

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



MORRİS SU LABİRENTİ ANALİZİNİN VİDEO
İŞLEME İLE GERÇEKLEŞTİRİLMESİ

Semih YAVUZKILIÇ

Yüksek Lisans Tezi
Anabilim Dalı: Elektrik ve Elektronik Teknolojileri
Tez Danışmanı: Prof. Dr. Abdulkadir ŞENGÜR

TEMMUZ-2017

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MORRİS SU LABİRENTİ ANALİZİNİN VIDEO İŞLEME İLE
GERÇEKLEŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

Semih YAVUZKILIÇ

151138104

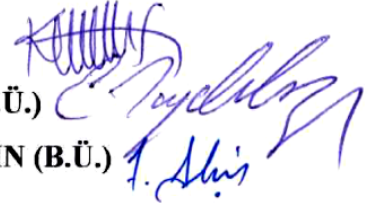
Tezin Enstitüye Verildiği Tarih: 20/06/2017

Tezin Savunulduğu Tarih: 14/07/2017

Tez Danışmanı :Prof. Dr. Abdulkadir ŞENGÜR

Diğer Jüri Üyeleri :Doç. Dr. Erkan TANYILDIZI (F.Ü.)

Yrd. Doç. Dr. Ömer Faruk ALÇİN (B.Ü.)



TEMMUZ-2017

ÖNSÖZ

Bu tez çalışmasında, değerli bilgilerini benimle paylaşan, çalışma azmi ve başarılarıyla hayatımda yeni ufuklar açan, eğitimime destek olan ve hiçbir zaman bu desteğini benden esirgemeyen saygıdeğer danışman hocam Prof. Dr. Abdulkadir ŞENGÜR'e ve aileme sonsuz teşekkürlerimi sunarım.

Semih YAVUZKILIÇ

ELAZIĞ-2017

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	II
İÇİNDEKİLER.....	III
ÖZET	V
SUMMARY	VI
ŞEKİLLER LİSTESİ	VII
SİMGELER VE KISALTMALAR.....	VIII
1. GİRİŞ.....	1
1.1. Tezin Tanımı	4
1.2. Tezin Amacı	4
2. GÖRÜNTÜ İŞLEME VE ANALİZ METOTLARI.....	5
2.1. Görüntünün İki Boyutlu Bir Sinyal Olarak Tanımlanması	5
2.2. Bölütleme	5
2.2.1. Gri Ton Bölütleme.....	6
2.2.1.1. Eşikleme Yöntemi İle Bölütleme	6
2.2.1.2. Bölgesel Tabanlı Bölütleme	9
2.2.1.3. Harekete Dayalı Bölütleme	10
2.3. Kenar Belirleme (Ayrıt Sezme).....	12
2.4. Görüntü Histogramı.....	13
3. DENEYSEL ÇALIŞMA.....	15
3.1. Matlab GUIDE ile GKA Oluşturulması.....	16
3.1.1. Nesneleri Çalışma Alanı Penceresine Yerleştirme.....	17
3.1.2. Matlab GKA’da Kullanılan Nesneler.....	18
3.1.2.1. Düğme (Push Button).....	18
3.1.2.2. Kaydırıcı (Slider).....	18
3.1.2.3. Metin Kutusu (Edit Text)	18
3.1.2.4. Sabit Metin (Static Text)	18
3.1.2.5. Liste Kutusu (List Box).....	19
3.1.2.6. Panel	19
3.1.3. Matlab GKA Ara Yüzlerine Program Yazma	19
3.2. Tasarlanan Ara Yüz.....	21

3.3.	Bölütleme ve İzleme.....	22
3.3.1.	Su Labirenti Bölütlemesi.....	22
3.3.2.	Fare İzleme	26
3.4.	Deneysel Sonuç	29
4.	SONUÇLAR	31
	KAYNAKLAR	32
	ÖZGEÇMİŞ	34



ÖZET

MORRIS SU LABİRENTİ ANALİZİNİN VIDEO İŞLEME İLE GERÇEKLEŞTİRİLMESİ

Morris Su Labirenti (MSL) laboratuvar farelerinde ve sıçanlarda mekânsal öğrenmeyi ve hafızayı araştırmak için kullanılan bir davranış testidir. Morris (1981, 1984) tarafından yaklaşık olarak 35 yıl önce bulunan bu testte su ile dolu geniş bir tank ve tankın içinde bir platform bulunmaktadır. Bu platformun yeri çalışma boyunca sabit kalmaktadır. Çünkü MSL tankında platformun yerini gösteren herhangi bir işaret bulunmadığı için fare veya sıçanların konum bulma kabiliyetleri bu hayvanların labirent içindeki hareketlerine bağlı olmaktadır. Denemeler süresince hayvanlar görünür ya da saklı yerleştirilen platformu bularak sudan kaçmayı öğrenirler. MSL testindeki performans; dikkat, öğrenme, hafıza ve motor koordinasyonu gibi çeşitli faktörlere bağlıdır. Üstelik tank özellikleri, eğitim prosedürleri, suyun özellikleri (sıcaklık, saflık, vb.) cinsiyet, deney hayvanlarının ırkı, beslenme durumu ve daha birçok değişken MSL'deki performansı etkilemektedir. Araştırmacı tarafından yapılacak olan doğrudan bir analiz birçok davranışsal bilgi kaybına neden olacağından deney sırasında hayvan performansı bir kamera aracılığıyla kayıt altına alınır.

Bu tez çalışmasında, video analizi ile MSL testi gerçekleştirilmiştir. İlgili deneysel çalışmalar MATLAB'da gerçekleştirilmiştir. Analizin ilk aşaması olarak labirentin yeri bir eşikleme algoritması ile belirlenir. Labirentin yerini bulduktan sonra ilgi bölgeleri (İB) tanımlanır. Bir sonraki adımda fare labirent içinde bölütlenerek her bir video çerçevesi için farenin yeri tespit edilir. Farenin davranışlarını daha iyi analiz edebilmek için labirent dört bölüme ayrılır. Farenin her bir bölümde geçirdiği süre kayıt altına alınarak yörüngesi çizilir. Kullanılan örnek video kaydı ile ilgili bilgisayar ortamında şimdiye kadar yapılan çalışmalarda istenilen ve beklenen sonuçlara ulaşılmıştır.

Anahtar Kelimeler: Morris Su Labirenti, Görüntü İşleme, Bölütleme, Video İşleme

SUMMARY

MORRIS WATER MAZE ANALYSIS WITH VIDEO PROCESSING

Morris water maze (MWM) is a behavioral test used to study spatial learning and memory in laboratory mice and rats. This test, which was found about 35 years ago by Morris (1981, 1984), has a large tank filled with water and a platform inside the tank. The location of this platform is constant throughout the study. The ability of the mice or rats to locate their position depends on the movement of these animals in the maze. During the course of the trial, animals learn to escape from the water by finding the platform in which they are hidden. Because in the MWM test the performance depends on various factors such as learning memory and motor coordination. The characteristics of the training procedures, the characteristics of the aquatic species, (temperature, purity, etc.) The livestock feeding status of the test animals and many other variables affect the performance of the MWM. A direct analysis performed by the investigator will cause a lot of behavioral information loss, so animal performance is recorded through a camera during the experiment as well.

In this thesis, MWM test video analysis was performed. The work was done on MATLAB. The location of the maze in the first step of the analysis is determined by a thresholding algorithm. After finding the location of the maze, Region of Interest (ROI) is defined. Next, Mouse location is determined for each video frame by using segmentation of maze. Maze is divided into four sections in order to better analyze the behavior of the Mouse. The orbits are recorded and time spent in each section of mouse to draw. The desired and expected results have been achieved when working on the sample video recording used up to now in the computer environment.

Keywords: Morris Water Maze, Image Processing, Segmentation, Video Processing

ŞEKİLLER LİSTESİ

	<u>Sayfa No</u>
Şekil 2.1. İki doruklu histogram.....	7
Şekil 2.2. Birden fazla nesne içeren görüntünün histogramı.....	8
Şekil 2.3 Örnek bir görüntü histogramı [16]	9
Şekil 3.1. GKA Başlangıç Ekranı.....	16
Şekil 3.2. GKA Çalışma Alanı Penceresi.....	17
Şekil 3.3. (a) GKA örnek çalışma alanı. (b) örnek çalışma alanına bağlı olarak oluşan ara yüz	17
Şekil 3.4. GUIDE Ara Yüzünün Programlanması	19
Şekil 3.5. Tasarlanan ara yüz uygulaması	21
Şekil 3.6. (a) Havuzun ortasında yüzen siyah bir fare ile örnek video çerçevesi. (b) Video çerçevesinin histogramı.	22
Şekil 3.7. Eşik değerine dayalı bir video çerçevesinin bölütlemesi, Morris su labirentine ait bağlı bölgenin konturu.	23
Şekil 3.8. (a) Bölge Seçimi Paneli. (b) Sanal olarak dörde bölünmüş labirent.....	24
Şekil 3.9. İB Seçimi için hazırlanan İB Seçim Paneli	25
Şekil 3.10. İB Seçimi yapıldıktan sonra seçilen İB'lerin gösterimi	25
Şekil 3.11 (a) Arka plan çıkarma işleminden sonraki görüntü. (b) eşik değerini kullandıktan sonraki fare bölütlemesi.....	26
Şekil 3.12. (a) Farenin birinci bölgedeyken takibi (fare sarı renkli bir ışınla takip edilir). (b) Farenin ikinci bölgede takibi (yeşil ışın). (c) Farenin üçüncü bölgede takibi (kırmızı ışın).	27
Şekil 3.13. Farenin gerçek zamanlı olarak çizmiş olduğu yörünge ve diğer deney bilgilerini gösteren pencere.....	28
Şekil 3.14. Video Seçenekleri Paneli	29

SİMGELER VE KISALTMALAR

$f(x, y)$	: Görüntü Fonksiyonunu
$I(x, y)$	: Enerji Miktarı Grilik Seviyesi
$n(x, y)$	: Görüntünün Olasılığa Bağlı Kısmı
$g(x, y)$	: Görüntünün Belirleyici Kısmını
T	: Eşik değeri
μ	: Gri Seviyelerin Ortlaması
L	: Gri Seviye Piksel Değeri
$h(r_k)$	: Sayısal Bir Görüntünün Histogramı
r_k	: Gri Seviye Değeri
n_k	: Gri Seviye Değerine Sahip Piksellerin Sayısını
N	: Toplam Piksel Sayısı
GKA	: Grafik Kullanıcı Arabirimi
GUI	: Graphical User Interface
İB	: İlgi Bölgesi
MSL	: Morris Su Labirenti
MWM	: Morris Water Maze
ROI	: Region of Interest

1. GİRİŞ

Biyolojik ve biyomedikal alanda elde edilen üç boyutlu (3B) görüntü ve videolardan elde edilen büyük veri dosyalarından çeşitli yararlı bilgiler ortaya çıkmaktadır. Bu veriler, bilimsel ve ticari alanda giderek daha önemli bir yer tutmaktadır. Bununla birlikte üretilen görüntü veri miktarı katlanarak artarken bu verileri kullanma ve işleme kabiliyeti sınırlı kalmaktadır.

Modern analiz yöntemleri kullanılmadan elde edilen bu verilerin işlenmesi mümkün olamamaktadır. Çünkü çok boyutlu hareketli görüntü verilerinin modern metotlar kullanmadan yapılan analizi ile görsel bir bilginin yalnızca küçük bir bölümünün analiz edilmesine imkân sağlamakla beraber bazen gözlemlenen biyolojik süreçlerin eksik ve hatalı bir şekilde anlaşılmasına yol açar. Görüntüler sayesinde elde edilen önemli bilgilerin büyük bölümü analiz edilirken zaman kısıtlamaları ve nesnellik sebepleriyle kaybolabilir. Ayrıca bu tür videolardan bilgiyi işleyebilmek için kullanılan geleneksel prosedürler, beraberinde ciddi sınırlamalar getirmekte ve kritik olayları tespit etmede başarısız olmaktadır.

Morris Water Maze (MWM) testi çağdaş sinir bilimi araştırmalarında önemli bir yer edinmiştir. Yıllar boyunca, Morris yüzme havuzu, Morris labirenti, su labirenti (özellikle Morris ve yardımcıları tarafından), yüzme labirenti, uzaysal navigasyon görevi, vb. gibi çeşitli isimler verilmiştir. Bu tez çalışmasında en yaygın isimlendirme olan Morris Su Labirenti (MSL) kullanılacaktır.

Bu açığı kapatmak için günümüzde birçok analiz metodu kullanılmaktadır. Morris Su Labirentinin (MSL) [1, 2]'deki çalışmaları bu alanda yaygın kullanılan metotlardan biridir. Bu metot laboratuvar farelerinde ve sıçanlarda mekânsal öğrenmeyi ve hafızayı araştırmak için kullanılan bir davranış testidir. MSL'in bu testi, son yıllarda davranışsal sinir biliminde en sık kullanılan laboratuvar araçlarından biri olmuştur. MSL testinde, berrak suyla dolu dairesel bir havuzun içine yüzeye batırılmış bir platform konulmaktadır. Fare havuza konulduğunda fareden beklenen davranış suyun içindeki gizli platformu bulup bu platformun üstüne çıkmasıdır. Fareye uygulanan bir dizi eğitimden sonra fare platformu bulmayı öğrenir. Gözlemci fareyi tekrar tekrar havuza bıraktığında farenin davranışlarındaki değişimleri gözlemleyerek farenin öğrenip öğrenmediğini tespit edebilir.

Brandeis ve arkadaşlarının çalışmasından bu yana, MSL'yi deneysel bir cihaz olarak uygulayan birçok araştırma raporu ortaya çıkmıştır [3]. MSL'yi kullanan çalışmalar, sinir biliminin çeşitli alanlarına önemli derecede yenilikler getirmiştir.

MSL, araştırma grupları tarafından birçok farklı metodolojik varyasyon ve birçok farklı uygulamada kullanılmıştır. Araştırmacılar, MSL performansının, cihazların veya eğitim prosedürünün farklılıkları gibi deney hayvanlarının özelliklerine bağlı faktörlerden etkilendiğinin farkına vardılar. Bundan dolayı MSL deneylerinde elde edilen sonuçlarda, hayvan özellikleri, eğitim prosedürleri ve test cihazlarının etkisine bağlı birçok çalışma ortaya çıkmıştır.

MSL analizinde kullanılan eğitim prosedürleri, kullanılan test cihazları ve aletlerle beraber çeşitli çalışmalarda değiştirilmiştir. Saf süt veya süt tozu, toksik olmayan beyaz boya veya sentetik matlaştırıcılar da dahil olmak üzere çeşitli matlaştırıcılar kullanılmıştır. Özellikle küçük kemirgenlerin kullanıldığı deneylerde, su sıcaklığının 25°C civarında veya üstünde tutulmasının gerekliliği ve sütün bu sıcaklık değerlerinde bozulması nedeni ile matlaştırıcı olarak sütün kullanımını zorlaştırmaktadır. Bu yüzden sentetik matlaştırıcılar daha çok tercih edilmektedir. Örneğin Lytron 621 gibi sentetik matlaştırıcılar ile mükemmel bir şekilde çalışılabilmesine rağmen Lytron 621 elde edilmesi güç olabilmektedir [4].

Deneylerde farklı olarak kullanılan aletlere örnek olarak havuzun büyüklüğü de örnek verilebilir. Deneylerde çapı 2 metrenin altında 1 metrenin üstünde değişen farklı havuz boyutları kullanılmaktadır. Deney hayvanı olarak laboratuvar farelerini kullanan araştırmacıların çoğu büyük boyutlu tankları deneylerinde kullanmıştır. Morris başlangıçta sıçanlarda 132 cm ve 214 cm çaplı iki su tankı kullanmıştır [5]. Birkaç yazar, farelerde kullanılmak üzere tank boyutlarını küçültmüştür. Örneğin, Riekkinen ve arkadaşları, yaptıkları çoğu deneysel çalışmada 120 cm'lik su tankını başarıyla kullandıkları görülmektedir [6]. Bununla birlikte Lamberty ve Gower, 90 cm çapında daha küçük bir tank kullandıklarında oldukça memnun edici sonuçlar elde etmişlerdir [4]. Means ve arkadaşları farelerde kullanılmak üzere sadece 61 cm çapında küçük bir tank tercih ederken, Transjenik fare modelleri üzerine yapılan son çalışmalarda 150 cm çaplı su tankları kullanılmıştır [7, 8].

Farenin havuzda takibi genellikle otomatik olarak bilgisayar destekli bir video kamera yardımı ile yapılmaktadır. Tank sanal olarak dört kısma bölünür ve bölünen bu dört parçadan herhangi birine, su seviyesinin iki cm altında kalacak ve üstten bakıldığında görülmeyecek şekilde bir platform yerleştirilir. Deneyin yapıldığı ortam ve çevre sürekli aynı tutularak sabit

sıcaklık parametreleri ve görsel işaretlerle donatılır. Eğitimler esnasında deneyleri yapan kişinin fareyi her zaman aynı noktadan suya bırakması beklenir. Yapılan bazı deneylerde araştırmacılar, deneylerin ilk gününde dış etkileri en aza indirmek amacıyla su tankının etrafını bir perde ile çevirirler. Bu araştırmacılar, hayvanları suya alışmalarını sağlamak ve hayvanlara platforma çıkmalarını öğretmek için onları platforma yakın bir yerde tanka bırakırlar. Sonra onları platforma doğru yönlendirirler. Eğitim süresi ilerledikçe fareyi suya bırakma noktası ile platform arasındaki mesafe arttırılmalıdır. Bunu yapmadaki asıl amaç farenin tankın içinde bulunan platformun yerini öğrenmesidir. Ancak platformun yeri, platformun yerini alıştırma eğitimi ile ana eğitim sırasındaki yeriyle farklıdır. Ana eğitim genellikle her bir hayvanda günde 4 deneme ile 4 – 6 gün arasında sürer. Günlük 4 denemede fareler yüzleri havuzun duvarına bakacak şekilde havuzun çevresinde rastgele seçilmiş dört farklı ama tüm denekler için aynı noktalardan havuza bırakılır. Hayvan suya bırakıldıktan sonra platformu bulana kadar veya 60 sn suda kalır. Fareler su tankına her atıldığında, yüzmelerine izin verilen süreler eşit tutulmakta olup, bu süre çeşitli araştırmalarda 30 sn ile 180 sn arasında değişmektedir [9]. Farenin platforma çıktıktan sonra platform üzerinde bekletilme süreleri genel olarak 10 – 15 sn olup, 3 sn [10] ile 30 sn [9] arasında değişmektedir. Genellikle gruptaki tüm farelere ilk deneme gerçekleştirildikten hemen sonra ikinci deneme başlatılır ve bundan sonraki denemeler arasında ara verilen süreler biraz uzar. Seyrek eğitim, sık yapılan eğitimle karşılaştırıldığında daha başarılıdır. Ancak yapılan farklı çalışmalarda denemeler arasındaki süreler farklılık gösterebilmekte, denemeler arasındaki bu süre saniyeler ve dakikalar mertebesinde değişkenlik gösterebilmektedir.

MSL analizi kapsamlı bir şekilde kullanılmasına rağmen her zaman en iyi sonuçların elde edilmesi mümkün olmamaktadır. Mümkün olan en iyi veriyi elde etmek için kullanılan aygıtların ve test prosedürlerinin yetersiz olması bunun birkaç sebeplerindendir.

Araştırmacı tarafından yapılacak olan doğrudan bir analiz birçok davranışsal bilgi kaybına neden olacağından deney sırasında hayvan performansı bir kamera aracılığıyla kayıt altına alınmaktadır. Ticari ürün olarak da satılan bu tür deneysel video kayıt ve analiz sistemleri mevcut olmakla beraber bu ürünlerin özel donanım sistemlerine ihtiyaçları vardır ve bu ürünler ciddi verimlilik ve doğruluk sınırlamaları getirmektedir. Bu sınırlamalardan dolayı davranışsal sinir bilimi deneylerinde kullanılan mevcut sistemlerden elde edilen veriler gözlem sonucu elde edilen parametrelerin sürekli insan eliyle düzeltilmesine yol açmaktadır. Bunlar sonuçlarda hatalara ve tutarsızlıklara yol açan zaman alıcı işlerdir.

1.1. Tezin Tanımı

Morris Su Labirenti (MSL) hipokampüse bağılı olarak uzamsal öğrenme ve hafıza çalışmaları için kullanılan bir testtir. Birçok araştırmacı tarafından kullanılan bu testte içi su ile dolu geniş bir tank ve tank içinde bir platform bulunmaktadır. Bu platformun yeri çalışma boyunca sabit kalmaktadır. Bunun sebebi MSL tankında platformun yerini gösteren işaret bulunmadığından farelerin konum bulma kabiliyetlerini kullanarak etkili bir biçimde havuz içindeki labirentin konfigürasyonunu kullanmalarını sağlamaktır. Denemeler süresince hayvanlar dışarıdan görünen ya da tank içine gizlenen platformu bularak sudan kaçmayı öğrenirler. MSL testindeki performans hafıza, öğrenme, dikkat ve motor koordinasyonu gibi çeşitli etkenlere bağılıdır. Bununla beraber tankın özellikleri, hayvanları yüzdürme yöntemi, cinsiyet, kullanılan deney hayvanlarının ırkı, beslenme durumu ve stres teknik ve teknik olmayan pek çok değişken MSL analizindeki performansı etkilemektedir. Araştırmacı tarafından yapılacak olan doğrudan bir analiz birçok davranışsal bilgi kaybına neden olacağından deney sırasında hayvan performansı bir kamera aracılığıyla kayıt altına alınarak bilgisayar ortamına aktarılmaktadır.

1.2. Tezin Amacı

Bu çalışmada, kullanılacak örnek bir video kaydı için farenin davranışları (aldığı yol, süre, çizdiği yörünge vs.) takip edilecektir. Hataları ortadan kaldırmak veya en aza indirmek ve güvenilir sonuçlar elde etmek amacıyla tamamen bilgisayar ortamında analiz yapabilmek için MATLAB ve MATLAB GUIDE programları kullanılan otomatik bir sistem geliştirilecektir. Önerilen sistem ile çeşitli biyolojik ve biyomedikal araştırmalarda davranış modelleri üzerindeki etkisini ortaya çıkarılması ve araştırma için analiz gücünün artırılması amaçlanmaktadır.

2. GÖRÜNTÜ İŞLEME VE ANALİZ METOTLARI

2.1. Görüntünün İki Boyutlu Bir Sinyal Olarak Tanımlanması

Görüntü iki boyutlu bir ışık yoğunluğu fonksiyonu olarak tanımlanabilir. Görüntü fonksiyonunu $f(x, y)$ olarak ifade edersek, $f(x, y)$ herhangi x ve y koordinatlarındaki bir pikselin sahip olduğu ışık enerjisine ait bir ölçüdür. $f(x, y)$ fonksiyonu, ışıklandırma ve yansıtma olmak üzere iki ayrı fonksiyonun süper-pozisyonundan oluşur. Monokrom bir f görüntüsüne ait (x, y) koordinatlarındaki enerji miktarı grilik seviyesi olarak adlandırılır ve $I(x, y)$ biçiminde gösterilir [11].

Başka bir yaklaşıma göre, görüntü iki ayrı tabakanın üst üste konulmuş hali olarak düşünülebileceği belirtilmiştir [12]. Tabakalardan biri görüntünün belirleyici kısmını veren ve varyasyon göstermeyen kısmı, diğeri ise pikseller arasında büyük enerji farkları barındıran ve detayları veren, olasılıkla ifade edilebilen bölümüdür. Bir elma fotoğrafına bakıldığında elmanın silüetini ve şeklini belli eden kısım belirlenimci, üzerindeki tonlamaları, yüzeyindeki noktacıkları veren kısmın da olasılıksal kısım olduğu düşünülebilir. Resimlerin içinde bu olasılık bilgilerini barındıran tabakası cisimle ilgili önemli özelliklerin bilgilerini barındırır ve bu bilgiler cismin farklı bölgelerinin ışığı farklı derecelerde soğurmasına ve yansıtmasına bağlıdır.

Bir cismin üzerine gelen ışığa karşı cevabı bir atım fonksiyonuna cevabıdır ve bu anlık ışık tepkisine de görüntü denir. Görüntü yukarıda açıklandığı gibi iki ayrı bileşenden oluşur, $f(x, y) = g(x, y) + n(x, y)$. Burada $n(x, y)$ olasılığa bağlı olan kısımdır. Bu modelleme sayesinde görüntüyü istatistik biliminin bir malzemesi olarak görmek mümkündür. Böylece görüntüyü karakterize edebilen birinci dereceden ve ikinci dereceden istatistik bilgiler görüntü analizi ve sınıflandırmasında kullanılır [11].

2.2. Bölütleme

Görüntü bölütleme, homojen bir küme içerisinde aynı karakteristiğe sahip alanların çıkarılması, arka plandan ayrılması ve belirgin bir hale getirilmesi işlemine denir [13]. Bu anlamda görüntü bölütleme, bir görüntüyü her biri içerisinde farklı özelliklerin tutulduğu anlamlı bölgelere ayırmak olarak tarif edilebilir. Bu özellikler örneğin, görüntü içerisindeki

benzer parlaklıklar olabilir ve bu parlaklıklar ilgili görüntünün farklı bölgelerindeki nesneleri temsil edebilir. Görüntü içerisinde aynı parlaklıklara sahip nesne parçacıklarının belirlenmesi, sınıflandırılması ve tanınması amacı ile kullanılabilir. Onlarca görüntü bölütleme yöntemi olup her görüntüye uygulanabilecek ortak bir bölütleme yöntemi yoktur. Diğer görüntü işleme yöntemlerinde olduğu gibi (görüntü iyileştirme ve onarma) görüntü bölütleme için uygulanan yöntemler ve bu yöntemlerden istenen sonuçların alınması görüntüden görüntüye ve yöntemden yönteme bağlı olarak değişiklik gösterilebilmektedir.

2.2.1. Gri Ton Bölütleme

Gri-ton görüntü bölütleme algoritmaları, gri seviye değerlerinin iki temel özelliği olan süreksizlik (discontinuity) ve benzerlik (similarity) ile ilişkilendirilerek tasarlanır.

- Gri seviye değerlerindeki süreksizliklere göre görüntü bölütleme işlemindeki temel yöntem, gri seviyelerdeki ani değişimlere göre bir görüntüyü bölmelemektir. Bu da bir görüntüdeki kenar ve ayrıntıların belirlenmesine (edge detection) karşılık gelmektedir [14].

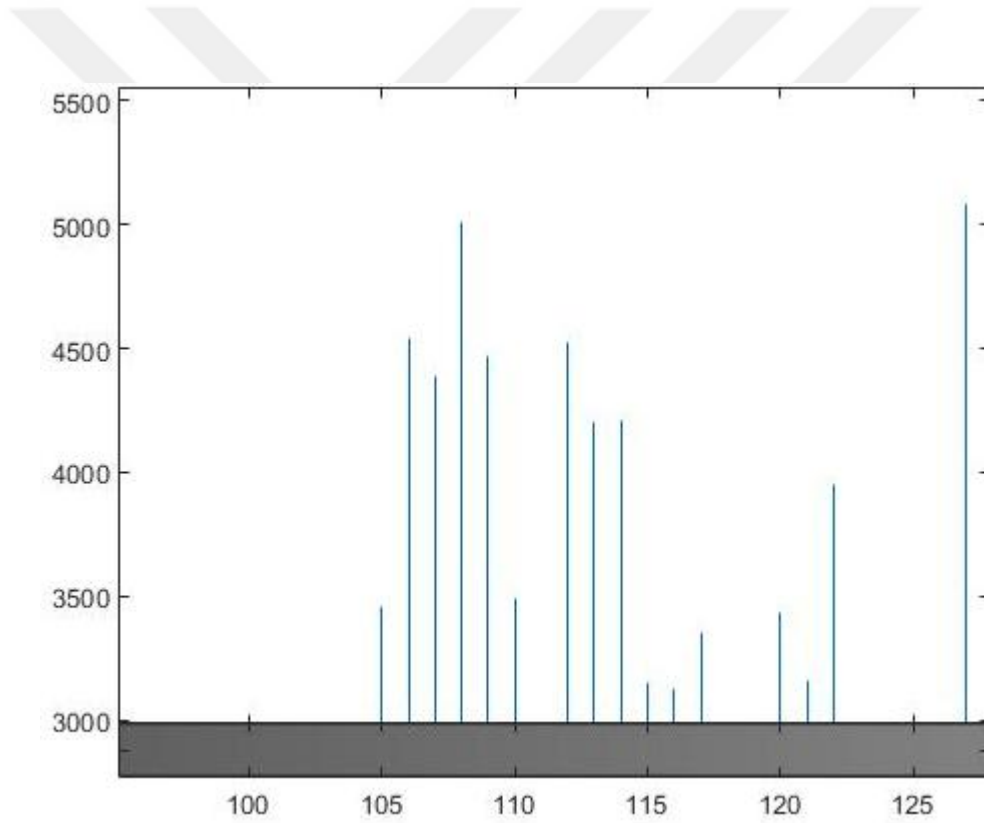
- Gri seviye değerlerindeki benzerliklere göre görüntü bölütleme ise, bölge bölütlemesi (region segmentation) olarak adlandırılır ve eşikleme (thresholding), büyütme (growing), ve yarma - kaynaştırma (split- and -merge) işlemlerine dayalı olarak gerçekleştirilir. Piksellerin gri seviye değerlerindeki benzerlik veya farklılıklara dayalı olarak bir görüntünün bölütlenmesi yöntemi hem durağan hem de dinamik (zamanla değişen) görüntülere uygulanabilir [14].

2.2.1.1. Eşikleme Yöntemi İle Bölütleme

Eşikleme, gri seviye ölçeğinde L adet değer bulunan bir görüntüyü ele alıp, bu görüntüden gri ton ölçeği ile sadece iki adet değer içeren ikili görüntünün elde edilmesidir. Birçok uygulamada $[0, L - 1]$ aralığında L adet değerden bir değer seçilir. Hangi değer seçileceği görüntü histogramında yapılacak çalışmaya göre belirlenir. Eşikleme global olarak uygulanırken gri seviye üzerinde bir eşik değeri saptanır ve bu eşik değerinden küçük olan değerler yerine sıfır, büyük olan değerler yerine $T - 1$ yerleştirilir. Basit eşikleme ifadesi Eşitlik 2.1'de tanımlanmıştır [15].

$$g(x,y) = \begin{cases} L - 1, & f(x,y) > T \\ 0, & f(x,y) \leq T \end{cases} \quad (2.1)$$

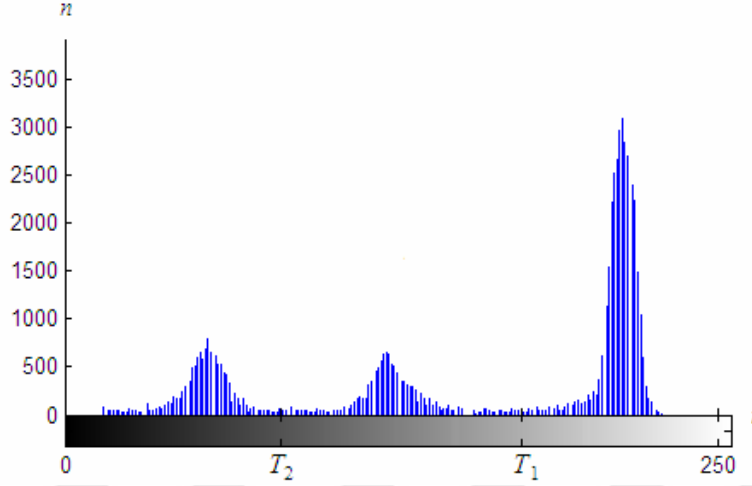
Yukarıdaki eşitlikte T eşik değeridir. Yüksek gri seviye değerli piksellere sahip arka plan üzerinde, düşük gri seviye değerli piksellere sahip nesneden oluşan bir görüntünün histogramı görülmektedir. Histogramdan anlaşılabileceği gibi görüntü, homojen gri seviyeye sahip bir nesneden ve arka plandan oluşuyorsa bu tip görüntülerin histogramına iki doruklu histogram denir [16]. İki doruklu histograma sahip görüntülerde nesneyi arka plandan ayırmak için iki doruğu ayıracak eşik değeri T seçilir. Herhangi bir (x,y) noktasında $f(x,y) < T$ sağlıyor ise nesne noktası, aksi durumda ise arka plan noktası olarak belirlenir.



Şekil 2.1. İki doruklu histogram

Görüntüde birden fazla nesne olduğunda birden fazla eşik değeri kullanılabilir. Şekil 2.2’de, yüksek gri seviye değerli piksellere sahip arka plan üzerinde, düşük gri seviye değerli piksellere sahip iki ayrı nesneden oluşan bir görüntünün histogramı görülmektedir. Herhangi bir (x,y) noktasında $T_1 \geq f(x,y) > T_2$ ise o nokta nesnelerden bir tanesine ait olur,

$f(x, y) < T_2$ ise o nokta diğer nesneye ait olur. Genelde birden fazla eşik değeri gerektiren bölütleme işlemlerinde bölge büyütme tekniği daha iyi sonuç vermektedir [16].



Şekil 2.2. Birden fazla nesne içeren görüntünün histogramı

Eşikleme tekniğinde eşik değerinin seçimi görüntünün histogramına dayanır. Bir gözlemci görüntü histogramına bakarak eşik değerini belirleyebilir. Bu tür bir buluşsal eşik değeri seçimi yani histogram gözlemine dayalı bir seçim aydınlatmanın kontrollü olduğu ve otomatik olmayan bir seçimdir. Görüntü histogramı dorukların anlamlı ve belirgin olduğu bir histogramda eşik değeri T otomatik yöntemlerle iki doruğun arasında ortalarda bir değer seçilebilir [16].

İki doruklu histograma sahip görüntülerde otomatik eşik değeri bulan ve sık kullanılan yöntemlerden olan iki eşik değeri bulma algoritması aşağıdaki adımlardan oluşmaktadır [16]:

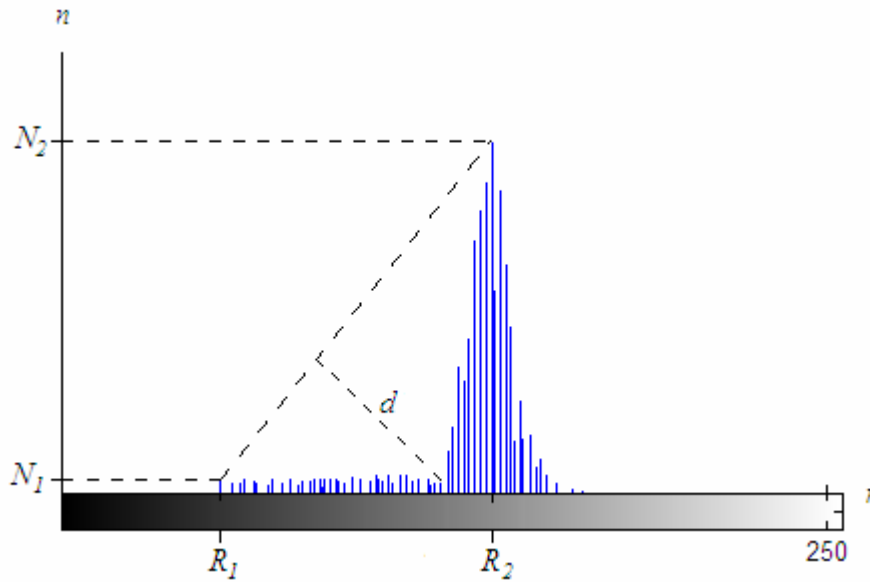
- 1. Adım:** Rastgele bir eşik değeri T seçilir.
- 2. Adım:** Gri seviye değeri T değerinden küçük veya eşit olan piksellerin gri seviyelerinin ortalaması μ_1 ve gri seviye değeri T değerinden büyük olan piksellerin gri seviyelerinin ortalaması μ_2 hesaplanır.
- 3. Adım:** Yeni eşik değeri Eşitlik 2.2'deki gibi hesaplanır.

$$T = (\mu_1 + \mu_2)/2 \quad (2.2)$$

4. Adım: Algoritmanın döngüsü sonrasındaki T değerleri arasındaki fark sıfır olana kadar, 2. ve 3. adımlar tekrar edilir.

Birinci adımda rastgele seçilen eşik değeri T için iyi bir seçim olarak (görüntüden bölütlenecek nesnenin ve arka planın kapladığı alanlar birbirine yakınsa) gri seviyelerin ortalaması alınabilir. Eğer nesnenin ve arka planın kapladığı alanlar arasında fark çok ise eşik değeri T için, en yüksek ve en düşük gri seviyelerin ortasında bir değer alınması iyi bir seçim olabilir.

Şekil 2.3’de yüksek gri seviye değerli piksellere sahip arka plan üzerinde, düşük gri seviye değerli piksellere sahip ve arka plana göre çok az yer kaplayan nesneden oluşan görüntünün histogramı görülmektedir [16]. Bu histogramda arka plan piksellerinin oluşturduğu doruk, nesne piksellerinin oluşturduğu doruğa göre çok yüksektir.



Şekil 2.3 Örnek bir görüntü histogramı [16]

2.2.1.2. Bölgesel Tabanlı Bölütleme

Bölgesel tabanlı bölütleme temel olarak, R ile temsil edilen görüntünün aşağıdaki özellikleri sağlayacak biçimde, $R_1, R_2, \dots, R_n$ ile temsil edilen parçalarına bölütlenmesi işlemidir [13].

1. $\bigcup_{i=1}^n R_i = R$
2. R_i komşulukları olan bir bölgedir. $i = 1, 2, \dots, n$
3. Tüm i 'ler ve j 'ler için $i \neq j$ olmak üzere $R_i \cap R_j = \emptyset$

4. Tüm i 'ler için $P(R_i) = DOĞRU$.

(R_i bölgesinin tüm piksel değerleri bir benzerlik kriterine göre benzer ise *DOĞRU* değeri üretilir).

5. $i \neq j$ iken $P(R_i \cup R_j) = YANLIŞ$

Birinci özellik, bölütleme işleminin tamamlandığını gösterir. Yani bir bölgeye (bu bölge nesne veya arka plan bölgesi olabilir) dahil olmayan piksel kalmadığını belirtir. İkinci özellik, bölütlenen bölgelerdeki piksellerin, kendi bölgesindeki piksellerle bağlı olduğunu gösterir. Üçüncü özellik, ise bir bölgeye dahil olan bir pikselin başka bir bölgeye dahil olmayacağını gösterir. Dördüncü özellik, bir bölgeye dahil olan piksellerin bağlılık koşulunda belirlenen gri seviye değerlerindeki benzerlik kriterini sağladığını gösterir. Beşinci özellik, iki farklı bölgeye dahil olan piksellerin bağlılık koşulunda belirlenen gri seviye değerlerindeki benzerlik kriterlerinin birbirinden farklı olduğunu gösterir.

Bölge büyütme ve bölge yarma-kaynaştırma (split - merge) yöntemleri bölgesel tabanlı bölütlemeye örneklerdir [16].

2.2.1.3. Harekete Dayalı Bölütleme

Hareket, insanlar tarafından nesneleri arka plandan ve gereksiz ayrıntıdan ayırmak için kullanılan çok güçlü bir işarettir [16].

Birbirini izleyen sıralı görüntülerde, hareket eden nesneyi bulmak bilgisayar görmesinin en önemli konularından bir tanesidir [17]. Hareket eden nesneyi bulmadaki en büyük problemde o nesneyi görüntüden bölütlemektir. Harekete dayalı bölütleme tekniklerinin çoğu, birbirini izleyen sıralı görüntülerdeki farkı bulmaya dayanır.

Görüntü işleme sistemlerinde nesnelerin hareketleri, hareket eden nesnenin bulunduğu ortamdan t_i ve t_j zamanlarında sabit kamera ile alınan $f(x,y,t_i)$ ve $f(x,y,t_j)$ gibi iki birbirini izleyen sıralı görüntü piksel piksel kıyaslanarak algılanır [17]. Farklı zamanlarda aynı ortamdan sabit kamera ile elde edilmiş görüntüleri kıyaslamanın en temel yolu fark görüntüsünü elde etmektir. Hareket eden nesnenin olmadığı sadece sabit elemanların olduğu görüntüyü referans görüntüsü kabul edip hareketli nesnenin olduğu görüntü referans görüntüsünden çıkarılarak sabit elemanların iptal edildiği, hareketli nesnelerin belirgin olduğu fark görüntüsü bulunur.

t_i ve t_j gibi iki farklı zamanda elde edilmiş görüntülerden bulunan fark görüntüsünün ifadesi Eşitlik 2.3'te gösterilmiştir.

$$d_{ij}(x, y) = \begin{cases} 255, & |f(x, y, t_i) - f(x, y, t_j)| > T \\ 0, & \text{diğer durumlarda} \end{cases} \quad (2.3)$$

Fark görüntüsünün ifadesindeki T değeri eşik değerini ifade eder. Referans görüntüsü ve çıkarılan görüntünün piksellerinin gri seviyeleri arasında eşik değeri ile belirlenen hatırı sayılır derecede fark var olduğu durumlarda $d_{ij}(x, y)$ 255 değerini alır ve 255 değerini alan pikseller hareketli nesne pikselleri olarak değerlendirilir [17].

Eşik değeri, hareket algılama uygulamalarında çok kritiktir [18]. Yanlış seçilen eşik değeri uygulamalarda hatalı sonuçlar verebilir. Çok düşük seçilen eşik değeri, fark görüntüsüne yapay değişimler ekler, çok yüksek seçilen eşik değeri önemli değişimlerin fark görüntüsünün belirgin olmasını önler. Doğru eşik değeri, ortama, değişken kamera seviyesine ve aydınlatma gibi görüntüleme koşullarına dayanır [18]. Bu da gösterir ki, eşik değeri, görüntü içeriğine göre dinamik olarak eşik değeri bulan yöntemler kullanılarak seçilmelidir ve deneysel olarak seçilen eşik değeri otomatik görüntü sistemleri için uygun değildir.

Harekete dayalı bölütlemeye her zaman referans görüntüsü yani hareketli nesnenin olmadığı sadece hareketsiz elemanların olduğu görüntüyü elde etmek mümkün olmayabilir. Bu durumda hareketli nesnenin olduğu iki görüntü kullanılarak fark görüntüsü bulunur. Referans görüntünün elde edilemediği problemlerde, fark görüntüsünden hareketli nesnenin harekete başladığı ve bitirdiği noktaları bulmak mümkündür. Bu tür problemlerde Eşitlik 2.4'teki fark görüntüsünün bulunuşuyla paralel fakat daha farklı bir yol izlemek gerekir. Hareketli nesneyi görüntüden bölütlemek için ilk olarak iki görüntü birbirinden çıkarılır ve Eşitlik 2.4'teki $g_{ij}(x, y)$ görüntüsü bulunur.

$$g_{ij}(x, y) = |f(x, y, t_i) - f(x, y, t_j)| \quad (2.4)$$

Birebir piksel değerlerinin birbirinden çıkarılmasından dolayı bulunan $g_{ij}(x, y)$ görüntüsünün gri seviye değer aralığı -255 ile 255 arasına kayar. Fark görüntüsünün gri seviye değerlerinin 0 ile 255 arasında olmasını istediğimiz için $g_{ij}(x, y)$ görüntüsünde ölçekleme yapmak gerekir. Yapılacak ölçekleme işleminin 0 ile 255 arasını tam olarak ve doğru şekilde kapsamı gerekir [16]. Bu şekilde bir ölçekleme algoritması aşağıdaki 2 adımdan oluşur.

1. Adım: $g_{ij}(x, y)$ görüntüsünün en küçük gri seviye değerinin eksilisi, kendisinin bütün pikselleri ile toplanır. Bu adım sonucunda $g_{ij}(x, y)$ görüntüsünün en küçük piksel değeri sıfır olur.

2. Adım: 1. adım sonucunda bulunan $g_{ij}(x, y)$ görüntüsünün bütün pikselleri $255/\text{Maksimum}$ değeri ile çarpılarak piksel aralığı tam olarak $[0, 255]$ arasını kapsayacak şekilde değiştirilir ve fark görüntüsü bulunur. *Maksimum* değeri 1. adım sonunda bulunan $g_{ij}(x, y)$ görüntüsünün en büyük gri seviye değeridir.

2.3. Kenar Belirleme (Ayrıt Sezme)

Kenar belirleme, görüntü işlemede temel öneme sahip konulardan biridir. Bir görüntüdeki kenar, aydınlatma veya yüzey yansımaları gibi görüntünün fiziksel görünüşünde oluşan önemli değişimlerdir. Bu değişim kendisini parlaklık, renk ve doku olarak gösterir. Ancak burada kenar anlamında, sadece görüntü parlaklıklarındaki değişikliklerle ilgilenilecektir. Bu anlamda, bir görüntünün gri seviyelerinde ani değişikliklerin olduğu bölgelere kenar adı verilecektir. Görüntüye ilişkin kenarların belirlenmesi birçok durumda kullanışlıdır. Nesne tanıma problemi buna bir örnek olarak verilebilir. Nesne tanımadaki temel adım, bir görüntünün farklı nesnelere karşı düşen farklı bölgelere bölmelemektir. Kenar belirleme işleminin kullanışlı olduğu diğer bir örnek, sadece görüntüye ilişkin kenarların kodlandığı düşük bit oranlarında görüntü kodlama uygulamasıdır.

Bir görüntüdeki fiziksel değişimin önemi uygulamaya bağlıdır. Şöyle ki, bazı uygulamalarda kenar olarak sınıflandırılan parlaklık değişimi diğer uygulamalarda kenar olarak incelenmeyebilir. Bir nesne tanıma sisteminde nesnenin sınırları, tanıma için yeterli olabilir ve nesne içerisindeki ek ayrıntılar kenar olarak değer görmeyebilir. Sonuç olarak kenar, uygulamanın içeriği dışında tanımlanamaz.

Cisimlerin fiziksel özellikleri ile kenarları arasında doğrudan bir ilişki söz konusudur. Dolayısıyla görüntünün birçok fiziksel özelliği kenar bilgisinden ortaya çıkarılabilir. Konunun önemi çerçevesinde bu bölümde kenar belirleme için çeşitli yöntemler sunulacaktır. Bunun için, görüntüye ilişkin ilgilenilen bölgelerin kendi içerisinde yeterince homojen oldukları varsayılacak ve dolayısıyla iki bölge arasındaki geçiş sadece gri seviye süreksizliklerine dayalı olarak tanımlanabilecektir.

İyi bir kenar kestirimci;

- Kenarları iyi bir biçimde sezebilmelidir,
- Kenarları doğru konumlarda belirleyebilmelidir,
- Bir kenar için bir kenar görüntüsü oluşturabilmeli yani yapay kenarlar üretmemelidir.

Görüntü içerisinde gürültünün varlığı (gürültü yüksek frekanslı bileşenlerden oluştuğu için gürültü ile kenarları birbirinden ayırt etmek güçleşecektir), kenar belirleme ve konumlama ölçütleri arasındaki karşılıklı ilişki ve kenarların çok ölçekli yapısı, kenar belirleme aşamasında karşılaşılabilecek önemli sorunlardır.

2.4. Görüntü Histogramı

Histogram, sayısal bir resim içerisinde her renk değerinden kaç adet olduğunu gösteren grafiklerdir. Bu grafiğe bakılarak resmin parlaklık durumu ya da tonları hakkında bilgi sahibi olunabilir [19]. Gri seviye değerleri $[0, L-1]$ aralığında değişen sayısal bir görüntünün histogramı, Eşitlik 2.5'te tanımlanmış bir ayrık fonksiyondur.

$$h(r_k) = n_k \quad (k = 0, 1, K, L-1) \quad (2.5)$$

Burada r_k . gri seviye değerini, n_k . gri seviye değerine sahip piksellerin sayısını gösterir [16]. $h(r_k)$ fonksiyonu görüntüdeki toplam piksel sayısını gösteren N değerine bölünerek $p(r_k) = n_k/N$ şeklinde normalize edilir. $p(r_k)$, gri seviye değerlerinin görüntüde bulunma olasılıklarının vermektedir.

Görüntü histogramı, görüntünün genel görünümü hakkında bilgi verir. Histogramda, piksellerin yoğunlaştığı bölgelere bakılarak yorum yapılabilir. Pikseller, histogramın düşük gri seviye değerlerini gösteren bölgelerinde yoğunlaşmış ise, görüntünün genel olarak karanlık olduğu sonucu çıkarılabilir. Pikseller yüksek gri seviye değerlerinde yoğunlaşıyorsa görüntünün aydınlık olduğu sonucuna varılabilir. Pikseller histogramın orta bölgelerinde yoğunlaşıyorsa, görüntüde gri tonun hakim olduğu sonucu çıkarılabilir. Pikselleri histogram üzerinde eşit dağılmış ise görüntünün karşıtlığının yüksek olduğu sonucuna varılabilir. Histogramda pikseller dar alanlarda yoğunlaşıyorsa, görüntü üzerinde uygulamalar yapmadan önce ön işlem olarak kontrast artırma işlemi ile görüntünün karşıtlığı artırılabilir.

Histogram, her bir gri seviye değerine sahip piksellerin sayısını vermesine rağmen, gri seviye değerlerinin görüntü düzlemindeki dağılımı hakkında bir bilgi vermediği için görüntü

içeriği hakkında histograma bakılarak fikir edinilemez. Görüntünün belli bir parçası ile ilgilenilirse, bu görüntü parçasına ait histogramın hesaplanması gerekir.



3. DENEYSEL ÇALIŞMA

Deneysel çalışmalardaki video analizi iki temel aşamaya göre yapılmıştır. İlk aşamada görüntülerde bulunan nesnelerin tanınması ve takip edilmesi; diğer aşama ise görüntüyü anlama ve yorumlama sürecidir. Her video çerçevesinde ayrı nesneleri tanımlamak için ilk önce görüntüler işlenmeli ve bölütlenmelidir. MSL gibi harekete dayalı deneylerde hareketi algılayabilmek için iki yol vardır. İlk yol hayvanın yüzdüğü havuzun dalga hareketlerinden, ikinci yol ise hayvanın labirent içindeki hareketlerinden hareket algılanabilir.

Labirent tanıma ve konumlandırma, labirentin geometrik şekli ile görüntü arka planından çıkarılan sonuca göre belirlenir. Bu tanıma, labirent ve kameranın deneyde sabit bırakılıp hareket edemeyeceği için deneyin başında yapılır.

Labirentin ışık renginin farenin rengine karşı iyi bir kontrast sergilemesi fare bölütlenmesine daha iyi imkan sağlayarak farenin görüntü içerisinde bulunmasını kolaylaştırmıştır. Farenin hareketlerini video dizileri içinde alan-zaman eksenleri boyunca izlemek çok önemlidir. Çünkü, farenin labirent içinde sergileyeceği hareketler farenin davranışlarını analiz etmemize olanak sağlayacaktır.

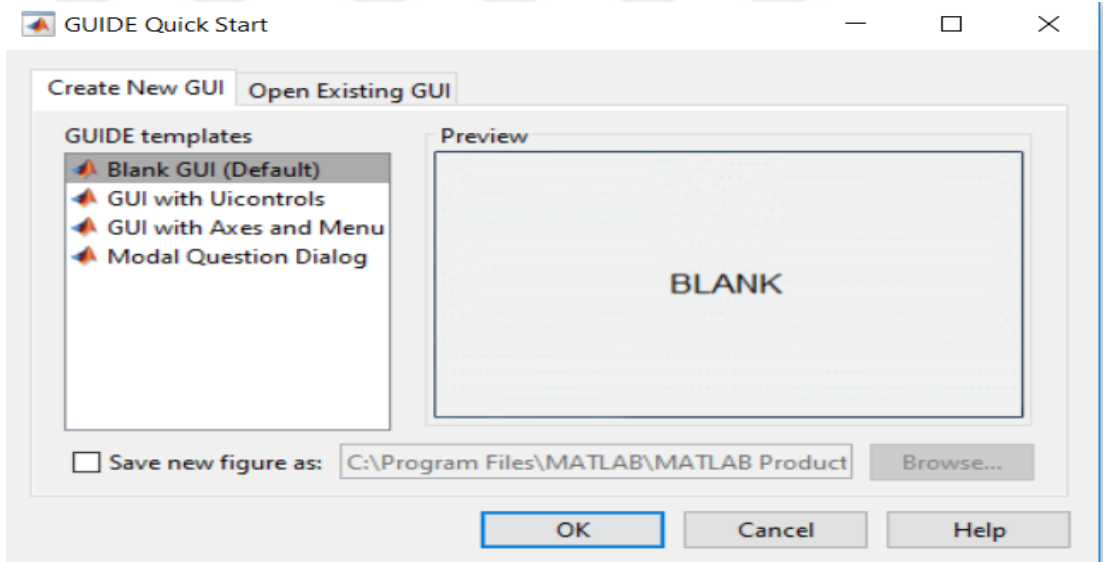
Video analizinin ikinci aşaması görüntüyü anlama, yorumlamadır. Farenin eylemlerinin analizi deneysel videolardan elde edilen veri ve bilgilerin ışığında yapılmalıdır. Farelerin labirent içinde hareket ederken çizdiği yörüngelerdeki verileri kullanarak video oynatılırken hareketlerin otomatik olarak saptanması ele alınmaktadır.

Bu çalışmada video analizinden elde edilen veri ve bilgileri daha etkin kullanabilmek için daha önce de belirtildiği gibi günümüzde yaygın bir sistem yazılımı programı olan Matlab tercih edilmiştir. Yazılımın Matlab ortamında gerçekleştirilmesinin temel sebebi Matlab'ın mühendislik uygulamalarında gittikçe artan yadsınamaz gücüdür. Yüksek fonksiyonelliği ve kullanışlılığı ile birlikte GUIDE uygulaması ile kullanılan Matlab GUI (Graphical User Interface – Grafik Kullanıcı Ara Yüzü (GKA)) yazılım geliştirme ortamının hem programlamada hem de kullanımda görsel ve kolaylıkla kullanılabilir olmasıdır. Deney için örnek bir video kaydı kullanılmıştır. Burada daha önce Matlab ve Matlab GKA ortamında yazılan Automated phenotyping of behavior (autotyping) Toolbox [20] yazılımı düzenlenmiş ve geliştirilerek kullanılmıştır.

3.1. Matlab GUIDE ile GKA Oluřturulması

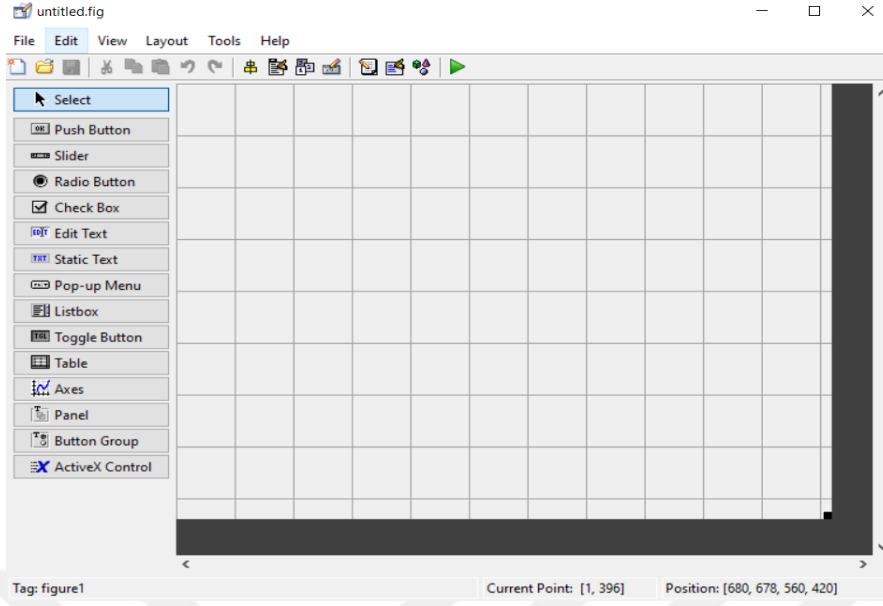
GUIDE Matlab'ın GKA tasarımcılarına sunduđu içerisinde çeřitli araçlar içeren ve tasarımcıya kolaylık sağlayan bir grafiksel GKA geliştirme ortamıdır. GUIDE kullanılarak tıkla ve sürükle-bırak tekniđi ile GKA'ya nesneler (örneğin düğmeler, metin kutuları, liste kutuları, grafikler v.s.) kolaylıkla eklenebilir. Ayrıca, eklenen nesnelerin hizalanması, tab sırasının deđiřtirilmesi, görsel ayarlar üzerinde deđiřiklikler yapılması da bu ortamın tasarımcılara sunduđu imkânlardan bazılarıdır [21].

Matlab GKA'yı çalıştırmak için iki seçenek vardır. Matlab komut penceresinde (command window) guide komutu vermek ya da Start düğmesi tıklanarak MATLAB/GUIDE komutu verilir. Bu adımdan sonra Şekil 3.1'deki gibi bir pencere açılır.



Şekil 3.1. GKA Başlangıç Ekranı

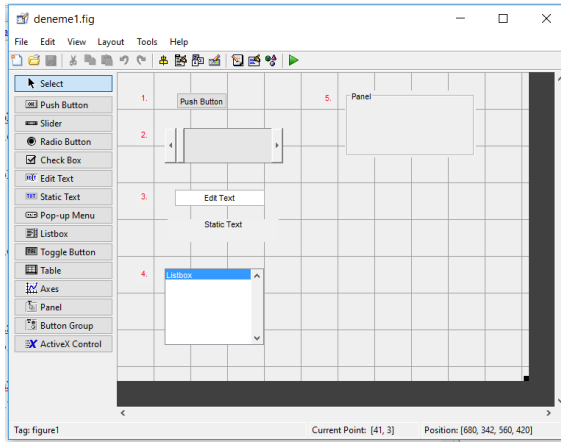
Açılan pencereden Blank GUI seçeneđi seçilerek yeni bir GKA tasarımı yapılmaya başlanır. Şekil 3.2'teki GUIDE LAYOUT Editor (GUIDE Görünüm Çalışma Alanı) penceresi açılarak tasarım yapılmaya başlanır.



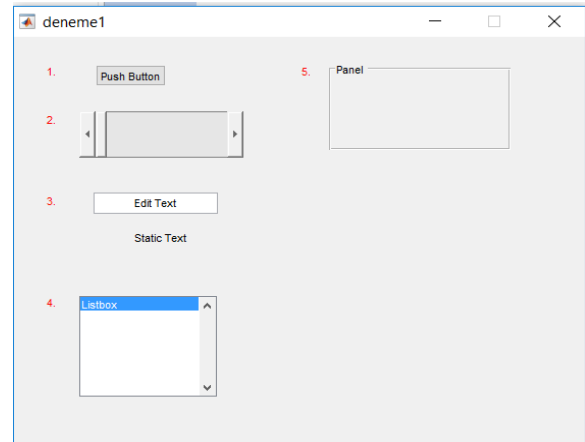
Şekil 3.2. GKA Çalışma Alanı Penceresi

3.1.1. Nesneleri Çalışma Alanı Penceresine Yerleştirme

Oluşturulan ara yüzde kullandığımız metot gereği nesneler Gui çalışma alanına sürükleyip bırakarak yerleştirilmiştir. Şekil 3.3.(a)'da gösterilen örnek çalışma alanına yerleştirilmiş olan nesneler görülmektedir. Şekil 3.3.(b)'de ise bu örnek çalışma alanına bağlı olarak oluşan ara yüz görülmektedir. Kullanılan nesneler bu tez çalışması boyunca kullandığımız nesnelerdir. Bu bölümde sadece tasarladığımız ara yüzde kullandığımız nesneler tanıtılacaktır.



(a)



(b)

Şekil 3.3. (a) GKA örnek çalışma alanı. **(b)** örnek çalışma alanına bağlı olarak oluşan ara yüz

3.1.2. Matlab GKA’da Kullanılan Nesneler

Bu tez çalışması için hazırlanan Matlab Gui’de tasarlanan ara yüzde düğme (push button), metin kutusu (edit text), sabit metin (static text), kaydırıcı (slider), liste kutusu (list box), panel kullanılmıştır.

3.1.2.1. Düğme (Push Button)

En çok kullanılan ve neredeyse her ara yüz üzerinde görebileceğimiz bir Matlab GKA nesnesidir. Normal bir buton özelliği göstermektedir. Bu butona basıldığında yapılacak işlemler için yazılan kodlar bu buton ile ilgili geri bildirimlerin (callback) altına yazılmaktadır. Bu çalışmada video yükleme, arka plan bölütlemesi, Roi seçimi gibi birçok işlemi gerçekleştirmek için kullanılmıştır.

3.1.2.2. Kaydırıcı (Slider)

Kullanıcıdan bir giriş değerini kaydırmak suretiyle rahatlıkla o değerin alınmasına imkân veren bir Matlab GKA nesnesidir. Bu çalışmada slider nesnesi bölütleme yaparken isteğe bağlı eşik değeri (threshold) seçimi için kullanılmıştır.

3.1.2.3. Metin Kutusu (Edit Text)

Kullanıcının ara yüzde programa bilgi ve veri girişi yapılması için edit text uygulaması en sık kullanılan nesnedir. Burada yapılan deneyleri isimlendirmek, videonun başlama ve bitiş sürelerini belirlemek amacıyla kullanılmıştır.

3.1.2.4. Sabit Metin (Static Text)

Bir Matlab GKA uygulaması çalıştırıldığı zaman çıkış verilerinin ekrana yazdırılması için veya ekranda sürekli görünmesi istenen tasarımın bir unsuru olan sabit yazıların yazılması için kullanılır. Burada videonun geçerli konumunun öge dizininin ekranda yazdırılması için ve ara yüz kullanılırken hangi adımların izleneceğini göstermek için kullanılmıştır.

3.1.2.5. Liste Kutusu (List Box)

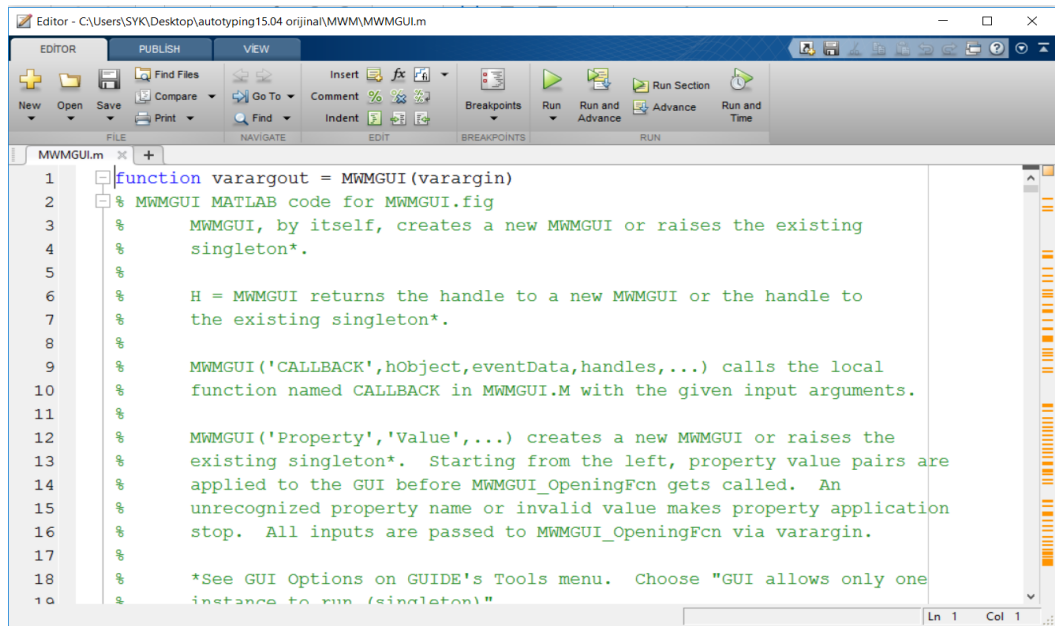
Kullanıcıya bilgi vermek için kullanılabildiği gibi farklı durumları listeden seçmek için de kullanılan sabit bir liste kutusu olan bir nesnedir. Bu çalışmada seçilen klasör içindeki farklı video dosyalarını göstermek ve seçmek için kullanılmıştır.

3.1.2.6. Panel

Matlab GKA ara yüzünde kullanılan nesnelerin kullanıcıya daha anlamlı ve güzel görünmesini sağlayan ve tasarımcıya tasarımında kolaylık sağlayan bir nense olmakla beraber GKA ara yüzünde kullanılan nesnelerin gruplanması ve beraber gösterilmesi için kullanılır.

3.1.3. Matlab GKA Ara Yüzlerine Program Yazma

Bir GKA ara yüzünün programlanması demek çalışmanın kaydedildiği isimle aynı zamanla ve aynı adda oluşturulan “.m” uzantılı dosya içerisine kodlama satırlarının eklenmesi demektir. Bu dosyanın içeriğini görebilmek, değişiklik yapabilmek için GUIDE çalışma ekranı penceresinden View/M-File Editor komutu işletilebilir [22]. Bu komut çalıştırıldıktan sonra Şekil 3.34’teki pencere açılacaktır.



Şekil 3.4. GUIDE Ara Yüzünün Programlanması

Şekil 3.4'teki pencerede tasarlanan Matlab GKA'ya ait kodlar görülmektedir. Burada birçok kodun hazır olarak eklendiği görülmektedir. Görülen bu kodlar Matlab GUIDE tarafından otomatik olarak eklenmiştir. Bu çalışmada kullanılan nesnelere ait callback (geri bildirim) isimli alt program parçalarına o nesneyle ilgili kodlar yazılır. Kullanılan nesnelere ait callback'in bulunduğu satırı bulmak için callback'i açılmak istenen nesnenin üstüne sağ tıklanarak View Callbacks/Callback sekmesi tıklanarak o nesneye ait callback açılır.

Fonksiyon içerisine gönderilen giriş değerleri ile alakalı olarak bilinmesi gereken iki çeşit değişken vardır. Bunlar sırasıyla *nargin* ve *varargin* değişkenleridir.

- *nargin* değişkeni: Fonksiyona (ya da GKA uygulamasına) dışarıdan gönderilen toplam parametre sayısını tutar. “Number of Function Arguments In” söz öbeğinin kısaltılmış halidir. Böylelikle fonksiyon kendisine gönderilen parametre sayısına göre farklı işlemler yapabilir [23].

- *varargin* değişkeni: Fonksiyona gönderilen değerlerin alınmasını sağlar. Hücre dizisi yapısında olduğu için değerlerin alınması için dizi indislerinin “{“ ”}” işaretleri arasında yazılması gerekmektedir. Hücre yapısının tercih edilmesinin sebebi birçok türde farklı verinin (integer, float, string, vs.) öbeklenerek gönderilebilmesidir.

Matlab GKA uygulaması sonlandırılırken fonksiyonunun içinden dışarıya gönderilen çıkış değerleri ile alakalı olarak bilinmesi gereken iki çeşit değişken vardır. Bunlar sırasıyla *nargout* ve *varargout* değişkenleridir.

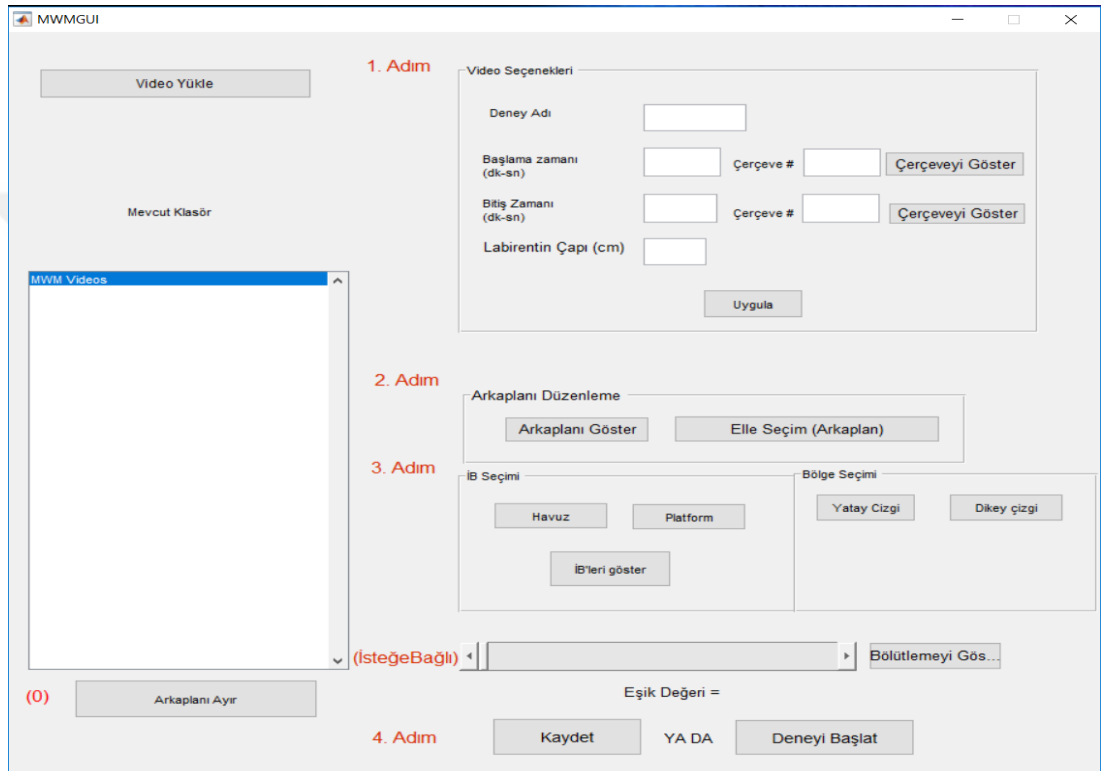
- *nargout* değişkeni: Fonksiyondan (ya da GKA uygulamasından) dışarı gönderilen toplam parametre sayısını tutar. “Number of Function Arguments Out” söz öbeğinin kısaltılmış halidir. Böylelikle fonksiyon kendisinden gönderilen parametre sayısına göre farklı işlemler yapabilir [23].

- *varargout* değişkeni: Fonksiyondan dışarıya değerlerin gönderilmesini sağlar. Hücre dizisi yapısında olduğu için değerlerin bir değişkene atanması sırasında dizi indislerinin “{“ ”}” işaretleri arasında yazılması gerekmektedir.

Programda bahsi geçen bu dört komut kullanılarak bir fonksiyonun ne kadar miktarda girdi kabul edebileceğini, ne kadar miktarda çıktı üretebileceğini ve bu girdi - çıktıları hangi değişkenlerde saklayacağını, arzulananndan daha az girdiyle çağrıldığında programın ne yapması istendiğini belirleyerek fonksiyon oldukça esnek ve hatasız çalışabilecek bir duruma getirilebilir.

3.2. Tasarlanan Ara Yüz

MSL analizini yapmak için tasarlanan ara yüzde ilk olarak deney videosu yükleme işlemi gerçekleştirilir. Programda ‘.avi, .mpeg, .mpg, .mp4’ gibi farklı video formatındaki video dosyaları oynatılabilmektedir. Kullanıcı istediği video dosyalarını liste kutusundan seçebilmektedir.



Şekil 3.5. Tasarlanan ara yüz uygulaması

Video yüklendikten sonra farelin takibini yapabilmek için arka plan belirlenir. Daha sonra videonun işleneceği zaman aralığı ve havuzun çap bilgileri kullanıcı tarafından elle girilir. Önceden belirlenen arka plan üzerinden platform ve labirent için İB'ler tanımlanarak labirent sanal olarak dört bölgeye ayrılır. İsteğe bağlı olarak kaydırıcı üstünden bir değer seçilerek bölütleme yapılabilir. Bu ön işlemler tamamlandıktan sonra analiz başlatılır. Video işleme bittikten sonra farelin gerçek zamanlı çizdiği yörüngeyle beraber platforma çıkana kadar aldığı yol ile süre ve farelin hız bilgilerinin olduğu yeni bir pencere açılır. Tasarlanan ara yüz uygulaması Şekil 3.5'te gösterilmiştir.

3.3. Bölütleme ve İzleme

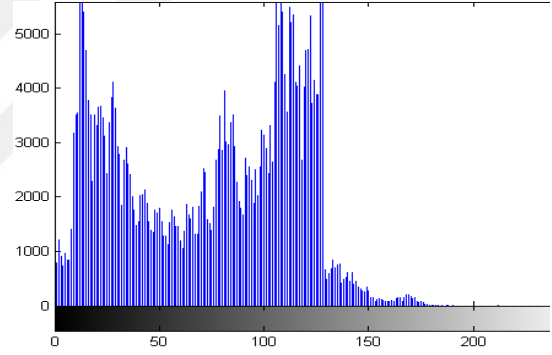
Video analizine başlarken yapılması gereken ilk işlem video çerçeveleri içinde labirent ve farenin tanınması ve takip edilmesi işlemidir. Bu işlem yapılırken fare labirentten bölütlenmelidir. Daha sonra her bir video çerçevesinde farenin takibi yapılmalıdır. Yapılan bu takiplerin sonucunda elde edilen verilerden deneyin analizi yapılmaktadır.

3.3.1. Su Labirenti Bölütlemesi

MSL analizinin ilk aşaması labirentin yerinin belirlenmesidir. Labirenti bulabilmek için öncelikle labirentin arka plandan ayrılması gerekir. Bu bölütleme işlemi, bir eşikleme algoritması kullanılarak gerçekleştirilir.



(a)



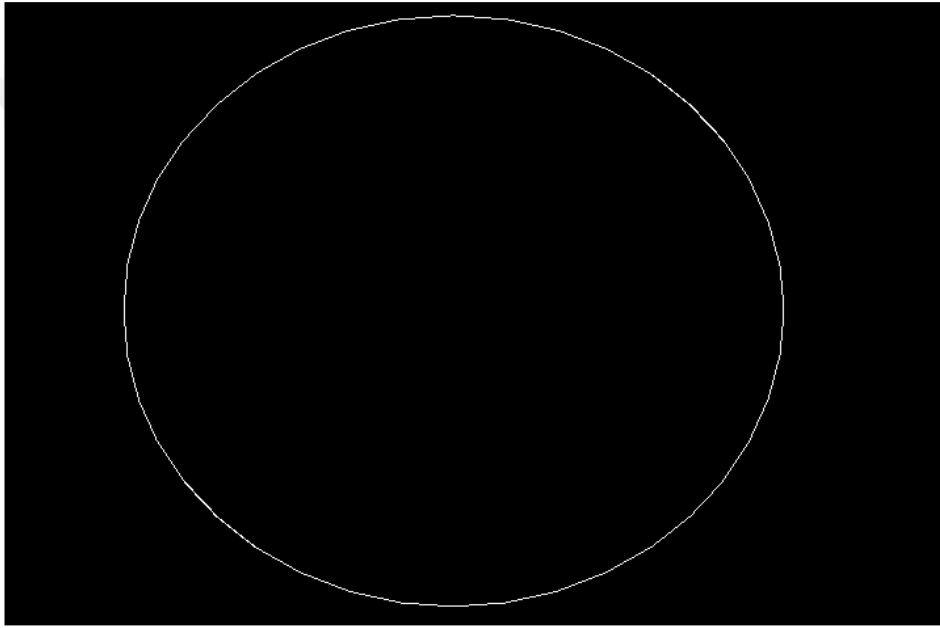
(b)

Şekil 3.6. (a) Havuzun ortasında yüzen siyah bir fare ile örnek video çerçevesi. (b) Video çerçevesinin histogramı.

Şekil 3.6’da görüldüğü gibi histogramın sol tarafındaki (daha koyu) birikim, arka plan (ek olarak fare) ve sağ tarafı (daha açık) su labirentidir. Deneyde kullanılan video kamera için arka plan ile labirent arasında iyi bir kontrast bulunmaktadır. Labirent, suyla doldurulmuş ve suya opak renklendirici ilave edilmiş (Örneğin koyu renkli fareler için süt kullanılmıştır) bir havuzdan ibarettir. Bu yüzden, video çerçevelerinin gri seviyeli histogramları iki dorukludur (bimodal) ve iki temel doruk (sırasıyla arka plan ve labirent) birbirinden iyi bir şekilde ayrılmıştır. Uygulanan yöntem histogramda en yüksek iki yerel en büyüğü (maksimum) bulur ve ardından eşik değeri, aralarındaki en küçük (minimum) değer olarak belirlenir.

Algoritma her çerçeve için, labirentin eşleştirilmesi, tanınması ve karakterize edilmesi yinelenir. Video okuması çerçeve çerçeve olarak yapıldığı için labirent 2 Boyut (2B) şekline dayanarak yapılır. Ekrandaki her bölge için algoritma, labirent şeklinin olup olmadığını denetler.

Hareketin yörünge fonksiyonu, labirent şeklini karakterize ve tespit etmek için kullanılır. Bu fonksiyon, çeşitli şekillerde üretilebilen bir konturun fonksiyonel bir temsidir. Ancak bu durumda, uzaklık mesafesi-açı fonksiyon değişkenleri olarak kullanılır. Buradaki yörünge fonksiyonu, konturun geometrik merkezinden sınırına kadar olan mesafeyi bir açı fonksiyonu olarak göstermektedir.

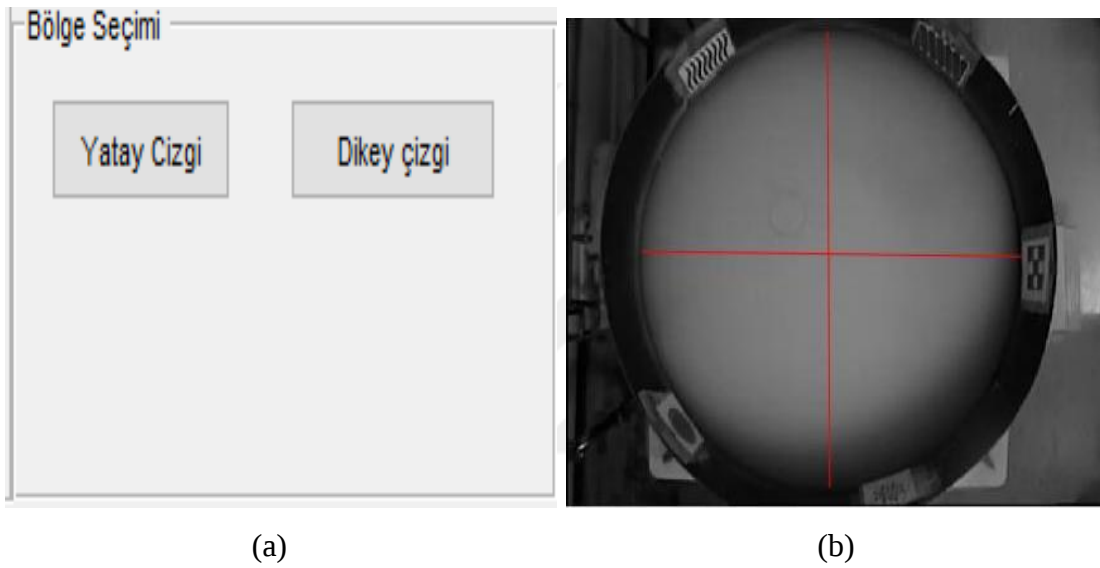


Şekil 3.7. Eşik değerine dayalı bir video çerçevesinin bölütlemesi, Morris su labirentine ait bağlı bölgenin konturu.

Mesafe-açı fonksiyon değişkenleri, kartezyen koordinat sisteminin polar koordinat sistemine dönüştürülmesiyle bulunur. Şeklin geometrik merkezini koordinatların orijini olarak kullanmak bu dönüşümlerin doğru bir şekilde yapılmasını sağlamaktadır. Bununla beraber mesafeler en büyüğe göre normalize edilir. Böylece dönüşümle beraber ölçek de değişmez. Ancak hareketin yörünge fonksiyonu halen orijinal şeklinin yönüne bağlıdır. Dönüşe göre değişmeyen bir fonksiyon üretebilecek bir başlangıç noktası seçebilmek için önerilen birkaç yöntem vardır [16]. Burada kullanılacak yöntem; en uygun fonksiyonu bulana kadar şekillerden birini diğerinin üzerinde döndürüp işleyerek bulunmasıdır.

Labirent tanımladıktan sonra, korelasyon testine uygun olarak konturu labirent modeli yörünge fonksiyonu ile eşleştirerek labirent konumu (geometrik merkez), ebat (ortalama çap) ve hareket yörüngesi fonksiyonu kullanılarak karakterize edilir.

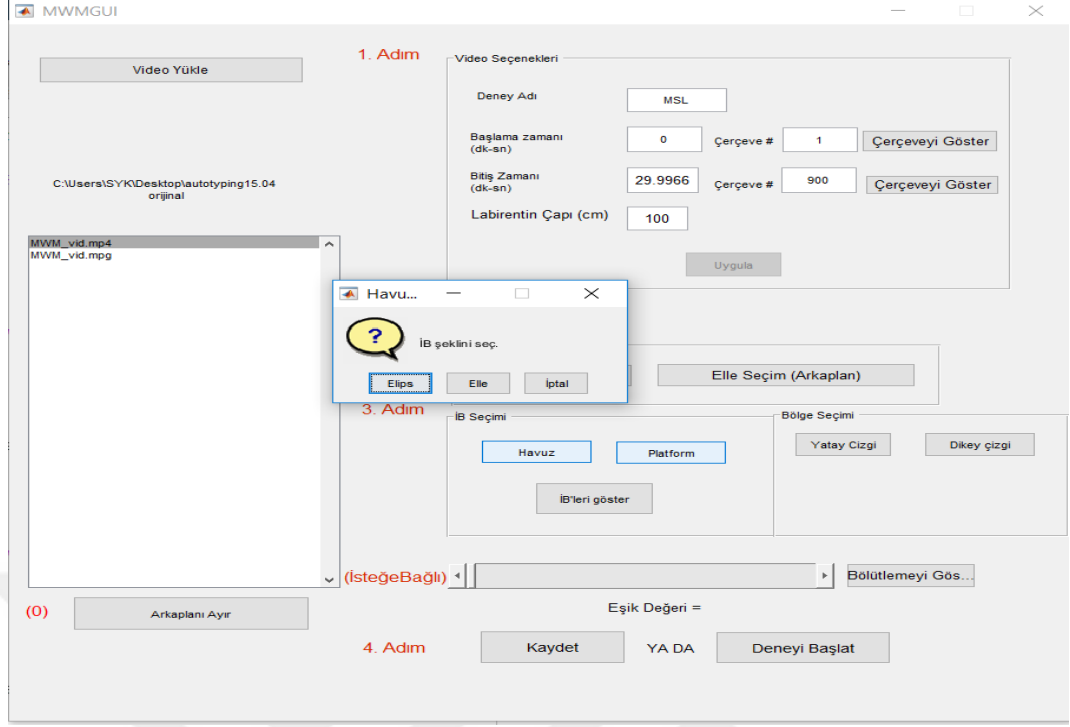
İlk önce labirent sanal olarak dört bölgeye ayrılır. Bu işlem Matlab Gui'de bölge seçimi panelindeki düğmeler (push button) kullanılarak gerçekleştirilir. Havuzun yönlendirilmesini havuzun kuzeyi ile kameranın görüntüleme alanının kuzey yönüyle eşleştirdiğini varsayarak doğudan batıya doğru *imline* komutu ile çizilen yatay çizgi ile kuzeyden güneye yine *imline* komutu ile çizilen dikey çizgi labirentin sanal olarak dört ayrı bölgeye ayrılmasına olanak sağlar.



Şekil 3.8. (a) Bölge Seçimi Paneli. (b) Sanal olarak dörde bölünmüş labirent

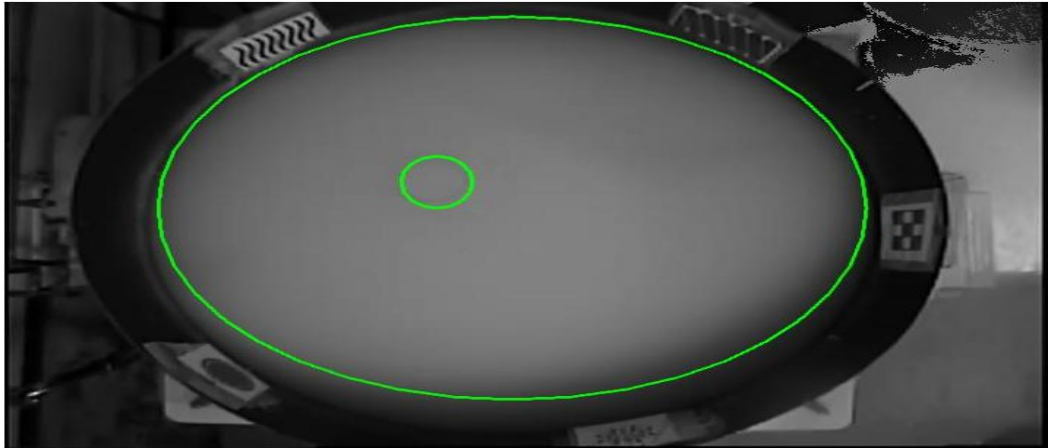
Havuzun yönlendirilmesini yine havuzun kuzeyi ile kameranın görüntüleme alanının kuzey yönüyle eşleştirdiğini varsayarak havuzu otomatik olarak kuzeybatı, kuzeydoğu, güneybatı ve güneydoğu bölgeleri olacak şekilde dört ayrı bölüme ayrılır. Burada labirentin kuzeybatı bölümü (platformun bulunduğu kısım) birinci bölge, güneybatı bölümü ikinci bölge, güneydoğu bölümü üçüncü bölge kuzeydoğu bölümü ise dördüncü bölge olarak adlandırılmaktadır.

Analiz yaparken bazen araştırmacı da video çerçevelerinde görülür. Dış etkileri en aza indirmek ve daha sağlıklı analiz yapabilmek için havuzun çevresine ve kaçış platformuna İB'ler tanımlanır. Bu işlem Matlab GKA ara yüzünde bulunan İB Seçimi panelindeki düğmeler aracılığıyla sağlanır. Tasarlanan ara yüzde İB seçimleri değer girilmeden elle çizdirilerek otomatik olarak yapılmaktadır.



Şekil 3.9. İB Seçimi için hazırlanan İB Seçim Paneli

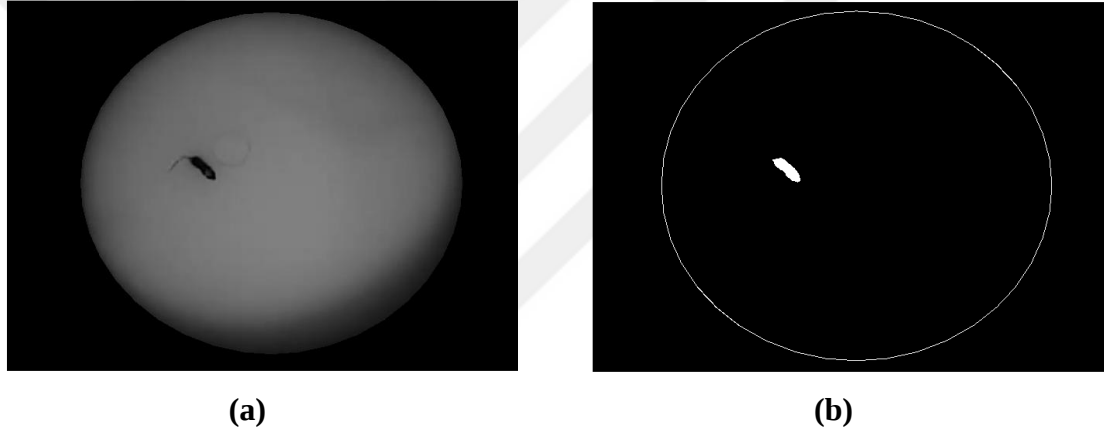
Ara yüz havuzun ve platformun şeklinin daire biçiminde olduğu göz önüne alınarak Matlab'ın *imellipse* komutu kullanılarak elips şeklinde çizdirilerek ya da *imfreehand* komutu kullanılarak serbest bir biçimde İB çizdirecek biçimde hazırlanmıştır. Platform ve labirent için ayrı olarak İB seçimi yapılmaktadır. İB'ler seçildikten sonra İB'leri göster düğmesi ile seçilen İB'leri görmeye olanak sağlanmıştır.



Şekil 3.10. İB Seçimi yapıldıktan sonra seçilen İB'lerin gösterimi

3.3.2. Fare İzleme

Video analizindeki bir sonraki adım, fareyi labirente bulmaktır. Analiz edilen videolarda farenin rengi genellikle siyah, havuzdaki suyun rengi ise fareyi daha iyi ayırt edebilmek için beyaz renklidir. Havuza beyaz rengi verebilmek için suya, suda çözülebilen bir tür beyaz boya eklenir. Farenin video çerçevelerindeki görüntüsü havuz içerisindeki en karanlık pikselleri oluşturacaktır. Buna göre görüntü histogramına bağlı olarak bir eşik değeri seçilir. Gri seviyede farenin piksel değerleri suyun piksel değerlerinden düşük olacağından düşük bir piksel değeri seçilir. Şekil 3.11’de, seçilen eşik değeri sonucunda fareyi bulmamızı sağlayan bu görüntü bölütleme algoritmasının sonuçları gösterilmektedir.

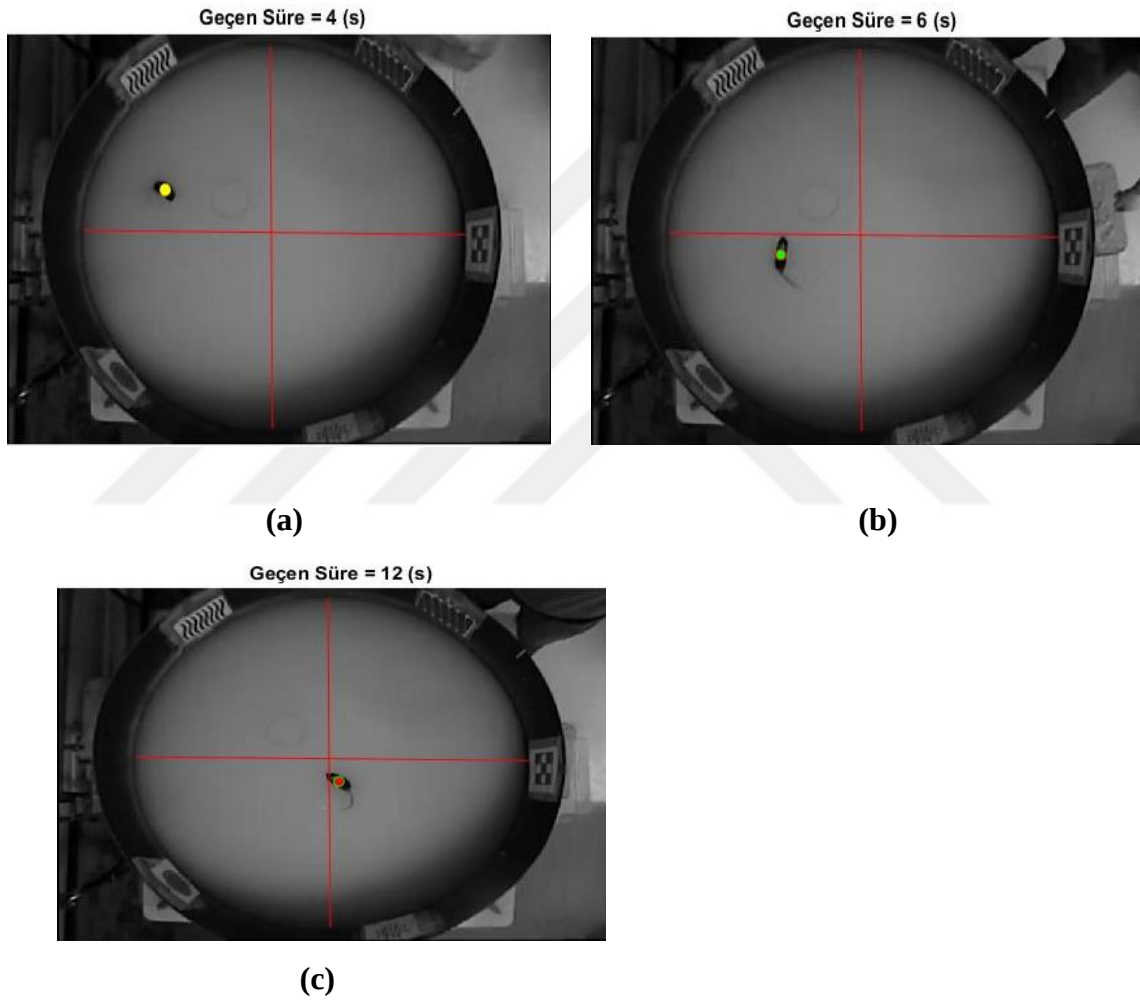


Şekil 3.11 (a) Arka plan çıkarma işleminden sonraki görüntü. **(b)** eşik değerini kullandıktan sonraki fare bölütlemesi.

Yukarıda değinildiği gibi deney fareleri genellikle siyah renkli olup bu çalışmada kullanılan örnek videoda kullanılan fare siyah renklidir. Ancak bu her zaman için geçerli bir ifade değildir. Çünkü deneysel fare ırkları genellikle beyaz renklidir. Beyaz fareleri havuzun içinde ayırt edebilmek için bu sefer suya suda çözünebilen koyu renkli bir tür boya eklenir. Buna göre siyah renkli fareler için kullanılan eşik değerinin tersi bir eşik değeri seçilir.

Fareyi tanımlayan bazı temel parametreleri video çerçevesinde oluşan görüntünün piksel bölgelerini analiz ederek ölçeriz. Bu parametreler boyut (piksel cinsinden), konum (ağırlık merkezi), yönlendirme açısı (ana eksen) ve şekil (uzunluk ve genişlik) bilgileri kullanılarak oluşturulur. Fare videonun zaman eksenini boyunca havuzun içinde yüzer. Bu yüzden farenin takibi Şekil 3.12’de görüldüğü gibi her video çerçevesinde yapılmalıdır.

Farenin hareket takibi, havuz içinde hareketi esnasında her hareketinde fareyi takip eden ve fare üzerinde oluşturulan bir noktasal ışınla yapılır. Bu ışın yoluyla takibi, farenin havuza bırakıldıktan sonraki her görüntü çerçevesinden farenin henüz havuza bırakılmadan önce kaydedilen ilk çerçevesi çıkarılarak yapılır. İlk çerçevede fare olmadığı için fare olan diğer çerçevelerden farkı alındıktan sonra geriye sadece fare kaldığı için farenin hareketini takip etmek daha kolay olmaktadır. Şekil 3.12’te de görüldüğü gibi farenin davranışlarını analiz etmek için dört bölüme ayrılan havuzda farenin hareket ettiği her bir bölümde noktasal ışığın rengi değişmektedir.

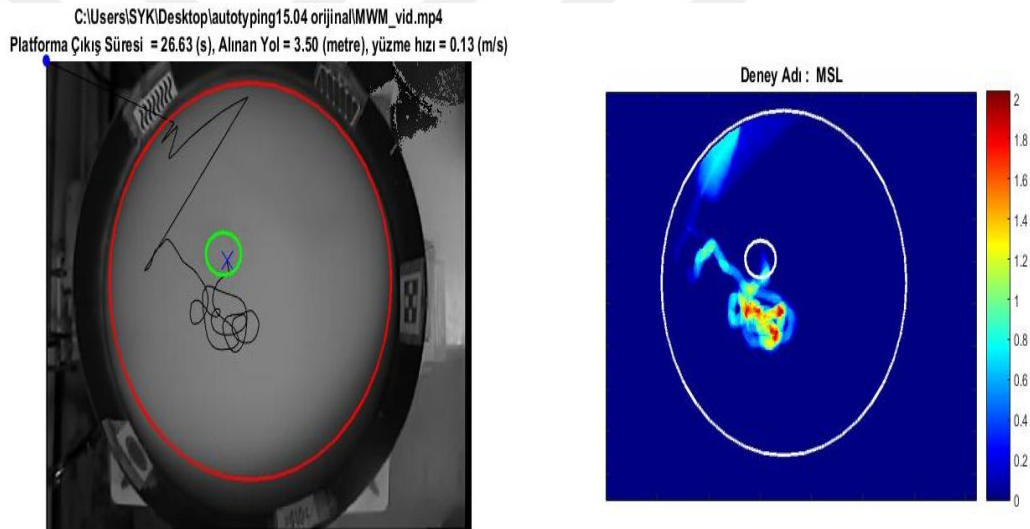


Şekil 3.12. (a) Farenin birinci bölgedeyken takibi (fare sarı renkli bir ışınla takip edilir). **(b)** Farenin ikinci bölgede takibi (yeşil ışın). **(c)** Farenin üçüncü bölgede takibi (kırmızı ışın).

MSL testi laboratuvar farelerinde mekânsal öğrenmeyi ve hafızayı araştırmak için kullanılan bir davranış testidir. Deney sonuçlarını mümkün olan en iyi şekilde analiz edebilmek için deneyde kullanılan farelerin labirente sergilemiş olduğu her davranış kayıt altına alınmalıdır. Labirentin sanal olarak dört bölgeye ayrılmasıyla mekânsal öğrenme ve

hafızanın ne kadar geliştiğini öğrenebilmek amaçlanmıştır. Ardı ardına yapılan deneylerden sonra farenin hangi bölgelerde daha çok zaman geçirdiği MSL analizinde en çok dikkat ettiği parametrelerden biridir. Eğer fare platforma çıkmadan önce en çok platformun konulduğu bölgede yüzüyorsa bu durum farenin mekânsal öğrenme ve hafızada iyi bir seviyede olduğunu gösterir. Aksi takdirde yapılan deneyin amacına bağlı olarak farenin mekânsal öğrenme ve hafızasının istenen seviyede olmadığı kanısına varılabilir.

Analizin son adımı yapılan deney sonuçlarını değerlendirmektir. Tasarlanan Matlab Gui ara yüzü vasıtasıyla kaydedilen videodan gelen verilerle analiz yapılır. Video işleme bittikten sonra farenin gerçek zamanlı çizdiği yörüngeyle birlikte platforma çıkana kadar aldığı yol ile süre ve farenin hız bilgilerinin olduğu Şekil 3.13'te görüldüğü gibi yeni bir pencere açılır. Elde edilen bu veriler ışığında analiz yapılır.



Şekil 3.13. Farenin gerçek zamanlı olarak çizmiş olduğu yörünge ve diğer deney bilgilerini gösteren pencere.

Platforma çıkış süresi Matlab GKA ara yüzünde tasarlanan Şekil 3.14'te gösterilen video seçenekleri panelinde girilen verilerden bulunur. Platforma çıkış süresi farenin platforma çıkış zamanından (bitirme zamanı) farenin labirente bırakıldığı zamanın (başlangıç zamanı) çıkarılmasıyla bulunur.

Video Seçenekleri

Deney Adı

Başlama zamanı (dk-sn)

Çerçeve #

Çerçeveyi Göster

Bitiş Zamanı (dk-sn)

Çerçeve #

Çerçeveyi Göster

Labirentin Çapı (cm)

Uygula

Şekil 3.14. Video Seçenekleri Paneli

Piksel değerlerinin metreye çevrilmesiyle farenin aldığı yol bulunur. Elde edilen farenin aldığı yol farenin platforma çıkış süresine bölünmesiyle farenin ortalama hızı m/sn cinsinden bulunur.

3.4. Deneysel Sonuç

Yapılan deneyde kullanılan örnek video kaydı [24] için farenin davranışları (aldığı yol, süre, çizdiği yörünge vs.) takip edilmiştir. Sanal olarak dört kısma ayrılan labirentte farenin hangi bölgede ne kadar süre kaldığı kayıt altına alınmıştır. Tablo 3.1’de kayıt altına alınan bu bilgiler görülmektedir.

Tablo 3.1. Deneysel sonuçlar

Bölgeler	Kalma Süresi (sn)	Platforma Çıkma Süresi (sn)	Ortalama Hız (m/sn)	Alınan Toplam Yol (m)
1.Bölge	4,26	16,65	0,11	1,95
2.Bölge	11,56			
3.Bölge	0,83			
4.Bölge	0			

Farenin tüm bölgelerde aldığı süreler, platforma çıkış süresi, ortalama hızı ve aldığı toplam yol Tablo 3.1’de görülmektedir. Bu bilgilere bakılarak ilk denemeler olduğundan dolayı farenin eğitiminin tam olarak tamamlanmadığı anlaşılmaktadır. Farenin platformun

bulunduđu birinci ve platforma yakın olan ikinci bölgede daha çok vakit geçirmesi; farenin platformu bulması, hafıza ve öğrenme testi için önem arz etmektedir. Farenin üçüncü ve dördüncü bölgelerde neredeyse hiç süre geçirmemesi belleğin gelişmesi için platforma yakın olan bölgede farenin daha fazla sürede bulunması farenin eğitim süreci için beklenen davranışı sergilediğı söylenebilir. Farenin eğitim sürecinin devam ettirilip istenen seviyeye ulaşıldığında farenin platformu bulmak için kat ettikleri yüzme mesafesi ve platformu bulma süresi azalacak; farenin hızı ise artacaktır.



4. SONUÇLAR

Bu çalışmada örnek bir video kaydı [24] kullanılarak MSL testi analizi gerçekleştirilmiştir. Yapılan deneysel çalışma ile ilgili tüm analizler Matlab ortamında geliştirilen bir program ile yapılmıştır. İlk olarak labirent tanımlanarak İB'ler aracılığıyla fare labirentten bölütlenmiştir. Böylece fare, tüm video çerçevelerinde takip edilmiştir. Labirent sanal olarak dört ayrı bölüme ayrılarak farenin her bölümde geçirdiği süre gerçek zamanlı olarak kayıt altına alınmıştır. Ayrıca farenin hareket ederken çizdiği yörüngeler de kayıt altına alınmıştır. Kullanılan örnek video kaydı ile ilgili bilgisayar ortamında şimdiye kadar yapılan çalışmalarda istenilen ve beklenen sonuçlara ulaşılmıştır.

Elde edilen bilgilere bakılarak ilk denemeler olduğundan dolayı farenin eğitiminin tam olarak tamamlanmadığı anlaşılmaktadır. Çünkü farenin platformun bulunduğu birinci bölgede geçirdiği süre platformun bulunmadığı diğer bölgelerde geçirdiği toplam süreden daha azdır. Ancak farenin üçüncü ve dördüncü bölgelerde neredeyse hiç süre geçirmemesi farenin eğitim süreci için farenin beklenen davranışı sergilediği söylenebilir. Farenin eğitim sürecinin devam ettirilip istenen seviyeye ulaşıldığında farenin aldığı yol ve süre azalacak; farenin hızı ise artacaktır.

KAYNAKLAR

- [1] **Morris RGM**, Spatial localization does not require the presence of local cues. *Learn Motiv.* 1981. 12 pp. 239-260
- [2] **Morris RGM**, Developments of a water-maze procedure for studying spatial learning in the rat. *Journal of Neuroscience Methods.* 1984. 11, pp. 47-60
- [3] **R. Brandeis, Y. Brandys, S. Yehuda**, The use of the Morris water maze in the study of memory and learning, *Int. J. Neurosci.* 48 (1989) 29–69. *of Neuroscience Methods.* 1984. 11, pp. 47-60
- [4] **Y. Lamberty, A.J. Gower**, Spatial processing and emotionality in aged NMRI mice: a multivariate analysis, *Physiol. Behav.* 54
- [5] **R. Morris**, Developments of a water-maze procedure for studying spatial learning in the rat, *J. Neurosci. Methods* 11 (1984) 47–60.
- [6] **S. Ikonen, B.H. Schmidt, P. Riekkinen Jr.**, Characterization of learning and memory behaviors and the effects of metrifonate in the C57BL strain of mice, *Eur. J. Pharmacol.* 372 (1999) 117–126.
- [7] **L.W. Means, J.L. Higgins, T.J. Fernandez**, Mid-life onset of dietary restriction extends life and prolongs cognitive functioning, *Physiol. Behav.* 54 (1993) 503–508.
- [8] **R. D’Hooge, F. Franck, L. Mucke, P.P. De Deyn**, Age-related behavioural deficits in transgenic mice expressing the HIV-1 coat protein gp 120, *Eur. J. Neurosci.* 11 (1999) 4398–4402.
- [9] **Yun YJ, Lee B, Hahm DH, Kang SK et al.** Neuroprotective effect of palmulchongmyeong-tang and ischemia-induced learning and memory deficits in the rat. *Biol Pharm Bull* 2007;30:337-42.
- [10] **Lieben CK.J, Oorsouw KV, Deutz NEP, Blokland A.** Acute tryptophan depletion induced by a gelatin-based mixture impairs object memory but not affective behavior and spatial learning in the rat. *Behav Brain Res* 2004;151:53-64.
- [11] **Atmaca, V.; Ertuğrul,Ş.**, “Örme Kumaşlardaki Üretim Hatalarının Görüntü İşleme Teknikleri İle Otomatik Tespiti Ve Sınıflandırılması” <http://hdl.handle.net/11527/10056>
- [12] **Heijden, F.**, 1994. *Image Based Measurement Systems*, John Wiley & Sons, West Sussex.

- [13] **Sengur, A.; Turkoglu, I.; Cevdet Ince, M.,** "Image segmentation applications with unsupervised neural networks," Signal Processing and Communications Applications Conference, 2005. Proceedings of the IEEE 13th , vol., no.pp. 271- 274, 16-18 May 2005
- [14] akizilkaya.pamukkale.edu.tr/Bölüm4_goruntu_isleme.pdf
- [15] **PITAS, Ioannis,** Digital Image Processing Algorithms, 1st Edition, Prentice Hall, 1993
- [16] **GONZALES, C. Rafael and WOODS, Richard E.,** Digital Image Processing, 2nd Edition, Prentice Hall, 2001.
- [17] **L. Li and M. Leung,** "Integrating intensity and texture differences for robust change detection," IEEE Trans. Image Processing, vol. 11, pp. 105–112, Feb. 2002.
- [18] **P. Rosin,** "Thresholding for change detection," in Proc. Int. Conf. Computer Vision, 1998, pp. 274–279.
- [19] web.firat.edu.tr/iaydin/bmu357/bmu_357_bolum2.pdf
- [20] Patel TP, Gullotti DM, , Hernandez P, O'Brien WT, Capehart BP, Bass C, Eberwine JE, Abel T, Meaney DF. An open-source toolbox for automated phenotyping of mice in behavioral tasks. Frontiers in Behavioral Neuroscience. 2014 Oct; 8:349. PMC4189437 www.seas.upenn.edu/~molneuro/autotyping.html Copyright 2014, Tapan Patel PhD, University of Pennsylvania
- [21] **Uzunoğlu M,** Her Yönü İle Matlab, Türkmen Kitabevi, 2003.
- [22] **Kançullah A,** Yakın resim fotogrametrisinde ölçme sistemleri için matlab arayüz çalışması 2010
- [23] **İnan A,** Matlab Kılavuzu, Papatya Yayıncılık, 200
- [24] http://www.seas.upenn.edu/~molneuro/autotyping/autotyping14.8/sample_videos/MWM/MWM_vid.mpg

ÖZGEÇMİŞ

22 Aralık 1990 tarihi, Diyarbakır ili Silvan ilçesi doğumluyum. İlk, Orta ve Lise eğitimlerimi Diyarbakır ilinde tamamladım. 2008 – 2013 yılları arasında Gaziantep Üniversitesi Mühendislik Fakültesinde Elektrik – Elektronik Mühendisliği (İngilizce) eğitimi aldım. 2015 yılında Denizli 11. Motorlu Piyade Tugayında askerlik görevimi ifa ettim. 2015 yılında Fırat Üniversitesi Fen Bilimleri enstitüsü Telekomünikasyon ana bilim dalında yüksek lisans eğitimine başladım. 2016 yılından beri, Gençlik ve Spor Bakanlığı Kredi ve Yurtlar Kurumunda çalışmaktayım.

