocam Prof.Dr. M. Fatih DEMİRCİ, kıymetli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Bilgisayar Mühendisliği Bölümü öğretim üyelerine çok teşekkür ederim.





İÇİNDEKİLER

	Sayfa
ÖZET	vii
ABSTRACT	ix
TEŞEKKÜR	xi
İÇİNDEKİLER	xiii
ŞEKİL LİSTESİ	xv
ÇİZELGE LİSTESİ	xvii
KISALTMALAR	xix
RESİM LİSTESİ	xxi
1. GİRİŞ	1
2. LİTERATÜR ARAŞTIRMASI	5
3. PROBLEM TANIMI	9
4. 1-1 ÇATI	11
5. DENEYLER	15
5.1 Veri Setleri ve Seyrek Öznitelikler	15
5.2 Metrikler	19
5.3 Uygulama Detayları	19
6. BAŞARIMLAR	21
6.1 n-n Çatısı Referans Modellerine Ait Başarımlar	21
6.2 Sıralamadan Bağımsız 1-1 Ağların Başarımları	22
6.3 Sıralamaya Bağımlı 1-1 Ağların Başarımları	22
6.3.1 Bağlam-0 Yapısını Kullanan ESA Mimarilerinin Başarımları	41
6.3.2 Bağlam-1 Yapısını Kullanan ESA Mimarilerinin Başarımları	42
6.3.3 Bağlam-2 Yapısını Kullanan ESA Mimarilerinin Başarımları	45
6.3.4 Bağlam-3 Yapısını Kullanan ESA Mimarilerinin Başarımları	48
6.4 Genel Başarım Karşılaştırması	48
7. DURUM ÇALIŞMASI: YAŞ SINIFI TAHMİNİ	57
7.1 Durum Çalışmasının Amacı	57
7.2 Yaş Tespiti Problemi	57
7.3 Ağ Mimarileri ve Veri Seti	59
7.3.1 6 Katmanlı Ağ	59
7.3.2 RESNET	59
7.3.3 Veri Seti	62
7.4 Görüntü Ön-İşleme	62
7.4.1 Görüntü Hizalama	63
7.4.2 Yüz Tespiti	63
7.5 Deney Detayları	64
7.5.1 Veri Setindeki Hataların Giderilmesi	64

7.5.2 Deney Ortamı	64
7.6 Ağ Başarımlarının Kıyaslaması	65
7.7 İdeal Ağ Mimarisinin Elde Edilmesi	67
8. SONUÇ VE ÖNERİLER	79
KAYNAKLAR	81
ÖZGEÇMİŞ	87



ŞEKİL LİSTESİ

	Sayfa
Şekil 1.1: n - n çatıya (a) ve önerilen 1 - 1 (b) çatıya ait girdi yığın yapıları.	2
Şekil 4.1: Önerilen 1 - 1 çatının yapısı.	12
Şekil 6.1: PointCN ağı başarımı, (E=4, $\sin_kayıp$)	23
Şekil 6.2: PointCN ağı başarımı, (E=4, $\sin_kayıp+geo_kayıp$)	24
Şekil 6.3: PointCN ağı başarımı, (E=4, $\sin_kayıp+temel_kayıp$)	25
Şekil 6.4: PointCN ağı başarımı, (E=6, $\sin_kayıp$)	26
Şekil 6.5: PointCN ağı başarımı, (E=6, $\sin_kayıp+geo_kayıp$)	27
Şekil 6.6: PointCN ağı başarımı, (E=6, $\sin_kayıp+temel_kayıp$)	28
Şekil 6.7: OANET ağı başarımı, (E=4, $\sin_kayıp$)	29
Şekil 6.8: OANET ağı başarımı, (E=4, $\sin_kayıp+geo_kayıp$)	30
Şekil 6.9: OANET ağı başarımı, (E=4, $\sin_kayıp+temel_kayıp$)	31
Şekil 6.10: OANET ağı başarımı, (E=6, $\sin_kayıp$)	32
Şekil 6.11: OANET ağı başarımı, (E=6, $\sin_kayıp+geo_kayıp$)	33
Şekil 6.12: OANET ağı başarımı, (E=6, $\sin_kayıp+temel_kayıp$)	34
Şekil 6.13: OANETiter ağı başarımı, (E=4, $\sin_kayıp$)	35
Şekil 6.14: OANETiter ağı başarımı, (E=4, $\sin_kayıp+geo_kayıp$)	36
Şekil 6.15: OANETiter ağı başarımı, (E=4, $\sin_kayıp+temel_kayıp$)	37
Şekil 6.16: OANETiter ağı başarımı, (E=6, $\sin_kayıp$)	38
Şekil 6.17: OANETiter ağı başarımı, (E=6, $\sin_kayıp+geo_kayıp$)	39
Şekil 6.18: OANETiter ağı başarımı, (E=6, $\sin_kayıp+temel_kayıp$)	40
Şekil 6.19: ESA-dar-0 ağı başarımı.	43
Şekil 6.20: ESA-geniş-0 ağı başarımı.	44
Şekil 6.21: ESA-dar-1 ağı başarımı.	46
Şekil 6.22: ESA-geniş-1 ağı başarımı.	47
Şekil 6.23: ESA-dar-2 ağı başarımı.	49
Şekil 6.24: ESA-geniş-2 ağı başarımı.	50
Şekil 6.25: ESA-dar-3 ağı başarımı.	51
Şekil 6.26: ESA-geniş-3 ağı başarımı.	52
Şekil 6.27: CLNET [1] ağı başarımı.	54
Şekil 6.28: ConvMatch [2] ağı başarımı.	55
Şekil 6.29: ConvMatchimp [3] ağı başarımı.	56
Şekil 7.1: 6 sınıflı yaş sınıflandırması için 6 katmanlı ESA mimarisi.	59
Şekil 7.2: Düz bağlantılar ve artık bağlantılar.	60
Şekil 7.3: 6 sınıflı yaş sınıflandırması için ResNet-18 mimarisi.	61
Şekil 7.4: Görüntü ön-işleme adımları.	63
Şekil 7.5: 1-hata kriterine göre ağların başarılı olduğu görüntü oranları.	68
Şekil 7.6: 16 ağa ait eğitilme süreleri	68
Şekil 7.7: 2, 3, 4 ve 5 ESA katmanından oluşan ağ parçaları	69
Şekil 7.8: 1, 2, 3 ve 4 tam bağlantılı katmandan oluşan ağ parçaları	70

Şekil 7.9: 16 ağa ait tam başarımlarına göre test sonuçları	72
Şekil 7.10: 16 ağa ait ilk-3 başarımlarına göre test sonuçları	73
Şekil 7.11: 16 ağa ait tam başarımlarına göre standart sapmalar	74
Şekil 7.12: 16 ağa ait 1-hata başarımlarına göre standart sapmalar	75



ÇİZELGE LİSTESİ

	Sayfa
Çizelge 6.1: Referans mimarilerin [4, 5] performans sonuçları.	21
Çizelge 6.2: SKD-dar ve SKD-geniş mimarileri kanal sayıları.	41
Çizelge 6.3: SKD-dar ve SKD-geniş ağlarının performans sonuçları.	41
Çizelge 6.4: ESA-dar ve ESA-geniş mimarileri kanal sayıları.	42
Çizelge 6.5: Bağlam-0 yapısı ESA mimarilerinin performans sonuçları.	42
Çizelge 6.6: Bağlam-1 yapısı ESA mimarilerinin performans sonuçları.	45
Çizelge 6.7: Bağlam-2 yapısı ESA mimarilerinin performans sonuçları.	45
Çizelge 6.8: Bağlam-3 yapısı ESA mimarilerinin performans sonuçları.	48
Çizelge 6.9: n - n ağlar ile 1 - 1 çatısındaki ağların performans kıyaslaması.	53
Çizelge 7.1: 6 yaş sınıfının sınırları ve sınıflarda bulunan görüntü sayıları.	65
Çizelge 7.2: 6-katmanlı ESA sınıflandırma sonuçları.	66
Çizelge 7.3: ResNet-18 sınıflandırma sonuçları.	66
Çizelge 7.4: ResNet-34 sınıflandırma sonuçları.	66
Çizelge 7.5: Ağların tam ve 1-hata başarımları.	67
Çizelge 7.6: 2 evrişim, 4 tam bağlantılı katmanlı ağın test sonuçları.	77
Çizelge 7.7: 2 evrişim, 4 tam bağlantılı katmanlı ağın tam başarımları.	77



KISALTMALAR

AGM	: Aktif Görünüm Modeli (Active Appearance Model)
BaN	: Bağlam Normalizasyonu (Context Normalization)
BEB	: Bağlam Etkileşim Bloğu
CPU	: Merkezi İşlem Birimi (Central Processing Unit)
ÇKA	: Çok Katmanlı Algılayıcı (Multi Layer Perceptron)
DDO	: Düzeltilmiş Doğrusal Operatör (Rectified Linear Unit, ReLU)
EAKA	: Eğri altında kalan alan (Area Under Curve)
EKAR	: En Küçük Açık Regresyonu (Least Angle Regression)
ESA	: Evrişimli Sinir Ağı (Convolutional Neural Network)
GPU	: Grafik İşlem Birimi (Graphics Processing Unit)
KDDO	: Kaçaklı Düzeltilmiş Doğrusal Operatör (Leaky Rectified Linear Unit, Leaky ReLU)
MORPH	: Craniofacial Longitudinal Morphological Face Database
NORM	: Normalizasyon (Normalization)
oD	: Ortalama Doğruluk (Mean Average Precision)
SIFT	: Ölçekten Bağımsız Öznitelik Dönüşümü (Scale-Invariant Feature Transform)
SKD	: Sadece Kodlayıcı Dönüştürücü (Encoder Only Transformer)
SRA	: Sınıflandırma ve Regresyon Ağacı (Classification and Regression Trees)
TPU	: Tensor İşlem Birimi (Tensor Processing Unit)
YİD	: Yerel İkili Desenler (Local Binary Patterns)



RESİM LİSTESİ

	Sayfa
Resim 5.1: Kullanılan veri setlerinden Aya Sofya görüntü örnekleri.	16
Resim 5.2: Kullanılan veri setlerinden Londra Köprüsü görüntü örnekleri. . . .	17
Resim 5.3: Kullanılan veri setlerinden Tac Mahal görüntü örnekleri.	18
Resim 7.1: Kullanılan Morph Veri Setine [6] ait örnek görüntüler.	62



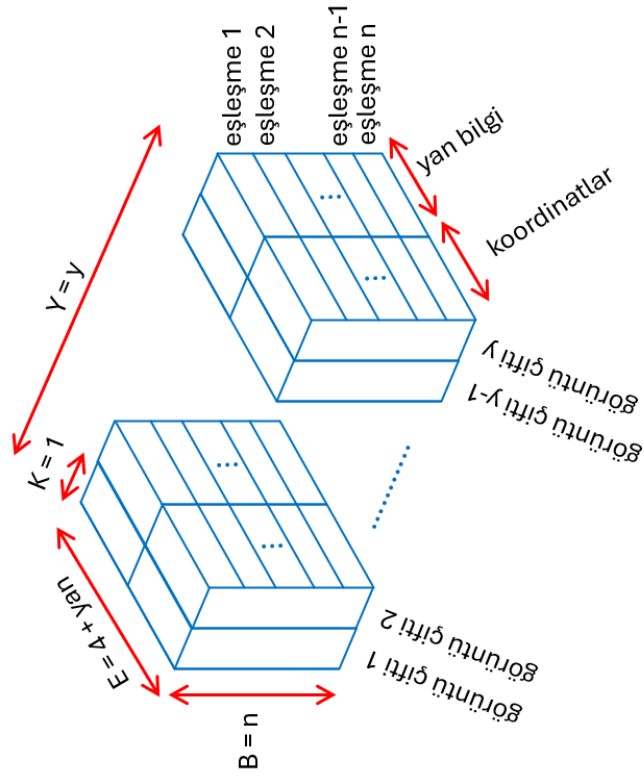


1. GİRİŞ

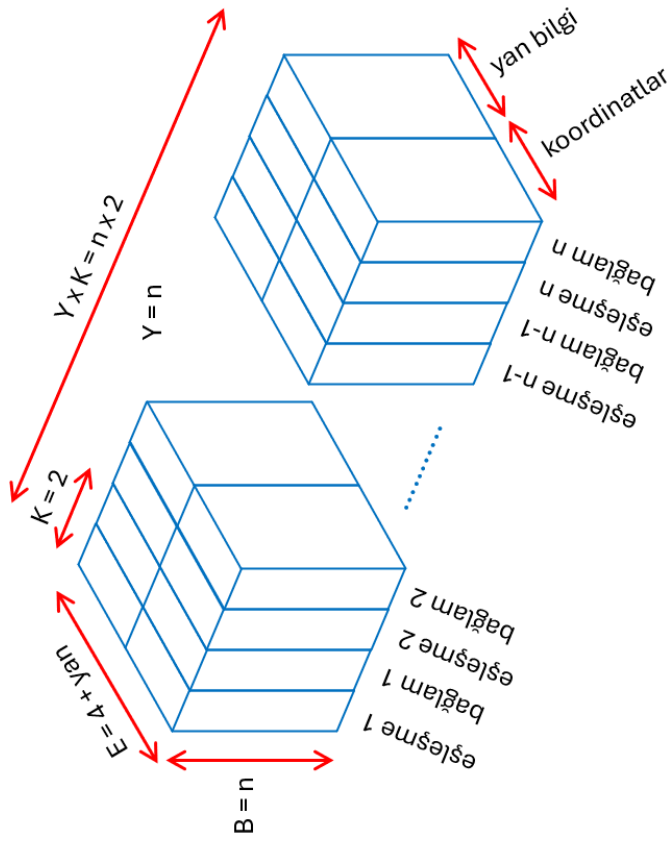
Görüntü çiftleri üzerinde tespit edilen öznitelik noktalarının başarılı bir şekilde eşleştirilmesi birçok bilgisayarlı görü problemi için temel operasyonlardan biridir. Bu tez kapsamında öznitelik noktalarının eşleştirilmesi için çalışmalar yapılırken, görel kamera pozisyon tahmini problemi, özniteliklerin eşleştirme başarımının tespitinde temel uygulama olarak seçilmiştir. Görüntü çiftlerinin görel pozisyon tahmini, en eski ve en popüler bilgisayarla görme problemleri arasındadır. Görüntülerin görel pozisyonlarının tahmin edilmesi, eşzamanlı konumlandırma ve haritalama, otonom sürüş, üç boyutlu yeniden yapılandırma ve sürü robotları koordinasyonu gibi birçok problem için kritik öneme sahiptir. Görel kamera pozisyon tahmini için gürbüz bir algoritma geliştirmek, görüntülerde bulunan benzer görüntü parçaları, değişken aydınlatma koşulları, yerel engellemeler ve bakış açılarının çeşitliliği gibi faktörler nedeniyle zorluklar içermektedir.

Görel kamera pozisyon tahmini problemini çözmek için geleneksel yöntemler, yinelemeli seyrek yaklaşımlara dayanır. İlk olarak, SIFT [7], LIFT [8] veya SuperPoint [9] gibi seyrek öznitelik noktaları, görüntü çiftlerinden çıkarılır. Öznitelik uzayında en benzer özniteliklerin görüntü koordinatları, varsayılan eşleşmeler olarak ikiye bölünür. Ardından, RANSAC [10] veya USAC [11] gibi yinelemeli algoritmalar, temel matris veya fundamental matris için geometrik olarak tutarlı en büyük eşleşme kümesini elde etmek için kullanılır. Hipotez matrislerinin hesaplanması sırasında, görüntüleyicilerin içsel parametrelerinin mevcudiyetine göre 8-nokta algoritması [12] veya 5-nokta algoritması [13] kullanılabilir. Tüm görüntü çiftlerinden maksimum performans elde ederken bir hesaplama süresi sınırı belirlemenin zorluğu, yinelemeli yöntemlerin pratikte kullanımını kısıtlamaktadır.

Ulaşılabilir verilerin sayısındaki artışlar ve özel donanımların gelişimi, diğer birçok bilgisayarla görme problemi gibi görel kamera pozisyon tahmini probleminin çözümünü de önemli ölçüde iyileştiren derin öğrenme tabanlı çözümlerin uygulanabilir olmasının yolunu açtı. Görel kamera konumu tahmini ağlarının çoğunluğu, giriş yapısı Şekil 1.1'in solunda gösterilen n - n çatısında eğitilmektedir. PointCN [4], sıralamadan bağımsız girdi kabul eden bu tip ağları eğitmek için n - n çatısını kullanan öncü çalışmadır. Görüntü çiftlerinin eşleşmelerinden bağımsız olarak, n - n çatıda eğitilen ağlar, her bir yığın örneği için küme türündeki n giriş eşleşmesine karşılık n sınıflandırma çıktısı üretmeyi öğrenmelidirler. Küme türündeki girdiler kullanarak bu işlevselliği sağlamak için, Set Transformer [14] kullanımına benzer şekilde ortalama, toplam, maksimum gibi havuz işlemlerine [15] ihtiyaç duyulmaktadır. Havuz işlemlerinin ortak özelliği, farklı sıralamalara sahip girdiler için aynı çıktıları üretmeleridir. PointCN'de, bağlam normalizasyonu operasyonunda bulunan ortalama ve varyans alma işlemleri ile havuz operasyonu ağda uygulanmıştır. PointCN ve n - n çatısıyla eğitilen tüm sonraki ağlar



(a) $n-n$ çatıya ait girdi yığm yapısı.



(b) Önerilen $I-I$ çatıya ait girdi yığm yapısı.

Şekil 1.1: $n-n$ çatıya (a) ve önerilen $I-I$ (b) çatıya ait girdi yığm yapıları.

için, ağ içindeki tüm işlemler, mimarilerin girdi sıralamasından bağımsız olarak aynı çıktıyı üretme özelliğini korumalıdır. Bağlam normalizasyonu işlemleri global bağlam çıkarımı yapabilse de yerel bağlam çıkarımı yapamamaktadır. Dolayısıyla, $n-n$ çatısında eğitilen önceki mimariler, yerel bağlam çıkarımı yapabilmek için özelleşmiş ağ katmanları kullanmak zorunda kalmışlardır. Global ve yerel bağlam çıkarımları yapmak için ayrı ağ öğeleri eklemek yerine, ağların her iki bağlamı da daha verimli bir şekilde yakalamasına yardımcı olacak, bu tez kapsamında önerdiğimiz çatıya benzer daha işlevsel çatılar geliştirilebilir.

On yıl önce gerçekleşen derin öğrenme yükselişinden sonra, son yıllarda, büyük dil modelleri yeniliklerin yeni odak noktası haline gelmiştir. Büyük dil modelleri bulut donanımlarındaki gelişmelerin yardımıyla eğitilebilir hale gelmiş ve günlük hayatta insan hayatını kolaylaştıran modellerin geliştirilmesine olanak sağlamıştır. Google Bulut Tensor İşlem Birimleri (TPU) için geliştirilen yongalara benzer yapıdaki özelleşmiş yongalar, eğitim sürelerini en aza indirmek ve daha büyük aktivasyon haritalarının ağ mimarilerinde kullanılabilmesine imkân vermek için üretilmektedir. İşlem yapma hızı açısından CPU'lar, GPU'lar ve TPU'lar kabaca şu şekilde kıyaslanmaktadır: metin girdileri için, CPU'ların 1 kelimeyi işleyebildiği sürede GPU'lar 1 satırı işleyebilme kapasitesine sahipken TPU'lar 1 sayfayı işleyebilecek kapasitededirler. TPU'ların sahip olduğu bu üstün işlem gücü ve daha büyük aktivasyon haritalarını barındırabilme yeteneği, bizi görüntü çiftlerine ait eşleşmeler için bağlam etkileşimini en üst düzeye çıkarmak amacıyla tasarlanmış yeni bir çatı oluşturmaya teşvik etmiştir.

Bu tezde, görelî kamera pozisyon tahmini problemine çözüm sağlayan ağlarda kullanılmak üzere görüntü çiftlerinden gelen eşleşmeler arasındaki bağlam etkileşimini artırmak için yeni bir $1-1$ çatı önerilmiştir. Önerilen çatının giriş yapısı Şekil 1.1'in sağında gösterilmektedir. Bu çatıda, yığınlar yalnızca bir görüntü çiftine ait verileri barındırır ve her bir yığın örneğinde yalnızca bir eşleşmeye ait veriler yer alır. Her yığın örneği iki kanal içerir: İlk kanal, varsayılan eşleşmelerin tekrarlayan koordinatlarını ve yan bilgilerini tutar; ikinci kanal ise her bir eşleşme için özel olarak yapılandırılmış bağlam bilgilerini barındırır. Ayrıca, çatının yapısı bağlam kanallarının sayısının artırılabilmesini de mümkün kılar. Bu da ağların bağlam çıkarım yeteneklerini arttırabilir. Her bir eşleşme için özel bağlam bilgilerinin eklenmesiyle, $1-1$ çatısında eğitilen ağların, $n-n$ çatısında eğitilenlerden daha yüksek performans gösterme potansiyeline sahip olacağı bu tez kapsamında savunulmaktadır. Önerdiğimiz ve $1-1$ çatısında eğittiğimiz mimarilere ek olarak, literatürde iyi bilinen $n-n$ mimariler de transfer öğrenme yapmadan, sıfırdan eğitilmiştir. Böylece bu tez, yalnızca $1-1$ çatısında eğitilen ağların performans artış adımlarını irdeleyen bir çalışma olarak kalmak yerine, alandaki bundan sonraki çalışmalara kapsamlı bir temel sağlayacaktır. Önerilen çatımız, geleneksel $n-n$ çatısı gibi küme türündeki giriş verilerine sahip herhangi bir problem için kullanılabilir. $n-n$ çatısından farklı olarak, önerilen $1-1$ çatısı girdi sıralamasına göre çıktısı değişen ve değişmeyen ağ mimarilerine ev sahipliği edebilmektedir. Tüm eğitim ve test çalışmaları, önceki çalışmalarda yaygın olarak kullanılan YFCC100M [16] dış mekân veri seti kullanılarak gerçekleştirilmiştir.

Tezde görelî kamera pozisyon çıkarım problemi için ortaya koyduğumuz temel katkılar şunlardır:

- Sıralamaya bağı ağların kullanımına izin veren yeni bir $1-1$ çatısının tanıtılması.
- Performansımızın çok yakın olduğı tek bir yaklaşım hariç, tüm $n-n$ ağların performansını önemli ölçüde geride bırakan $1-1$ mimarilerinin geliştirilmesi.
- Sabit sayıda eğitim iterasyonu yerine, doğrulama geometrik kaybında önemli bir iyileşme gözlenmeyene kadar eğitim iterasyonlarına devam ederek, önerdiğimiz tüm $1-1$ mimarilerin ve iyi bilinen $n-n$ mimarilerin gerçek performanslarını gözlemleyebilmek için kendi öğrenme süreçlerimizi yürütmek.

Tez kapsamında kararlı öznitelik eşleme yöntemleri geliştirerek görel kamera pozisyonu tahmini problemine ait çözümleri geliştirmenin yanı sıra derin öğrenme tabanlı çözümlere aşinalık sağlamak amacıyla bir durum çalışması yapılmıştır. Bu durum çalışması ile 6 yaş sınıfına ayrılmış ve portre fotoğrafları içeren bir veri seti üzerinden yaş sınıflandırması yapan farklı ESA geliştirme çalışmaları yapılmıştır.

Tezin bundan sonraki akışı şu şekildedir: Bölüm 2’de görel kamera pozisyon tespiti problemi literatür çalışmaları irdelenmiştir. Görel kamera pozisyonu tespiti problemi Bölüm 3’de ayrıntılı olarak açıklanmış, önerilen $1-1$ çatı Bölüm 4’de aktarılmıştır. Bölüm 5’de bu problemin çözümü için gerçekleştirdiğimiz deneyler ve Bölüm 6’da deneylerde elde edilen başarımlar sunulmuştur. Bölüm 7’de durum çalışması olarak geliştirilen yaş sınıflandırması ağlarından bahsedilmiştir. Tezden elde edilen sonuçlar ve gelecekte yapılabilecek çalışmalar Bölüm 8’de anlatılmıştır.

2. LİTERATÜR ARAŞTIRMASI

Görelî kamera pozisyon tahmini problemine yönelik güncel öncü çözümlerin tamamında öğrenilmiş seyrek yöntemler kullandığından, bu bölümde yalnızca bu tür yaklaşımlar üzerinde durularak, tartışmanın özlu olması tercih edilmiştir. 3 boyutlu nokta bulutlarının sınıflandırılması ve segmentasyonu sorunuyla başa çıkmak için, Qi ve arkadaşları [17] PointNet adında bir derin öğrenme mimarisi sunmaktadır. PointNet mimarisi, eş zamanlı çalışan iki paralel daldan oluşmaktadır. Birinci dal, nokta bulutunun sınıfını tahmin ederken, diğeri bulut içindeki her bir noktanın bireysel segmentini tahmin etmektedir. Bu iki dalda da nokta bulutlarının küme türündeki 3 boyutlu koordinatlarını işleyebilen temel bileşenler olarak ortak ağırlıklı çok katmanlı algılayıcılar (ÇKA) kullanılmaktadır. PointNet, her ne kadar 3 boyutlu bulut segmentasyonu ve sınıflandırması için geliştirilmiş olsa da stereo görüntülerin birbirlerine göre konumlarının tahminini gerçekleştiren mimariler için ilham kaynağı olmuştur; zira her iki problem de küme türü girdiler içermektedir. PointNet'in girdi sıralamalarından bağımsız çalışan yapısı küme tipi girdiler kullanarak gürbüz derin öğrenme mimarileri geliştirmeyi amaçlayanlar için özellikle cazip olmuştur.

Yi ve arkadaşları tarafından önerilen PointCN mimarisi [4], görelî kamera pozisyonu tahmini üzerine yapılan sonraki çalışmaların çıkış noktası olmuştur. PointCN mimarisi kendisinin devamı olan başka bir makalede [5] ilk kez PointCN olarak adlandırılmaya başlanmış ve takip eden çalışmalarda da bu isimlendirme korunmuştur. PointCN'in böyle adlandırılmasında, ilham aldığı PointNet [17] mimarisinin ismiyle uyum ön plandadır. PointCN'nin giriş yapısı, PointNet'e benzerdir. PointNet buluttaki her noktanın 3 boyutlu nokta koordinatlarını kullanırken, PointCN varsayılan öznitelik eşleşmelerinin 4 boyutlu birleştirilmiş ve normalleştirilmiş koordinatlarını ve isteğe bağılı olarak varsayılan öznitelik eşleştirmelerine ait yardımcı bilgileri kullanmaktadır. Her iki mimari de küme yapısında girdiler işleyerek girdi sıralamasından bağımsız olarak sonuç üretmektedir. PointCN, varsayılan giriş eşleşmelerini tek seferde doğru eşleşme ya da hatalı eşleşme olarak sınıflandıran bir $n-n$ ileri beslemeli sinir ağıdır. Bunu yapmak için, sıralamadan bağımsız çalışan ardışık ÇKA katmanlarını kullanır. Mimaride kalıntı bağlantılar [7] kullanılarak gradyan kaybolması minimize edilir. PointCN'nin en etkileyici kısmı, Bağlam Normalizasyonu (BaN) katmanlarıdır. BaN katmanları, ağa eğitilebilir parametre eklemeyen, mimarinin performansını yükseltir. Girdi olarak kullanılan varsayılan eşleşmeleri etiketlemek için "zayıf etiketleme" adı verilen bir yöntem kullanılır. Zayıf etiketlemede, varsayılan eşleşmeler, öznitelik konumları ile görüntü çiftleri arasındaki epipolar çizgilere olan normalleştirilmiş mesafelere göre etiketlenir. Bu mesafeler belirli bir eşik değerini aşmıyorsa, varsayılan eşleşmeler doğru eşleşmeler olarak; aksi takdirde hatalı eşleşmeler olarak etiketlenir. Zayıf etiketleme yöntemi kullanılarak etiketlenmiş veri setleri, PointCN ve PointCN'i örnek alan tüm sonraki çalışmalar tarafından kullanılmıştır.

PointCN’de, eğitim sırasında iki farklı kayıp fonksiyonu kullanılır. Sınıflandırma kaybı (*sin_kayıp*), varsayılan eşleşmelerin gerçek etiketleri ile ağ tarafından üretilen sınıflandırma skorları arasındaki farktan elde edilen kayıptır. İkinci kayıp, her bir görüntü çiftinin gerçek temel matrisi ile ağdan hesaplanan temel matris arasındaki fark kullanılarak hesaplanan temel kayıp (*temel_kayıp*)’tır. Bu kayıp, ağın eğitimi sırasında, şans eseri epipolar çizgiler üzerinde bulunan yanlış eşleşmeleri ayırt edebilmeyi sağlaması açısından faydalıdır. Temel matrisi türevlenebilir bir şekilde hesaplamak için, 8-nokta algoritmasının [12] ağırlıklı-türevlenebilir yeniden formülasyonu [18] kullanılır. Her iki kayıp da eğitim sırasında hesaplanır, ancak başlangıç iterasyonlarında yalnızca *sin_kayıp* ağın parametrelerini güncellemek için kullanılır. Belirli bir sayıda iterasyondan sonra, *temel_kayıp* da geri yayılım aşamasında biriktirilir ve model parametreleri her iki kayıp kullanılarak güncellenir. Bu tür bir eğitim stratejisi, model parametrelerinde kontrolsüz güncellemelerin oluşmasını ve eğitim sürecinin başarısız olmasını engeller. Sonuçları iyileştirmek için RANSAC ile son işleme de uygulanmıştır. RANSAC yardımıyla sonuçlar iyileştirilse de daha önce tartışıldığı gibi tüm görüntü çiftlerinden minimum performans elde etmek için bir tekrarlamaya sınırı belirlemenin kolay olmaması nedeniyle, tekrarlamalı algoritmalar ile her görüntü çiftinde başarılı bir şekilde çalışan gürbüz bir çözüme ulaşmak zordur.

Zhang ve arkadaşları, tek seviyeli mimaride olan PointCN’i çok seviyeli bir mimariye dönüştüren Order-Aware Network’u (OANET) geliştirmiştir [5]. OANET, PointCN mimarisi ile ilgili iki önemli eksikliğe işaret etmektedir. İlk eksiklik olarak, BaN işleminin global bağlamı yeterince kodlayamadığı ileri sürülmektedir. İkinci eksiklik ise, ÇKA’ların her bir noktaya ayrı ayrı uygulanması nedeniyle yerel bağlamın kodlanmasının eksikliği olarak belirtilmiştir. PointCN’nin 12 katmanlı yapısı, OANET’in üst seviyesinde korunurken, bu eksiklikleri gidermek ve üst seviyedeki PointCN mimarisinin performansını artırmak için paralel bir alt seviye eklenmiştir. OANET’in alt seviyesini oluşturmak için üç farklı ağ bileşeni tanıtılmıştır. İlk bileşen, [19] çalışmasından ilham alınan Türevlenebilir Havuzlama Katmanıdır (Differentiable Pooling Layer). Bu katman, sıradan havuzlama katmanına benzer, ancak her bir düğüm için sabit bir atama kullanmak yerine, alt düzeydeki 500 küme verisini seçmek için öğrenilebilir bir atama matrisi kullanır. Türevlenebilir Havuzlama Katmanı yardımıyla yerel bağlam çıkarımı yapmak mümkün olmaktadır. Tanıtılan ikinci ağ bileşeni, işlenen veriyi alt seviyeden üst seviyeye düzgün bir şekilde geri iletilmesini sağlayan Türevlenebilir Ters Havuzlama Katmanıdır. Seçilen veri kümeleri, alt seviyeye yerleştirilen ve tasarlanan son ağ bileşeni olan Sıra-Farkındalıklı Filtreleme Bloğu (Order-Aware Filtering Block) üzerindeki ÇKA’lar kullanılarak filtrelenebilir. Bu blok, veri kanal boyutunda filtreleme yapmasının yanı sıra, uzaysal boyutta da filtreleme yaparak OANET’in global bağlamı kodlama yeteneğini arttırmaktadır. OANET’in yeteneklerini sunmadan önce, yazarlar PointCN ile büyük veri kümeleri üzerinde deneyler yapmış ve veri seti boyutunun ağın maksimum yeteneklerine ulaşabilmesi için kritik olduğunu göstermiştir. Daha sonra OANET performansını en üst düzeye çıkarmak için karşılaştırma çalışmaları yapılmıştır. Ağ seviyelerinin sayısını ikiden üçe çıkarmak performansı artırmaya da ağın parametre sayısını önemli ölçüde değiştirmeden aşamalı hale getirilmesinin ağın performansını arttırdığı gözlemlenmiştir. Yazarlar geometrik olarak anlamsız *temel_kayıp* yerine geometrik kaybın (*geo_kayıp*) kullanılmasının da OANET’in performansını arttırmaya yardımcı olduğunu savunmaktadır. Yazarlar ayrıca Türevlenebilir Havuzlama

Katmanı tarafından küme seçimlerinin görüntü çiftleri üzerindeki etkisini göstermiştir. Aynı görüntü çifti için, farklı kümelerin normalleştirilmiş öznitelik konumları açısından farklı yerel bağlamları kodlama yeteneğine sahip olduğu gösterilmiştir. Ayrıca, farklı görüntü çiftleri için, aynı kümenin normalleştirilmiş öznitelik konumları açısından benzer yerel bağlamları kodlama yeteneğine sahip olduğu gösterilmiştir. Sınırlı sayıda görüntü çiftine ait sonuçlar sunulmuş olsa da bu deneysel bölüm etkileyicidir çünkü bu bölüm OANET çalışması ile PointCN mimarisine yapılan eklemelerin gerçekten faydalı olduğunu göstermektedir.

Jin ve arkadaşları [20], hem stereo hem de çoklu-görüş yeniden yapılandırma için bir karşılaştırma çalışması sunmaktadır. Görüntü yeniden yapılandırma sürecinin adımları olan öznitelik çıkarma, eşleştirme ve pozisyon tahmini gibi adımlar için cazip performanslara sahip yöntemler geliştirmenin, bu yöntemlerin yapılandırma sürecinin kendisinde faydalı olacaklarını garanti etmediğini savunmaktadırlar. PointCN [4], hem stereo hem de çoklu-görüş görevleri için bir dışsal hatalı eşleşme ön filtreleme operatörü olarak tanıtılmıştır. Göreli kamera pozisyon tahmini sürecinin bir parçası olan farklı parametrelerin en iyi değerleri araştırılmıştır. Stereo yapılandırmada kullanılacak bir görüntü çifti için minimum ortak görünürlük eşiği sunulmuştur. Tüm görüntü çiftlerinden 8K öznitelik eşleşmesi kullanmanın maliyet ve performans arasında iyi bir uzlaşma noktası olduğunu ve 2K öznitelik eşleşmesinin ise daha ucuz bir alternatif olduğunu savunmaktadırlar.

CLNET [1], hatalı eşleşmeleri ayıklamak için eşleşme budama yönteminin tercih edildiği bir ağıdır. Mimaride, yerelden globale uzanan uzlaşma öğrenmesi yoluyla eşleşmeleri kademeli olarak elemek için birden fazla budama bloğu kullanılmıştır. Mimarilerini, küme türü girdilere sahip bir diğer problem olan doğru çizgi tespiti probleminde test etmişlerdir. Budama bloklarının yerel uzlaşma öğrenme kısmında, aktivasyon haritalarındaki en benzer eşleşmelerin verilerini yeni kanallar olarak ekleyerek her bir eşleşme için yerel bağlam etkileşimini arttırmışlardır. Ek kanalları işlemek için halka biçimli konvolüsyonlar kullanmışlardır ve her halka grubunun boyutu deneysel olarak belirlenmiştir.

Liu ve arkadaşları [21], LMCNET adında bir mimari önermişlerdir. LMCNET’te, Laplace Hareket Uyumlandırma (Laplacian Motion Fitting) olarak adlandırılan hareket uyumunun yeni bir formülasyonu sunulmaktadır. Ağlara, küresel hareket ve yerel hareket uyumlarını yakalamak için, Uyum Artık Katmanı (Coherence Residual Layer) ve Yerel Uyum Katmanı (Local Coherence Layer) olarak adlandırılan özel bloklar yerleştirilmiştir. Ayrıca doğru eşleşme ve hatalı eşleşme verilerinin hareket artık normlarının dağılımlarını ve doğru eşleşmeleri hatalı eşleşmelerden ayırt etmek için kullandıkları eşiği de sunmaktadırlar. Mimari yapılarında, Öklid mesafelerine göre oluşturulan sabit grafiğin ve seyrek anlamsal benzerlikler kullanılarak oluşturulan dinamik grafiklerin performanslarını karşılaştırmışlardır.

MS²DG-Net [22] çalışmasında, Liu ve arkadaşları, seyrek eşleşmelerin seyrek anlamsal benzerliklere sahip olduğunu savunmaktadırlar. Bu benzerlikleri kullanarak, dinamik grafikler oluşturmaktadırlar. İnsan eşleştirme sürecinden ilham almışlardır; bu süreçte, Öklid mesafeleri benzer olanlar yerine, anlamsal benzerliklere sahip olan eşleşmeler gruplandırılmaktadır. Önerdikleri mimaride, çok ölçekli bilgiyi birleştirmek için

birden fazla dinamik olarak oluşturulmuş grafik dahil edilmiştir.

NCMNET [23] çalışmasında, tutarlı komşu eşleşmelerini bulma zorluğunu ele almak için grafik tabanlı bir yöntem önerilmiştir. Mimari, uzun menzilli bağlamları keşfetmeye olanak tanıyan k -en yakın komşu algoritmasını kullanarak, uzaysal, özellik uzayı ve global komşu eşleşmelerini yakalamaktadır. Bu üç tür komşu gömüsünü işlemek için, model CLNET [1] mimarisine benzer iki budama bloğu kullanır. Yazarlar özellikle RANSAC algoritması ile ayıklama sonrası ağlarının performansında bir iyileşme olmadığını vurgulamışlardır.

ConvMatch [2], n - n çatısında eğitilmiş en başarılı ağlardan birini sunmaktadır. Ağlarını, seyrek hareket vektörlerinin ve yoğun hareket alanlarının enterpolasyon ve örnekleme ile birbirine dönüştürülebileceği gerçeğine dayandırmışlardır. Düzenleme, düzeltme ve geri kazanım işlemleri, Düzeltici Evrişim Katmanı (Rectifying Convolution Layer) olarak adlandırdıkları blok ile gerçekleştirilmiştir ve bu bloktan ağda ardışık olarak birden fazla kullanmışlardır. Düzenleme işlemi için, varsayılan hareket vektörleri yoğun hareket alanları oluşturmak üzere görüntü üzerindeki ızgaralarda gruplandırılmaktadır. Yoğun hareket alanlarındaki hatalı eşleşmelerden kaynaklanan hataları düzeltmek için ESA kullanılmıştır. Kendi çalışmalarının, görelî kamera pozisyon tahmini problemleri için ESA bloklarının ilk kullanımı olduğunu savunmaktadırlar. Doğru hareket vektörleri düzeltilmiş hareket alanlarından elde edilmektedir. Optimum ızgara boyutunu ve ardışık düzeltici evrişim katmanlarının sayısını bulmak için tarama çalışmaları yapılmıştır.

Zhang ve arkadaşları, mimarilerinin [2] performansını ConvMatchimp [3] olarak adlandırdığımız bir çalışma ile geliştirmişlerdir. ESA'ların komşu algılama mekanizması nedeniyle, yüzeysel katmanların yeterince küresel bilgi alamadığını savunmaktadırlar. Bunu aşmak için, kanal bazlı küresel ortalama havuzlamanın küresel bağlamı yaymak amacıyla uygulandığı Küresel Bilgi Enjeksiyonu (Global Information Injection) bloğunu tanıtmışlardır. Ayrıca, hareket alanının düzgünlük varsayımının her zaman geçerli olmadığını belirtmektedirler. Hareket alanındaki büyük süreksizlikleri daha iyi koruyan Çift Taraflı Evrişim (Bilateral Convolution) bloğunu bu sorunu çözmek için kullanmışlardır. Bu yeni tanıtılan blokları kullanarak, ESA bloklarının güncellenmiş bir versiyonunu oluşturmuşlardır. Ağlarının önceki versiyonunu [2] başarımlar olarak geride bıraktıklarını savunmaktadırlar.

Bu bölümde tartışılan literatürdeki mimariler, küme türündeki girdilerden hem küresel hem de yerel bağlamı yakalamayı amaçlamaktadır ve bu, n - n çatısı üzerinde gerçekleştirilmektedir. Görelî kamera pozisyonu tahmini problemine ait çözümlerin başarımlarının artırılması için bu çalışmada önerdiğimiz çatıya benzer çatıların öne sürülmesi cazip bir araştırma konusu olarak değerlendirilmektedir.

3. PROBLEM TANIMI

Görelî kamera pozisyon tahmininde amaç 3 boyutlu bir ortamı görüntüleyen iki kamera tarafından oluşturulan 2 boyutlu görüntüler kullanılarak kameraların birbirlerine göre pozisyonunu belirlemektir. Bir başka deyişle kameraların normallerine göre oluşturulan koordinat sistemlerini hizalamak için gerekli olan dönme ve öteleme miktarlarını tahmin etmektir. Bu süreç genellikle görüntü çiftlerinde benzer bölgelerin belirlenmesi ve bu bölgelerin koordinatları kullanılarak görüntülerdeki pikseller arasında geometrik bir ilişkinin kurulmasıyla başlar. Bu bölgeleri tanımlamak için, döndürme, ölçek ve bakış açısı değişikliklerine dayanıklı, kendine özgü seyrek öznitelik noktaları görüntü çiftlerinden çıkarılır. Görüntü çiftlerindeki en benzer öznitelikler, öznitelik tanımlayıcılarının öznitelik uzayındaki benzerliklerine göre eşleştirilerek varsayılan öznitelik eşleşmeleri oluşturulmuş olur. Görüntü çiftlerinin aynı kameralarla elde edilmesi zorunluluğu olmadığından eşleşmelerin görüntü koordinatları yerine eşleşmelerin normalleştirilmiş kamera koordinatları kullanılır. Bu varsayılan eşleşmeler şu şekilde ifade edilebilir:

$$C = [c_1; c_2; \dots; c_n] \in \mathbb{R}^{n \times 4}, c_i = (x_1^i, y_1^i, x_2^i, y_2^i), \quad (3.1)$$

burada (x_1^i, y_1^i) ve (x_2^i, y_2^i) , görüntüleyicilerin içsel parametreleri kullanılarak hesaplanan normalleştirilmiş kamera koordinatlarıdır.

Görelî kamera pozisyonu tahmini probleminde kullanılacak olan bir sinir ağı, varsayılan eşleşmelerden hatalı eşleşmeleri filtreleme yeteneğine sahip olmalıdır. Bu nedenle, sinir ağı bir ikili sınıflandırma problemini çözmelidir. Sinir ağı tarafından belirlenen doğru eşleşmeler, görüntü çiftleri için temel matrislerin minimum hatayla tahmin edilmesinde kullanılır. Bir görüntü çifti için hesaplanmış temel matris, görüntülerden birinde bulunan normalleştirilmiş öznitelik noktasının diğer görüntüde bulunacağı normalleştirilmiş konumun tespit edilmesinde yardımcı olur. Temel matrisler sayesinde, görüntü çiftleri arasındaki dönme ve öteleme açıları tam olarak, aralarındaki öteleme miktarı bilinmeyen bir ölçek dahilinde belirlenebilir. Sinir ağları aracılığıyla temel matris hesaplanması, şu şekilde ifade edilebilir:

$$\hat{z} = f(C), \quad (3.2)$$

$$\hat{w} = \text{DDO}(\tanh(\hat{z})), \quad (3.3)$$

$$\hat{E} = g(\hat{w}, C), \quad (3.4)$$

burada $\hat{z}$, eşleşmeler için ağların çıktıları, $\hat{w}$ ise eşleşmelerin $[0, 1)$ aralığına eşlenen doğru eşleşme puanları ve $\hat{E}$ ise tahmin edilen temel matristir. 8 doğru eşleşme, temel matrisi hesaplamak için yeterli olsa da, [4]'de kullanılan ağırlıklı 8-nokta algoritması, her eşleşmenin doğru eşleşme puanlarını çıkaran ağlarda kullanılan Eşitlik 3.4'teki g fonksiyonu için daha uygun bir alternatiftir.

Görüntü çiftlerinden elde edilen varsayılan eşleşmeleri doğru/yanlış eşleşme olarak etiketlemek için, literatürde yaygın olarak zayıf etiketleme [4] kullanılmaktadır. Zayıf etiketlemede kameraların içsel parametreleri kullanılarak epipolar çizgiler hesaplanır. Bu epipolar çizgilere uzaklığı 10^{-4} eşik değerinden daha küçük olan öznitelik eşleşmeleri doğru eşleşmeler olarak etiketlenir. Ağları yalnızca *sin_kayıp* kullanarak eğitmek mümkün olsa da, ağıın gerçek temel matrisleri ile tahmin edilen temel matrisler arasındaki farka göre hesaplanan *temel_kayıp* gibi regresyon kayıplarını kullanmak da faydalıdır. Bu tür regresyon kayıplarının kullanılması, epipolar çizgilere yakın olan hatalı eşleşmeleri elemeye yardımcı olur. Bu nedenle, bileşik bir kayıp elde etmek için birden fazla kayıp birleştirilir, örneğin Eşitlik 3.5’de olduğu gibi

$$kayıp = \alpha * sin_kayıp + \beta * temel_kayıp \quad (3.5)$$

değerleri deneysel olarak belirlenen α ve β katsayıları ile kayıpları ağırlıklandırmak mümkündür.

Daha üstün bir regresyon kaybı olan *geo_kayıp*, [24, 25], ilk olarak Zhang ve arkadaşları tarafından [5], *temel_kayıp*’a alternatif olarak kullanılmıştır. *geo_kayıp* hesaplamasında, her görüntü çiftinin ilk görüntüsü referans görüntü olarak seçilir ve bu görüntü üzerinde sanal özellik noktaları iki boyutlu bir ızgara üzerinde eşit aralıklarla dağıtılır. Bu sanal noktalar, daha sonra her görüntü çiftinin ikinci görüntüsüne hem gerçek temel matris hem de tahmin edilen temel matrisler kullanılarak yansıtılır. *geo_kayıp*, bu yansıtma konumları arasındaki farklara dayalı olarak hesaplanır. Bu tür bir kayıp hesaplama yöntemi, ağların yalnızca varsayılan eşleşmelerin koordinatlarını değil, aynı zamanda görüntü çiftlerindeki herhangi bir koordinat konumunu da doğru bir şekilde yansıtmayı öğrenmesini sağladığı için daha anlamlıdır ve literatürde [5] daha başarılı ağların eğitilmesini sağladığı bilinmektedir. Göreli kamera poz tahmini problemi için ilk kullanıldığından bu yana [5], *geo_kayıp* literatürde yaygın olarak benimsenmiş ve *temel_kayıp*’un yerini almıştır. *geo_kayıp* aslında *temel_kayıp*’ın bir uzantısı olduğundan, ağırlıklı 8-nokta algoritmasının türevlenebilir yapısı *temel_kayıp*’ın geri yayılımını kolaylaştırdığı gibi *geo_kayıp*’ın da geri yayılımını kolaylaştırmaktadır.

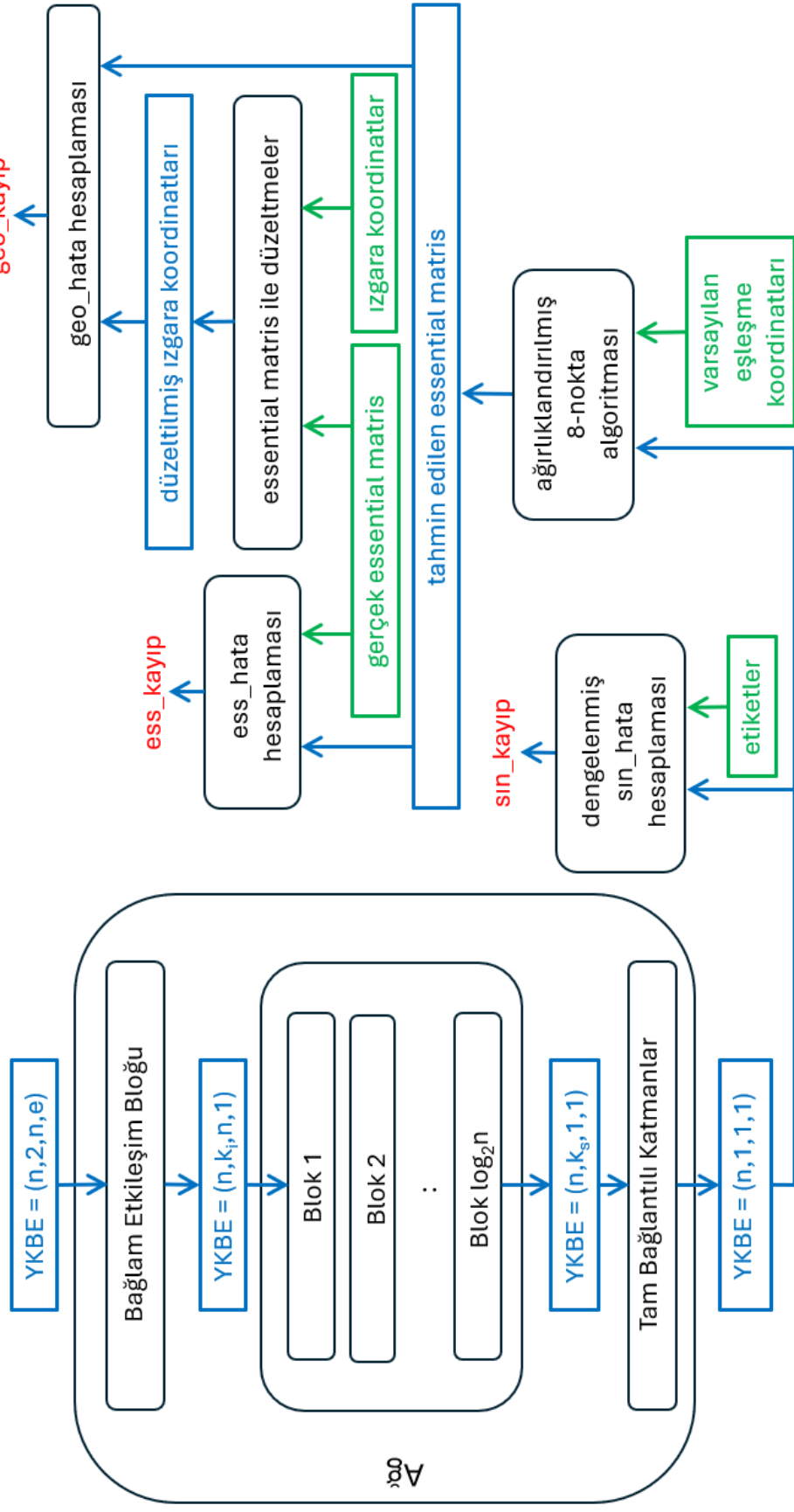
4. 1-1 ÇATI

Görelî kamera pozisyon tahmini problemini çözmek için üstün ağların eğitilebileceği ideal bir çatının iki temel özelliğe sahip olması gerektiğini savunuyoruz:

- Çatı, bir görüntü çiftindeki her bir eşleşmeye ait veri ile görüntü çiftinin diğer eşleşmelerine ait veriler arasındaki etkileşimi en üst düzeye çıkarabilmelidir.
- Çatı, görüntü çiftindeki her bir eşleşme için özel olarak yapılandırılmış bağlam sağlayabilmelidir.

Bu tez kapsamında önerilen ve Şekil 4.1’de gösterilen çatı, bu özellikleri sağlamak-tadır. Şekil 4.1’de, yığın (Y), kanal (K), boy (B), en (E) boyutlarına sahip aktivasyon haritaları mavi renkte gösterilmiştir. Ayrıca kayıpların hesaplanmasında kullanılan ger-çek değer değişkenleri yeşil renkte, ağ blokları siyah renkte ve kayıplar kırmızı renkte gösterilmiştir. Yığının her bir örneği, bir varsayılan eşleşmenin verilerini içerirken, her yığın yalnızca bir görüntü çiftine ait tüm varsayılan eşleşmelerin verilerini içermekte-dir. İlk özelliği gerçekleştirmek için, Şekil 1.1’te gösterilen girdi yığın yapısı kullanıl-mıştır. Görüntü çiftlerindeki her bir eşleşme verisi tekrarlanarak her bir yığın örneğinin birinci kanalına yerleştirilir. Yığın örneğinin diğer kanalı ise görüntü çiftine ait bağlam verilerinden oluşmaktadır. Bu kanallar, Şekil 4.1’te gösterildiği gibi 1-1 çatısındaki ağ-ların ilk bloğu olan Bağlam Etkileşim Bloğunda (BEB) evrişim işlemine tabi tutulur. Bu, eşleşme verileri ve bağlam verileri arasındaki etkileşimin, ilk bloğun çıkış akti-vasyon haritasından itibaren başlamasını sağlar. Ek olarak, her bir eşleşme verisinin ve bağlam verisinin bireysel evrişimleri bu etkileşimi daha da artırır. $n-n$ çatısındaki ağlarda, genel bağlam olarak adlandırılan birinci özellik genellikle BaN katmanlarının havuzlama işlemlerinin gerçekleştirdiği ÇKA blokları ile sağlanır. Teorik olarak, ha-vuzlama işlemleri kullanılarak bağlam bilgisini dağıtmak mümkündür. Ancak pratikte, eklenen her ÇKA bloğu ağın derinliğini artırarak gradyan kaybolması sorunlarına yol açmaktadır. Ayrıca, havuzlama işlemleri ile dağıtılan bağlam verisi miktarı sınırlıdır. Bu nedenle, BEB’in çıkış aktivasyon haritasının ve 1-1 çatısındaki ağların sonraki akti-vasyon haritalarının, $n-n$ çatısındaki ağlara kıyasla genel bağlam açısından daha zengin içeriğe sahip olduğu bu tez kapsamında ileri sürülmektedir.

1-1 çatı ile, bir görüntü çiftindeki her bir eşleşme için bağlam kanalının satırlarını sıra-larken farklı yöntemler kullanmak mümkündür. Bununla daha önce bahsedilen ideal çatının ikinci özelliğini sağlanmaktadır. Giriş bağlam kanallarının satırlarını sırala-yarak, birbirine yakın yerleştirilen satırların etkileşimine öncelik verilebilir. $n-n$ çatı literatüründe, bu özellik yerel bağlam kavramına karşılık gelir. $n-n$ çatısındaki ağlar, yerel bağlam etkileşimini artırmak için çeşitli ağ blokları türlerini kullanır. Ancak, bu blokların çoğu, yerel bağlamı dikkate almayan ilk blokların ardından yerleştirilir. Bu



Şekil 4.1: Önerilen I-I çatının yapısı.

nedenle, sonraki blokların yerel bağlam çıkarımı yapma yeteneği ilk bloklar tarafından sınırlandırılmış olmaktadır. Buna karşılık, $I-I$ çatı, herhangi bir operasyon gerçekleştirilmeden önce bağlam kanallarının satırlarının sıralanmasına olanak tanır. Böylece sıralanmış bu bağlam satırlarıyla yerel bağlam etkileşimi ağlardaki işlemlerin en başından başlayarak en üst düzeye çıkarılmış olur.

Bu tezde tartışılan $I-I$ ağlarda, BEB'den gelen çıkış aktivasyon haritaları, her biri aktivasyon haritalarının yükseklik boyutunu yarıya indiren bloklardan geçirilerek işlenir. Dolayısıyla, yığın hacminin başlangıçtaki yükseklik boyutu, 2'nin bir kuvveti olmalıdır ve bu da blok sayısının $\log_2 n$ olmasını gerektirmektedir. Aktivasyon haritaları bu bloklarda işlenirken, aktivasyon haritalarındaki zenginleştirilmiş verileri barındırmak için aktivasyon haritalarındaki kanal sayısı artırılır. Kanal sayısının artırılma şekli, ağ için kritik bir meta parametredir. Ayrıca, başlangıç aktivasyon haritalarındaki kanal sayısı k_i ve son aktivasyon haritalarındaki kanal sayısı k_s de önemli meta parametrelerdir. $I-I$ çatısı üzerine kurulmuş tüm ağlarda tüm işleme bloklarından sonra, tek bir görüntü çiftindeki tüm n eşleşme için doğru eşleşme puanları elde etmek üzere, ardışık 2 tam bağlantılı katman kullanılır. Geliştirme aşamasında hem düzeltilmiş doğrusal operatör (DDO) [26] hem de kaçaklı düzeltilmiş doğrusal operatör (KDDO) [27] aktivasyonları test edilmiş ve KDDO için daha düşük performans sonuçları elde edildiğinden, $I-I$ çatısında eğitilen tüm ağlarda DDO aktivasyonları tercih edilmiştir.

Sunulan tüm $I-I$ ağlardaki blokların sayısı Şekil 4.1'te gösterildiği gibi $\log_2 n$ olsa da, üstün performansların elde edilip edilemeyeceğini görmek için daha fazla blok içeren derin ağlarla da deneyler yapılmıştır. Ancak, aktivasyon haritalarının boyunu değiştirmeyen işleme blokları ekleyerek ağların derinliğini artırmak elde edilen performans verilerini iyileştirmemiştir. Aksine, $I-I$ çatısındaki ağları daha derin hale getirmek oluşturulan ağların problemin çözümünü öğrenmesini imkânsız hale getirmektedir.

Bir görüntü çiftinin eşleşmeleri için doğru eşleşme puanları *sin_kayıp*, *temel_kayıp* ve *geo_kayıp* kullanılarak Şekil 4.1'te gösterildiği gibi hesaplanır. $I-I$ çatısındaki kayıp hesaplamaları, $n-n$ çatısındaki kayıp hesaplamalarına [5] benzerdir. Tek fark, $I-I$ çatısında hesaplanan kayıpların yalnızca bir görüntü çiftine ait verileri içerirken $n-n$ çatısındaki kayıp hesaplamalarının bir yığın içindeki çok görüntü üzerinde gerçekleştirilmesidir. Bu durum, ağların genelleme yeteneğini sınırlayabileceğinden $I-I$ çatısı için bir dezavantaj olarak kabul edilebilir. Yığınlardaki görüntü sayısını artırmak genelleme yeteneğini geliştirebilir, ancak bu, donanıma sığamayacak kadar büyük aktivasyon haritaları oluşturacaktır. Beklenenin aksine $I-I$ çatısındaki ağlardan elde edilen sonuçlar incelendiğinde, genelleme yeteneğinde ciddi bir sınırlama gözlemlenmemiştir. Bununla birlikte birden fazla görüntü çiftini içeren yığınlar kullanarak $I-I$ ağlar eğitmek ağ performansları açısından irdelenmelidir.

$I-I$ çatısında kayıpların hesaplanmasında kullanılan ağ elemanları hesaplama grafiğinin bir parçası olsalar da öğrenilebilir parametrelere sahip değildir. İleri geçişler sırasında tüm kayıplar hesaplanmış olsa da ağ parametrelerini optimize etmek için yalnızca seçilen kayıplar geri yayılma ile ağ parametrelerine aktarılmaktadır. Geri yayılma için yalnızca *sin_kayıp* seçilebilir veya performansı artırmak için *sin_kayıp*'a *temel_kayıp* veya *geo_kayıp* eklenebilir.



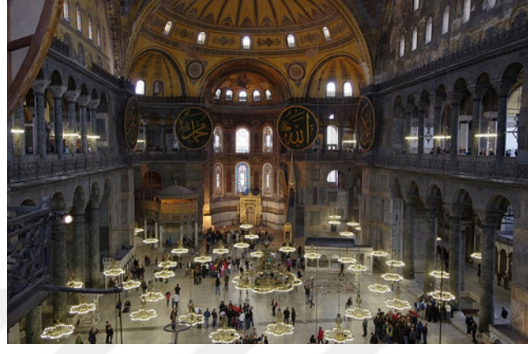
5. DENEYLER

Bu bölümde $1-1$ çatısında eğitilen ağların performansını, $n-n$ çatısında eğitilen ağlarla karşılaştırmadan önce metodolojimizin önemli unsurları paylaşılmıştır.

5.1 Veri Setleri ve Seyrek Öznitelikler

$n-n$ çatısına ait literatürle uyumlu olmak amacıyla OANET [5] tarafından kullanılan görüntü dizilerini olduğu gibi kullanmayı tercih ettik. Yahoo'nun YFCC100M veri setindeki [16] 72 ikonik turistik yapıya ait görüntü dizilerini kullanarak, Heinly ve arkadaşları bu yapıların 3 boyutlu rekonstrüksiyonlarını oluşturmuşlardır. OANET'te [5], bu dizilerin 67'si eğitim ve doğrulama için, 4'ü ise testler için kullanılmıştır. Görüntü dizilerinden görüntü çiftleri seçilirken seçilen çiftlerinin 3 boyutlu rekonstrüksiyonlarında minimum görsel örtüşmeye sahip olması kriteri göz önünde bulundurulmuştur. Bu kısıt, seçilen görüntü çiftlerinin ortak görüş alanlarında minimum sınırdan daha fazla 3 boyutlu noktanın bulunmasıyla sağlanmaktadır. Veri setlerinden örnek görüntüler Şekil 5.1- 5.3'de gösterilmektedir.

Kullanılan veri setlerindeki her görüntü çifti için varsayılan eşleşme sayısı değişkenlik göstermektedir. $n-n$ çatısında, eğitim yığınlarındaki örneklerin eşit boyutlara sahip olmasını sağlamak için her yığında en az varsayılan eşleşmeye sahip olan görüntü çifti referans çift olarak seçilir. Yığındaki tüm örneklerdeki eşleşme sayısının referans çiftin boyutuna eşit olması için her bir yığın örneğinden gerekli sayıda eşleşme rastgele olarak yığın dışı bırakılır. Böylelikle yığındaki tüm örneklerin boyutlarının eşit olması sağlanır. Önerdiğimiz $1-1$ çatısında farklı bir yaklaşım kullanılmıştır. Çatımızda her yığın yalnızca bir görüntü çifti içerdiğinden, her yığındaki her bir örneğin bağlam kanalı eşit boydadır. Bu nedenle, çatımız herhangi bir yığındaki herhangi bir örnek için bağlam kanalının satır sayısında bir azaltmaya ihtiyaç duymaz. Ancak, test ettiğimiz bazı $1-1$ ağların, aktivasyon haritalarının yükseklik boyutunun 2'nin bir kuvveti olmasını gerekliliği vardır. Bu gereklilik $1-1$ çatısındaki ağlarda kullanılan işleme bloklarının her birinin aktivasyon haritalarının boyunu yarıya düşürmesinden kaynaklanmaktadır. Bu tez kapsamındaki deneylerde, literatürdeki tüm $n-n$ ağlar ile bizim $1-1$ ağlarımız için adil bir karşılaştırma yapabilmek adına, tüm giriş aktivasyon haritalarının yükseklik boyutu $n = 2048$ olacak şekilde arttırılmıştır. Bu n seçimi, literatürdeki çalışmalar [20] ile uyumludur. Ayrıca, tekrarlanan satırların rastgele seçilmesinin bir veri artırma mekanizması sağladığını ve bu sayede eğitilen modellerin girdiyi ezberlemesinin önüne geçilerek modellerden daha yüksek performans elde edilebildiği savunulmaktadır. Bu tür bir veri artırma, *temel_kayıp* ve *geo_kayıp*'ta kullanılan ağırlıklı 8-nokta hesaplamaları için matematiksel bir sorun oluşturmamaktadır.



Resim 5.1: Kullanılan veri setlerinden Aya Sofya görüntü örnekleri.



Resim 5.2: Kullanılan veri setlerinden Londra Köprüsü görüntü örnekleri.



Resim 5.3: Kullanılan veri setlerinden Tac Mahal görüntü örnekleri.

5.2 Metrikler

Tahmin edilen temel matris kullanılarak bir görüntü çifti için görüntüler arasındaki dönme açıları tam olarak, öteleme ise ölçek belirsizliğiyle hesaplanabilir. Ötelemenin büyüklüğündeki belirsizlik nedeniyle, görelî kamera pozisyon tahmini literatürü, hata metrikleri olarak tahmin edilen dönüş ve tahmin edilen öteleme açıları ile bu değerlerin gerçek değerleri arasındaki farkı kullanmaktadır. Bu metrikler arasında, her görüntü çifti için daha büyük olan değer, tüm performans değerlendirmelerinde referans hata olarak kullanılır. Bu referans hatalarına dayanılarak, testlerde kullanılan tüm görüntü çiftleri için normalleştirilmiş bir kümülatif eğri oluşturulur. Eğri altında kalan alan (EAKA), 5, 10 ve 20 derece için maksimum eşiklerde hesaplanır ve ortalama doğruluk (oD) ile ilişkilendirilir. EAKA hesaplamalarında, bazı çalışmalar [4, 5, 22] Riemann toplamları yakınsaması yöntemini kullanırken, sonraki çalışmalar [1–3, 21, 22] Yamuk yakınsaması yöntemini kullanmıştır. Biz de bu tezdeki deneylerde başarıyı daha iyi yansıtmayı sebebiyle yamuk yöntemini EAKA hesaplamalarında kullanmayı tercih ettik. Eşikler için hesaplanan oD değerlerini oD5, oD10 ve oD20 olarak adlandırılmıştır. Her ne kadar oD10 ve oD20 sonuçlarını rapor etsek de performans tartışmalarımızda ana odak noktamız oD5'tir. Ağlar tarafından tahmin edilen açılardaki hataların 5 derece ve altında olmasının anlamlı olduğu daha büyük hata açılarının hedef olarak seçmenin anlamlı olmadığı değerlendirilmiştir. Paylaştığımız sonuçlar arasında, giriş aktivasyon haritalarında herhangi bir rastgelelik varsa, testleri üç kez gerçekleştirdik ve bu üç testin ortalamasını rapor ettik. Önceki çalışmalarda doğru eşleşmelerin verilerini yinelemeli yöntemlerle iyileştirilmesi sonucunda tüm yöntemler için benzer iyileşmeler gerçekleştiğinden bu çalışmada eğitilen ağların performansını yinelemeli yöntemlerle arttırmak tercih edilmemiştir.

5.3 Uygulama Detayları

Bugüne kadar yapılan tüm görelî kamera pozisyon tahmini çalışmaları, ağların önceden belirlenmiş sabit bir iterasyon sayısı için eğitilmesini öngören PointCN [4] metodolojisine bağlı kalmıştır. Bununla birlikte bu tezde, farklı kayıp yakınsama dinamikleri ile karakterize edilen iki farklı çatıya ait ağların karşılaştırılması amaçlanmıştır. Dolayısıyla, sabit bir iterasyon sayısına bağlı kalmak yerine daha dinamik bir eğitim yaklaşımı benimsenmiştir. Tüm ağların eğitimleri, ağların doğrulama *geo_kayıp*'ında önemli bir azalma gözlemlenmeyene kadar sürdürülmüştür. Böylelikle her bir ağı ait kaybın tam olarak yakınsaması ve ağı performansının en iyi seviyeye ulaşabilmesi için yeterli fırsat sağlanmış olmaktadır. Bu yaklaşım ile farklı ağların kendine özgü yakınsama hızları göz önünde bulundurulmaktadır. Sabit sayıda devirli eğitimin sınırlamaları pasivize edilerek, nihayetinde daha doğru ve güvenilir performans değerlendirmelerine ulaşılmaktadır. *n-n* ağlar için, ilgili makalelerde sunulan öğrenme oranları ile öğrenme süreçleri başlatılmıştır. Doğrulama *geo_kayıp* sabit hale geldikçe, öğrenme oranı 10'da 1'ine düşürülerek öğrenme süreci devam ettirilmiştir. Başlangıç öğrenme oranı olarak 10^{-2} kullanıldıktan sonra aynı öğrenme oranı azaltma stratejisi *1-1* ağlar için de uygulanmıştır. *n-n* ağlar için literatürde ADAM [28] optimizasyon algoritması kullanılırken, *1-1* ağlarımız için SGD [29] optimizasyon algoritmasının daha iyi performans gösterdiği gözlemlenmiştir. Bunu, *1-1* çatısındaki kayıp yüzeylerinin yüksek

düzlemselliği sahip olmalarına bağlıyoruz. Bu tip durumlarda SGD optimizasyonunun daha yüksek performansa erişebildiği örnekler bilinmektedir [30].

1-1 ağlar için yığın boyutu 2048 olarak seçilmiştir ve her yığın yalnızca bir görüntü çiftine ait verileri içermektedir. Yığındaki her bir örnek, yalnızca o görüntü çiftinden bir eşleşme ile ilgili verileri içermektedir. Tüm *n-n* ağlar için ise yığın boyutu 32 olarak kullanılmıştır, bu da her yığının 32 görüntü çifti içerdiği anlamına gelmektedir. Her iki çatıda da geri yayılım sırasında kayıpların birikimi için kullanılan katsayılar, *n-n* çatısında kullanıldığı gibi *sin_kayıp* için 1, *temel_kayıp* için 0,1 ve *geo_kayıp* için 0,5 olarak kullanılmıştır. Eşleşme koordinatlarının yanında yardımcı bilgi kullanılmayan deneylerde giriş aktivasyon haritasının genişliği dört olarak kullanılmıştır. Yan bilgilerin eklendiği deneylerde ise varsayılan eşleşmelerin karşılıklı olarak eşleşmiş olmaları bilgisi ve Lowe'un [7] oran testi sonuçları kullanılarak, giriş aktivasyon haritalarının genişliği altıya çıkarılmıştır.

Her iki çatının uygulanması için Pytorch XLA [31] kütüphanesini kullanılmıştır. Pytorch XLA, TPU donanımı için Pytorch işlemlerini alt seviyede gerçekleştirilmesine imkân veren bir kütüphanedir. Alt seviye TPU uygulamaları eklenen PyTorch fonksiyonlarının sayısı sürekli artmakta olup, kullanıcılar ek fonksiyonların alt seviyede uygulanmasını talep edebilmektedir. Kullanıcılar tarafından kullanılan fonksiyonlar henüz TPU donanımı için alt seviyede oluşturulmamışsa, Pytorch XLA, bu işlemleri daha yavaş bir şekilde uygulamak için aktivasyon haritalarını CPU'ya aktararak operasyonları gerçekleştirmektedir. *n-n* ağları, sırasıyla 8 GB ve 16 GB yüksek bant genişlikli belleğe sahip 8 çekirdekli TPuv2 ve TPuv3 cihazlarında eğitildi. Diğer yandan, *1-1* ağları, her bir çekirdeği 32 GB yüksek bant genişlikli bellek ile donatılmış 4 çekirdekli TPuv4 cihazları kullanılarak eğitildi. Tüm deneylerde, TPU'ların çoklu çekirdek çalışma modu kullanıldı. Ancak, dönüştürücü tabanlı *1-1* ağları, tek TPU cihazı içinde model seviyesinde paralellik sağlayan tek program çoklu veri (single program multiple data) modu ile eğitildi.

6. BAŞARIMLAR

6.1 n-n Çatısı Referans Modellerine Ait Başarımlar

1-1 çatı ile eğitilen ağların performansını değerlendirmeden önce, her iki çatı için de ortak olan bazı önemli meta parametrelerin incelenmesine öncelik verilmiştir. Farklı meta parametrelere sahip PointCN, OANET ve OANETiter ağları eğitilmiştir. Bu ağlar, literatürde referans mimariler olarak kabul edildikleri ve göreceli olarak kolay eğitilebilir olmaları sebebiyle seçilmişlerdir. Burada taraması yapılan meta parametreler, giriş aktivasyon haritalarının genişliği ve kullanılan farklı kayıp fonksiyonlarıdır. Yapılan tarama çalışmasına ait performans sonuçları Tablo 6.1’de ayrıntılı olarak sunulmaktadır. Önceki çalışmalar ile bu mimarilere ait seçilmiş performans sonuçları bildirilmiş olsa da burada sunulan sonuçlar *n-n* ağların performansını etkileyen meta parametreleri irdeleyen en kapsamlı karşılaştırmaları içermektedir.

Çizelge 6.1: Referans mimarilerin [4, 5] performans sonuçları.

Ağ	En	Kayıp	oD5	oD10	oD20
PointCN	4	sın	11,77	28,13	48,30
PointCN	4	sın+geo	12,22	29,56	50,22
PointCN	4	sın+temel	11,21	27,89	49,09
PointCN	6	sın	14,48	32,46	53,16
PointCN	6	sın+geo	15,75	34,56	55,41
PointCN	6	sın+temel	13,57	32,38	54,60
OANET	4	sın	14,39	33,02	54,32
OANET	4	sın+geo	14,96	33,51	54,72
OANET	4	sın+temel	14,48	33,17	54,61
OANET	6	sın	17,29	37,70	59,88
OANET	6	sın+geo	17,79	38,58	60,62
OANET	6	sın+temel	16,67	37,05	59,44
OANETiter	4	sın	17,62	37,22	57,44
OANETiter	4	sın+geo	18,28	38,55	59,03
OANETiter	4	sın+temel	17,93	38,35	59,30
OANETiter	6	sın	19,27	40,40	61,43
OANETiter	6	sın+geo	19,87	41,38	62,41
OANETiter	6	sın+temel	18,93	40,40	61,96

Bu karşılaştırmalar, *temel_kayıp*’un oD20 performansını artırdığını ancak bu çalışmaların odak noktası olan oD5 performansını olumlu yönde etkilemediğini göstermektedir. Yalnızca *sın_kayıp* kullanılarak eğitilen ağlar, hem *sın_kayıp* hem de *geo_kayıp* ile eğitilen ağlara kıyasla %2-8 daha az başarılıdır. Buradaki karşılaştırma sonuçlarından

hareketle, $1-1$ ve $n-n$ ağılar ile yapılan tüm sonraki deneylerde, ağıları hem *sin_kayıp* hem de *geo_kayıp* kullanarak eğitime yoluna gidilmiştir. Bununla birlikte, yan bilgilerin kullanılmasının ağ performansını %5-28 oranında artırabildiği gözlemlenmiştir. Bu nedenle, bundan sonraki deneylerin mümkün olan hepsinde yan bilginin kullanılması tercih edilmiştir. Eğitilen bu ağlara ait eğitim ve test grafikleri Şekil 6.1–6.18 de verilmiştir.

6.2 Sıralamadan Bağımsız 1-1 Ağların Başarımları

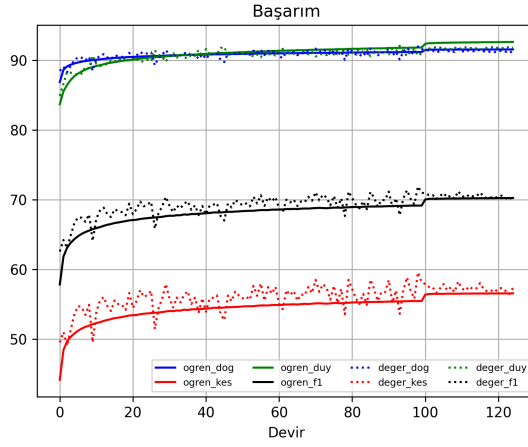
$1-1$ çatısında öncelikli olarak çatının tüm avantajlarını kullanmasa da $1-1$ sıralamadan bağımsız ağılar ile deneyler gerçekleştirilmiştir. Bu ağılar, $1-1$ çatısında bulunan gelişmiş bağlam-eşleşme etkileşiminden faydalanırken, her bir eşleşme için oluşturulabilen özel bağlam sıralamasından faydalanmazlar.

Sıralamadan bağımsız $1-1$ ağılarla deney yapmak için, Şekil 4.1’te gösterilen ağın işleme bloklarına sadece kodlayıcı dönüştürücü (SKD) blokları entegre edilmiştir. Bu bloklardan her biri, aktivasyon haritalarındaki ilgisiz bilgileri filtreleyerek satır sayısını yarıya indirir ve kanal sayısını artırır. SKD-dar ve SKD-geniş olarak adlandırdığımız iki ağ için öğrenme deneyleri gerçekleştirilmiştir. Bu iki ağ sadece kodlayıcı dönüştürücü blokları tarafından üretilen aktivasyon haritalarındaki kanal sayıları açısından farklılık göstermektedir. SKD-dar’da, ilk aktivasyon haritalarındaki kanal sayısı SKD-geniş’e kıyasla daha fazladır. Ancak daha sonraki aşamalarda, Çizelge 6.2 gösterildiği gibi SKD-geniş daha fazla kanal sayısına sahiptir. Kullanılan SKD blokları konumsal yakınlıklardan yararlanma yeteneğine sahip olmadığından giriş aktivasyon haritalarındaki bağlam kanallarının satırlarına herhangi bir sıralama uygulanmamıştır. Giriş aktivasyon haritası bağlam kanallarındaki satırlara ait tekrarlama operasyonları rastgele yapılmamaktadır. Bunun yerine, toplam 2048 satıra ulaşmak için yeterli sayıda satır ilk satırlardan başlanarak tekrar edilmiştir.

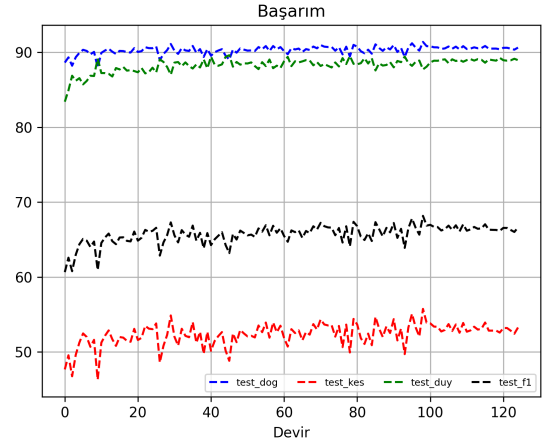
Ağların performans sonuçları Çizelge 6.3’te gösterilmektedir. Sonuçlara göre, dar ağ geniş ağına kıyasla daha üstün performans sonuçlarına ulaşmıştır. Bu durum, ağların erken aşamalarında fazla sayıda kanala sahip olmalarının sonraki aşamalarda fazla sayıda kanala sahip olmalarına göre performansa daha olumlu katkıda bulunduğunu göstermektedir. Sıralı bağlam satırlarından faydalanmayan, $1-1$ SKD-dar ve SKD-geniş ağıları Çizelge 6.1’te gösterilen tüm referans ağlara ait performanslardan daha üstün üstün performanslar elde etmişlerdir. Elde edilen bu ilk sonuçlar dahi eşleşme satırlarının bağlam satırları ile entegrasyonunu en üst seviyeye çıkaran BEB’in küme yapısında girdi içeren problemlerde sağlayabileceği faydayı açıkça göstermektedir.

6.3 Sıralamaya Bağımlı 1-1 Ağların Başarımları

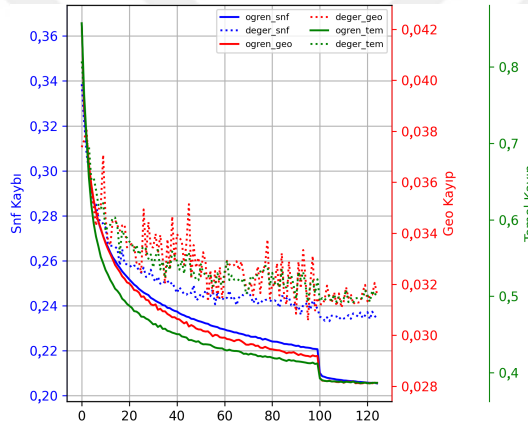
Sıralamaya bağımlı $1-1$ ağılar, performansı artırmak için bağlam kanallarındaki satırların belirli bir strateji dahilinde sıralanmasından faydalanabilir. Bağlam kanallarının satırlarında oluşturulmuş sıralamaları etkili bir şekilde kullanabilme yetenekleri sebebiyle $1-1$ ağılar ile gerçekleştirilen bundan sonraki deneylerde işleme birimi olarak ESA bloklarının kullanılmasını tercih edilmiştir. Sıralamadan bağımsız girdi ile çalı-



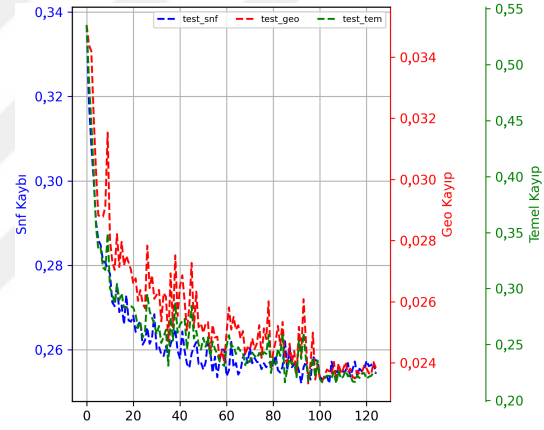
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



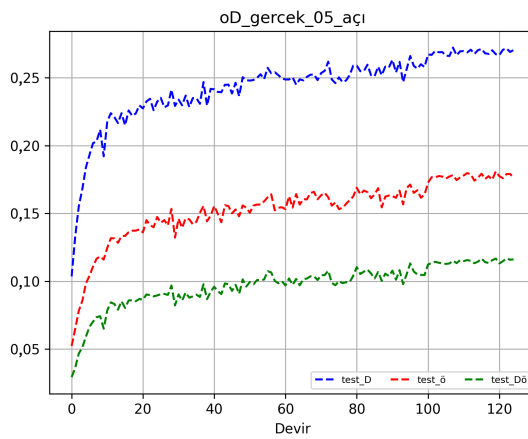
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



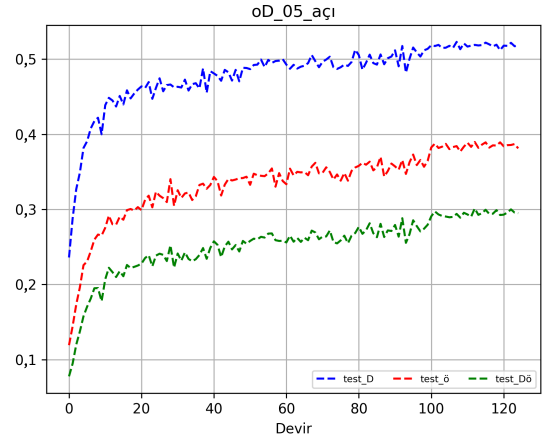
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

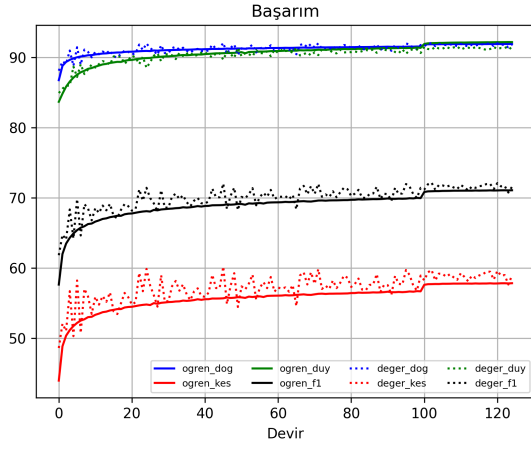


(e) Başarım-yamuk yöntemi (Test)

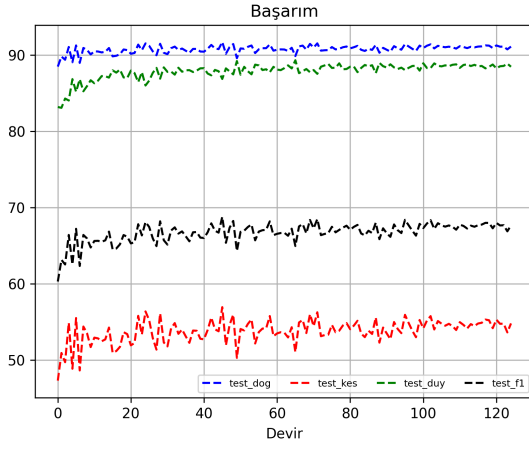


(f) Başarım (Test)

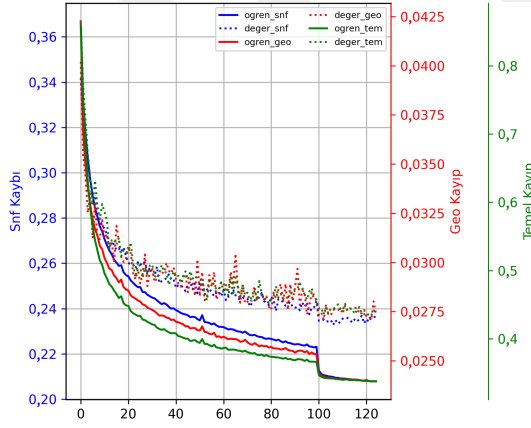
Şekil 6.1: PointCN ağı başarımları, ($E=4$, $sin_kayıp$)



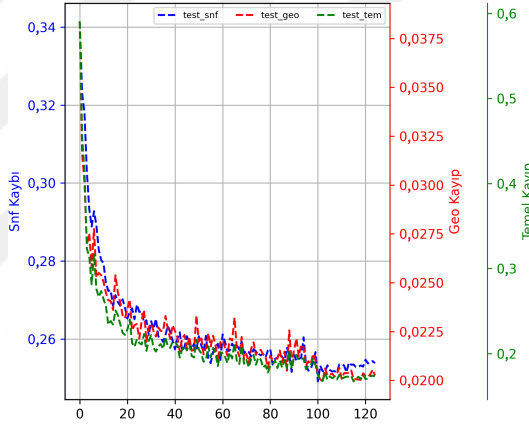
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



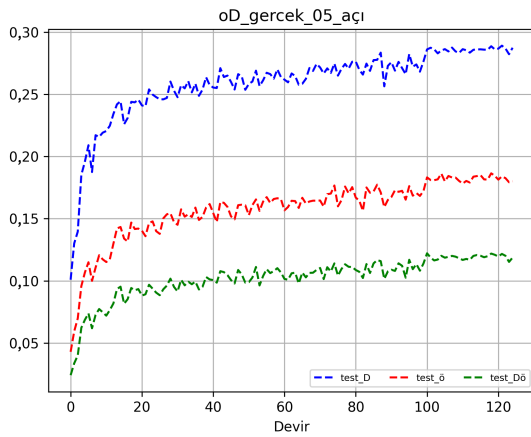
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



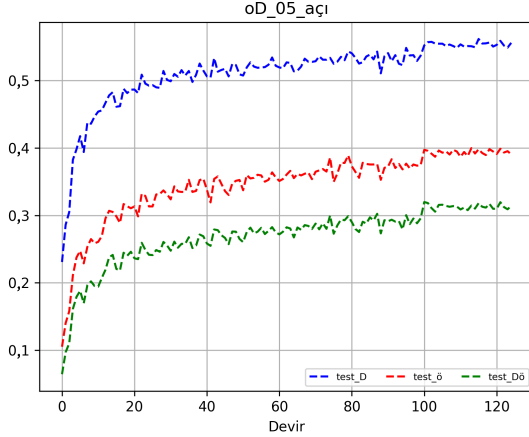
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

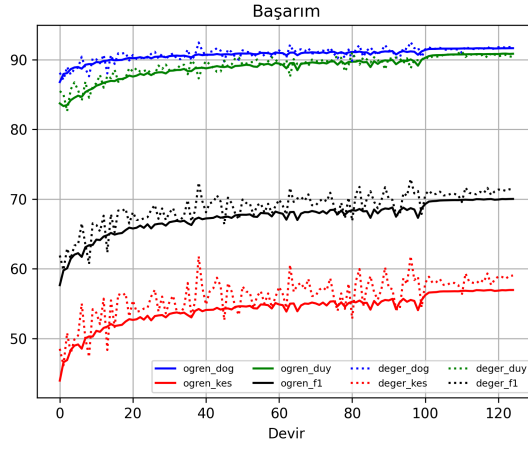


(e) Başarım-yamuk yöntemi (Test)

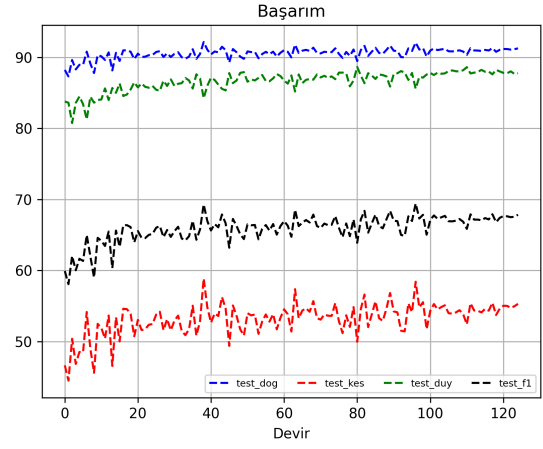


(f) Başarım (Test)

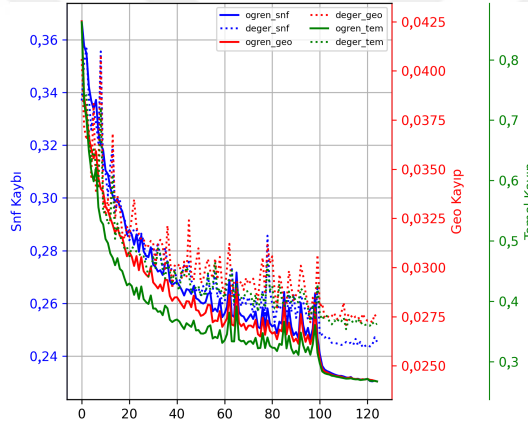
Şekil 6.2: PointCN ağı başarımları, ($E=4$, $sn_kayıp+geo_kayıp$)



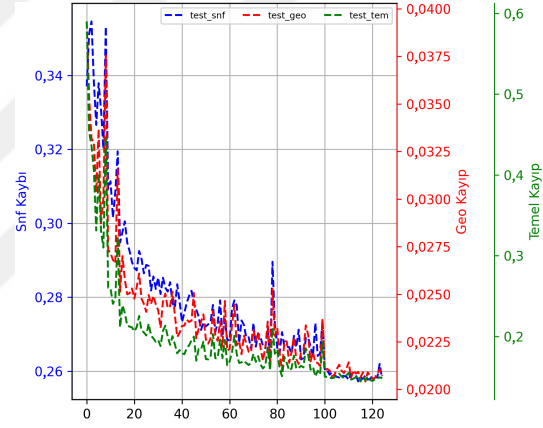
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



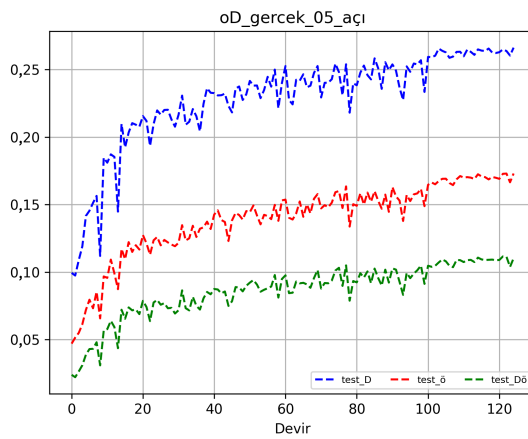
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



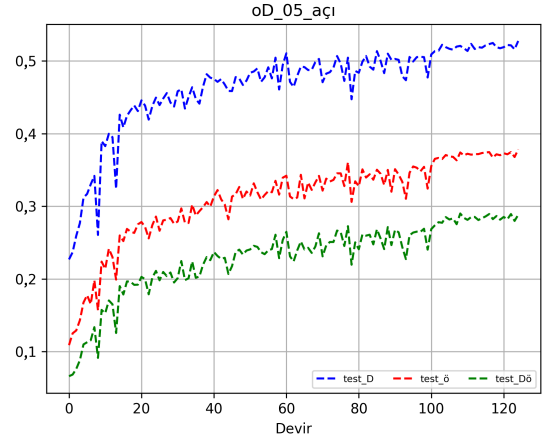
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

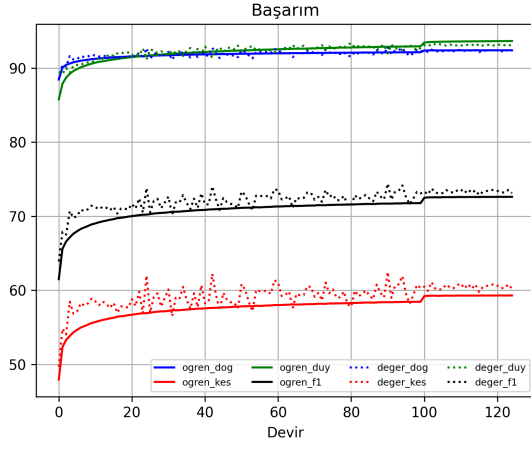


(e) Başarım-yamuk yöntemi (Test)

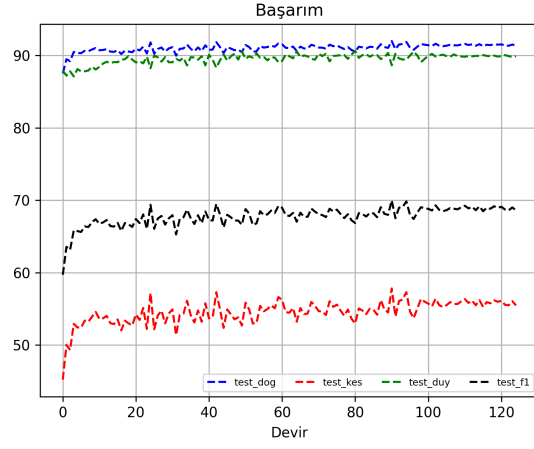


(f) Başarım (Test)

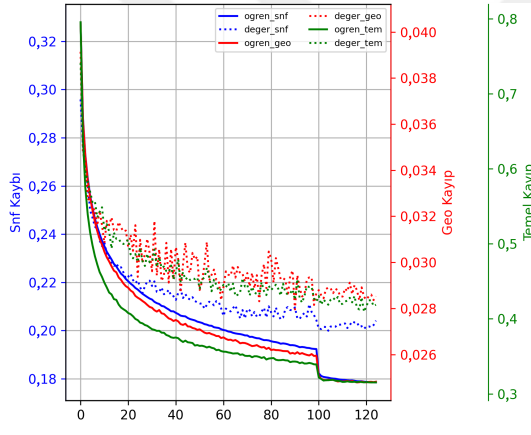
Şekil 6.3: PointCN ağı başarımı, ($E=4$, $sin_kayıp+temel_kayıp$)



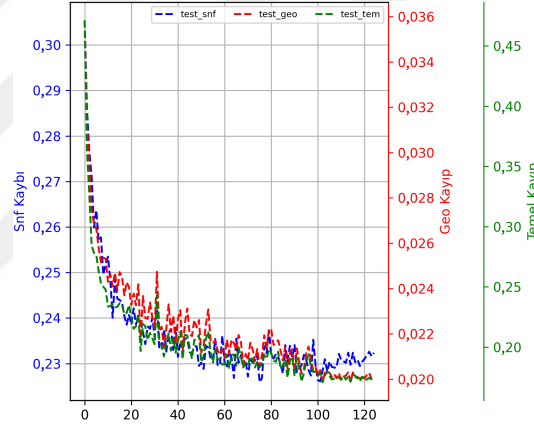
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



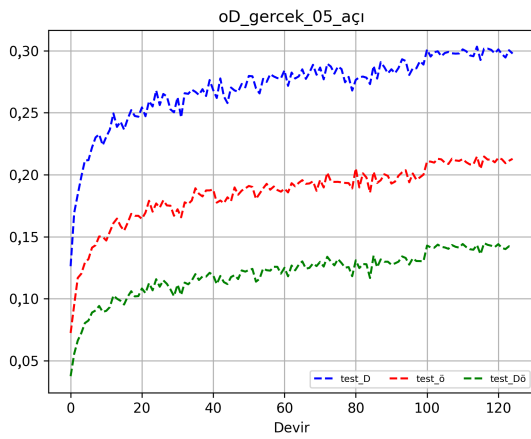
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



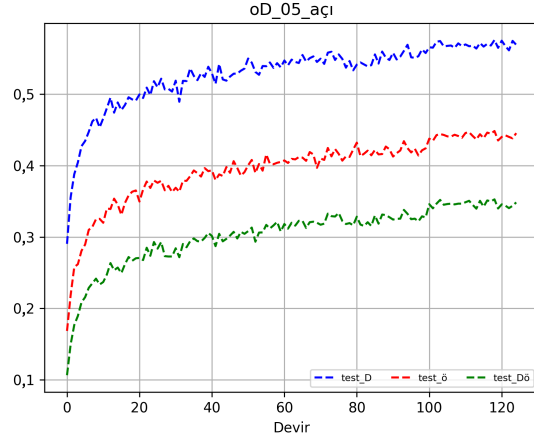
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

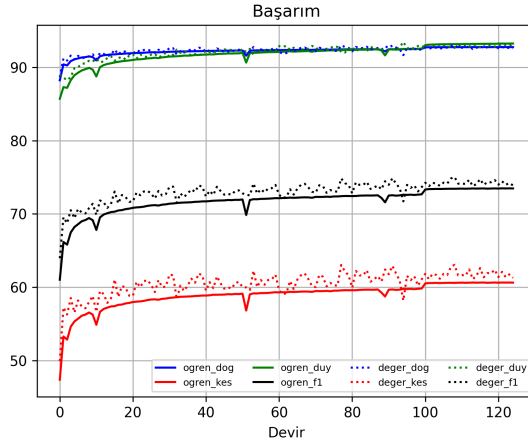


(e) Başarım-yamuk yöntemi (Test)

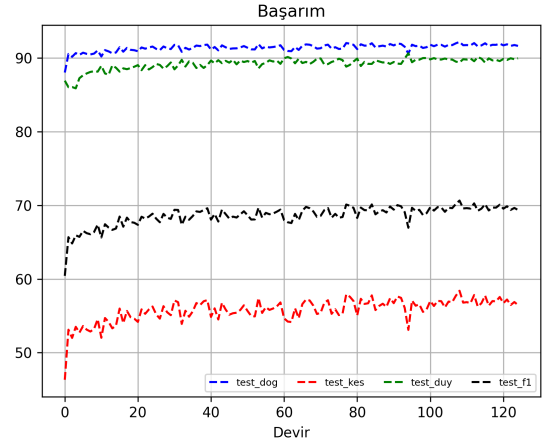


(f) Başarım (Test)

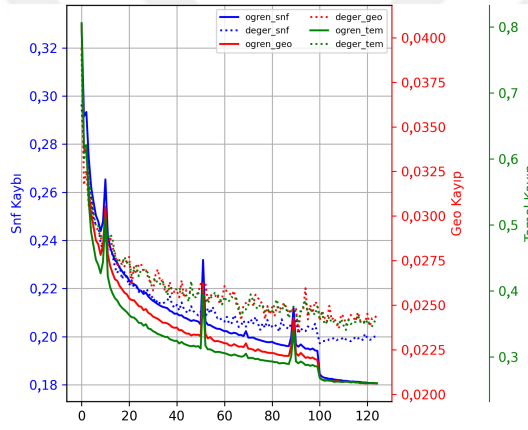
Şekil 6.4: PointCN ağı başarımı, ($E=6$, $sin_kayıp$)



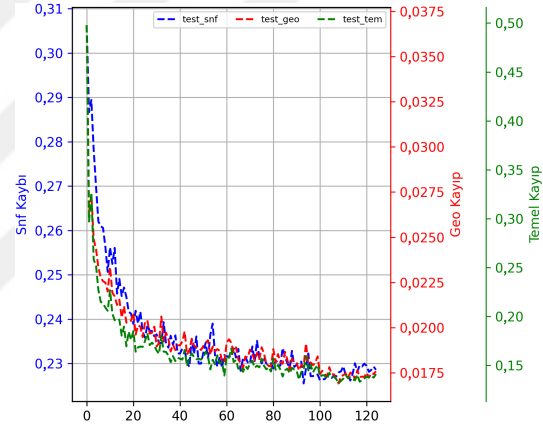
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



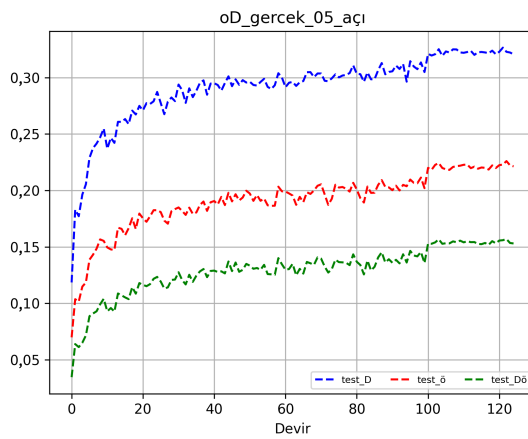
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



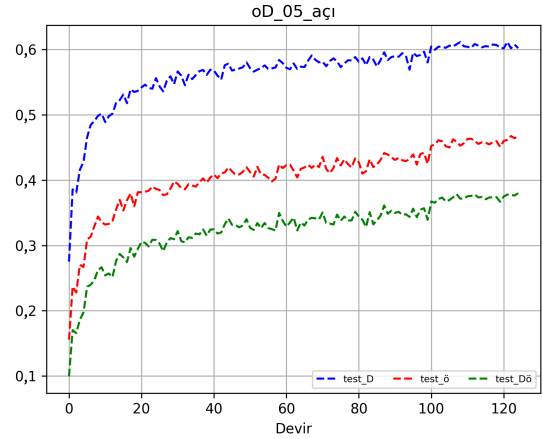
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

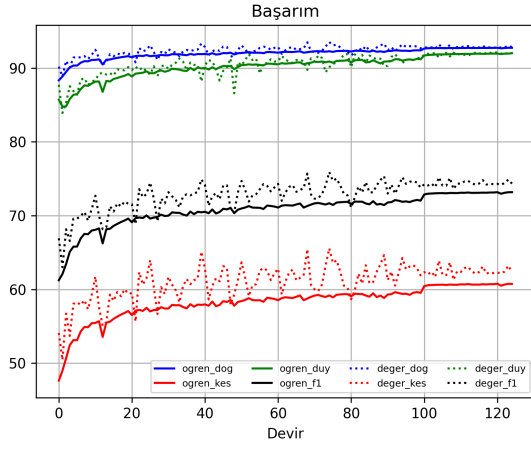


(e) Başarım-yamuk yöntemi (Test)

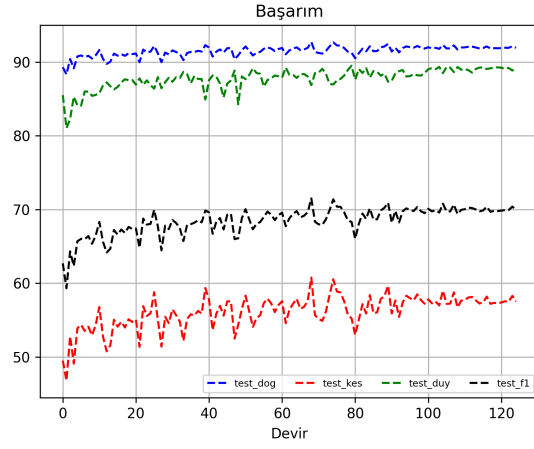


(f) Başarım (Test)

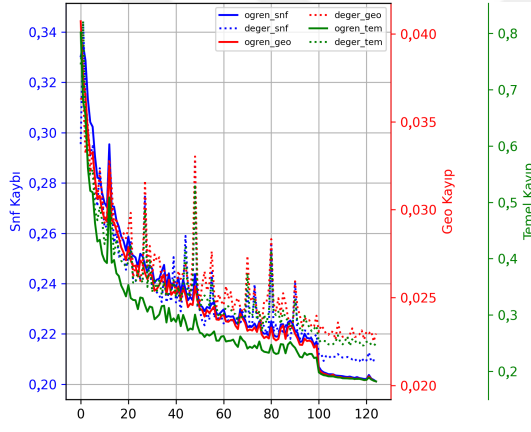
Şekil 6.5: PointCN ağı başarımı, ($E=6$, $sin_kayıp+geo_kayıp$)



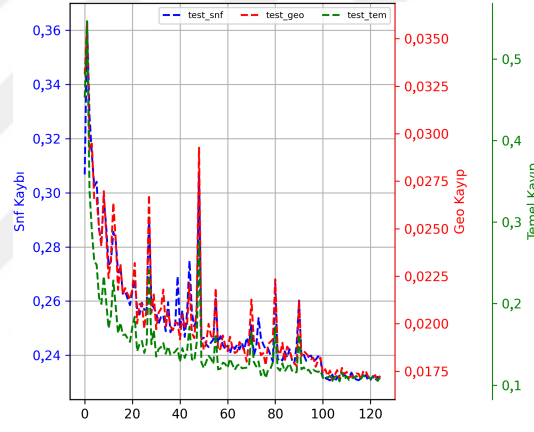
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



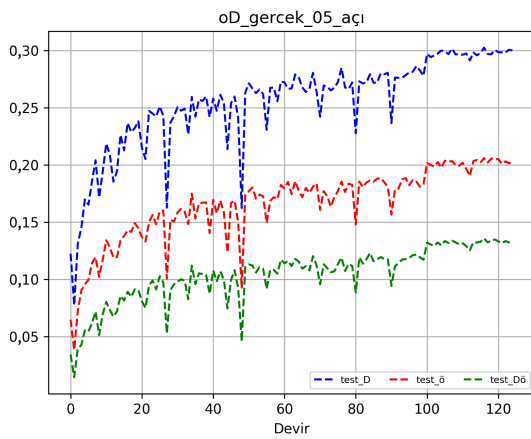
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



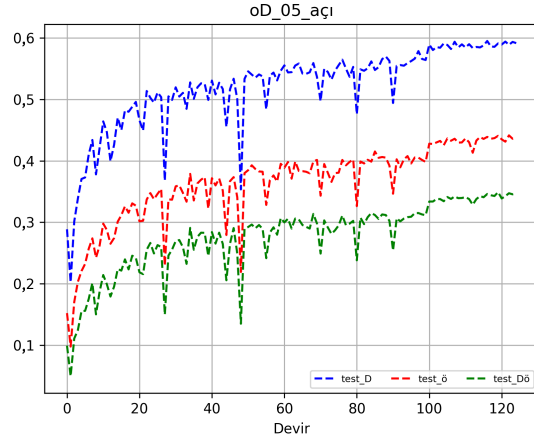
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

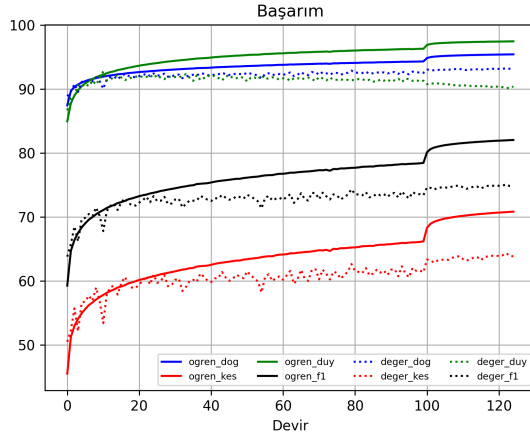


(e) Başarım-yamuk yöntemi (Test)

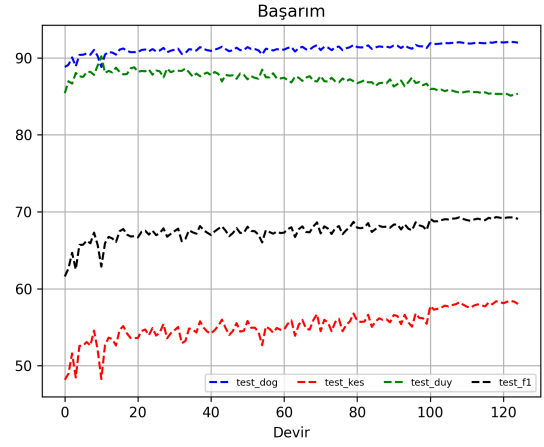


(f) Başarım (Test)

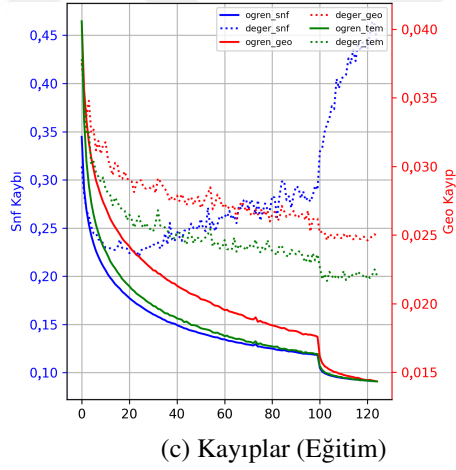
Şekil 6.6: PointCN ağı başarımı, ($E=6$, $sn_kayıp+temel_kayıp$)



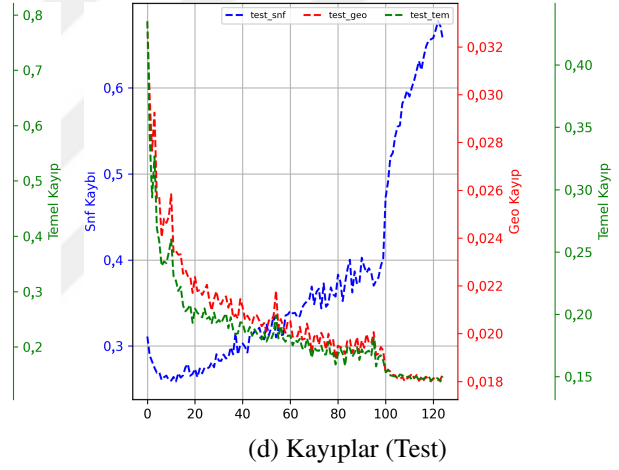
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



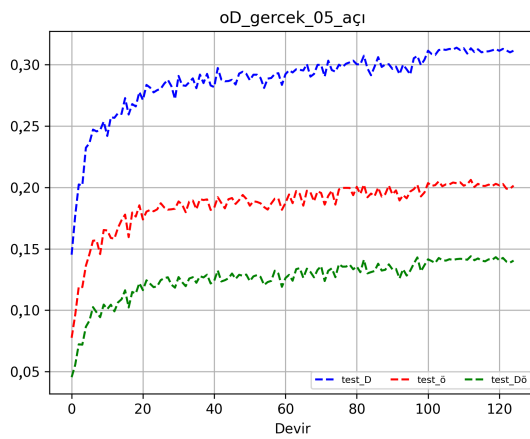
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



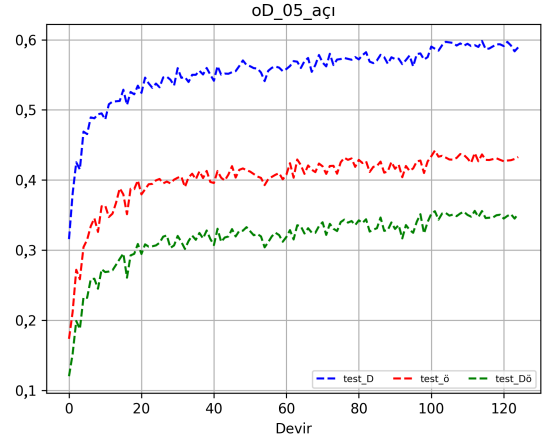
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

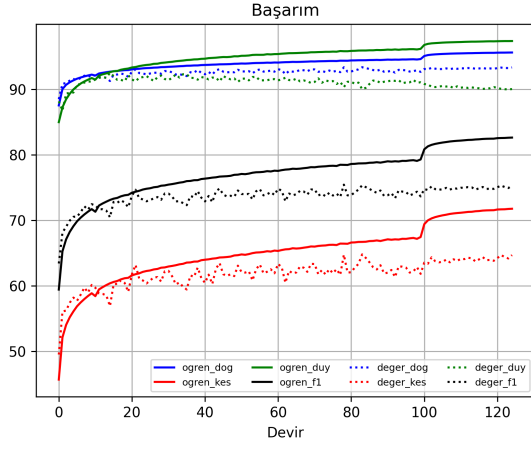


(e) Başarım-yamuk yöntemi (Test)

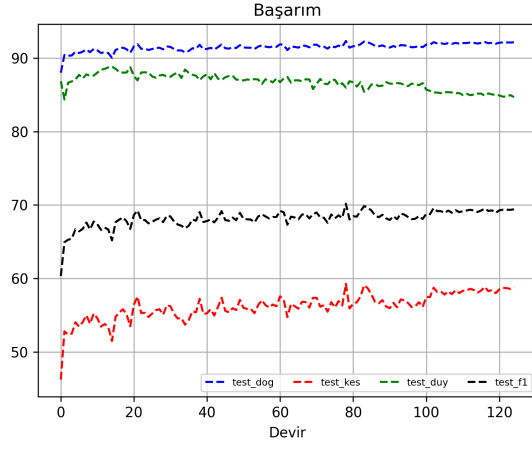


(f) Başarım (Test)

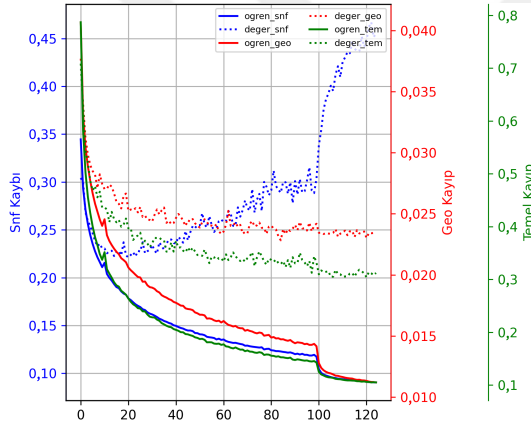
Şekil 6.7: OANET ağı başarımı, ($E=4$, sınıf_kayıp)



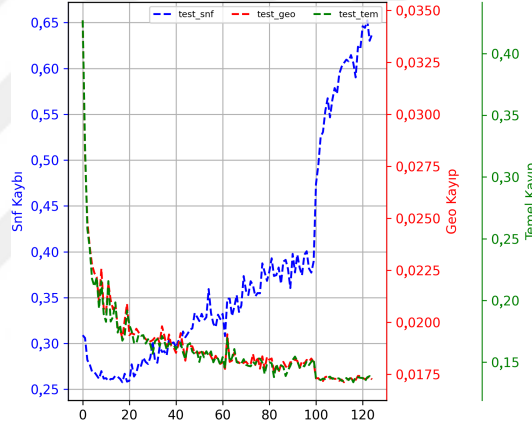
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



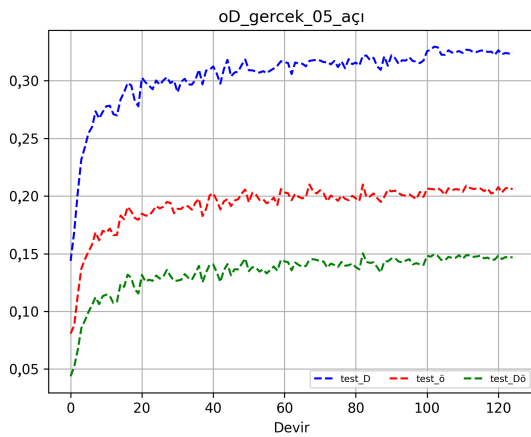
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



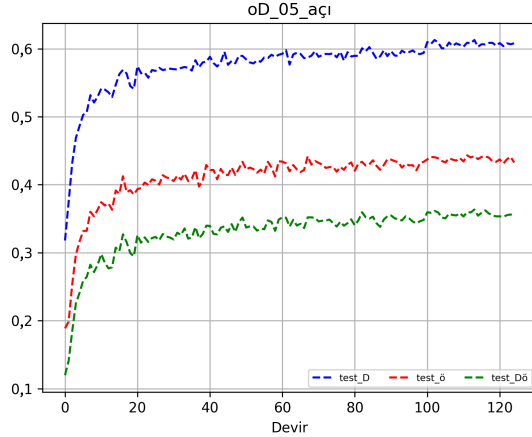
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

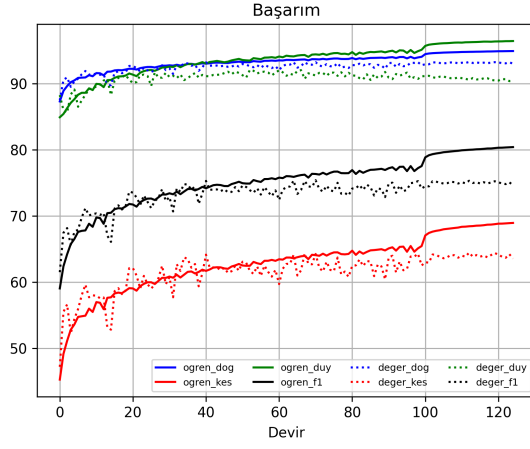


(e) Başarım-yamuk yöntemi (Test)

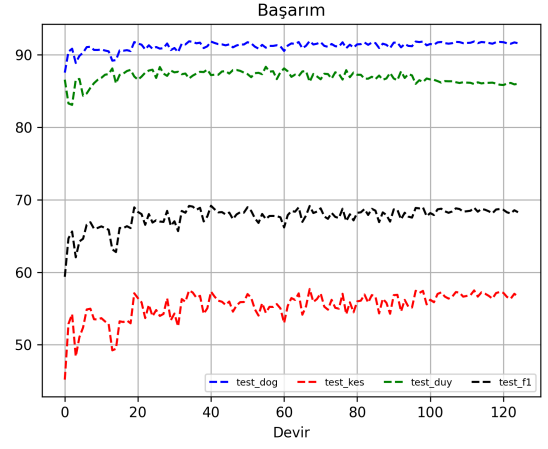


(f) Başarım (Test)

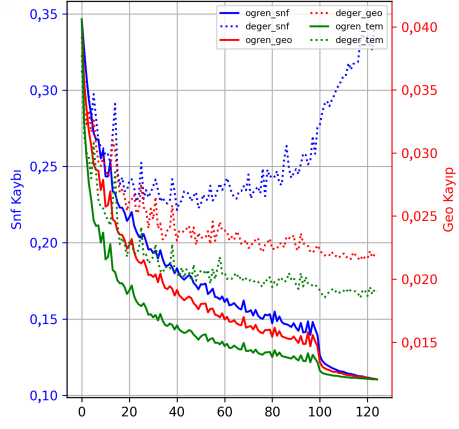
Şekil 6.8: OANET ağı başarımı, ($E=4$, $sin_kayıp+geo_kayıp$)



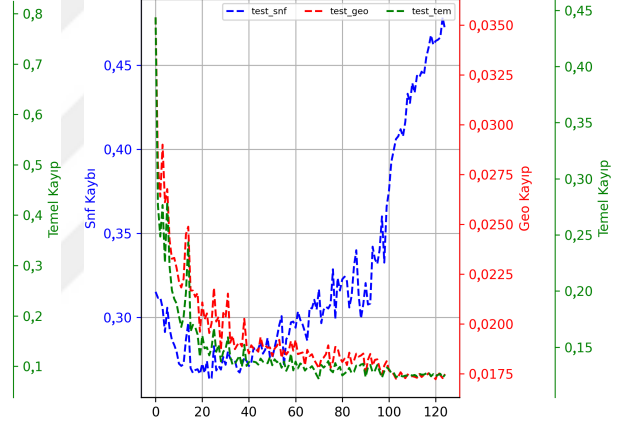
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



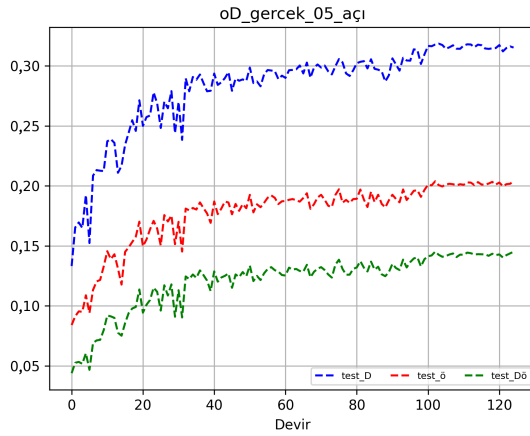
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



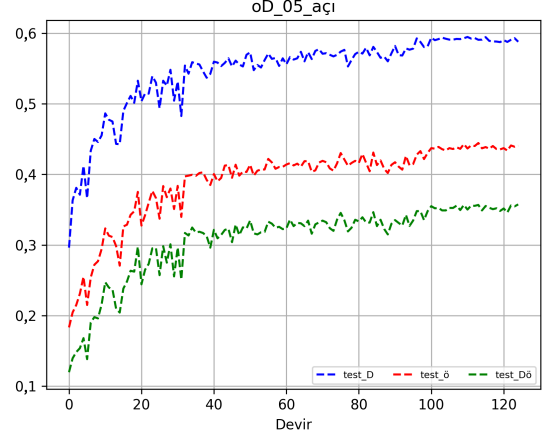
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

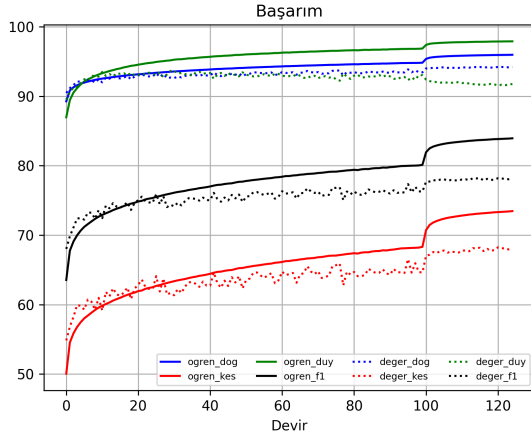


(e) Başarımlar-yamuk yöntemi (Test)

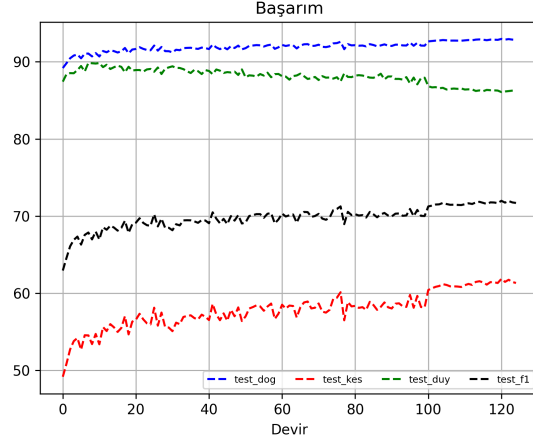


(f) Başarımlar (Test)

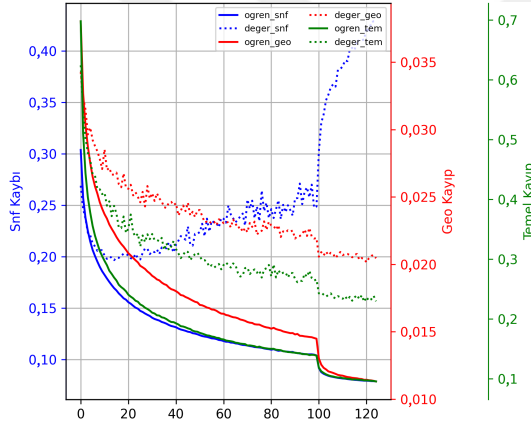
Şekil 6.9: OANET ağı başarımları, ($E=4$, $sn_kayıp+temel_kayıp$)



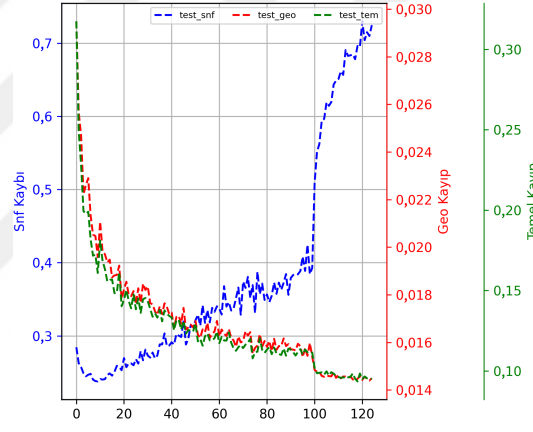
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



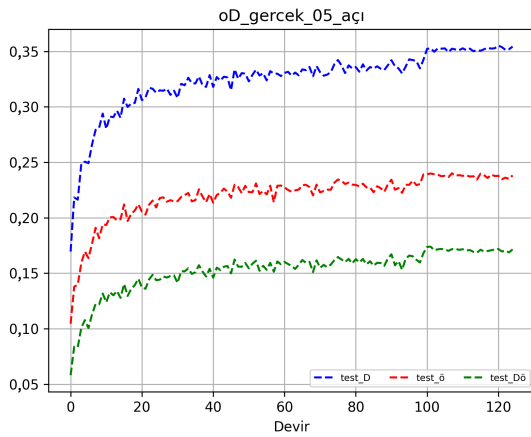
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



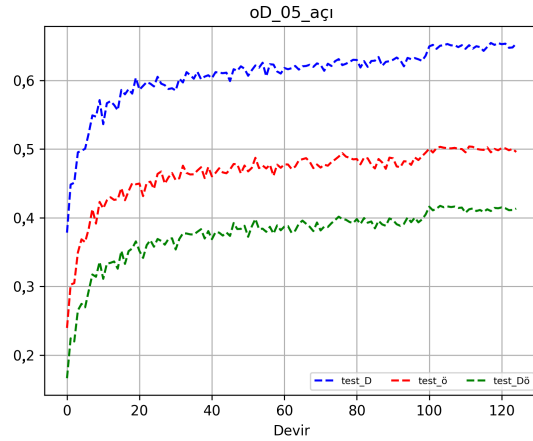
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

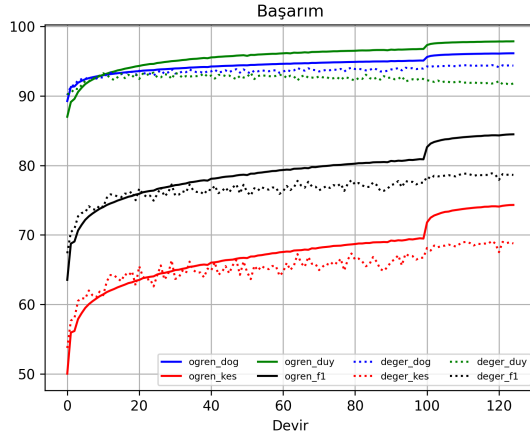


(e) Başarım-yamuk yöntemi (Test)

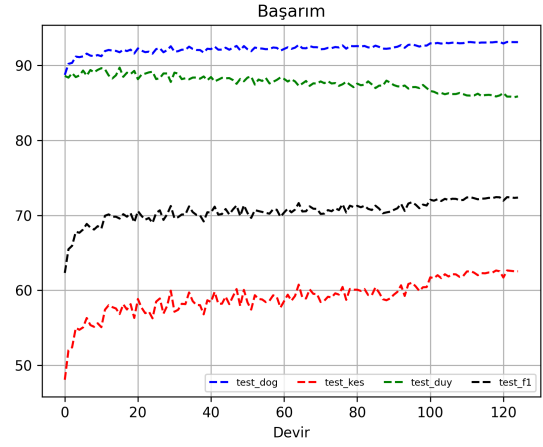


(f) Başarım (Test)

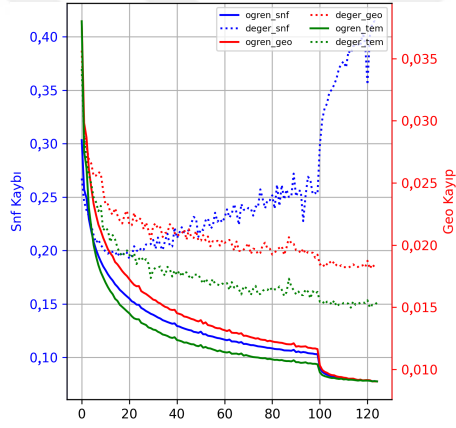
Şekil 6.10: OANET ağı başarımı, ($E=6$, $sin_kayıp$)



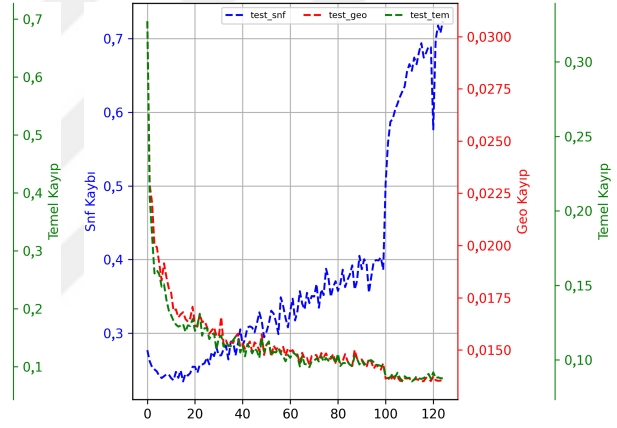
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



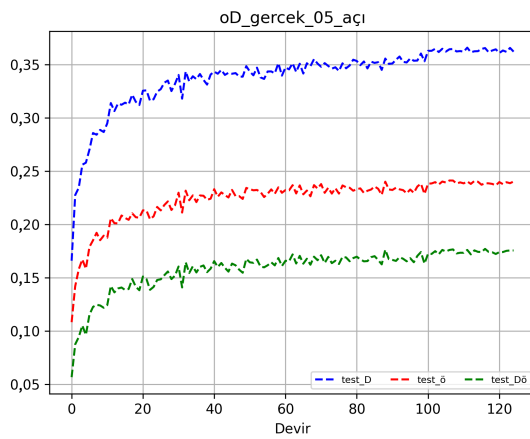
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



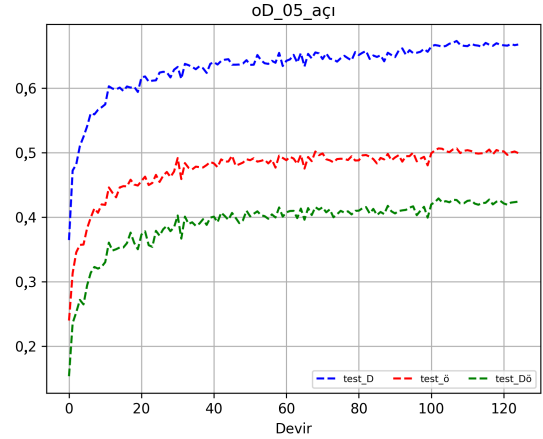
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

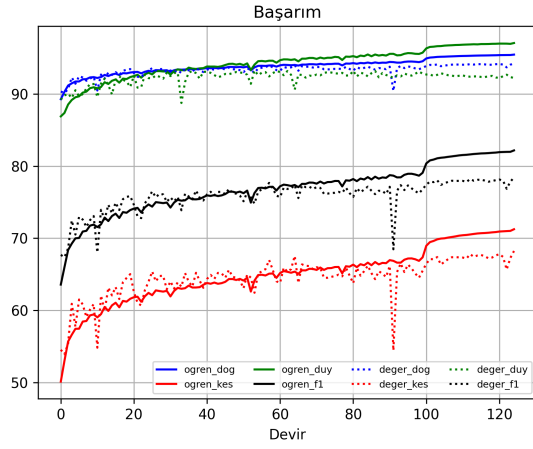


(e) Başarım-yamuk yöntemi (Test)

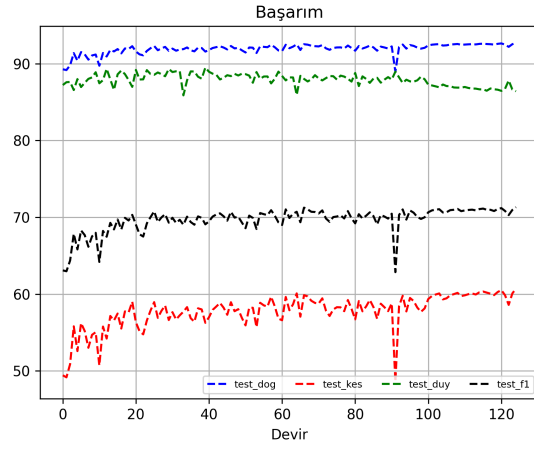


(f) Başarım (Test)

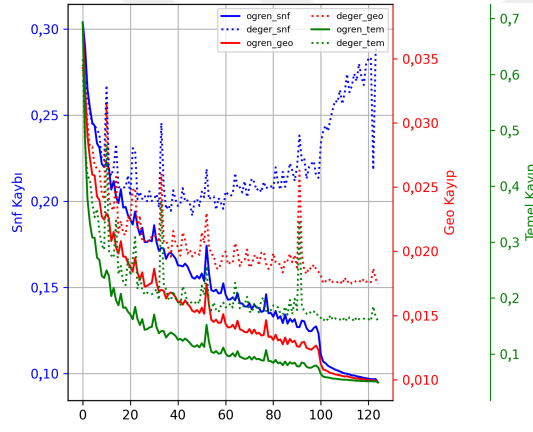
Şekil 6.11: OANET ağı başarımı, ($E=6$, $sin_kayıp+geo_kayıp$)



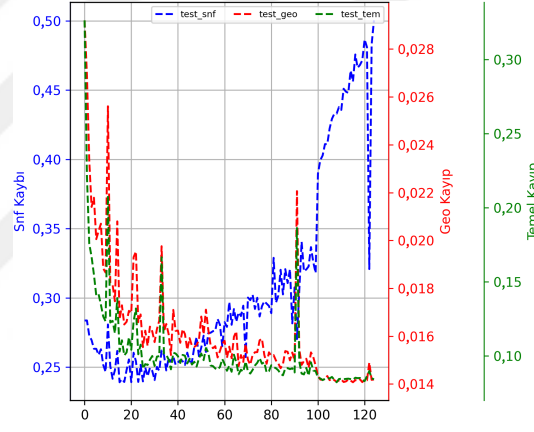
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



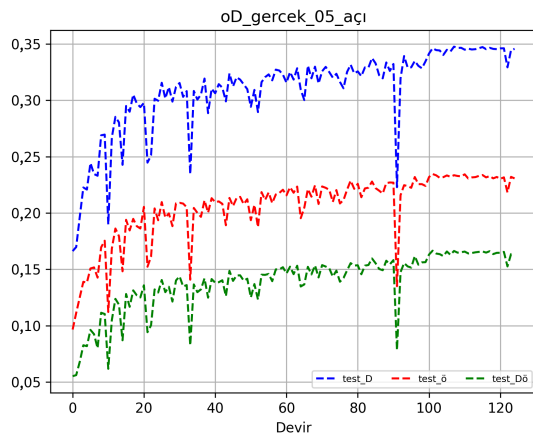
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



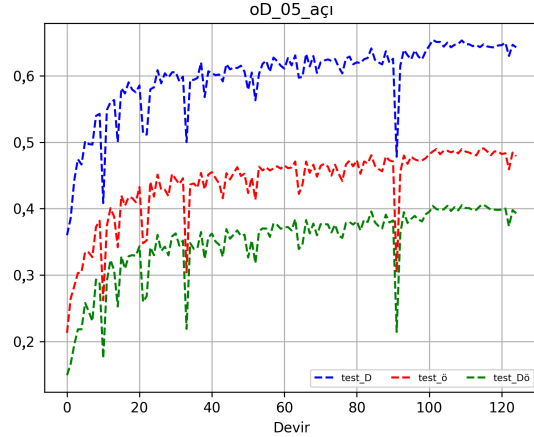
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

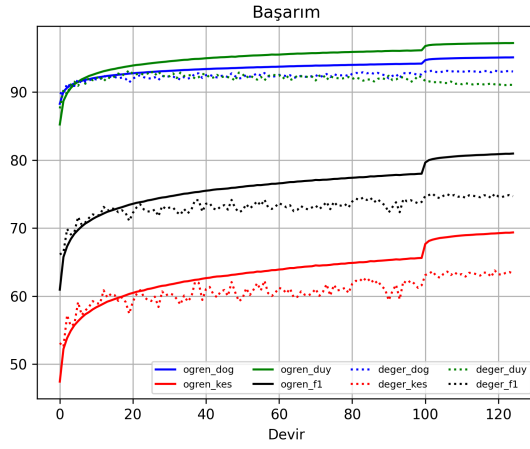


(e) Başarım-yamuk yöntemi (Test)

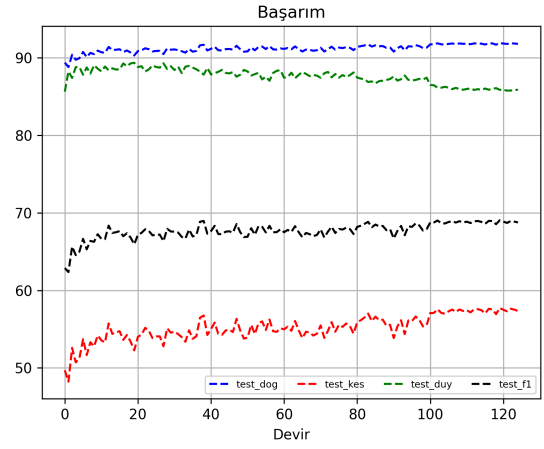


(f) Başarım (Test)

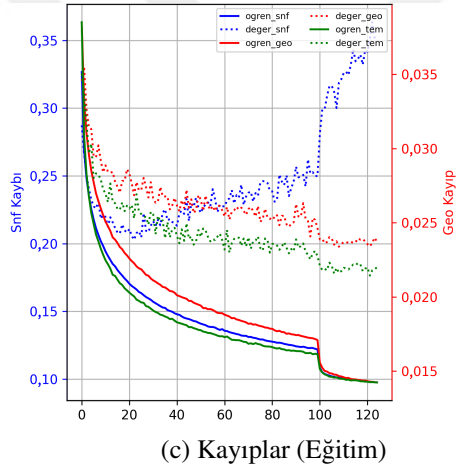
Şekil 6.12: OANET ağı başarımı, ($E=6$, $sın_kayıp+temel_kayıp$)



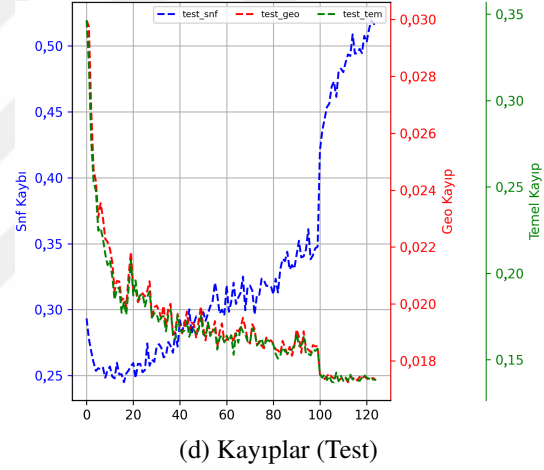
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



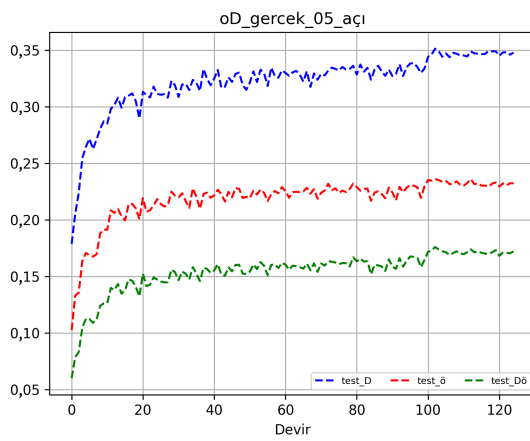
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



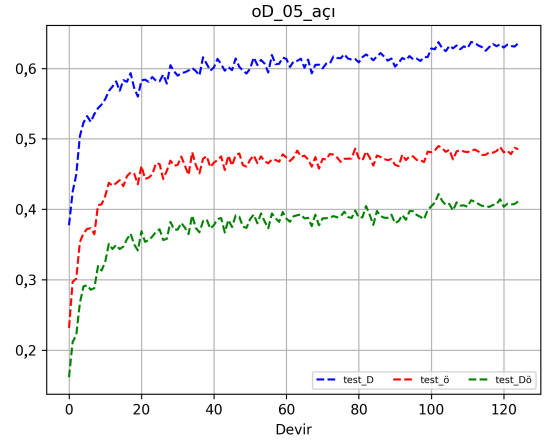
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

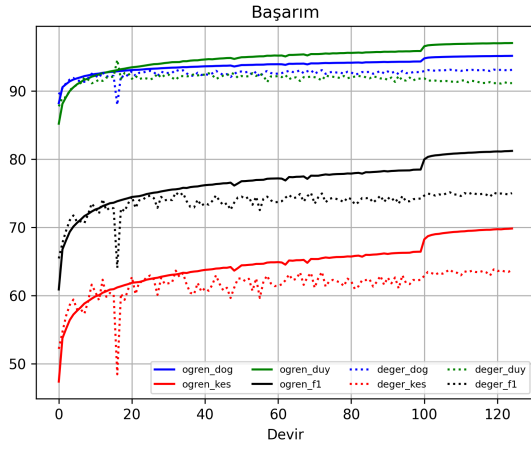


(e) Başarım-yamuk yöntemi (Test)

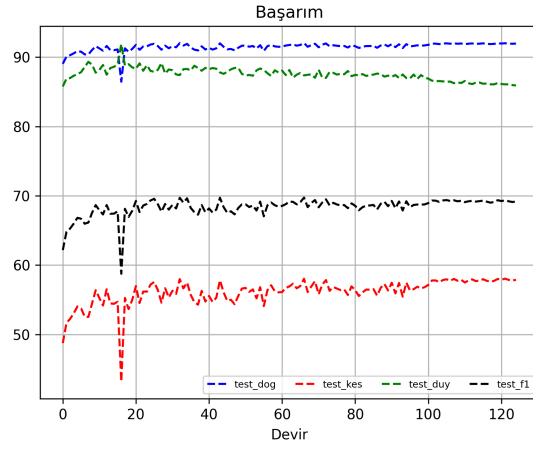


(f) Başarım (Test)

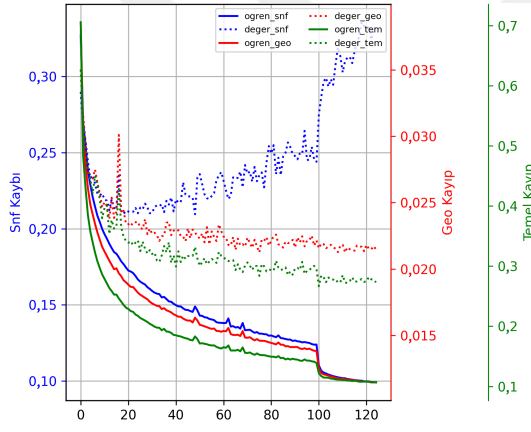
Şekil 6.13: OANETiter ağı başarımı, ($E=4$, $sin_kayıp$)



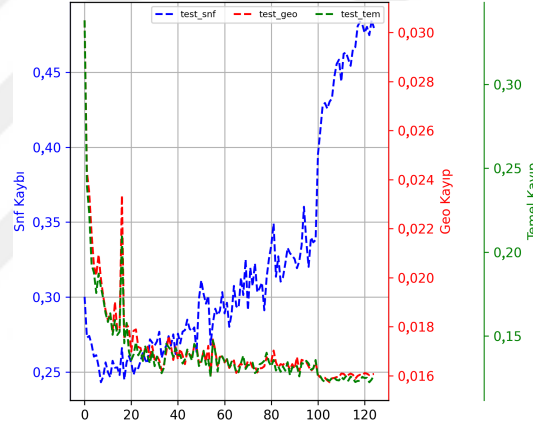
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



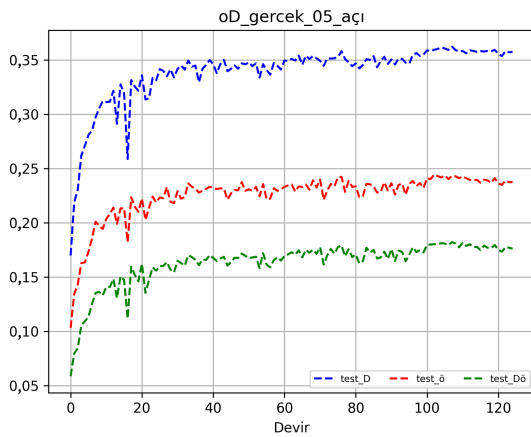
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



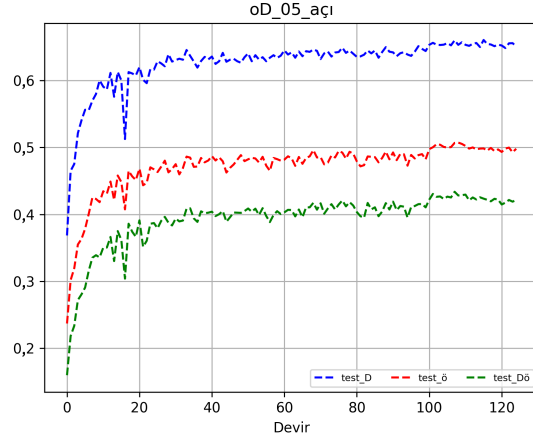
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

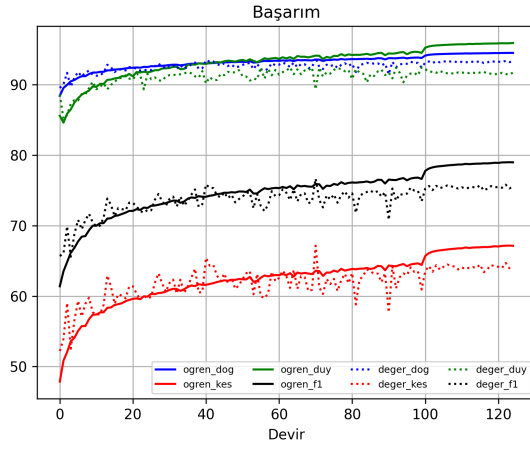


(e) Başarım-yamuk yöntemi (Test)

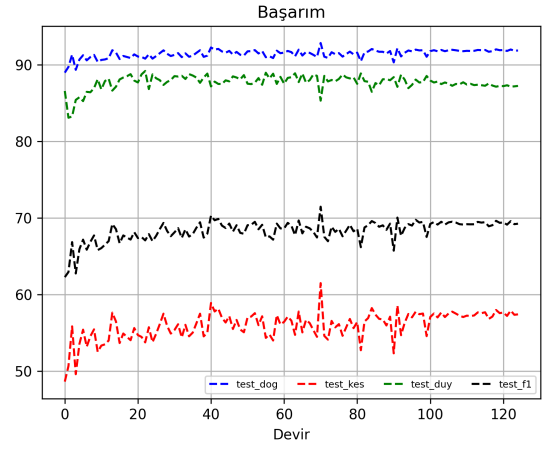


(f) Başarım (Test)

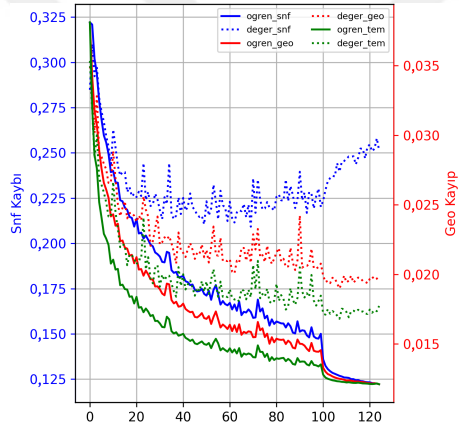
Şekil 6.14: OANETiter ağı başarımı, ($E=4$, $sn_kayıp+geo_kayıp$)



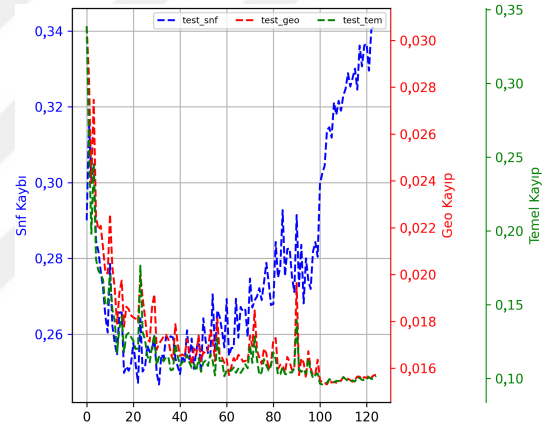
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



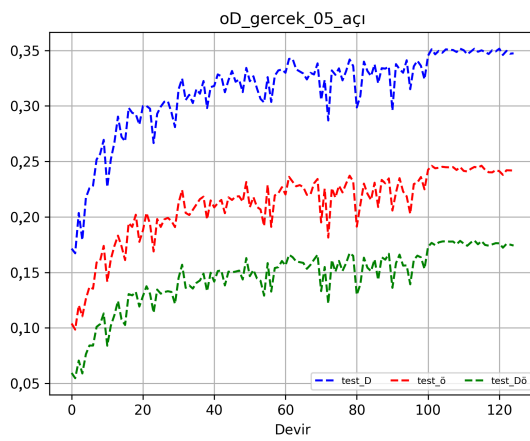
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



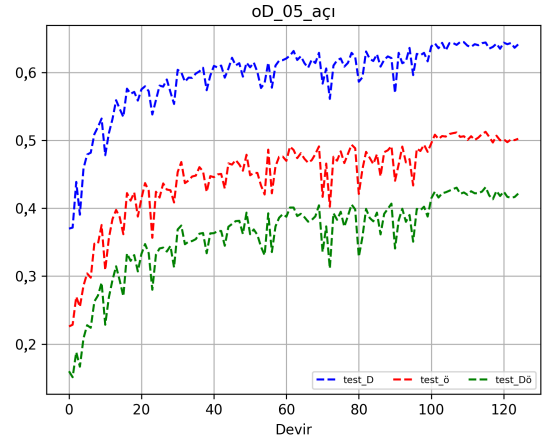
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

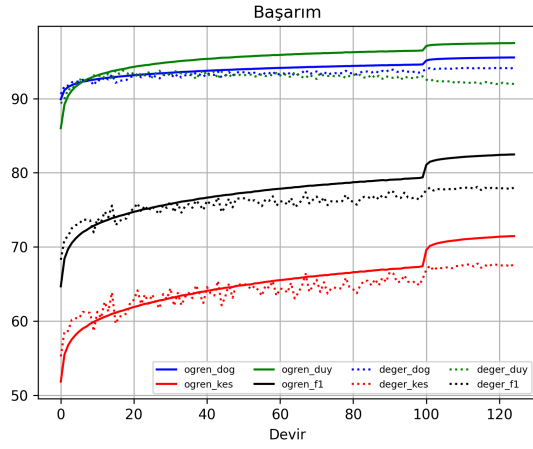


(e) Başarım-yamuk yöntemi (Test)

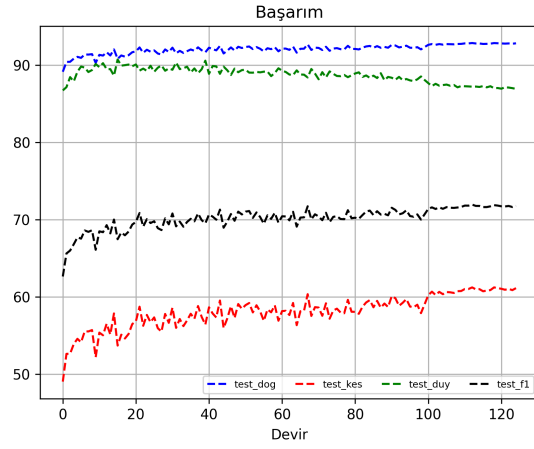


(f) Başarım (Test)

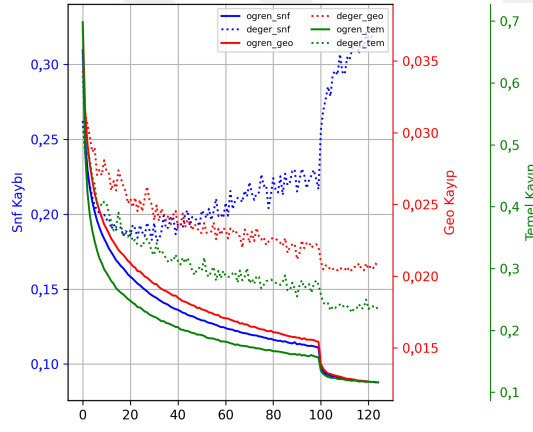
Şekil 6.15: OANETiter ağı başarımı, ($E=4$, $sin_kayıp+temel_kayıp$)



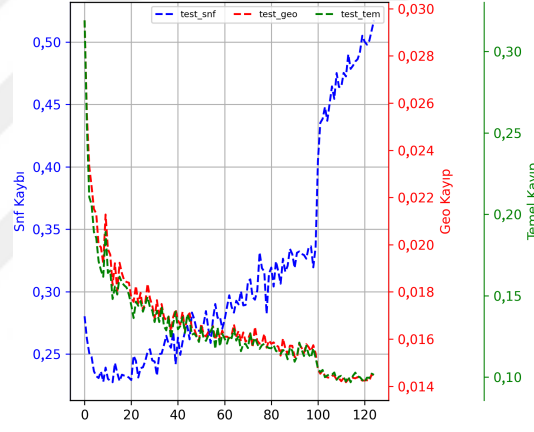
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



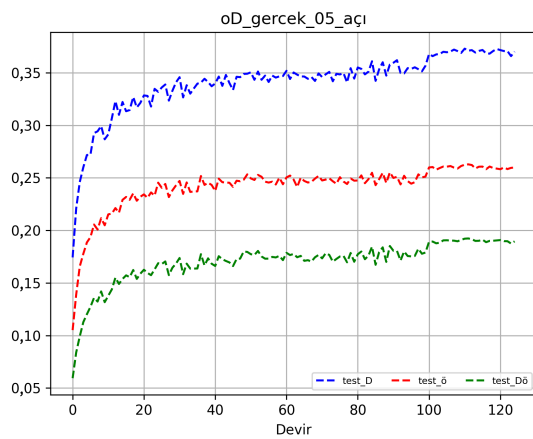
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



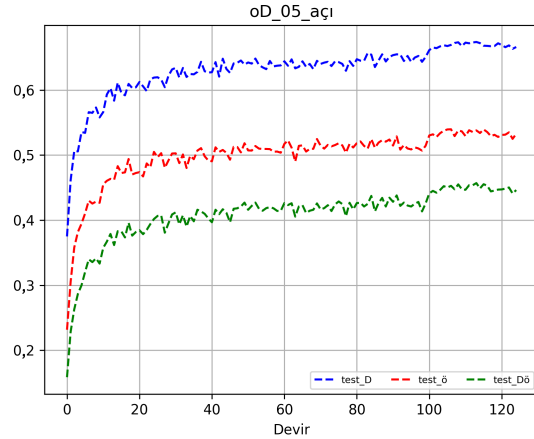
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

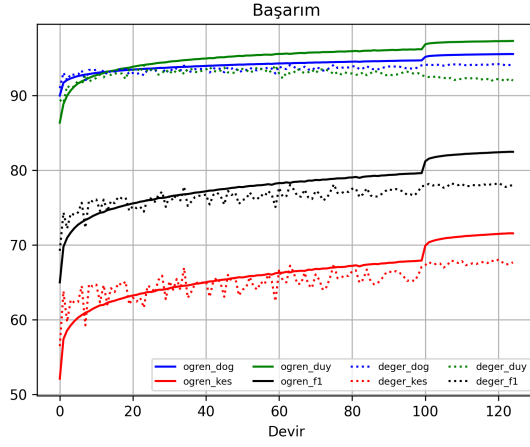


(e) Başarım-yamuk yöntemi (Test)

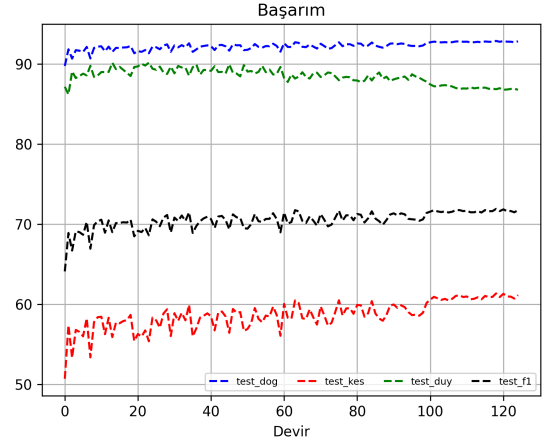


(f) Başarım (Test)

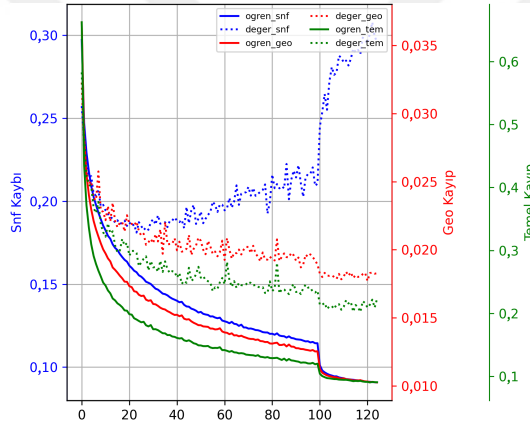
Şekil 6.16: OANETiter ağı başarımı, ($E=6$, $sn_kayıp$)



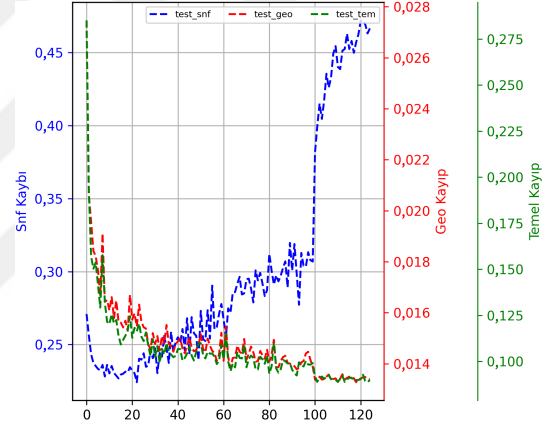
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



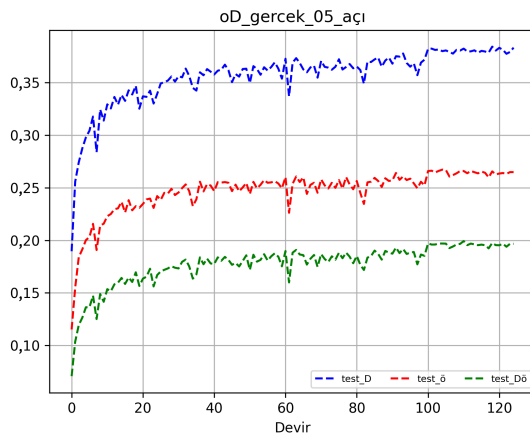
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



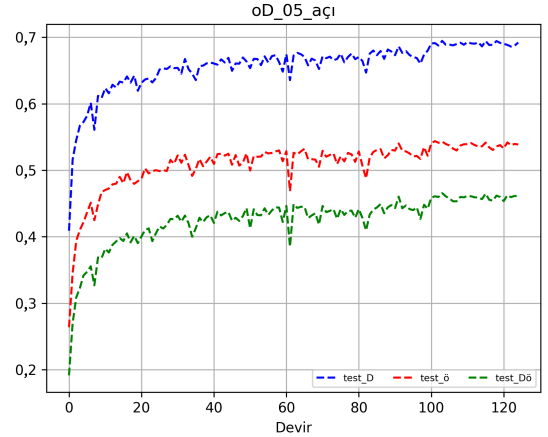
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

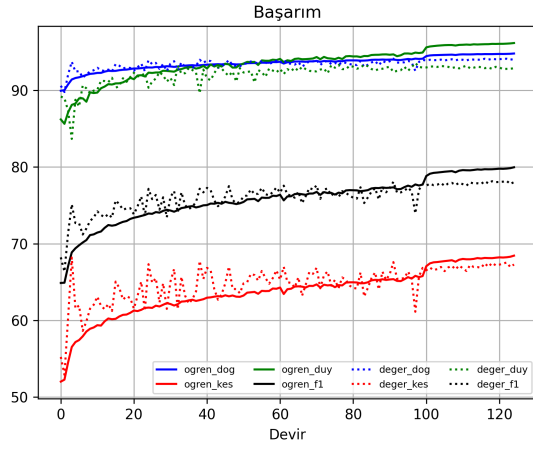


(e) Başarım-yamuk yöntemi (Test)

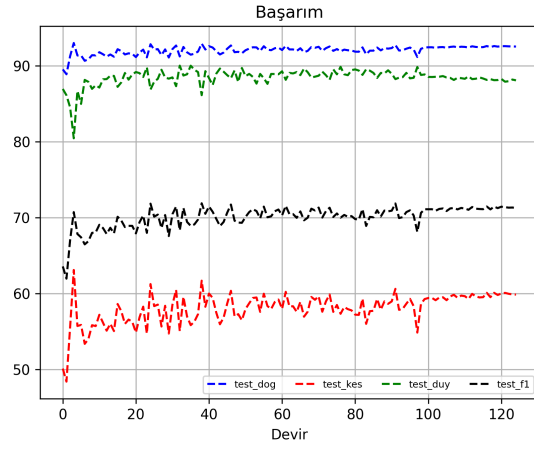


(f) Başarım (Test)

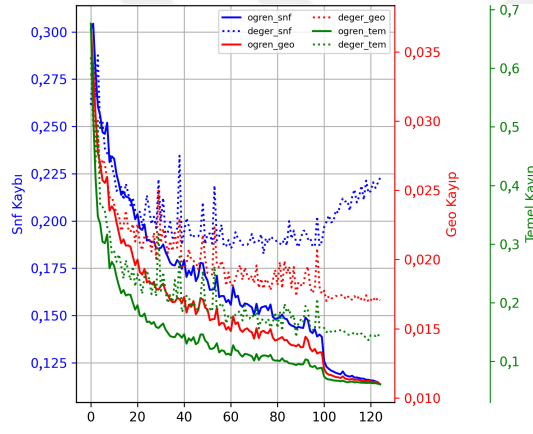
Şekil 6.17: OANETiter ağı başarımı, ($E=6$, $sn_kayıp+geo_kayıp$)



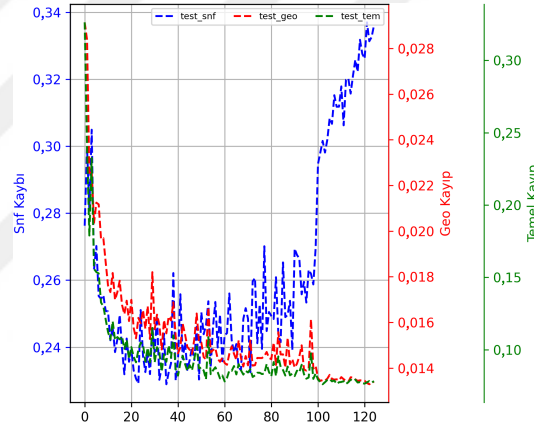
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



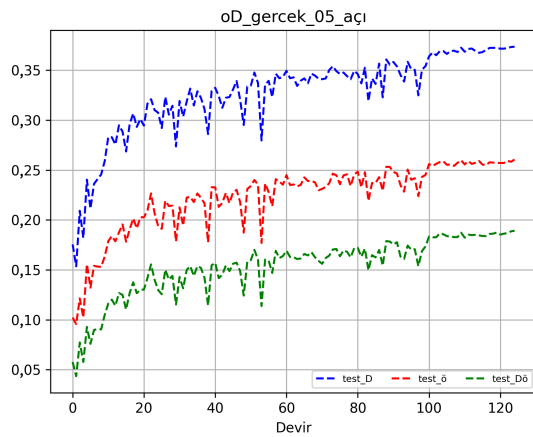
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



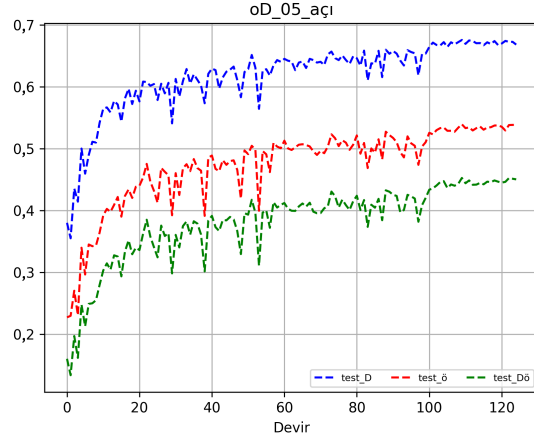
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)



(e) Başarım-yamuk yöntemi (Test)



(f) Başarım (Test)

Şekil 6.18: OANETiter ağı başarımı, ($E=6$, $sn_kayıp+temel_kayıp$)

Çizelge 6.2: SKD-dar ve SKD-geniş mimarileri kanal sayıları.

# Blok	SKD-dar	SKD-geniş
BEB	128	64
1	192	96
2	256	128
3	384	192
4	512	256
5	640	384
6	768	512
7	896	768
8	1024	1024
9	1280	1536
10	1536	2048
11	2048	3072

Çizelge 6.3: SKD-dar ve SKD-geniş ağlarının performans sonuçları.

Ağ	En	oD5	oD10	oD20
SKD-dar	6	21,71	42,63	62,69
SKD-geniş	6	20,07	40,99	61,55

şan SKD ağlarındaki duruma benzer şekilde, oluşturulan ESA'lar da ESA bloklarının her biriyle aktivasyon haritasının satır sayısı yarıya indirilmiştir. Yine benzer şekilde her bir ESA bloğunun çıkışında oluşan aktivasyon haritasındaki kanal sayılarının da ağ içinde arttırarak, ağ içinde elde edilen eşleşme çıkarımlarına aktivasyon haritalarının ev sahipliği edebilmelerini imkân sağlanmıştır. Bundan sonraki deneylerde farklı kanal sayılarına sahip aktivasyon haritaları ile oluşturan iki farklı ağ kullanılmıştır. Kullandığımız ESA-dar ve ESA-geniş ağlarına ait kanal sayıları Çizelge 6.4'te ayrıntılı olarak verilmiştir. Bu ağ mimarilerini kullanarak, farklı yapılandırılmış giriş aktivasyon haritaları ile birçok eğitim çalışması gerçekleştirilmiştir ve ağların farklı koşullar altındaki performansları değerlendirilmiştir.

6.3.1 Bağlam-0 Yapısını Kullanan ESA Mimarilerinin Başarımları

ESA mimarilerinin performansını sadece kodlayıcı dönüştürücü ağları ile karşılaştırmak için, SKD-dar ve SKD-geniş'i eğitmekte kullanılan giriş aktivasyon haritası yapısı, ESA mimarilerini eğitmek için de kullanılmıştır. Bu bağlam yapısında, her bir bağlam düzlemi için satır sayısını 2048'e çıkarmak amacıyla bağlam düzlemlerinin satırları ilk satırlardan başlanarak tekrarlanmıştır. Bağlam düzlemlerinin satırları yeniden sıralanmamış, bunun yerine veri setinin başlangıç sıralaması korunmuştur. Bu deneyin performans sonuçları Çizelge 6.5'da sunulmuştur. Eğitilen bu ağlara ait eğitim ve test grafikleri Şekil 6.19-6.20 de verilmiştir.

Çizelge 6.4: ESA-dar ve ESA-geniş mimarileri kanal sayıları.

# Blok	ESA-dar	ESA-geniş
BEB	128	128
1	384	256
2	640	384
3	896	512
4	1152	768
5	1408	1024
6	1664	1280
7	1920	1536
8	2176	2048
9	2432	2560
10	2688	3072
11	2944	3584

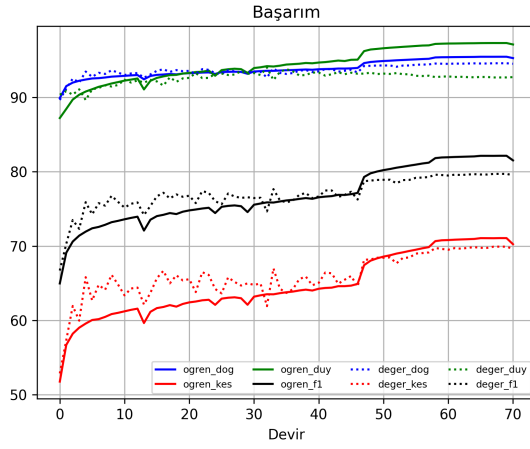
Çizelge 6.5: Bağlam-0 yapısı ESA mimarilerinin performans sonuçları.

Ağ	En	oD5	oD10	oD20
ESA-dar-0	6	24,32	45,75	65,41
ESA-geniş-0	6	23,46	45,11	65,24

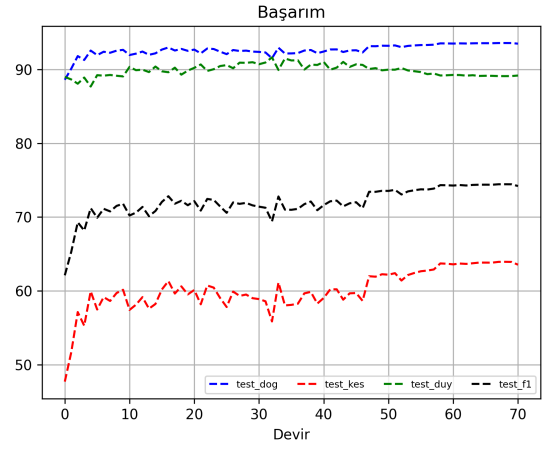
Bu sonuçlara göre, bağlam düzlemlerinin satırları sıralanmamış ve mekânsal olarak ilişkili bağlam düzlemleri oluşturulmamış olsa bile, hem ESA-dar-0 hem de ESA-geniş-0, SKD-dar ve SKD-geniş'den %15'ten fazla daha iyi performans göstermektedir. Bu, bağlam yapıları evrişim işlemleri için uygun girdi sıralamaları içermese bile, *I-I* çatısında ESA bloklarının SKD bloklarından daha iyi performans gösterdikleri kanıtlanmaktadır. Bağlam yapılarındaki ilişkili satırlar, Bağlam-0 yapısında olduğu gibi, birbirine yakın olmasa bile, sonraki ESA bloklarının bağlam satırları arasındaki ilişkilerden bir kısmını çıkarabildiği gözlemlenmiştir. SKD deneylerine benzer şekilde, ESA-dar mimarisi ESA-geniş mimarisine göre daha başarılı performans göstermiştir. Bu da ağların erken aşamalarında daha fazla sayıda kanalın kullanılmasının önemini bir kez daha vurgulamaktadır.

6.3.2 Bağlam-1 Yapısını Kullanan ESA Mimarilerinin Başarımları

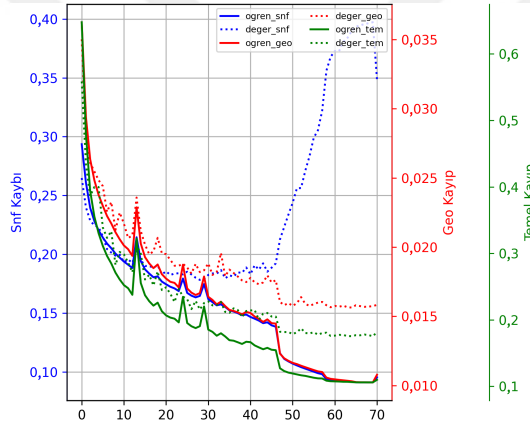
ESA mimarilerinin *I-I* çatısında SKD ağlarından daha iyi performans gösterdiği göz önüne alındığında, giriş aktivasyon haritalarının bağlam yapılarını değiştirerek daha yüksek bir performansa ulaşması amaçlanmıştır. Deneylerin bu aşamasında, bağlam düzlemlerinin satır sayısını 2048'e çıkarmak için başlangıç satırlarını tekrar etmek yerine, eklenecek satırları rastgele seçilmiştir. Ayrıca, bağlam düzlemlerinin satırlarını sıralamasına rastgele karıştırma işlemi uygulanmıştır. Bu iki işlem, *I-I* çatısında veri artırma sağlamaktadır. Bu işlemlerle oluşturulan bağlam düzlemlerine sahip ağların performansı Tablo 6.6'de gösterilmiştir. Eğitilen bu ağlara ait eğitim ve test grafikleri Şekil 6.21-6.22 de verilmiştir.



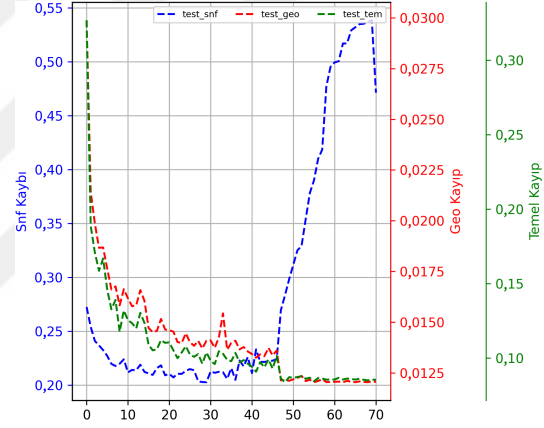
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



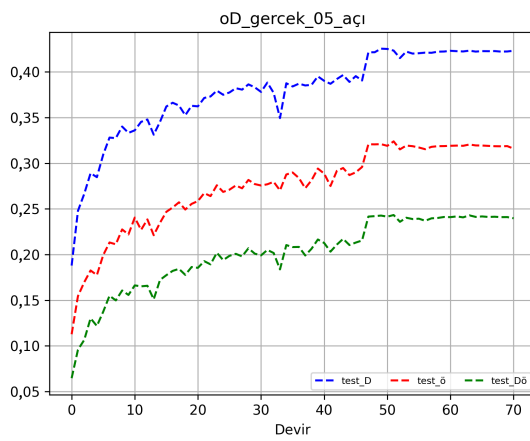
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



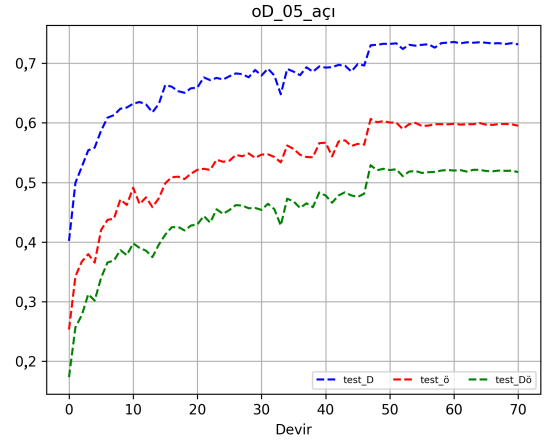
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

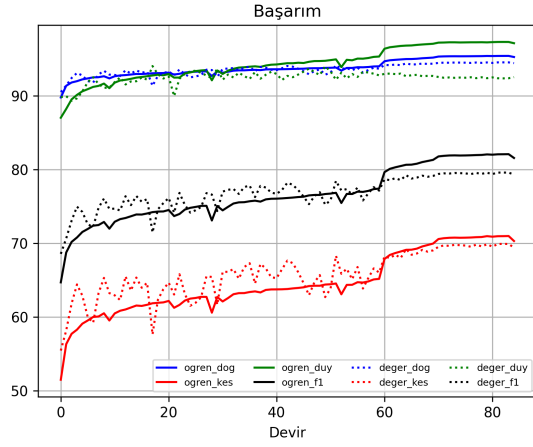


(e) Başarım-yamuk yöntemi (Test)

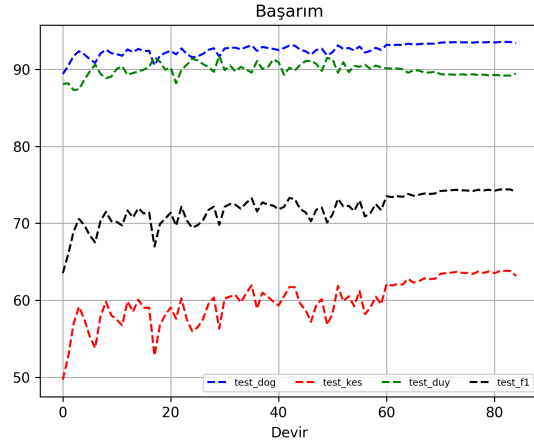


(f) Başarım (Test)

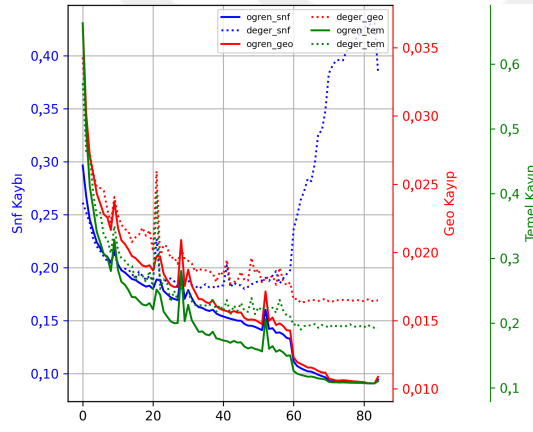
Şekil 6.19: ESA-dar-0 ağı başarımı.



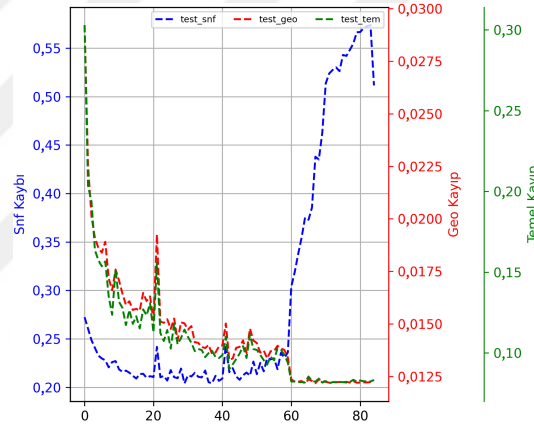
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



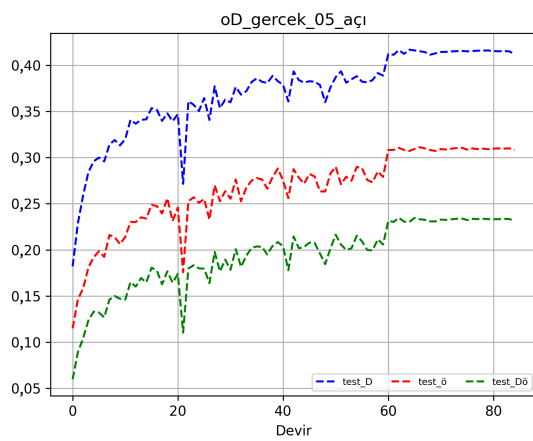
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



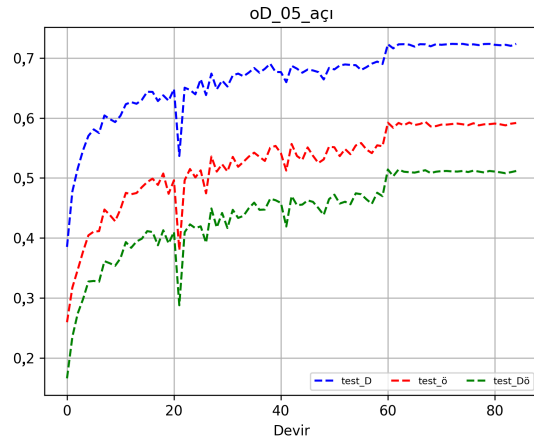
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)



(e) Başarım-yamuk yöntemi (Test)



(f) Başarım (Test)

Şekil 6.20: ESA-geniş-0 ağı başarımı.

Çizelge 6.6: Bağlam-1 yapısı ESA mimarilerinin performans sonuçları.

Ağ	En	oD5	oD10	oD20
ESA-dar-1	6	25,41	46,99	66,34
ESA-geniş-1	6	24,92	46,40	66,10

Sonuçlar, eklenen veri artırma işlemlerinin ağların genelleme yeteneğine önemli ölçüde katkı sağladığını göstermektedir. Bağlam-0 yapısında olduğu gibi, Bağlam-1 yapısı da uygun evrişim işlemleri için ilişkili satırlar sağlamaz, bu da sonraki ESA bloklarının yerel bağlam çıkarılmasını kolaylaştırmaz. Bununla birlikte hem geniş hem de dar versiyonlardaki ESA mimarilerinin performansı Bağlam-0 yapısında eğitilen muadillerine göre %4'ten fazla artmıştır. Bu bağlam yapısında da dar ağ geniş ağa göre daha üstün sonuç vermiştir.

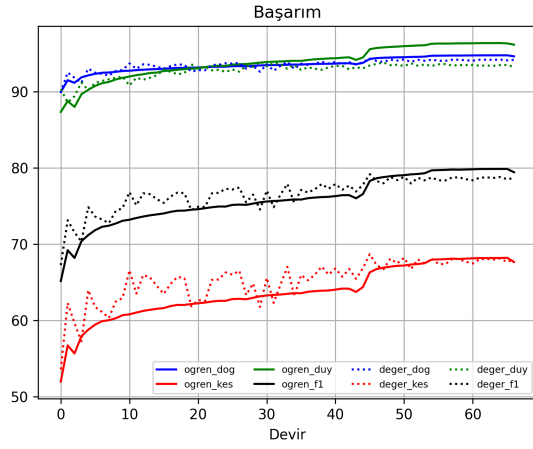
6.3.3 Bağlam-2 Yapısını Kullanan ESA Mimarilerinin Başarımları

1-1 çatısında eğitilen ESA mimarilerinin performansını daha da artırmak için, her yığın örneğinin bağlam kanallarının satırlarını o yığın örneğine ait eşleşme için özel olarak sıralamayı tercih ettik. Bu özel olarak yapılandırılmış bağlam kanalları ile, evrişim katmanlarının eşleşmeler arasındaki daha karmaşık ilişkileri çıkarmasına yardımcı olmak amaçlandı. Bu sıralama yapılırken her yığın örneği için, o yığın örneğindeki eşleşme ile bağlam düzlemindeki diğer eşleşmeler arasındaki Öklid mesafeleri hesaplanır. Her bir varsayılan eşleşmeye ait bağlam kanalındaki satırlar hesaplanan Öklid mesafelerinin küçükten büyüğe doğru sıralanmasıyla konumlandırılır. Hesaplamalarda, her görüntü çiftindeki ilk görüntü referans görüntü olarak seçilir ve referans görüntüdeki eşleşme verilerindeki öznitelik konumları esas alınır. Bağlam düzlemleri bu yöntemle göre sıralanmış giriş aktivasyon haritalarını işleyen ESA mimarilerinin performans sonuçları Çizelge 6.7'de gösterilmiştir.

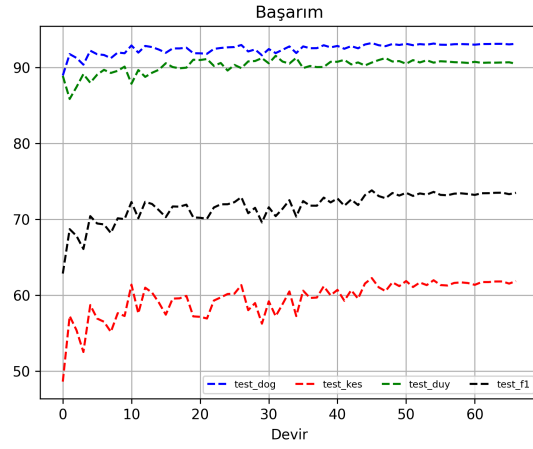
Çizelge 6.7: Bağlam-2 yapısı ESA mimarilerinin performans sonuçları.

Ağ	En	oD5	oD10	oD20
ESA-dar-2	6	26,65	48,84	67,86
ESA-geniş-2	6	26,69	48,63	68,06

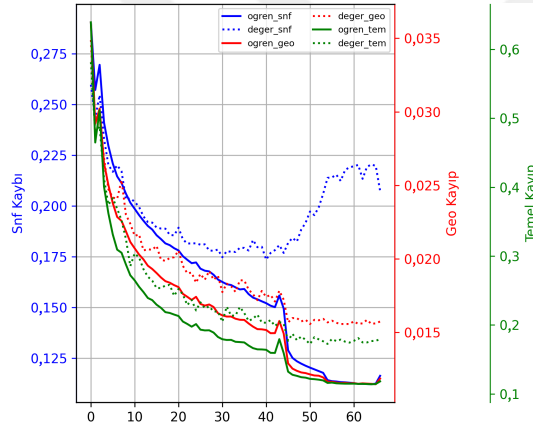
Öklid mesafelerine dayalı olarak sıralanan bağlam satırlarıyla elde edilen performans iyileşmesi Bağlam-1 yapısında eğitilen muadillerine göre %4-%7 arasındadır. Bağlam-1 yapısında yararlanılmayan bağlam satırlarının özel olarak sıralanabilir olması özelliği bu deneylerde yerel bağlam çıkarımına katkı sağlayarak performans artışı elde edilmesini sağlamıştır. Bağlam-2 yapısıyla görüntü çiftlerinden sadece 1 tanesine göre sıralı bağlam kanalı oluşturulmaktadır. Aynı satır sıralaması stratejisi uygulanarak, görüntü çiftlerindeki her iki görüntüyü ayrı ayrı referans alan bağlam kanallarının oluşturulmasıyla, Bağlam-2 yapısıyla elde edilen performans sonuçlarının daha yukarılara çekilmesi olasıdır. Bu bağlam yapısında, dar ve geniş ESA mimarilerinin performans farkı minimaldir. ESA-geniş mimarisi, performansı artırmayan sahte veri ilişkilerini öğrenmek yerine, ilgili ilişkileri yakalamayı öğrenmiştir. Bunun, bağlam satırlarının



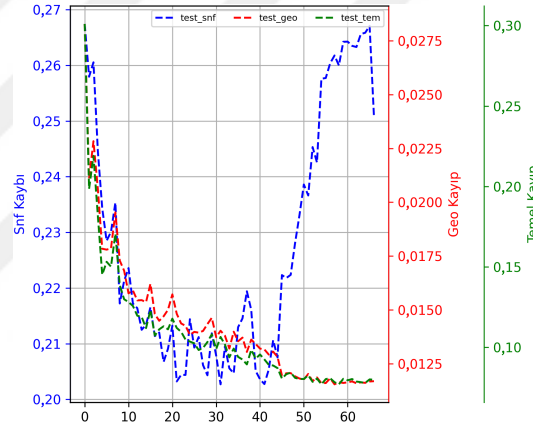
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



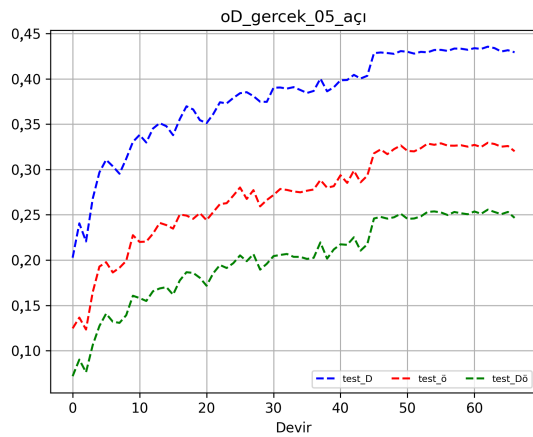
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



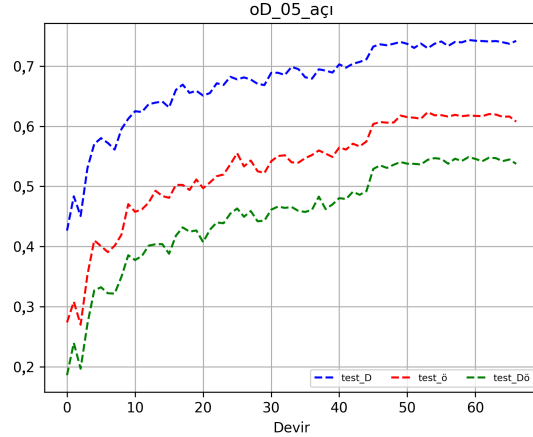
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

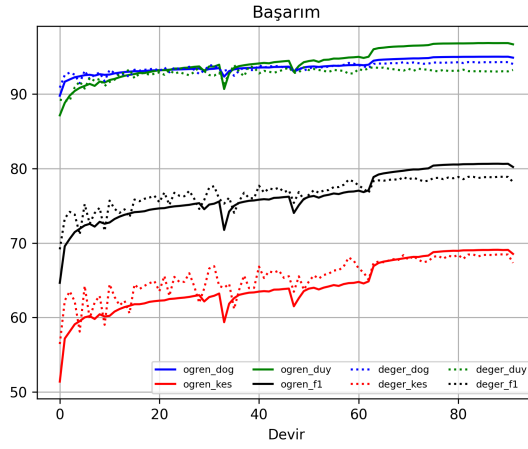


(e) Başarım-yamuk yöntemi (Test)

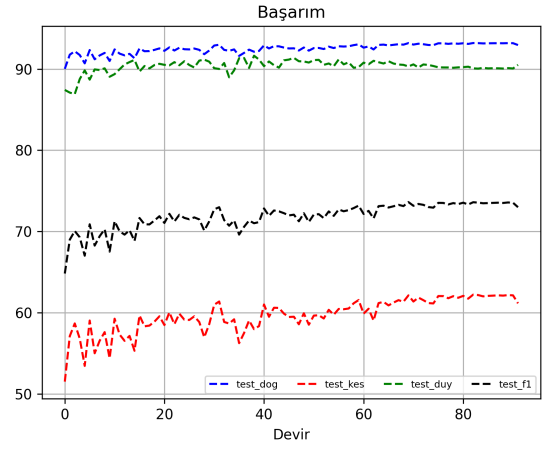


(f) Başarım (Test)

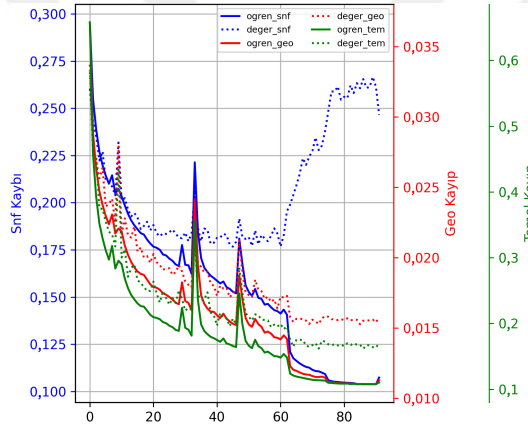
Şekil 6.21: ESA-dar-1 ağı başarımı.



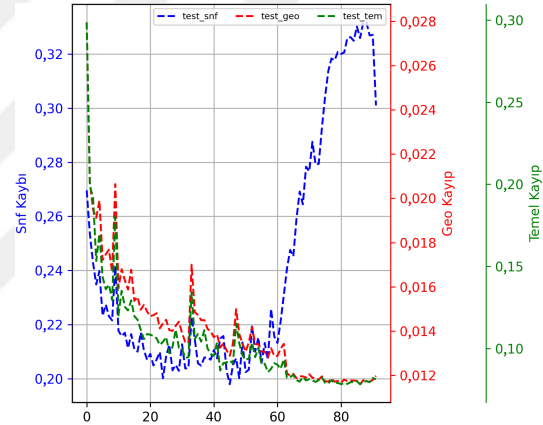
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



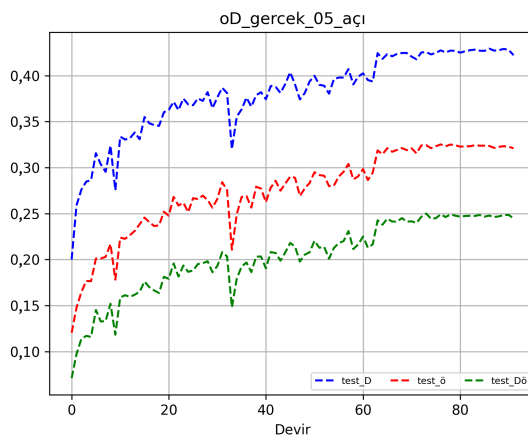
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



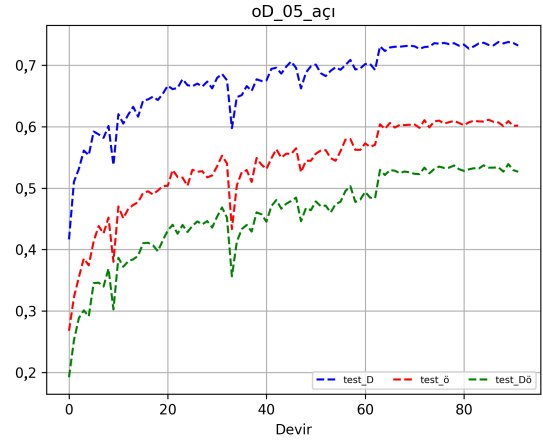
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)



(e) Başarımlar-yamuk yöntemi (Test)



(f) Başarımlar (Test)

Şekil 6.22: ESA-geiş-1 ağı başarımları.

disparitelere göre sıralanmasının sağladığı faydanın bir başka göstergesi olduğunu savunmaktayız. Eğitilen bu ağlara ait eğitim ve test grafikleri Şekil 6.23-6.24 de verilmiştir.

6.3.4 Bağlam-3 Yapısını Kullanan ESA Mimarilerinin Başarımları

ESA bloklarının iyi bağlam düzlemleri elde ederek blokların daha karmaşık ilişkiler çıkarmasına yardımcı olmak amacıyla başka bir sıralama yöntemi uygulanmıştır. Görüntü çiftlerindeki varsayılan eşleşmelerin disparite büyüklüğüne göre bağlam düzlemlerinin satırlarının sıralanması tercih edilmiştir. Disparite açılarının döngüsel doğası doğrusal bir sıralama oluşturmayı zorlaştırdığından sıralama işlemi sırasında disparite açılarının kullanılmaması tercih edilmiştir. Aynı bağlam düzlemi, bir görüntü çiftinin tüm yığın örneklerinin bağlam kanallarında kullanılmıştır. Bu yöntemin performansı Çizelge 6.8’te gösterilmiştir.

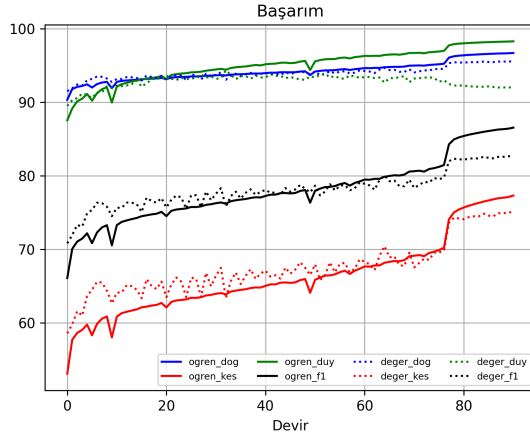
Çizelge 6.8: Bağlam-3 yapısı ESA mimarilerinin performans sonuçları.

Ağ	En	oD5	oD10	oD20
ESA-dar-3	6	28,81	50,54	69,14
ESA-geniş-3	6	28,80	50,55	68,97

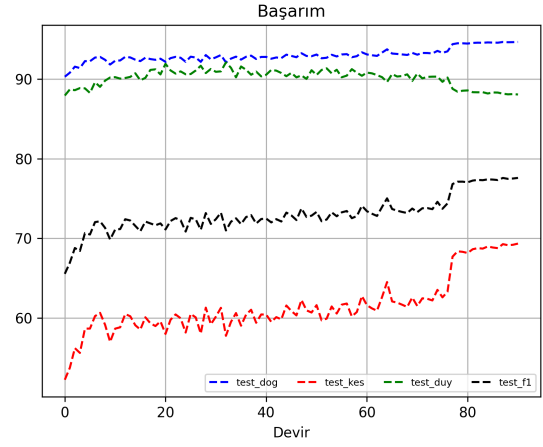
Burada kullanılan bağlam yapısı ile ulaşılan performans değerleri Bağlam-2 yapısı ile elde edilen muadillerine göre %8’den daha yüksektir. Benzer disparitelere sahip öz-nitelik noktalarını bağlam düzlemlerinin komşu satırlarına yerleştirerek yerel bağlam etkileşiminin geliştirebileceği sonucuna varılmıştır. Benzer disparitelere sahip öz-nitelik noktaları görüntülerde yakın konumlarda bulunmasalar dahi, yine de birbirleri için yerel bağlam sağlayarak ağların daha yüksek performansa ulaşmasına yardımcı olabilmişlerdir. Bağlam düzlemlerinin disparitelere göre sıralanmasının bir diğer avantajı, daha fazla ilişkili satırın birbirine yakın yerleştirilmesiyle, yanlış eşleme noktalarının enjekte ettiği gürültünün ağdaki baskınlığının azaltılabilmesidir. Benzer şekilde, bir araya toplanan doğru eşleşme noktalarının sayısı artar, bu da ESA bloklarının ilişkileri daha kolay çıkarmasına yardımcı olur. Bu bağlam yapısında da Bağlam-2 yapısına benzer şekilde dar ve geniş ESA mimarilerinin performans farkı minimaldir. Eğitilen bu ağlara ait eğitim ve test grafikleri Şekil 6.25-6.26 de verilmiştir.

6.4 Genel Başarım Karşılaştırması

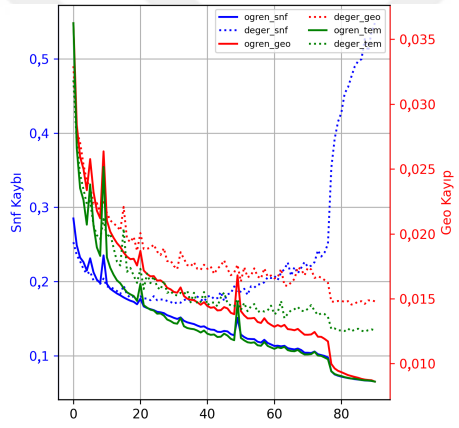
1-1 ağların performansları, bilinen $n-n$ ağların performansları ile Tablo 6.9’te karşılaştırılmıştır. Bu tezdeki çalışmalarda tüm mimariler doğrulama *geo_kayıp*’ında daha fazla iyileşme gözlemlenmeyene kadar eğitildiği için, bazı $n-n$ ağlar için literatürde sunulan performanslarına kıyasla önemli ölçüde daha üstün sonuçlar elde edilmiştir. Bu ağlar PointCN [4], OANET [5], OANETiter [5], MS2DGNET [22] varyantları ve ConvMatch [2] ağlarıdır. Öte yandan, CLNET [1] ve ConvMatchimp [3] için elde ettiğimiz performans sonuçları, ağların kendi çalışmalarında rapor ettikleri performans sonuçlarından daha düşüktür. Bu fark, bu ağların giriş aktivasyon haritalarında uygulanan



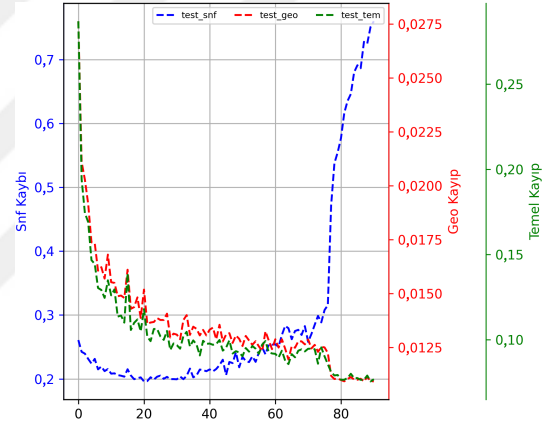
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



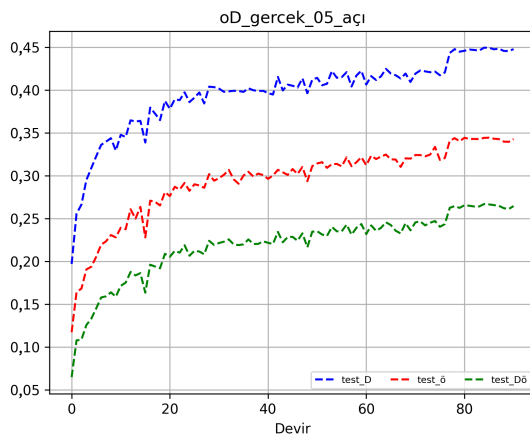
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



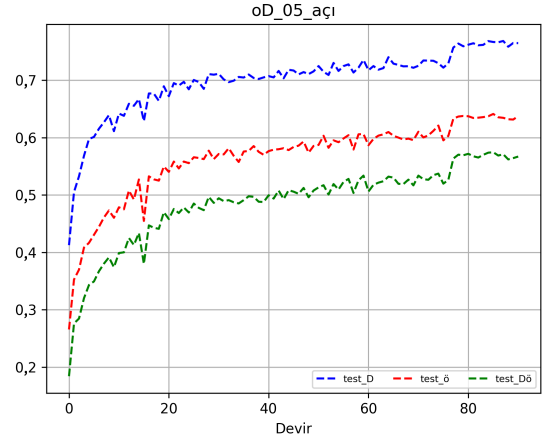
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

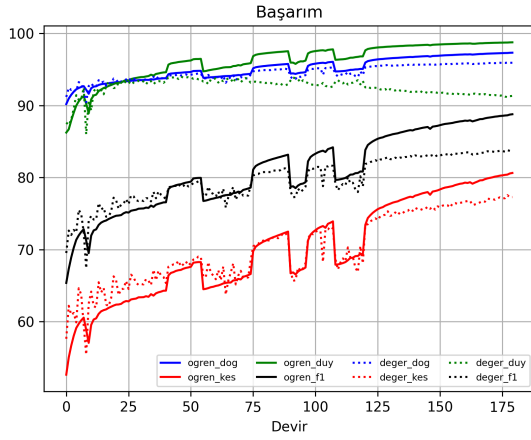


(e) Başarım-yamuk yöntemi (Test)

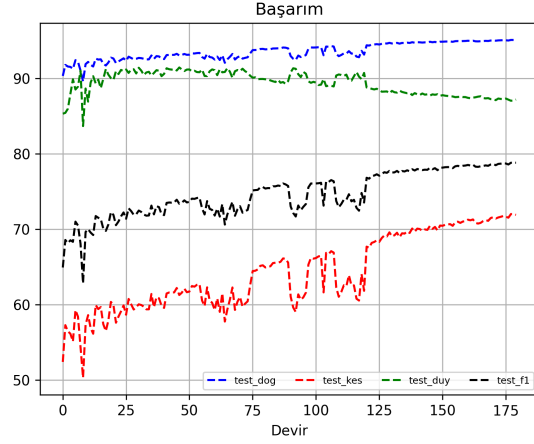


(f) Başarım (Test)

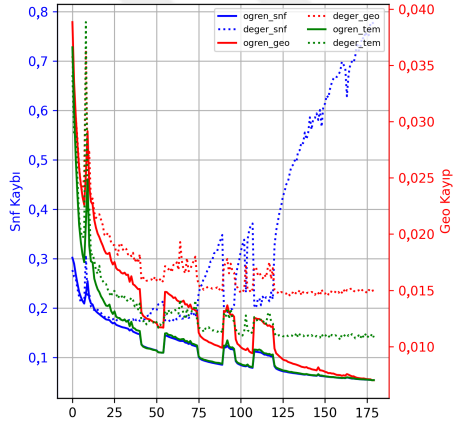
Şekil 6.23: ESA-dar-2 ağı başarımı.



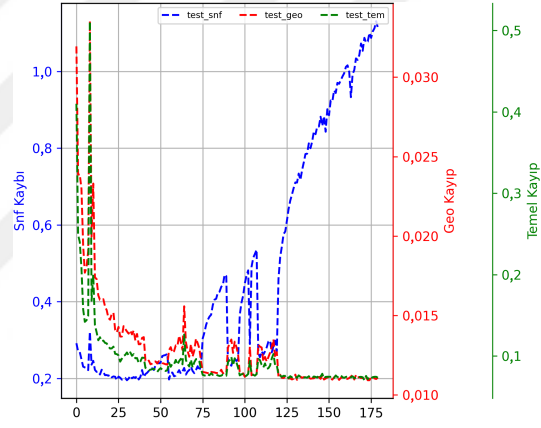
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



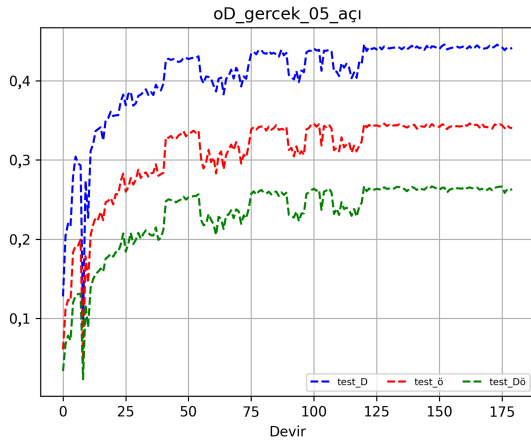
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



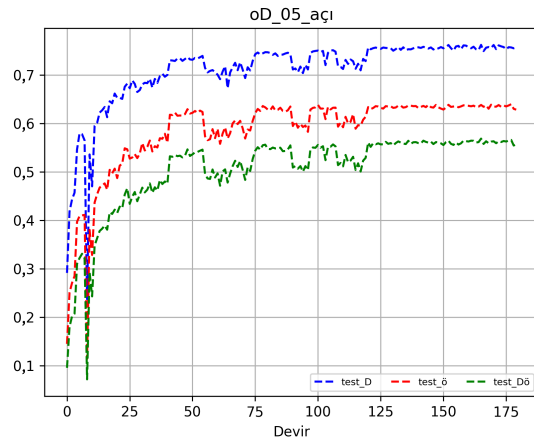
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

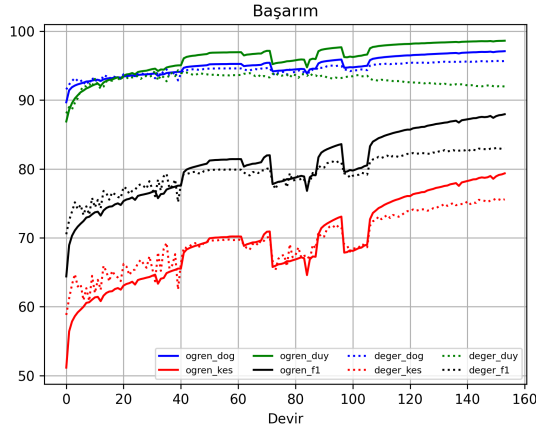


(e) Başarım-yamuk yöntemi (Test)

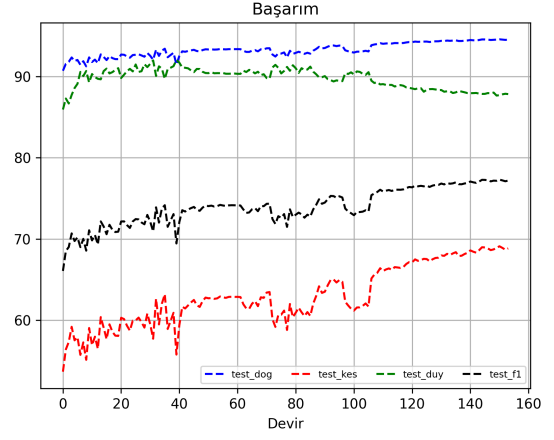


(f) Başarım (Test)

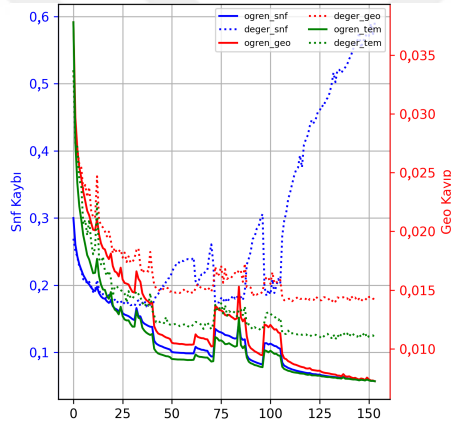
Şekil 6.24: ESA-geniş-2 ağı başarımları.



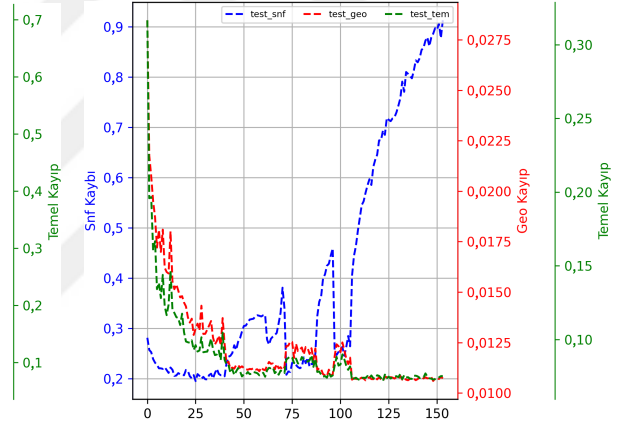
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



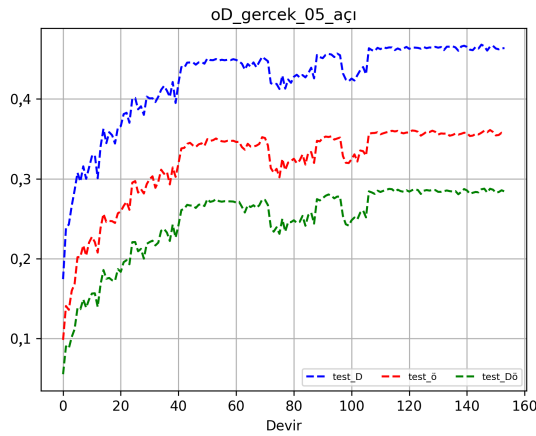
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



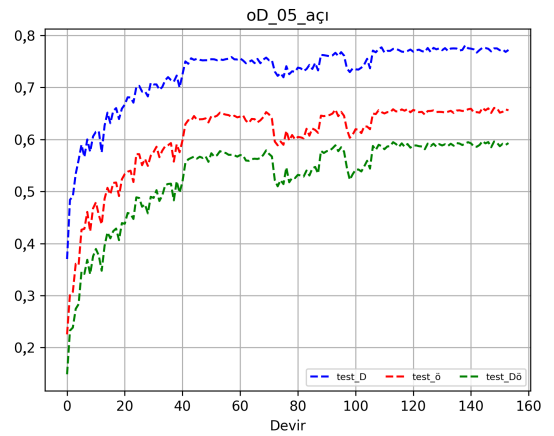
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

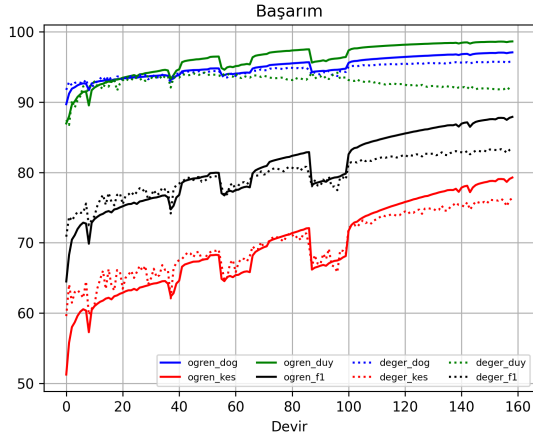


(e) Başarım-yamuk yöntemi (Test)

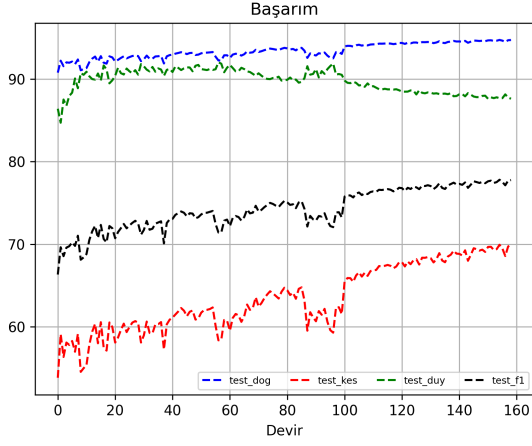


(f) Başarım (Test)

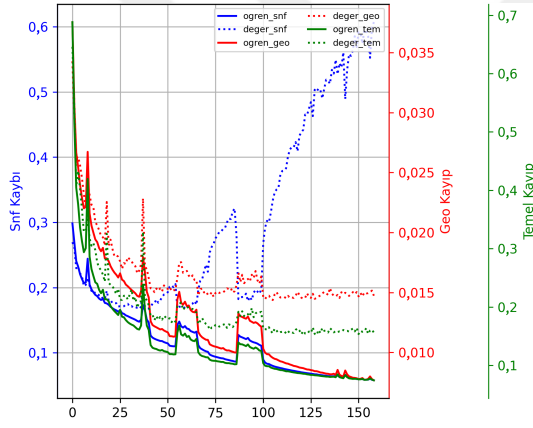
Şekil 6.25: ESA-dar-3 açığı başarımları.



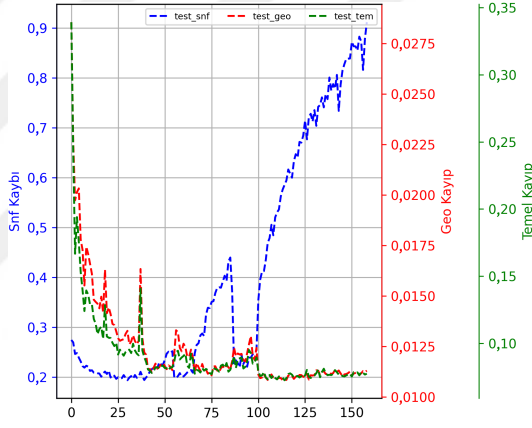
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



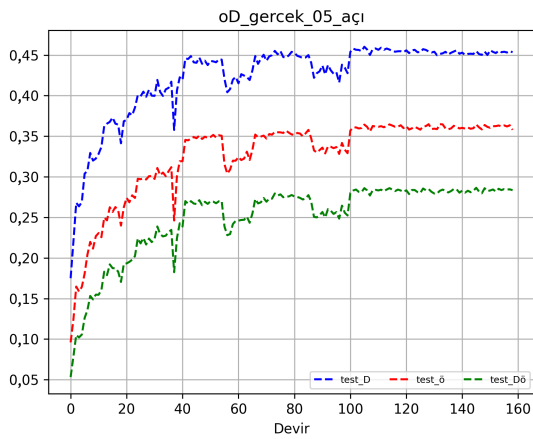
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



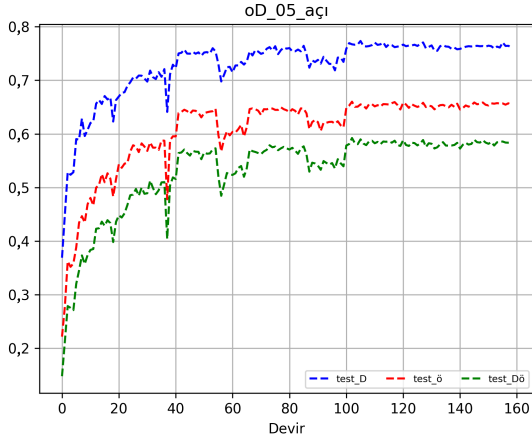
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)



(e) Başarım-yamuk yöntemi (Test)



(f) Başarım (Test)

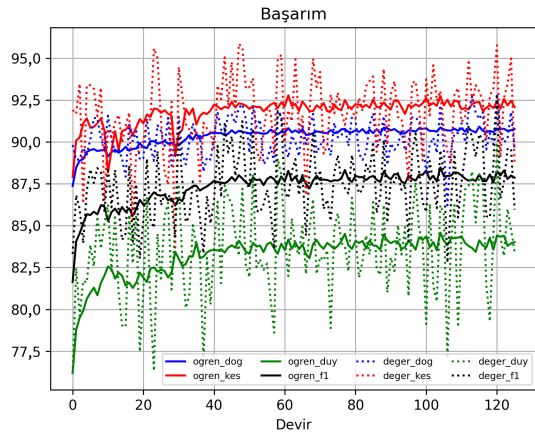
Şekil 6.26: ESA-geniş-3 ağı başarımı.

satır tekrarlarına uyum sağlayamamasından ya da TPU'lar üzerindeki uygulama farklarından kaynaklanıyor olabilir. MS2DGNET [22], eğitim için TPU'lar yerine GPU'ları kullandığımız tek ağıdır.

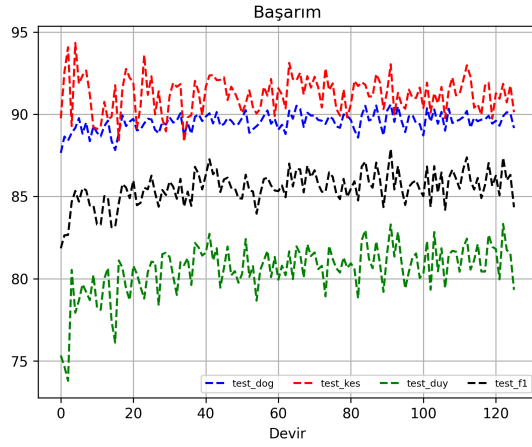
Çizelge 6.9: n - n ağlar ile 1 - 1 çatısındaki ağların performans kıyaslaması.

Ağ	En	oD5	oD10	oD20
PointCN [4]	6	15,75	34,56	55,41
OANET [5]	6	17,79	38,58	60,62
OANETiter [5]	6	19,87	41,38	62,41
CLNET [1]	4	20,57	41,49	62,05
MS2DGNET [22]	4	21,41	43,93	65,10
MS2DGNET [22]	6	20,65	42,94	64,46
ConvMatch [2]	4	33,03	55,10	72,72
ConvMatchimp [3]	4	32,56	54,86	72,40
SKD-dar	6	21,71	42,63	62,69
ESA-dar-0	6	24,32	45,75	65,41
ESA-dar-1	6	25,41	46,99	66,34
ESA-dar-2	6	26,65	48,84	67,86
ESA-dar-3	6	28,81	50,54	69,14

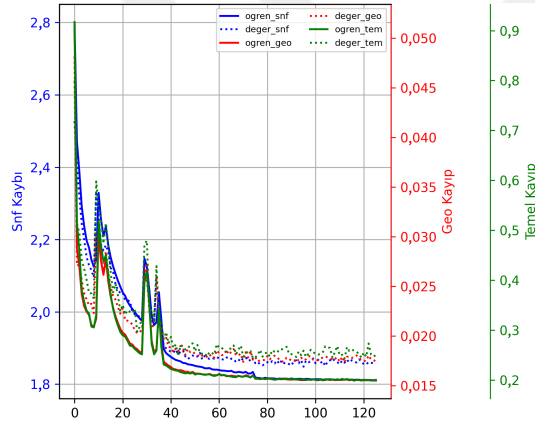
Sonuçlara göre, ConvMatch varyantları [2, 3] dışında, 1 - 1 çatısında eğitilen tüm ağlar, bilinen n - n ağlardan daha iyi performans göstermektedir. Bu, özel olarak sıralanmış bağlam kanalları kullanılsa bile, BEB'lerin sağladığı bağlam etkileşiminin ağlara öznitelik eşleşmelerini sınıflandırırken yardımcı olduğunu göstermektedir. ESA-dar-3 mimarisi, ConvMatch varyantlarına görece yakın performans göstermekte ve geri kalan n - n ağların tamamını önemli ölçüde geride bırakmaktadır. Bu performansa temel ESA bloklarıyla ulaşılması, 1 - 1 çatısının potansiyelini ortaya koymaktadır. Ayrıca, ESA-dar-2 ve ESA-dar-3 mimarileriyle elde edilen performans artışları, önerilen çatıda sıralı bağlam satırlarının faydalı olduğunu göstermektedir. Eğitilen bu ağlardan CLNET'e [1] ait eğitim ve test grafikleri Şekil 6.27'de, ConvMatch'e [2] ait eğitim ve test grafikleri Şekil 6.28'de ve ConvMatchimp'e [3] ait eğitim ve test grafikleri Şekil 6.29'de verilmiştir.



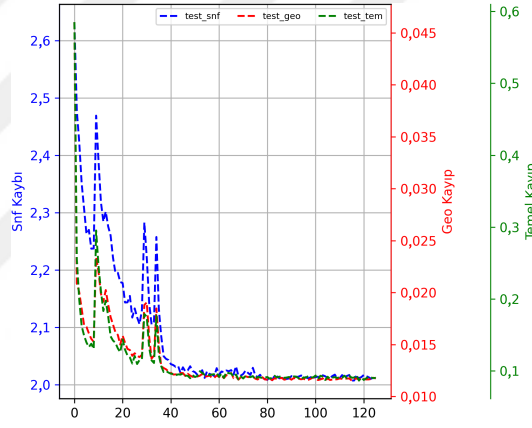
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



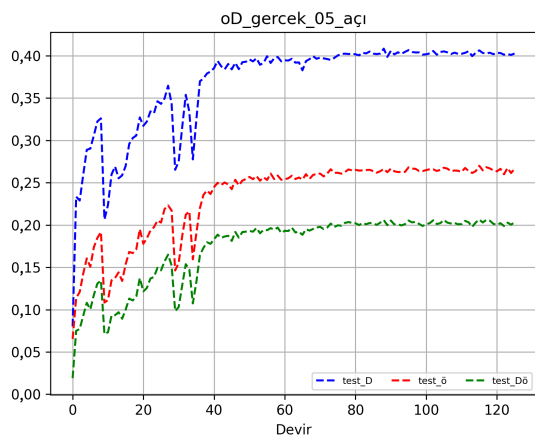
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



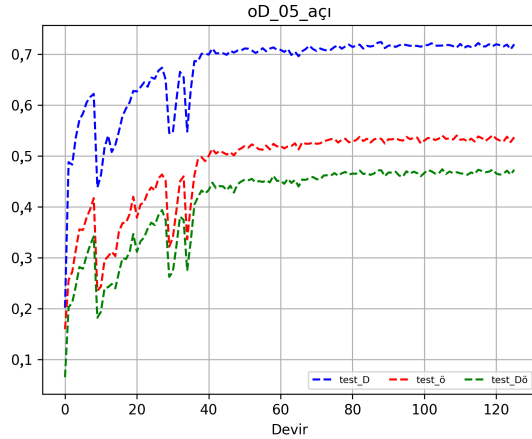
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

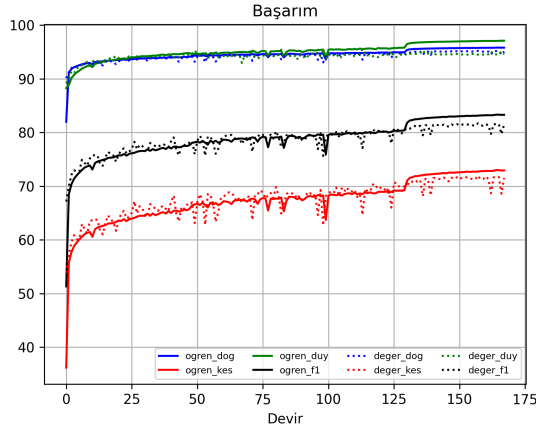


(e) Başarım-yamuk yöntemi (Test)

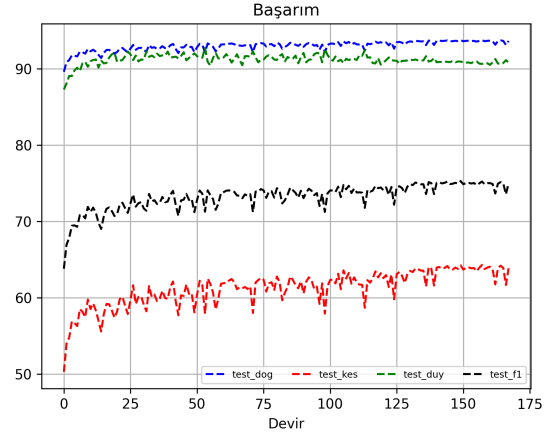


(f) Başarım (Test)

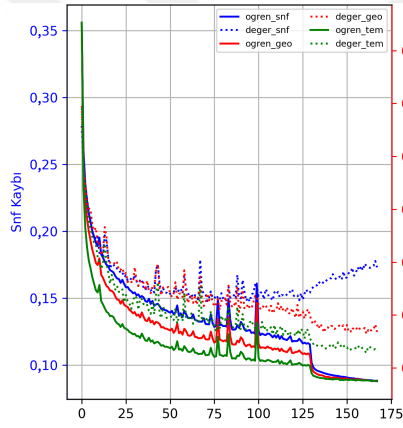
Şekil 6.27: CLNET [1] ağı başarımı.



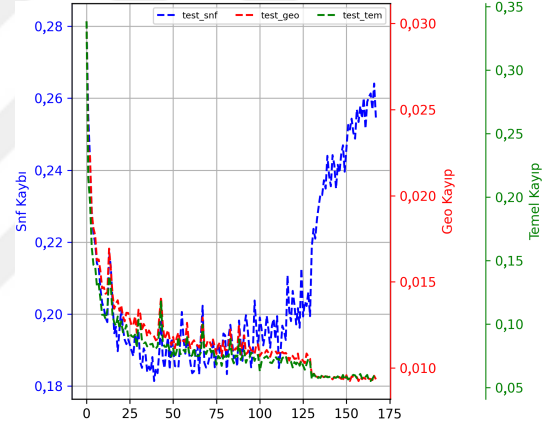
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



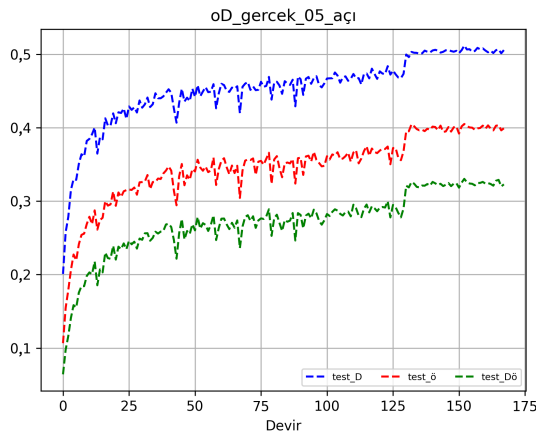
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



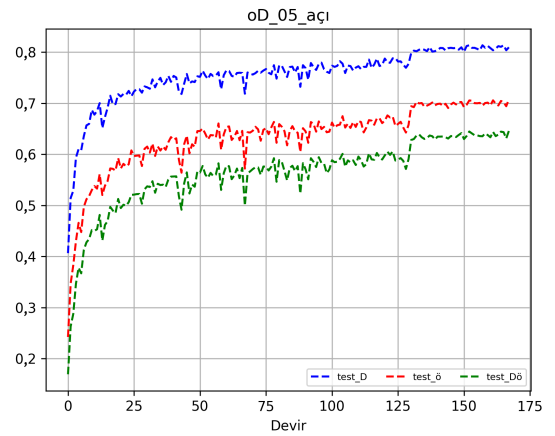
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)

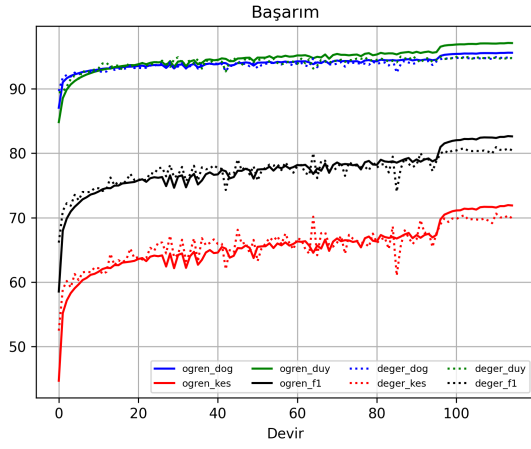


(e) Başarım-yamuk yöntemi (Test)

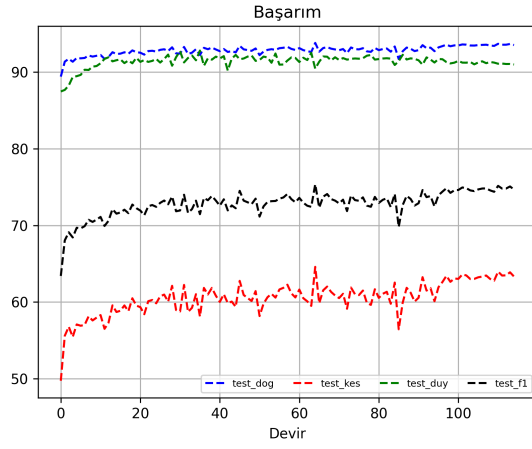


(f) Başarım (Test)

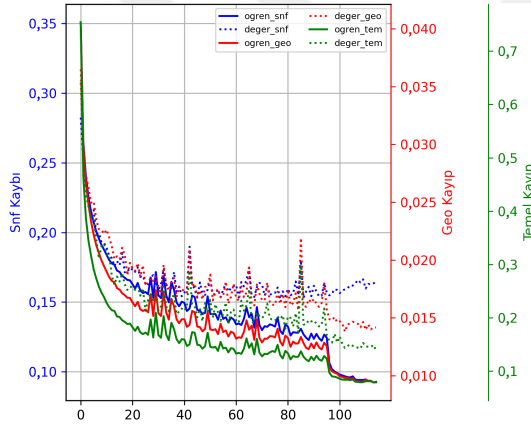
Şekil 6.28: ConvMatch [2] ağı başarımı.



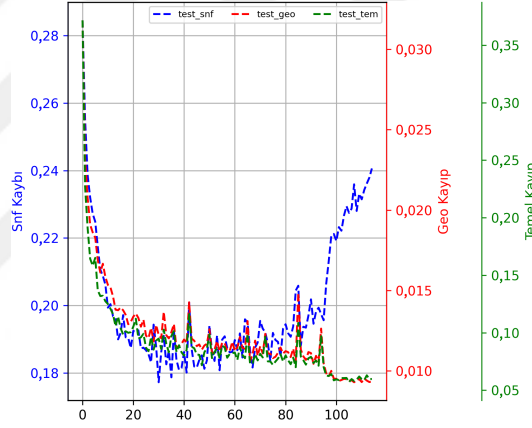
(a) Doğruluk-duyarlılık-kesinlik-f1 (Eğitim)



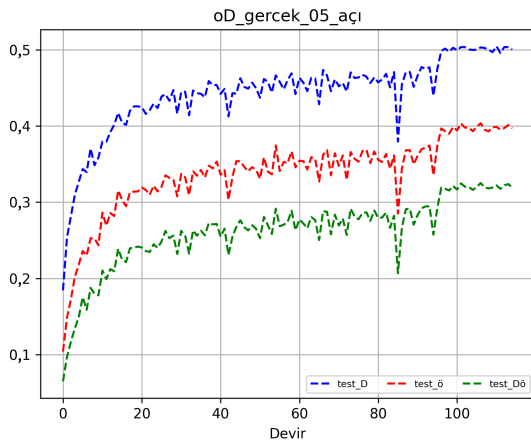
(b) Doğruluk-duyarlılık-kesinlik-f1 (Test)



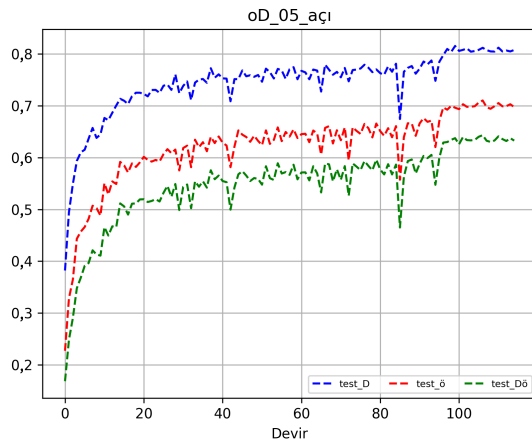
(c) Kayıplar (Eğitim)



(d) Kayıplar (Test)



(e) Başarım-yamuk yöntemi (Test)



(f) Başarım (Test)

Şekil 6.29: ConvMatchimp [3] ağı başarımı.

7. DURUM ÇALIŞMASI: YAŞ SINIFI TAHMİNİ

7.1 Durum Çalışmasının Amacı

2010'lu yıllardan önce bilgisayarlı problemlerin büyük bir kısmı derin öğrenme tabanlı olmayan klasik yöntemler ile çözülmekteydi. Erişilebilir veri miktarındaki artışlar ve yüksek paralel işlem yapabilme kapasitesine sahip donanımların geliştirilmesiyle bilgisayarlı görü problemlerinin klasik çözümlerine alternatif olarak derin öğrenme tabanlı çözümlerin kullanılmasına imkân bulunmuş oldu. Derin öğrenme tabanlı çözümlerin klasik çözümlere göre daha başarılı olduğu problem sayısı arttıkça araştırmacıların derin öğrenme tabanlı çözümler üretme motivasyonları katlanarak arttı.

Bu tezin asıl konusu üzerinde çalışmaya başlamadan önce, derin öğrenme tabanlı çözümlerin geliştirilmesine aşinalık sağlamak amacıyla yaş sınıflandırma problemi seçilmiş ve bu problem üzerine bir durum çalışması gerçekleştirilmiştir. Her ne kadar durum çalışması MatConvNet [32] kütüphanesi kullanılarak yapılsa da MatConvNet'in daha fazla güncellenmemesi, MatConvNet'in kullanım alanının çok dar olması ve diğer derin öğrenme kütüphanelerinin çok daha yaygın olarak kullanılmasından dolayı tez çalışmasının diğer kısımlarında MatConvNet kullanılmamıştır.

7.2 Yaş Tespiti Problemi

Yüz görüntülerinden otomatik yaş tahmini, bilgisayarlı adli bilişim, insan bilgisayar etkileşimi, biyometri, eğlence, örüntü tanıma ve bilgisayarlı görü gibi çeşitli alanlarda incelenen önemli bir problemdir. Doğru bir yaş tahmini, bir kişinin daha genç veya daha yaşlı görüntülerinin üretilmesi gibi uygulamalar için temel oluşturur. Üretilen bu görüntüler kullanılarak kayıp bir kişiyi veya suçluyu bulmak kolaylaşabilmektedir. Bir yaş tahmin sistemi insan sağlığına zararlı alkol ve tütün gibi ürünlerin otomatlarda yaşı tutmayan bireylere satılmasını engellemek için kullanılabilir. İnsanların tercihleri yaşlarına göre değiştiğinden, otomatik yaş tahmininin bir dizi potansiyel uygulaması da ortaya çıkmaktadır. Bu uygulamalarda, bireyin kimliğinin belirlenmesi yerine, bireyin yaşının tahmin edilmesi yeterli olmaktadır. Bahsedilen bu farklı kullanım alanları sebebiyle yaş tahmini yapan algoritmalar hem endüstride hem de akademik çevrelerde büyük ilgi görmüştür.

Yaş tespiti sorunun çözümündeki ana zorluk, farklı insanların farklı yaşlanma kalıplarına sahip olmasından kaynaklanmaktadır. Yaşlanma kalıpları genler, cinsiyet, yaşam tarzı, etnik köken ve ırk gibi hem iç hem de dış faktörlere bağlıdır [33]. Bu nedenle, bir kişinin gerçek yaşı, görünüş yaşından farklı olabilir. Bir yüz görüntüsü verildiğinde, bilgisayar tabanlı yaş tahmininin amacı, yüze, ait olduğu kesin yaş veya yaş grubunu

belirten bir etiket üretmektir. Bu tezde yapılan çalışma kapsamında, kesin yaş tahmininden ziyade yaş grubu sınıflandırması problemine odaklanılmıştır.

Yaş ve yüz arasındaki ilişkiyi, insan yüzleri üzerindeki yaşlanma etkilerini simüle etme bağlamında inceleyen çalışmalar vardır [34–37]. Burt ve arkadaşları [34] çalışmalarında yaşlanma değişimlerini, yüz görüntülerine yaşlanma sonucu meydana gelen tipik şekil ve renk değişikliklerinin bindirilmesiyle simüle etmişlerdir. Hutton ve arkadaşları [37] çalışmalarında, büyüme ve yaşlanmaya bağlı şekil değişikliklerini ifade etmek için yoğun yüzey nokta dağılım modeli sunmuşlardır. Bu teknikler doğrudan yaş tahmini yapmasa da yaştan yüze olan haritalamaya ve çeşitli otomatik yaş tahmin çerçevelerine ilham vermektedir.

En erken yaş tahmin çatılarından biri, Lanitis ve arkadaşları [38] tarafından geliştirilmiştir. Bu çalışmada, yüz görüntülerinin kompakt parametrik bir tanımını elde etmek için kullanılan istatistiksel bir yüz görünümü modeli oluşturulmuştur. Yaş tahmin edici tasarımı en kısa mesafe sınıflandırıcısı, denetimli ve denetimsiz yapay sinir ağları kullanılmıştır. Umut verici sonuçlar bildirilmiş olmasına rağmen, yaşlanma fonksiyonunun ampirik olarak belirlenmesi, yalnızca bireyin yüz görüntülerine dayalı olarak bireyin yaşlanma modelinin öğrenilmesi ve daha önce görülmemiş bir yüz görüntüsü için yaşlanma fonksiyonunun bilinen yaşlanma fonksiyonlarının basit bir lineer kombinasyonu olarak hesaplanması, bu yaklaşımın başlıca dezavantajları arasında yer almaktadır.

Bu sorunları ele almak için Geng ve arkadaşları [33], AGES çerçevesini sunmuşlardır. Burada yaşlanma modeli, belirli bir kişinin yüz görüntülerinin zamana göre sıralanmış dizisine dayalı olarak temsil edici bir alt uzay elde edilerek modellenmiştir. Görülmemiş bir yüz görüntüsünün yaşlanma örüntüsü, bu alt uzaydaki projeksiyon yoluyla, yüz görüntüsünü minimum yeniden yapılandırma hatası ile yeniden oluşturacak şekilde hesaplanır. Tam yaşlanma örüntülerinin eksikliği, eksik eğitim verisine yol açmaktadır. Bu sorunu çözmek için, küresel yaşlanma modeli kullanılarak eksik kişisel yaşlanma örüntüsünün bir kısmını tahmin eden bir iteratif öğrenme algoritması benimsenmiştir.

Ricanek ve diğerleri [39], ilgili yaşlanma özelliklerini elde etmek ve en önemli olanları En Küçük Açık Regresyonu (EKAR, Least Angle Regression) yöntemi ile belirlemek için Aktif Görünüm Modeli (AGM, Active Appearance Model) kullanmıştır. AGM, yüz görüntüsünün şekil ve dokusunu bir dizi parametre ile tanımlayabilen bir yöntem olup, uygun bir eğitimden sonra görünüm özelliklerini çıkarmak için çeşitli tekniklerde de kullanılmıştır [40, 41]. AGM'nin yüz yaşlanması üzerindeki performansı, daha sonraki bir çalışmada [42] incelenmiştir. Literatürde yüz görüntüleri üzerinden yaş tahmini üzerine bir derleme çalışması [43] bulunmaktadır.

Tez kapsamındaki bu çalışmada, yaş sınıflandırması için üç farklı mimariye sahip derin ESA'ların performansları incelenmiştir. Derin ESA kullanma motivasyonumuz, çalışmayı yaptığımız zaman diliminde bu ağların görüntü sınıflandırma [44, 45] ve nesne tespiti [46, 47] gibi çeşitli bilgisayarlı görü görevlerinde büyük başarı sağladığını gösteren çalışmalardan gelmektedir. ESA'lar ile, tüm sistemin uçtan uca eğitilmesi sonucu oluşan düşük, orta ve yüksek seviyeli özelliklerden oluşan uygun bir özellik vektörü elde edilir. Yaş tahmini için ESA kullanımı yeni bir yaklaşım değildir, zira yakın zamanda böyle bir yaklaşım [48] sunulmuştur. Bu yaklaşımla başarılı sonuçlar sergilen-

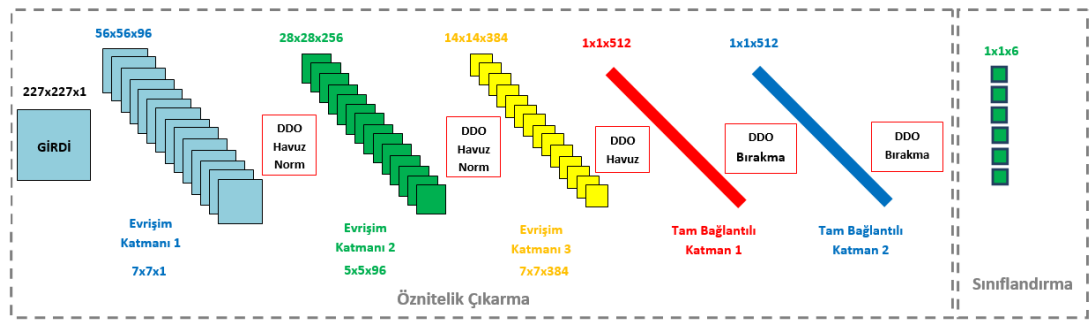
miş olmasına rağmen, önerilen ağın yalnızca üç evrişim katmanı ve az sayıda nörona sahip iki tam bağlantılı katmandan oluşan basit bir ağ mimarisine sahip olması irdelenmeye değer görülmüştür. Öte yandan, ağ derinliğinin önemli bir meta parametre olduğu gösterilmiştir [49] ve en iyi görüntü sınıflandırma tekniklerinin çok daha derin modellerden [50, 51] yararlandığı da bilinmektedir.

Bu tez kapsamındaki çalışmalarımızda, daha derin ESA'lar ile daha iyi yaş sınıflandırma skorları elde edilemeyeceğini görmek için, sırasıyla 6, 18 ve 34 katmandan oluşan [48], ResNet-18 ve ResNet-34 [52] mimarileri kullanılmıştır. ResNet-18 ve ResNet-34, 1.000 sınıf içeren ImageNet sınıflandırma veri setinde [53] başarıyla kullanılıyor olması motivasyonumuzu arttıran temel faktörlerdendir. Deneyler için mahkumlara ait, 55 binden fazla önden çekilmiş yüz görüntüsü içeren Craniofacial Longitudinal Morphological Face Database (MORPH) görüntüleri [6] kullanılmıştır.

7.3 Ağ Mimarileri ve Veri Seti

7.3.1 6 Katmanlı Ağ

6 katmanlı ESA'nın mimarisi [48] Şekil 7.1'de gösterilmiştir. Ağ, üç evrişim katmanı ve üç tam bağlantılı katmandan oluşmaktadır. Her bir katmanın veri hacimleri şekilde katmanların üst kısımlarında belirtilmiştir. Orijinal ağ mimarisi 227x227 boyutunda RGB görüntüler kullanmasına rağmen, bu çalışmada aynı çözünürlükteki gri tonlamalı görüntüler kullanılmıştır. Üç evrişim katmanından her birini, DDO katmanları ve iki piksel adımı 3x3 bölgelerinin maksimum değerini alan yığın normalizasyonu NORM katmanları takip etmektedir. İlk iki tam bağlantılı katmandan sonra, DDO ve 0,5 bırakma oranına sahip bırakma katmanları gelmektedir. Son tam bağlantılı katman, referans çalışmada 8 yaş sınıfı için sınıflandırma skorları üretirken bu çalışmada veri seti 6 sınıfa ayrıldığından dolayı son tam bağlantılı katman 6 yaş sınıfı için skor üretecek şekilde oluşturulmuştur.

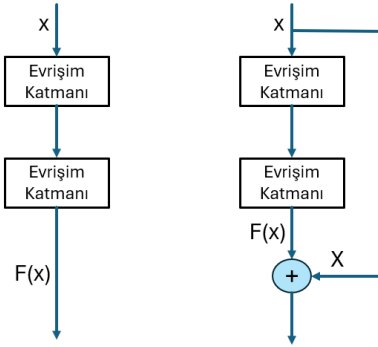


Şekil 7.1: 6 sınıflı yaş sınıflandırması için 6 katmanlı ESA mimarisi.

7.3.2 RESNET

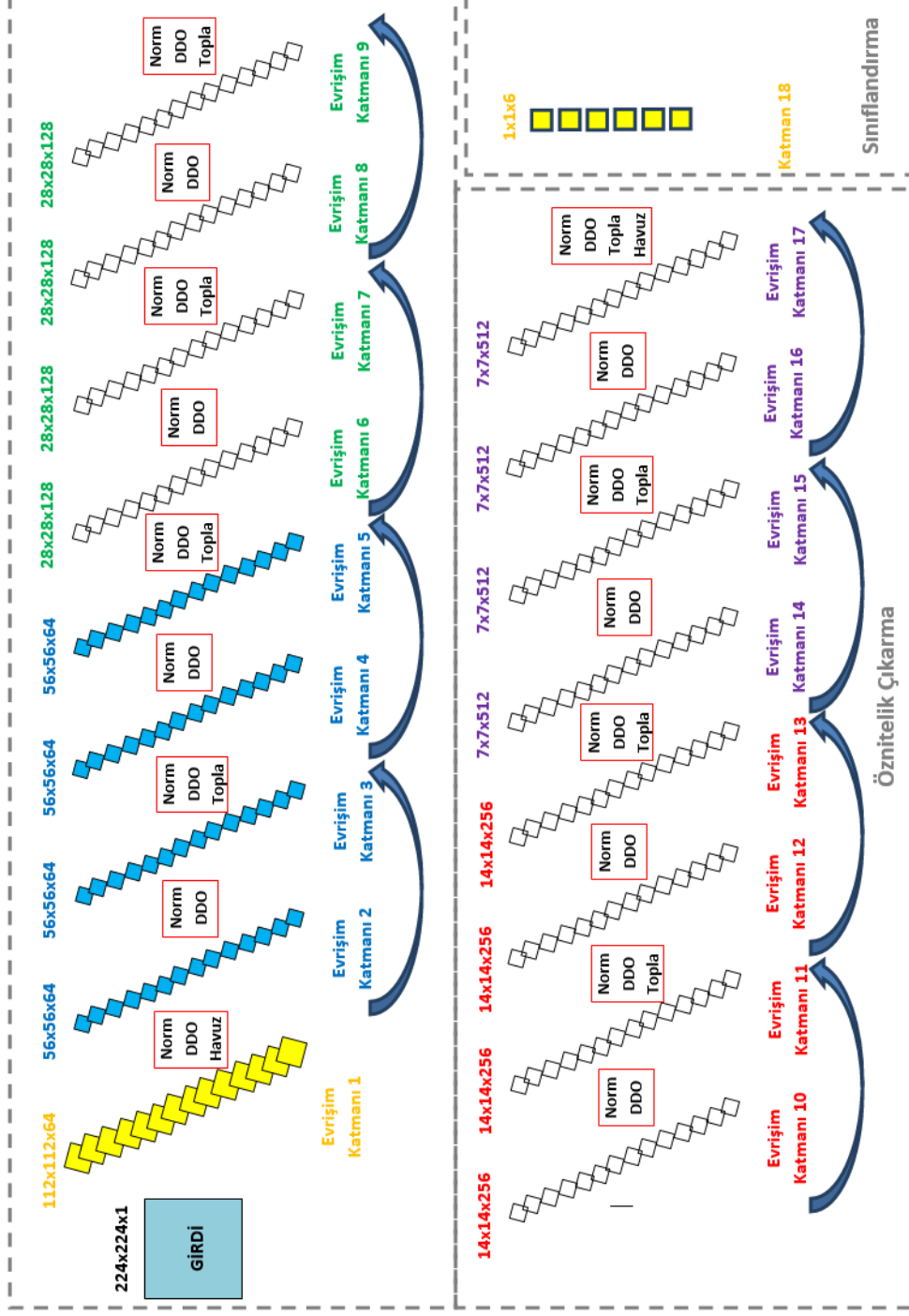
Yapı taşları Şekil 7.2'in solunda gösterilen düz evrişim katmanlarının birleşimiyle oluşan ağlardan elde edilen başarımlar, ağlar derinleştikçe ortaya çıkan gradyan geri yayılım

sorunu sebebiyle kötüleşme eğilimindedir [52, 54, 55]. Gradyan geri yayılım sorununu minimize ederek karmaşık problemlere daha derin ağlar kullanarak çözümler üretmek amacıyla artık ağlar kullanılmaktadır. Literatürde artık ağların daha önceki uygulamaları bulunmasına rağmen, artık ağların avantajları yakın tarihli bir çalışmada [52] tüm ayrıntılarıyla tartışılmıştır. Bu çalışmadaki tartışmaya göre, artık ağlar düz ağların derinleştikçe daha kötü başarımlar elde etmesi sorununa karşı daha dirençlidir. Bir ağda artık bağlantılar oluşturmaktaki ana motivasyon, düzenli bağlantılara alternatif bağlantılar sağlamaktır. Şekil 7.2'in sağında gösterildiği gibi, artık bağlantılar aktivasyon haritalarının kendinden sonraki aktivasyon haritalarına olduğu gibi ulaşmasını sağlamaktadır. Eğer birbirine artık bağlantılar vasıtasıyla bağlanan iki aktivasyon haritasının büyüklüğü aynı ise artık bağlantılar herhangi bir eğitilebilir bir parametre içermezler. Eğer iki aktivasyon haritasının boyutları farklı ise artık bağlantılar sadece bu boyut dönüşümünü yapacak kadar eğitilebilir parametre içerirler. Bir eğitim sürecinde, eğer düzenli bir bağlantının katsayısı sıfıra yaklaşma eğilimindeyse, artık bağlantı ağın bütünlüğünü sağlar. Düz bir bağlantı, artık bir bağlantı tarafından kısa devre yapıldığında, düzenli bağlantıdan önce hesaplanan aktivasyon haritasındaki veri, ağın geri kalanına iletilir. Artık bağlantıların kullanılması, ağ eğitilirken gerektiğinde artık bağlantıların düz bağlantılara alternatif olarak kullanılabilmesi olanağını sağlar.



Şekil 7.2: Düz bağlantılar ve artık bağlantılar.

Yayınlandığı tarih göz önünde bulundurulduğunda, ResNet [52], farklı amaçlar için kullanılan en başarılı artık ağıdır. ResNet temel alınarak, farklı artık ESA mimarileri oluşturulmuş ve ILSVRC ve COCO yarışmalarına sunulmuştur. Bu yarışmalarda sunulan ağlar, ImageNet sınıflandırma, ImageNet tespit, ImageNet yerelleştirme, COCO tespit ve COCO segmentasyon görevlerinde birincilik kazanmıştır. Bu çalışmada kullanılan ve 18 evrişim katmanından oluşan bir ResNet mimarisi örneği Şekil 7.3'de gösterilmektedir. Bu mimaride, her bir evrişim katmanından sonra NORM ve DDO işlemleri kullanılmaktadır. Artık bağlantılar toplam işlemleri ile gerçekleştirilmekte ve kavisli oklar, toplama işlemlerine bağlanan artık bağlantıları göstermektedir. Dört grup evrişim katmanından sonra, son evrişim katmanı, girdi görüntülerine sınıflandırma işlemi uygulamak üzere yerleştirilmiştir. Daha önce tartışılan 6 katmanlı ağdan farklı olarak, ResNet mimarisi herhangi bir bırakma katmanı içermemektedir. ResNet mimarisinin, bazılarının 1.000'den fazla evrişim katmanına sahip olduğu farklı versiyonları vardır. ResNet versiyonları arasında, derinlik ve başarı açısından 6 katmanlı ağ ile karşılaştırmak üzere ResNet-18 ve ResNet-34 seçilmiştir.



Şekil 7.3: 6 sınıflı yaş sınıflandırması için ResNet-18 mimarisi.

7.3.3 Veri Seti

Bu makalede kullanılan görüntü veri tabanı, MORPH [6] veri tabanının ikinci albümüdür. Veri setindeki görüntüler, beş yıllık bir dönem içinde bireylerin yüzleri görünür şekilde önden çekilmiş fotoğraflarından oluşmaktadır. Resim 7.1’de üst 2 sırada veri setindeki orijinal görüntüler gösterilmiştir. Veri seti, yaşları 16 ile 77 arasında değişen 13.000’den fazla kişiye ait 55.134 görüntü içermektedir. Veri setindeki medyan yaş 33’tür ve her bir kişi için ortalama olarak 4 görüntü bulunmaktadır. Kullanıldığı tarihte araştırma amaçları için kullanılabilen en büyük veri setlerinden biri olup, veri setinin büyüklüğü öğrenme deneyleri için görece olarak yeterlidir. Ayrıca, veri setindeki kişiler farklı ırkları ve ruh hallerini temsil ettiğinden, veri seti ağların eğitimi için zengin bir girdi çeşitliliği sağlamaktadır.



(a) Orijinal görüntüler.

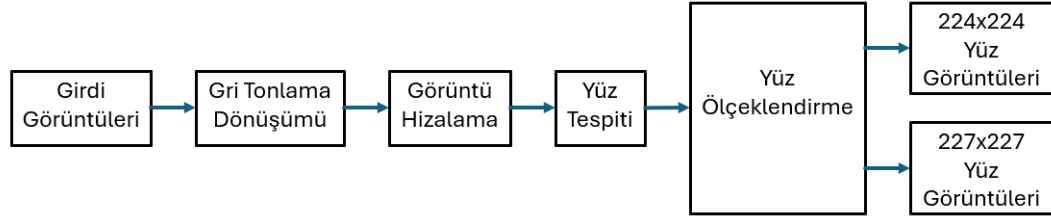


(b) Ön işleme çıktısı görüntüler.

Resim 7.1: Kullanılan Morph Veri Setine [6] ait örnek görüntüler.

7.4 Görüntü Ön-İşleme

Öğrenme sürecinin başarısını artırmak amacıyla, girdi görüntülerini çıktı sınıflarına bağlayan amaç fonksiyonu basitleştirilmiştir. Her bir görüntüden ağa gelen faydalı bilgiyi eşitlemek amaçlanmıştır. Bunu yapmak için, veri tabanındaki görüntülere gri tonlama dönüşümü, görüntü hizalama, yüz tespiti ve yüz yeniden ölçeklendirme işlemleri uygulanmıştır, bu durum Şekil 7.4’de gösterilmektedir.



Şekil 7.4: Görüntü ön-işleme adımları.

7.4.1 Görüntü Hizalama

Veri setindeki tüm ham görüntüler gri tonlamalı hale dönüştürüldükten sonra, görüntü hizalama işlemi uygulanır. Bu adımla, tüm yüzlerin dikeyde görüntü kenarlarıyla paralel olması amaçlanmıştır. Görüntü hizalama işlemi iki aşamadan oluşan bir yöntemle gerçekleştirilir. İlk aşamada, göz çiftleri bulunarak ve göz çiftlerini bağlayan çizginin yönüne göre görüntülere döndürme işlemi uygulanır. Göz çiftlerinin bulunamaması ve bu nedenle görüntülerin ilk aşamada başarıyla döndürülememesi durumunda, yüz yönlerini bulunur ve yöntemin ikinci aşamasında görüntüler yüzlerin yönlerine göre döndürülür.

Daha detaylı anlatılacak olunursa, yöntemin ilk aşamasında, her iki göz ayrı ayrı tespit edilir ve gözlerin dikey pozisyonlarına göre gerekli döndürme açıları belirlenir. Gözleri tespit etmek için Viola Jones algoritmasının iki farklı varyantı kullanılır [56]. Sınıflandırma ve regresyon ağacı (SRA) [57] tabanlı göz tespit algoritması, veri setindeki 55.134 görüntünün 54.121'inde doğru göz pozisyonlarını vermiştir. Karar kütüğü tabanlı göz tespit algoritması [58], kalan görüntülerin 500'ünde göz pozisyonlarını belirlemiştir. Veri setinde kalan 523 görüntü için göz tespit algoritmaları, gözlerin doğru pozisyonlarını verememiştir. Bu görüntüler için ikinci aşamaya geçilerek görüntü hizalama için göz tespiti yerine yüz tespiti yönteminden faydalanılmıştır. İkinci aşamada, döndürme açılarını elde etmek için yüz tespiti amacıyla Viola Jones algoritmasının iki varyantı kullanılmıştır. Bunun için öncelikle her bir görüntü -45° ile 45° arasındaki açılar için 1° 'lik adımlarla görüntünün döndürülmüş versiyonları oluşturulur. Döndürülmüş görüntüler üzerinde Yerel İkili Desenler (YİD) tabanlı yüz tespiti [59] ve SRA tabanlı yüz tespiti [60] algoritmaları uygulanarak iki algoritma için de hangi açılarda başarılı yüz tespiti işlemlerinin yapılabildiği kaydedilir. Başarılı olarak yüz tespiti yapılabilen döndürme açılarını ortalaması hesaplanır ve hesaplanan bu ortalama açığa göre görüntülere döndürme işlemi uygulanarak görüntü hizalaması gerçekleştirilmiş olur.

7.4.2 Yüz Tespiti

Veri tabanından elde edilen hizalanmış görüntülere uygulanan bir sonraki işlem yüz tespitidir. Yaş sınıflandırması için bir ESA ağı eğitilirken, yüz bölgelerinin dışındaki piksellerin değerli bilgi sağlamayacağı açıktır. Bu nedenle, yüze ait piksellerin dışında kalan pikseller, hizalanmış görüntülerden kırılmaktadır. Bu adımda tespit edilen yüz-

ler, dikey ve yatay piksel sayıları eşit olan kare bölgelerdir. Görüntülerdeki yüzleri tespit etmek için, önceki görüntü döndürme adımına benzer şekilde Viola Jones yüz tespit algoritmasının iki varyantı kullanılır. İlk olarak, SRA tabanlı yüz tespit algoritması [60] görüntülere uygulanır. Bu algoritma yüzleri bulmada başarılı olmazsa, YİD tabanlı yüz tespiti kullanılır. Yüz tespit sonuçlarına göre, SRA tabanlı yüz tespit algoritması veri setindeki döndürülmüş 55.038 görüntü için yüz bulmada başarılı olmuştur. YİD tabanlı yüz tespit algoritması, kalan görüntülerin 35'inde yüzleri bulmuş ve geriye kalan 61 görüntünün hiçbirinde bu algoritmalarla yüz tespit edilememiştir. Ayrıca, SRA tabanlı yüz tespiti ile yapılan tespitlerden 41'inin ve YİD tabanlı yüz tespiti ile yapılan tespitlerden 9'unun hatalı olduğu tespit edilmiştir. Tüm veri setini kullanabilmek için, yüz tespiti yapılamayan 61 görüntüde ve hatalı yüz tespiti yapılan 50 görüntüde yüzler manuel olarak belirlenmiştir. Bu çalışmada kullanılan ağlar, girdi olarak 224x224 ve 227x227 kare görüntüler gerektirdiğinden, tespit edilen yüzler ağların giriş boyutlarına uyacak şekilde yeniden ölçeklendirilmiştir. Resim 7.1'in üst 2 satırında bulunan fotoğrafların tüm görüntü ön-işleme aşamalarından geçtikten sonraki halleri alt 2 satırda gösterilmiştir.

7.5 Deney Detayları

7.5.1 Veri Setindeki Hataların Giderilmesi

Eğitim ve test aşamalarına geçmeden önce, tüm veri seti incelenmiş ve görüntülerin meta verilerinde olası tutarsızlıklar taranmıştır. Metaverilerde bazı bariz hatalar fark edildiğinden, tüm veri seti için doğum tarihleri ve görüntü çekim tarihleri kontrol edilmiştir. Aynı kişiye ait bazı görüntüler için farklı doğum tarihleri olduğu gözlemlenmiştir. Daha spesifik olarak, 10.163 görüntünün güvenilir olmayan doğum tarihlerinin olduğu tespit edilmiştir. Veri setinden elde edilen yaş etiketlerini daha güvenilir hale getirmek için aşağıdaki düzeltme işlemleri uygulanmıştır. Aynı kişiye ait yüz görüntüleri birden fazla doğum tarihi ile ilişkilendirildiğinde ve bu doğum tarihlerinden biri diğerlerinden daha sık gözlemlendiğinde, bu kişinin tüm doğum tarihleri bu sık gözlemlenen doğum tarihi olarak kabul edilmiştir. Bu yöntemle 9.913 görüntünün doğum tarihleri düzeltilmiştir. İkinci olarak, aynı kişinin yüz görüntüleri birden fazla doğum tarihine sahip ve sık gözlemlenen bir doğum tarihi yoksa, herhangi bir değişiklik yapılmamıştır. Güvenilir olmayan doğum tarihine sahip görüntüler için, görüntülerin yaş etiketleri yeniden oluşturulmuş ve deneylerde yeni hesaplanan yaş etiketleri kullanılmıştır.

7.5.2 Deney Ortamı

Deneylerde kullanılan görüntü sayısını artırmak için görüntülerin yatay olarak çevrilmiş halleri üretilmiştir. Çevrilmiş versiyonların orijinal versiyonlarla birlikte kullanılmasıyla kullanılan toplam görüntü sayısı iki katına çıkarılarak 110.268 olmuştur. Deneylerde, görüntülerin %80'i ESA'ların eğitimi için kullanılırken, kalan %20'si test için kullanılmıştır. Veri setindeki görüntüler, sınıfların yaş sınırları ve sınıflardaki görüntü sayıları Çizelge 7.1'de gösterildiği gibi olacak şekilde 6 farklı yaş sınıfına ayrıl-

mıştır.

Çizelge 7.1: 6 yaş sınıfının sınırları ve sınıflarda bulunan görüntü sayıları.

16-20	21-27	28-34	35-41	42-48	49-
18.910	23.638	18.990	22.784	16.654	9.292

Görüntüler yaş sınıflarına göre ayrıldıktan sonra, eğitim ve test setleri oluşturulmuştur. Her sınıftaki görüntüler eğitim-test oranı korunarak rastgele bir şekilde eğitim ve test setlerine dağıtılmıştır. Görüntülerin dağıtımı sırasında, bir kişiye ait bir görüntü eğitim veya test setlerinden birine yerleştirildiğinde, aynı kişiye ait tüm görüntüler ve bu görüntülerin yatay olarak çevrilmiş halleri de aynı sete yerleştirilmiştir. Bu ilke ile, eğitim ve test setlerinin veri ayrıklığı korunmuştur.

ESA'ların eğitimi sırasında, [52] ile benzer şekilde ağırlık azalması (weight decay) ve momentum değerleri sırasıyla 0,0001 ve 0,9 olarak seçilmiştir. Öğrenme oranı 0,001'den başlatılmış ve öğrenme ilerlemesine göre azaltılmıştır. Eğitim süreçleri, test hatalarındaki azalmalar durağanlaşmaya kadar devam etmiştir. Bu durağanlaşmaların eğitimler sırasında 60 devir öncesinde gerçekleşmediği gözlemlenmiştir. NVIDIA Quadro K4000 192 bit GPU ve 3GB bellek ile, 6 katmanlı ağırlık eğitim süreci yaklaşık 1 gün sürmüştür. ResNet-18 ve ResNet-34'ün eğitim süreçleri 6 katmanlı ağırdan sırasıyla 8 kat ve 16 kat daha uzun sürmüştür. Deneylerde adil bir karşılaştırma yapmak için, tüm ağlara ait eğitim süreçlerinde aynı eğitim ve test setleri kullanılmıştır.

7.6 Ağ Başarılarının Kıyaslaması

Eğitilen ESA'ların performans başarısını karşılaştırmak için iki farklı kriter kullanılmıştır. İlk kriter, bir kişiye ait görüntünün o kişinin ait olduğu yaş sınıfı ile sınıflandırılıp sınıflandırılmadığını belirten tam başarı kriteridir. İkinci kriter ise 1-hata olarak tanımlanır ve bir giriş görüntüsüne için elde edilen en yüksek sınıflandırma puanına sahip sınıfın gerçek sınıf veya gerçek sınıfın komşularından biri olması durumunu kapsar. 1-hata başarı, özellikle bir kişinin yaşının iki yaş sınıfı arasındaki sınıra yakın olduğu durumlar için faydalıdır. Bu durumda, ağların giriş görüntüsünü komşularından biri olarak sınıflandırdığı için cezalandırmanın fazla olmaması mantıklıdır, çünkü genellikle bir kişinin yüzü deneylerde kullanılan bireysel yaş sınıflarından birden fazlasının özelliklerini yansıtabilir. Ayrıca, bir kişinin yaşı iki yaş sınıfının sınırına yakınsa, eğitilmiş bir ağın bu kişinin yaşını bu iki sınıftan biri olarak tahmin etmesi büyük bir hata olarak değerlendirilmemelidir. 1-hata başarısının gerekliliğini desteklemek için yaş kavramı incelenebilir. Aslında, yaş sürekli bir niceliktir ve sürekli değişkenleri tahmin etmek için regresyon yöntemleri kullanılır. Bu çalışmada sınıflandırma yöntemi kullanıldığından, sınıflandırma yönteminin bir regresyon problemine kıyasla yetersizliğini telafi etmek amacıyla 1-hata başarı kavramı kullanılmıştır.

Genel yaş sınıflandırması sonuçları sırasıyla Çizelge 7.2, Çizelge 7.3 ve Çizelge 7.4'te sunulmuştur. Çizelgelerdeki satırlar gerçek yaş sınıflarını, sütunlar ise test görüntülerinin tahmin edilen yaş sınıflarını göstermektedir. Koyu yeşil renkle renklendirilmiş köşegen hücreler, yaşları doğru tahmin edilen görüntülerin sayısını belirtmektedir. Her bir satır için, köşegen üzerinde bulunmayan hücreler yaşlarının yanlış tahmin edildiği

görüntülerin sayısını içerir ve bu hücrelerin sütunları yanlış tahmin edilen yaş sınıfını göstermektedir. Her satır için köşegen üzerinde bulunmayan hücrelerde, açık yeşil renkler 1-hata başarı oranını artıran komşu sınıf tahminlerinin sayısını göstermektedir.

Çizelge 7.2: 6-katmanlı ESA sınıflandırma sonuçları.

Yaş Sınıfları	Sınıflara göre Yaş Sınıfı Tahminleri					
	16-20	21-27	28-34	35-41	42-48	49-
16-20	1790	1730	74	6	0	0
21-27	405	3309	895	128	5	0
28-34	22	867	1596	990	87	2
35-41	3	183	1171	2719	653	37
42-48	1	31	176	1348	1409	337
49-	0	1	9	138	690	792

Çizelge 7.3: ResNet-18 sınıflandırma sonuçları.

Yaş Sınıfları	Sınıflara göre Yaş Sınıfı Tahminleri					
	16-20	21-27	28-34	35-41	42-48	49-
16-20	2497	1002	94	7	0	0
21-27	1135	2707	751	136	13	0
28-34	137	923	1333	1023	146	2
35-41	28	205	1042	2593	841	57
42-48	3	35	192	1206	1486	380
49-	1	2	16	172	700	739

Çizelge 7.4: ResNet-34 sınıflandırma sonuçları.

Yaş Sınıfları	Sınıflara göre Yaş Sınıfı Tahminleri					
	16-20	21-27	28-34	35-41	42-48	49-
16-20	2231	1217	123	28	1	0
21-27	923	2808	766	226	19	0
28-34	122	924	1216	1108	192	2
35-41	25	249	897	2568	958	69
42-48	4	30	215	1175	1498	380
49-	2	6	16	191	730	685

Çizelge 7.2, Çizelge 7.3 ve Çizelge 7.4’de gösterildiği gibi, her bir satır için köşegen üzerinde bulunan hücrelerden uzaklaştıkça hücrelerdeki sayıların 0’a yaklaştığı görülmektedir. Bu durum, tüm ağların 6 sınıflı yaş sınıflandırma problemi için uygun adaylar olduklarını kabaca göstermektedir. Çizelge 7.5’te, ağların tam ve 1-hata başarıları gösterilmektedir. Bu sonuçlara göre hem tam başarı hem de 1-hata başarımda, 6 katmanlı ağın daha derin olan artık ağlardan daha başarılı olduğu gözlemlenmektedir. Özellikle, 6 katmanlı ağ, 1-hata başarı kriteri için %95’in üzerinde ve tam başarı kriteri için %53’ün üzerinde başarıya sahiptir. Farklı yaş sınıflarındaki başarı oranları arasındaki standart sapma açısından bakıldığında, 6 katmanlı ağ en düşük değere sahiptir. Bu, 6 katmanlı ağın başarımının yaş sınıflarına daha az bağımlı olduğunu göstermektedir. ResNet-34, ResNet-18’den daha derin ve daha fazla sınıflandırma potansiyeline sahip olmasına rağmen, daha sıkı olan ağ her iki kriterde de daha başarılıdır.

Çizelge 7.5: Ağların tam ve 1-hata başarımları.

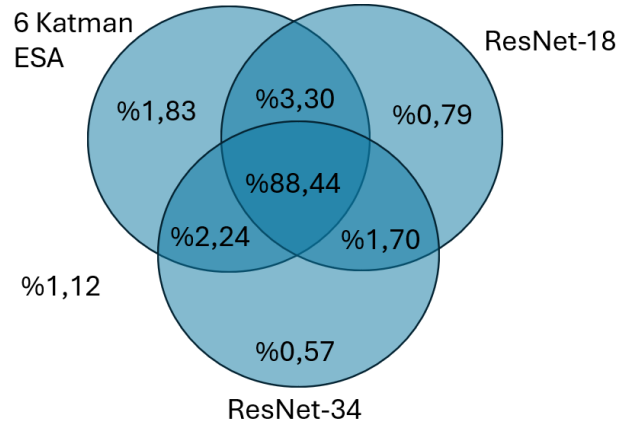
Yaş Sınıfları	6-Katman Başarım%		ResNet-18 Başarım%		ResNet-34 Başarım%	
	Tam	1-hata	Tam	1-hata	Tam	1-hata
16-20	49,72	97,78	69,36	97,19	61,97	95,78
21-27	69,78	97,20	57,09	96,86	59,22	94,83
28-34	44,78	96,89	37,40	92,00	34,12	91,13
35-41	57,05	95,32	54,41	93,92	53,88	92,80
42-48	42,67	93,70	45,00	93,03	45,37	92,46
49-	48,59	90,92	45,34	88,28	42,02	86,81
Ortalama	53,76	95,82	52,56	94,23	50,94	92,96
Standart Sapma	9,98	2,61	11,29	3,31	10,76	3,17

Ağların sınıflandırma performansları tartışılırken, ResNet [52] makalesinin yeniden irdelenmesinde fayda vardır. Bu makalede, ResNet versiyonlarının sınıflandırma gücü keşfedildikten sonra, yazarlar 10 sınıflı bir problem için 1202 katmanlı bir ResNet versiyonunu deneyerek ResNet prototipinin başarısını araştırmışlardır. Sonuç olarak, ResNet-1202'nin ResNet-110 kadar başarılı olmadığı ve ResNet-1202'nin gereksiz yere büyük olduğu ve daha sığ olanlara göre aşırı öğrenmeye daha yatkın olduğu sonucuna varırlar. Buradaki çalışmada da ResNet-18 ve ResNet-34'ün sınıflandırma performansları analiz edilirken benzer bir durum gözlemledik. ResNet-34, ResNet-18'e göre 6 sınıflı yaş sınıflandırma probleminde gereksiz yere derin kalmaktadır. 6 katmanlı ağın ResNet'lerden daha başarılı olduğu göz önüne alındığında, öğrenme sürecinde ResNet'lerin yüz görüntülerinden ilgisiz veriler öğrendiğini ve bunun da performans düşüşlerine neden olduğunu savunulabilir. ResNet [52] makalesinde de belirtildiği gibi, bırakma katmanlarının eksikliği, ResNet mimarilerinin başarısını azaltan nedenlerden biri olarak düşünülebilir.

Son olarak, üç mimari de kullanılarak 1-hata kriterine göre doğru sınıflandırılan görüntülerin oranları Şekil 7.5'de analiz edilmiştir. Görüldüğü üzere görüntülerin %88,44'ü tüm ağlar tarafından doğru sınıflandırılmıştır. Test görüntülerinin %1,12'sinde hiçbir ağ başarılı olamamıştır. Bunun yanı sıra, tüm ağlar, diğer ağların başarılı olamadığı bazı test görüntülerini sınıflandırma yeteneğine sahiptir. Özellikle, görüntülerin %1,83'ü yalnızca 6 katmanlı ağ tarafından doğru sınıflandırılırken, %0,79'u ResNet-18 ve %0,57'si ResNet-34 tarafından doğru sınıflandırılmıştır. Bu oranlar daha önceki sonuçlarla tutarlıdır ve 6 katmanlı ağın sunduğu görece gelişmiş sınıflandırma yeteneğini ortaya koymaktadır. Yaş sınıfı tahmini konusunda buraya kadar anlatılan çalışmalar uluslararası bir konferansta sözlü bildiri şeklinde sunulmuştur [61].

7.7 İdeal Ağ Mimarisinin Elde Edilmesi

Önceki bölümde bahsedildiği üzere 6-katmanlı ESA mimarisinin ResNet muadillerine göre 6 sınıflı yaş tahmini problemi için daha iyi sonuç verdiği görülmüştür. Bu aşamadan sonra 6 katmanlı mimariye benzer yapıda olan mimarilerden hangisinin daha iyi sonuç verdiğini irdelemek adına bir dizi deney gerçekleştirilmiştir. 6 katmanlı mimaride 3 ESA katmanı ve 3 tam bağlantılı katman bulunmaktadır. İdeal ağ mimarisini

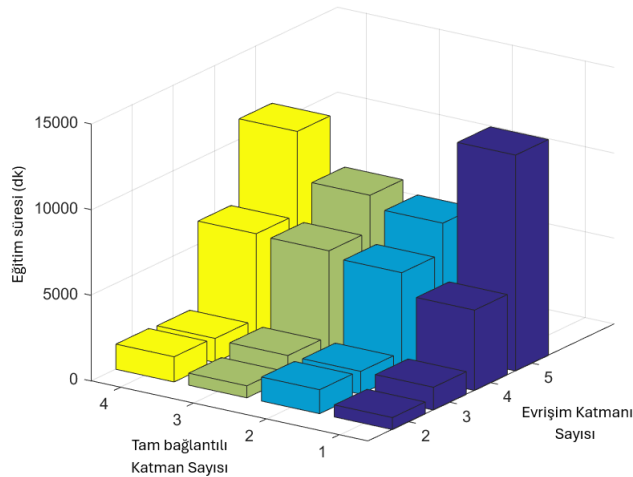


Şekil 7.5: 1-hata kriterine göre ağların başarılı olduğu görüntü oranları.

ararken mimari ESA katmanları ve tam bağlantılı katmanlar olarak 2 parça halinde parçalanmıştır. ESA katmanlarından oluşan ilk parça için Şekil 7.7’de gösterildiği gibi 2,3,4 ve 5 ESA katmanı içeren ağ parçaları varyantları oluşturulmuştur.

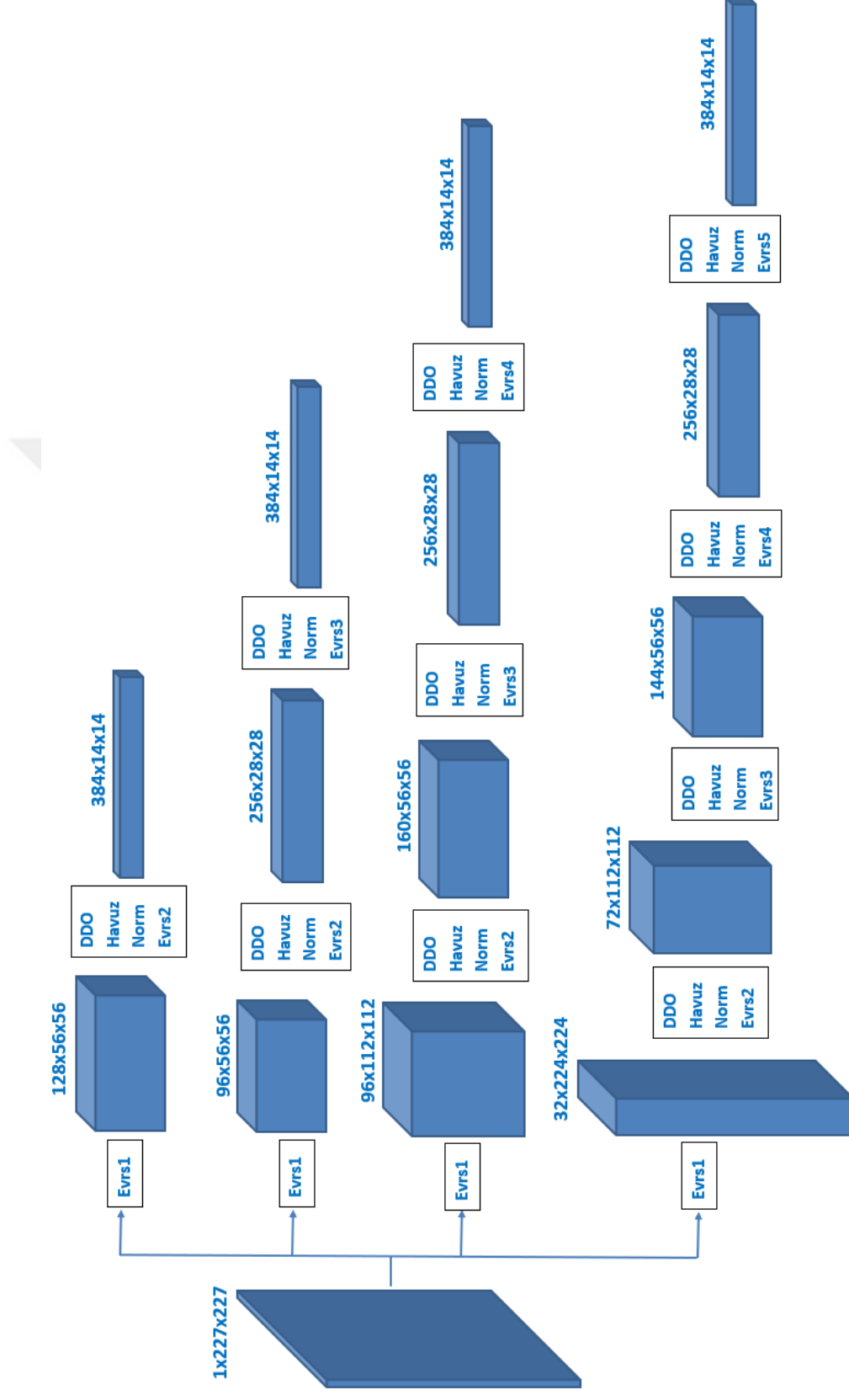
Bu 4 ağ parçasının sonunda oluşan aktivasyon haritasının boyutları aynıdır. ESA katmanlarından oluşan ağ parçası varyantlarına benzer şekilde tam bağlantılı katmanlardan oluşan ağ parçaları için de 4 farklı varyant oluşturulmuştur. Oluşturulan bu varyantlar Şekil 7.8’de gösterilmektedir.

Bu varyantların hepsi ESA katmanlarından oluşan varyantların son aktivasyon haritasını girdi olarak alabilecek yapıdadır. Böylelikle 4 farklı ESA katman parçası ile 4 farklı tam bağlantılı katman parçasının her biri birbirlerine bağlanarak 6 sınıflı yaş tahmini problemi için eğitilebilir yapıya sahip olmuşlardır. Toplamda 16 farklı ağ için deneyler yapılmıştır. Bu 16 farklı ağın eğitilme süreleri Şekil 7.6 gösterilmektedir.

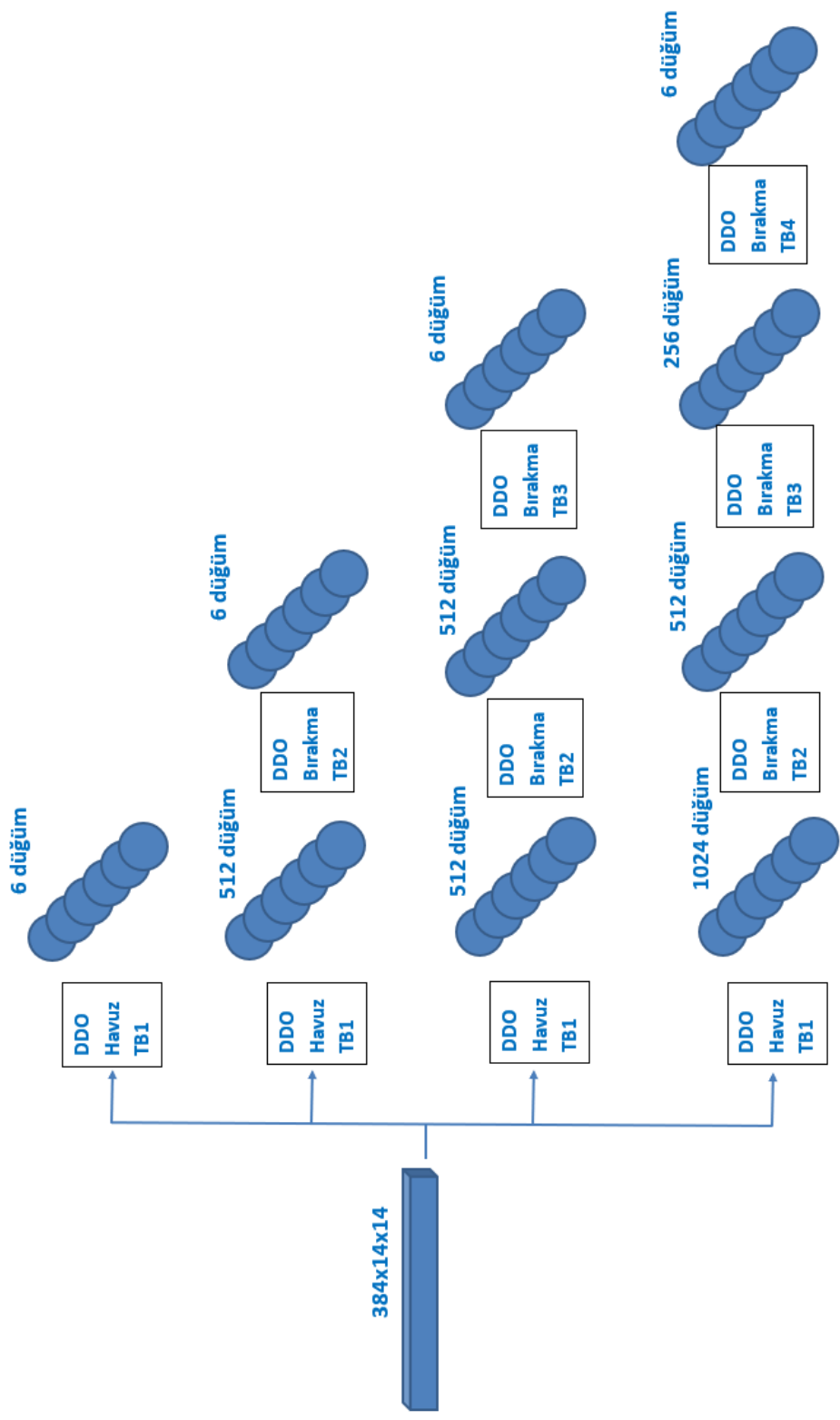


Şekil 7.6: 16 ağa ait eğitilme süreleri

Eğitim sürelerine göre, evrişim katmanlarının sayısı arttıkça eğitim süresinin de arttığı görülmektedir. Bu, beklentilerimizle tutarlıdır, çünkü evrişim katmanlarının eklenmesi



Şekil 7.7: 2, 3, 4 ve 5 ESA katmanından oluşan ağ parçaları



Şekil 7.8: 1, 2, 3 ve 4 tam bağlantılı katmandan oluşan ağ parçaları

işlenecek veri hacmini artırmaktadır. Tam bağlantılı katmanların sayısının artırılmasının her durumda eğitim süresini uzatmadığı gözlemlenmektedir. Veri hacimlerinin aksine, tam bağlantılı katmanlar mimarilere düğümler ekleyerek bellek erişim sürelerini çeşitlendirebilmektedir. Bu nedenle, daha az sayıda tam bağlantılı katmana sahip ağlar için daha uzun eğitim sürelerine sahip olmak mümkündür.

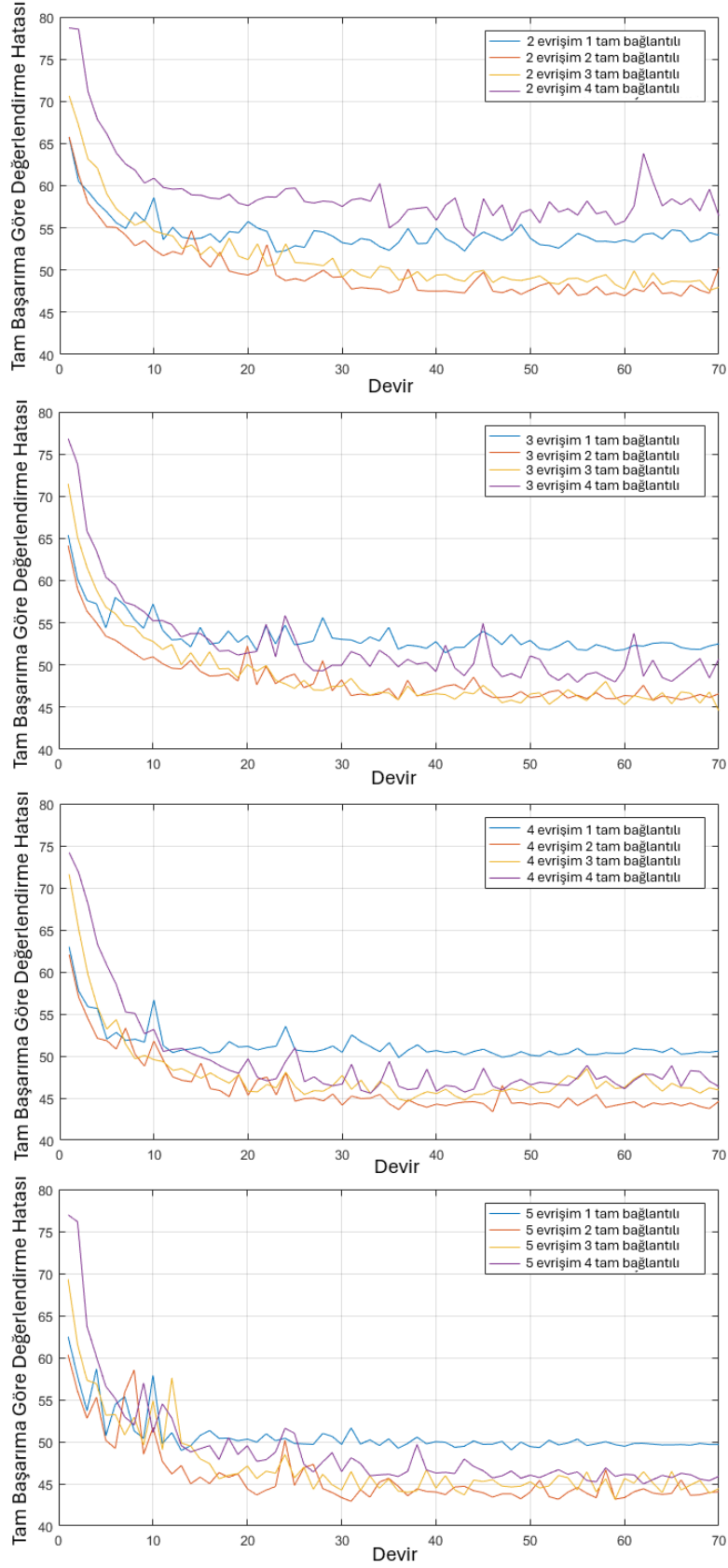
Çalışmanın bu aşamasında çalışmanın önceki aşamasında kullanılan tam ve 1-hata başarımlarının yanı sıra literatürde kullanılan ilk-3 başarımlarının da kullanılmaya başlanmıştır. İlk-3 başarımlarında yapay sinir ağının en yüksek skor verdiği 3 yaş sınıfının görüntünün gerçek yaş sınıfını içerip içermediği göz önüne alınır. Oluşturulan 16 yapay sinir ağına ait tam başarımlarına göre ve ilk-3 başarımlarına göre test sonuçları sırasıyla Şekil 7.9'da ve Şekil 7.10'da verilmiştir.

Bir tam bağlantılı katmana ve dört tam bağlantılı katmana sahip ESA'ların, iki veya üç tam bağlantılı katmana sahip ESA'lara göre daha düşük başarımlara ve daha düşük ilk-3 başarımlara sahip oldukları gözlemlenebilir. İki ve üç tam bağlantılı katmana sahip ESA'lar arasında, iki tam bağlantılı katmana sahip olanlar hem tam başarımlarına göre hem de ilk-3 başarımlarına göre daha az başarılıdır. Dolayısıyla, iki tam bağlantılı katmana sahip ESA'lar daha iyi performans gösterme eğilimindeyken, bir veya dört tam bağlantılı katmana sahip ESA'lar daha düşük performans gösterme eğilimindedir.

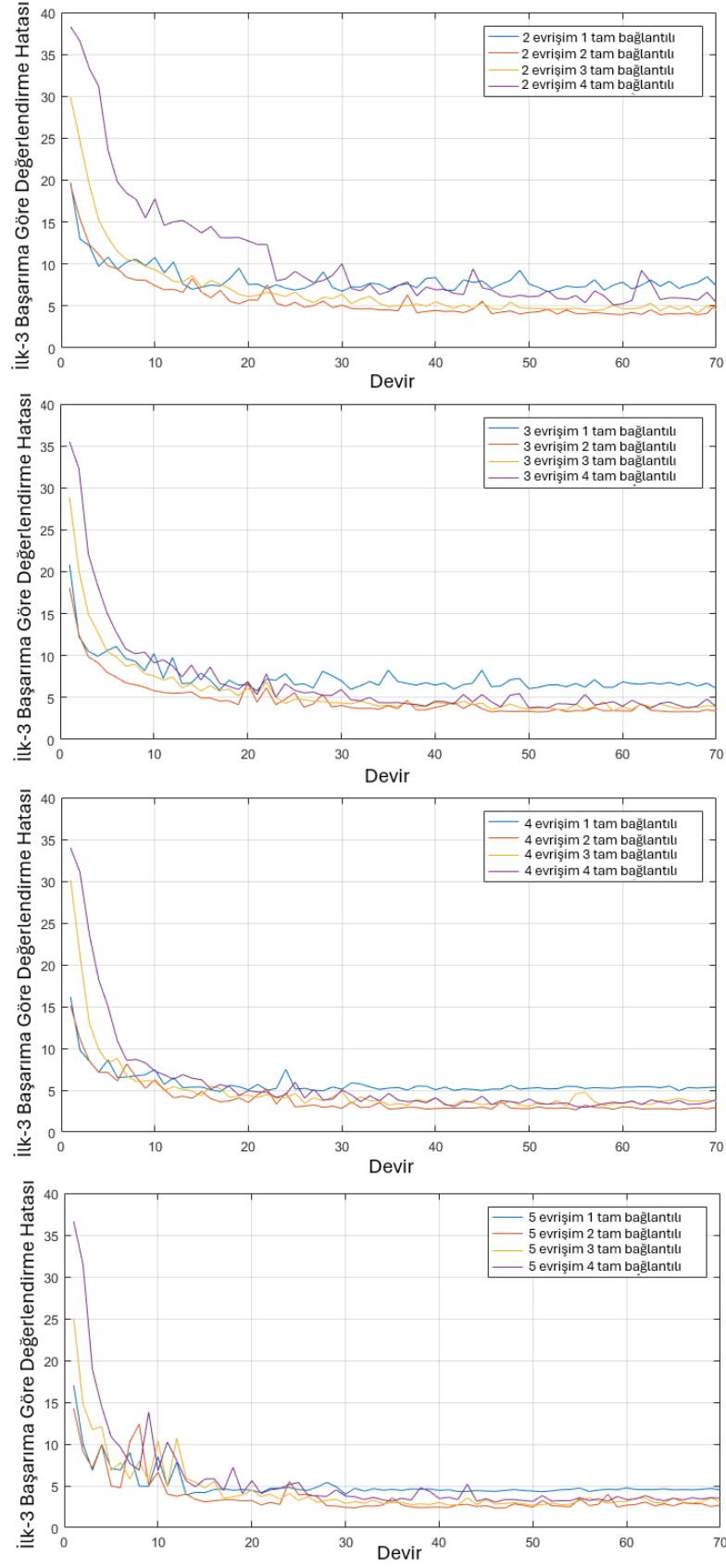
Şekil 7.9'da ve Şekil 7.10'da gösterilen doğrulama hata eğrileri, farklı sayıda evrişim katmanına sahip ESA'ların performansını karşılaştırmak için de kullanılmıştır. Kesin başarımlarına göre, 2 ve 3 evrişim katmanına sahip ESA'ların doğrulama hata oranları %45'ten düşük değildir. Öte yandan, 4 ve 5 evrişim katmanına sahip ESA'lar %45'ten daha düşük doğrulama hata oranlarına ulaşmaktadır. Ayrıca, 4 ve 5 evrişim katmanına sahip ESA'lar ile hem tam başarımlarına göre hem de ilk-3 başarımlarları için neredeyse aynı hata oranı elde edilmektedir. 4 ve 5 evrişim katmanına sahip ESA'lar benzer hata oranlarına sahip olduğundan ve Şekil 7.6'den görüldüğü gibi, daha derin olanlar daha uzun eğitim süreleri gerektirdiğinden, daha az derin ağı en uygun ağ seçiminde ön plana çıkarmayı tercih ediyoruz. Bu nedenle, hata eğrilerini inceleyerek, problemi için belirlediğimiz en uygun ESA, 4 evrişim ve 2 tam bağlantılı katmana sahip ESA'dır.

Genel olarak, iyi bir sınıflandırıcı, her farklı sınıf için benzer düzeyde sınıflandırma yeteneğine sahip olmalıdır. Benzer yaş sınıflandırma başarısına sahip ESA mimarileri arasında, yaş sınıflarına göre daha düşük standart sapma değerine sahip olan tercih edilmelidir. Şekil 7.11 ve Şekil 7.12, her ESA için tam ve 1-hata başarımların skorları kullanılarak hesaplanan standart sapmaları göstermektedir. Daha önce tartışılan öğrenme eğrilerini oluşturmak için yalnızca doğrulama görüntüleri kullanılırken, bu şekillerde gösterilen başarı ve standart sapma sonuçlarını oluşturmak için hem doğrulama hem de test görüntüleri kullanılmıştır. Eğitilmiş ESA versiyonlarını seçerken, tam ve ilk-3 başarımları için minimum doğrulama hatalarına sahip devir kaydı kullanılmıştır. Seçilen ESA versiyonları doğrulama ve test görüntüleri ile test edildikten sonra, tam ve 1-hata başarımları için elde edilen en başarılı test sonuçları ilgili şekillerde gösterilmektedir.

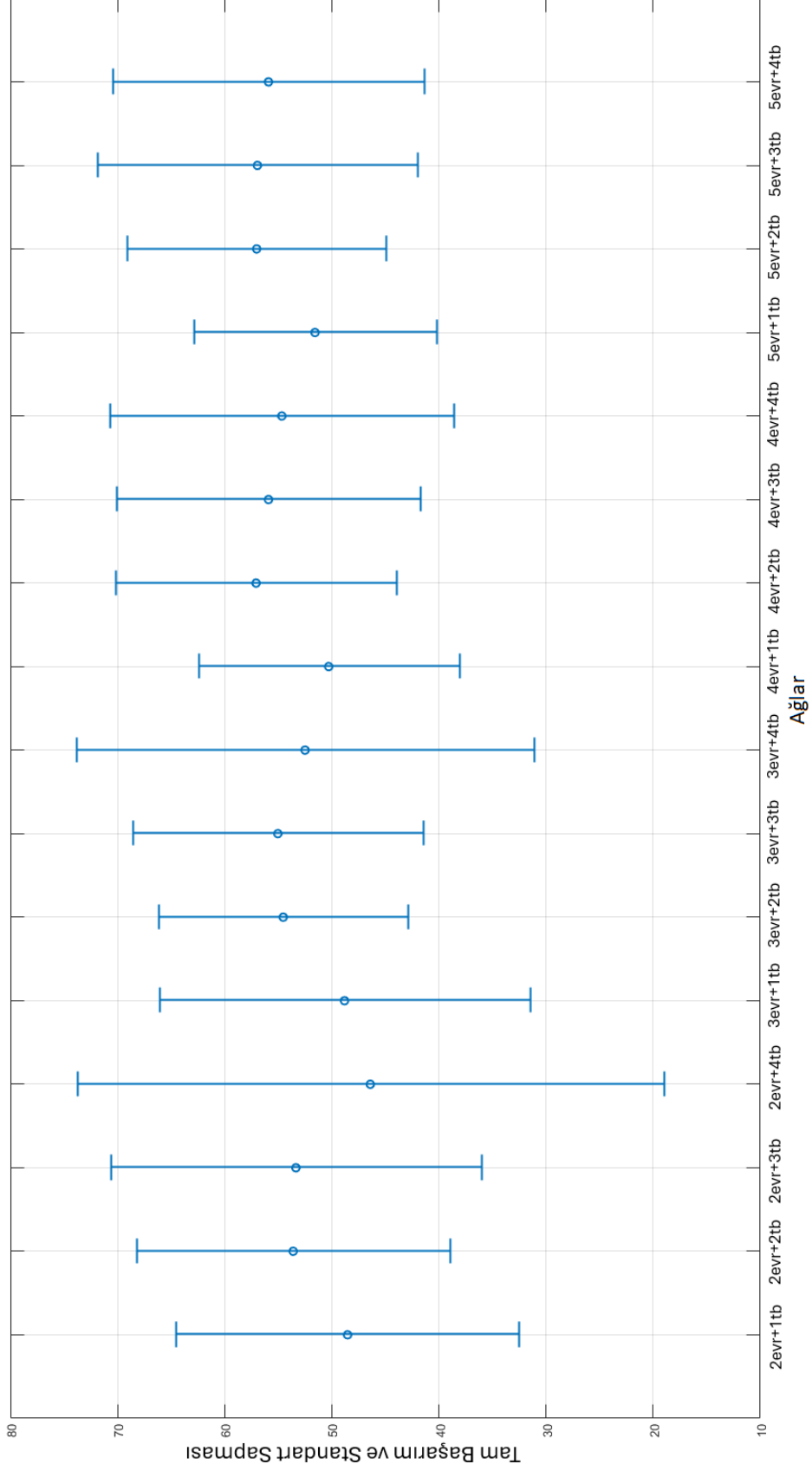
Şekil 7.11'ye yakından baktığımızda, 1 ve 4 tam bağlantılı katmana sahip ESA'lar için



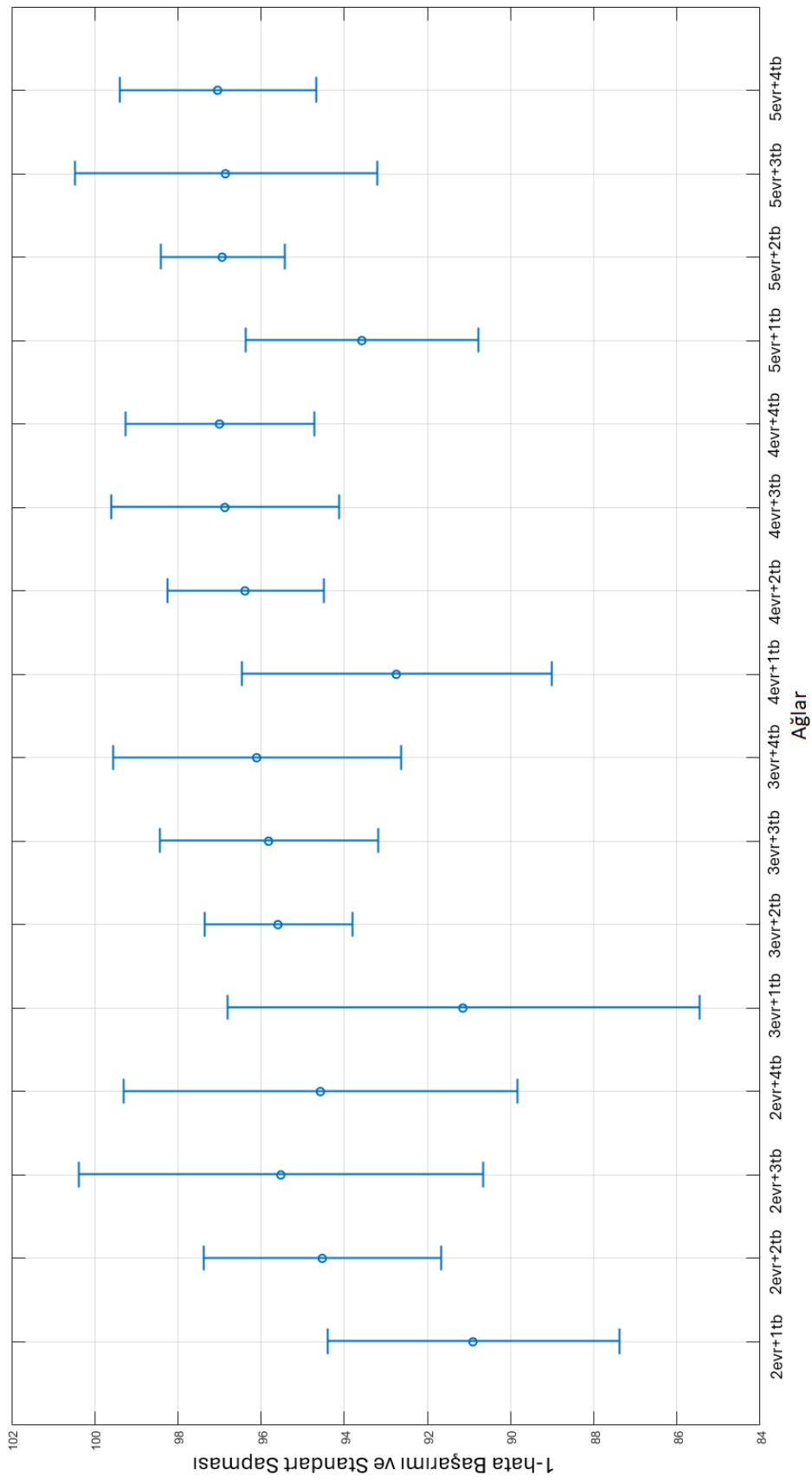
Şekil 7.9: 16 ağıta ait tam başarıma kriterine göre test sonuçları



Şekil 7.10: 16 ağa ait ilk-3 başarım kriterine göre test sonuçları



Şekil 7.11: 16 ağa ait tam başarımlarına göre standart sapmalar



Şekil 7.12: 16 ağa ait 1-hata başarımlarına göre standart sapmalar

daha önce tartışılan öğrenme eğrilerine benzer bir davranış gözlemliyoruz; yani, 2 ve 3 tam bağlantılı katmana sahip ESA'lar, 1 ve 4 tam bağlantılı katmana sahip ESA'lardan daha başarılıdır. Ayrıca, 2 ve 3 tam bağlantılı katmana sahip ESA'ların benzer kesin başarı oranlarına sahip olmalarına rağmen, 2 tam bağlantılı katmana sahip olanların daha düşük standart sapma değerlerine sahip olduğunu çıkarabiliriz. Bu nedenle, evrişim katmanlarının sayısından bağımsız olarak, kesin başarı açısından 2 tam bağlantılı katmana sahip ESA'lar, 3 tam bağlantılı katmana sahip olanlara göre tercih edilir. Ek olarak, 4 ve 5 evrişim katmanına sahip ESA'lar, daha az sayıda evrişim katmanına sahip olan benzerlerinden daha başarılıdır. 4 evrişim katmanına sahip ESA'ya bir evrişim katmanı daha eklemenin kesin başarıyı önemli ölçüde artırmadığı gözlemlenmektedir. Bu nedenle, tam başarı ve standart sapma değerlerini ve ağ eğitim sürelerini dikkate alarak, 4 evrişim katmanı ve 2 tam bağlantılı katmana sahip ESA tercih edilir.

Şekil 7.12'de gösterilen 1-hata başarı sonuçlarına göre, 1 tam bağlantılı katmana sahip ESA'nın performansı, aynı sayıda tam bağlantılı katmana fakat farklı sayıda evrişim katmanına sahip varyantlarından daha düşüktür. Yalnızca eşit sayıda evrişim katmanına ve 1'den fazla tam bağlantılı katmana sahip ESA'lar grup olarak incelendiğinde, tam bağlantılı katman sayısının 1-hata başarıyı önemli ölçüde etkilemediği gözlemlenmiştir. Bunun yerine, 1-hata başarı standart sapma, tam bağlantılı katman sayısına göre değişmektedir. Evrişim katmanlarının sayısının 2'den 5'e değiştiği tüm 4 durumda, 2 tam bağlantılı katmana sahip ESA'ların en düşük standart sapma değerlerine sahip olduğu görülmektedir. Bu nedenle, 1-hata başarı ve standart sapma açısından, 2 tam bağlantılı katmana sahip ESA'lar tercih edilir. Ayrıca, bu şekilden, 4 ve 5 evrişim katmanına sahip ESA'ların neredeyse eşit derecede başarılı oldukları ve 2 ve 3 evrişim katmanına sahip ESA'lardan daha iyi performans gösterdikleri de gözlemlenmektedir. 1-hata başarı ve standart sapma açısından, 5 evrişim ve 2 tam bağlantılı katmana sahip ESA, 4 evrişim ve 2 tam bağlantılı katmana sahip olandan daha iyi performans gösterse de kesin başarıya dayalı olarak aynı performansı sergilediği ve daha kısa eğitim süresine sahip olduğu için 4 evrişim ve 2 tam bağlantılı katmana sahip ESA'yı en uygun mimari olarak değerlendiriyoruz.

Şekil 7.11'de, 2 evrişim aşaması ve 4 tam bağlantılı aşaması olan ESA'nın sonuçları incelendiğinde, ağın kesin başarı açısından büyük bir standart sapmaya sahip olduğu gözlemlenmiştir. Bu durumu tartışmak amacıyla, ağın detaylı test sonuçları Çizelge 7.6 ve Çizelge 7.7'te gösterilmiştir. Sonuçlara göre, bu ağ ile [49-] sınıfındaki hiçbir görüntü doğru sınıflandırılmamış ve [28-34] sınıfındaki görüntülerin yalnızca %22,56'sı doğru sınıflandırılmıştır. Ayrıca, bu sınıflardaki görüntülerin çoğu, komşu sınıflardan biri olarak sınıflandırılmıştır. Şekil 7.11'de gösterildiği gibi, 4 tam bağlantılı katmana sahip ESA'larda evrişim katmanlarının sayısı arttıkça standart sapmaların azaldığı açıktır. Bu nedenle, evrişim katmanları açısından daha sık olan ESA'lara tam bağlantılı katman eklemenin, ağların bazı sınıflardaki görüntüleri komşu sınıflardan biri olarak sınıflandırma eğilimini artırarak tam başarı kriteri açısından standart sapmayı artırdığı sonucuna varılmıştır. İdeal ağ mimarisine ulaşmak için yapılan çalışmalar uluslararası bir konferansta sözlü bildiri olarak sunulmuştur [62].

Çizelge 7.6: 2 evrişim, 4 tam bağlantılı katmanlı ağı test sonuçları.

Yaş Sınıfları	Yaş Sınıflarına göre Test Sonuçları					
	16-20	21-27	28-34	35-41	42-48	49-
16-20	1547	2033	18	2	0	0
21-27	481	3826	310	115	10	0
28-34	37	1725	804	861	137	0
35-41	19	756	907	2111	973	0
42-48	8	154	278	1128	1734	0
49-	1	24	31	209	1365	0

Çizelge 7.7: 2 evrişim, 4 tam bağlantılı katmanlı ağı tam başarımı.

Yaş Sınıfları	% Tam Başarım
16-20	42,97
21-27	80,68
28-34	22,56
35-41	44,29
42-48	52,51
49-	00,00
Ortalama	46,39
Standart Sapma	27,35



8. SONUÇ VE ÖNERİLER

Bu tez kapsamında, göreceli kamera pozisyon tahmini probleminin çözümünde kullanılan ve varsayılan öznetelik eşleşmelerinden yanlış olanları ayıklamakla sorumlu ağların eğitilmesi için yeni bir $1-1$ çatı sunulmuştur. $1-1$ çatısı dahilinde ağları eğitebilmek için yapılan çalışmalar sırasında elde edilen deneyim paylaşılmıştır. Çatı içerisinde bağlam etkileşimini arttırmak için yerleştirilen BEB bloğunun gerçekleştirilen deneylerde fayda sağladığı gözlemlenmiştir. $1-1$ çatı üzerinde bağlam satırlarının sıralamasından bağımsız ve bağımlı ağların başarıyla eğitilebildikleri raporlanmıştır. $n-n$ ağların eğitiminde kullanılan *sın_kayıp*, *temel_kayıp* ve *geo_kayıp* fonksiyonlarının $1-1$ ağların eğitiminde de kullanılabildiği tecrübe edilmiştir. $1-1$ ağların eğitimi sırasında oluşan kayıp yüzeylerinin yüksek düzlemselliği sahip olması sebebiyle $n-n$ kullanılan ADAM optimizasyonu yerine $1-1$ ağlarda SGD optimizasyonun daha başarılı olduğu gözlemlenmiştir. Deneylerde kullanılan $1-1$ ağların derinliklerini daha fazla arttırmanın gradyan geri yayılımını engellediği tecrübe edilmiştir. Aynı derinliklere ve farklı kanal sayılarına sahip ağların eğitilmesiyle kanal sayılarının ağ başarımına etkisi sorgulanmıştır. ESA bloklarının çatı içerisinde işlem blokları olarak kullanılmasıyla literatürdeki ağların çoğunu geride bırakan veya görece benzer düzeyde başarıma erişen ağlar eğitilmiştir. Hem öne sürülen $1-1$ ağların eğitiminde hem de literatürdeki bilinen $n-n$ ağların eğitiminde Google Bulut TPU'ları kullanılmıştır. Tek bir TPUv4 cihazının $1-1$ çatısındaki tüm ağların teker teker eğitimi için yeterli donanım kapasitesine sahip olduğu görülmüştür. Tez kapsamında gerçekleştirilen deneylerde TPU cihazlarının kümelenilerek kullanılmasına ihtiyaç duyulmamıştır. Bu ve benzeri çalışmalarda geliştirilen ağların metaparametrelerini yeterince tarayabilmek için gerekli donanım kaynaklarının ancak bulut sistemleri vasıtasıyla sağlanabileceği gözlemlenmiştir.

$1-1$ ağlardan elde edilen performansı arttırmak amacıyla gelecekte çalışmalar yapılabilir. Basit ESA mimarileri kullanmak yerine, DenseNet [63], MobileNet [64–66], EfficientNet [67, 68] veya ConvNeXt [69, 70] gibi mimarilerden ağ elemanları, $1-1$ çatısındaki işleme bloklarına entegre edilebilir. Bağlam kanallarındaki satırlar, disparite açıları gibi farklı kriterlere göre sıralanabilir. Farklı yöntemlerle yapılandırılan bağlam kanalları bir arada kullanılarak giriş aktivasyon haritalarından elde edilebilecek veriler zenginleştirilebilir. $1-1$ ve $n-n$ 'lik çatıların eğitim ve çıkarım yapma zamanları kıyaslanabilir. Her iki çatı kullanılarak eğitilen ağlara ait modellerin parametrelerinde bit azaltması ile elde edilebilecek sonuçlar sorgulanabilir. $n-n$ ağlarda kullanılan ve başarımı arttıran model yapılarının $1-1$ ağlara uygulanabilirlikleri gözden geçirilebilir.

Tez kapsamında, bir durum çalışması olarak, insanların yüz görüntülerinden yaş sınıflandırması yapan derin öğrenme tabanlı ağlar geliştirilmiştir. Tezin asıl konusu üzerinde çalışmaya başlamadan önce yapılan bu çalışma, bilgisayarlı görü problemlerine yönelik derin öğrenme çözümlerinin geliştirilmesine aşinalık kazandırmıştır. Elde edi-

len sonuçlar, çalışmanın gerçekleştirildiği dönem için anlamlı olsa da, derin öğrenme sistemlerindeki gelişmelere uyum sağlamak adına güncellenmelidir. Yüz görüntülerinden yaş tahmini problemünde, güncel ve zengin veri setlerinin kullanımı ile derin modellerin, daha küçük modellere kıyasla sağlayacağı olası başarımların artışları ortaya konulabilir.



KAYNAKLAR

- [1] **Zhao, C., Ge, Y., Zhu, F., Zhao, R., Li, H., Salzmann, M.** Progressive correspondence pruning by consensus learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision (2021), pp. 6464–6473.
- [2] **Zhang, S., Ma, J.** Convmatch: Rethinking network design for two-view correspondence learning. In Proceedings of the AAAI Conference on Artificial Intelligence (2023), vol. 37, pp. 3472–3479.
- [3] **Zhang, S., Ma, J.** Convmatch: Rethinking network design for two-view correspondence learning. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023).
- [4] **Yi*, K. M., Trulls*, E., Ono, Y., Lepetit, V., Salzmann, M., Fua, P.** Learning to find good correspondences. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018).
- [5] **Zhang, J., Sun, D., Luo, Z., Yao, A., Zhou, L., Shen, T., Chen, Y., Quan, L., Liao, H.** Learning two-view correspondences and geometry using order-aware network. International Conference on Computer Vision (ICCV) (2019).
- [6] **Ricanek, K., Tesafaye, T.** Morph: A longitudinal image database of normal adult age-progression. In 7th International Conference on Automatic Face and Gesture Recognition (FGR06) (2006), Ieee, pp. 341–345.
- [7] **Lowe, D. G.** Distinctive image features from scale-invariant keypoints. International journal of computer vision 60, 2 (2004), 91–110.
- [8] **Yi, K. M., Trulls, E., Lepetit, V., Fua, P.** Lift: Learned invariant feature transform. In Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VI (Cham, Switzerland, 2016), Springer, pp. 467–483.
- [9] **Detone, D., Malisiewicz, T., Rabinovich, A.** Superpoint: Self-supervised interest point detection and description. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (2018), pp. 224–236.
- [10] **Fischler, M. A., Bolles, R. C.** Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 6 (1981), 381–395.

- [11] **Raguram, R., Chum, O., Pollefeys, M., Matas, J., Frahm, J. M.** Usac: A universal framework for random sample consensus. *IEEE transactions on pattern analysis and machine intelligence* 35, 8 (2012), 2022–2038.
- [12] **Longuet-Higgins, H. C.** A computer algorithm for reconstructing a scene from two projections. *Nature* 293, 5828 (1981), 133–135.
- [13] **Nistér, D.** An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence* 26, 6 (2004), 756–770.
- [14] **Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., Teh, Y. W.** Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning* (2019), Pmlr, pp. 3744–3753.
- [15] **Zaheer, M., Kottur, S., Ravanbakhsh, S., Póczos, B., Salakhutdinov, R. R., Smola, A. J.** Deep sets. *Advances in neural information processing systems* 30 (2017), 3391–3401.
- [16] **Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L.-J.** Yfcc100m: The new data in multimedia research. *Communications of the ACM* 59, 2 (2016), 64–73.
- [17] **Qi, C. R., Su, H., Mo, K., Guibas, L. J.** Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), pp. 652–660.
- [18] **Ionescu, C., Vantzos, O., Sminchisescu, C.** Matrix backpropagation for deep networks with structured layers. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 2965–2973.
- [19] **Ying, Z., You, J., Morris, C., Ren, X., Hamilton, W., Leskovec, J.** Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems* 31 (2018), 4805–4815.
- [20] **Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K. M., Trulls, E.** Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision* 129, 2 (2021), 517–547.
- [21] **Liu, Y., Liu, L., Lin, C., Dong, Z., Wang, W.** Learnable motion coherence for correspondence pruning. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2021), pp. 3237–3246.
- [22] **Dai, L., Liu, Y., Ma, J., Wei, L., Lai, T., Yang, C., Chen, R.** Ms2dg-net: Progressive correspondence learning via multiple sparse semantics dynamic graph. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition* (2022), pp. 8973–8982.
- [23] **Liu, X., Yang, J.** Progressive neighbor consistency mining for correspondence pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vi-*

sion and Pattern Recognition (2023), pp. 9527–9537.

- [24] **Hartley, R., Zisserman, A.** Multiple view geometry in computer vision. Cambridge university press, 2003.
- [25] **Ranftl, R., Koltun, V.** Deep fundamental matrix estimation. In Proceedings of the European conference on computer vision (ECCV) (2018), pp. 284–299.
- [26] **Krizhevsky, A., Sutskever, I., Hinton, G. E.** Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems 25 (2012), 1097–1105.
- [27] **Maas, A. L., Hannun, A. Y., Ng, A. Y., et al.** Rectifier nonlinearities improve neural network acoustic models. In Proc. icml (2013), vol. 30, Atlanta, Ga, p. 3.
- [28] **Kingma, D. P.** Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
- [29] **Bottou, L., et al.** Stochastic gradient learning in neural networks. Proceedings of Neuro-Nimes 91, 8 (1991), 12.
- [30] **Zhou, P., Feng, J., Ma, C., Xiong, C., Hoi, S. C. H., et al.** Towards theoretically understanding why sgd generalizes better than adam in deep learning. Advances in Neural Information Processing Systems 33 (2020), 21285–21296.
- [31] **Pytorch Team** PyTorch XLA: Accelerated Linear Algebra. <https://github.com/pytorch/xla>, 2024. Accessed: 2024-08-03.
- [32] **Vedaldi, A., Lenc, K., Ehrhardt, S., Gupta, A., Jaderberg, M.** Matconvnet. www.vlfeat.org/matconvnet, 2017. Convolutional neural networks for MATLAB. Accessed: 2024-08-23.
- [33] **Geng, X., Zhou, Z.-H., Smith-Miles, K.** Automatic age estimation based on facial aging patterns. IEEE Transactions on pattern analysis and machine intelligence 29, 12 (2007), 2234–2240.
- [34] **Burt, D. M., Perrett, D. I.** Perception of age in adult caucasian male faces: Computer graphic manipulation of shape and colour information. Proceedings of the Royal Society of London B: Biological Sciences 259, 1355 (1995), 137–143.
- [35] **Tiddeman, B., Burt, M., Perrett, D.** Prototyping and transforming facial textures for perception research. IEEE computer graphics and applications 21, 5 (2001), 42–50.
- [36] **O’Toole, A. J., Vetter, T., Volz, H., Salter, E. M.** Three-dimensional caricatures of human heads: distinctiveness and the perception of facial age. Perception 26, 6 (1997), 719–732.
- [37] **Hutton, T. J., Buxton, B. F., Hammond, P., Potts, H. W.** Estimating average

growth trajectories in shape-space using kernel smoothing. *IEEE transactions on medical imaging* 22, 6 (2003), 747–753.

- [38] **Lanitis, A., Draganova, C., Christodoulou, C.** Comparing different classifiers for automatic age estimation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 34, 1 (2004), 621–628.
- [39] **Ricanek, K., Wang, Y., Chen, C., Simmons, S. J.** Generalized multi-ethnic face age-estimation. In *Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on (2009)*, Ieee, pp. 1–6.
- [40] **Luu, K., Ricanek, K., Bui, T. D., Suen, C. Y.** Age estimation using active appearance models and support vector machine regression. In *Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on (2009)*, Ieee, pp. 1–5.
- [41] **Luu, K., Bui, T., D., Suen, C. Y., Ricanek, K.** Spectral regression based age determination. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops (2010)*, Ieee, pp. 103–107.
- [42] **Sethuram, A., Ricanek, K., Patterson, E.** A hierarchical approach to facial aging. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops (2010)*, Ieee, pp. 100–107.
- [43] **Han, H., Otto, C., Jain, A. K.** Age estimation from face images: Human vs. machine performance. In *2013 International Conference on Biometrics (ICB) (2013)*, Ieee, pp. 1–8.
- [44] **Zeiler, M. D., Fergus, R.** Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (2014)*, Springer, pp. 818–833.
- [45] **Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., Lecun, Y.** Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229* (2013).
- [46] **Girshick, R., Donahue, J., Darrell, T., Malik, J.** Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (2014)*, pp. 580–587.
- [47] **Ren, S., He, K., Girshick, R., Sun, J.** Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (2015)*, pp. 91–99.
- [48] **Levi, G., Hassner, T.** Age and gender classification using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2015)*, pp. 34–42.
- [49] **Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan,**

- D., Vanhoucke, V., Rabinovich, A.** Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (2015), pp. 1–9.
- [50] **He, K., Zhang, X., Ren, S., Sun, J.** Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International Conference on Computer Vision (2015), pp. 1026–1034.
- [51] **Ioffe, S., Szegedy, C.** Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015).
- [52] **He, K., Zhang, X., Ren, S., Sun, J.** Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015).
- [53] **Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.** Imagenet large scale visual recognition challenge. International journal of computer vision 115, 3 (2015), 211–252.
- [54] **He, K., Sun, J.** Convolutional neural networks at constrained time cost. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015), pp. 5353–5360.
- [55] **Srivastava, R. K., Greff, K., Schmidhuber, J.** Highway networks. arXiv preprint arXiv:1505.00387 (2015).
- [56] **Viola, P., Jones, M.** Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (2001), vol. 1, Ieee, pp. I–511.
- [57] **Yu, S.** Shiqi Yu’s Homepage, 2016.
- [58] **Castrillón, M., Déniz, O., Guerra, C., Hernández, M.** Encara2: Real-time detection of multiple faces at different resolutions in video streams. Journal of Visual Communication and Image Representation 18, 2 (2007), 130–140.
- [59] **Ojala, T., Pietikainen, M., Maenpaa, T.** Multiresolution grayscale and rotation invariant texture classification with local binary patterns. IEEE Transactions on pattern analysis and machine intelligence 24, 7 (2002), 971–987.
- [60] **Lienhart, R., Kuranov, A., Pisarevsky, V.** Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In Joint Pattern Recognition Symposium (2003), Springer, pp. 297–304.
- [61] **Aydogdu, M. F., Celik, V., Demirci, M. F.** Comparison of three different cnn architectures for age classification. In 2017 IEEE 11th international confe-

rence on semantic computing (ICSC) (2017), Ieee, pp. 372–377.

- [62] **Aydogdu, M. F., Demirci, M. F.** Age classification using an optimized cnn architecture. In Proceedings of the International Conference on Compute and Data Analysis (2017), pp. 233–239.
- [63] **Huang, G., Liu, Z., VAN DER Maaten, L., Weinberger, K. Q.** Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (2017), pp. 4700–4708.
- [64] **Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.** Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017).
- [65] **Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L. C.** Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (2018), pp. 4510–4520.
- [66] **Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al.** Searching for mobilenetv3. In Proceedings of the IEEE/CVF international conference on computer vision (2019), pp. 1314–1324.
- [67] **Tan, M., Le, Q.** Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning (2019), Pmlr, pp. 6105–6114.
- [68] **Tan, M., Le, Q.** Efficientnetv2: Smaller models and faster training. In International Conference on Machine Learning (2021), Pmlr, pp. 10096–10106.
- [69] **Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., Xie, S.** A convnet for the 2020s. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2022), pp. 11976–11986.
- [70] **Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I. S., Xie, S.** Convnext v2: Co-designing and scaling convnets with masked auto-encoders. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023), pp. 16133–16142.

ÖZGEÇMİŞ



