

168303

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ROBOTLARIN YAPAY SİNİR AĞLARI İLE EĞİTİLMESİ

Elektronik ve Haberleşme Müh. Ülkü YÜCEL

**FBE Elektronik ve Haberleşme Mühendisliği Anabilim Dalı Elektronik Programında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Doç. Dr. Tülay YILDIRIM

Prof. Dr. Galip Cansever

İSTANBUL, 2005

İÇİNDEKİLER

	Sayfa
KISALTMA LİSTESİ.....	iv
ŞEKİL LİSTESİ.....	v
ÇİZELGE LİSTESİ	vi
ÖNSÖZ.....	vii
ÖZET.....	viii
ABSTRACT.....	ix
1 GİRİŞ.....	1
1.1 Robotların Tarihçesi.....	1
1.2 Önceki Çalışmalar.....	3
2. YAPAY SİNİR AĞLARI VE ÖĞRENME.....	10
2.1 Yapay Sinir Ağlarının Genel Tanımı ve Öğrenme Kavramı.....	10
2.2 Yapay Sinir Ağlarının Yapısı ve Çalışma Prensibi.....	10
2.3 Çalışmada Kullanılan Yapay Sinir Ağı Modelleri.....	14
2.3.1 Çok Katmanlı Algılayıcı (MLP).....	14
2.3.2 Kendi Kendini Organize Eden Harita (SOM).....	17
3. SİSTEMİN TASARIMI.....	20
3.1 Sistemin Tanımı.....	20
3.2 Sistem Donanımı.....	22
3.2.1 Mekanik Bölümü.....	23
3.2.2 Ana Kontrol Bölümü.....	23
3.2.2.1 Besleme Bölümü.....	24

	Sayfa
3.2.2.2 Haberleşme Bölümü.....	24
3.2.2.3 Motor Sürme Bölümü.....	24
3.2.2.4 Analog Sinyal Ölçme Bölümü.....	25
3.2.2.5 Mikro-denetleyici Bölümü.....	26
3.2.3 Ultrasonik Mesafe Ölçme Bölümü.....	26
3.2.4 Bluetooth Haberleşme Bölümü.....	27
3.3 Sistem Yazılımları.....	28
3.3.1 Mikro-denetleyici Yazılımı.....	28
3.3.2 Haberleşme Ara Yüz Yazılımı.....	31
3.3.3 Matlab Uygulama Yazılımı.....	32
 4. SİSTEMİN GERÇEKLEŞTİRİLMESİ.....	 35
4.1 Robot.....	35
4.2 Labirent.....	35
4.3 Robotun Çalışma Algoritması.....	36
4.3.1 Çevresel verilerin ölçülmesi.....	37
4.3.2 Robotta engellerden uzaklaşma.....	37
4.3.3 Robotun Hareket Etmesi.....	39
4.3.4 Yapılan Hareketin Doğruluk Kontrolü.....	39
4.4 Uygulama.....	39
 5. SONUÇLAR ve ÖNERİLER.....	 42
 KAYNAKLAR.....	 43
 EKLER.....	 44
EK1a Ana Kontrol Kartı Devre Şeması.....	45
EK1b Ana Kontrol Kartı Malzeme Listesi.....	46
EK1c Ana Kontrol Kartı Baskılı Devre Yerleşim Şeması.....	47
EK2a SOM Yapay Sinir Ağı Modelini Oluşturmak için Kullanılan Veri Kümesi.....	48

EK2b	SOM Yapay Sinir Ağı Modelini Oluşturan Verilerin Test Verisi Olarak Kullanılması ve Sonuçları.....	49
EK2c	SOM Yapay Sinir Ağı Modeline Giriş Olarak Verilen Test Verisi ve SOM Ağının Sonuçları.....	50
EK2d	Robot Tarafından Oluşturulan MLP Eğitim Veri Kümesi.....	52
EK2e	Eğitilen MLP Ağı için Test Veri Kümesi.....	53
ÖZGEÇMİŞ.....		55



KISALTMA LİSTESİ

ADC	Analog Digital Converter
AT	Attention (modem)
IY	Işık Yoğunluğu
LVQ	Learning Vector Quantization
MLP	Multi Layer Perceptron
NM	Nesnelere Mesafe
SOM	Self-Organizing Maps



ŞEKİL LİSTESİ

	Sayfa
Şekil 1.1	Böcek robot4
Şekil 1.2	Kullanılan hopfield ağı.....5
Şekil 1.3	Bir odanın topoloji modeli.....8
Şekil 1.4	MLP (a) ve LVQ (b) yapay sinir ağı modelleri ile yapılmış sınıflandırma.....8
Şekil 2.1a	Biyolojik sinir hücre yapısı – 1.....11
Şekil 2.1b	Biyolojik sinir hücre yapısı – 2.....11
Şekil 2.2	Yapay sinir hücresinin yapısı.....12
Şekil 2.3	Yapay sinir ağı örneği.....13
Şekil 2.4	Çok katmanlı algılayıcı yapısı.....15
Şekil 2.5	SOM ağ yapısı.....18
Şekil 3.1	Sistem yapısının genel diyagramı.....20
Şekil 3.2	Gezgin robotun resimleri.....22
Şekil 3.2	Gezgin robotun donanım blok diyagramı.....22
Şekil 3.4	Ana kontrol kartı.....23
Şekil 3.5	Motor sürmede kullanılan transistör köprüsü.....25
Şekil 3.6	LDR ile ışık şiddetini ölçen devre.....26
Şekil 3.7	Ultrasonik mesafe ölçme modülü.....27
Şekil 3.8	Bluetooth seri modülü.....28
Şekil 3.9	Mikro-denetleyici yazılımı akış diyagramı.....30
Şekil 3.10	Haberleşme ara yüz yazılımı akış diyagramı.....32
Şekil 3.11	Matlab uygulama yazılımının akış diyagramı.....34
Şekil 4.1	Kullanılan labirentin şekli ve ölçüleri.....35
Şekil 4.2	Robot çalışma algoritması blok diyagramı.....36
Şekil 4.3	SOM ağı eğitim veri kümesi oluşturulurken izlenen güzergahlar.....37
Şekil 4.4	Elde edilen SOM nöron uzayı.....38
Şekil 4.5	MLP ağı eğitilmemiş robotun labirent içinde yapmış olduğu hareket adımları.....40
Şekil 4.6	MLP ağı eğitildikten sonra robotun labirent içinde yapmış olduğu hareket adımları.....41

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 1.1 Örneklerin sayısal sonuçları.....	8
Çizelge 1.2 MLP ve LVQ yapay sinir ağı modellerinin test verileri ile elde edilmiş doğruluk oranları.....	10
Çizelge 3.1 Transistörlerin durumları ile motorun besleme uçları polarizasyonu arasındaki ilişki.....	26
Çizelge 4.1 Hareketlerin sayısal dağılımı.....	42



ÖNSÖZ

Tez çalışmamın gerçekleşmesi sürecinde yardımlarından dolayı tez danışmanım Sayın Doç. Dr. Tülay YILDIRIM' a teşekkür ederim.

Tez çalışmam boyunca her zaman yanımda olan ve desteğini esirgemeyen eşime teşekkür ederim.



ÖZET**Ülkü YÜCEL****Elektronik ve Haberleşme Mühendisliği, Yüksek Lisans Tezi**

Yapay sinir ağı modellerinden kendi kendine organize olan harita (SOM) ve çok katmanlı algılayıcının (MLP) birlikte çalışması koordine edilerek gezgin bir robot üzerindeki davranışları incelenmiştir. Hareket kabiliyeti yüksek, üzerinde ultrasonik ve ışık şiddeti algılayıcıları olan, kablosuz Bluetooth teknolojisi ile haberleşebilen, mikro-denetleyici kontrollü bir gezgin robot tasarlanmıştır. Robotun bir labirent içerisinde ışık kaynağı ile belirlenen hedefe yapay sinir ağları yardımı ile duvarlara çarpmadan ve en kısa sürede gitmesi planlanmıştır.

Kullanılan yapay sinir ağı modellerinden SOM duvarlardan kaçınma amaçlı, MLP ise robotun hedefe doğru hareket edeceği yönün belirlenmesi için kullanılmıştır. Yapay sinir ağı modelleri kişisel bilgisayarda, MATLAB araç kutusu kullanılarak oluşturulmuştur. Yapay sinir ağı eğitilmeden önce ve eğitildikten sonra robotun hedefe ilerleyişi incelenmiş ve karşılaştırılmıştır.

Anahtar Kelimeler: Robot, kendi kendine organize olan haritalar (SOM), çok katmanlı algılayıcı (MLP), Bluetooth, ultrasonik sensör, yapay sinir ağları.

ABSTRACT

Ülkü YÜCEL

Electronics & Comm. Engineering, M.S. Thesis

Cooperative work of self-organizing maps (SOM) and multi layer perceptron (MLP), which are kinds of neural network models, on a mobile robot was examined. A microprocessor controlled mobile robot that has high movement capacity, ultrasonic and light intensity sensors on it has wireless communication feature using Bluetooth was built. Mobile robot tracking in a maze to a target signed with a light source was planned in a less time and with keeping away from walls using neural network models.

SOM is used to keep the robot away from walls and MLP is used to determine the movement of mobile robot to travel to the target. Neural network models were built on a personal computer using MATLAB Neural network toolbox. In this work the results of mobile robot tracking to the target before and after neural network training was examined and compared.

Keywords: Robot, self-organizing maps (SOM), multi layer perceptron (MLP), Bluetooth, ultrasonic sensors, neural network.

1.GİRİŞ

Robotların ilk tasarlanış amacı, kuşkusuz insanların yapmış oldukları işlerde insanlara yardımcı olmasıdır. Bu amaçla yapılan tasarım sürecinde robotlar, yıllar geçtikçe insanoğlunun yaptığı işe yardımcı olmanın ötesine geçerek, en az insan gücü ile en fazla iş yapabilme görevini üstlenmişlerdir. Özellikle insanların fiziksel özelliklerinin yetersiz kaldığı işlerde robotların başarılı olacağı düşünülmektedir. Bundan sonraki hedef, hiçbir insan gücüne gerek kalmaksızın birçok işin robotlar tarafından yapılması ve düşünülen robotların geliştirilmesi olduğu açıkça görülmektedir.

Düşünen, öğrenen ve kendi kendine karar verebilen robotlar üzerinde çalışmalar hızla devam ederken bunun nasıl yapılması gerektiği üzerine araştırmalar yapılmaktadır. Robotlarda düşünme, öğrenme, karar ve tepki verme ne kadar hızlı ve doğru olursa proje de o kadar başarılı olacaktır.

Robotlarda, öğrenme işleminin gerçekleştirilmesi yolunda yapay sinir ağlarının kullanımı günümüz teknolojisinde en doğru çözüm olarak durmaktadır. En güncel ve hızla gelişen bilimsel ve teknolojik konulardan olan robotlar ve yapay sinir ağlarının bir kombinasyonu ile yapılan bu çalışmada insan yapımı robotların geliştirilen yapay sinir ağı modelleri ile nasıl eğitilebileceği gerçekleştirilmiş ve irdelenmiştir.

Bu bölümde robotların tarihçesine değinilecek ve uygulamayla ilgili daha önce yapılmış olan çalışmalar sunulacaktır.

1.1 Robotların Tarihçesi

1928'de Londra'da elektrikle çalışan bir robot yapıldı. Elektrik motoru, elektromıknatıslar, makaralar, çarklar içermesine rağmen bu robot yalnızca kendi içinde hareketliydi, yani gezemiyordu, sabit bir erişim sahası ile sınırlıydı.

1930'larda uçak tasarımcıları uçaklar için otomatik pilotu tasarladılar. Aynı dönemde duvarları boyayan endüstriyel robotlar yapıldı. Bu makineler verilen bir görevi yerine getirmek için önce alıştırma ve eğitim evresinden geçiyorlar, bu evrede yaptıkları hareketlerin bilgilerini kaydediyorlar ve daha sonra bu kayıtları kullanarak, hareketleri tekrar ediyorlardı.

1940’larda Westinghouse yatay düzlemde bağımsız olarak tümüyle hareket eden iki robot yarattı. “Electro” adlı robot dans ediyor, 10’a kadar sayıyor, Westinghouse ürünlerini tanıtıyordu. Hiçbir insan müdahalesi olmadan çevreyi algılayıp, tepki vermek üzere programlanabilen ilk robot yapay zeka laboratuvarlarında algılama ve görme ile ilgili teorileri test edebilmek için tasarlanmıştır. Bunlardan biri Shannon tarafından geliştirilen labirent çözebilen fare dir. Bu fare basit bir öğrenme algoritması ile çalışıyordu.

1953 yılında Grey Walter robot bir kaplumbağa geliştirdi. Oval şekilli bu kaplumbağanın hareket etmesi ve yön değiştirmesi iki motor ile sağlanıyordu. Kaplumbağa, ufak noktasal ışık kaynaklarının yerleştirildiği karanlık bir odada ışık detektörleri ile ışığı yakalayıp, ışık şiddetine bağlı olarak ışık kaynağına yöneliyor veya kaynaktan uzaklaşıyordu. Kaplumbağa aynı zamanda enerjisi azalınca priz bulup, kendini şarj edebiliyordu. Aynı yıl Japon firması Seiko farklı saat parçasını montajını yapan minyatür bir robot geliştirdi.

1954’de George Devol ilk bilgisayar kontrollü endüstriyel robotun patentini aldı ve General Motors’ a üretim hattı için güçlü robot kolları üretmeye başladı. Böylece endüstriyel robot devrimi başlamış oldu.

1960’da Hughes Aircraft adlı uçak şirketi “Mobot”ları üretti. Mobotlar tamamen uzaktan kumandalı makinelerdi. İnşaat, kimyasal denemeler, ve nükleer reaktörler gibi, insanların bulunamayacağı ortamlarda veya yapamayacağı işlerde radyo dalgaları ve kamera yardımıyla kontrol ediliyorlardı. Aynı yıllarda General Electric tarafından tasarlanan ve ayakları üzerinde 7 km/s hızla yürüyebilen tonlarca ağırlıkta ki “Yürüyen Kamyon” bilgisayar beyinli iki ayaklı araçtır.

1968’de görme organına ve yapay zekaya sahip ilk robot, “Shakey”, Stanford araştırma merkezinde geliştirildi. Shakey, etraftaki eşyalara çarpmadan odalar arasında dolaşıyor, sesli komutlara göre kutuları dizebiliyor, hatta kutuların düzgün durup durmadığını kontrol ediyor, gerekiyorsa düzeltiyordu.

1970’de Lunokhod1, insansız bir Rus aracı, dünyadan kumanda edilerek ayın yüzeyinde keşif gezisi yaptı.

1976’da NASA Viking1 ve Viking2 araçları mars yüzeyinden örnekler topladılar. Aynı yıl Stanford Üniversitesinde “Stanford Kolu” olarak bilinen bir robot kol geliştirildi.

1977’de Stanford Araştırma Enstitüsü, çalışan bir robot görme sistemi geliştirdi. Hareketli robotlar alanında önemli bir diğer bir çalışma da Odetics şirketinin 6 bacaklı “Functionid” isimli robotudur. Bu robot bazı böceklerin ve insanların bacak yapıları incelenerek tasarlanmıştır.

1980’lerden günümüze kadar artık iyice büyümüş olan robot endüstrisi her ay yeni kurulan robot şirketleriyle canlı bir şekilde ilerlemeyi sürdürmektedir.

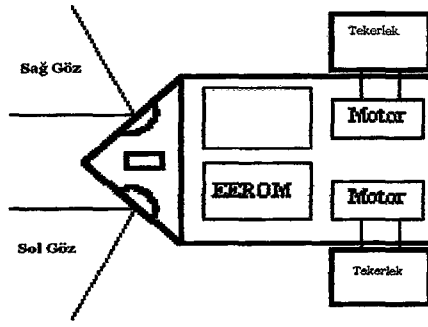
(Critchlow, 1985; Fuller, 1999)

1.2 ÖNCEKİ ÇALIŞMALAR

Bir Mobil Robotta Yapay Sinir Ağı Kontrolü (Mitchell ve Keating, 1998):

Cybernetics bölümünde yapılan çalışmada basit bir mobil robot geliştirilmiştir. Bu robotlar basit bazı sensörlerden aldıkları bilgiler doğrultusunda bulundukları çevreyi inceleyebilmektedirler. Bu robotların, tekli ve çoklu organizmaların değişik davranışlarını incelemek için test amaçlı kullanılması hedeflenmiştir. R.J.Mitchell ve D.A.Keating, tarafından çalışmada uygulanan yapay sinir ağları ve ilgili teknikleri derlenmiştir. Bu yapay sinir ağları, hem sensör bilgilerini proses etmekte hem de robotu kontrol etmek için strateji geliştirmek için kullanılmıştır. Burada robotlar, sensörleri ve kullanılan yapay sinir ağları açıklanmıştır.

Bu çalışmada basit ve etrafında bulunan cisimleri sezerek, hızlı hareketle onlardan kaçabilen küçük bir böcek robot yapılmıştır. Bu mobil robota iki adet ultrasonik sensör bulunmaktadır ve robot bazı kurallar içersinde ve sensörlerden aldığı bilgilere göre hareket etmektedir. Daha sonra bu kuralları öğrenebilmesi için robota öğrenme sistemi eklenmiştir. Çalışmanın bir sonraki aşamasında ise birleşik bir göz sistemi eklenmiş ve robotun oda içerisinde bulunduğu farklı pozisyonları öğrenmesi için yapay sinir ağı kullanılmıştır. Öğrenme özelliği, robotun gözü ve sensörleri de kullanarak içersinde bulunduğu bölgede dolaşmasını ve tanıdığı olduğu yere geri dönmesini sağlayacaktır.



Şekil 1.1 Böcek robot

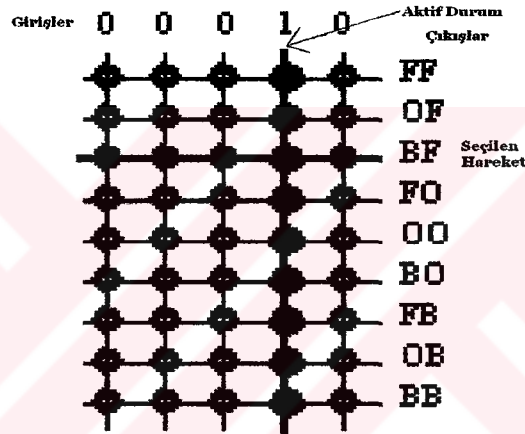
Yapılan mobil robotun ilk sürümünde mikroişlemci kullanılmayıp bunun yerine önceden içine tablo kopyalanmış bir EEROM kullanılmıştır. Bu tablo içindeki her satırda sensörlerden gelen bilgiler, motorların dönüş yönü ve hız bilgileri tutulmaktadır. Dolayısıyla robot sensörlerden aldığı bilgileri kullanarak EEROM’ da bulunan tablodan motorların dönüş yönlerinin ve hızlarının ne olması gerektiğini bulmaktadır. Robotun ilk sürümü başarılı olarak çalışmıştır ancak buna rağmen bu robot sadece önceden programlanmış rotalarda çalışmaktadır, öğrenmemektedir. Şekil1.1 ‘de bu böcek robotun yapısı verilmiştir.

Robotun ikinci sürümünde EEROM’ daki tabloyu simule eden veya robotun hareketlerini öğrenmesini sağlayacak olan bir mikroişlemci kullanılmıştır. Bu mikroişlemci sensörlerden bilgileri alarak, bu bilgileri işlemekte ve motorları sürme işlemini yapmaktadır.

Robotun öğrenme yeteneği kazanması için bebek örneğinden yola çıkılmıştır. Şöyle ki, bir bebeğe bir oyuncak gösterildiğinde o oyuncacı almak ister ama nasıl yapacağını bilemez. Bebek kollarını ve bacaklarını oynatabilmektedir. Ancak hangi kasların vücudun hangi kısımlarını kontrol ettiğini bilmemektedir. Bu sebepten rasgele hareketler yapar. Örneğin kollarını ve bacaklarını hareket ettirmeye başlar. Bir süre sonra bebek, oyuncacı almaya yaramadığını anlayarak bacaklarını oynatmayı durduracaktır ve daha sonra iki elini koordine etmeyi öğrenerek oyuncacı alacaktır. Uygun hareketi denemeler ve hatalarla öğrenmektedir. Farklı hareketler denemiş ve hareket sonucu başarılı veya başarısız olması bir sonraki hareketin ne olması gerektiği belirlemiştir. Bu düşünce ile robotlarda da çok çeşitli hareketlere, hareketi seçmek için yöntemlere, hareketi yargılamak için kriterlere, hareketlerin seçimini de yargılamak için stratejiye ihtiyacı vardır. Fuzzy automata konseptinin orijinalinden gelen bu fikirleri uygulamak için kullanılan yapay sinir ağı modelleri Hebbian öğrenme algoritması ile modifiye edilmiş Hopfield ağıdır.

Hopfield Ağı tek katmanlı ve geri dönüşümlü bir ağıdır. Proses elemanlarının tamamı hem girdi hem de çıktı elemanlarıdır. Ağı bağlantı değerleri bir giriş fonksiyonu olarak saklanmaktadır. Hopfield ağının eğitilmesinde Çok Katmanlı Ağlarda olduğu gibi iki faz vardır. Bunlar, “Ağırlıkları belirleme ve saklama fazı” ve “bilgilere ulaşma fazı”.

Yapay sinir ağına girişler ikili (ON veya OFF) değerlerdir. Bu değerler seçilmiş olan hareketi temsil ederler. Şekil 1.2’ de sadece 4.giriş ON dur. O kolonda bulunan bütün elemanlar ilgili hareketin olasılığıdır. Olası çıkışlar her bir motorun durumunu göstermektedir (B(geri), F(ileri) ve O (kapalı)). Olasılıklara dayanılarak bir hareket seçilir ve şekilde seçilen hareket BF ‘ dir. Yani sol motor geri ve sağ motor ileri hareket edecektir. Bu hareket değerlendirilir ve eğer sonuç iyi ise seçilen kolondaki hareketin olasılığı artırılır. Seçilen hareket ile verilen girişler arasında ki ilişkilendirme daha da kuvvetlendirilir.



Şekil 1.2 Kullanılan hopfield ağı

Çalışmanın sonuçları:

Kullanılan bu teknik önce simülasyon yapılarak denenmiştir. Bu şekilde çalışma çok iyi sonuç vermiştir. Daha sonra bu teknik gerçek ortamda denenmiştir. Bu durumda simülasyonda ki kadar net başarılı bir sonuç vermemiştir. Çünkü simülasyon sırasında ultrasonik mesafe sensör cevabının mükemmel olacağı kabul edilmiştir. Ultrasonik sensörlerin dik gitmediği açılarda hatalı ölçüm yapmasının, yanlış hareketlere neden olduğu görülmüştür. Bir sonraki çalışmada üçüncü bir göz eklenerek sorunun azaltılması hedeflenmiştir.

Robot gezinmesine (navigation) işbirliği yapan yapay sinir ağlarıyla basit bir yaklaşım: (Alonso vd., 2002)

Bu çalışma, iki yapay sinir ağ modeli olan SOM ve MLP' nin beraber kullanılmasını temel alan, robotun engellerden kaçarak hedefe ulaşmasını amaçlayan yeni bir nöral navigasyon (gezinme) modelini sunuyor.

Modeli düşünürken labirentte bulunan peynire ulaşmaya çalışan fare modellemesinden yola çıkılmıştır. Koku kaynağı (peynir) hedef, fare olarak robot ve labirent çözülmesi gereken harita olarak düşünülmüştür. Fare'ye bakıldığında 2 ana bilgi kaynağı bulunmaktadır. 1.si görüntü sensörleri (gözler) yardımı ile alınan labirent duvarlarını içeren veri, 2.si koku sensörleri yardımı ile alınan peynir pozisyonuna göre labirentte yayılan kokudur.

Robot üzerindeki sensörler en yakın obje (duvarlar) ile koku yayılımı ile ilgili bilgileri vermektedir. Robot üzerinde çarpışmayı engellemek için mesafe ölçen 3 farklı açıda yerleştirilmiş 5 adet sensör vardır. 0° 'deki sensör ileri, $\pm 45^\circ$ deki sensörler sol ve sağ, $\pm 90^\circ$ deki sensörler sol ve sağ taraflardaki uzaklık bilgilerini alırlar. Robot üzerinde aynı zamanda labirent üzerinde gitmesi gereken noktayı belirleyen cisimden yayılan kokuyu algılaması içinde bir koku sensörü vardır.

Robot yapı olarak 3 modülden oluşmaktadır. Mesafe ölçme modülü, hareket seçme modülü ve karar verme sistemi.

Mesafe ölçme modülü, iki farklı mesafe için ölçüm yapar. Bunlar, engellere karşı olan mesafe ve hedefe karşı olan mesafe.

Engellere karşı olan mesafe için robotun t anında bulunduğu pozisyonda 5 mesafe sensöründen elde edilen bilgiler, sensörlerin ölçebileceği tüm olası mesafe vektörlerini temsil eden 5 bileşen vektör ile önceden eğitilmiş olan iki boyutlu SOM üzerine yansıtılır. Bu durumda SOM ağındaki kazanan nöron, robotun ideal durumunu yani engel olmayan durumu göstermektedir. Topolojik mesafe (SOM içindeki konumda), robotun t anında ki pozisyonuyla aktif olan nöron ile kazanan nöron arasındaki değerlere bağlı olarak hesaplanır. Bir sonraki harekette bu mesafenin küçülmesi engelden uzaklaşıldığını gösterir.

Hedefe olan bağıl mesafe, art arda yapılan hareket sonucunda iki ayrı pozisyondan alınan koku yoğunluğunun eğim farklarını bularak ölçülür.

Hareket seçme modülünde, kendini yenileyen MLP yapay sinir ağı kullanılmıştır. Giriş vektörleri; 5 sensörden alınan mesafe verileri, koku sensörü verisi ve ağ çıkış vektörlerinden oluşmaktadır. MLP cevabı ise, 4 muhtemel hareketi kodlayan iki binary çıkıştan oluşmaktadır (Sağa Dön, Sola Dön, İleri Doğru ve Geriye Doğru). Ağ çıkışı, yapılması önerilen hareketin doğru veya yanlış olduğuna karar verecek olan modüle yollanır.

Karar verme modülünde, yapılması önerilen hareketin bilgisi ağdan çıktıktan sonra hedefe olan bağıl mesafe ve engele olan mesafe hesaplanır. Hareketin doğru olduğuna her iki değerin 0'dan büyük veya eşit olması durumunda karar verilir. Aksi takdirde robot bazı rasgele hareketler yapar (koku olarak az yoğunluklu bölgeden çok yoğunluklu bölgeye geçmek gibi).

Robotun hareketlerini doğrulamak için 3 ayrı harita örneği ile denenmiştir. Çizelge 1.1 hareket seçme modülünü eğitmeden öncesi ve sonrası için olan sayısal verileri içermektedir. Kolon 1, 4 ve 7 yani adımlar kolonlarında başlangıçtan bitişe kadar yapılan tüm doğru yada yanlış hareket sayısını, kolon 2, 5 ve 8 yani hatalar kolonlarında hatalı hareket sayıları görülmektedir. Dönme hareketindeki 45°'lik adımlardan kaynaklanan kısıtlamalar doğru olmayan hareketlerin sayısını artırmıştır. Kolon 3, 6 ve 9 eğitme algoritmasının gradient değerlerini göstermektedir.

	ÖRNEK 1			ÖRNEK 2			ÖRNEK 3		
	Adımlar	Hatalar	Gradient	Adımlar	Hatalar	Gradient	Adımlar	Hatalar	Gradient
Eğitilmemiş	144	64		303	186		233	78	
Eğitilmiş	65	2	0.09706	126	29	0.14921	154	11	0.20393

Çizelge 1.1 Örneklerin sayısal sonuçları

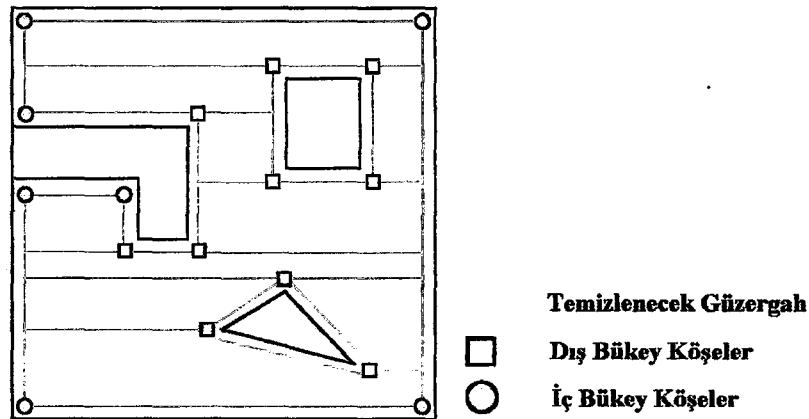
Bu sonuçlar simülasyon sonuçlarıdır. Çalışma yayınlandığı dönemde gezinme modeli, Motorola 68HC11 8-bit mikro denetleyici içeren küçük bir robot üzerinde uygulanılmaya çalışılmaktadır. Bu mikro denetleyici aldığı sensör bilgilerini bilgisayara iletecek, bilgisayar ise üzerinde bulunan yapay sinir ağı programı sayesinde robotun bir sonraki hareketini hesaplayıp mikro denetleyiciye iletecektir. Bu çalışmadan sonra hedeflenen yapay sinir ağı yapısının tamamının mikro denetleyici içersinde oluşturmaktır.

Elektrikli süpürge görevi gören robotlarda yapay sinir ağları ile dönüm noktası algılama (Wong vd., 2001):

Bu çalışmada dönüm noktası algılama işlemi için gerçek bir robotta iki çeşit yapay sinir ağı modeli (MLP ve LVQ) eğitilmiş ve test edilmiştir. Bu uygulama elektrikli süpürge görevi gören robotlar için tasarlanmıştır. Bu robot için bir oda içerisindeki eşyaların konumları ve çevresi bir bilinmeyendir. Bu uygulamada robotun eşya konumlarını ve duvarları tanınması topoloji modeli ve sınır doldurma algoritması ile sağlanır. Topoloji modeli robot bulunduğu çevreyi temizlerken oluşturulur. Bu topoloji modelinin uyarlanması, dönüm noktalarının güçlü ve doğru algılanmasına bağlıdır. Robotun seçilen her iki yapay sinir ağı modeli için de dönüm noktalarını başarılı bir şekilde algıladığı görülmüştür.

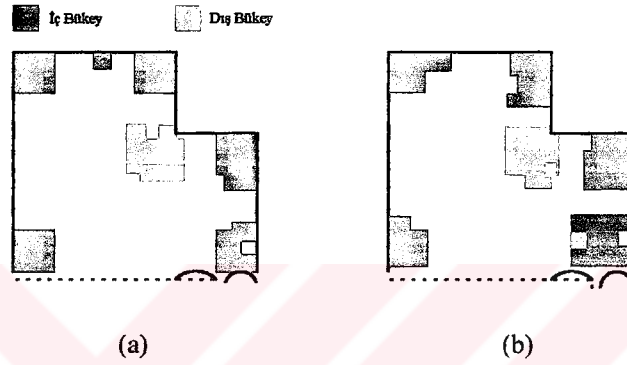
Dönüm noktaları düğüm noktalarını ve düğüm noktalarının birleştiği yerlerde köşeleri oluşturarak bir topoloji modelini meydana getirirler. Bu çevresel ifade insanların konum ifade etme biçimine çok benzemektedir. Örneğin marketin konumunu tanımlarken kitapçının sağında ve restoranın karşısında gibi ifadeler kullanırız.

Bu topoloji modelinde, temizlenmesi gereken çevre, iç bükey ve dış bükey köşeler düğüm noktaları olarak alınarak tanımlanmaktadır. Bu düğüm noktaları birbirlerine robotun güzergahı dahilindeki yollar ile bağlanmıştır. Şekil 1.3'de tipik bir odanın bir topoloji modeliyle nasıl ifade edildiği görülmektedir. Bu model, dönüşlerin düğüm noktası ve güzergahın da köşeler olduğu tek yönlü bir grafikte saklanabilir. Robotun pozisyonu o an bulunduğu düğüm noktasıyla ve üzerinde bulunduğu köşeyle ifade edilebilir.



Şekil 2.3 Bir odanın topoloji modeli

Robot oda içersinde zikzak hareketi ile temizlik yaparken topolojiyi çıkartmaya başlar ve iç bükey ve dış bükey köşeleri ve güzergahı işaretler. Bu topolojiyi çıkartmak için geliştirilen robot üzerindeki ultrasonik sensör 360°'lik bir bölgeyi tarayarak verileri toplar ve iç bükey, dış bükey ve diğerleri olmak üzere 3 sınıfa ayırır. İki yapay sinir ağı modelinin görevi bu sınıflandırmayı öğrenmek ve yeni bir veri geldiğinde bunun hangi gruba ait olduğunu çıkarmaktır. MLP ve LVQ ile yapılan sınıflandırma sonucu Şekil 1.4'de ve test verileri ile bulunmuş doğruluk oranları Çizelge 1.2'de görülmektedir.



Şekil 1.4 MLP (a) ve LVQ (b) yapay sinir ağı modelleri ile yapılmış sınıflandırma

	MLP	LVQ
İç Bükey	% 90	% 90
Dış Bükey	% 60	% 100
Diğerleri	% 95	% 88

Çizelge 1.2 MLP ve LVQ yapay sinir ağı modellerinin test verileri ile elde edilmiş doğruluk oranları

MLP yapay sinir ağı 3 katmanlı olarak tasarlanmıştır. En küçük karesel hata, 8 saklı nöronlu, öğrenme oranı 0.4 olan MLP yapay sinir ağı modelinde 0.0955 olmuştur. LVQ ağı 40 öğrenme adımıyla eğitilmiş ve 30 nörona sahip bir ağ yapısı en iyi sonucu vermiştir.

Bu uygulama için eğitici yapay sinir ağı modelleri kullanılmıştır. Çünkü dönüm noktalarının tipleri daha önceden tanımlanmıştır. Yapay sinir ağı modelleri sensörden gelen bilgileri sınıflandırmak üzere eğitilmiştir. Yapılan çalışma sonucunda bir oda içersindeki eşyalara veya duvarlara ait dönüm noktaları ve köşelerin, ultrasonik alıcı ve yapay sinir ağı modelleri ile rahatlıkla algılanabildiği görülmüştür.

2. YAPAY SİNİR AĞLARI VE ÖĞRENME

2.1 Yapay Sinir Ağlarının Genel Tanımı ve Öğrenme Kavramı

Öğrenme kavramı, “zaman içinde yeni bilgilerin keşfedilmesi yolu ile davranışların iyileştirme süreci” olarak tanımlanmaktadır. Bundan yola çıkarak, Makine Öğrenmesi, “bilgisayarın bir olay ile ilgili bilgileri ve tecrübeleri öğrenerek, benzeri olaylar hakkında karar verebilmesi ve problemlere çözüm üretebilmesidir” denilebilir. Bilgisayarların öğrenebilmesi, ilgili olaylarla ilgili bilgiler ile donatılmasına bağlıdır.(Simon , 1983)

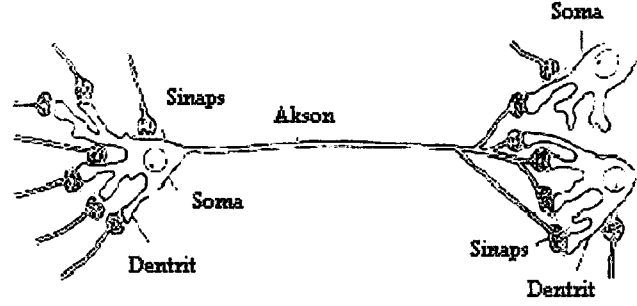
Yapay sinir ağları örneklerden öğrenmeye dayanmaktadır. Örneklerden öğrenmenin temel mantığı bir olay hakkındaki gerçekleşmiş örnekleri kullanarak olayın girdi ve çıktıları arasındaki ilişkileri öğrenmek ve bu ilişkilere göre daha sonra oluşacak olan yeni örneklerin çıktılarını belirlemektir. Burada bir olay ile ilgili örneklerin girdi ve çıktıları arasındaki ilişkinin olayın genelini temsil edecek bilgiler içerdiği kabul edilmektedir. Değişik örneklerin olayı değişik açılardan temsil ettiği varsayılmaktadır. Farklı örnekler kullanılarak olay değişik açılardan öğrenilmektedir. Burada bilgisayara sadece örnekler gösterilmektedir, başka herhangi bir ön bilgi verilmemektedir. Öğrenmeyi gerçekleştirecek yapay sinir ağı sistemi aradaki ilişkiyi kendi algoritmasını kullanarak bulmaktadır.

Yapay sinir ağları biyolojik sinir sisteminden etkilenecek geliştirilmiştir. İnsan beyninin en temel özelliği olan öğrenme fonksiyonunu gerçekleştiren bilgisayar sistemleridir. Bu ağlar birbirine bağlı yapay sinir hücrelerinden (işlem elemanlarından) oluşmaktadır. Her bağlantının bir ağırlık değeri vardır. Yapay sinir ağının sahip olduğu bilgi bu ağırlık değerlerinde saklı olup ağı yayılmıştır.

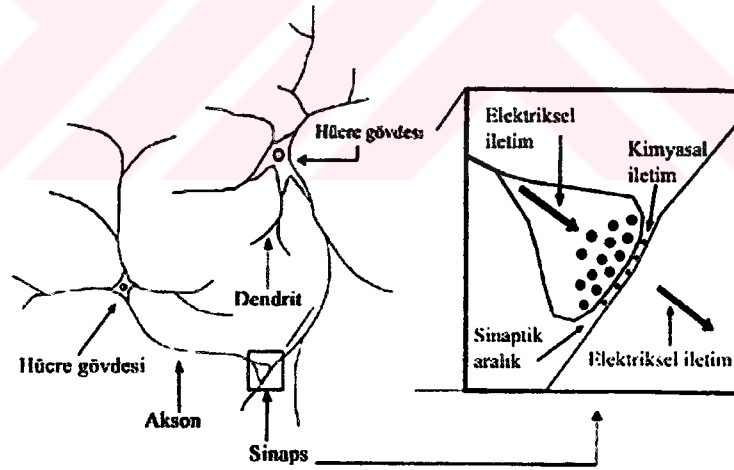
2.2 Yapay Sinir Ağlarının Yapısı ve Çalışma Prensibi

Yapay sinir ağı yapısından söz etmeden önce örnek alınan biyolojik sinir hücresi yapısından bahsedilmelidir. Şekil 2.1a ve Şekil 2.1b’ de görüldüğü gibi temel biyolojik sinir hücresi *sinaps*’lar, *soma*, *akson* ve *dendrit*’lerden oluşmaktadır. *Sinaps*’lar sinir hücreleri arasında ki bağlantılardır, bir hücrenin *dendrit*’i ile diğer hücrenin *akson*’una bağlantı noktasıdır. Bunlar fiziksel olmayıp bir hücreden diğerine elektrik sinyallerinin geçmesini sağlayan boşluklardır. Bu sinyaller hücrenin gövdesi olan *soma*’ya gider. *Soma* bunları işleme tabi tutar, sinir

hücresi kendi elektrik sinyalini oluşturur ve çıkış darbelerinin üretildiği aktif gövde uzantısı olan *akson* aracılığı ile *dentrit*'lere gönderilir. *Dentrit* diğer hücrelerden gelen bilgiyi toplayan elektriksel olarak pasif yollardır. *Dentrit*' ler bu gelen sinyalleri *sinaps*' lara göndererek diğer hücelere iletir. Bu özellikte milyarlarca sinir hücresi bir araya gelerek sinir sistemini oluşturmaktadır. Yapay sinir ağıları hücrelerin bu özellikleri temel alınarak geliştirilmiştir.

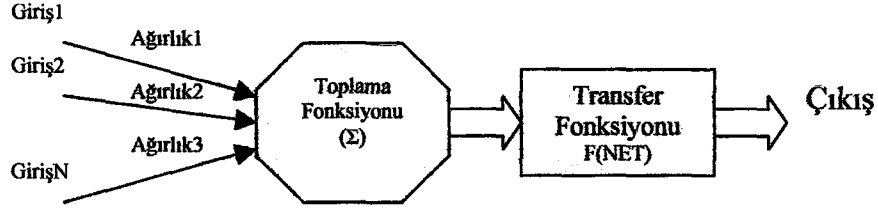


Şekil 2.1a Biyolojik sinir hücresi yapısı-1



Şekil 2.1b Biyolojik sinir hücresi yapısı -2

Biyolojik sinir ağlarının sinir hücreleri olduğu gibi yapay sinir ağlarının da yapay sinir hücreleri vardır. Bunlar “İşlem Elemanı” olarak adlandırılmaktadır. Her işlem elemanı 5 temel öğeden oluşur. Bunlar, n -vektör giriş, ağırlıklar, toplama fonksiyonu, transfer fonksiyonu ve çıkış. Şekil 2.2 ‘de bir işlem elemanının yapısı verilmiştir.



Şekil 2.2 Yapay sinir hücresinin yapısı

Girişler, ağırlık öğrenmesi istenen örnekler tarafından belirlenir. Bunlar dış dünyadan yada diğer işlem elemanlarından gelen bilgiler olabilir.

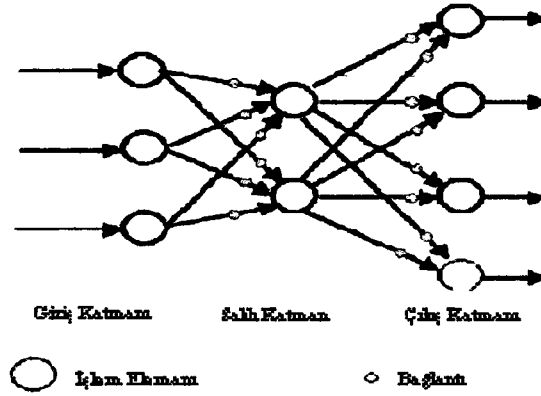
Ağırlıklar, bir işlem elemanına gelen girişin önemini ve eleman üzerindeki etkisini gösterir.

Toplama fonksiyonu, bir işlem elemanına gelen net girişi hesaplar. Değişik fonksiyonlar kullanılmaktadır, bunlardan en yaygın olanı ağırlıklı toplamı bulma fonksiyonudur.

Aktivasyon fonksiyonu, işlem elemanına gelen net girişi işleyerek, elemanın bu girişe karşılık üreteceği çıkışı belirler. Çeşitli aktivasyon fonksiyonları mevcuttur. Ağırlık elemanları farklı aktivasyon fonksiyonları kullanabilir.

Çıkış, aktivasyon fonksiyonu tarafından belirlenen çıkış değeridir. Üretilen çıkış dış dünyaya veya diğer bir işlem elemanına gönderilir. Ayrıca eleman kendi çıktısını kendisine giriş olarak da gönderebilir.

Yukarıda anlatılan yapıdaki yapay sinir hücreleri (işlem elemanları) bir araya gelerek yapay sinir ağını oluşturur. İşlem elemanları 3 katman olarak ve her katman içinde paralel olarak biraraya gelerek ağı meydana getirirler. Şekil 2.3’de bir yapay sinir ağı örneği verilmiştir.



Şekil 2.3 Yapay sinir ağı örneği

Ağ Katmanları:

- Giriş Katmanı
- Saklı Katman
- Çıkış Katmanı

Bilgiler ağı giriş katmanından iletilir, saklı katmanda işlenerek, çıkış katmanına aktarılırlar. Bilgi işleme, gelen bilgilerin ağı ağırlık değerleri kullanılarak çıkışa dönüştürülmesi şeklinde olur. Ağı girişlere karşılık doğru çıkışları oluşturabilmesi için ağırlıkların doğru değerlerde olması gerekmektedir. Ağırlıkların doğru değerlerinin bulunması işlemine ağı eğitilmesi denilmektedir. Ağırlık değerleri genellikle başlangıçta rasgele atanırlar, daha sonra eğitim sırasında her örnek ağı gösterildiğinde, ağı öğrenme kuralına bağlı olarak ağırlıklar değişir ve ağırlıkların en doğru değerleri bulunmaya çalışılır. Bu işlemler ağ eğitim setindeki tüm girişler için doğru çıkışlar elde edinceye kadar devam eder. Daha sonra test setindeki örnekler ağı verilir, eğer ağ test setindeki örneklerle doğru cevap verirse ağ eğitilmiş kabul edilir. Ağı bir olayı o olay için en uygun yapay sinir ağ modelini seçmekle mümkündür. Geliştirilmiş çok sayıda yapay sinir ağı modeli vardır. Bunlardan en yaygın olarak kullanılanları, tek ve çok katmanlı algılayıcılar (MLP), Radyal Temelli Fonksiyon Ağları (RBF), LVQ, ART ağları, SOM ve Elman ağları gibi ağlardır.

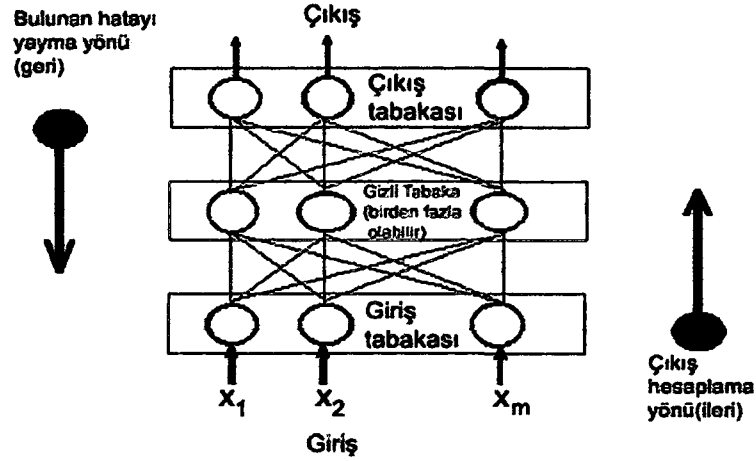
2.3 Çalışmada Kullanılan Yapay Sinir Ağı Modelleri:

2.3.1 Çok Katmanlı Algılayıcı (MLP):

Yapay sinir ağlarının günümüzde en yaygın olarak kullanılan modeli Çok Katmanlı Algılayıcı ağlarıdır. Çok katmanlı algılayıcı yapısı birçok birimin (perceptron) oluşturduğu bir kümedir. Bu algılayıcı birimler bir araya gelerek ağdaki katmanları oluşturur. Bu katmanlar da bir araya gelerek ağı oluşturur. MLP’ de üç temel katman vardır. Bunlar giriş katmanı, saklı katman ve çıkış katmanıdır. Giriş ve çıkış katmanı dışındaki tüm katmanlar saklı katman olarak adlandırılır. Giriş katmanındaki her bir işlem elemanı saklı katmandaki işlem elemanlarının hepsine, onlarda çıkış katmanındaki işlem elemanlarının hepsine bağlıdır.

Bir yapay sinir ağının öğrenmesi istenen olayların giriş ve çıkışları arasındaki ilişkiler doğrusal olmayan ilişkiler ise tek katmanlı algılayıcı ile öğrenme gerçekleştirmek mümkün değildir. Bu tür olayların öğrenilmesi için çok katmanlı algılayıcı gibi daha gelişmiş modellere ihtiyaç vardır. Çok katmanlı algılayıcıların ilk ortaya çıkışı doğrusal olmayan bir ilişki içeren XOR probleminin çözülmeye çalışılması aşamasında olmuştur. *Rumelhart* ve *arkadaşları* tarafından geliştirilen bu model özellikle, sınıflandırma, tanıma ve genelleme yapma konularında çok başarılıdır. Bu ağlar eğitici öğrenme stratejileri kullandıklarından eğitim sırasında hem girişler hem de bu girişlere karşılık ağın üretmesi gereken çıkışlar ağı öğretilir. Çok katmanlı algılayıcı modeli en küçük kareler yöntemine dayalı *Delta Öğrenme Kuralı* denilen bir öğrenme yönteminin genelleştirilmiş halini kullanmaktadır. Bu kuralın temel amacı ağın beklenen çıktısı ile ürettiği çıktı arasındaki hatayı en aza indirmektir. Genelleştirilmiş “Delta Kuralı ” iki aşamadan oluşur.

- İleri doğru hesaplama: Bu aşamada eğitim için kullanılan giriş, sisteme beslenir ve bunun sonuçları her bir katmanı geçerek çıkış katmanına kadar gelir ve girişe karşılık bir çıkış elde edilir. Bu aşama sırasında ağ üzerindeki ağırlık değerleri sabit tutulur.
- Geriye doğru hesaplama: Bu aşamada ise üretilen çıkış ile beklenen çıkış karşılaştırılarak aradaki hata geriye doğru dağıtılarak ağırlıklar değiştirilirler. Bu sırada, oluşan hatayı minimuma indirecek şekilde ağırlık değerleri güncellenir.



Şekil 2.4 Çok katmanlı algılayıcı yapısı

Şekil 2.4’de çok katmanlı algılayıcı yapısı verilmiştir.

Çok katmanlı algılayıcı ağlarının öğrenme performansını etkileyen faktörler şunlardır.

- **Örneklerin seçilmesi**, ağın performansı için çok önemli bir kriterdir. Seçilen örnekler ile oluşturulan veri tabanı problem uzayını temsil edebilecek nitelikte olmalıdır.
- **Giriş ve çıkış değerlerinin sayısal olması**: Sayısal olmayan örnekleri dönüştürmek gereklidir. Dönüştürme yöntemi ağın performansı etkileyen bir kriterdir.
- **Ağırlıkların başlangıç değerlerinin atanması**: Genel olarak ağırlıklar belirli aralıklarla atanmaktadır, bu aralık eğer büyük tutulursa ağın lokal çözümler üzerinde sürekli dolaştığı, küçük olması durumunda ise öğrenmenin geç gerçekleştiği görülmüştür.
- **Öğrenme katsayısı ve momentum katsayılarının belirlenmesi**: Öğrenme katsayısı ağırlıkların değişim oranını belirlemektedir. 0-1 aralığında olması gereken bu değer, eğer büyük seçilirse, ağın lokal çözümler arasında dolaşması ve osilasyon yaşanması söz konusu olmaktadır. Küçük değerler ise öğrenmeyi yavaşlatır. Momentum katsayısı bir önceki iterasyondaki değişimin hangi oranında yeni değişim miktarına ekleneceğini belirler. Bu değer küçük olması lokal çözümlerden kurtulmayı zorlaştırabilir. Çok büyük değerleri ise tek bir çözüme ulaşmakta sorun yaşanabilir.
- **Örneklerin ağa sunum şekli**: Örnekler ağa sıralı veya rasgele sunulabilir. Probleme göre karar verilmelidir.

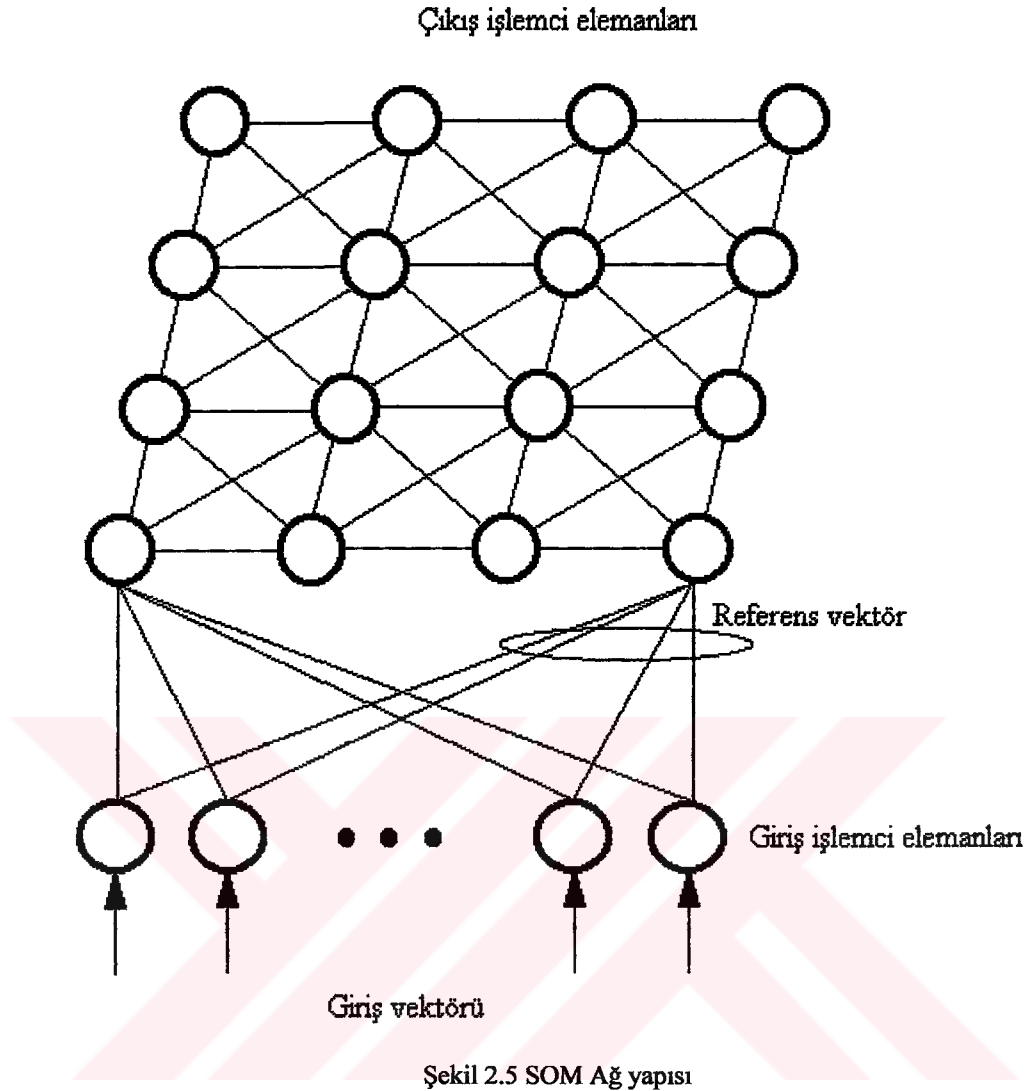
- **Ağırlıkların değiştirilme zamanı:** Ağırlıkların değiştirilmesi öğrenme kuralına göre yapılmaktadır. Problemin durumuna göre ağırlıkların ne zaman değiştirileceğine karar vermek gerekir. Doğru zamanlama ağı performansını etkilemektedir.
- **Giriş ve çıkış değerlerinin ölçeklendirilmesi:** Değerlerin ölçeklendirilmesi dağılımı düzenli hale getirdiğinden ağı performansını olumlu yönde etkiler. Bütün girişlerin belirli bir aralıkta (genellikle 0-1 aralığında) ölçeklendirilmesi hem farklı ortamlardan gelen bilgilerin aynı ölçek üzerine indirgenmesini hem de yanlış alınan örneklerin etkisini azaltılmasını sağlar. Bazı durumlarda çıkış değerlerinin de ölçeklendirilmesi gerekebilir. Sigmoid fonksiyonu kullanılması isteniyorsa ağ çıkışı 0-1 aralığında olacaktır, bu durumda karşılaştırmının doğru sonuç vermesi için beklenen çıkış değerlerinin de 0-1 aralığında olması gerekecektir, bu problemin çözümü için çıkış değerlerine ölçeklendirme yapılmalıdır. Değişik ölçeklendirme yöntemleri kullanılabilir, en yaygın giriş vektörünün normalize edilmesidir.
- **Durdurma kriterlerinin belirlenmesi:** Ağı gereğinden fazla eğitilmesi, ağırlıkların doğru değerleri bulunduktan sonra daha fazla değişmesine neden olur. Bu da ağı performansını olumsuz etkileyen bir faktördür. Ağı eğitimin ne zaman durdurulması gerektiği önemli bir karardır. İki türlü durdurma kriteri kullanılmaktadır. Bunlar: Hatanın belli bir değerin altına düşmesi durumuna durdurma ve ağı belirli bir iterasyon sayına ulaşması sonucu durdurma. Probleme bakarak uygun olan kriter seçilmelidir.
- **Saklı katman sayısı ve işlem elemanı sayılarının belirlenmesi:** Saklı katman sayısı ve işlem elemanı sayısı ağı performansını önemli derece etkilemektedir. Fakat net bir belirleme yöntemi yoktur. Probleme göre deneme yanılma yöntemi ile en iyi performans veren sayılar belirlenir.(Rumelhart vd., 1986; Haykin, 1998; Öztemel, 2003)

2.3.2 Kendi Kendini Organize Eden Harita (SOM):

SOM (Self-Organizing Maps) ağıları Kohonen tarafından geliştirilmiş, eğiticiye çalışabilen, rekabete dayalı öğrenme (competitive learning) ile giriş verisinin yapısına göre kendini organize edebilen yapılardır (Kohonen, 1989). Bu ağların giriş vektörlerini sınıflandırma ve giriş vektörlerinin dağılımını öğrenebilme yetenekleri çok yüksektir. SOM ağının en temel özelliği yukarıda da bahsedildiği gibi olayları öğrenmek için bir eğiticiye veya ağın üretmesi gereken çıkışların ağa söylenme zorunluluğunun olmamasıdır.

SOM, giriş paternlerini topolojik düzendeki iki boyutlu özellik haritasına aktarır. Giriş uzayındaki birbirine benzer verileri gruplar, bunları iki boyutlu özellik haritasında yakın birimlere atar (Ham, 2001). Şekil 2.5' de özellik haritalandırma yapısı görülmektedir (giriş vektör setinin iki boyutlu SOM haritasına aktarılması).

SOM ağları yapısal olarak diğerlerinden farklıdır. Ağ bir giriş ve bir de çıkış olmak üzere iki katmandan oluşur. Çıkış katmanı iki boyutlu bir düzlemi göstermektedir. İşlem elemanları bu düzlem üzerinde dağılmış vektörleri göstermektedir. Çıkıştaki her işlem elemanı, bütün giriş işlem elemanlarına bağlıdır. SOM ağları yarışmayı kazanma ve kazanan işlem elemanının 1 diğerlerinin 0 değerini alması ilkesine dayanmaktadır. Ağa bir giriş verildiğinde çıkış uzayında yarışmayı kazanan ve onun etrafındaki komşuları eğitim sırasında ağırlıklarını değiştirmektedir.



SOM ağının eğitim algoritmasının adımları şu şekilde özetlenebilir. (Schalkoff, 1997)

Örnek setinden bir örnek ağa verilir.

1. Ağırlık vektörlerine rasgele küçük değerler verilir.
2. Tüm işlem birimlerinin yeni girişe olan uzaklığı hesaplanır. (Genelde uzaklık hesaplaması için öklid uzaklığı kullanılır)

$$d_j = \| i(k) - m_j(k) \|$$

$i(k)$: Giriş

$m_j(k)$: j işlem biriminin ağırlık vektörü

d_j : Girişin j işlem elemanına uzaklığı

3. Kazanan işlem elemanı seçilir (örneğin c). Bu girişe en yakın ağırlık vektörüne sahip olan işlem elemanıdır.

$$c = \min(d_j)$$

4. Kazanan işlem elemanının ve onun komşularının ağırlık vektörleri güncelleştirilir. Bu yaklaştırma, kazanan işlem elemanı için en fazla, bu işlem elemanından uzaklaştıkça daha az yapılacaktır.
5. İterasyon yeterli ise işlem bitirilir, değil ise birinci adıma dönülür.

Öğrenme ilerledikçe komşuların sayısı azalmakta ve öğrenmenin sonunda sadece kazanan işlemci elemanının ağırlığı güncellenmektedir.

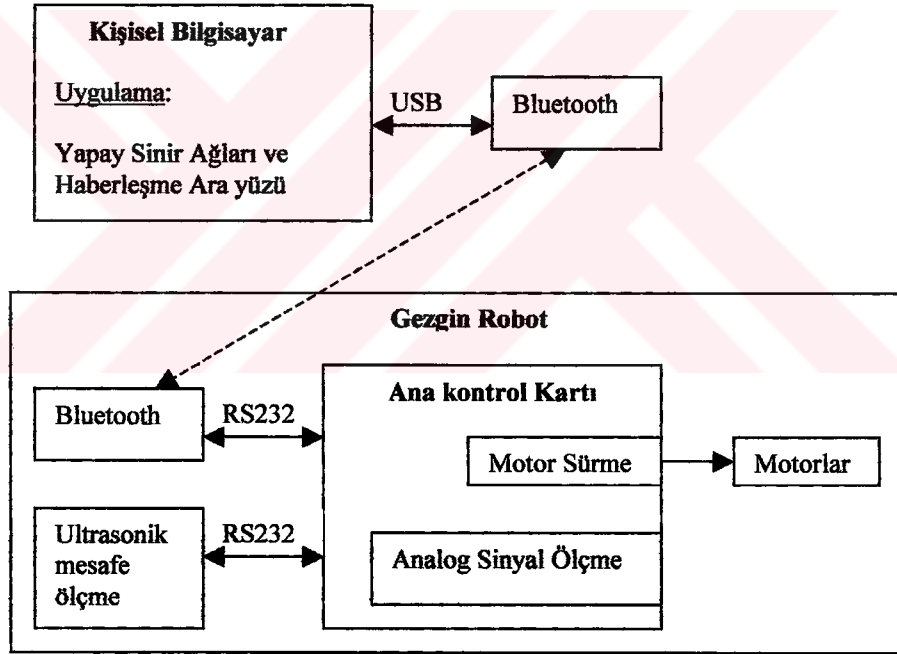
SOM ağı birçok alanda kullanımı başarılı sonuçlar vermektedir. Özellikle beklenen çıkışın belirlenemediği problemlerde ve sınıflandırma işlemlerinde tercih edilmektedir. SOM ağının pratik olarak en önemli uygulama alanları arasında

- Veri analizi
- Şekil tanıma
- Ses analizi
- Tıbbi teşhisler
- Kontrol
- Robot uygulamaları sayılabilir.

3. SİSTEMİN TASARIMI

3.1 Sistemin Tanımı

Robotlarda yapay sinir ağları ile öğrenme uygulamasını gerçeklemek için üzerinde çeşitli algılayıcılar bulunan, hareket kabiliyeti yüksek, doğru akım motorları yardımı ile hareket edebilen, kişisel bilgisayar ile kablosuz haberleşebilen bir robot tasarlanmıştır. Robot, kişisel bilgisayar ile etkileşimli çalışarak MATLAB®’da tasarlanmış yapay sinir ağı modelleri yardımı ile hareket etmektedir. Robotun içerisinde hareket edebileceği nitelikte bir labirent oluşturulmuştur. Yapılan çalışmada amaç, bir labirent içindeki robotun duvarlara çarpmadan ilerlemesi ve belirlenen hedef noktaya en kısa yoldan ve dolayısı ile en kısa sürede ulaşmasının sağlanmasıdır. Robot için hedef nokta belirteci olarak ışık kaynağı kullanılmıştır. Sistem yapısının genel diyagramı Şekil 3.1’ de görülmektedir.



Şekil 3.1 Sistem yapısının genel diyagramı

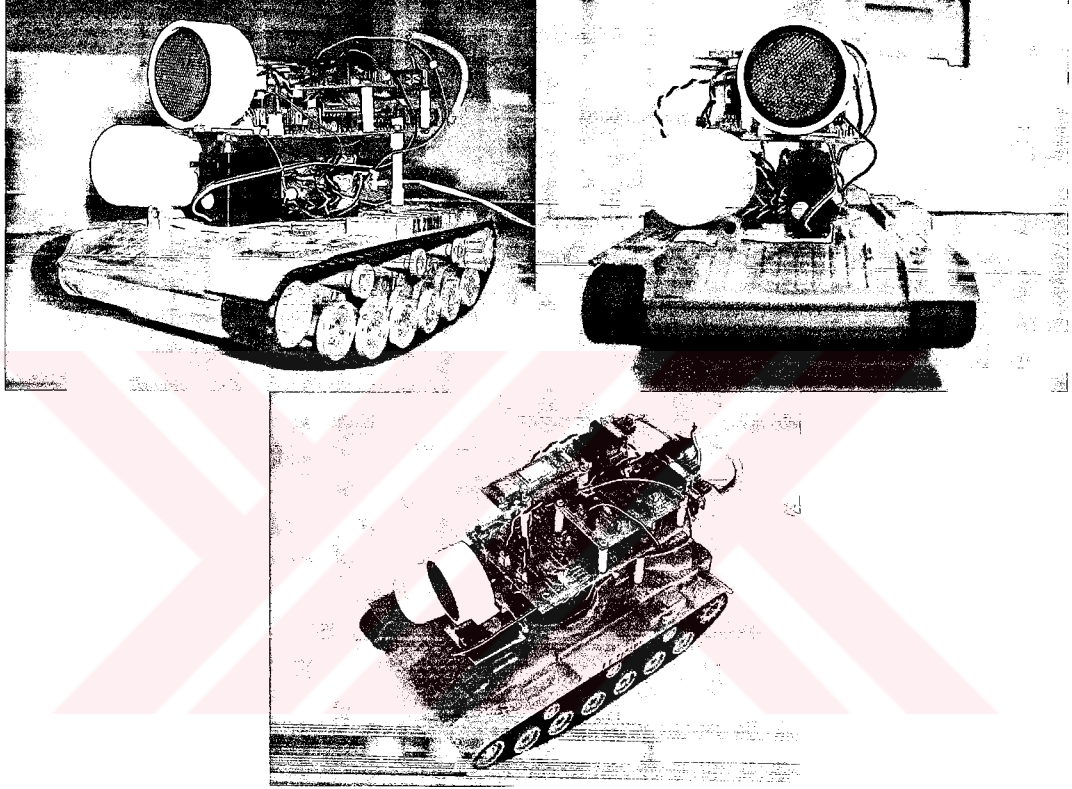
Bu çalışmada standart yapay sinir ağı modelleri olan MLP (Multi Layer Perceptron) ve SOM (Self-Organizing Map)’un biyolojik yapılardan esinlenerek beraber kullanımının sağladığı yeteneklerden faydalanılmıştır.

Robottan alınan ve labirentin duvarlarına olan mesafeyi gösteren algılayıcı verileri, küçük ölçütlü bir harita da diyebileceğimiz SOM ağına aktarılır. SOM ağ modeli robotun bulunduğu konumdaki ışık şiddetini dikkate alan MLP (Multi-Layer Perceptron) ağ modeli ile desteklenerek çalışma için uygun ağ yapısını oluştururlar.



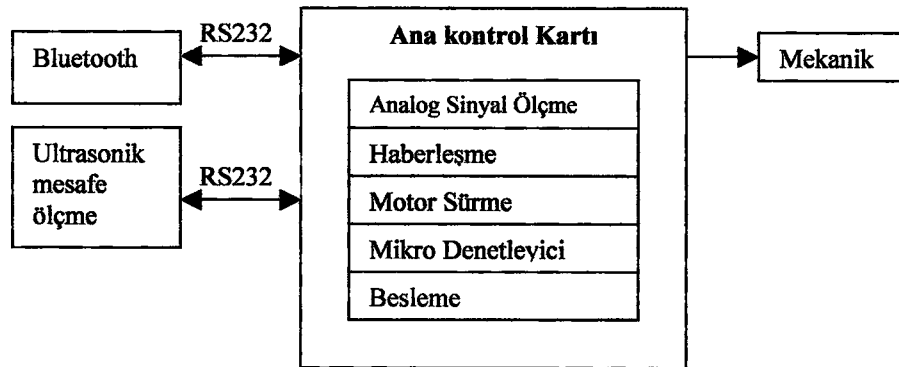
3.2 SİSTEM DONANIMI

Çalışmada kullanılan gezgin robot üstlendikleri görevler bakımından mekanik bölüm, ultrasonik mesafe ölçme bölümü, ana kontrol bölümü ve Bluetooth haberleşme bölümü olmak üzere 4 ana bölümden oluşur. Robotun genel görünümü ile ilgili resimler Şekil 3.2’de verilmiştir. Robotun fiziksel ölçüleri en:15.5 cm, boy:27 cm ve yükseklik:16 cm şeklindedir.



Şekil 3.2 Gezgin robotun resimleri

Tasarlanan robotun donanım blok diyagramı Şekil 3.3’de görülmektedir.



Şekil 3.3 Gezgin robotun donanım blok diyagramı

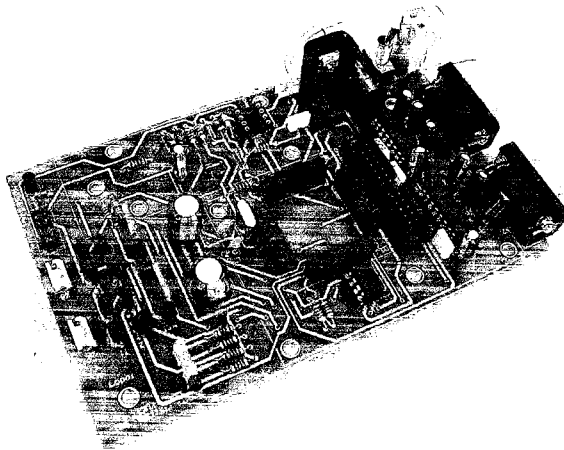
3.2.1 MEKANİK BÖLÜMÜ

Gezgin robotun yürüten aksamı, labirent içerisinde dar alanlarda rahatça manevra yapabilmesi için tank modelinden meydana getirilmiştir. Tank modeli paletli sistemden meydana gelmektedir. Sağ ve sol paletler için ayrı DC motor kullanılması robotun manevra kabiliyetini çok yükseltmiştir. Motorlar 5- 12 VDC gerilimlerde çalışmaktadır.

3.2.2 ANA KONTROL BÖLÜMÜ

Ana kontrol modülü gezgin robotun tüm işlevlerini kontrol eden bölümdür. Üzerinde NEC 78F0078 model 8-bit mikro-denetleyici bulunmaktadır. Mikro – denetleyici içerisinde C++ ile yazılmış bir alt düzey program çalışmaktadır. C++ ile tasarlanmış olan bu alt düzey program sistem yazılımı bölümünde detaylı olarak anlatılacaktır. Bu modülün görevi; kişisel bilgisayardan Bluetooth ile gelecek olan komutları işlemek, istenen verileri bilgisayara iletmek, ultrasonik alıcı modülünden mesafe bilgilerini almak, ışık yoğunluğunu ölçmek, robot hareketini sağlamak üzere motorları sürmek olarak özetlenebilir. Ana kontrol kartının devre şeması, malzeme listesi ve baskılı devre yerleşim şeması EK1’de bulunabilir.

Tasarlanan ana kontrol kartı üzerinde başlıca 5 adet bölüm vardır. Ana kontrol kartı Şekil 3.4’de görülmektedir.



Şekil 3.4 Ana kontrol kartı

3.2.2.1 BESLEME BÖLÜMÜ

Ana kontrol kartı besleme bölümü, gerilim girişini anahtarla kontrol eden ve bir sigorta ile kullanılan tüm devreleri koruma altına alan bölümdür. 12VDC gerilim girişini 5VDC'ye çevirerek tüm modüller için gerekli enerjiyi sağlamaktadır. Kullanılan LM 7805 regülatörü 12 VDC – 5 VDC regülasyonunu yapmaktadır.

3.2.2.2 HABERLEŞME BÖLÜMÜ

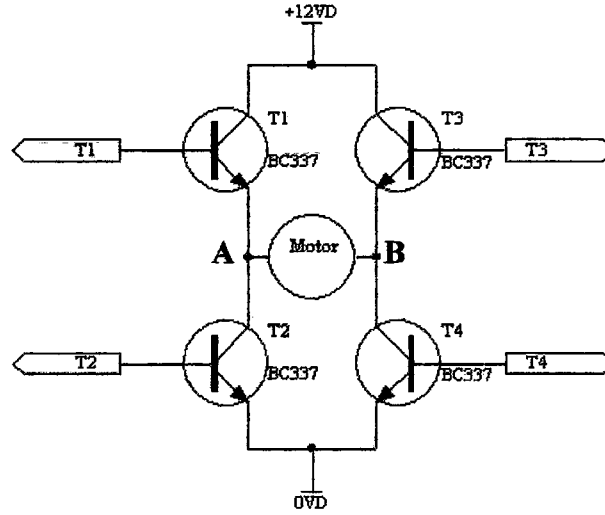
Tasarlanan ana kontrol modülünde mikro-denetleyiciye bağlı iki adet RS232 seri arabirimi vardır. Bunlardan biri BLUETOOTH modülü diğeri ise ultrasonik mesafe ölçme modülüne bağlantı için kullanılır.

Haberleşme bölümünde kullanılan MAX232 entegresi seri haberleşme sinyallerinin voltaj seviyelerinin dönüştürülmesi için kullanılmaktadır. Mikro-denetleyici 0VDC-5 VDC arasında bir sinyal çıkışı üretmektedir ancak BLUETOOTH ve Ultrasonik mesafe ölçme modüllerinin seri haberleşme sinyal seviyeleri 0 VDC– 15 VDC olduğundan dolayı MAX232 entegresi kullanılmıştır.

3.2.2.3 MOTOR SÜRME BÖLÜMÜ

Motor sürme bölümü, robotun mekanik bölümünde bulunan DC motorları sağa ve sola olmak üzere iki yönde de döndürebilen ve iki adet motora hizmet verebilen bölümdür.

Motor sürme bölümünün yapısı Şekil 3.5'de görüldüğü gibi bir transistör köprüsü şeklindedir. Bu transistörler mikro-denetleyici tarafından sürülürler ve iletimde olan transistörlere göre motorlar sağa ya da sola dönüş yaparlar. Çizelge 3.1'de transistörlerin iletim ve kesim durumları ile bağlı olan motorun besleme uçları polarizasyonu arasındaki ilişki görülmektedir.



Şekil 3.5 Motor sürmede kullanılan transistör köprüsü

T1	T2	T3	T4	A	B
0	0	0	0	0V	0V
1	0	0	1	12V	0V
0	1	1	0	0V	12V

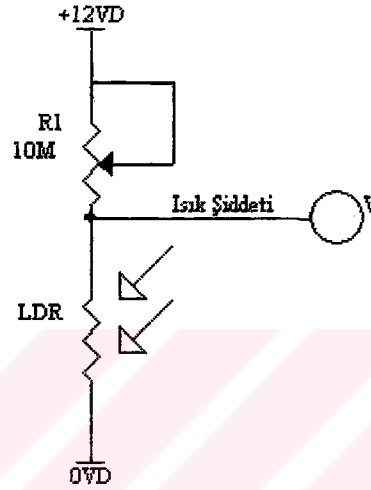
Çizelge 3.1 Transistörlerin durumları ile motorun besleme uçları polarizasyonu arasındaki ilişki

3.2.2.4 ANALOG SİNYAL ÖLÇME BÖLÜMÜ

Ana kontrol kartı üzerindeki ölçme bölümünün görevi, sisteme ait 3 adet ölçme bilgisi gerilimini mikro-denetleyicinin Analog - Dijital çevirici kısmında ölçülebilir hale getirmektir. Bundan dolayı ölçme bölümü kazanç devreleri içermektedir. Ölçme bölümünün işlediği değerler sırasıyla besleme gerilimi, robotun bulunduğu ortam sıcaklığı ve ışık yoğunluğudur. Besleme gerilimi 12VDC olduğu için bu değer mikro-denetleyicinin ölçebileceği sınırlar içinde değildir. Bu yüzden opampli kazanç devresi 12 VDC'yi 5 VDC yapacak şekilde ayarlanmıştır.

Robotun bulunduğu ortam sıcaklığını ölçmek için LM35 analog çıkışlı sıcaklık ölçme entegresi kullanılmıştır [3]. Bu algılayıcı, kılıfı üzerinde ölçtüğü sıcaklık değerini gerilim olarak çıkışına yansıtmaktadır. Her 1 C° için 10 mVDC çıkış vermektedir. Dolayısıyla bu sinyalin mikro-denetleyici tarafından algılanabilmesi için kuvvetlendirilmiştir.

Robotun labirent içerisindeki ışık yoğunluğunu, bir anlamda da hedefe olan uzaklığını ölçmek için LDR olarak bilinen ve üzerine düşen ışık şiddeti ile direnci değişen bir algılayıcı kullanılmıştır. Üzerine düşen ışık şiddeti ile direnci değişen LDR' nin seri bir dirençle kullanılması durumunda üzerine düşen gerilimde değişmektedir. Mikro-denetleyici LDR üzerine düşen gerilimi ölçerek ışık şiddetini belirlemektedir. LDR ile ışık şiddetini ölçen devre Şekil 3.6'da görülmektedir.



Şekil3.6 LDR ile ışık şiddetini ölçen devre

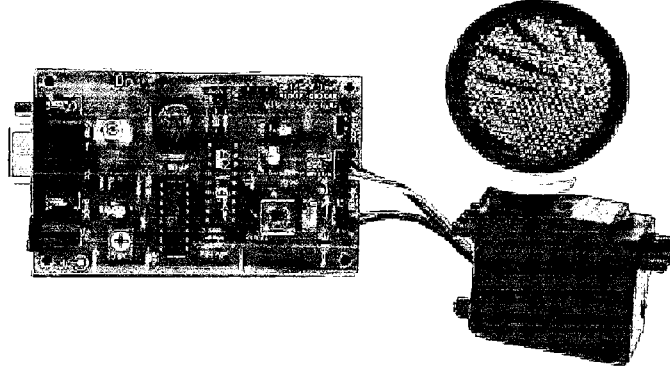
3.2.2.5 MİKRO-DENETLEYİCİ BÖLÜMÜ

Ana kontrol kartı üzerinde bulunan diğer bölümler ve donanım üzerinde yer alan modüller ile sürekli etkileşimli çalışan bölümdür. Bu bölümde kullanılan NEC-78F0078 kodlu 8 bitlik mikro-denetleyici, 8 kanal 10-bit analog-dijital çevrici girişine, 2 adet seri haberleşme kanalına, 60 Kbyte program ve 1024 byte veri belleğine, 2 adet 16 bit dahili sayıcıya sahiptir [4]. Mikro-denetleyici bölümü ile ilgili detaylı bilgi sistem yazılımı bölümünde bulunabilir.

3.2.3 ULTRASONİK MESAFE ÖLÇME BÖLÜMÜ

Ultrasonik mesafe ölçme bölümü, ses sinyalinin kaynaktan çıktıktan sonra engele çarpıp yansıtılarak tekrar kaynağa geri gelmesi ilkesine dayanarak çalışmaktadır. Ultrasonik vericiden çıkan ses dalgasının kaynaktan çıktıktan geri gelinceye kadarki geçen sürenin hesaplanması engele olan mesafenin belirlenmesinde esastır. Geçen bu zamana uçuş zamanı denir ve ses hızının bilinen değeri için engele olan mesafe hesaplanır.

Bu çalışmada kullanılan ultrasonik mesafe ölçüm modülü [2] üzerinde bir adet mikro-denetleyici, bir adet 0°-180° derece dönme özelliğine sahip servo-motor ve 0,15m ile 2,6m arasında mesafe ölçümü yapabilen ultrasonik verici-alıcı bulunmaktadır. Modül Şekil 3.7’de görülmektedir.

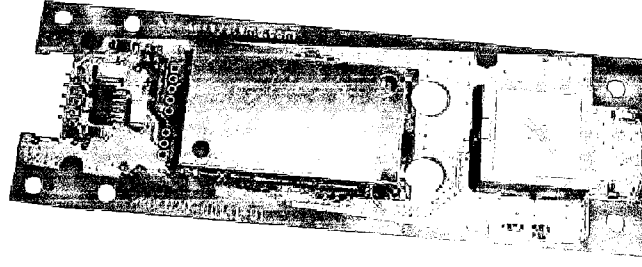


Şekil 3.7 Ultrasonik mesafe ölçme modülü

Modül RS232 seri arabirime sahiptir ve dışardan gelecek komut dizisine göre servo-motoru istenen açı değerine göre döndürür, ses kaynağından çıkışı sağlar ve yankı sonunda hesapladığı mesafe değerini 0-255 arasında bir sayı değeri olarak RS232 seri arabirimden iletir. RS232 seri haberleşme 9600bps veri aktarım hızında yapılmaktadır. Modül ölçüm yapabilmek için seri arabirimden 2-bayt gelmesini bekler. Bu baytlar sırasıyla “ T ” karakteri ve servo-motor pozisyonunu belirleyen 0-255 arasında bir açı değeridir. Bu iki bayt alındıktan sonra modül mesafe bilgisini tek bayt olarak iletir.

3.2.4 BLUETOOTH HABERLEŞME BÖLÜMÜ

Gezgin robot ile kablosuz haberleşmeyi sağlamak amacıyla TDK firmasının Bluetooth seri modülü ürünü kullanılmıştır [5]. Modül Şekil 3.8’ de görülmektedir. Bu modül seri haberleşme (RS232) ile kontrol edilebilen bluetooth teknolojisine sahip bir kablosuz modemdir. Genel olarak görevi, bluetooth kanalından gelen verileri seri haberleşme kanalına, seri haberleşme kanalından gelen verileri de bluetooth kanalına aktarmak olarak özetlenebilir.



Şekil3.8 Bluetooth seri modülü

Bu modülün tıpkı modemde olduğu gibi bir AT komut seti bulunmaktadır. Modül için gerekli ayarları yapmak, bir bağlantı kurmak veya bağlantı isteğini cevaplamak gibi işlemler bu komut seti kullanılarak yapılmaktadır.

Bluetooth modülü, kişisel bilgisayardan kablosuz olarak gönderilen komutları seri haberleşme sinyaline dönüştürerek mikro-denetleyiciye iletir ve mikro-denetleyiciden gelen ölçüm bilgilerini de kablosuz haberleşme kullanarak kişisel bilgisayara aktarır.

3.3 SİSTEM YAZILIMLARI

Gerçeklenmiş olan robotta yapay sinir ağları ile öğrenme uygulamasında Matlab Yapay Sinir Ağı araç kutusu, Microsoft Visual Basic ve NEC mikro-denetleyiciler için C++ derleyici [1] olmak üzere 3 farklı yazılım geliştirme programı kullanılmıştır. Farklı geliştirme ortamında tasarlanmış olan yazılımlar ve sistemdeki görevleri bu bölümde anlatılacaktır.

3.3.1 Mikro-denetleyici Yazılımı

Mikro-denetleyici içinde çalışan yazılım C++ ile tasarlanmış ve derleme işlemi ile makine diline çevrilmiştir. Bu yazılımın görevleri maddeler halinde şu şekilde sıralanabilir;

- Ultrasonik mesafe ölçme modülüne uzaklığı ölçülecek olan açı değerini göndermek ve ultrasonik mesafe ölçme modülünden RS232 ile gelen ölçüm sonucunu kaydetmek ve kişisel bilgisayardan bir istek geldiği durumda bu bilgileri yollamak.
- Gezgin robotun besleme gerilimini, bulunduğu ortam sıcaklığını ve ışık yoğunluğunu analog dijital dönüştürücüler yardımı ile ölçmek ve istek geldiği durumda bu değerleri bluetooth modülü ile kişisel bilgisayara yollamak.
- Eğer kişisel bilgisayardan robotun bir hareket yapması isteği gelirse istenen komutu gerçekleştirmek için motorlara hareket vermek.

- Bluetooth modülünün bağlantıları kabul edebilmesi için AT komut seti kullanarak Bluetooth modülünde gerekli ayarları yapmak.

Mikro-denetleyici yazılımının akış diyagramı Şekil 3.9' da verilmiştir.



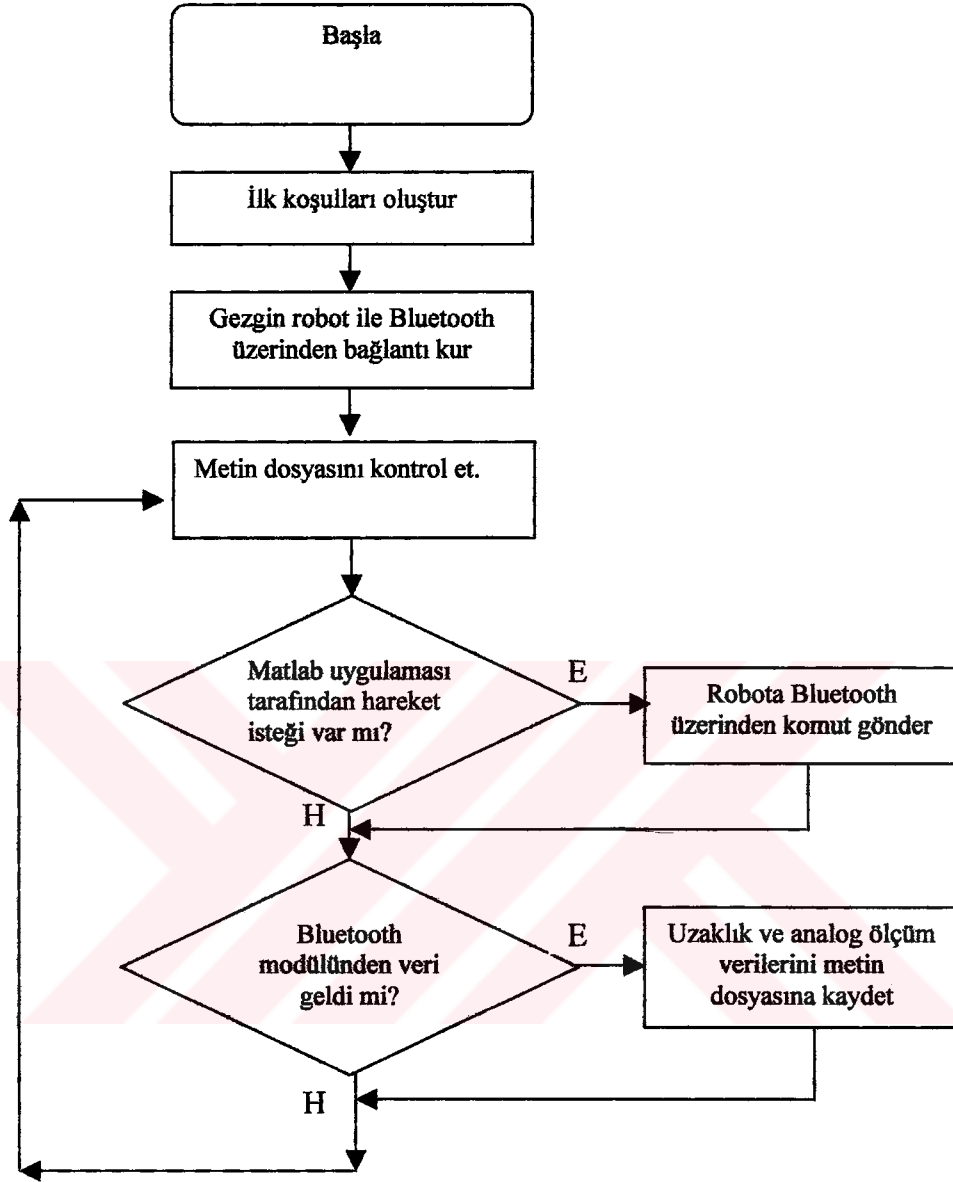
Şekil 3.9 Mikro-denetleyici Yazılımı Akış Diyagramı

3.3.2 Haberleşme Ara Yüz Yazılımı

Matlab araç kutusu kullanılarak oluşturulan yapay sinir ağı modellerinin bulunduğu kişisel bilgisayarın bluetooth üzerinden gezgin robot ile haberleşme işlemini gerçekleştirecek bir ara yüz yazılımı Visual Basic geliştirme ortamında tasarlanmıştır.

Haberleşme ara yüz yazılımının sistem içindeki görevi Matlab yapay sinir ağı uygulamasının oluşturduğu çıktıları (robotu sağa, sola, ileri, geri yönde hareket ettirmek) USB' ye bağlanmış bluetooth modülüne iletmektir. Böylece yapay sinir ağı modelinden çıkan robot hareketi bilgisi, gezgin robotun bluetooth modülüne ulaşmakta ve robot üzerindeki mikro-denetleyici yazılımı gelen isteği harekete çevirmektedir. Aynı zamanda bilgisayarın bluetooth modülüne gezgin robot tarafından analog ölçüm ve duvarlara uzaklık verileri gönderilirse bu verileri Matlab uygulamasına iletme görevini de yerine getirmektedir.

Haberleşme ara yüz yazılımı Matlab uygulaması ile veri alışverişini metin dosyaları ile yapmaktadır. Haberleşme ara yüz yazılımının akış diyagramı Şekil 3.10'da görülmektedir.



Şekil 3.10 Haberleşme ara yüz yazılımı akış diyagramı

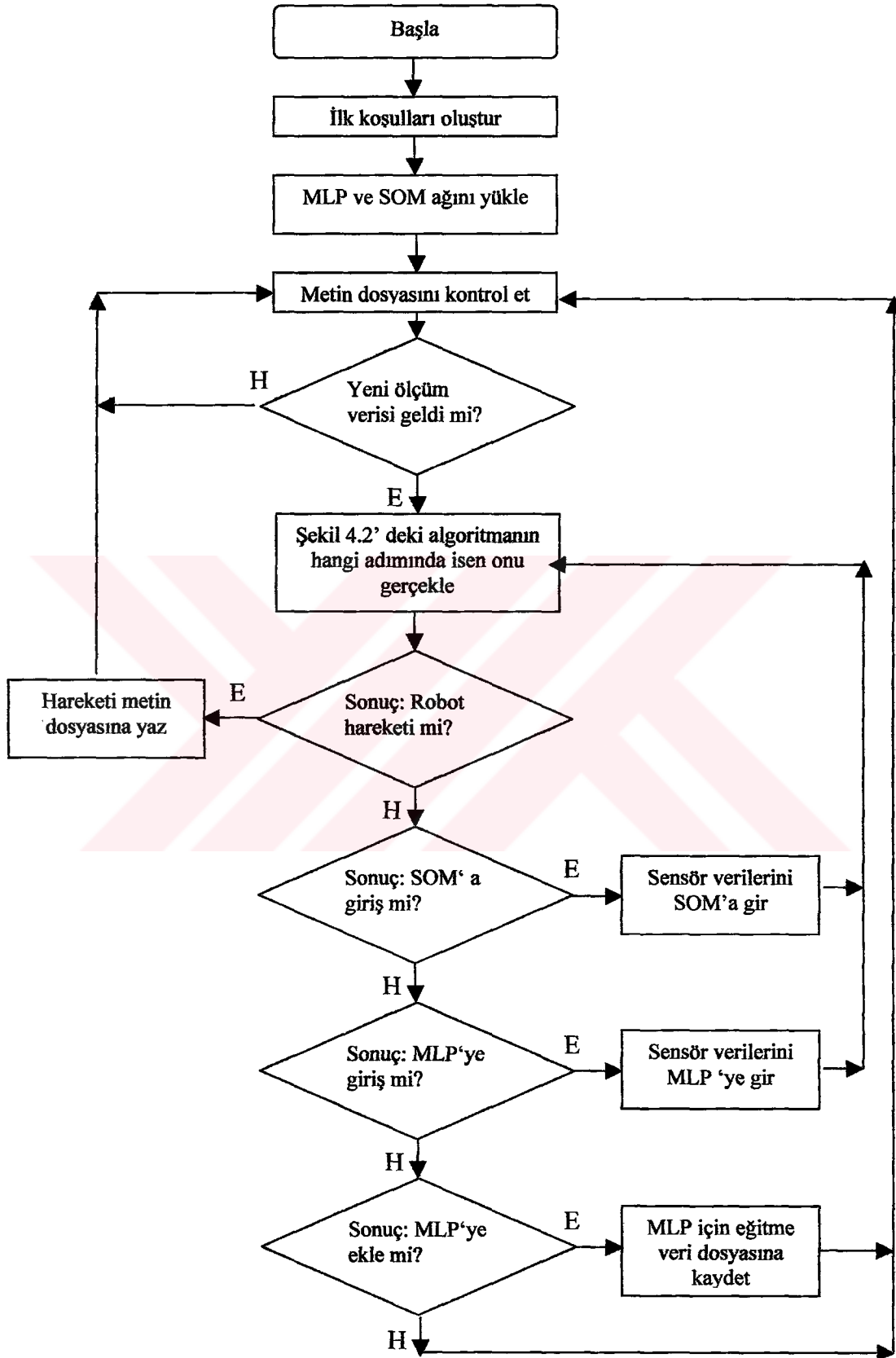
3.3.3 Matlab Uygulama Yazılımı

Tasarlanan Matlab uygulama yazılımı, gezgin robottan gelen uzaklık ve ışık yoğunluğu ölçüm değerlerini SOM ve MLP yapay sinir ağı modellerine giriş olarak verir ve bu ağ modellerinin çıktıları ile robotun bir sonraki hareketini belirler (Markbeale, 1998).

Daha önceden bahsedildiği gibi Matlab uygulama yazılımı haberleşme ara yüz programı ile etkileşimli çalışır ve haberleşme ara yüz yazılımı tarafından metin dosyasına yazılan uzaklık ve ışık yoğunluğu değerlerini okur. Gerekli algoritmaları çalıştırarak robotun yapması gereken hareketi yine metin dosyasına yazarak haberleşme ara yüz yazılımının okumasını sağlar.

Matlab uygulama yazılımının işleyiş adımları içinde Şekil 4.2' de görülen Robot çalışma algoritması blok diyagramı da bulunmaktadır. Şekil 3.11' de Matlab uygulama yazılımının akış diyagramı görülmektedir.





Şekil 3.11 Matlab uygulama yazılımının akış diyagramı

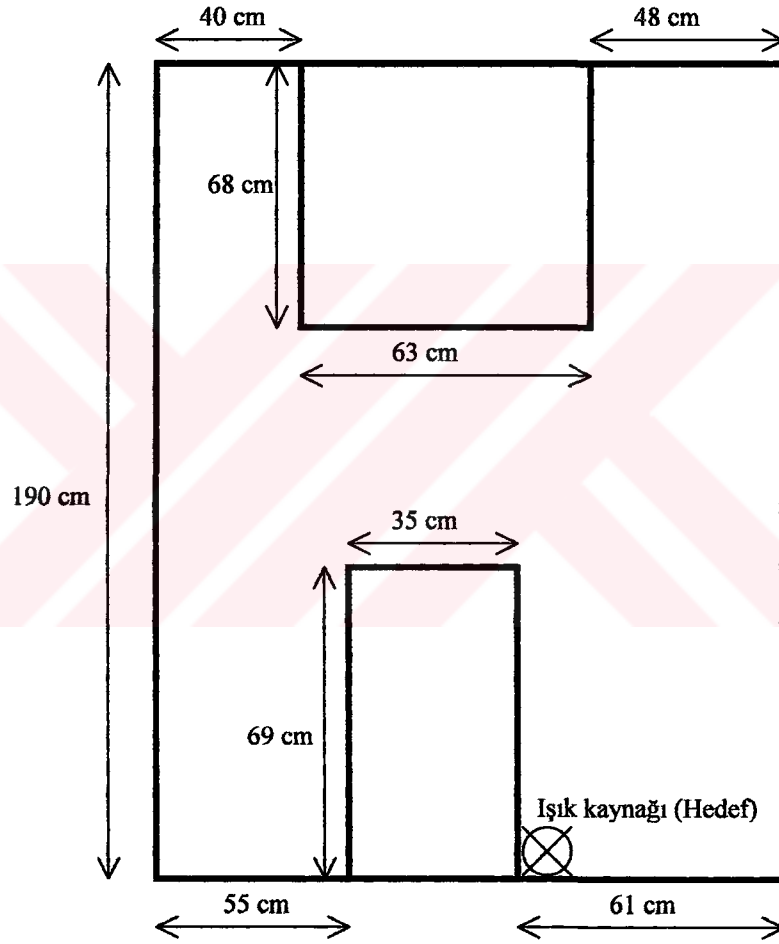
4. SİSTEMİN GERÇEKLEŞTİRİLMESİ

4.1 Robot

Uygulamada kullanılacak olan gezgin robotun bölüm 3.2 ve bölüm 3.3’de açıklandığı gibi donanım ve yazılım yapısı oluşturulmuştur.

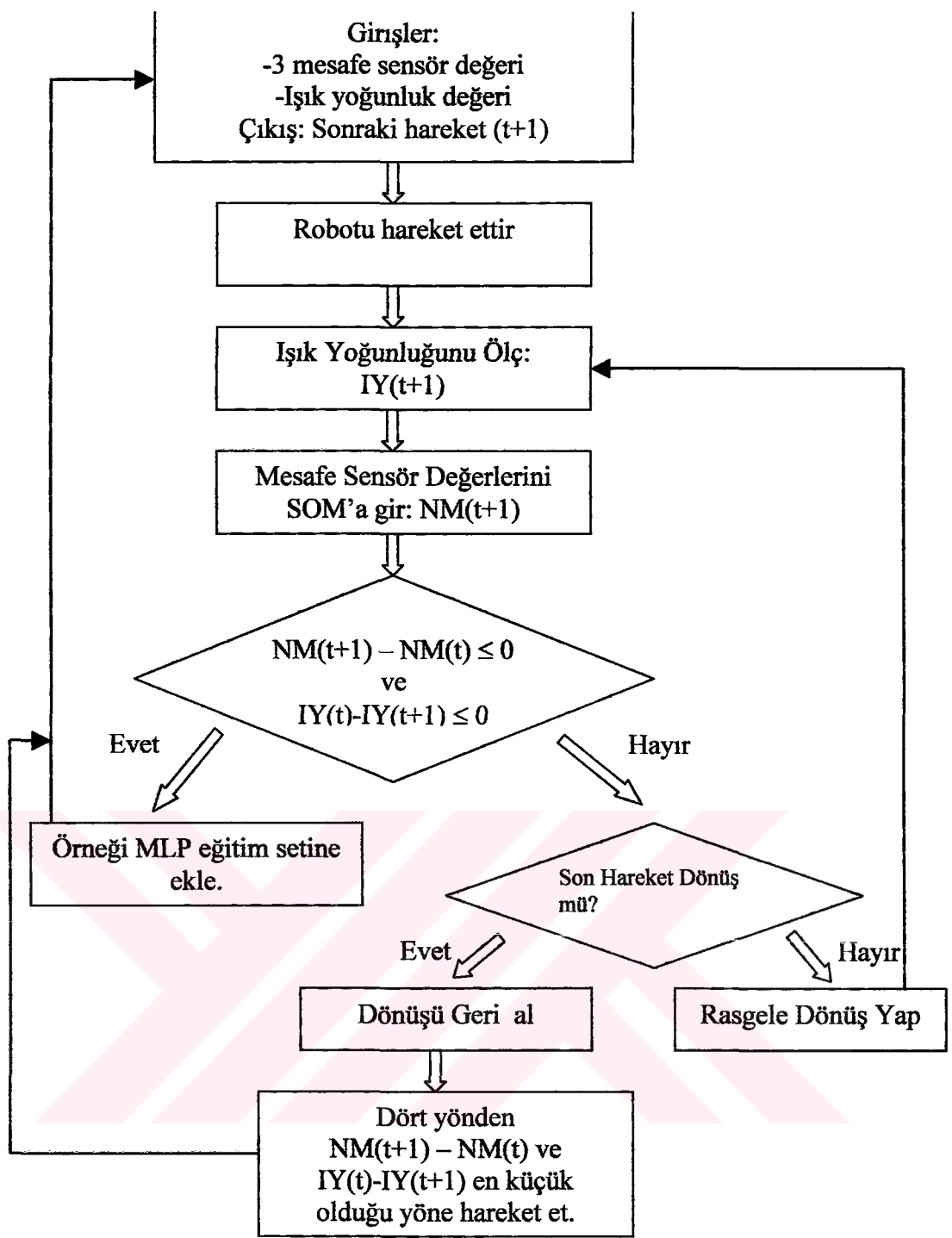
4.2 Labirent

Gezgin robotun yapay sinir ağları ile öğrenme çalışmasını gerçekleştireceği labirent ortamı Şekil 4.1’de görüldüğü biçimde ve ölçülerde oluşturulmuştur.



Şekil 4.1 Kullanılan labirentin şekli ve ölçüleri

Robotun gideceği hedef olarak ışık kaynağı kullanılmış ve bu amaçla labirentin içindeki hedef nokta olarak seçilen bölgeye 60W’lık bir ampul konmuştur.



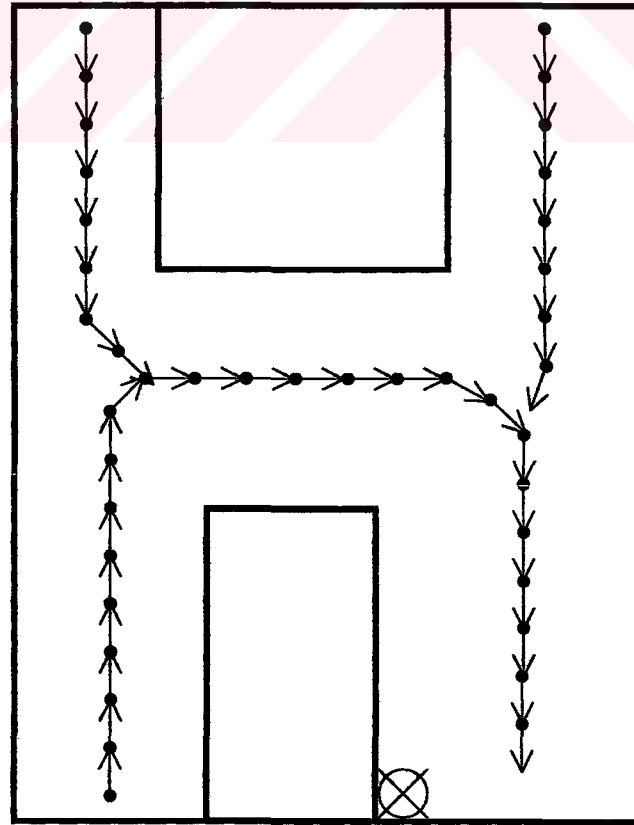
Şekil 4.2 Robot çalışma algoritması blok diyagramı

4.3.1 Çevresel verilerin ölçülmesi

Robot labirent içinde o an bulunduğu konumdaki ışık yoğunluğunu Bölüm 3.2.2.4' de anlatıldığı gibi ölçebilmektedir. Aynı zamanda ultrasonik mesafe sensörleri ile bulunduğu konumda 0° , 90° ve 180° 'lik açılardaki duvara olan mesafeleri de ölçebilmektedir.

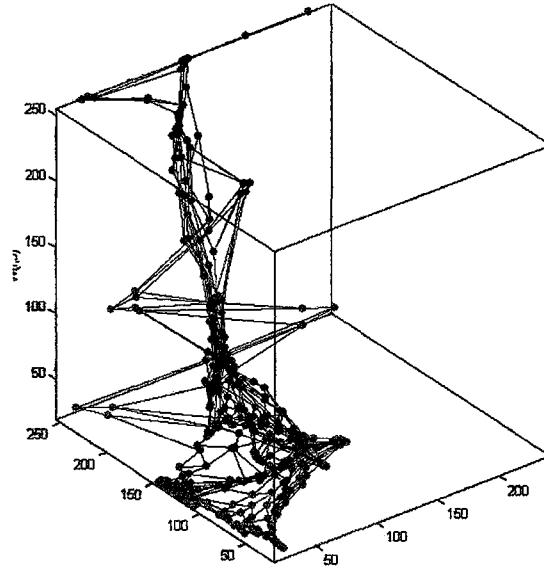
4.3.2 Robotta engellerden uzaklaşma

Robotun labirentin duvarlarına çarpmadan ilerleyebilmesi için SOM yapay sinir ağı modeli kullanılmıştır. SOM ağı giriş uzayındaki birbirine benzer verileri gruplayıp bunları iki boyutlu özellik haritasında yakın birimlere ataması özelliğinden dolayı tercih edilmiştir. Robot, labirent içersindeki izleyeceği güzergah üzerinde adım adım ilerletilmiş, her noktada 0° , 90° ve 180° 'lik açılarda ultrasonik mesafe ölçüm verileri kaydedilmiştir. Bu verileri kaydetmek için izlenen güzergah Şekil 4.3'de görülmektedir. Kaydedilen bu veriler SOM' un giriş uzayını oluşturacaktır. Kaydedilen veriler EK2a'da bulunmaktadır. SOM ağı bu giriş verileri kullanılarak 100, 1000, 10000 ve 50000 iterasyon ile eğitilmiştir. 50000 iterasyonla elde edilen nöron uzayı şekil 4.4'de görüldüğü gibidir.



Işık kaynağı (Hedef)

Şekil 4.3 SOM ağı eğitim veri kümesi oluşturulurken izlenen güzergahlar



Şekil 4.4 Elde edilen SOM nöron uzayı

Girişe en yakın ağırlık vektörüne sahip olan nöron (kazanan nöron) engel olmayan bir noktayı temsil etmektedir. Dolayısıyla SOM' a girilen veri bu nörona ne kadar yakın ise engellerden o derece uzak denebilir. Oluşturulan SOM ağı önce kendi giriş uzayından seçilen değerler için test edilmiştir. Bu testin sonucunun küçük değerler çıkması SOM ağının eğitildiğini göstermiştir. Ek2b'de girişler ve SOM ağının bu girişlere karşılık çıkışları verilmiştir. Daha sonra robot labirentte tekrar dolaştırılarak yeni bir veri kümesi elde edilmiş bu SOM ağını test etmek amacı ile kullanılmıştır. Bu test kümesi SOM girişine verilmiş ve SOM'un büyük oranda beklenen çıkışları ürettiği gözlenmiştir. Ek2c' de test kümesi ve SOM sonucu görülmektedir. SOM' un bazen hatalı çıkışlar vermesi ultrasonik sensör ölçümlerindeki kısıtlamalardan kaynaklanmaktadır. Robotun dönüş yaptığı adımlarda ultrasonik sensörlerin gönderdiği ses dalgasının dönüşte yansımadan yüzeyi sıyırması, bu noktalarda ölçümü hatalı yapmasına ve ölçüm sonucu olarak 255 değeri döndürmesine neden olmaktadır. 255 değeri ultrasonik sensörün ölçebileceği maksimum değere (2.6 metre) karşılık gelir ve sensör ölçüm yapamadığı durumlarda da bu değeri göndermektedir. SOM ağının test sonucu incelendiğinde test setindeki 255 değerini içeren satırlarda, SOM çıkışının çok büyük değerler aldığı görülmektedir. Diğer durumlarda SOM sonucu başarılıdır.

4.3.3 Robotun Hareket Etmesi

Robotun içinde bulunduğu çevresel koşullara göre dört farklı hareketten birini, Şekil 4.2' de verilen robotun çalışma algoritma çevriminin ilk hareketi olarak yapmasını belirleyen yapay sinir ağı modeli MLP' dir.

Oluşturulan MLP yapay sinir ağının girişleri, 3 farklı açıda alınmış ultrasonik mesafe ölçüm verisi ve ışık yoğunluğu bilgisidir. MLP cevabı ise dört olası hareketi kodlayan iki binary çıkıştan oluşmaktadır (ileri=00, geri=10, sağa=11, sola=01). MLP ağının verdiği hareket çıkışının doğruluğunun kontrolü algoritmanın daha sonraki adımlarında yapılmaktadır.

4.3.4 Yapılan Hareketin Doğruluk Kontrolü

Robotun çalışma algoritmasında, MLP çıkışı elde edilip ve robot bu çıkış yönünde hareketini gerçekleştirdikten sonraki aşama yapılan bu hareketin istenen kriterlere uygun olup olmadığına karar verme aşamasıdır.

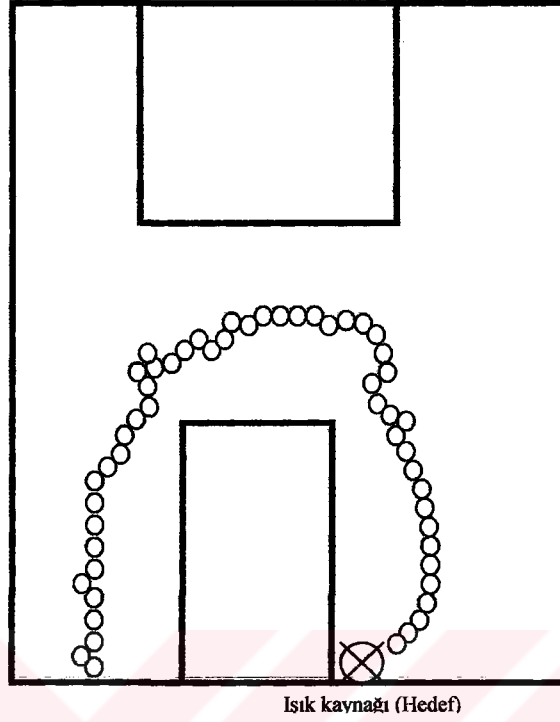
Hareketin doğruluk kontrolünün esası, robotun bu hareketi yapmadan önceki pozisyonunda ölçtüğü ışık yoğunluğu ($IY(t)$) ve ultrasonik sensör verilerine karşılık SOM çıkış değeri ($NM(t)$) bilgilerinin, hareketi yaptıktan sonraki pozisyonda ışık yoğunluğu ($IY(t+1)$) ve SOM çıkışı ($NM(t+1)$) bilgilerinin karşılaştırılmasına dayanmaktadır. Eğer $NM(t+1) - NM(t) \leq 0$ ve $IY(t) - IY(t+1) \leq 0$ koşulu sağlanıyorsa yapılan hareket doğru kabul edilir ve bu pozisyondaki MLP' nin giriş ve çıkış değerleri MLP eğitim setine eklenir. Aksi halde yapılan hareketin tipine göre bir dizi hareket yapılarak, ışık yoğunluğunu en fazla olduğu ve SOM çıkışının en az olduğu yön seçilerek, bu yönde robot hareket ettirilir ve bu pozisyondaki MLP' nin giriş ve çıkış değerleri eğitim setine eklenir .

4.4 Uygulama

Uygulamanın amacı, robotun öğrenmeden önce hedefe ulaşma şekli ile öğrendikten sonra hedefe ulaşma şeklini karşılaştırmaktır

İlk olarak robotun öğrenmeden önce hedefe ulaşırken aynı zamanda MLP eğitim seti oluşturması işlemi gerçekleştirilmiştir. Bu işlem sırasında robotun labirent içinde yapmış

olduğu hareket adımları Şekil 4.5’de görülmektedir. Bu işlemin sonucunda elde edilen eğitim veri kümesi Ek2d’ de verilmiştir.

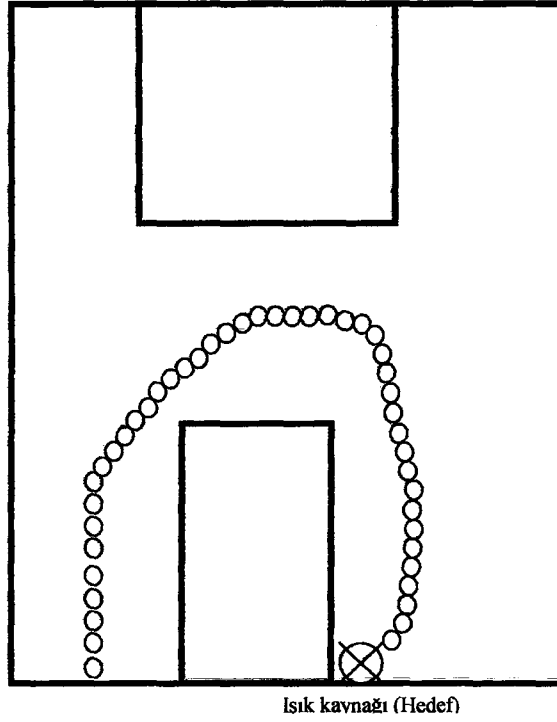


Şekil 4.5

MLP ağı eğitilmemiş robotun labirent içinde yapmış olduğu hareket adımları

Uygulamanın sonraki aşamasında, öncelikle elde edilen eğitim setindeki giriş verileri normalize edilmiştir. Şöyle ki, 0-255 aralığındaki 3 ultrasonik sensör verisi ve 0-1024 aralığındaki ışık yoğunluğu verisi 0-1 aralığında olacak şekilde normalize edilmiştir. MLP ağı eğitim setinin bu düzenlenmiş şekli ile eğitilmiştir. Birden fazla MLP ağ yapısı ile testler yapılmış ve en iyi sonucu (en kısa öğrenme zamanı ve en az hata) 8-10-10-2 yapılı olan ağ vermiştir. Saklı katman ve çıkış katmanı düğümlerinde sigmoidal aktivasyon fonksiyonları kullanılmıştır. Eğitim algoritması olarak standart geriye yayılma algoritması seçilmiştir. Öğrenme oranı 0.3 olarak alınmıştır. Eğitilen MLP ağının test edilmesi için bir test seti oluşturulmuştur. Bu test seti Ek2e’ de verilmiştir. Test sonucu başarı oranı ortalama %70’dir.

Artık uygulama eğitilmiş bir MLP ağına sahiptir. Bundan sonraki aşamada öğrenmiş olan robotun labirentin başlangıç noktasından hedefe ulaşırken yaptığı davranış incelenmiştir. Eğitilmiş robotun labirent içinde yapmış olduğu hareket adımları Şekil 4.6’da görülmektedir



Şekil 4.6

MLP ağı eğitildikten sonra robotun labirent içinde yapmış olduğu hareket adımları

Çizelge 4.1’ de MLP eğitilmeden önce ve eğitildikten sonra robotun labirent içersinde yapmış olduğu hareketlerin sayısal dağılımı görülmektedir. 1.sütun başlangıç noktasından hedefe kadar yapılan bütün hareketlerin toplam sayısıdır. Doğru ve yanlış hareketleri içerir. Her kabul edilmeyen yanlış hareketten sonra doğru hareketi bulmak için 8 hareket yapılır. 2.sütun da hatalı hareket sayısını gösterir

	Adım Sayısı	Hatalı Hareket Sayısı
Eğitilmemiş MLP	585	73
Eğitilmiş MLP	64	2

Çizelge 4.1 Hareketlerin sayısal dağılımı

5. SONUÇLAR ve ÖNERİLER

Yapılan bu çalışmada, iki farklı yapay sinir ağı modelinin koordineli çalışarak bir robotun öğrenme işlemini gerçekleştirmesini sağladıkları gözlenmiştir.

Kendi kendine organize olabilen haritalar (SOM) kullanılarak robotun ölçmüş olduğu uzaklık verileri ile labirent duvarlarını SOM uzayına taşıyabildiği ve başarılı sonuçlar verdiği yani robotun hedefe ilerlerken duvarlardan uzak durduğu görülmüştür. Aynı zamanda robotun hedefe doğru ilerlemesini sağlayan çok katmanlı algılayıcı (MLP) ağının da eğitildikten sonra verdiği çıkışlar ile robotun hedefe varış süresini kısaltmış olduğu gözlenmiştir.

Uygulamanın gerçekleşmesi sırasında yaşanan sıkıntı ultrasonik sensörden kaynaklanmıştır. Özellikle robotun dönüş yaptığı anlarda yüzeyden yansıyan ses dalgasının alıcıya ulaşmamasından dolayı mesafe ölçümü yapılamamaktadır. Ultrasonla mesafe ölçümündeki ana dezavantaj nesnelerin yüzeyinden gerçekleşen yansıma ile ilgili problemlerdir. Buna “aynasal yansıma” adı da verilir. Yansıma yönü gelen ses dalgasının yüzeye yaptığı açıyla ve yüzeyin şekliyle ilgilidir. Geliş açısı ne kadar ufak olursa, sesin yansıma yapmadan yüzeyi sıyırması ihtimali o kadar yükselir ve bu şekilde hatalı bir uzak mesafe ölçümü yapılır. Bu duruma aynasal denmesi sebebi ise, kaygan yüzeylerin yansıtıcı özellikleri ile bu sorunun büyümesine yol açmalarıdır. Bu dezavantajlarına rağmen ultrason ölçümleri hareketli robot uygulamalarında sıklıkla uygulanmaktadır, bu uygulamalar arasında iç mekan ve dış mekan haritalarının çıkarılması da yer almaktadır.

Sistemin maliyetini ve toplam hacmini düşüreceğinden yapay sinir ağı modelini mikro-denetleyici yazılımı içersinde oluşturmak; mesafe sensörü olarak yansıma yapmayan yüzeylerden etkilenmeyen bir teknoloji kullanmak veya ultrasonik sensör kullanılacaksa daha az zamanda daha çok açı tarayabilecek bir sensör seçmek; bu çalışmada yapılan veri toplama sonunda ağı eğitmek yerine, gerçek zamanlı eğitimin (veri alınırken yapılan eğitim) nasıl sonuç vereceğini incelemek; bu konu ile ilgili ileride yapılacak çalışmalara öneri olarak sunulmaktadır.

KAYNAKLAR

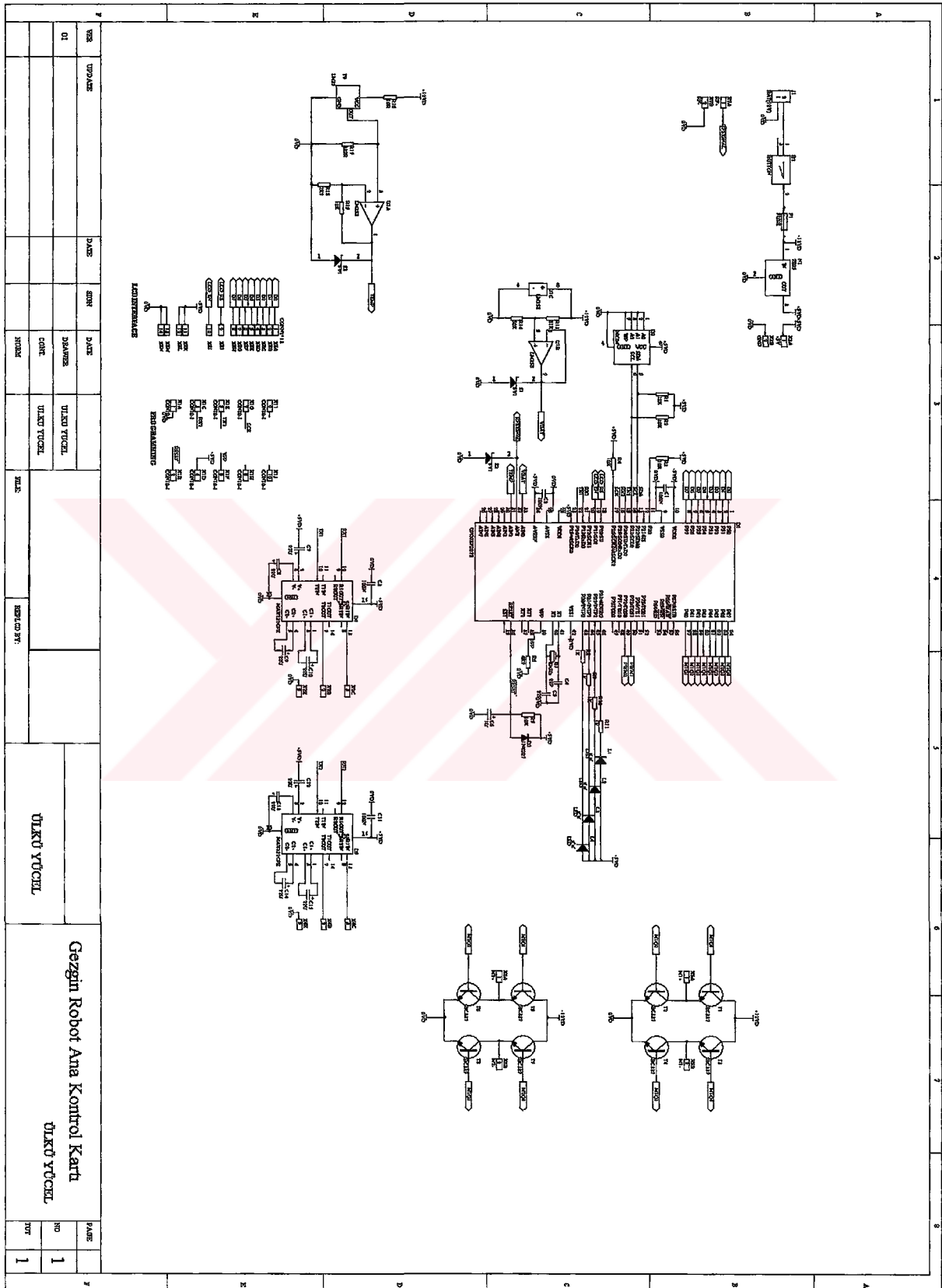
- Alonso R.H. ve Marques N.M. ve Brio B.M., (2002), “A Simple Approach to Robot Navigation Based on Cooperative Neural Networks”, Spain –2002 IEEE.
- Critchlow A.J. ,(1985), Introduction To Robotics, MacMillan Publishing Company
- Fuller J.L., (1999), Robotics Introduction, Programming and Projects, Prentice Hall.
- Ham F.M., (2001), Principle NeuroComputing for Science and Engineering, McGrawHill.
- Haykin S., (1998), Neural Networks: A comprehensive Foundation, Prentice Hall.
- Kohonen T., (1989), Self-Organization and Associative Memory”, SpringerVerlag.
- Markbeale H.D., (1998), Neural network toolbox for use with Matlab.
- Mitchell R.J. ve Keating D.A., (1998), Neural Network Control of Simple Mobile Robot, Springer Verlag.
- Öztemel E., (2003), Yapay Sinir Ağları, Papatya Yayıncılık.
- Rumelhart D.E. ve Hinton G.E. ve Williams R.J., (1986), “Learning representations by backpropagation errors”, Nature, vol 323, pp 533-536.
- Schalkoff R.J., (1997), Artificial Neural Networks, MIT Press&McGrawHill.
- Simon H.A., (1983), Why Should Machines Learn? In: Machine Learning- An Artificial Intelligence Approach, Tioga Publishing.
- Wong S.C. ve Coghill G.G. ve MacDonald B.A.,(2001), “Natural Landmark Recognition using Neural Networks for Autonomous Vacuuming Robots”, New Zealand

İNTERNET KAYNAKLARI

- [1] www.iar.com
- [2] www.milinst.com
- [3] www.national.com
- [4] www.nec.de
- [5] www.tdk.com

EKLER

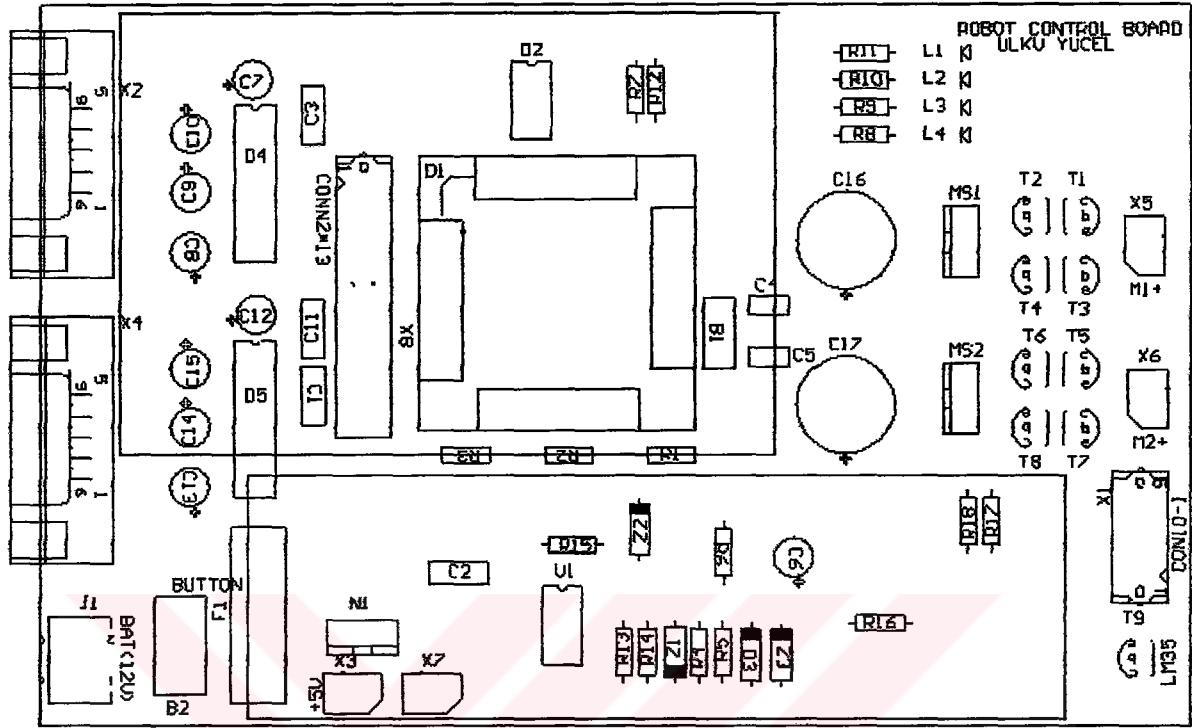
EK1a Ana Kontrol Kartı Devre Şeması



EK1b Ana Kontrol Kartı Malzeme Listesi

Parça Tipi	Referans Numarası	Parça Değeri
Direnç	R11,R7,R12,R8,R9,R10	1K
Diyot	D3	1N4007
Kondansatör	C6	1uF
Direnç	R18	2K7
Direnç	R6	4K7
Zener	Z1,Z2,Z3	5V1
Kristal OSC	B1	8 MHz
Direnç	R1,R2,R3,R4,R5,R15	10K
Direnç	R16	10R
Kondansatör	C4,C5	22pF
Kondansatör	C7,C8,C9,C10,C12,C13,C14,C15	22uF
Eeprom	D2	24C04
Direnç	R14	50K
Direnç	R13	81K
Kondansatör	C1,C2,C3,C11	100nF
Kondansatör	C16,C17	100uF
Direnç	R17	330R
Regülatör	N1	LM 7805
Transistör	T1,T2,T3,T4,T5,T6,T7,T8	BC337
Led	L1,L2,L3,L4	Led
Sensör	T9	LM35
Opamp	U1	LM358
Entegre	D4,D5	MAX232
Mikro-Denetleyici	D1	UPD78F0078

EK1c Ana Kontrol Kartı Baskılı Devre Yerleşim Şeması



EK2a SOM Yapay Sinir Ağı Modelini Oluşturmak için Kullanılan Veri Kümesi

Ultrasonik Sensör Ölçüm Değerleri		
0°	90°	180°
22	151	22
22	144	22
22	137	22
22	130	23
22	123	23
22	117	24
22	110	24
21	103	117
20	96	117
20	89	118
23	99	84
23	93	85
25	86	21
24	79	21
23	72	21
23	65	21
23	58	20
23	51	84
20	44	83
81	37	83
80	30	84
81	23	84
26	55	111
26	70	111
25	48	28
24	48	28
25	33	27
24	27	28
23	29	29
25	23	26
23	15	27
17	149	19
17	142	20
17	134	21
16	126	22
16	119	21
15	112	23
70	104	23
116	97	23
115	89	23
22	146	21
22	138	21
21	133	21
21	124	21
21	116	20
20	108	18
19	100	76
20	93	76
20	85	120

EK2b SOM Yapay Sinir Ağı Modelini Oluşturan Verilerin Test Verisi Olarak Kullanılması ve Sonuçları

Ultrasonik Sensör Ölçüm Değerleri			SOM Çevabı
0°	90°	180°	
22	151	22	2,39
22	144	22	0,70
22	137	22	0,05
22	130	23	0,35
22	123	23	0,13
22	117	24	0,11
22	110	24	0,14
21	103	117	2,14
20	96	117	1,81
20	89	118	1,97
23	99	84	1,12
23	93	85	0,39
25	86	21	0,21
24	79	21	0,91
23	72	21	1,43
23	65	21	2,33
23	58	20	0,16
23	51	84	2,34
20	44	83	2,67
81	37	83	2,85
80	30	84	0,57
81	23	84	2,87
26	55	111	0,82
26	70	111	0,57
25	48	28	0,12
24	48	28	0,13
25	33	27	0,24
24	27	28	0,58
23	29	29	0,78
25	23	26	1,98
23	15	27	2,98
17	149	19	2,06
17	142	20	2,17
17	134	21	0,12
16	126	22	0,14
16	119	21	0,13
15	112	23	0,24
70	104	23	0,08
116	97	23	4,03
115	89	23	4,03
22	146	21	0,90
22	138	21	0,06
21	133	21	0,33
21	124	21	0,13
21	116	20	0,14
20	108	18	0,22
19	100	76	1,21
20	93	76	0,97
20	85	120	1,92

EK2c SOM Yapay Sinir Ağı Modeline Giriş Olarak Verilen Test Verisi ve SOM Ağıının Sonuçları

Ultrasonik Sensör Ölçüm Değerleri			SOM Cevabı
0°	90°	180°	
21	151	26	6,01
21	148	26	4,91
22	144	26	4,39
22	139	26	5,00
23	135	25	2,77
24	130	25	3,63
22	126	26	3,52
23	121	26	3,51
24	117	25	2,82
23	113	26	3,68
22	108	27	3,75
24	103	117	2,24
23	98	117	2,86
25	94	117	3,82
25	89	117	4,06
24	86	117	3,51
145	28	255	114,00
255	68	255	190,00
255	63	28	143,00
255	58	26	144,00
23	54	255	57,10
23	48	255	62,30
28	88	24	4,04
26	83	23	2,19
26	78	23	2,98
26	73	22	3,10
26	68	22	3,22
27	63	21	4,46
27	58	21	4,66
27	53	21	5,50
25	48	84	2,15
27	44	82	2,22
86	39	82	3,82
86	34	81	3,54
128	43	255	91,40
127	38	255	93,90
126	34	255	96,00
32	255	27	6,63
255	68	255	190,00
28	63	255	46,90
255	57	25	144,00
255	52	24	146,00
31	44	24	8,33
31	39	23	8,12
31	35	23	8,30

32	29	23	9,25
32	25	23	8,67
31	20	22	8,48
31	15	22	9,92



EK2d Robot Tarafından Oluşturulan MLP Eğitim Veri Kümesi

Ultrasonik Sensör Ölçüm Değerleri			Işık Yoğunluğu	MLP Çıkışı	
0°	90°	180°		1	2
0.07421875	0.99609375	0.1171875	0.015625	0	1
0.08203125	0.52734375	0.09375	0.015625	0	1
0.08203125	0.5078125	0.09375	0.017578125	0	0
0.07421875	0.48046875	0.1015625	0.018554688	0	0
0.08203125	0.99609375	0.1015625	0.020507813	1	1
0.0703125	0.45703125	0.109375	0.021484375	1	0
0.07421875	0.4453125	0.109375	0.0234375	1	0
0.06640625	0.421875	0.11328125	0.01953125	1	0
0.3203125	0.1796875	0.32421875	0.024414063	1	0
0.99609375	0.28515625	0.12109375	0.029296875	1	1
0.47265625	0.99609375	0.08203125	0.030273438	1	0
0.1484375	0.0859375	0.99609375	0.04296875	1	1
0.12109375	0.375	0.30859375	0.041015625	1	1
0.08984375	0.99609375	0.09375	0.047851563	1	1
0.08203125	0.99609375	0.109375	0.049804688	1	0
0.078125	0.99609375	0.11328125	0.059570313	1	0
0.0625	0.140625	0.34765625	0.069335938	1	0
0.56640625	0.12109375	0.99609375	0.07421875	1	0
0.06640625	0.28515625	0.3359375	0.08203125	1	1
0.05859375	0.25390625	0.12109375	0.084960938	1	0
0.05859375	0.23828125	0.125	0.102539063	1	0
0.05859375	0.21875	0.12890625	0.225585938	1	0
0.07421875	0.99609375	0.35546875	0.229492188	1	1
0.36328125	0.21484375	0.359375	0.306640625	1	0
0.5	0.2109375	0.35546875	0.370117188	1	0
0.16015625	0.99609375	0.0703125	0.456054688	1	0
0.5546875	0.28125	0.08203125	0.478515625	1	0
0.12890625	0.23046875	0.07421875	0.515625	1	1
0.12109375	0.21484375	0.0859375	0.518554688	1	0
0.109375	0.19140625	0.09765625	0.569335938	1	0
0.1015625	0.16015625	0.10546875	0.62109375	1	0
0.08984375	0.99609375	0.57421875	0.6640625	1	0
0.10546875	0.99609375	0.21484375	0.673828125	1	1
0.546875	0.1171875	0.11328125	0.74609375	1	0
0.09765625	0.08984375	0.109375	0.827148438	1	0
0.0859375	0.07421875	0.11328125	0.880859375	1	0
0.0859375	0.05859375	0.11328125	0.89453125	1	0
0.58203125	0.078125	0.99609375	0.91015625	1	1

EK2e Eğitilen MLP Ağı için Test Veri Kümesi

Ultrasonik Sensör Ölçüm Değerleri			Işık Yoğunluğu
0°	90°	180°	
0,082031	0,589844	0,101563	0,005859
0,082031	0,578125	0,101563	0,006836
0,085938	0,5625	0,101563	0,006836
0,085938	0,542969	0,101563	0,007813
0,089844	0,527344	0,097656	0,007813
0,09375	0,507813	0,097656	0,008789
0,085938	0,492188	0,101563	0,009766
0,089844	0,472656	0,101563	0,010742
0,09375	0,457031	0,097656	0,011719
0,089844	0,441406	0,101563	0,012695
0,085938	0,421875	0,105469	0,013672
0,09375	0,402344	0,457031	0,014648
0,089844	0,382813	0,457031	0,014648
0,097656	0,367188	0,457031	0,015625
0,097656	0,347656	0,457031	0,015625
0,09375	0,335938	0,457031	0,015625
0,566406	0,109375	0,996094	0,020508
0,996094	0,265625	0,996094	0,022461
0,996094	0,246094	0,109375	0,024414
0,996094	0,226563	0,101563	0,02832
0,089844	0,210938	0,996094	0,03418
0,089844	0,1875	0,996094	0,040039
0,109375	0,34375	0,09375	0,03418
0,101563	0,324219	0,089844	0,041016
0,101563	0,304688	0,089844	0,048828
0,101563	0,285156	0,085938	0,056641
0,101563	0,265625	0,085938	0,067383
0,105469	0,246094	0,082031	0,088867
0,105469	0,226563	0,082031	0,239258
0,105469	0,207031	0,082031	0,25293
0,097656	0,1875	0,328125	0,245117
0,105469	0,171875	0,320313	0,240234
0,335938	0,152344	0,320313	0,233398
0,335938	0,132813	0,316406	0,232422
0,5	0,167969	0,996094	0,324219
0,496094	0,148438	0,996094	0,339844
0,492188	0,132813	0,996094	0,34375
0,125	0,996094	0,105469	0,410156
0,996094	0,265625	0,996094	0,44043
0,109375	0,246094	0,996094	0,467773
0,996094	0,222656	0,097656	0,469727
0,996094	0,203125	0,09375	0,510742
0,121094	0,171875	0,09375	0,578125
0,121094	0,152344	0,089844	0,640625

0,121094	0,136719	0,089844	0,706055
0,125	0,113281	0,089844	0,783203
0,125	0,097656	0,089844	0,868164
0,121094	0,078125	0,085938	0,944336
0,121094	0,058594	0,085938	0,986328



ÖZGEÇMİŞ

Doğum tarihi	28.09.1977	
Doğum yeri	Sakarya	
Lise	1991-1994	Adapazarı Atatürk Lisesi
Lisans İngilizce Hazırlık	1995-1996	Kocaeli Üniversitesi Mühendislik Fakültesi
Lisans	1996-2000	Kocaeli Üniversitesi Mühendislik Fakültesi Elektronik ve Haberleşme Mühendisliği
Yüksek Lisans	2002- Devam ediyor	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği Elektronik Ana bilim dalı.

Çalıştığı Kurumlar

2000-2001	Netaş Nortel Networks Ar-Ge Departmanı, Yazılım Tasarım Mühendisi
2001-Devam ediyor	Tübitak UEKAE Enstitüsü, Araştırmacı