Journal of the Faculty of Engineering and Architecture of Gazi University 36:3 (2021) 1715-1731

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Detection of bee diseases with a hybrid deep learning method

Sedat Metlek[1]*, Kıyas Kayaalp[2]

[1]Burdur Mehmet Akif Ersoy University, Vocational School of Technical Sciences, Electronics and Automation Department, 15100, Burdur, Turkey
[2]Isparta University of Applied Sciences, Faculty of Technology, Computer Engineering, 32080, Isparta, Turkey

Highlights:

- A new data set was created by editing an existing data set for the detection of bee diseases.
- A new hybrid method has been developed for the detection of bee diseases.
- A different perspective has been developed by using deep learning methods in detecting bee diseases

Keywords:

- Bee Diseases
- Deep Learning
- Convolution Neural Networks
- Classification

Article Info:

Received: 08.06.2020
Accepted: 09.03.2021

DOI:

10.17341/gazimmfd.749443

Correspondence:

Author: Sedat Metlek
e-mail: sedatmetlek@mehmetakif.edu.tr
phone: +90 248 213 4580

Graphical/Tabular Abstract

Bees are one of the oldest living species in the world, having a major impact on the development of living species. The continuity of plants at the bottom of the food chain is directly related to the pollination of bees. Bees are a global insurance because of this characteristic. For this reason, it is very important to check the health status of bees. Depending on the technology developed nowadays, it is possible to control the health status of bees remotely with real-time image processing applications. In the study, the feature extraction feature, which is one of the strengths of deep learning, was applied from two different arms as shown in Figure A, and aggressive changes in the images were detected. In the classification process; Instead of Softmax classifier based on probability calculation, multi-layer feedback artificial neural network (MLFB-ANN) model has been used. The success of the designed system has also been compared with the Softmax classifier. As a result of experimental studies, 93.07% success rate can be achieved with softmax classifier for six different bee diseases on the same data set, while 95.04% success rate has been obtained with the developed system.

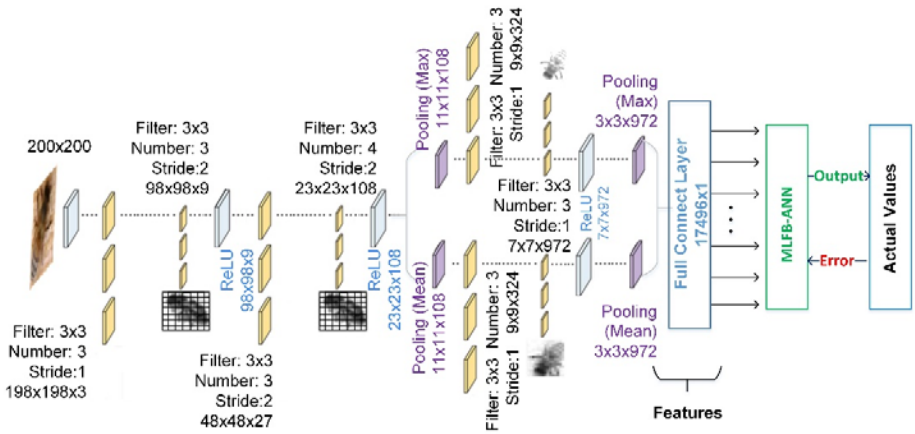


Figure A. System Architecture

Purpose: In this study, it is aimed is to demonstrate that the health status of bees can be detected by artificial intelligence by developing a different perspective in the diagnosis of six different bee diseases and to develop a hybrid deep learning algorithm for this diagnosis.

Theory and Methods: In the study, feature extraction methods, which are the strengths of deep learning, were operated from two different arms and aggressive changes in images were detected. In the classification process; Instead of Softmax classifier based on probability calculation, multi-layer feedback artificial neural network (MLFB-ANN) model has been used.

Results: The success of the designed system has also been compared with the Softmax classifier. As a result of experimental studies, 93.07% success rate can be achieved with Softmax classifier for six different bee diseases on the same data set, while 95.04% success rate has been obtained with the developed system.

Conclusion: In this study, a hybrid method based on deep learning methods was proposed for the classification of bee diseases and successful results were obtained.

Journal of the Faculty of Engineering and Architecture of Gazi University 36:3 (2021) 1715-1731

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University
Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Arı hastalıklarının hibrit bir derin öğrenme yöntemi ile tespiti

Sedat Metlek[1]*, Kıyas Kayaalp[2]

[1]Burdur Mehmet Akif Ersoy Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Elektronik ve Otomasyon Bölümü, 15100, Burdur, Türkiye
[2]Isparta Uygulamalı Bilimler Üniversitesi, Teknoloji Fakültesi, Bilgisayar Mühendisliği, 32080, Isparta, Türkiye

Ö N E Ç I K A N L A R

- Arı hastalıklarının tespiti için var olan bir veri seti düzenlenerek yeni bir veri seti oluşturulmuştur
- Arı hastalıklarının tespiti için yeni hibrit bir yöntem geliştirilmiştir
- Arı hastalıklarının tespit edilebilmesinde derin öğrenme yöntemleri kullanılarak farklı bir bakış açısı geliştirilmiştir

Makale Bilgileri

Araştırma Makalesi
Geliş: 08.06.2020
Kabul: 09.03.2021

DOI:
10.17341/gazimmfd.749443

Anahtar Kelimeler:
Arı hastalıkları,
derin öğrenme,
evrişimsel sinir ağları,
yapay sinir ağı,
sınıflandırma

ÖZET

Canlı türlerinin gelişiminde büyük bir etkisi olan arılar Dünya'da ki en eski canlı türlerinden birisidir. Besin zincirinin en altında bulunan bitkilerin devamlılığı arıların tozlaşma yapmasıyla doğrudan ilgilidir. Arılar bu özelliğinden dolayı küresel bir sigorta konumundadır. Bu nedenle arıların sağlık durumlarının kontrol edilmesi oldukça önemlidir. Günümüzde gelişen teknolojiye bağlı olarak, arıların sağlık durumlarının uzaktan gerçek zamanlı görüntü işleme uygulamaları ile kontrol edilebilmesi mümkün olabilmektedir. Gerçekleştirilen çalışmada derin öğrenmenin güçlü yanlarından olan öznitelik çıkarma yöntemleri iki farklı koldan işletilerek, görüntülerdeki agresif değişiklikler tespit edilmiştir. Sınıflandırma işleminde, olasılık hesabına dayanan ve sınıf sayısı kadar çıkış veren Softmax sınıflandırıcısı ile tek bir çıkış verebilen ve bu çıkışta da sınıf bilgisini sunabilen çok katmanlı geri beslemeli yapay sinir ağı (ÇKGB-YSA) kullanılmıştır. Yapılan deneysel çalışmalar neticesinde, aynı veri seti üzerinde altı farklı arı hastalığı için Softmax sınıflandırıcısı ile %93,07 başarım oranı yakalanabilirken, geliştirilen sistem ile %95,04 başarım oranı elde edilmiştir. Bu çalışmada arı hastalıklarının sınıflandırılması için derin öğrenme yöntemlerine dayalı hibrit bir yöntem önerilmiş ve başarılı sonuçlar elde edilmiştir.

Detection of bee diseases with a hybrid deep learning method

H I G H L I G H T S

- A new data set was created by editing an existing data set for the detection of bee diseases
- A new hybrid method has been developed for the detection of bee diseases
- A different perspective has been developed by using deep learning methods in detecting bee diseases

Article Info

Research Article
Received: 08.06.2020
Accepted: 09.03.2021

DOI:
10.17341/gazimmfd.749443

Keywords:
Bee diseases,
deep learning,
convolution neural networks,
artifical neural network,
classification

ABSTRACT

Bees are one of the oldest living species in the world, having a major impact on the development of living species. The continuity of plants at the bottom of the food chain is directly related to the pollination of bees. Bees are a global insurance because of this characteristic. For this reason, it is very important to check the health status of bees. Depending on the technology developed nowadays, it is possible to control the health status of bees remotely with real-time image processing applications. In the study, feature extraction methods, which are the strengths of deep learning, were executed from two different arms and aggressive changes in images were detected. In the classification process; Instead of Softmax classifier based on probability calculation, multi-layer feedback artificial neural network (MLFB-ANN) model has been used. The success of the designed system has also been compared with the Softmax classifier. As a result of experimental studies, 93.07% success rate can be achieved with Softmax classifier for six different bee diseases on the same data set, while 95.04% success rate has been obtained with the developed system. In this study, a hybrid method based on deep learning methods was proposed for the classification of bee diseases and successful results were obtained.

*Sorumlu Yazar / Yazarlar / Corresponding Author / Authors : *sedatmetlek@mehmetakif.edu.tr, kiyaskayaalp@isparta.edu.tr / Tel: +90 248 213 4580

1. GİRİŞ (INTRODUCTION)

Yaklaşık 20.000 farklı arı türünden biri olan bal arıları, dünyadaki ekolojik dengeyi sağlayan ve koloniler halinde yaşayan en önemli canlı türlerinden birisidir. Yapılan araştırmalara göre varlığı 100 Milyon yıl öncesine dayanan arıların, en küçüğü 2 mm, en büyüğü 4 cm boyundadır [1]. Arıların Dünya'da ki en eski canlı türlerinden birisi olması nedeniyle, dolaylı olarak canlı türlerinin gelişimine de büyük bir etkisi bulunmaktadır. Örneğin besin zincirinin en üstünde bulunan insan ve etçil hayvanlar, protein kaynağı ihtiyacını otçullardan ve bitkilerden karşılamaktadır. Otçullar ise besin ihtiyacını doğrudan bitkilerden karşılamaktadır. Besin zincirinin en altında bulunan bitkiler ise devamlılığını arıların tozlaşma yapmasıyla sağlayabilmektedir [2]. Arıların besin zincirindeki yerinden çıkması durumunda, her yıl bitki çeşitliliğinin %50 azalacağı ön görülmektedir. Bu ön görünün sonucunda dört yıl gibi kısa bir süre zarfında yer yüzünde bulunan bitkisel kaynakların %93,75'inin tükeneceği ön görülmektedir [3]. Bitkilerin azalmasıyla doğru orantılı olarak yağış sistemleri değişecek ve bunun sonucunda da iklimler değişerek, su sorunlarını ortaya çıkaracaktır. Bu yaşam döngüsünü etkileyen su, iklim, beslenme gibi parametrelerin değişmesi durumunda, besin zinciri kırılacağı için yer yüzünde bulunan yaşam döngüsü değişecektir. Bu değişimin sonucu ilk etapta hayvanlar üzerinde görülecek ve çok kısa bir süre içerisinde de insanların sosyo ekonomik ve kültürel yaşantısını doğrudan etkileyecektir. Toplumların sosyo-ekonomik yapısının bozulması sonucunda da beslenme eksikliğinden kaynaklanan hastalıklar, yeni ve güvenilir beslenme habitatlarına doğru göçleri artıracaktır. Bunların sonucunda da sınır problemleri ortaya çıkacak ve toplumlar arasında güvenlik problemleri yaşanacaktır. Bütün bu durumlardan anlaşılacağı üzere arıların toplumlar için küresel bir sigorta olduğu görülmektedir [4].

Yukarıda bahsedildiği üzere bal arıları (Apis mellifera), hem doğal ortamda hem de insanoğlunun yetiştirdiği bitkileri tozlaştırarak, gıda üretimi ile biyolojik çeşitliliğin sürdürülmesine ve arttırılmasına doğrudan katkıda bulunmaktadır. Günümüzde son derece önemli olan bu canlı türünün geleceği, dünya çapında ki birçok hastalık ve farklı nedenlerden ötürü tehlike altındadır. Arı popülasyonlarının azalmasına, patojenler (parazitler, mantarlar, bakteriler, virüsler, vb.), akarisitler, fungisitler, herbisitler ve zirai kimyasal ilaç kullanımı gibi çeşitli faktörler de neden olmaktadır [5-7]. Ayrıca arıcılar bal üretimi esnasında farklı zamanlarda farklı lokasyonlara arı kovanlarını taşımaktadırlar. Bunun sonucunda da bir bölgede olan hastalığı, diğer bölgelerdeki arı kolonilerine de bulaştırma ihtimalleri olmaktadır. Tüm bu olumsuzluklara rağmen arıların bağışıklık sistemi de bu değişimlere karşı kendini adapte etmeye çalışmaktadır. Bunların dışında genel olarak arıcılık, dünya çapında arıların ve kovan ürünlerinin sürekli hareket ettiği küresel bir endüstridir. Bu nedenle, bu hastalıkların yayılmasındaki ana mekanizmaları tanımlamak, nihayetinde ortadan kaldırmak ve dünya çapında güvenli ticareti sağlamak için bal arısı hastalıkları üzerine uluslararası bir bakış açısı gereklidir [8, 9]. Arıcılık küresel endüstrisini, güvenli hale getirebilmek adına günümüz teknolojisinin gelmiş olduğu en son yazılım ve donanım imkanlarından yararlanmakta fayda vardır. Özellikle donanım sektöründe yaşanan gelişmelerin sonucunda, yazılım sektöründe çok hızlı gelişmeler görülmektedir. Son 10 yıla kadar yüksek maliyeti nedeniyle birçok sektörde kullanılamayan kameralar, günümüzde çok daha ucuza ve daha gelişmiş özellikleri ile yaygın olarak kullanılabilmektedir. Kameraların bu şekilde kolay tedarik edilebilmesi sonucunda endüstride birçok alanda ve farklı uygulamalarda kullanılmaya başlanmıştır [10-12]. Aynı şekilde bu kameraların arıcılık sektöründe de önümüzdeki yıllarda yaygın olarak kullanılmaya başlanacağı ön görülmektedir. Kamera kullanımının artmasının bir diğer nedenide geliştirilen yazılımlardır. Özellikle yapay zeka alanında yaşanan gelişmeler birçok alanda uygulanabilir sonuçlar üretilmesine vesile olmuştur. Son yıllarda görüntüler üzerinden nesne tespiti ve sınıflandırma işlemleri için yapay zekanın bir alt konusu olan derin öğrenme yöntemleride kullanılmaya başlanmıştır [13]. Donanım mimarisinindeki gelişmeler sonucunda, derin öğrenme mimarisi ile çalışan sistemlerde yüksek doğruluk oranlarının elde edilmesi [14, 15] sayesinde, derin öğrenme algoritmalarının hastalık tespitinde kullanıldığı görülmektedir [16-18]. Geliştirilecek kameralı bir otomasyon sistemi ile arı hastalıkları tespit edilip, hasta arının kovana girişi engellenerek kolonideki diğer arılara hastalığın bulaştırılması engellenebilir veya hasta olan arıların hastalıklarının tespit edilerek, iyileştirilmesi için uygun ilaç kullanımı sağlanabilir. Burada esas olan konu, geliştirilecek yazılım ve arı hastalıklarının tespit edilerek sınıflandırılmasıdır. Bu sınıflandırma işlemi birçok algoritma ile yapılabilecek olmasına rağmen, sisteme gelecek farklı arı türlerinin ve farklı hastalıkların olması tasarımı kısıtlayan en önemli etmenlerdir.

Bu çalışmada küresel bir sigorta olan bal arılarının, hastalıklarını tespit edebilecek bir yazılım geliştirilmiştir. Tasarlanan sistemde derin öğrenme yöntemlerinin görüntü işleme alanındaki uygulanabilirliği göz önüne alınarak, derin öğrenme yöntemleri ve görüntü işleme algoritmaları birlikte kullanılmıştır. Gerçekleştirilen çalışmada farklı hastalık türlerini de ayırt edebilecek hibrit bir derin öğrenme yöntemi geliştirilmiş ve test edilmiştir. Çalışmada veri seti olarak "The BeeImage Dataset: Annotated Honey Bee Images" veri seti kullanılmıştır. 5.172 arı görüntüsünü içeren bu veri setinde, her bir arı görüntüsüne ait dosya ismi, tarih, saat, yer, bölge kodu, alt türler, hastalık, polen taşıma durumu ve sınıf (işçi, kraliçe) bilgileri bulunmaktadır. Bu görüntüler arı video görüntülerinden, kesitler alınarak elde edilmiştir. Arıları ön plana çıkarmak için görüntülerden arka plan çıkarılmıştır. Resim üzerindeki arı görüntüleri çerçeve dolaştırma yöntemi ile tek tek elde edilmiş ve diğer bilgilerle birlikte veri tabanına kayıt edilmiştir. Kullanılan bu veri seti üzerinde derin öğrenme ve çok katmanlı geri beslemeli yapay sinir ağı hibrit olarak kullanılmıştır.

Bu çalışmanın katkısı;

- Derin öğrenme yöntemleri ile arı hastalıklarının tespitine yönelik farklı bir bakış açısı öne sürülmüştür. Bu bakış açısı doğrultusunda da bir uygulama geliştirilmiştir.
- Kullanılan veri setindeki anlamsız görüntüler çıkarılarak ve geri kalanları da standart hale getirilerek yeni bir veri seti oluşturulmuştur.
- Oluşturulan veri seti üzerinde yüksek performans değerlerine sahip hibrit bir yöntem geliştirilmiştir.

Gerçekleştirilen çalışmanın genel sunum şekli dört bölümden oluşmaktadır. Birinci bölümde arı hastalıkları ve konu ile ilgili yapılan literatür çalışmaları anlatılmıştır. İkinci bölümde kullanılan veri seti ve önerilen yöntemde kullanılan mimariler anlatılmıştır. Üçüncü bölümde önerilen yöntem detaylı olarak sunulmuştur. Son bölümde ise elde edilen sonuçlar aktarılmıştır.

1.1. Arı Hastalıkları (Bee Illness)

Dünyada birçok arı hastalığı (kovan hırsızlığı, varroa, küçük kovan böceği, karınca problemi, kraliçe kaybı, vb.) bulunmakla beraber, bazı arı hastalıkları da bölgesel (Avrupa yavru çürüklüğü, Amerika yavru çürüklüğü, vb.) olabilmektedir [19]. Ancak arıların bal üretimi veya bitki üretimini desteklemek için arı kolonileri farklı zamanlarda farklı bölgelere taşınabilmektedir. Bu durumda da bir bölgede görülen arı hastalığı, diğer bölgelerede taşınabilmektedir. Bu nedenle birçok arı hastalığı farklı bölgelerde de görülebilmektedir. Bunun sonucunda çalışmada sadece Şekil 1'de gösterilen ve veri setinde belirtilen, doğal ortamda en fazla görülebilen arı hastalıkları ile ilgili temel bilgiler verilmiştir.

1.1.1. Kovan hırsızlığı (Robbed hive)

Arılar koloni halinde yaşayan canlılardır ve besinlerini bir arada yaşadıkları kovanlarında depo etmektedirler. Arı kolonilerinin kovanlarında kendilerini besleyen besin depoları aşağıda belirtilen durumlarda azalma gösterebilmektedir;

- Arıların kış için yeterli bal depolayamadıkları zamanlarda,
- Arı sayısının arttırılacağı zamanlarda,
- Kraliçe arının yumurtlamaya teşvik edileceği zamanlarda,
- Açlık tehlikesi olduğu zamanlarda,
- İlkbaharın geç geldiği veya kötü hava koşullarının nektar toplamayı engellediği zamanlarda, arıları besleyebilecek polen, bal ve nektar miktarında azalma görülebilmektedir. Bu durumda bazı arı kolonileri, savunmaları kendilerinden daha zayıf veya sayıca az olan diğer kolonilerin bulunduğu kovanlardan besin çalabilmekte ve besinlerin depolandığı peteklere Şekil 1(a)'da görüldüğü üzere zarar verebilmektedirler. Kolonilerinden besin çalınan arıların yetersiz beslenmeden dolayı bağışıklık sistemleri zayıflamakta ve daha kolay hasta olabilmektedirler. Bunun sonucunda da bazı durumlarda tüm arı kolonisi yok olabilmektedir. Bu nedenle kovan hırsızlığına maruz kalmış arıların tespit edilip kovanlarına yeterli besin takviyesi yapılması arı kolonisi için hayati derecede önemlidir [20-22].

1.1.2. Varroa (Varroa)

Dünya'da ve ülkemizde bal arılarının en önemli paraziti olarak bilinen Varroa aslen Asya bal arısının bir parazitidir. Ancak Avrupa bal arıları geçen yüzyılda Asya'ya götürüldüğünde, Avrupa bal arısını da parazitleştirmeye başlamıştır. Şekil 1(b)'de gösterilen Varroa düz düğme şekline sahiptir, genellikle 1–1,8 mm uzunluğunda ve 1,5–2 mm genişliğinde olup sekiz bacağı vardır. Varroalar, beslenme için hem yetişkin arıların hem de arı larvalarının yağ hücrelerini emerler. Yağ hücreleri, hormon, enerji regülasyonu ve bağışıklık gibi birçok hayati fonksiyon için gerekli olduğundan, yağ hücrelerinin emilimi arıyı ciddi şekilde zayıflatır. Varroa paraziti bal arılarının aynı zamanda hemolenfini emerek kolonilerin zayıflamasına ve birkaç yıl içerisinde de yok olmasına neden olmaktadır [23].

1.1.3. Küçük kovan böceği (Small hive beetles)

Dünyada birçok bölgede görülen küçük kovan böceği (KKB), arıcılık zararlısı olmakla beraber aynı zamanda zararlı bir böcek türüdür [24]. Yetişkin bir KKB koyu kahverengi ile siyah renk arasında bir renk tonuna sahiptir ve yaklaşık 5-6 mm uzunluğundadır. KKB larvaları, polen ve balla beslenirler. Bunun sonucunda da KKB'ler Şekil 1c'de gösterildiği üzere peteklere büyük ölçüde zarar verebilirler. Bununla birlikte, şiddetli yıkım genellikle larvaların toplu olarak geliştiği dönemde olur. Böylece arı kolonisi, kovanı tamamen terk eder [25, 26]. Bunun ile birlikte arı kovanında kalan ürünler ise larva dışkısından fermente olduğundan dolayı kullanılamaz hale gelir [27-29].

1.1.4. Karınca problemi (Ant problems)

Karıncalar, arı kovanlarının içerisinde bulunan polen ve bal gibi arıların besin kaynaklarını çalmak, bal arısı yavrularını avlamak gibi Şekil 1d'de gösterildiği üzere arı kolonileri için

(a) (b) (c) (d) (e)

Şekil 1. Arı hastalıkları (Bee illness) [23-26]

zararlı faaliyetlerde bulunabilirler. Karıncalar ve bal arıları arasında en yaygın etkileşim nektar ve bal gibi şeker kaynaklarının soyulmasıdır. Bunun sonucunda kovanda aç kalan arı kolonisi kovanlarını terk etmek zorunda kalabilirler. Ayrıca karıncalar, arı kovanı içerisine dışarıdan deforme kanat virüsü, kara kraliçe hücre virüsü, israil akut felç virüsü, akut arı felci virüsü, keşmir arı virüsü gibi zararlı virüsleri taşıyarak arı kolonisine zarar verebilmektedirler [30].

1.1.5. Kraliçe kaybı (Missing queen)

Bir kraliçe bal arısının ömrü 1-5 yıl arasındadır. Kraliçesiz bir arı kolonisinin bulunduğu kovanda kaotik durumlar olabilir ve koloni uzun süre hayatta kalamaz. Şekil 1e'de kraliçe ve işçi arı birlikte gösterilmiştir. İşçi arılar kraliçenin kaybolduğu bir durumda, larva veya döllenmiş yumurtaları üç günden daha az bir süre içinde bulabilirlerse, yeni bir kraliçeyi yumurtadan çıkartabilirler. Yumurtadan çıkan kraliçe arıyı olgunluğa ulaşıncaya kadar besleyebilirler. Kraliçe arı öldüğünde döllenmiş yumurta yoksa, arıcıların arı kolonisini kurtarmasının tek yolu dışarıdan kovanın içerisine yeni bir kraliçe arının getirilmesidir [20]. Bir kraliçe arının kaybolmasından sonraki en kötü senaryo, işçi arıların yeni bir kraliçe arı yetiştirmeyi başaramaması veya arıcının yeni bir kraliçe arı bulamamasıdır. Kraliçesi olmayan bir koloni çoğalamadığı için uzun süre hayatta kalamaz [19].

1.2. İlgili Çalışmalar (Related Works)

Arı kolonilerinin devamlılığını sağlayabilmesi için sağlıklı olmaları gerekmektedir. Günümüzde teknolojinin gelişmesine paralel olarak arı hastalıklarının teşhisi için modern sistemler geliştirilmektedir. Bu başlık altında, arılar ile ilgili yapılan esnek hesaplama yöntemlerinin kullanıldığı çalışmalar, arı hastalıklarını tespit etmeye yardımcı sistemler ve özellikle görüntü işleme ile ilgili derin öğrenme konularında literatür özeti sunulmuştur. 2018 yılında Sledevic, polen taşıyan arı görüntülerini evrişimli sinir ağı (Convolutional Neural Network, CNN) kullanarak sınıflandırmayı amaçlamıştır. Bu amaç doğrultusunda ilk olarak arı kovanının girişine koyduğu kameradan aldığı 1.000 adet arı görüntüsünden yeni bir veri seti oluşturmuştur. Oluşturduğu veri seti ve CNN yöntemi ile %94 doğruluk oranına ulaşmıştır [31]. König 2019 yılında, geliştirdiği SmartComb sistemi ile kovanın iç sıcaklığını, nemini ve ağırlığını takip etmektedir. Buna ek olarak, yabani arıların istilasını ve kovan içinde oluşabilecek hastalıkları tespit etmek için geliştirdiği sisteme gaz sensörlerini entegre etmiştir. Ayrıca kovandaki varroaları saymak için bir sistem sunmuştur [32]. Braga vd., 2020 yılında geliştirdikleri sistemde, bal arısı kolonilerinin sağlık durumunu tahmin etmek için makine öğrenimine dayanan entegre bir bilgisayar sistemi ile arı kovanlarını izlemişlerdir. Sistemde kullandıkları sensörler ile anlık olarak kovan içi ve dış hava durumu ile ilgili verileri almayı başarmışlardır. Böylece geliştirdikleri sistem ile arıcıların kolonilerini kaybetmemesi ve kovanlarını doğru yönetebilmelerine imkan sunmuşlardır. Ayrıca, geliştirilen erken uyarı sistemi ile arı bakımının daha az yapıldığı kış mevsiminde, meydana gelebilecek koloni kayıplarının önlenmesine de yardımcı olmuşlardır [33]. Rodriguez 2019 yılında yaptığı çalışmada tasarladığı görüntü elde etme sistemi ile bir haftada elde ettiği video görüntülerinden, bal arılarının beslenme davranışını, üretilen bal miktarını makine öğrenmesi ve video işleme yöntemleri ile tespit etmeye çalışmıştır. Önerdiği sistemde %95'in üzerinde başarı oranı elde etmiştir [34].

Mohd-Isa vd. 2019 yılında Malezya'daki yerel bir bal arısı çiftliğinden toplanan videolardan 400 resim ile özel bir veri seti oluşturmuşlardır. Kullandıkları görüntüler çok küçük bir bal arısı türü olan Meliponine aittir. Elde ettikleri görüntülerden arka planı çıkarmak için CNN çekirdek modülünü kullanmışlardır. Veri setindeki görüntüleri sınıflandırmak için Faster R-CNN algoritmasını kullanmışlar ve %74'lük başarı sonucu elde etmişlerdir [35]. 2018 yılında Lim vd. geliştirdikleri mobil uygulama ile doğadaki farklı böcek türlerini sınıflandırmışlardır. Cep telefonu ile aldıkları görüntüler bir web sunucuya gönderilerek, sunucu üzerinde sınıflandırma işlemini gerçekleştirmişlerdir. Uygulamalarında sınıflandırma için CNN algoritmasını kullanarak, 30 farklı türdeki böceği %94 başarı oranı ile sınıflandırmayı başarmışlardır [36]. Bjerge vd. 2019 yılında, geliştirdikleri görüntü alma sistemi ile kovana giren bal arılarını tek tek görüntülemeyi başarmışlardır. Aldıkları görüntüler üzerinden varroaları tespit etmek için farklı dalga boylarındaki ışıkları test etmişlerdir. Yaptıkları testler sonucunda varroaların en iyi 700 nm dalga boyunda görüntülendiğini tespit etmişlerdir. Varroa tespitini, görüntüler üzerinde CNN algoritması sayesinde %91 F1-skoru ile gerçekleştirmişlerdir. Bal arısı kolonilerindeki varroa istila seviyelerinin tahribatsız ve otomatik olarak izlenmesinde yeni bir yöntem sunmuşlardır [37]. Bunların dışında 2018 yılında literatüre kazandırılan ve çalışmada da kullanılan veri seti ile ilgili yapılan sadece iki çalışma bulunmaktadır. Bunlardan ilki, 2019 yılında Üzen ve arkadaşlarının, CNN ile arıların sağlık durumlarını sınıflandırdıkları çalışmadır. Yaptıkları çalışmada beş farklı CNN modelini uygulamışlar ve %92,42 başarı oranı ile arı hastalıklarını sınıflandırmışlardır [38]. Aynı veri setini kullanan diğer çalışmada ise Margapuri vd. arı türlerini Google'ın Inception, Microsoft'un ResNet 50, Oxford'un VGG16 ve VGG19 mimarilerini kullanarak sınıflandırmışlardır [39].

Bütün bu çalışmalardan da anlaşılacağı üzere arı davranışlarının ve hastalıklarının sınıflandırılmasında güncel bir algoritma olan CNN ve derin öğrenmeye doğru bir yönelme vardır. Gerçekleştirilen çalışmada ise CNN ve derin öğrenme yöntemlerinin öznitelik çıkartmadaki güçlü yönleri alınarak, derin öğrenme algoritmalarında kullanılan Softmax sınıflandırıcısından farklı bir sınıflandırma algoritması ile birleştirilerek yeni bir hibrit yapı oluşturulması sağlanmıştır. Bu hibrit yapının başarım sonuçları da üçüncü bölümde kıyaslamalı olarak sunulmuştur. Böylece arı hastalıklarına yönelik farklı bir bakış açısıda geliştirilmiş olup, hibrit bir derin öğrenme yöntemi ile arı hastalıklarının tespit edilmesi de sağlanmıştır.

2. MATERYAL VE METOT
(MATERIAL AND METHOD)

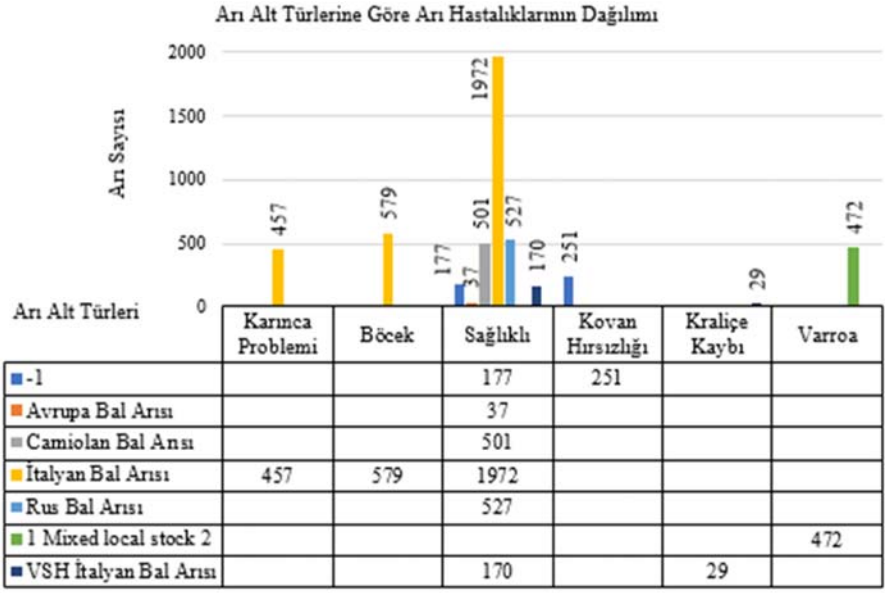
Gerçekleştirilen çalışmanın materyal ve metot kısmı, 3 bölümden oluşmaktadır. Birinci bölümde veri setinin hazırlanması ile ilgili alt işlemler gerçekleştirilmiştir. İkinci bölümde öznitelik elde etmek için kullanılan derin öğrenme algoritmasının katman yapısı oluşturulmuştur. Üçüncü bölümde ise arı hastalıklarının sınıflandırılması için derin öğrenme algoritmasının sonuna çok katmanlı geri beslemeli bir yapay sinir ağı eklenmiştir.

2.1. Arı Veri Setinin Hazırlanması (Prepearing Bee Dataset)

Çalışmada Yang tarafından 26 Ağustos 2018 tarihinde, literatüre sunulan "The BeeImage Dataset: Annotated Honey Bee Images" isimli görüntü veri seti kullanılmıştır [40]. Kullanılan veri setinde, her bir arı görüntüsü için dosya adı, tarih, saat, bölge, posta kodu, alt türler, hastalık türü, polen taşıma durumu ve statüsü (işçi, kraliçe) olarak etiketlenen 5.172 adet arı görüntüsü bulunmaktadır. Amerika Birleşik Devletleri'nde, 7 farklı şehirde bulunan 8 farklı bölgedeki arı kovanlarına yerleştirilen kameralardan elde edilen arı görüntüleri kullanılmıştır. Şekil 2'de görüldüğü üzere veri setindeki görüntüler hastalık durumlarına göre; karınca problemi, KKB, sağlıklı, kovan hırsızlığı, kraliçe kaybı ve varroa olmak üzere altı farklı sınıfa ayrılmıştır. Bunun ile birlikte beş farklı arı alt türünden hariç olarak, türü belirlenemeyen ("-1") ve yerel ırkla karışık ("1 mixed local stock 2") türler olmak üzere toplamda yedi farklı arı alt türü bulunmaktadır. Kullanılan veri setinde, farklı ortam şartlarında (ışık, gölgelenme, kamera açıları, vb.) ve farklı resim boyutlarında elde edilen görüntüler bulunmaktadır. Bu nedenle görüntüler üzerinde bir ön işleme ve eleme işleminin yapılması gerekmektedir. Ön işleme ve eleme için yapılan incelemede, görüntünün çözünürlüğünün çok düşük olması (Şekil 3a), görüntülerde arının bulunmaması (Şekil 3b), ya da görüntü üzerinde arının çok küçük bir parçasının bulunması (Şekil 3c) kriterlerine göre veri setindeki birçok görüntünün kullanılmaya uygun olmadığı tespit edilmiştir.

Şekil 3. Veri setinden elimine edilen bazı görüntü örnekleri
(Some examples of images eliminated from the dataset)

Hâlbuki derin öğrenme algoritmalarının uygulanabilmesi için çok sayıda anlamlı ve etiketli veriye gereksinim duyulmaktadır [41]. Ayrıca tercih edilen veri seti üzerinde, daha iyi bir sınıflandırma yapılabilmesi için veri sayısı az olan sınıflara ait veri sayısının arttırılması gerekmektedir [42-44]. Bu nedenle dengeli sınıf dağılımının sağlanabilmesi için veri arttırma ve veri boyutunun standart hale getirilmesine ihtiyaç duyulmaktadır. Yeni bir veri seti oluşturulurken de kraliçe kaybı hastalığına ait olan 29 adet görüntü farklı görüntü arttırma işlemlerine (90 derece ile tam tur döndürme, her döndürme hareketinde aynalama, en yakın komşu enterpolasyonu, üç çizgili enterpolasyon işlemi) tabi tutularak toplam 493 görüntü elde edilmiştir. Kovan hırsızlığına ait olan 251 görüntü ise bir kez 90 derece döndürülürken en yakın komşu enterpolasyonu işlemine tabi tutularak toplam 502 görüntü elde edilmiştir. Böylece veri setindeki toplam görüntü sayısı 5.887'ye çıkarılmıştır. Veri arttırma işleminden sonra en az görüntüye sahip olan karınca problemi hastalığı baz alınarak, her bir hastalık için 450 adet görüntü alınmıştır. Bu nedenle görüntüler yeniden düzenlenmiş ve her bir hastalık türünden 450 adet olmak üzere toplamda 2.700 adet etiketli görüntüden oluşan yeni bir görüntü veri seti oluşturulmuştur.



Şekil 2. Arı alt türlerine göre arı hastalıklarının dağılımı (Distribution of bee diseases according to bee subspecies)

2.2. Derin Öğrenme (Deep Learning)

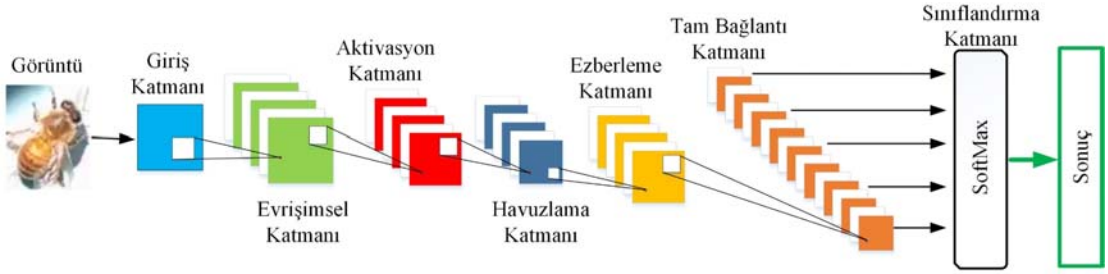
Bilgi teknolojilerinin gelişmesinin sonucunda çok miktarda veri artışı meydana gelmiştir. Artan veri miktarından anlamlı özniteliklerin elde edilmesi için geliştirilen birçok farklı esnek hesaplama yöntemi bulunmasının yanında, anlamlı verileri aramak için de geliştirilen birçok arama algoritması bulunmaktadır [45, 46]. Özellikle Yapay Sinir Ağları (YSA) gibi yöntemlerin kullanılmasında ilk olarak veri seti içerisinden birçok öznitelik çıkarmak gerekmektedir. Daha sonrasında uzman kişi yardımı ile bu öznitelikler içerisinden en iyi özniteliklerin tespit edilmesi gerekmektedir.

Veri üzerinden spesifik özelliklerin elde edilmesi gerektiğinde, bu işlemleri YSA modelleri ile gerçekleştirmek sorun olabilmektedir. Fakat derin öğrenme modellerinde ise uzman bir kişiye gereksinim duyulmadan özniteliklerin otomatik olarak çıkartılabilmesi, derin öğrenmeyi popüler hale getirmiştir. Bu nedenle derin öğrenme modelleri ekonomiden eğitime kadar birçok farklı alanda kullanılabilmektedir [47-49]. Klasik bir derin öğrenme algoritmasında veriler Şekil 4'de gösterildiği sırayla; giriş, evrişimsel, aktivasyon, havuzlama, ezberleme, tam bağlantı ve sınıflandırma katmanlarından geçirilerek sonuç üretilmektedir. Şekil 4'de ki katmanlar, farklı derin öğrenme algortimalarında arka arkaya farklı sayılarda ve kombinasyonlarda eklenerek AlexNET, VGG16, GoogleNet gibi farklı derin öğrenme mimarilerini oluşturabilmektedir.

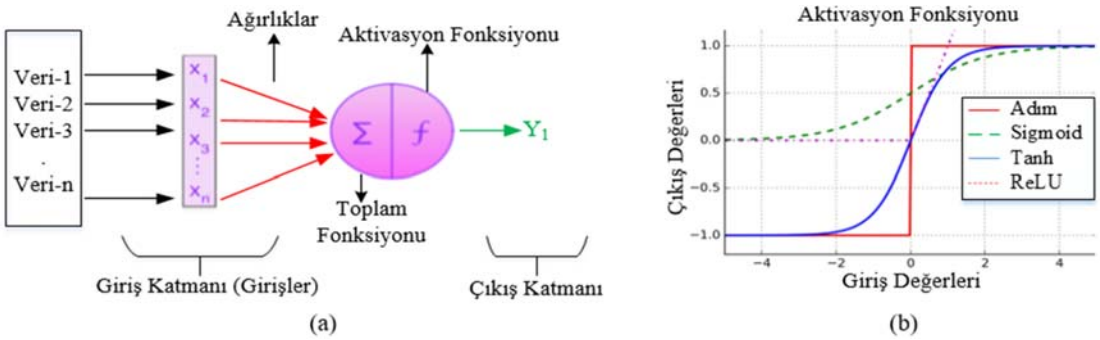
2.3. Yapay Sinir Ağları (Artifical Neural Network)

Yapay sinir ağları günümüzde askeriyeden tıba, tıptan ziraate, ziraatten eğitime kadar birçok multidisipliner alanda tercih edilen bir hesaplama yöntemidir [50-53]. Bu yöntem biyolojik sinir ağlarının dış ortamdan aldığı sinyale göre uyarılma, aldığı uyarıyı iletme, uyarıya göre karar verme gibi düşünsel davranışları örnek alarak geliştirilmiş bir esnek hesaplama yöntemidir [54-56]. YSA metodlarının geliştirilen ilk versiyonu Perceptron modelidir ve Şekil 5a'da sunulmuştur. Perceptron modelinde hazırlanan veriler, giriş olarak nitelendirilen katmanda alındıktan sonra, rastgele üretilen ağırlıklar ile çarpılır ve elde edilen çarpım sonuçları toplam fonksiyonuna aktarılır. Toplam fonksiyonundan elde edilen değerlerde bir aktivasyon fonksiyonuna aktarılarak çıkış değeri üretilir. Literatürde kullanılan temel toplam fonksiyonları; çarpım, minimum, maksimum, çoğunluk ve kümülatif fonksiyonlardır [57]. Toplam fonksiyonundan elde edilen değerlerin hızlı bir şekilde sonuç vermesi için türevi kolay hesaplanan aktivasyon fonksiyonları tercih edilmektedir. YSA'larda en çok tercih edilen aktivasyon fonksiyonları Şekil 5b'de verilmiştir [58]. Aktivasyon fonksiyonundan elde edilen değerler doğrudan çıkış değeri olabileceği gibi, başka bir katmanda bulunan hücreye giriş değeri olarak da aktarılabilmektedir. Elde edilen değerler çıkışta beklenen değerler ile karşılaştırılır. Karşılaştırma işlemlerinden elde edilen hata değeri, ortalama karesel hata fonksiyonu gibi çeşitli hata değerlendirme fonksiyonları kullanılarak hesaplanmaktadır [59].

YSA, bir hücreden oluşabileceği gibi birçok hücrenin bulunduğu katmanlardan da oluşabilmektedir. Literatürde Tek Katmanlı Algılayıcı (TKA), Çok Katmanlı Algılayıcı (ÇKA), Geri Beslemeli, ADALINE / MADALINE gibi lineer ve lineer olmayan sistemler için geliştirilen birçok farklı YSA modelleri de bulunmaktadır. Geliştirilen sistemde Çok Katmanlı Geri Beslemeli Yapay Sinir Ağı (ÇKGB-YSA) modeli temel alındığı için, aşağıda bu model sunulmuştur.



Şekil 4. Derin öğrenme mimarisi (Architecture of deep learning)



Şekil 5. Percepton modeli (Perceptron model)

2.3.1. Çok katmanlı geri beslemeli yapay sinir ağı (ÇKGB-YSA) modeli
(Multi-layer feedback neural network (MLFB-ANN) model)

ÇKGB-YSA doğrusal bir çözümü olmayan sistemler için geliştirilmiş danışmanlı bir YSA modelidir. Bu modelde kullanılan veriler ilk olarak eğitim ve test verileri olmak üzere ikiye ayrılır. Eğitim aşamasında, modele aktarılan giriş verilerinden bir çıkış verisi üretmesi beklenir. Üretilen değer ile beklenen çıkış değeri kıyaslanır, elde edilen hata değeri istenilen seviyeye indirildiğinde ya da istenilen iterasyon sayısı tamamladığında, sistem danışmanlı eğitim işlemini tamamlamış olur. Bu aşamadan sonra sistemin başarım oranını hesaplamak için test verileri hazırlanan eğitilmiş modele sunulur [60]. Sistemin genel çalışması iki aşamada gerçekleşmektedir. Birinci aşamada sistem ileriye doğru çalışarak bir çıkış değeri üretmektedir. İkinci aşamada ise sistemin üretmiş olduğu değer ile beklenen değer arasındaki fark, hata fonksiyonları ile hesaplanır ve elde edilen hata değeri istenilen oranda değilse, Şekil 6'da gösterildiği gibi çıkış katmanından gizli katmana, gizli katmandan da giriş katmanına doğru geri yayılması sağlanır.

2.3.1.1. ÇKGB-YSA'nın ileri yönde eğitilmesi
(MLFB-ANN training forward)

Bu model ilk olarak ileri yönde çalıştırılır. Bunun için veriler doğrudan giriş katmanına sunulur. Bir hücre (n) için giriş katmanına sunulan *i.* veri ($x(i)$), rastgele üretilen *i.* ağırlık değeri ($w(i)$) ile Eş. 1'de gösterildiği gibi çarpılıp toplanarak, toplam fonksiyonuna aktarılır. Böylece bir hücre için *NET* girdi değeri hesaplanmış olur.

$$\boldsymbol{NET_n = \sum_i^n x(i)w(i)} \tag{1}$$

$$Ç_n = \frac{1}{1+e^{-(NET_n+\beta_n)}} \tag{2}$$

Hesaplanan değer aktivasyon fonksiyonuna gönderilerek hücrenin çıkışı hesaplanmaktadır. Eğer sistemde aktivasyon fonksiyonu olarak Sigmoid fonksiyonu tercih edilirse Eş. 2 kullanılmaktadır. Eş. 2'de ki β_n, gizli katmandaki n. hücreye bağlı eşik değerini ifade etmektedir. Gizli katmanlardaki ve çıkış katmanında bulunan tüm hücreler için aynı işlem tekrarlanmaktadır. Böylece çıkış katmanındaki, çıkış değerleri ($Ç_1, Ç_2, \ldots Ç_n$) hesaplanarak sistemin ileri yönde çalışmış olur.

2.3.1.2. ÇKGB-YSA'nın geri yönde eğitilmesi
(MLFB-ANN training backward)

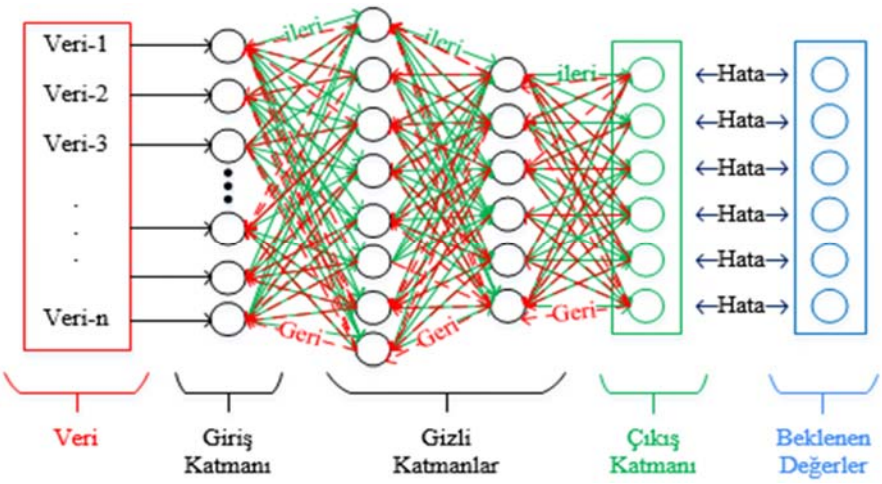
m adet çıkışa sahip ÇKGB-YSA modelinin ileri yönde çalıştırılması sonucunda elde edilen çıkış değerleri ($Ç_n$) ile beklenen değerler (B_n) arasındaki hata değerleri (E_n) Eş. 3 ile hesaplanır. Bu hata değerleri, kendisinden önceki elemanlara sondan başa doğru belli oranlarda dağıtılarak, ağın ağırlıklarının güncellenmesi sağlanmaktadır. Hatanın geri dağıtılması Şekil 6'da kırmızı oklarla gösterilmiştir. Bu işlem bir katmandaki hücreler arasında olabileceği gibi, katmanlar arasındaki hücreler arasında da gerçekleştirilebilir. Hatanın geri dağıtılması işlemine, kısaca geri beslemede denilmektedir.

$$E_n = B_n - Ç_n;\ n = 1, .., m \tag{3}$$

$$TH = \frac{1}{n}\sum_1^n (E_n)^2 \tag{4}$$

Çıktı katmanından elde edilen hataların (E_n) toplanması ile Toplam Hata (TH) Eş. 4 ile hesaplanmaktadır. ÇKGB-YSA modelinin temel amacı, elde edilen hata değerinin, bu hataya neden olan işlem elemanlarının ağırlıklarına dağıtılarak, TH'ı en aza indirmektir. Bu hatanın ağırlıklara geri dağıtılması da sondan başa doğru iki aşamada gerçekleştirilmektedir.

1. Aşama: Çıktı katmanı ile ara katman arasındaki ağırlıkların güncellenmesinde, ara katmandaki j. işlem elemanını çıktı katmanındaki n. işlem elemanına bağlayan bağlantının ağırlığındaki değişim miktarı ΔA^a olarak kabul edilirse; herhangi bir t zamanındaki (t. iterasyonda) ağırlığın değişim miktarı Eş. 5 ile hesaplanmaktadır.



Şekil 6. Çok katmanlı geri beslemeli YSA modeli (Multi-layer feedback ANN model)

$$\Delta A_{jn}^{a}(t) = \eta\delta_n Ç_j^a + \alpha\Delta A_{jn}^{a}(t-1) \quad (5)$$

$$\delta_n = f'(NET).E_n \quad (6)$$

$$A_{jn}^{a}(t) = A_{jn}^{a}(t-1) + \Delta A_{jn}^{a}(t) \quad (7)$$

Eş. 5'de η öğrenme katsayısını, α momentum katsayısını, δ_n ise n. çıktı ünitesinin hatasını ifade etmektedir. Öğrenme katsayısı, ağırlıkların değişim miktarının bir sonraki değişime belli miktarda etki etmesini sağlamak için kullanılmaktadır. Momentum katsayısını ise YSA'nın öğrenme sürecinde yerel bir optimum noktaya takılıp kalmaması ve ağırlık değişim değerinin bir sonraki değişime eklenmesini sağlamak için kullanılmaktadır. n. çıktı ünitesinin hatası Eş. 6 ile hesaplanmaktadır. Eş. 6'daki $f'(NET)$ aktivasyon fonksiyonun türevini ifade etmektedir. Ağırlıkların değişim miktarı hesaplandıktan sonra t. iterasyonda ki yeni ağırlık değerleri Eş. 7'deki formül ile hesaplanmaktadır. Aynı şekilde eşik değer ünitesindeki, değişim miktarları tespit edilerek, ağırlıkların Eş. 8 ve Eş. 9'a göre güncellenmesi gerekmektedir. Çıktı katmanında bulunan işlem elemanlarının eşik değer ağırlıkları $\beta^{ç}$ ile gösterilirse; bu ünitenin çıktısının sabit ve 1 olması nedeni ile değişim miktarı, Eş. 8'e göre hesaplanmaktadır. Eşik değerinin t. iterasyonda ki ağırlığının yeni değeri Eş. 9'da gösterilmektedir.

$$\Delta\beta_n^{ç}(t) = \eta\delta_n + \alpha\Delta\beta_n^{ç}(t-1) \quad (8)$$

$$\beta_n^{ç}(t) = \beta_n^{ç}(t-1) + \Delta\beta_n^{ç}(t) \quad (9)$$

2. Aşama: Gizli katman ile girdi katmanı arasındaki ya da gizli katmanlar arasındaki ağırlıkların güncellenmesinde, her ağırlık için çıktı katmanındaki sadece bir işlem elemanının hatası işleme alınmaktadır. Bu hataların oluşmasında girdi katmanı ve gizli katman arasındaki ağırlıkların (eğer varsa birden çok gizli katman arasındaki ağırlıkların) etkisi vardır. Çünkü, çıkış katmanına gelen bütün bilgiler, girdi katmanı ve önceki gizli katmandan gelmektedir. Bu nedenle girdi katmanı ile gizli katman arasındaki (veya gizli katmanlar arasındaki) ağırlıkların, çıktı katmanındaki işlem elemanlarından kaynaklanan hatadan payını alması gerekmektedir. Bu ağırlıklardaki değişim miktarı (örneğin girdi katmanı ile gizli katman arasındaki ağırlıkların değişimi) ΔA^i ile gösterilirse, bu değer Eş. 10 ile elde edilmektedir.

$$\Delta A_{kj}^{i}(t) = \eta\delta_j^a Ç_k^i + \alpha\Delta A_{kj}^{i}(t-1) \quad (10)$$

$$\delta_j^a = f'(NET).\sum_n \delta_n A_{jn}^a \quad (11)$$

$$A_{kj}^{i}(t) = A_{kj}^{i}(t-1) + \Delta A_{kj}^{i}(t) \quad (12)$$

Eş. 10'da kullanılan hata terimi, Eş. 11'deki gibi hesaplanmaktadır. Ağırlıkların yeni değerleri ise Eş. 12 ile hesaplanmaktadır. Ara katmandaki eşik değer aralıkları β^a ile gösterilirse değişim miktarı Eş. 13 ile hesaplanır. Eşik değer ünitesinin yeni ağırlık değerleri de Eş. 14 ile hesaplanmaktadır.

$$\Delta\beta_j^a(t) = \eta\beta_j^a + \alpha\Delta\beta_j^a(t-1) \quad (13)$$

$$\beta_j^a(t) = \beta_j^a(t-1) + \Delta\beta_j^a(t) \quad (14)$$

Böylece ileri doğru çalışan YSA, geriye doğru da çalıştırılarak tüm ağırlıkları güncellenmektedir. İleri geri çalışma işlemi, istenilen hata seviyesine ya da iterasyon sayısına ulaşıncaya kadar devam edecektir. İstenilen değere ulaştığında sistemin eğitimi tamamlanmaktadır.

3. DERİN ÖĞRENME VE ÇKGB-YSA İLE ARI HASTALIKLARININ TESPİT EDİLMESİ
(DETERMINATION OF BEE DISEASES WITH DEEP LEARNING AND MLFB-ANN)

Yapılan çalışmada derin öğrenmenin güçlü yanlarından biri olan otomatik öznitelik çıkarma işlemi ile ÇKGB-YSA'nın güçlü yanlarından olan sınıflandırma işlemi birleştirilerek hibrit bir sınıflandırma algoritması geliştirilmiştir. Geliştirilen algoritma iki bölümden oluşmaktadır. Birinci bölümde temel derin öğrenme işlemleri uygulanarak öznitelik vektörü elde edilmiştir. İkinci bölümde ise ilk bölümden elde edilen öznitelik değerleri kullanılarak, ÇKGB-YSA ile sınıflandırma işlemleri gerçekleştirilmiştir. Ayrıca aynı öznitelikler, Softmax sınıflandırıcısı kullanılarak elde edilen değerler ile geliştirilen algoritmadan elde edilen değerler birbiriyle karşılaştırılmıştır. Çalışmada kullanılan veri setindeki görüntülerin 1.890 adeti (%70) eğitim, 405 adeti (%15) test ve 405 adeti (%15) doğrulama verisi olmak üzere üçe ayrılmıştır. Buna ek olarak gerçekleştirilen çalışma Matlab ortamında, 2x Intel Xeon Gold 6130 (32 Core, 2.10GHz, 3.70GHz Turbo, 22MB - İşlemci), 64GB DDR4 (2666MHz ECC – Ram), Pny Quadro P4000 256 Bit GDDR5 8 GB (GPU), X99 Huananzhi Çift İşlemci DDR4 (AnaKart) ve 500GB SSD (Harddisk) donanıma sahip bir iş istasyonu üzerinde geliştirilmiştir.

3.1. Evrişimli Sinir Ağının Eğitilmesi Esnasında Kullanılan Optimizasyon Fonksiyonu
(Optimization Function used During Training of Convolutional Neural Network)

Evrişimsel sinir ağı için hiperparametre optimizasyonunda literatürde, Izgara Arama (Grid Search), Genetik Algoritma (Genetic Algorithm, GA), Parçacık Sürü Optimizasyonu (Particle Swarm Optimization, PSO), Harmonik Arama (Harmonic Search, HS), Rastgele Arama ve Bayes Optimizasyonu algoritmaları kullanılmaktadır [61]. Gerçekleştirilen çalışmada da Izgara Arama algoritması kullanılmıştır. Izgara Arama algoritması, hiperparametrelerin sonsuz sayıda değer alabileceği durumlarda kullanılır. Problem hakkındaki ön bilgiler kullanılarak, hiperparametrelerin alabilecekleri değer aralıkları tahmin edilebilir. Tahmin edilen aralıklarda ki hiperparametreler için bir değer listesi oluşturulur [62]. Izgara Arama ile hiperparametre seçim işleminde; belirlenen aralıkta bulunan tüm değerlerin kombinasyonları sırasıyla test edilip, elde edilen sonuçlar değerlendirilerek en iyi hiperparametreler tespit edilebilir [63]. Evrişimli Sinir Ağının eğitilmesinde, test edilen 27 hiperparametre ve aralıkları ile elde edilen en iyi hiperparametre değerleri Tablo 1'de sunulmuştur.

Tablo 1. Test edilen hiperparametre değerleri (Tested hyper parameters and their values)

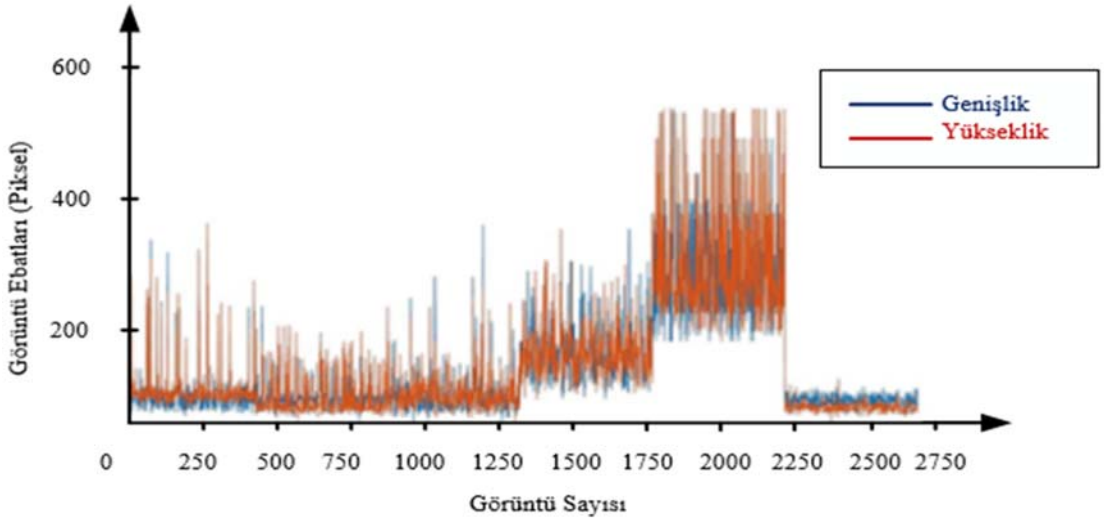
	Katman	Hiper Parametreler	Test Edilen Aralık	En İyi Değerler
1. Bölüm	Evrişimsel katmanı	Katman sayısı (Eks)	2-10	4
		Filtre ebatı (Efe)	2x2, 3x3, 4x4, 5x5	Tüm katmanlar: 3x3
		Filtre sayısı (Efs)	2, 3,4	1.katman:3, 2.katman:3 3.katman:3, 4.katman:4
		Atlama oranı (Eao)	1,2	1.katman:1, 2.katman:2 3.katman:2, 4.katman:2
		Aktivasyon fonksiyonu (Eaf)	Sigmoid, Hiperbolik tanjant sigmoid, Adım ve ReLU	ReLU, ReLU
2. Bölüm	Havuzlama	Fonksiyon (Hf)	Minumum, Ortalama, Maksimum	Maksimum
		Filtre ebatı (Hfe)	2x2, 3x3, 4x4	2x2
		Atlama oranı (Hao)	1,2	2
	Evrişimsel katmanı	Katman sayısı (Eks)	2-10	2
		Filtre ebatı (Efe)	2x2, 3x3, 4x4, 5x5	Tüm katmanlar: 3x3
		Filtre sayısı (Efs)	2, 3,4	1.katman:3, 2.katman:3
		Atlama oranı (Eao)	1,2	1.katman:1, 2.katman:1
		Aktivasyon fonksiyonu (Eaf)	Sigmoid, Hiperbolik tanjant sigmoid, Adım ve ReLU	ReLU
	Havuzlama	Katman sayısı (Hks)	Minumum, Ortalama, Maksimum	Maksimum
		Fonksiyon (Hf)	2x2, 3x3, 4x4	2x2
		Atlama oranı (Hao)	1,2	2
3. Bölüm	Havuzlama	Fonksiyon (Hf)	Minumum, Ortalama, Maksimum	Ortalama
		Filtre ebatı (Hfe)	2x2, 3x3, 4x4	2x2
		Atlama oranı (Hao)	1,2	2
	Evrişimsel katmanı	Katman sayısı (Eks)	2-10	2
		Filtre ebatı (Efe)	2x2, 3x3, 4x4, 5x5	Tüm katmanlar: 3x3
		Filtre sayısı (Efs)	2, 3,4	1.katman:3, 2.katman:3
		Atlama oranı (Eao)	1,2	1.katman:1, 2.katman:1
		Aktivasyon fonksiyonu (Eaf)	Sigmoid, Hiperbolik tanjant sigmoid, Adım ve ReLU	ReLU
	Havuzlama	Katman sayısı (Hks)	Minumum, Ortalama, Maksimum	Ortalama
		Fonksiyon (Hf)	2x2, 3x3, 4x4	2x2
		Atlama oranı (Hao)	1,2	2

3.2. Derin Öğrenme Yöntemleri ile Özniteliklerin Elde Edilmesi
(Obtaining Attributes with Deep Learning Methods)

2.700 adet arı görüntüsünden oluşan veri seti, derin öğrenmede kullanılan farklı katmanlardan geçirilerek bir resim için 17.496×1 boyutunda bir öznitelik vektörü elde edilmiştir. Veri setinde bulunan tüm görüntülerin genişlik ve yüksekliklerinin dağılımını gösteren grafik Şekil 7'de gösterilmiştir. Resimlerin farklı boyutlardan dolayı, giriş katmanında genişlik ve yüksekliklerinin ortalaması tespit edilerek tüm görüntüler 200×200 boyutlarına getirilmiştir.

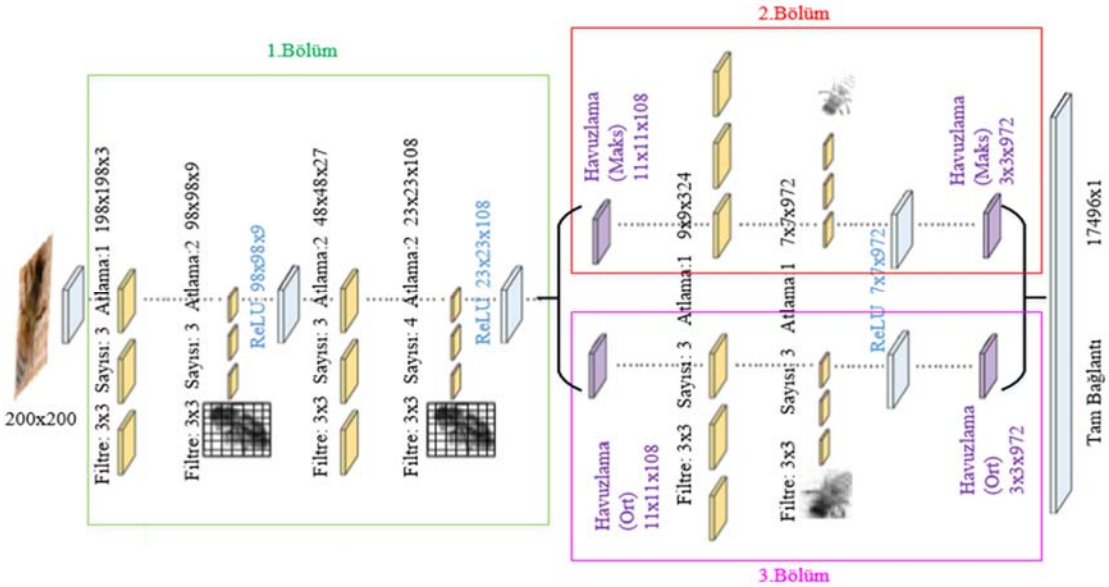
Çalışmada kullanılan derin öğrenme katmanlarının genel özellikleri ve katman giriş-çıkış durumları Şekil 8'de sunulmuştur.

Şekil 9'da sunulan ve tasarımı yapılan derin öğrenme mimarisi, veri setindeki tüm görüntülere sırasıyla uygulanarak 2.700×17.496 boyutunda bir öznitelik matrisi elde edilmiştir. Kullanılan veri setinden alınan her bir görüntü RGB renk uzayından gri renk uzayına dönüştürüldükten sonra 200×200 boyutlarına getirilmiştir. Bu şekilde ön işlemden geçirilen görüntüler Şekil 9'da gösterilen sırayla katmanlar arasında işleme tabi tutulmuştur. Bu katmanlar; birinci bölümde evrişimsel_1 (3×3), evrişimsel_2 (3×3), aktivasyon_1 (ReLU), evrişimsel_3 (3×3), evrişimsel_4 (3×3), aktivasyon_2 (ReLU) katmanlarıdır. Bu aşamaya kadar 6 farklı katmandan geçirilen görüntü 200×200 boyutlarından, 23×23×108 boyutlarına getirilmiştir. Sonraki aşamada ise Şekil 9'da gösterildiği gibi iki farklı koldan (2. Bölüm ve 3. Bölüm) öznitelik çıkarma işlemine devam edilmiştir. Görüntü üzerinde normal bir dağılıma sahip olan değerlerde agresif olarak meydana gelen artışların ve bu durum için gerekli özniteliklerin tespit edilebilmesi için geliştirilen derin öğrenme mimarisi bu aşamada iki ayrı kola ayrılmıştır.



Şekil 7. Görüntülerin ebatlarının dağılımı (Distribution of the sizes of the images)

Şekil 8. Geliştirilen sistemde kullanılan derin öğrenme katmanları (Deep learning layers used in the developed system)



Şekil 9. Geliştirilen derin öğrenme mimarisi yapısı (Developed deep learning architecture structure)

Bu nedenle sistemin normal durumu için ortalama, agresif durumların özniteliklerinin tespiti içinde maksimum havuzlama katmanı kullanılmıştır. Böylece geliştirilen algoritma ilerleyen dönemlerde sadece tek bir resim için değil, video görüntüleri içerisindeki agresif değişimlerin de tespit edilmesi için kullanılabilecek niteliktedir. Bu kollardan ilkinde (2. Bölüm), havuzlama_1a (maksimum, 2×2) katmanında en büyük değerler elde edildikten sonra sırasıyla, evrişimsel_5a (3×3), evrişimsel_6a (3×3), aktivasyon_3a (ReLU) ve havuzlama_2a (maksimum, 2×2) katmanlarından geçirilmiştir. Diğer kolda ise (3. Bölüm), havuzlama_1b (ortalama, 2×2) katmanında ortalama değerler elde edildikten sonra sırasıyla, evrişimsel_5b (3×3), evrişimsel_6b (3×3), aktivasyon_3b (ReLU) ve havuzlama_2b (ortalama, 2×2) katmanlarından geçirilmiştir. Her bir koldan gelen 8.748 adet öznitelik değeri, tam bağlantı katmanında tek boyutlu 17.496×1 ebatında bir öznitelik vektörüne dönüştürülmüştür. Elde edilen bu öznitelik vektörü, ÇKGB-YSA'nın girişini oluşturmuştur. 2.700 adet görüntüden öznitelik çıkarma işlemi, Bölüm 3'de özellikleri verilen bilgisayar ile 97,45 dakikada gerçekleştirilmiştir.

Öznitelik çıkarma işlemleri Şekil 9'da gösterildiği üzere tam bağlantı katmanına kadar devam etmektedir. Tam bağlantı katmanında 17.496×1 ebatında bir öznitelik matrisi oluşturulmuştur. Bu sayede görüntü öznitelikleri bir katmandan diğerine aktarılarak öznitelik kaybı azaltılmış ve öznitelik kullanım oranı arttırılmıştır. Derin öğrenme algoritmalarında genellikle son katmanda Softmax [64, 65], hidden markov [66] ve destek vektör makineleri (DVM) [67] gibi sınıflandırıcılar kullanılmaktadır.

3.3. Softmax (Softmax)

Derin öğrenme algoritmalarında sınıflandırıcı olarak Softmax seçilirse, Eş. 15 ile ifade edildiği gibi bu katmana gelen öznitelikler ($\boldsymbol{x}$), esnek hesaplama yöntemleri ile belirlenen ağırlık değeri ($\boldsymbol{w}$) ile çarpılıp, fişekleme değeri ($\boldsymbol{b}$) ile toplanarak, çıkış değeri ($\boldsymbol{y}$) üretilmektedir. Eş. 16 ile 0-1 arasında olasılıksal üretilen çıkış değerlerinin sonuçlarının ($\boldsymbol{S}(\boldsymbol{y_i})$), toplamı bire eşittir. Üretilen olasılık değerleri arasında en yüksek değere sahip olan sonuç, Softmax'in çıkışı kabul edilir ve bu sonuç ilgili sınıfa dahil edilir. Softmax'de sınıf sayısı kadar sistemin çıkışı bulunmaktadır ve üretilen sonucun hangi sınıfa ait olduğu olasılık hesabına dayanmaktadır. Yapılan çalışmada arıların altı farklı hastalık durumu olduğu için Softmax katmanının çıkışında her bir örnek için altı farklı olasılıksal çıkış bulunmaktadır. Üretilen Softmax değerlerinin hangisinin olasılıksal değeri bire en yakın ise, sonucun o sınıfta olduğu kabul edilmektedir.

$$y = wx + b \tag{15}$$

$$S(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{k} e^{y_j}}, i = 1,2,\ldots k \tag{16}$$

3.4. ÇKGB-YSA ile Sınıflandırma (Classification with MLFB-ANN)

Çalışmada Softmax, hidden markov ve DVM gibi sınıflandıcılardan farklı olarak ÇKGB-YSA modeli kullanılmıştır. Böylece derin öğrenme katmanları ile ÇKGB-YSA modelinin katmanları birbiri ardına eklenerek Şekil 10'daki hibrit yapı oluşturulmuştur. Derin öğrenme katmanlarından elde edilen öznitelikler bu model ile yaklaşık olarak 62,35 saatte eğitilmiştir. Modelin eğitimi, erken sonlandırma geri çağrımı işlemi ile gerçekleştirilmiştir. Bu modelde derin öğrenme algoritmasının son katmanından gelen veriler doğrudan ÇKGB-YSA modelinin giriş katmanına aktarılmıştır.

ÇKGB-YSA modeli için 6 adet hiperparametre kullanılmış olup, Izgara Arama algoritmasında test edilen aralıklar ve tespit edilen en iyi değerler Tablo 2'de sunulmuştur.

Tablo 2. Test edilen ÇKGB-YSA hiperparametre değerleri
(Tested MLFB-ANN hyper parameters and their values)

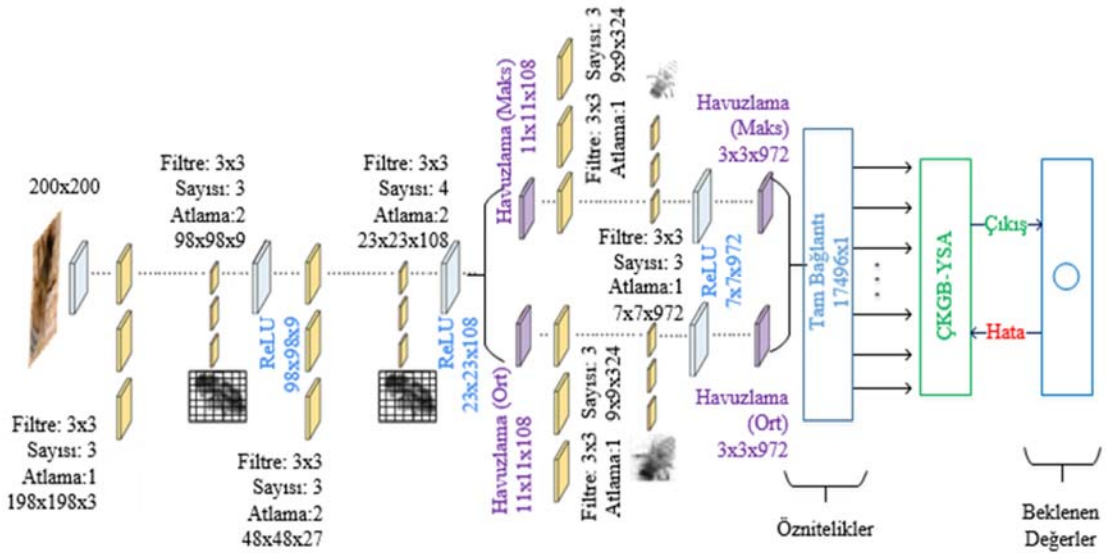
Model	Hiper Parametreler	Test Edilen Aralık	En iyi değer
ÇKGB-YSA	Gizli katman sayısı	1-5	2
	Nöron sayısı	4-20	10
	Aktivasyon fonksiyonu	Adım, Sigmoid, Tanh, ReLU	Tanh
	Öğrenme oranı (α)	0,1-0,9	0,7
	Momentum (m)	0,1-0,9	0,2
	İterasyon sayısı (is)	100-10.000	1.000

Kullanılan ÇKGB-YSA modeli Şekil 11'de gösterildiği üzere, bir adet giriş katmanı, iki adet gizli katman ve bir adet çıkış katmanından oluşmaktadır. Her iki gizli katmanda ise 10'ar adet nöron bulunmaktadır. Bu nöronlarda aktivasyon fonksiyonu olarak hiperbolik tanjant sigmoid transfer (Tanh) fonksiyonu kullanılmıştır. Gerçekleştirilen çalışmada derin öğrenme algoritmalarında genel olarak kullanılan Softmax sınıflandırıcısından farklı olarak ÇKGB-YSA modeli kullanılarak sistemin tek bir çıkış değeri üretmesi sağlanmıştır.

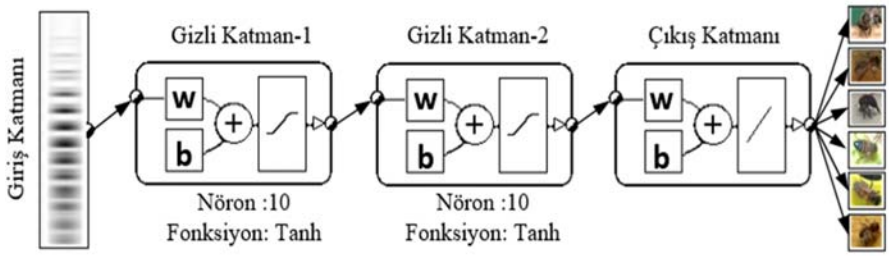
Üretilen çıkış değeri arıların altı farklı hastalık durumundan sadece bir tanesini ifade eder. Softmax sınıflandırıcısında olduğu gibi olasılıksal herhangi bir hesaplama işlemi yapılmamaktadır. Sistem doğrudan ÇKGB-YSA'nın üretmiş olduğu sınıf bilgisini vermektedir. Kullanılan ÇKGB-YSA modeli ile %95,04 başarı oranı ile arıların altı farklı hastalık durumunun sınıflandırılması sağlanmıştır. Bu durum Şekil 12'de gösterilmiş ve aynı zamanda sistemin genel performansını değerlendirmek için karmaşıklık matrisi de kullanılmıştır.

3.5. Karmaşıklık Matrisi (Confussion Matrix)

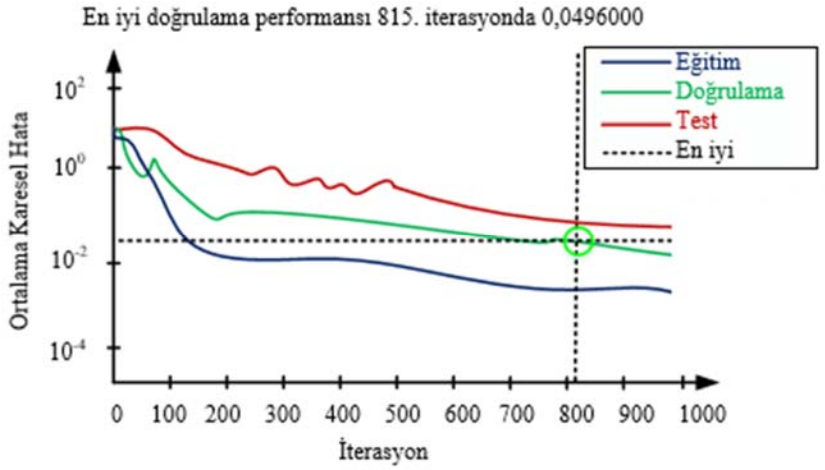
Karmaşıklık matrisleri, eğitim ve test verilerinin tahmin sonuçlarını değerlendirmek için kullanılmaktadır. Bu yöntemde genellikle doğruluk, kesinlik, duyarlılık ve F1-skor değerleri ölçülmektedir. Doğruluk (Accuracy), sınıflandırmanın ya da tahminin ne kadar doğru sonuç verdiğinin hesaplanması işlemidir. Bu işlem Eş. 17 ile gerçekleştirilmektedir. Kesinlik (Precision), doğru olarak tahmin edilen örneklerin kaç tanesinin gerçekten doğru (TP) olduğunu ölçer. Hedef, yanlış-pozitiflerin (FP) sayısını sınırlamak olduğunda, kesinlik bir performans ölçütü olarak kullanılır. Modelin çok sayıda yanlış pozitif üretmemesi yani yüksek bir hassasiyete sahip olması önemlidir. Kesinlik, FP'nin FN'den daha yüksek olduğu durumlarda kullanışlı bir ölçümdür ve Eş. 18 gösterilmektedir. Duyarlılık (Recall), tüm sınıflardan ne kadar doğru tahmin edildiğini ölçmek için kullanılır ve Eş. 19 ile hesaplanmaktadır. Kesinlik ile duyarlılık arasında bir denge vardır. Kesinlik ile duyarlılık



Şekil 10. Gerçekleştirilen hibrit yapı (Described hybrid structure)



Şekil 11. Kullanılan ÇKGB-YSA modelinin içeriği (Content of the MLFB-ANN model used)



Şekil 12. Sistemin performansı (System performance)

Tablo 3. Karmaşıklık matrisi (Confussion matrix)

	Karınca problemi		KKB		Sağlıklı		Kovan Hırsızlığı		Kraliçe kaybı		Varroa	
	ÇKGB-YSA	SOFTMAX	ÇKGB-YSA	SOFTMAX	ÇKGB-YSA	SOFTMAX	ÇKGB-YSA	SOFTMAX	ÇKGB-YSA	SOFTMAX	ÇKGB-YSA	SOFTMAX
Karınca problemi	432	427	6	7	0	1	7	7	5	6	0	2
KKB	16	16	417	411	4	7	6	7	4	5	3	4
Sağlıklı	10	10	4	7	425	419	6	7	3	4	2	3
Kovan Hırsızlığı	4	8	5	7	3	5	434	421	0	3	4	6
Kraliçe kaybı	5	7	4	6	7	7	5	8	427	418	2	4
Varroa	3	7	4	7	5	9	4	5	3	5	431	417

çok önemli ölçümler olsada, bunlardan sadece bir tanesi değerlendirilerek iki modeli karşılaştırmak zordur. Bunları karşılaştırılabilir hale getirmenin bir yolu da kesinlik ile duyarlılığın harmonik ortalaması olan F1-skorudur ve Eş. 20 ile ifade edilmektedir.

$$\boldsymbol{Doğruluk} = \frac{TP+TN}{TP+TN+FP+FN} \tag{17}$$

$$\boldsymbol{Kesinlik} = \frac{TP}{TP+FP} \tag{18}$$

$$\boldsymbol{Duyarlılık} = \frac{TP}{TP+FN} \tag{19}$$

$$\boldsymbol{F1\ skoru} = 2x\frac{Duyarlılık\ x\ Kesinlik}{Duyarlılık+Kesinlik} \tag{20}$$

Eş. 17, Eş. 18 ve Eş. 19'daki; TP; Gerçekte doğru olan verinin, tahmin sonucunda da doğru olmasını, TN; Gerçekte yanlış olan verinin, tahmin sonucunda da yanlış olmasını, FP; Gerçekte yanlış olması gereken verinin, tahmin sonucunda doğru olmasını, FN; Gerçekte doğru olması gereken verinin, tahmin sonucunda yanlış olmasını, ifade etmektedir. Gerçekleştirilen çalışmada, arıların altı farklı sağlık durumu için elde edilen karmaşıklık matrisi, ÇKGB-YSA ve Softmax sınıflandırıcısı için ayrıntılı olarak Tablo 3'de verilmiştir.

Çalışmada Izgara Arama algoritması kullanılarak tespit edilen en iyi hiperparametrelerle, literatürdeki çalışmalardan farklı olarak, derin öğrenme algoritması ile elde edilen öznitelikler, ÇKGB-YSA modeli ve Softmax sınıflandırıcısı ile sınıflandırılmış olup, sonuçları Tablo 4'de sunulmuştur. Kıyaslama sonucunda Tablo 4'de görüldüğü üzere Softmax ile %93,07, ÇKGB-YSA ile %95,04 başarım oranı elde edilmiştir. Karmaşıklık matrisinden elde edilen kesinlik, duyarlılık, F1-skoru ve genel doğruluk oranları Tablo 5'de detaylı olarak sunulmuştur.

Tablo 4. Başarım Sonuçları (Accuracy Results)

Kullanılan Model	Başarım Sonuçları (%)
ÇKGB-YSA	95,04
SOFTMAX	93,07

Tablo 5. Karşılaştırmalı kesinlik, duyarlılık ve F1-skorları
(Comparative precision, sensitivity and F1-scores)

		Kesinlik	Duyarlılık	F1-skor
Karınca problemi	ÇKGB-YSA	0,9600	0,9191	0,9391
	SOFTMAX	0,9489	0,8989	0,9232
KKB	ÇKGB-YSA	0,9267	0,9477	0,9371
	SOFTMAX	0,9133	0,9236	0,9184
Sağlıklı	ÇKGB-YSA	0,9444	0,9572	0,9508
	SOFTMAX	0,9311	0,9353	0,9332
Kovan Hırsızlığı	ÇKGB-YSA	0,9644	0,9394	0,9518
	SOFTMAX	0,9356	0,9253	0,9304
Kraliçe kaybı	ÇKGB-YSA	0,9489	0,9661	0,9574
	SOFTMAX	0,9289	0,9478	0,9383
Varroa	ÇKGB-YSA	0,9578	0,9751	0,9664
	SOFTMAX	0,9267	0,9564	0,9413

4. SONUÇLAR (CONCLUSIONS)

Doğal yaşamda önemli bir konuma sahip bal arılarının sağlık ve yaşam ortamlarının denetlenmesi oldukça önemli bir araştırma alanıdır. Bu nedenle Türkiye'de de son derece hayati öneme sahip olan arıcılığın geliştirilmesine katkıda bulunmak için bölgesel bazda arı izleme istasyonlarının oluşturulup, ülkesel çapta bir veri tabanının hazırlanması elzemdir. Hazırlanacak veri tabanına bilgi aktarmak için de kovanlardaki arı hareketlerinin ve arı hastalıklarının izlenmesi gerekmektedir. Gerçekleştirilen çalışma ile görüntü bazlı çalışacak hastalık izleme sistemlerinde kullanılmak üzere hibrit bir derin öğrenme algoritması geliştirilmiştir. Geliştirilen sistemde derin öğrenme algoritmasının güçlü yanlarından olan öznitelik çıkarma özelliği ile ÇKGB-YSA'nın güçlü yanlarından sınıflandırma özelliği birleştirilmiştir.

Bunun ile birlikte normal bir dağılıma sahip olan değerlerde agresif olarak meydana gelen artışları tespit etmek için gerekli özniteliklerin çıkarılmasını sağlamak amacıyla, sistem iki farklı koldan öznitelik çıkartma işlemi gerçekleştirmektedir. Sistem bu yapısı ile de temel derin öğrenme algoritmasından veya evrişimsel sinir ağından ayrılmaktadır. Yapılan deneysel çalışmalar neticesinde altı farklı arı hastalığı için %91,33 ile %96,44 arasında kesinlik, %89,89 ile %97,51 arasında duyarlılık, %91,84 ile %96,64 arasında F1-skorları elde edilmiştir. Tüm sistemin genel başarım oranı ise %95,04'dür. Elde edilen bu sonuçlar yapay zeka yöntemlerinin arıların doğal yaşam kalitesinin denetimi için kullanılabileceğini göstermektedir.

Geliştirilen sistem öznitelik çıkarma ve sınıflandırma olmak üzere iki bölümden oluşmaktadır. Öznitelik çıkarma işlemi 97,45 dakika sürerken, sınıflandırma eğitimi 62,35 saat sürmüştür. Sistemin sınıflandırma katmanında bulunan ÇKGB-YSA ile genel derin öğrenme algoritmalarında kullanılan Softmax sınıflandırıcısı kıyaslandığında, aynı veri seti üzerinde Softmax sınıflandırıcısının kullanıldığı çalışmada %92,42 [38] ve Izgara Arama algoritması sonucunda elde edilen hiperparametreler ile Softmax sınıflandırıcısı uygulandığında %93,07 başarım oranı yakalanabilirken, geliştirilen sistem ile %95,04 başarım oranı elde edilmiştir. Bu nedenle arı hastalıklarının sınıflandırılacağı sistemlerde, doğru sınıflandırma oranının zamandan çok daha ön planda olduğu durumlarda, geliştirilen algoritmanın kullanılması tercih sebebi olacaktır. Bunun yanında ilerleyen dönemlerde teknolojinin gelişmesine paralel olarak kullanılacak sistemlerde, geliştirilen algoritmanın başarım oranının yüksek olması nedeniyle yine tercih sebebi olacağı aşikardır.

KAYNAKLAR (REFERENCES)

1. Muz, M. N., Demir, N., Dilek, M., Küresel Arı Sağlığı ve Veteriner Hekimlik., Veteriner Farmakoloji ve Toksikoloji Derneği Bülteni, 10 (1), 24–30, 2019.
2. Tekkaya, C., Çapa, Y., Yılmaz, Ö., Biyoloji Öğretmen Adaylarının Genel Biyoloji Konularındaki Kavram Yanılgıları., Hacettepe Üniversitesi Eğitim Fakültesi Dergisi, 18 (18), 140-147, 2000.
3. Huckle, J., British Bee Journal, British Bee Publications, London, England, 1882.
4. D'Ascenzi, C., Formato, G., Martin, P., Chemical Hazards in Honey. Chemical hazards in foods of animal origin, ECVPH Food safety assurance, Volume 7, Editör: Smulders F.J.M., Rietjens I.M.C.M., and Rose M., Wageningen Academic Publishers, Nederland, 443-475, 2019
5. Richard, F. J., Aubert, A., Grozinger, C. M., Modulation of Social Interactions by Immune Stimulation in Honey Bee, Apis Mellifera, Workers., BMC biology, BioMed Central, 6 (1), 1-13, 2008.
6. Strauss, U., Human, H., Gauthier, L., Crewe, R. M., Dietemann, V., Pirk, C. W. W., Seasonal Prevalence of Pathogens and Parasites in the Savannah Honeybee (Apis Mellifera Scutellata)., Journal of Invertebrate Pathology, Elsevier, 114 (1), 45–52, 2013.
7. Larsen, A., Reynaldi, F. J., Guzmán-Novoa, E., Fundaments of the Honey Bee (Apis Mellifera) Immune System. Review., Rev Mex Cienc Pecu, 10 (3), 705–728, 2019.
8. Forsgren, E., Locke, B., Sircoulomb, F., Schäfer, M. O., Bacterial Diseases in Honeybees., Current Clinical Microbiology Reports, Springer, 5 (1), 18–25, 2018.
9. Yost, D. G., Tsourkas, P., Amy, P. S., Experimental Bacteriophage Treatment of Honeybees (Apis Mellifera) Infected with Paenibacillus Larvae, The Causative Agent of American Foulbrood Disease., Bacteriophage, Taylor & Francis, 6 (1), e1122698 1-15, 2016.
10. Kayaalp, K., & Metlek, S., Classification of Robust and Rotten Apples by Deep Learning Algorithm. Sakarya University Journal of Computer and Information Sciences, 3 (2), 112-120, 2020.
11. Kemaloglu, N., Aydogan, T., Metlek, S., Classification of Travertine Tiles with Supervised and Unsupervised Classifiers and Quality Control Algorithms, 10 (3), 221-226, 2018.
12. Kılınç, E. E., & Metlek, S., Su Altı Görüntülerinden Nesne Tespiti. Avrupa Bilim ve Teknoloji Dergisi, (23), 368-375, 2021.

13. Aktaş A., Doğan B., Demir Ö., Tactile Paving Surface Detection with Deep Learning Methods., Journal of the Faculty of Engineering and Architecture of Gazi University, 35 (3), 1685-1700, 2020.

14. Sharma, H., Zerbe, N., Klempert, I., Hellwich, O., Hufnagl, P., Deep Convolutional Neural Networks for Automatic Classification of Gastric Carcinoma Using Whole Slide Images in Digital Histopathology., Computerized Medical Imaging and Graphics, Elsevier, 61, 2–13, 2017.

15. Zhao, X., Wu, Y., Song, G., Li, Z., Zhang, Y., Fan, Y., A Deep Learning Model Integrating FCNNs and CRFs for Brain Tumor Segmentation., Medical image analysis, Elsevier, 43, 98–111, 2018.

16. Jun, T. J., Eom, Y., Kim, C., Kim, D., Tournament Based Ranking CNN for the Cataract Grading, 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 1630–1636, 23-27 July 2019.

17. Cho, G., Yim, J., Choi, Y., Ko, J., Lee, S.-H., Review of Machine Learning Algorithms for Diagnosing Mental Illness., Psychiatry Investigation, Korean Neuropsychiatric Association, 16 (4), 262, 2019.

18. Huang, H., Zhou, H., Yang, X., Zhang, L., Qi, L., Zang, A.-Y., Faster R-CNN for Marine Organisms Detection and Recognition Using Data Augmentation., Neurocomputing, Elsevier, 337, 372–384, 2019.

19. Yilmaz, O., Erturk, Y. E., Coskun, F., Ertugrul, M., Honey Bee Biology in Turkey, VII International Scientific Agriculture Symposium, Jahorina, Bosnia and Herzegovina, 2413–2418, 6-9 October 2016.

20. Bell C., Back Yard Hive. https://backyardhive.com/blogs/managing-your-top-bar-hive/bees-robbing-a-hive. Erişim Tarihi Mayıs 30, 2020.

21. Gillespie, C., What Happens When a Queen Bee Dies?. https://sciencing.com/happens-queen-bee-dies-5159216.html. Erişim Tarihi Mayıs 30, 2020.

22. Calvo, J. A., Causes and Effects of Losing a Queen Bee. https://www.osbeehives.com/blogs/beekeeping-blog/queenless-statistics-for-the-summer-of-2018. Erişim Tarihi Mayıs 30, 2020.

23. Ramsey, S. D., Ochoa, R., Bauchan, G., Gulbronson, C., Mowery, J. D., Cohen, A., Lim, D., Joklik, J., Cicero, J. M., Ellis, J. D., Varroa Destructor Feeds Primarily on Honey Bee Fat Body Tissue and Not Hemolymph., Proceedings of the National Academy of Sciences, National Acad Sciences, 116 (5), 1792–1801, 2019.

24. Hood, W. M., The Small Hive Beetle, Aethina Tumida: A Review., Bee world, Taylor & Francis, 85 (3), 51–59, 2004.

25. Ellis, J. D., Small Hive Beetle (Aethina Tumida) Contributions to Colony Losses., Honey Bee Colony Health: Challenges and Sustainable Solutions, 7 (4), 135-144, 2012.

26. Ellis, J. D., Hepburn, H. R., An Ecological Digest of the Small Hive Beetle (Aethina Tumida), a Symbiont in Honey Bee Colonies (Apis Mellifera)., Insectes Sociaux, Springer, 53 (1), 8–19, 2006.

27. Lundie, A. E., The Small Hive Beetle, Aethina Tumida., Science Bulletin. Department of Agriculture and Forestry, Union of South Africa, Pretoria, 3 (1), 220-249, 1940.

28. Ritter, W., Bee Health and Veterinarians, OIE (World Organisation for Animal Health). Paris, France, 2014

29. Schmolke, M. D., A Study of Aethina Tumida: The Small Hive Beetle., Project Report, University of Rhodesia, 178, 1974.

30. Payne, A. N., Shepherd, T. F., Rangel, J., The Detection of Honey Bee (Apis Mellifera)-Associated Viruses in Ants., Scientific reports, Nature Publishing Group, 10 (1), 1–8, 2020.

31. Sledevič, T., The Application of Convolutional Neural Network for Pollen Bearing Bee Classification, IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Vilnius, Lithuania, 1-4, 8-10 Nov. 2018.

32. König, A., IndusBee 4.0–Integrated Intelligent Sensory Systems for Advanced Bee Hive Instrumentation and Hive Keepers' Assistance Systems., Sensors & Transducers, 237 (9–10), 109–121, 2019.

33. Braga, A. R., Gomes, D. G., Rogers, R., Hassler, E. E., Freitas, B. M., Cazier, J. A., A Method for Mining Combined Data from In-Hive Sensors, Weather and Apiary Inspections to Forecast the Health Status of Honey Bee Colonies., Computers and Electronics in Agriculture, Elsevier, 169, 105-161, 2020.

34. Rodriguez, I. F. R., Automatic Video Monitoring of Honeybee Foraging Behavior Using Convolutional Neural Networks, University of Puerto Rico, Rio Piedras (Puerto Rico), 2019.

35. Mohd-Isa, W.-N., Nizam, A., Ali, A., Image Segmentation of Meliponine Bee Using Faster R-CNN. Third World Conference on Smart Trends in Systems Security and Sustainablity (WorldS4), England, 235–238, 14-17 April 2019.

36. Lim, S., Kim, S., Park, S., Kim, D., Development of Application for Forest Insect Classification Using CNN. 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), Singapore, 1128–1131, 18-21 November 2018.

37. Bjerge, K., Frigaard, C. E., Mikkelsen, P. H., Nielsen, T. H., Misbih, M., Kryger, P., A Computer Vision System to Monitor the Infestation Level of Varroa Destructor in a Honeybee Colony., Computers and Electronics in Agriculture, Elsevier, 164, 104898, 2019.

38. Üzen, H., Yeroğlu, C., Hanbay, D., Development of CNN Architecture for Honey Bees Disease Condition. In 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya,Turkey, 1–5, 21-22 September 2019.

39. Margapuri, V., Lavezzi, G., Stewart, R., Wagner, D., Bombus Species Image Classification., arXiv preprint arXiv:2006.11374, 2020.

40. Yang, J. The BeeImage Dataset: Annotated Honey Bee Images. https://www.kaggle.com/jenny18 /honey-bee-annotated-images. Erişim Tarihi 24 Nisan 2020.

41. Lv, J.-J., Shao, X.-H., Huang, J.-S., Zhou, X.-D., Zhou, X., Data Augmentation for Face Recognition., Neurocomputing, Elsevier, 230, 184–196, 2017.

42. Khemakhem, S., Said, F. Ben, Boujelbene, Y., Credit Risk Assessment for Unbalanced Datasets Based on Data Mining, Artificial Neural Network and Support Vector Machines., Journal of Modelling in Management, Emerald Publishing Limited, 13 (4), 932-951. 2018.

43. Devarriya, D., Gulati, C., Mansharamani, V., Sakalle, A., Bhardwaj, A., Unbalanced Breast Cancer Data Classification Using Novel Fitness Functions in Genetic Programming., Expert Systems with Applications, Elsevier, 140, 112866, 2020.

44. Leng, B., Yu, K., Jingyan, Q. I. N., Data Augmentation for Unbalanced Face Recognition Training Sets., Neurocomputing, Elsevier, 235, 10–14, 2017.

45. Kilimci Z.H., Financial sentiment analysis with Deep Ensemble Models (DEMs) for stock market prediction, Journal of the Faculty of Engineering and Architecture of Gazi University, 35 (2), 635–650, 2020.

46. Hanbay K., Hyperspectral image classification using convolutional neural network and two dimensional complex Gabor transform, Journal of the Faculty of Engineering and Architecture of Gazi University, 35 (1), 443–456, 2020.

47. Yang, Z., Yu, W., Liang, P., Guo, H., Xia, L., Zhang, F., Ma, Y., Ma, J., Deep Transfer Learning for Military Object Recognition under Small Training Set Condition., Neural Computing and Applications, Springer, 31 (10), 6469–6478, 2019.

48. Grant-Jacob, J. A., Xie, Y., Mackay, B. S., Praeger, M., McDonnell, M. D. T., Heath, D. J., Loxham, M., Eason, R. W., Mills, B., Particle and Salinity Sensing for the Marine Environment via Deep Learning Using a Raspberry Pi., Environmental Research Communications, IOP Publishing, 1 (3), 35001, 2019.

49. Kuutti, S., Fallah, S., Bowden, R., Barber, P., Deep Learning for Autonomous Vehicle Control: Algorithms, State-of-the-Art, and Future Prospects., Synthesis Lectures on Advances in Automotive Technology, Morgan & Claypool Publishers, 3 (4), 1–80, 2019.

50. Metlek, S., Kayaalp, K., Basyigit, I. B., Genc, A., Dogan, H., The Dielectric Properties Prediction of the Vegetation Depending on the Moisture Content Using the Deep Neural Network Model., International Journal of RF and Microwave Computer-Aided Engineering, Wiley Online Library, 31 (1), e22496, 2021.

51. Palani, S., Liong, S.-Y., Tkalich, P., An ANN Application for Water Quality Forecasting., Marine pollution bulletin, Elsevier, 56 (9), 1586–1597, 2008.

52. Alsarraf, J., Moayedi, H., Rashid, A. S. A., Muazu, M. A., Shahsavar, A., Application of PSO–ANN Modelling for Predicting the Exergetic Performance of a Building Integrated Photovoltaic/Thermal System., Engineering with Computers, Springer, 1–14, 2019.

53. Wan, J., Li, S., Modeling and Application of Industrial Process Fault Detection Based on Pruning Vine Copula., Chemometrics and Intelligent Laboratory Systems, Elsevier, 184, 1–13, 2019.

54. Buchanan, B. G., Wilkins, D. C., Readings in Knowledge Acquisition and Learning: Automating the Construction and Improvement of Expert Systems, Morgan Kaufmann Publishers Inc., Massachusetts USA, 1993.

55. Michalski, R. S., Toward a Unified Theory of Learning: Multistrategy Task-Adaptive Learning, Morgan Kaufmann, Massachusetts, USA, 1993.

56. Fukunaga, K., Introduction to Statistical Pattern Recognition, Elsevier, Academic Press, USA, 1972.

57. Öztemel, E., Yapay Sinir Ağları, Papatya Yayıncılık, Ankara Türkiye, 2006.

58. Koptur, M., Yapay Sinir Ağları ve Derin Öğrenme – 3. https://makineogrenimi.wordpress.com/2017/07/18/yapay-sinir-aglari-ve-derin-ogrenme-3/. Erişim Tarihi 30 Mayıs 2020.

59. Emeksiz, C., Doğan, Z., Gökrem, L., Yavuz, A. H., Tokat Bölgesi Rüzgar Karakteristiğinin Istatistiksel Yöntemler ile İncelenmesi., Politeknik Dergisi, 19 (4), 481–489, 2016.

60. Metlek, S., Özkan, T., Analysis of Perceived Service Quality and Customer Satisfaction in the Aviation Sector with Artificial Neural Networks. Techno-Science 2nd Internatioanl Conference on Technology and Science, 583–864. Burdur, Türkiye, 23-25 Eylül 2018.

61. Gülcü, A., Kuş, Z., Konvolüsyonel Sinir Ağlarında Hiper-Parametre Optimizasyonu Yöntemlerinin Incelenmesi., Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji, 7 (2), 503–522, 2019.

62. Tanyıldızı, E., Demirtaş, F., Hiper Parametre Optimizasyonu Hyper Parameter Optimization. 1st International Informatics and Software Engineering Conference (UBMYK), Ankara Turkey, 1–5, 6-7 November 2019.

63. Zheng, A., Evaluating Machine Learning Models: A Beginner's Guide to Key Concepts and Pitfalls., O'Reilly Media, USA, 2015.

64. Alvear-Sandoval, R. F., Figueiras-Vidal, A. R., On Building Ensembles of Stacked Denoising Auto-Encoding Classifiers and Their Further Improvement., Information Fusion, Elsevier, 39, 41–52, 2018.

65. Ronao, C. A., Cho, S.-B., Human Activity Recognition with Smartphone Sensors Using Deep Learning Neural Networks., Expert Systems with Applications, Elsevier, 59, 235–244, 2016.

66. Alsheikh, M. A., Selim, A., Niyato, D., Doyle, L., Lin, S., Tan, H.-P., Deep Activity Recognition Models with Triaxial Accelerometers, Workshops at the Thirtieth AAAI Conference on Artificial Intelligence. Phoenix Arizona USA, 27-36, 12–13 February 2016.

67. Erfani, S. M., Rajasegarar, S., Karunasekera, S., Leckie, C., High-Dimensional and Large-Scale Anomaly Detection Using a Linear One-Class SVM with Deep Learning., Pattern Recognition, Elsevier, 58, 121–134, 2016.