ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

GPS+GSM+GPRS TEKNOLOJİLERİ İLE GERÇEK ZAMANLI TAKİP SİSTEMİ ÜZERİNDE GEOFENCE MEKANİZMASININ TASARLANMASI VE GPS KONUM VERİLERİNDEKİ HATA ORANININ DÜŞÜRÜLMESİ

Zeynep ÖZDEMİR

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2019

Her hakkı saklıdır

TEZ ONAYI

Zeynep ÖZDEMİR tarafından hazırlanan "**GPS+GSM+GPRS Teknolojileri ile Gerçek Zamanlı Takip Sistemi Üzerinde Geofence Mekanizmasının Tasarlanması ve GPS Konum Verilerindeki Hata Oranının Düşürülmesi**" adlı tez çalışması 02/01/2019 tarihinde aşağıdaki jüri tarafından oy birliği ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Bülent TUĞRUL

Jüri Üyeleri :

Başkan : Doç. Dr. İhsan Tolga MEDENİ

Yıldırım Beyazıt Üniversitesi Yönetim Bilişim Sistemleri Anabilim Dalı

Üye :Prof. Dr. İman Askerbeyli

Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Üye :Dr. Öğr. Üyesi Bülent TUĞRUL

Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım.

Prof. Dr. Atila YETİŞEMİYEN

Enstitü Müdürü

ETİK

Ankara Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez raporu içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

02/01/2019



Zeynep ÖZDEMİR

ÖZET

Yüksek Lisans Tezi

GPS+GSM+GPRS TEKNOLOJİLERİ İLE GERÇEK ZAMANLI TAKİP SİSTEMİ ÜZERİNDE GEOFENCE MEKANİZMASININ TASARLANMASI VE GPS KONUM VERİLERİNDEKİ HATA ORANININ DÜŞÜRÜLMESİ

Zeynep ÖZDEMİR

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Bülent TUĞRUL

Tarih boyunca insanlar nerede olduklarını merak etmişlerdir. Bu merak sonucunda GNSS konum belirleme sistemleri bulunmuştur. En yaygın olarak kullanılanı GPS(Global Positioning System) teknolojisidir. GPS uydularından yayılan sinyalleri yeryüzünde GPS alıcıları tespit ederek kendi konumunu hesaplar. Çalışmada elde edilmiş olan GPS verisi GSM ve GPRS yolu ile uzak bir sunucuya gönderilerek gerçek zamanlı bir takip sistemi haline getirilmiştir. Bilgisayar, telefon gibi cihazlardan harita ekranı gözlemlenerek GPS cihazının bulunduğu konum anlık olarak izlenebilir. Ek olarak cihazın belirlenen bir konumdan uzaklaştığında uyarı vermesi için Geofence alanı kullanılmaktadır. Harita üzerinde önceden belirlenmiş olan bir poligon kullanılabileceği gibi uygulama esnasında çizilen polygon alanı da Geofence olarak kullanılabilir. Uygulama ekranında Geofence alanından çıkışta sesli bir şekilde uyarı verilmektedir. GPS her zaman tam doğru konumu göstermez, belirli bir hata payına sahiptir. Bu durum uygulamanın yanlış alarmlar vermesine ve hatalı olan konumla çalışmasına sebep olur. Bu hatayı azaltmak amacı ile kalman filtresi, hareketli ortalama filtresi ve lojistik regresyon yöntemleri kullanılmıştır. GPS verisinin alınamadığı durumlar için de LSTM, GRU ve ARIMA algoritmaları kullanarak eksik veriler tahmin edilip sistemin kesintisiz bir şekilde çalışması sağlanmıştır. Hata azaltma ve bir sonraki konumu tahmin amacı ile kullanılmış olan algoritmaların sonuçları karşılaştırılmış doğruluk oranları tespit edilmiştir. Konumun gerçeğe en yakın şekilde tespit edilebilmesi için teknik ve öneriler sunulmaktadır.

Ocak 2019, 57 sayfa

Anahtar Kelimeler: GPS, Geofence, kalman filtre, ARIMA, LSTM, gerçek zamanlı takip sistemi

ABSTRACT

Master Thesis

DESIGN OF GEOFENCE MECHANISM ON REAL TIME TRACKING SYSTEM WITH GPS + GSM + GPRS TECHNOLOGY AND REDUCING THE ERROR RATE IN GPS LOCATION DATA

Zeynep ÖZDEMİR

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Asst. Prof. Dr. Bülent TUĞRUL

Throughout history, people have wondered where they are. As a result of this curiosity, GNSS geolocation systems were found. The most widely used GPS (Global Positioning System) technology. The signals emitted from GPS satellites calculate their location by detecting GPS receivers on earth. The GPS data obtained in the study has been sent to a remote server via GSM and GPRS and turned into a real time tracking system. The location of the GPS device can be monitored instantaneously by observing the map screen from devices such as computers and phones. In addition, the Geofence field is used to alert the device when it is away from a specified position. A polygon that is predetermined on the map can be used or the polygon area drawn during the application can also be used as Geofence. On the application screen, a warning is issued in the exit from the Geofence field. GPS does not always show the exact correct position, but has a certain margin of error. This causes the application to give false alarms and to work in the wrong position. Kalman filter, moving average filter and logistic regression methods were used to reduce this error. In cases where GPS data could not be obtained, LSTM, GRU and ARIMA algorithms were used to estimate missing data and to ensure that the system was run continuously. The results of the algorithms used for the purpose of error reduction and the prediction of the next location were compared with the comparison accuracy values. Technical and recommendations are provided to determine the location as close as possible to the reality.

January 2019, 57 pages

Key Words: GPS, Geofence, kalman filter, ARIMA, LSTM, real time tracking system

TEŞEKKÜR

Çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarıyla destekleyen, akademik ortamda olduğu kadar insani ilişkilerde de engin fikirleriyle yetişme ve gelişmeme katkıda bulunan danışman hocam sayın Dr. Öğr. Üyesi Bülent TUĞRUL'a (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı), çalışmalarım süresince desteklerini esirgemeyen değerli bölüm başkanımız sayın Prof. Dr. Refik SAMET'e (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) en derin duygularla teşekkür ederim.

Çalışmalarım süresince maddi, manevi pek çok fedakârlık göstererek beni destekleyen ve her anımda yanımda olan aileme ve arkadaşlarıma tüm içtenliğimle minnettarlığımı belirterek, en derin duygularla teşekkür ederim.

Zeynep ÖZDEMİR
Ankara, Ocak 2019

İÇİNDEKİLER

TEZ ONAY SAYFASI
ETİK ..... **i**
ÖZET ..... **ii**
ABSTRACT ..... **iii**
TEŞEKKÜR ..... **iv**
SİMGELER DİZİNİ ..... **vii**
ŞEKİLLER DİZİNİ ..... **viii**
ÇİZELGELER DİZİNİ ..... **ix**
1. GİRİŞ **1**
1.1 GPS **4**
1.2 GSM **5**
1.3 GPRS **5**
2. KURAMSAL TEMELLER ve KAYNAK ÖZETLERİ **7**
2.1 GPS-GSM-GPRS Kullanılan Sistem **7**
2.2 GPS Hata Azaltma **9**
2.2.1 Hareketli ortalama filtre **10**
2.2.2 Kalman filtresi **10**
2.2.3 Lojistik regresyon analizi **11**
2.2.4 Uzun-kısa vadeli bellek ağları **12**
3. MATERYAL VE YÖNTEM **13**
3.1 Materyal **13**
3.1.1 Arduino Uno **13**
3.1.2 Sim908 modülü **15**
3.2 Yöntem **17**
3.2.1 NodeJS **17**
3.2.2 Leaflet.js **17**
3.2.3 Kestirim **18**
3.2.4 Kalman filtresi **19**
3.2.5 LSTM uzun-kısa vadeli hafıza ağları **22**
3.2.6 GRU kapalı tekrarlayan hücre **25**
3.2.7 Hareketli ortalama filtresi **26**
3.2.8 ARIMA otoregresif tümleşik hareketli ortalamalar modeli **26**
3.2.9 Lojistik regresyon analizi **27**

4. BULGULAR ve TARTIŞMA 29
4.1 GPS Verisinin GSM ve GPRS Kullanılarak Uzak Sunucuya Gönderilmesi 29
4.2 GPS Hata Azaltma 37
4.3 Bir Sonraki Konum Tahmini 43
5. SONUÇ 51
KAYNAKLAR 54
ÖZGEÇMİŞ 57

SİMGELER DİZİNİ

Rx	Alıcı
Tx	Verici

Kısaltmalar

AGPS	Assisted GPS
ARIMA	Autoregressive Integrated Moving Average
ARMA	Autoregressive Moving Average
AT	Attention Command
DGPS	Differential GPS
EKF	Extended Kalman Filter
FTP	File Transfer Protocol
GLONASSS	Global Orbiting Navigation Satellite System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPRS	General Packet Radio Service
GRU	Gated Recurrent Units
GSM	Global System for Mobile Communications
HTTP	Hypertext Transfer Protocol
IDE	Bütünleşik geliştirme ortamı
IP	Internet Protocol
JSON	Javascript Object Notation
LSTM	Long Short Term Memory
MAE	Mean Absolute Error
MSE	Mean Squared Error
LCD	Liquid Crystal Display
NMEA	National Marine Electronic Association
RAM	Read Access Memory
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SD	Secure Digital
SIM	Subscriber Identity Module
SMS	Short Message Service
SQL	Structured Query Language
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UKF	Unscented Kalman Filter
UMTS	Universal Mobile Telelcommunication Service
USB	Universal Serial Bus
WAAS	Wide Area Augmentation System

ŞEKİLLER DİZİNİ

Şekil 1.1 Uydu sayısına göre GPS hesaplama 4
Şekil 1.2 NMEA veri formatı 5
Şekil 1.3 GPRS sistemi genel kullanımı 6
Şekil 3.1 Arduino Uno 14
Şekil 3.2 Arduino IDE 14
Şekil 3.3 Sim908 ön yüzü 15
Şekil 3.4 Sim908 arka yüzü 16
Şekil 3.5 Sim908 ve Arduino Uno takılmış hali 16
Şekil 3.6 Kestirim yöntemi diyagramı 18
Şekil 3.7 Kalman filtresi hesaplama adımları 21
Şekil 3.8 Kalman filtresi hata azaltma örneği 22
Şekil 3.9 RNN diyagramı 22
Şekil 3.10 RNN yapısında uzun vadeli bağımlılık problemi 23
Şekil 3.11 RNN'deki tekrarlayan modülün tek katmanı 23
Şekil 3.12 LSTM yapısı 24
Şekil 3.13 LSTM yapısında kullanılan formüller 24
Şekil 3.14 LSTM ve GRU karşılaştırılması 25
Şekil 3.15 S eğrisi biçimindeki verilerin dağılımına lojistik regresyon uygulanması 28
Şekil 4.1 Sistem mimarisi 32
Şekil 4.2 Leaflet Draw ile çizilmiş düzenleme durumundaki poligon örnekleri 34
Şekil 4.3 Geofence uygulaması 35
Şekil 4.4 Tüm kütüphanelerin kullanıldığı uygulama ekranı 36
Şekil 4.5 Konum 1 GPS hata azaltma 38
Şekil 4.6 Konum 2 GPS hata azaltma 40
Şekil 4.7 Konum 3 GPS hata azaltma 41
Şekil 4.8 Çeşitli sinyaller için LSTM ve Kalman filtresinin uygulanması 45
Şekil 4.9 2 boyutlu veriler için Kalman filtresi ve LSTM uygulaması 45
Şekil 4.10 Konum 1 tahmin uygulamaları 46
Şekil 4.11 Konum 2 tahmin uygulamaları 48
Şekil 4.12 Konum 3 tahmin uygulamaları 49

ÇİZELGELER DİZİNİ

Çizelge 4.1 Sim908 led durumu anlamı 30
Çizelge 4.2 Sim908 GPS AT komutları 30
Çizelge 4.3 Sim908 GSM AT komutları 31
Çizelge 4.4 Sim908 GPRS HTTP AT komutları 31
Çizelge 4.5 Konum 1 hata oranı azaltma 40
Çizelge 4.6 Konum 2 hata oranı azaltma 41
Çizelge 4.7 Konum 2 hata oranı azaltma 42
Çizelge 4.8 Kullanılan yöntemlerin gerçek değer ile uzaklık farkları 42
Çizelge 4.9 Konum 1 tahmin uygulamaları hata karşılaştırması 47
Çizelge 4.10 Konum 2 tahmin uygulamaları hata karşılaştırması 47
Çizelge 4.11 Konum 3 tahmin uygulamaları hata karşılaştırması 49
Çizelge 4.12 Konum tahmini mesafe ölçümleri 50

1. GİRİŞ

Günümüzde konum servisi denince çoğu kişinin aklına GPS(Global Positioning System) gelmektedir. GPS dünya genelinde oldukça popüler bir konum servisi olarak bilinse de, Rus alternatifi GLONASSS(Global Orbiting Navigation Satellite System) ile birlikte bu iki servis GNSS(Global Navigation Satellite System) adı verilen bir sistemin parçalarıdır. Hindistan, Çin, Avrupa ülkelerinin geliştirmiş oldukları farklı konum servisleri de mevcuttur. İlk çıkan servis olması sebebi ile genelde GPS olarak bilinir. GNSS kısaca uzaydaki uydulardan sinyaller yollayan ve dünya çapında konumlandırma için kullanılan bir uydu kümesi olarak tanımlanır. Cep telefonları gibi GNSS alıcıları, yörüngedeki uydular tarafından gönderilen içerisinde konumlandırma, zamanlama gibi verileri içeren sinyalleri çözerek kendi konumlarını tespit edebilir. Bir GNSS alıcısının sinyal aldığı uydu sayısı arttıkça hesapladığı konumun doğruluk değeri de artar, ancak GNSS servislerinin performansı sadece uydu sayısına bağlı değildir. Bu performansa etki eden 4 faktör vardır.

1. Doğruluk payı: Ölçülmüş olan konum ile gerçek konum arasındaki mesafe, hız, zaman farkı.
2. Bütünlük: Sistemin beklenmedik ve istenmeyen koşullarda doğru alarm ve bilgi vermesi
3. Devamlılık: Sistemin kesinti olmadan çalışmaya devam etmesi
4. Kullanılabilirlik: Gönderilen sinyalin diğer 3 faktörü sağladığı ve sistemin çalıştığı zaman dilimi.

Eğer bu koşullar yeterince sağlanmazsa, GNSS servisleri uydu tabanlı yerel geliştirme sistemlerinden destek alarak iyileştirme yapar. GPS günümüzde çok yaygın bir şekilde hem askeri hem de sivil amaçlarla kullanılmaktadır. Kara, hava ve deniz araçlarının navigasyonu, füze güdümü ile hedef bulma, uçakların görüşünün sınırlı ya da olmadığı durumlarda konumunun tespit edilmesi gibi askeri kullanımları olduğu gibi; hayvan, araç, bebek takip sistemleri, kadastro ölçmeleri, jeodezik ve jeodinamik amaçlı ölçmeler, coğrafi bilgi sistemi veri tabanlarının geliştirilmesi gibi sivil kullanımları da mevcuttur.

Bu çalışma gerçek zamanlı takip sistemi şeklinde planlanmıştır. Bu araç takibi, çocuk takibi olabileceği gibi alzaymır hastalarının kaybolma ihtimallerine karşı da kullanılabilir. Takip sistemlerinde kullanılan yöntemler çok farklıdır, bazen kısa mesaj yolu ile iletişim sağlanırken, bazen sadece hafıza kartına veri kayıt edilir. Gerçek zamanlı olan uygulamalarda coğrafi kısıtlama yapmak mümkündür. Yapılmış olan tez çalışmasında Sim908 cihazı ile alınan GPS verisinin GSM(Global System for Mobile Communication) ve GPRS(General Packet Radio Service) teknolojileri kullanarak uzak bir sunucuya konum verileri gönderilmiştir. İnternet ortamına aktarılan veriler harita ekranında gösterilerek takip edilmesi sağlanmıştır. Takip sistemi gerçekleştirildikten sonra Geofence eklenerek sistemin uyarı vermesi de sağlanmıştır.

GPS ve GSM modüllerini içeren Sim908 cihazı ile Arduino Uno kullanarak iletişim kurulmuştur. Sim908 ile iletişim kurabilmek için cihazın kullanım klavuzunda belirtilen AT(Attention Command) komut setleri kullanılmıştır. GPS verisi alındıktan sonra uygun komutlarla UDP(User Datagram Protocol) protokolü kullanarak uzak bir sunucuya gönderilmiştir. Veriler gönderilirken HTTP(Hypertext Transfer Protocol) Get komutları ile iletişim sağlanmıştır.

Projenin sunucu tarafında ise bir script yazılarak gelen Get komutlarının içine gömülmüş olan GPS verileri GeoJSON formatında bir dosyaya kayıt edilmiştir. Sonrasında ise NodeJS tabanlı geliştirilmiş olan çalışmada Leaflet.js kullanılarak haritada gerçek zamanlı bir şekilde gösterilmiştir. Harita ekranımızda ayrıca polygon, çizgi, nokta gibi çizim işlemleri yapılmaktadır. Bu çizimler Geofence alanı olarak kullanılmaktadır. Eklenmiş olan çizim üzerinde düzenleme, silme gibi işlemler de gerçekleştirilmektedir.

Geofence kısaca coğrafi kısıtlama olarak tanımlanabilir. Belli bir alandan dışarıya çıkıldığı zaman uyarı veren bir sistemdir. Bir çocuk takip edilmek istenen hedef, oyun oynadığı park alanı ise Geofence alanı olarak örnek verilebilir. Çocuğun anne ve babası takip etmek, durumundan haberdar olmak için çocuk üzerinde kullandığımız cihazları elektronik bileklik şeklinde kullanabilirler. Çocuk Geofence alanından dışarıya adım attığı anda anne ve babası SMS, sesli uyarı veya mail yoluyla uyarılabilir. Geofence iki

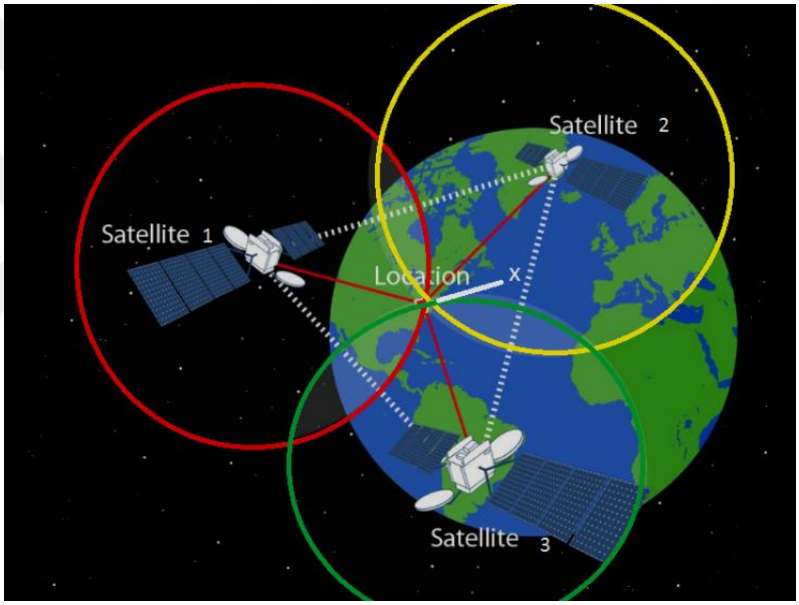
çeşit olarak tasarlanmıştır. İlki statik geofence, burada önceden tanımlı olan poligon alanları geofence olarak kullanılmıştır. İkincisi ise dinamik geofence, bunda ise harita ekranında takip edilirken o an çizilen poligon kayıt edilmeden geofence alanı olarak kullanılmaktadır. Bunun avantajı her an yeni çizimler yapmak ve çalışmayı aktif kullanmaktır.

GPS verisi alırken hava koşullarından, atmosferdeki yansımalardan, binalara çarpıp sıçramalar ve diğer çeşitli sebeplerden dolayı veriler hatalı elde edilir. Bu oran kullanılan cihaza göre, konum alınan yerdeki hava koşullarına, bina yoğunluğuna göre değişebilir. Verideki hata birçok sebepten kaynaklansa da veriler üzerinde düzeltmeler yapılarak hata oranı azaltılabilir. Kalman filtresi, hareketli ortalama ve lojistik regresyon yöntemleri kullanarak hata miktarı azaltılmaya çalışılmıştır. Ne kadar hata azaltabildiğimizi test etmek için de hatalı alınan konum verisinin gerçek değerine ihtiyaç vardır. Bu sebeple 3 adet konum belirlenip bu noktaların gerçek değerleri hata oranı 1-2 cm el hassasiyeti miktarında olan cihazla alınan veri, Sim908 cihazı ile alınan veri ile karşılaştırılarak deney yapılmıştır. Karşılaştırma sonuçları MAE(Mean Absolute Error), MSE(Mean Squarred Error) ve RMSE(Root Mean Squarred Error) hata hesaplama yöntemleri ile hesaplanıp karşılaştırmalı olarak verilmiştir. Haversive ve Vincenty yöntemleri ile konum verileri arasındaki uzaklık hesaplanıp gösterilmiştir.

GPS verisi belli bir hata payına sahiptir ancak sinyallerin bozulması, kapalı alanlara giriş çıkış gibi sebeplerle hesaplanamayabilir. Verinin hatalı da olsa hangi değer aralıklarında olduğunun tahminini yapmak bu noktada önem arz etmektedir. Geçmiş verilerden henüz alınmamış ya da eksik gelen verilerin tahmini yapabilmek amacı ile LSTM (Long Short Term Memory) ,GRU(Gated Recurrent Unit) ve ARIMA(Autoregressive Integrated Moving Average) yöntemleri ile bir sonraki verinin değeri tahmin edilmiştir. LSTM ve GRU bir geri beslemeli ağların türüdür. Günümüzde pek çok projede oldukça iyi sonuçlar vermektedir. Özellikle zaman serisi analizi şeklinde düzenlenebilen verilerde tercih edilir. GPS verisi ve bu algoritmalar ile yapılan çalışma sayısı çok sınırlıdır. Veri tahminleri hesaplanıp grafik haline getirilmiştir. Hata oranları da aynı şekilde hesaplanıp çizelge şeklinde sunulmuştur.

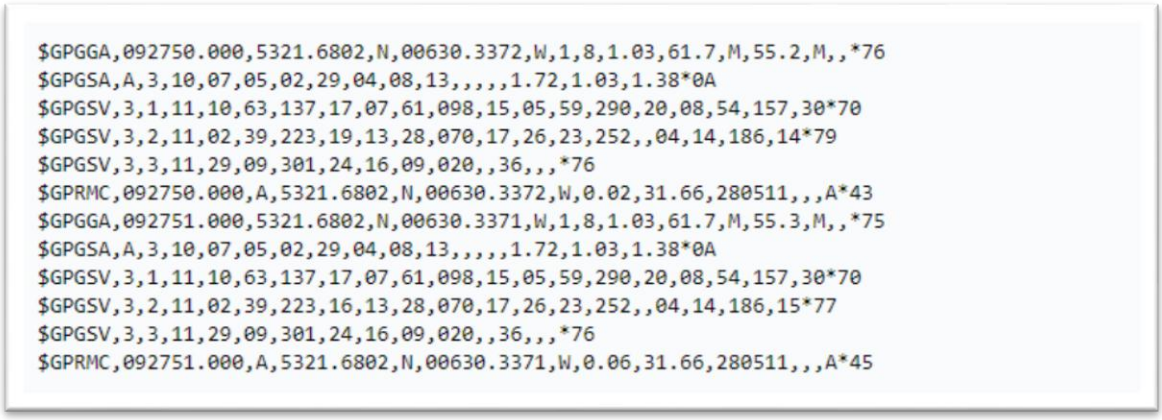
1.1 GPS

Amerika Birleşik Devletleri Savunma Bakanlığı'na ait 24 uydu vardır. Bunlardan 3 tanesi yedektir. Bu uydular bir tür radyo sinyali yayarlar. Bu sinyal içerisinde uydunun konumu ve zamanı mevcuttur. Uydularda atomik saatler kullanılır. Dünya yüzeyinin 20.200 km üzerindeki yörüngede bulunurlar. GPS hesaplamaları bu uydulardan alınan sinyallerle yapılır. Bu kadar yüksekte olduklarından yerçekimi daha az etki eder ve zaman daha hızlı akar. Bu sebeple Einstein'ın görelik kuramı göz önüne alıp hesaplamalar buna göre yapılmalıdır (Kaplan ve Hegarty 2005).



Şekil 1.1 Uydu sayısına göre GPS hesaplama (https://thesciencegeek.org 2018a)

Şekil 1.1'de belirtildiği gibi GPS cihazı 3 uyduyu gördüğünde enlem ve boylam değerlerini hesaplayabilir. 4 uydu verisi ile hesaplama yapılırsa yükseklik bilgisi de hesaplanır. Açık bir havada bir GPS alıcısı en az 4 en fazla 12 uyduyu görebilir. Uydular buna göre yörüngelere yerleştirilmişlerdir. GPS cihazları hesaplamaları sonucunda NMEA standardında veriler üretirler. Bu GPS koordinatlarının düzgün bir formatta kullanılması açısından önemlidir.



Şekil 1.2 NMEA veri formatı

GPS cihazı her saniye aldığı sinyaller ile NMEA formatındaki veriyi (Şekil 1.2) yeniden oluşturur. Bu verilerdeki her değer farklı bir anlam ifade eder. Biz sadece ihtiyacımız olan veri cümleciğini alıp işleriz.

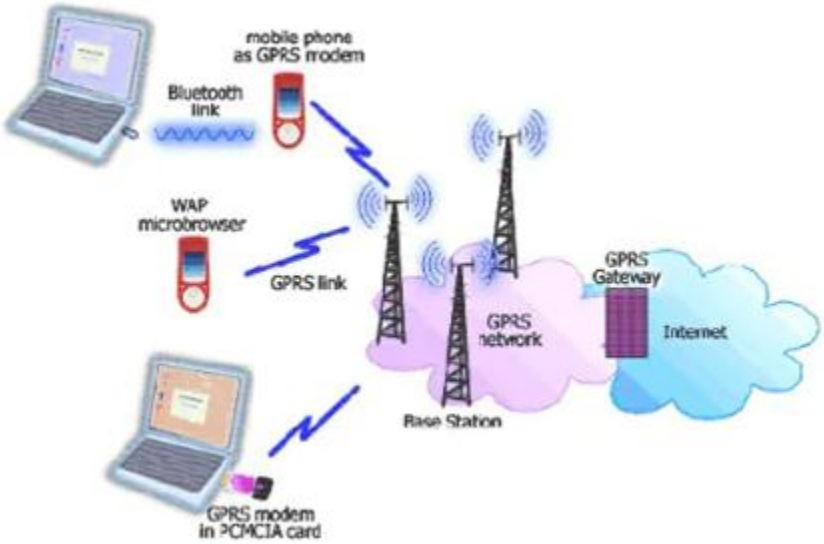
1.2 GSM

Türkçesi ile "Mobil İletişim İçin Küresel Sistem" şeklinde adlandırılan bir cep telefonu iletişim protokolüdür. Dünyadaki en yaygın cep telefonu ağıdır. Kullanıcı aynı hat ile Dünya'nın tüm ülkelerine ulaşabilir ve görüşme yapabilir. GSM standart olarak hücresel ağ kullanır. Dolaşım sırasında hücreler arasında geçiş yapabilir. Yani kapsama alanı dışına çıkılmaz ise tüm Dünya telefonla konuşularak dolaşılabilir (Heine 2003).

1.3 GPRS

GPRS cep telefonu alıcıları üzerinden veri aktarımı yapılmasını sağlayan mobil veri teknolojisidir. Paket anahtarlamalı veri aktarımını kullanır. İnternete bağlanmaya olanak sağlar. Paket anahtarlamalı veri iletişimi demek, verinin paket isimli parçalara ayrılıp gönderildikten sonra sıra numarasına göre tekrar birleştirilerek kullanılması demektir. İnternet ağı genel olarak bu teknoloji ile çalışır.

GPRS yaygın olan GSM teknolojisinden daha hızlıdır. GPRS üçüncü nesil UMTS'den (Universal Mobile Telecommunication Service) daha yavaştır. Bu sebeple 2.5G veya 2.75G diye adlandırılır. Günümüzde en yaygın 3G teknolojisi kullanılır. Bu teknoloji UTMS ağı üzerinde çalışır. 384 kilobit veri indirme hızına bir saniyede ulaşabilir. Bu teknoloji sayesinde internette hızlı dolaşabilir, web sitelerine hızla girebilir ve video izlenebilir. Günümüzde 4G iletişime de yavaş yavaş geçilmektedir. Gün geçtikçe iletişim hızının artması kaçınılmaz olmaktadır.



Şekil 1.3 GPRS sistemi genel kullanımı (https://tr.gearbest.com 2018b)

2. KURAMSAL TEMELLER ve KAYNAK ÖZETLERİ

Çalışma geniş kapsamlı olduğu için çeşitli alanlardan ve bilimsel çalışmalardan yararlanılmıştır. Arduino ve Sim908 ile ilgili olan kısım gömülü yazılım, GSM ve GPRS ile ilgili olan kısım veri iletişimi, NodeJS kullanılan kısım web yazılımı ve harita yazılımı yapılarak projenin teknik kısmı tamamlanmıştır. GPS verisisin hata oranını azaltmak ve nispeten uygun fiyatlı cihaz ile yüksek doğruluk elde etmek için Kalman filtresi, lojistik regresyon analizi ve hareketli otalama filtresi kullanılmıştır. Literatür taraması yaparken önceden çalışılmış olan mevcut sisteme benzeyen farklı cihazlarla, yöntemlerle yapılmış olan çalışmalar öncelikli ele alınmış olup kendi çalışmam ile karşılaştırılmalı olarak anlatılmıştır.

2.1 GPS-GSM-GPRS Kullanılan Sistem

Bir çalışmada (Le-Tien ve Phung-The 2010) ise GPS ve GSM temelli bir takip sistemi üzerinde çalışılmıştır. Farklı ve pratik bir model olarak araç takip sistemi tasarlamışlar. Yamaha şirketine ait olan Compass sensor YAS529 ve Koinix şirketine ait Accelerator sensörü KXSC72050 kullanılmıştır. Aracın yönü tahmin edilmiştir. Aynı şekilde GPS den alınan veriler SMS yolu ile ve GPRS ile bir telefona gönderilmiş bu şekilde takip edilmesi sağlanmıştır.

Bir diğer araç takip sistemini anlatan çalışmada (Almomani vd. 2011) GPS GSM ve GPRS teknolojileri kullanılmıştır. Coğrafi limitler yani Geofence alanları önceden tanımlı olarak sistemde mevcut olup bu alandan çıkınca SMS şeklinde uyarı veren bir sistem tasarlanmış. Araç hareketleri, araba alarmları, gerçek zamanlı takip edebilme işlemlerini sağlamış. Aşırı hızlanma, Geofence alanına giriş çıkış, arabanın durması ve hareket etmesi durumlarında uyarı vermektedir. Harita tabanı olarak da Google Maps Api kullanılmış. Web uygulamasında uyarı verileceği zaman konum verileri SMS şeklinde telefona gönderiliyor ve konum Google Maps uygulaması ile görülebiliyor. Geofence alanı olarak belirlenen kısımlar sabit ve çember şeklinde, yeni geofence alanları çizilip projeye dahil edilemiyor.

Bir çalışmada (Hu vd. 2012) ise otomobillere hırsız önleyici sistem şeklinde tasarım yapılmıştır. GSM ve GPS modülü kullanılmış olup araç çalındığı zaman titreşim sensörü ile haber verilmiştir. Sistem otomobilin sahibi ile sürekli bağlantı halindedir. Bunu da GSM modülü ile sağlar. Aynı şekilde GPS GSM modüllerini kullanarak araç takip ve güvenlik sistemi çalışmasının (Fleischer vd. 2012) amacı şehirle arası otobüs taşımacılığının gerçek zamanlı takibini gerçekleştirmektir. Araç kaza yaptığında ve hırsızlık olaylarında haber vermektedir. Takip sistemi şeklinde yazılmış olan (Verma ve Bhatia 2013) makalesinde ise aynı şekilde GPS ve GSM kullanılmış. Ekran ve Araç kısmı şeklinde 2 bölüm şeklinde anlatılmış. Cihazların kullanılıp GPS verilerinin kullanıldığı bölüm araç bölümü, harita uygulamasının ve gerçek takip işleminin yapıldığı bölüm ise ekran bölümüdür. Projede 16x2 LCD(Liquid Crystal Display), GSM ve GPS modülleri, Max32, AT Mega mikro denetleyici ve güç sağlayıcı kullanılmıştır. Detaylı bir şekilde NMEA verilerinin bölümleme işlemi anlatılmıştır. Geofence ve uyarı sistemi bu projede mevcut değildir.

Bir makalede (El-Medany vd. 2010) ise gerçek zamanlı takip sistemi geliştirilmiştir. Düşük maliyetli bir sistem tasarlamak için çaba harcanmıştır. GSM modülü olarak GM862 kullanılmıştır. Kullanıcı tarafında ise bir grafik arayüzü tasarlanmış bu sayede takip edilme sağlanmıştır. Microsoft Sql Server 2003 ile Asp.net kullanılarak özel bir haritada takip edilmesi sağlanmıştır. Sadece bulunduğu konum değil aynı zamanda aracın hızı ve aldığı yol miktarı da bulunmuştur. Farklı bir makalede (Lee vd. 2014) ise Arduino ve GPS GSM modülleri kullanarak bir takip sistemi geliştirilmiştir. GSM bağlantısı için herhangi bir kütüphane kullanmadan AT komutları ile iletişim kurulmuştur. Alınan GPS koordinatları GSM modülü kullanılarak SMS şeklinde bilgilendirilmek isteyen numaraya gönderilmiştir. Mesajı alan numara da uygun formattaki mesajı telefonun kendi Google Maps programından gözlemlemektedir. Tasarlanan bu sistem oldukça basit bir şekilde takip etmeyi sağlamaktadır. SMS ile sürekli iletişim sağlanamayacağından biraz maliyetlidir.

Farklı konularda takip ve kontrol amaçlı çeşitli projeler geliştirilmiştir. Vahsi yaşamdaki hayvanları takip etmek amacı ile (Thomas vd. 2011) geliştirilmiş bir çalışma mevcuttur. Nesli tükenmekte, avlanma yasağı olan hayvanlarda kullanılabilecek bir projedir.

Örneğin, dişleri için vahşice öldürülen filler takibe alınıp denetim altında tutulurken kullanılabilir. Bir projede (Beeco vd. 2014) seyahat ederken kaybolmamak için takip projesi geliştirilmiştir. Daha doğrusu gelen ziyaretçilerin arazide takibini sağlamak amacı ile. Hırsızların girmesini engelleyici bir sistem (Monti vd. 2016) şeklinde çalışmada mevcuttur.

Koyuncu ve Özdemir ise çalışmasında Arduino Mega, GPS Click, GSM Click , LCD ekran ve SD(Secure Digital) kart tan oluşan sistem tasarlamıştır. GPS verileri GSM ve GPRS yoluyla uzak bir sunucuya aktarılmış hem de LCD ekranda anlık olarak gösterilmiştir. Ayrıca veriler 8 saniyede bir gönderildiğinden SD karta loglama yapılmıştır (Koyuncu ve Özdemir 2016a).

Diğer bir çalışmada ise Sim900 cihazı, GPS Click ve Arduino Mega kullanarak sistem tasarlanmış. Harita yazılımında da Leaflet,js kütüphanesi kullanılmış. Basit çapta geofence eklenmiş olup veriler 5 saniyede bir gönderilip takip edilme işlemi sağlanmıştır (Koyuncu ve Özdemir 2016b).

2.2 GPS Hata Azaltma

GPS sinyalinde birçok sebep hata oluşmasına neden olur. Yörüngesel hatalar, uydu yörüngesindeki kaymalardan dolayı oluşur. Bu hatanın yok edilmesi duyarlılığın arttırılması ile olabilir. Uydu ve GPS alıcı cihazın saat farkından dolayı da hata gerçekleşir. Uydularda atomik saatler kullanılır. Alıcı cihazlarda ise daha az duyarlılığa sahip quartz saatler kullanılır. Aradaki mesafe farkından ve hassaslıktan dolayı gecikme meydana gelir. Sinyaller atmosferden geçerken bazı hatalar ve sapmalar gerçekleşir. Sinyallerin çok yönlü olması faz değişimleri, sıçramalarından da hatalar meydana gelebilmektedir. Ayrıca GPS cihazının faz merkezinden dolayı da sinyaller yanlış alınabilir. Alıcı cihazın kendine ait gürültüsü de mevcuttur.

Birçok araştırmacı GPS verisindeki hataları, sapmaları farklı yöntemler kullanılarak azaltmaya çalışmıştır. Kullanılan metodlar üç kategoriye ayrılabilir (Huang ve Tsai

2008). İlki WAAS (Wide Area Augmentation System), DGPS (Differential GPS) ve AGPS (Assisted GPS) gibi pahalı modüller kullanmaktır. İkincisi yardımcı bir periferik modül kullanmak iken sonuncusu ise veriyi farklı filtrelerden geçirerek yazılım ile düzeltmektir. Kalman filtresi ve Weiner Filtresi gibi algoritmalar ile hatalı gelen GPS verisi düzeltilir (Feldmann vd. 2009) (Kalman 1960). Bunların yanında hareketli ortalama filtresi, LSTM gibi zaman analizi yöntemleri de kullanılmaktadır(Iter, Kuck ve Zhuang 2016). Bu çalışmada farklı algoritmalar kullanılmıştır. Algoritma bazlı olarak kaynaklar gruplandırılıp anlatılmıştır.

2.2.1 Hareketli ortalama filtre

Japonya'da yapılan bir çalışmada ARMA(Autoregressive Moving Average) algoritması kullanılarak GPS verilerindeki hata azaltılmaya çalışılmıştır (Li vd. 2000). Başka bir çalışmada ise GPS verisinde önce koordinat dönüşümü yapılıp sonrasında özyinelemeli ortalama ve ARMA interpolasyon algoritması kullanılarak sonuçları paylaşılmış ve 4-10 metreye kadar iyileştirme sağlanmıştır (Islam ve Kim 2014). Diğer bir çalışmada ise hareketli pencere yaklaşımı konum verilerine uygulanmış ve sonuçları paylaşılmıştır (Song ve Lee 2015). Bu çalışmada hareketli ortalama filtresi ve hareketli standart sapma filtresi beraber kullanılmış ayrıca verilerdeki aşırı sapmalar, normal dışı verilerin etkinliğini azaltmış olup yumuşatma gerçekleştirmiştir.

2.2.2 Kalman filtresi

İlk olarak Rudoff Kalman tarafından bulunmuştur. Dinamik olan ve durum uzayı bilinen bir sistemde, daha önceki veriler kullanılarak bir sonraki sistemin olacağı yeri tahmin edebilen bir filtredir. Eğer sistemde skolastik veya rasgele bir gürültü varsa Kalman filtresi çok iyi çalışmaktadır (Kalman 1960). Kalman filtresinin gerçek amacı kovaryans matrisindeki hataları azaltmaktır. Temel Kalman filtresi ölçüm sonuçlarını lineer bir sistem olarak kabul eder, ancak gerçek dünyada birçok sistem bu şekilde değildir. Bu sebeple kalman filtresi geliştirilip çeşitlendirilmiştir.

Concept ve Riberio kalman filtresine yeni bir yaklaşım getirerek EKF (Extended Kalman Filter) bulmuş ve çeşitli uygulamalarda kullanılmaya başlamışlardır (Concept ve Ribeiro 2004). EKF filtresinde lineer olmayan sistem önce lineer hale getirilip sonrasında kalman filtresi uygulanması mantığına dayanır. Ancak doğrusal hale getirme kontrol edilemeyen bazı durumlara yol açabilir.

Wan ve Van (2000) UKF (Unscented Kalman Filter) isminde bir tahmin sistemi geliştirmişlerdir. UKF filtresi yinelemeli ve analitik bir çözüm sunar. Her adımda önceki tüm kestirimler kullanılarak, gelecekteki pozisyon tahmin edilir. Filtre değişkenleri uyumlu bir şekilde seçildiği takdirde oldukça iyi sonuçlar verir. EKF de olduğu gibi doğrusal hale getirme adımı olmadan ortalama ve kovaryansı uyumlu hale getirmek için kullanılır.

2.2.3 Lojistik regresyon analizi

Sürekli ya da süreksiz veriler arasındaki ilişkileri modellemek amacı ile farklı istatiksel modeller geliştirilmiştir. Amaç değişkenler arasındaki ilişkiyi en iyi uyuma sahip bir şekilde tanımlamaktır. Son yıllarda bu model tıp, biyoloji, ekonomi, tarım ve taşıma alanlarında sıklıkla kullanılmaktadır. Bu model biyolojik deneylerin analizi için kullanımı ilk olarak 1944 yılında önerilmiştir. 1970 yılında modele uygun çeşitli uygulamalar yapılmıştır. Cornfield (1962) lojistik regresyon uygulamalarında diskriminant yaklaşımını ilk kez kullanarak popüler hale gelmesini sağlamıştır.

Verilerin modele uyumuyla ilgili de çeşitli çalışmalar mevcuttur. Lasaffre ve Alber (1989) ise çoklu grup lojistik modellerde aykırı ve etkin gözlemler ile kullanımını incelemişlerdir. Bu çalışmadan sonra lojistik regresyon modelleri yaygın bir şekilde kullanılmaya başlanmıştır. Çeşitli hata tahmin yöntemleri geliştirilmiştir. Duffy (1990) ise hata terimlerinin nasıl dağıldığı ve parametre değerlerinin değişkenliği incelemiştir. Rocca (2001) da lojistik regresyonun gelişimi ile geniş çaplı tarama yapıp hangi aşamaların önemli olduğuna vurgu yapmıştır.

2.2.4 Uzun – kısa vadeli bellek ağları

İlk olarak Hochreiter ve Schmidhuber (1997) tarafından bulunmuştur. Gers ve arkadaşları (2000) tarafından da geliştirilmiştir. Bir tekrarlamalı yapay sinir ağı türüdür. Zaman serisi analizlerinde bir sonraki veriyi tahmin etmede başarılı sonuçları vardır. LSTM ağı zaman gecikmesi gibi hata durumlarında daha iyi bir çözüm sunar. Girdileri uzun süre hatırlama özelliğine sahiptir.

Graves (2009) LSTM ile el yazısı doğrulama ile ilgili bir çalışma yapmıştır. Doğal dil işleme, ses verilerini yazıya çevirme gibi projelerde kullanılmaktadır. Iter ve arkadaşları (2016) GPS ile hedef bulma amacı ile LSTM, kalman ve KNN kullanmışlardır. Singh (2017) ise çalışmasında anormal olan verileri bulmak için LSTM kullanmıştır.

3. MATERYAL VE YÖNTEM

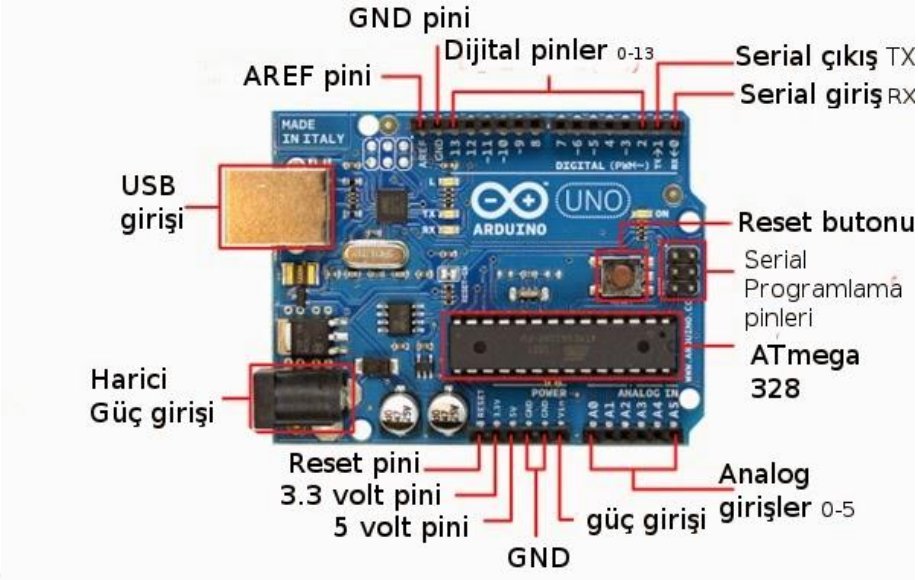
Materyal bölümünde kullanılan cihazlar ve yapılandırma ayarları anlatılmıştır. Yöntem bölümünde ise alınan verilerin harita ekranında nasıl kullanıldığı detaylı bir şekilde ele alınmıştır. Yazılım kütüphaneleri ve sistem burada tanıtılmıştır. Verilerdeki hata oranını düşürmek için kullanılan algoritmalar hakkında bilgi verilmiştir.

3.1 Materyal

3.1.1 Arduino Uno

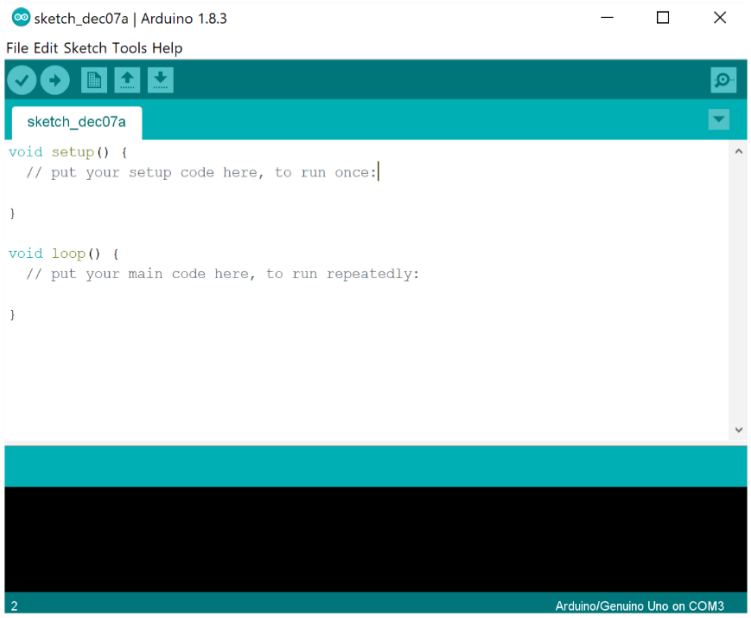
Arduino bir elektronik geliştirme kartıdır. İçerisinde kendine ait işlemcisi bulunur. Geliştirme arayüzünden bilgisayar vasıtası ile kodlar yüklenir ve çalıştırılır. Arduino kartlarının donanımında Atmel firmasına ait AVR mikro denetleyiciler bulunur. Arduino temel olarak kodların yazıldığı bir IDE(Integrated Development Environment), Arduino bootloader, kütüphaneleri, AVRDude diye adlandırılan mikrodenetleyici programlayan yazılım ve derleyiciden oluşur.

Arduino ile robotik uygulamalardan elektronik devrelere kadar birçok şey yapılabilir. Son zamanlarda çok farklı kartları mevcuttur. Örneğin giyilebilir teknolojiler için bir kart, görüntü işleyebilecek işlemciye sahip farklı bir kart da mevcuttur. Çalışmanın içeriğine göre, ihtiyacımız olan işlemci ve iletişim kuracağımız pin özelliklerine göre kart seçimi yapılabilir. Bu çalışma kapsamında, hem sim908 kartımızla uyumu olduğundan hem de bizim için yeterli özelliklere sahip olduğundan Arduino Uno kartı tercih edilmiştir.



Şekil 3.1 Arduino Uno (http://diyot.net 2018c)

Şekil 3.1'de kullanmış olduğumuz Arduino Uno kartının pinlerini anlatan bir resim vardır. Program çalışırken reset tuşuna basınca en baştan başlar. Usb girişi ile bilgisayara bağlanıp program yüklenir. Serial giriş ve Serial çıkış olarak gösterilen Tx ve Rx pinleri iletişim amaçlı kullanılmıştır.

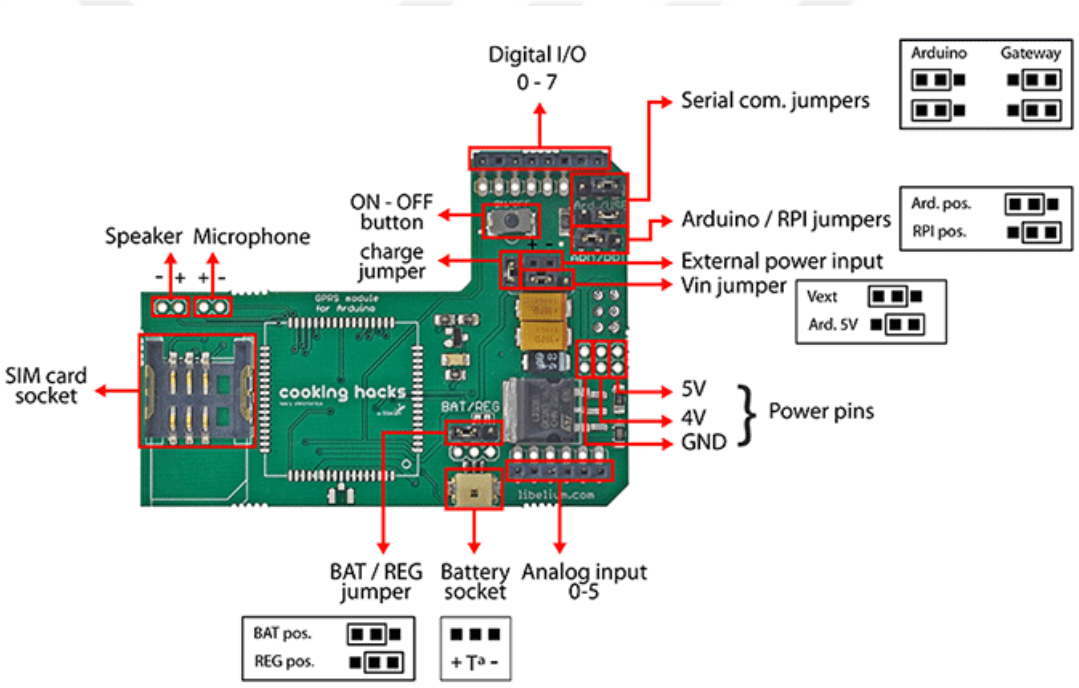


Şekil 3.2 Arduino IDE

Şekil 3.2'de ise Arduino IDE ekranı gösterilmiştir. Setup fonsiyonunda başlangıç için gerekli ayarlar yapılır. Örneğin kullanılacak pinler, baud rate ayarı gibi ayarlar burada yapılır.

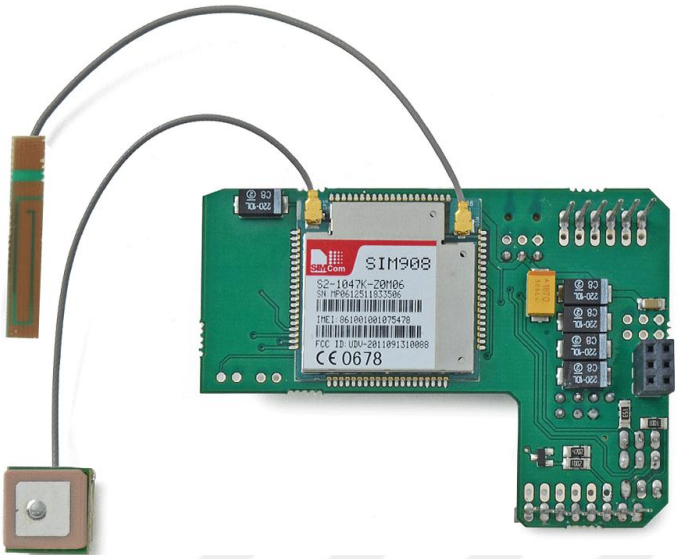
3.1.2 Sim908 modülü

İçerisinde GSM ve GPS modüllerini birlikte bulundurur. Arduino ve Raspberry Pi ile birlikte çalışacak şekilde tasarlanmıştır. Kullanım kılavuzunda verilen AT komutları ile iletişim kurulur.



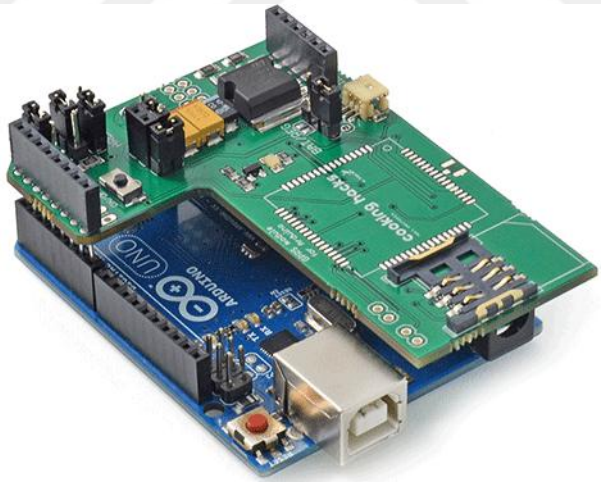
Şekil 3.3 Sim908 ön yüz (https://www.cooking-hacks.com 2018d)

Bağlantıları şekil 3.3'de görüldüğü gibidir. SIM (Subscriber Identity Module) kart takılır. Takılmadan önce pin kodu sıfırlanır ise doğrulama yapmadan çalıştırılabilir.



Şekil 3.4 Sim908 arka yüz (https://www.cooking-hacks.com 2018d)

Sim908 modülünün arka kısmının resmi şekil 3.4'de gösterilmiştir. Belirtilen yerlere GPS ve GPRS alıcıları takılmıştır.



Şekil 3.5 Sim908 ve Arduino Uno takılmış hali (https://www.cooking-hacks.com 2018d)

Arduino Uno kartı ile birleşik hali şekil 3.5'de gösterilmiştir. Pin ayarlamaları şekilde görüldüğü gibi yapılmalıdır. Bilgisayara bağlı iken çalıştığı gibi harici bir güç kaynağı da kullanılabilir.

3.2 Yöntem

3.2.1 NodeJS

NodeJS, hem arka plan hem de ön plan uygulamaları yapılabilen, çalışma zamanlı bir javascript platformudur. Yapı gereği asenkrondur. Yani istekleri teker teker yapmak yerine paralel olarak yapar. Her istekten sonra isteğin bittiğini gösteren bir fonksiyon çağırılır ve bu yapı sayesinde gerçek zamanlı uygulamalar, oyunlar, anlık mesajlaşma uygulamaları kolaylıkla yapılabilir. Gerçek zamanlı takip sistemi uygulamasında en önemli etken hız ve performans olduğu için NodeJS tercih edilmiştir. Arka plandan gelen GPS verisini sürekli kontrol edilip gerçek zamanlı olarak haritada gösterilmiştir. NPM(Node Packaged Modules) kütüphaneleri kolaylıkla indirebileceğimiz, kurabileceğimiz bir paket yöneticisidir. Diğer geliştiriciler tarafından hazırlanmış, yazılmış modülleri projeye kolaylıkla eklemeyi ve daha hızlı uygulama geliştirmeyi sağlar (https://www.npmjs.com/ 2018e)

Express.js isimli bir kütüphane kullanarak sunucu oluşturma, ayarlama ve çalıştırma gibi işlemler yapılmıştır. Bu kütüphane sayesinde çeşitli ayarları elle tek tek yapmak yerine otomatik yapılması sağlanmıştır (https://expressjs.com/ 2018f). Gulp bir Javascript görev çalıştırıcısıdır. "Gulpfile.js" isimli dosyadaki verilen her görevi sırası ile yerine getirir. Bu bizim derlemeden önce yapmamız gereken tüm işlemleri otomatik hale getirip, hızlandırır. Örneğin verilen görevlerden bir tanesi kullanılan Javascript kütüphanelerini dosya_adı.min.js şeklinde küçültmek olsun. Tek yapmamız gereken şey, javascript kütüphanelerinin olduğu dosya yolunu göstermek ve görev tanımı yapmaktır. İşlemleri otomatik hale getirerek, çalışma zamanını azaltır (https://gulpjs.com/ 2018g).

3.2.2 Leaflet.js

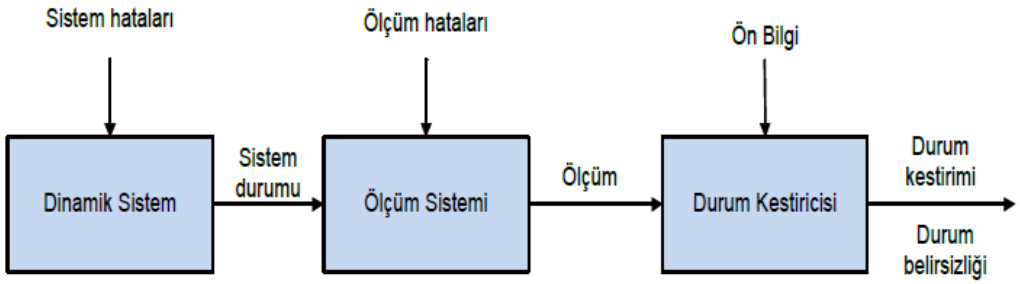
Leaflet harita oluşturmak için yazılmış açık kaynak kodlu bir javascript kütüphanesidir. Openlayers, Google Maps Api gibi alternatifleri olmasına karşın çok esnek, kolay

geliştirilebilir ve özelleştirilebilir olmasından dolayı tercih edilmiştir. Kullanım ihtiyaçlarına göre özelleştirilmiş çok fazla sayıda eklentisi kütüphanesi Github'dan bulunup kullanılabilir. Leaflet kendine ait GeoJSON formatını ve diğer harita veri formatlarını destekler. GeoJSON formatı JSON veri formatının harita uygulamalarında kullanımını sağlar. Yani nokta, poligon, çizgi gibi harita üzerinde kullanabileceğimiz değerler belli formattadır. GeoJSON, Leaflet kütüphanesine çok esneklik ve hızlılık katmıştır. Veriler tek bir katmanda eklenip işlenebilmektedir.

Leaflet Realtime ise harita ekranını yenilemeden bir web servisinden alınan verileri eş zamanlı gösterebilen bir kütüphanedir. Çalışmanın gerçekleşmesini sağlayan en önemli özellik aslında bu kütüphanede bulunmaktadır. Cihazdan anlık olarak gönderilen verilerin anlık olarak haritada gösterilmesine imkân tanımaktadır.

3.2.3 Kestirim

Kestirim adından da anlaşılacağı üzere kesin olmayan verilerden anlam çıkartma işlemidir. Sistemde ardışık olarak yapılan ölçümler vardır. Dinamik durum kestirimi şeklinde de adlandırılır. Sistem hatası, dinamik sistem modeli, ölçüm modeli, sistem ve ölçüm hataları ön bilgi olarak verilmelidir. Sonrasında durumun ne kadar belirsiz olduğu ve kestirim sonucu elde edilir.



Şekil 3.6 Kestirim yönetimi diyagramı

3.2.4 Kalman filtresi

Sistemin önceki verileri ile o anki giriş çıkış bilgileri ile durum tahmini yapan bir filtredir. Sistem belirli bir gürültüye sahipse minimum varyans tahmini yapar. Sistemin ölçülenmeyen durumlarını çok iyi tahmin eder. Algoritması özyinelemeli ve gerçek zamanlı çalışır. En az kareler eğriye sığdırma yöntemi kullanılarak filtre eder. Gelecek veriyi de kendi oluşturduğu matematiksel tahmine göre ayarlar.

Filtrenin tahmini gözlem ile karşılaştırıldığında fark "kalman kazancı" olarak adlandırılır. Kalman kazancı sistemin ne kadar değiştiğini gösteren bir çarpandır. Gelecek tahminler için sisteme girdi şeklinde uygulanır. Önceki tahminlerin hepsini kullanmak ve işleme koymak yerine bir değişkende sabitlemek sistemin yükünü azaltır. Kalman Kazancı değişebilir ve ayarlanabilir olması filtrenin performansını arttırır. Filtrenin önemli özelliklerinden bir tanesi de gelecek değerlerin tahminleri tüm belirsizliklerle birlikte üretebiliyor olmasıdır.

Filtre sensör füzyonu ve veri füzyonu için çok kullanışlıdır. Genelde sistem durumuna ait kesin sonuç elde etmek için tek bir ölçüm yapılmaz. Çok sayıda gürültülü ölçüm yapılıp, sonrasında durum tahmin edilmek üzere matematiksel olarak birleştirilir. Çalışma ile ilgili örnek vermek gerekirse, GPS cihazının bir çocuğun elinde olduğunu varsayalım. Çocuğun yerini hassas olarak belirlemek için veri alıyoruz. GPS verileri gürültülü verilerdir. Ortalama el GPS cihazlarında artı eksi 10 metre hata payı vardır. Okumalar her zaman yakın olmasına karşın hiçbir zaman gerçek değildir. Kalman Filtresi çocuğun hızı ve dönüşlerini gidiş açısını gözeterek, zamana göre analiz ederek tahmin edebilir. Parekete hesabı olarak bilinen bu teknik çocuğun yeri hakkında yaklaşık bir bilgi verir. Bu şekilde küçük hatalar birikince de sapmalar olacaktır. Çocuğun hızı ile orantılı olarak konumun değişmesi de beklenen diğer bir özelliktir (Çayıroğlu 2012).

Kalman filtresi iki farklı aşama şeklinde çalışır, bunlar tahmin ve güncellemedir. Tahmin etme aşamasında verinin ortak sapma miktarı olan kovaryans değeri hesaplanır. Bu değer hareketin hızı ile orantılıdır. Sonra güncelleme aşamasında yeni bir veri daha

gelir. Bu değer ile bir miktar daha belirsizlik oluşur. Bu belirsizlikten gelen kovaryansın bir önceki verinin kovaryansa oranı yeni gelen verinin ne kadar sapma yapabileceğini belirler. Yani yeni tahmini nasıl etkileyeceğine bu değer karar verir. Gerçek değerden uzaklaşıldıkça bu yöntem hızla gerçek değere doğru gürültüyü azaltarak çeker. Kalman filtresi, diğer filtreler içinde kestirim hatasını gittikçe azaltan tek filtredir. Doğrusal sistemler için en optimum tahmini yapar. Gerçek hayatta doğrusal olan çok az sistem vardır. Sistemi önce doğrusal hale getirip sonra kalman uygulama yöntemi de "Genişletilmiş Kalman Filtresi" olarak adlandırılır. Verileri lineer hale getirme kontrol dışı değişimler getireceğinden bazı negatif sonuçlar doğurabilir.

$$X_k = K_k \,.Z_k + (1 - K_k \,).\, X_{k-1} \tag{3.1}$$

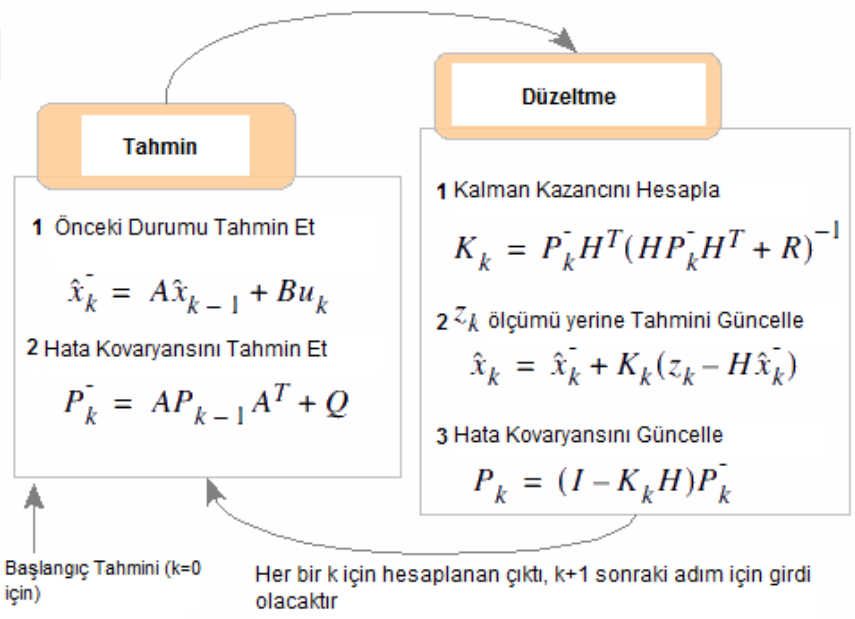
Formül 3.1 kalman filtresinin durum matrislerinin atıldıktan sonraki en yalın halidir. "k" harfi durumları (states) gösterir. Biz GPS verisi için analiz yapacağımız için zaman aralığı olarak düşünebiliriz. Amacımız X_k değerini bulmak. Bu değer bir sonraki tahmin değeridir k için bu değer bulunur. Z_k ise ölçülen değerdir. Z_k değeri daha önce açıklandığı gibi bazı hatalara sahiptir. K_k değeri ise Kalman Kazancı olarak adlandırdığımız değerdir. X_{k-1} değeri ise sinyalin bir önceki zamandaki tahmin değeridir. Önceki tahmin değeri ve ölçülen değerlere zaten sahibiz. Kalman Kazancı hesaplandıktan sonra denklemde bilinmeyen bir değer kalmıyor. Kalman kazancının 0,5 alındığı varsayılırsa ortalama bulan bir fonksiyon haline gelir. Buradan anlaşılan bu değer her zaman aralığında verideki değişme oranını verir. En uygun ortalama katsayısını bulur. Bu yöntem geçmişteki tüm verileri kullanmak yerine bunu bir katsayıya indirger. Bu sebeple kalman filtresi hızlı çalışır.

$$\begin{aligned} x_k &= Ax_{k-1} + Bu_k + w_{k-1} \\ Z_k &= H\, x_k + v_k \end{aligned} \tag{3.2}$$

Kalman filtresinin modeli yukarıdaki denklemler (3.2) ile oluşturulur. İlk denklem (doğrusal rastsal denklem) sayesinde x_k değeri bulunur. x_k değeri önceki değeri olan x_{k-1} değerine kontrol sinyali (u_k) ve bir önceki işlem gürültüsü olan (w_{k-1}) değeri ile

toplanarak bulunur. Kontrol sinyali her zaman kullanılmayabilir. *A, B* ve *H* değişkenleri matrislerin genel gösterimleridir. Sadece bir sayısal değer olabildiği gibi daha büyük boyutlu matris de olabilir. Probleme göre değişkenlik gösterir. Sabit olarak varsayabiliriz. İkinci denklemden de ölçüm değeri (Z_k), sinyalin değeri (x_k) ve ölçme gürültüsünün (v_k) toplamına eşittir. İşlem gürültüsü (w_{k-1}) ve ölçme gürültüsü (v_k) doğrusal değerler olarak hesaplanır. Bu değerler istatiksel olarak birbirinden bağımsızdır.

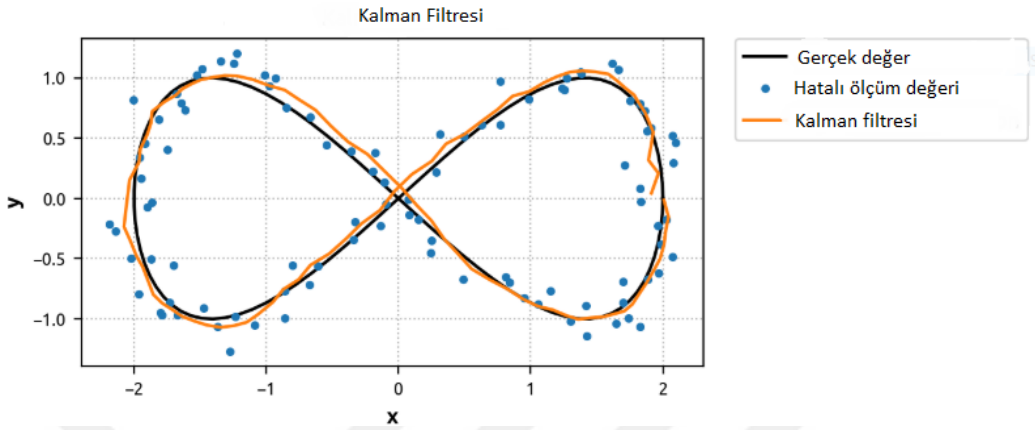
Sistemin düzgün çalıştığı varsayılırsa geriye sadece işlem gürültüsü (w_{k-1}) ve ölçme gürültüsü (v_k)'nın ortalamasını almak ya da standart sapmasını bulmak kalır. Gerçek dünyada sinyalimiz Gaussian olmasa bile yaklaşık olarak filtrenin doğru tahminlere doğru yakınsama yaptığını gözlemleriz. Gürültü parametreleri için yapılan doğru tahminler filtrenin daha erken düzgün sonuçlar elde etmesine olanak tanır.



Şekil 3.7 Kalman filtresi hesaplama adımları

Şekil 3.7'deki döngüsel hesaplamalar kalman filtresinin temelini oluşturmaktadır. Daha önceden açıklandığı gibi *A, B* ve *H* matristir. Genelde sabit sayı alınabilir. Belirlenmesi

zor olan katsayılar *R* ve *Q* dur. *R* ortamdaki gürültüye bağlıdır ve genelde verinin gürültülü olduğundan eminizdir.

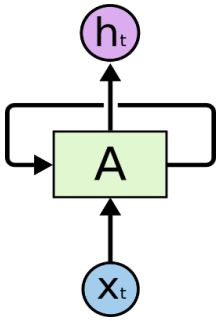


Şekil 3.8 Kalman filtresi hata azaltma örneği

Şekil 3.8'de görüldüğü üzere kalman filtresi ile gerçek veriye oldukça yakın bir sonuç elde edilmiştir.

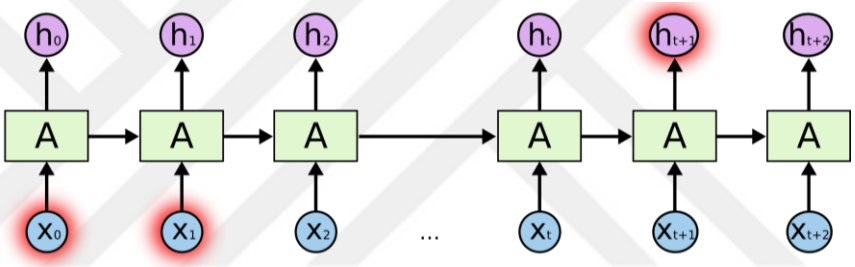
3.2.5 LSTM uzun kısa-vade hafıza ağları

İnsanların düşünceleri hep sıfırdan başlamaz, süreklilik içindedir. Geleneksel sinir ağları bunu yapamaz. Bunu yapabilmek için bir hafızaya sahip olması ve o hafızadaki veriyi ne zaman kullanacağını öğrenmesi gerekir. RNN(Recurrent Neural Network) yani "Tekrarlayan Sinir Ağları" bu sorunu ele alır ve bu yüzden döngüsel bir yapıya sahiptir. Döngüler bilginin devam etmesine izin verir.



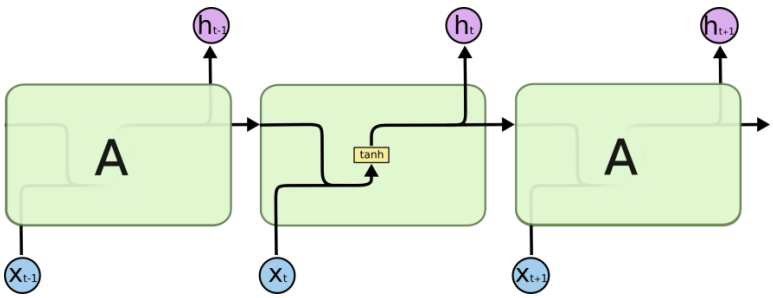
Şekil 3.9 RNN diyagramı

Şekil 3.9'da sinir ağının (A) bir parçası, X_t girişini alır ve H_t çıkışını üretir. Ok ile gösterilen döngüsel yapı da bilginin bir diğerine geçmesine izin verir. Düşündüğümüz zaman kendini sayısız bir şekilde tekrarlayan aynı ağın kopyalarından oluşan normal bir YSA yapısından farkı yoktur. RNN yapısı büyük ilerlemeler getirmiştir ve çok kullanım alanı vardır. LSTM aslında bir RNN çeşididir. RNN yapısının döngüler sayesinde hafızasında tutabildiği şey sadece yakın geçmiştir. Bir cümleyi tamamen tahmin edemez, sadece önceki geçen kelimenin ardından gelebilecek kelimeyi tahmin eder. Örneğin "Türkiye'de büyüdüm, akıcı Türkçe konuşuyorum." cümlesinde Türkiye ile Türkçe arasında bir bağlantı vardır. Bu bağlantıyı bulmak mesafe arttıkça daha da zorlaşır. RNN yapısı bu bilgileri birbirine bağlamayı öğrenemez. RNN'nin bu sorunu "Uzun vadeli bağımlılık problemi" olarak bilinir.



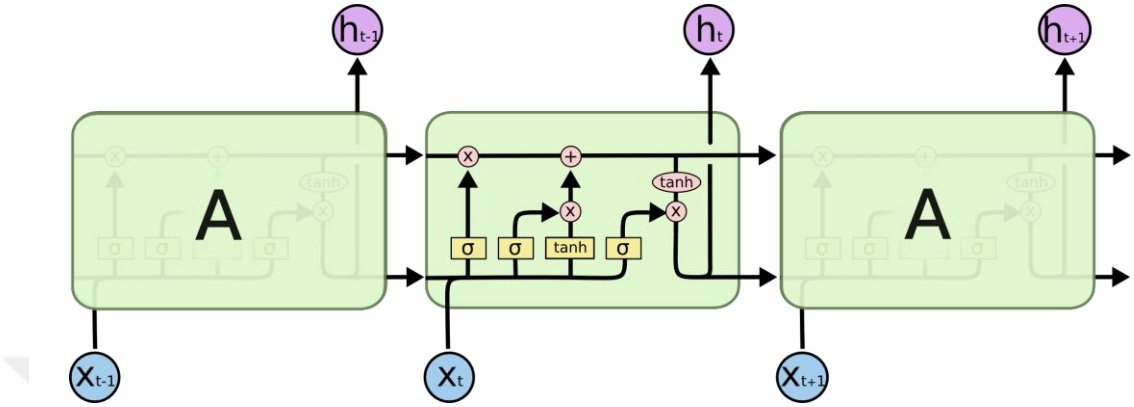
Şekil 3.10 RNN yapısında uzan vadeli bağımlılık problemi

LSTM uzun süreli bağımlılıkları öğrenebilme özelliği ile tanınır. Çeşitli problemler üzerinde kayda değer iyileştirmeler yapar. Uzun süreli bağımlılık sorununu çözmek için özel olarak üretilmişlerdir. Bilgiyi uzun süre boyunca öğrenmek onların beklenen davranışlarıdır. Tekrarlayan sinir ağları tekrar eden modüllerden oluşur.



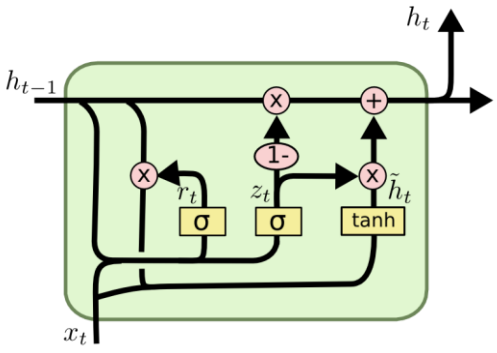
Şekil 3.11 RNN'deki tekrarlayan modülün tek katmanı

Şekil 3.11'de görüldüğü üzere tekrarlayan modül tek katman içerir LSTM yapısı tanh fonksiyonundan daha fazlasına sahiptir. Bir zincir içerisinde dört tane sinir ağı vardır ve bunlar özel bir iletişim halindedir.



Şekil 3.12 LSTM yapısı (http://colah.github.io 2018h)

LSTM kendi içerisinde hafızalara sahiptir. RNN de olduğu gibi geriye yayılım yaparak problemi çözmeye çalışmaz. Sonuçlarını hafızasına atar ve gerekli olduğu zaman kullanır. RNN her gelen veri ile çok çabuk güncellenen bir ağdır bu da yanılgıya neden olur. Bizim ihtiyacımız olan sürekli güncellenen bir sistem değil ne zaman güncellenmesini de zamanla öğrenen bir yapı. LSTM, hafızasını çok seçici bir şekilde günceller. Bunu gerçekleştirmek için özel öğrenme mekanizmaları kullanır. Gelen bilginin hangi kısmının hatırlanmaya değer hangi kısmının odaklanılmaya ihtiyaç duyulduğunu söyler.



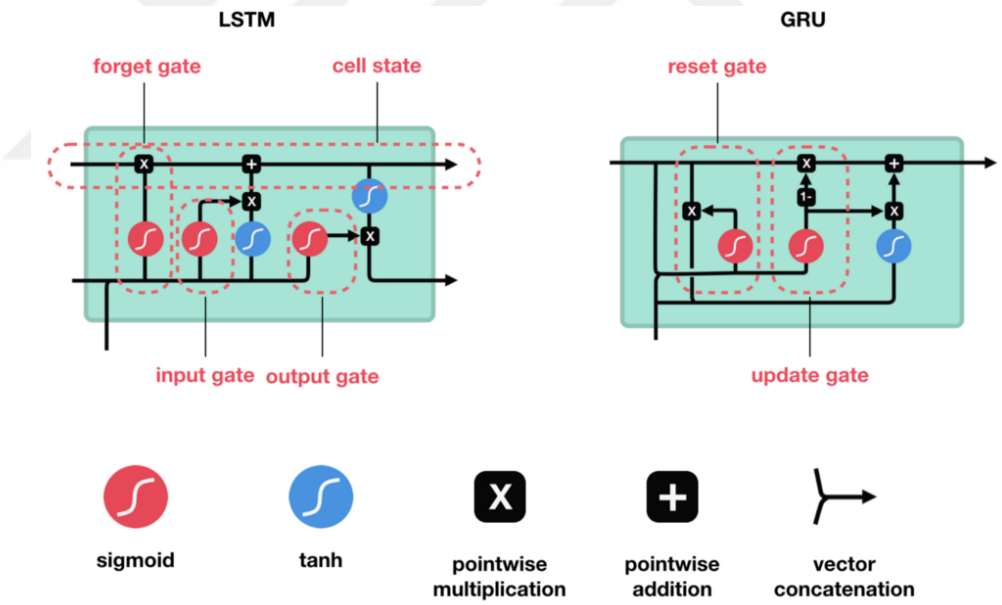
$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Şekil 3.13 LSTM yapısında kullanılan formüller

Yukarıdaki formüller LSTM değişkenlerinden sadece birkaç tanesidir. LSTM yapısının nasıl çalıştığı anlatılmayacaktır. Proje kapsamında Tensorflow kütüphanesi ile LSTM ağları çalıştırılmış ve kayda değer bir iyileştirme sağlanmıştır.

3.2.6 GRU kapılı tekrarlayan hücre

LSTM gibi tekrarlayan sinir ağlarının bir kapı mekanizmasıdır. Konuşma sinyali modelleme gibi çalışmalardaki sonuçları LSTM benzeri olduğu tespit edilmiştir. Çıkış kapısı olmadığından dolayı LSTM den daha az parametreye sahiptir. Yapısı basittir bu sebeple gizli durumunu güncellemek için daha az hesaplama yapması gerekmektedir. LSTM e nazaran daha hızlıdır. LSTM ve GRU kapılarının karşılaştırması Şekil 3.14 de verilmiştir. Diyagramdan anlaşılacağı üzere LSTM forget kapısı, input kapısı ve output kapısına sahipken GRU reset kapısına ve update kapısına sahiptir.



Şekil 3.14 LSTM ve GRU karşılaştırılması (https://towardsdatascience.com 2018ı)

3.2.7 Hareketli Ortalama Filtresi

Ortalama algoritması belirli sayıda verinin alınıp veri sayısına bölünerek ortalamanın bulunmasıdır. Hareketli ortalama filtresi ise belirli sayıda veriyi alıp FIFO yapısı ile filtre boyutuna göre ortalama almadır. Basit ama etkili bir filtredir.

$$y(n) = \frac{x(n-1) + x(n) + x(n+1)}{3} \tag{3.3}$$

3 boyutlu filtre formülde (3.3) verildiği gibidir. Kullanılan filtrenin dezavantajı filtre boyutunu aşana kadar hesaplama yapamamasıdır. Tahminler gecikme ile elde edilir. GPS verisi ile kullanılarak ani sıçramaların etkisi azaltılmıştır.

3.2.8 Otoregresif Tümleşik Hareketli Ortalamalar Modeli (ARIMA)

Bağımlı değişkenlerin kendisinden daha önce gelen değerlere bağlı olduğu ve bu değerlere ait fonksiyonu olarak ifade edilen modellere otoregresif model yani AR model denir. Otoregresif sürecin formülü aşağıdaki gibidir.

$$y_t = \delta + Ø_1 y_{t-1} + Ø_2 y_{t-2} + \ldots + Ø_p y_{t-p} + \varepsilon_t \tag{3.4}$$

Formülde (3.4) t zamanı, p gecikme değerini, δ sabit bir değeri, y ölçümleri, Ø otoregresif katsayıları, ε ise hata terimini gösterir. Tahmin edilmek istenen veri, kendinden önce gelen verilere, hata terimine ve belirli sabitlere bağlı olarak değişir. Dikkat edilmesi gereken nokta denklemin kaçıncı mertebeden olduğunun hesaplanmasıdır. AP(p) olarak gösterilir. Bağımlı değişkenlerin önceki değerlerinin hatalarına bağlı olduğu ve ağırlıklı hataların bir fonksiyonu olarak ifade edilebildiği zaman serisi modeline hareketli ortalamalar modeli (MA) adı verilir. Hareketli ortalamalar modeli ise aşağıdaki formülle temsil edilir.

$$y_t = \mu + \varepsilon_t - \Theta_1 \varepsilon_{t-1} - \Theta_2 \varepsilon_{t-2} - \ldots - \Theta_q \varepsilon_{t-q} \tag{3.5}$$

Burada (3.5) μ sabit bir değeri, Θ model parametrelerini, q ise gecikme değerini gösterir. Hareketli ortalamalar modeli MA(q) olarak gösterilir ve model mertebesi belirlenerek ilgili denklemde kaç adet değişken kullanılacağına karar verilir. Eğer bir zaman serisi hem hareketli ortalama hem de otoregresif modellerin özelliklerini taşıyorsa bu modele ARMA denir. Otoregresif hareketli ortalamalar modeli olarak adlandırılır. ARMA(p,q) şeklinde gösterilir, aşağıdaki bağlantı (3.6) ile ifade edilir.

$$y_t = \delta + \emptyset_1 y_{t-1} + \emptyset_2 y_{t-2} + \ldots + \emptyset_p y_{t-p} + \varepsilon_t - \Theta_1 \varepsilon_{t-1} - \Theta_2 \varepsilon_{t-2} - \ldots - \Theta_q \varepsilon_{t-q} \qquad (3.6)$$

ARMA(p,0) eşittir AR(p) ve ARMA(0,q) eşittir MA(q) şeklinde ifade edilebilir. İki sürecin birleşimidir çünkü. Bazı zaman serilerinin de ortalamaları sabit değildir. Böyle serilere durağan olmayan zaman serileri denir. Bu tür serilerde belirli bir bölümün ortalama değeri serinin geri kalan ortalama değerinden farklıdır. Farklılığı ortadan kaldırmak ve seriyi durağan hale getirmek için serinin farkı alınır. Süreç durağan olunca ARMA modeli ile tahmin yapılır. Bu tümleşik durum ise ARIMA(p,d,q) şeklinde gösterilir. Serinin 1 kere farkı alınıyor ise d=1 alınır. D =1 için denklem (3.7) aşağıdaki gibi olur.

$$y'_t = \delta + \emptyset_1 y'_{t-1} + \emptyset_2 y'_{t-2} + \ldots + \emptyset_p y'_{t-p} + \varepsilon_t - \Theta_1 \varepsilon_{t-1} - \Theta_2 \varepsilon_{t-2} - \ldots - \Theta_q \varepsilon_{t-q} \qquad (3.7)$$

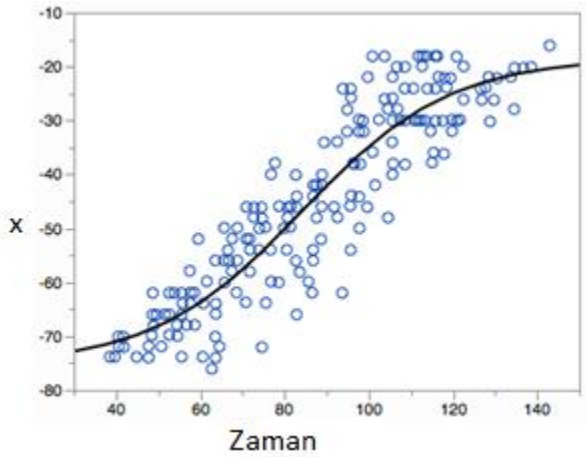
Fark alma işlemine seri durağan hale gelene kadar devam edilir. Aşağıdaki denklem (3.8) eşitliği iki ölçüm arasındaki farkı ifade eder.

$$y'_t = y_t - y_{t-1} \qquad (3.8)$$

3.2.9 Lojistik Regresyon Analizi

Basit doğrusal regresyon, ikili ve sürekli veriler arasındaki ilişkileri özetlememize yarayan istatiksel bir yöntemdir. Bağımlı ve bağımsız değişken arasındaki ilişkiyi tanımlar. Zamana bağlı değişen verilerde de bu durum kullanılır. Lojistik regresyon ise, sonucu belirleyen bir veya daha fazla bağımsız değişken bulunan verileri analizinde

kullanılır. Kısaca birçok değişkene bağlı ve sonucu evet ya da hayır olan bir sınıflandırma problemi gibi düşünülebilir. Sonuç olarak en uygun modeli bulur. Çok farklı problemlerde kullanılır. “Logit” fonksiyonunu kullanır. Doğrusal olmayan verilerle çalışırken verimli bir sınıflandırma sağlar. Yahut onlara en uygun geçen doğruyu bulur.



Şekil 3.15 S eğrisi biçiminde verilerin dağılımına lojistik regresyon uygulanması

4. BULGULAR VE TARTIŞMA

Daha önceden bahsedildiği gibi çalışma çok farklı alanlarda çalışmayı gerektirmektedir. İlk olarak cihazların kurulumu iletişimi verilerin internet ortamına aktarılması anlatılacaktır. İkinci olarak da verilerin harita üzerinde gösterilmesi, harita üzerinde çizim işlemlerinin ve geofence mekanizmasının nasıl çalıştığı anlatılacaktır. İkinci olarak ise GPS verisindeki hataları azaltmak amacı ile uygulanan algoritmaların sonuçları açıklanacaktır. GPS verisinin alınmadığı durumlarda hatalı yerleri tamir etmek ve bir sonraki noktayı tahmin etmek amacıyla kullanılan algoritmalar ve sonuçları da son olarak açıklanacaktır.

4.1 GPS Verisinin GSM ve GPRS Kullanarak Uzak Sunucuya Gönderilmesi

Proje kapsamında Arduino Uno ile Sim908 cihazı birlikte kullanılmış ve veriler alınıp uzak bir sunucuya kaydedilmiştir. Arduino IDE üzerine kodlarımızı yazdıktan sonra cihazımızı bir USB(Universal Serial Bus) ile bilgisayara takarız. Bilgisayar üzerinden uygun port ayarlaması yapılır. Arduino ile bağlantı kurabilmek için kullanılan baudrate değeri 11500 olarak seçilmiştir. Sim908 cihazının desteklediği oran bu şekildedir. Diğer belirlenmesi gereken durum da AT komutları arasındaki bekleme süresidir. Çok hızlı ya da yavaş konut göndermek cihazın düzgün çalışmasını engellediğinden bu değer komutlara göre değişebilmekle birlikte genelde 2000ms olarak kullanılmıştır. Kod yüklemesi bittikten sonra programımız Sim908 ile iletişim kurmaya çalışacaktır. Bu aşamada aktif hale getirebilmek için Sim908 üzerindeki ON – OFF butonuna basmamız gerekmektedir. Basıp belirli bir süre bekledikten sonra bağlantı kurulup kurulmadığı led ışığı sayesinde anlaşılır. Çizelge 4.1'de led ışığının hangi anlama geldiği açıklanmıştır.

Öncelikle GPS modülü ile iletişim kurup konum verisi alınmıştır. Her saniye konum alacak şekilde ayarlamaları yapılmıştır. Kullanılan komutlar çizelge 4.2'de gösterilmiştir. Alınan NMEA verisi oldukça fazla bilgiye sahiptir. Bu veri bölümlendirilip kullanılması gereken kısımlar ayrılmıştır. Enlem, boylam, zaman verileri bu proje için yeterli olduğundan bu kısımlar kullanılmıştır. Gönderilen veri

miktarı hem GSM hattına tanımlı olan internet kullanım oranını arttırdığı hem de daha geç iletildiği belirlendiğinden kullanılacak minimum veri miktarı seçilmiştir.

Çizelge 4.1 Sim908 led durumu anlamı

LED Durumu	Sim908 Davranışı
Sönük	Cihaz çalışmıyor
64ms yanıyor / 800ms sönüyor	Cihaz ağa kayıtlı değil
64ms yanıyor / 3000ms sönüyor	Cihaz ağa kaydedildi.
Sim908 ağa kaydedildi.	PPP GPRS İletişim Kuruldu.

Çizelge 4.2 Sim908 GPS AT komutları

AT Komutları	Komutun Cevabı	Tanımı
AT+CGPSPWR	OK	GPS cihazını çalıştır.
AT+CGPSRST	OK	GPS cihazına reset at.
AT+CGPSSTATUS	Unknown, Not Fix, 2D Fix and 3D Fix	GPS durumunu al.
AT+CGPSINF	NMEA cümlesi	NMEA cümlesini al

GSM modülünün de ayarlamaları, komutları test edilip kullanılmıştır. Öncelikle modüle içerisinde internet paketi tanımlı ya da kontör bulunan bir hat takılmalıdır. Hatta ait olan APN ayarları servis sağlayıcısından öğrenilip, AT komutları ile tanımlanmıştır. Sonrasında sim karta ait olan pin kodu kontrol edilip GSM hattı aktif hale getirilmiştir. Arama, SMS atma gibi işlemler komutlar yardımı ile gerçekleştirilmiştir. Alınan GPS verisi tanımlanmış farklı bir numaraya SMS olarak gönderilip bu şekilde basit çapta uyarı sistemi yapılmıştır. Çizelge 4.3'de en çok kullanıma ihtiyaç duyulan komutlar verilmiştir.

Çizelge 4.3 Sim908 GSM AT komutları

AT Komutları	Komutun Cevabı	Tanımı
AT+CPIN="****"	OK	Eğer sim kart pin kodu ile kilitli ise '****' kilidini açar.
AT+COPS?		Operatör bilgisini verir.
ATD*********;		'********' numarasını arar
AT+CMGF= (0/1)	OK	SMS formatını ayarlar. 0 PDU modu, 1 text modu şeklinde.
AT+CMGS		SMS gönderir
AT+CMGR=*		* numarasından gelen SMS okunur

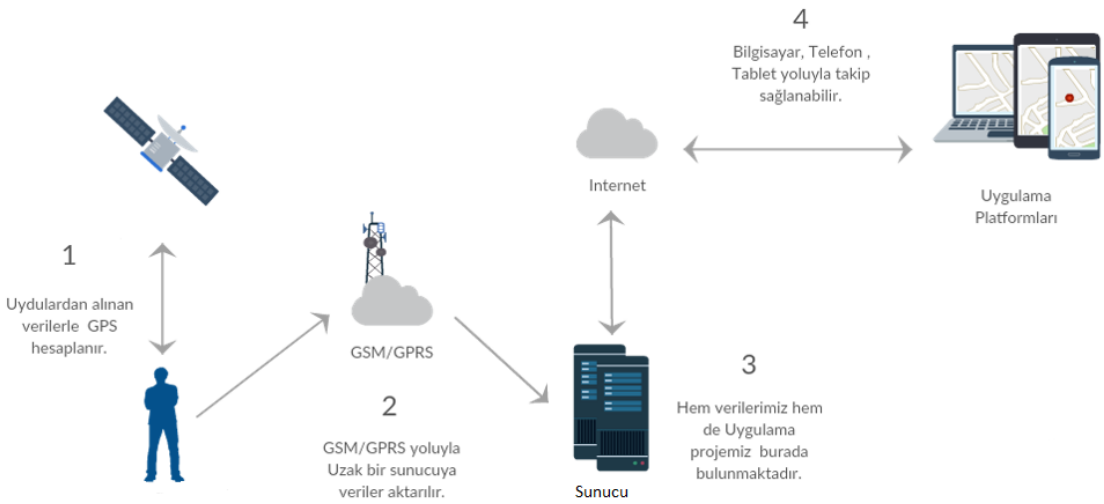
GSM modülü içerisinde internet bağlantısını sağlayan GPRS teknolojisini de içermektedir. Cihazımızın TCP(Transmission Control Protocol), UDP, IP (Internet Protocol), HTTP ve FTP(File Transfer Protocol) desteği mevcuttur. Verileri uzak bir sunucuya göndermek için HTTP protokolü tercih edilmiştir. HTTP protokolü GET ve POST metodlarını kullanır. GET metodu kullanarak veriler gönderilmiştir. Çizelge 4.4'de kullanılmış olan HTTP komutları vardır.

Çizelge 4.4 Sim908 GPRS HTTP AT komutları

AT Komutları	Komutun Cevabı	Tanımı
AT+SAPBR	OK	GPRS ayarlarını yapar.
AT+HTTPINIT	OK	HTTP servisini başlatır.
AT+HTTPPARA	OK	HTTP parametrelerini ayarlar.
AT+HTTPACTION=0	OK	Uzak Sunucuya GET komutunu gönderir.
AT+HTTPREAD	OK	HTTP verisini okur.
AT+HTTPTERM	OK	HTTP servisini kapatır.

Alınan NMEA formatındaki GPS verileri bölümlendirilip hali hazırda çalışmakta olan HTTP servisi yolu ile her saniye GET komutu ile uzak bir sunucuya gönderilmiştir. Verilerin iletilme süresi internet hızı, sunucunun uzaklığı, ağ yoğunluğu gibi durumlardan dolayı değişkenlik gösterebilir. Yaklaşık olarak 4 -5 saniyede bir verilerimiz iletilmiştir. Sunucu tarafında ise basit bir servis yazılıp alınan verilerin veri tabanına ya da bir dosyaya kayıt edilmesi sağlanmıştır.

Veritabanına kayıt edilmiş olan veriler SQL (Structured Query Language) sorguları ile elde edilip web uygulama ekranımızda gösterilmiştir. Bizim için önem arz eden veri hep en son eklenen en güncel veri olduğu için sorgulamalar bu şekilde yapılmıştır. En son konumdan önceki 10 konum da gittiği yolun görselleştirilmesi için kullanılmıştır.



Şekil 4.1 Sistem mimarisi

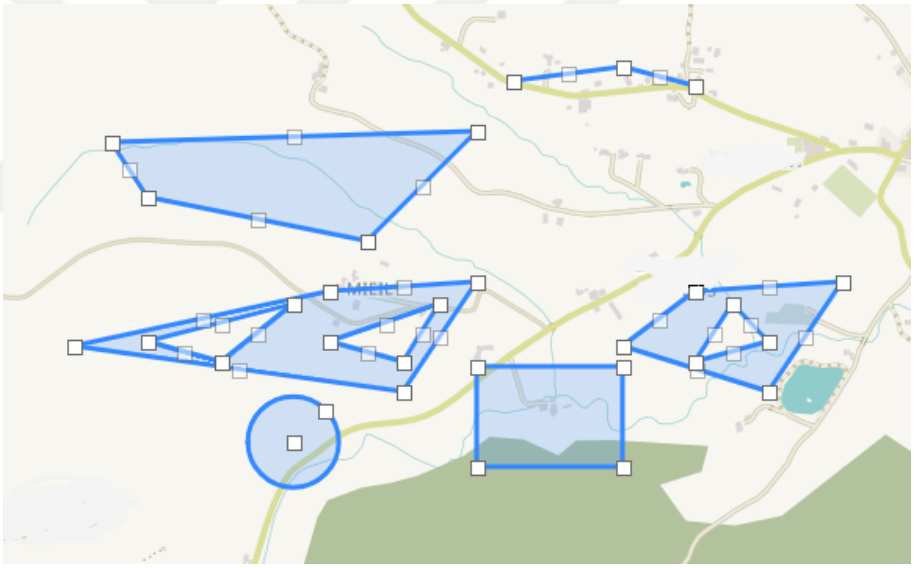
Şekil 4.1'de genel sistem mimarisi gösterilmiştir. Web uygulama projesi olarak da NodeJS tercih edilmiştir. NodeJS son zamanlarda en kolay ve güvenli şekilde web ve mobil projeler yazabilmek için kullanılmaktadır. Sunucu kısmı ve arka plan programlama kısmı tamamen Javascript dilinde yazılır. Yapısı gereği asenkrondur. Normal bir uygulama veritabanına ya da bir dosyaya istek gönderdiği zaman cevabını bekler. NodeJS ise istekleri teker teker yapmak yerine paralel bir şekilde yapar. Bu da oldukça hızlı çalışmasına olanak tanır. Fonksiyonel programlamaya uyumlu yapısı gereği modüler kod yazmaya teşvik eder. Leaflet.js ile de oldukça uyumlu ve hızlı

çalışır. Leaflet.js ise coğrafi bilgi sistemleri ile uğraşanların genelde tercih ettiği kolay ve hızlı geliştirme yapılması aynı zamanda açık kaynak olmasında dolayı tercih edilmiştir. Yapılması planlanan geofence işlemi Leaflet.js nin sahip olduğu ek kütüphaneler sayesinde çalıştırılabilmiştir. Google Maps Api kütüphanesinde verilerin işlenme şekline kolaylıkla müdahale edilememektedir. Programın aktif bir şekilde çalışması için ekranın yenilenmeden verilerin ekran üzerinde gösterilip istendiğinde silinmesinin sağlanması gerekmektedir. Bu sebeple Google Maps Api yerine Leaflet.js tercih edilmiştir. Öncelikle NodeJS ile tasarlanan sistemde kullanılan kütüphaneler tanıtılacaktır. NodeJS de package.json isimli bir dosya tanımlanır ve kullanılan tüm kütüphaneler buradan yönetilir. Yararlanılmış olan kütüphaneler şunlardır.

- gulp
- gulp-browserify
- gulp-clean
- gulp-concat
- gulp-csslint
- gulp-jshint
- gulp-live-server
- gulp-livereload
- gulp-minify-css
- gulp-nodemon
- gulp-notify
- gulp-rename
- gulp-sass
- gulp-uglify
- gulp-zip
- jshint
- express

Express.js server oluşturup web uygulamamızı görüntüleyebilmemizi sağlar. Gulp ise Node.js tabanlı işlemlerimizi otomatik hale getirmemizi sağlar. Yapılan işlemleri

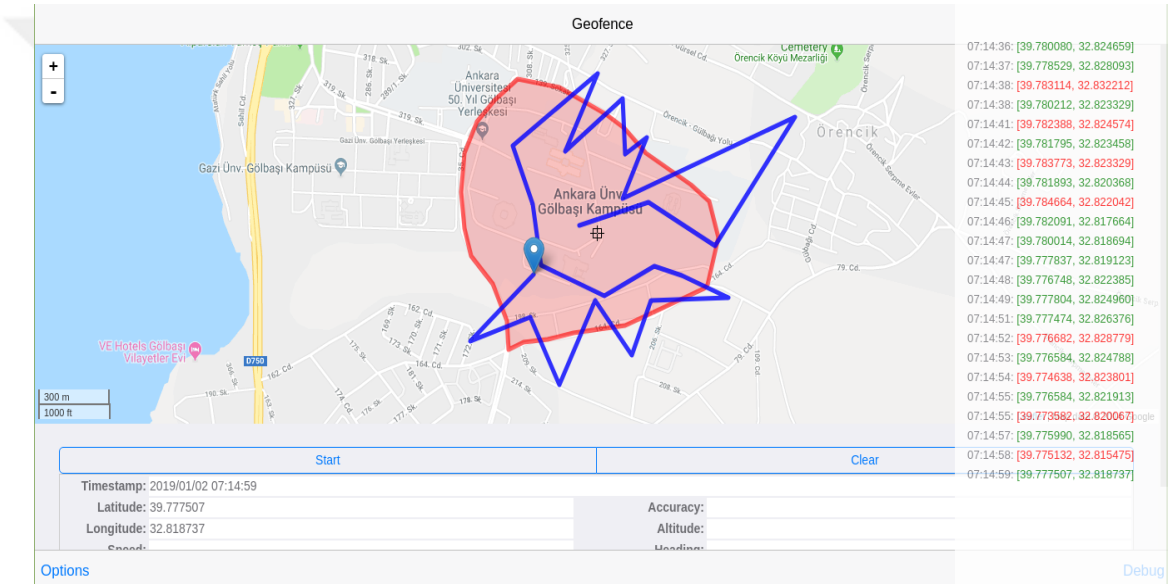
hızlandırmak, basitleştirmek amaçlı kullanılır. Tek bir komutla birçok iş yapabilir. Örneğin Jade-html formatında yazılmış olan kodları html yapısına çevirir. CoffeScript kodlarını JavaScript kodlarına çevirir. Bu sayede sistemde çok sayıda yapıyı, teknolojiyi kolaylıkla birlikte kullanabiliriz. NodeJS ve Leaflet.js birbiri ile oldukça uyumlu çalışır. Leaflet.js ile birlikte kullanılan, harita ekranımızı daha aktif kullanmamıza yardım eden en önemli kütüphaneler sırası ile açıklanacaktır. Leaflet Draw harita ekranı üzerinde çizim yapmamızı sağlar. Bir nokta, poligon, devamlı çizgi, çizgi, daire, çember gibi çizimleri kolaylıkla yapıp aktif kullanmamızı sağlar. Bu çizimler aynı zamanda düzenlenebilir, silinebilir özelliktedir. Kütüphane bu desteği sağlar. Bu çizimler geofence alanı belirlerken kullanılmıştır. Şekil 4.2 de çizilmiş olan şekiller düzenlenebilir hale getirilmiştir. Bu sayede yanlış çizilen alan doğru hale getirilir (https://github.com/Leaflet/Leaflet.draw 2018i).



Şekil 4.2 Leaflet Draw ile çizilmiş düzenlenme durumundaki poligon örnekleri

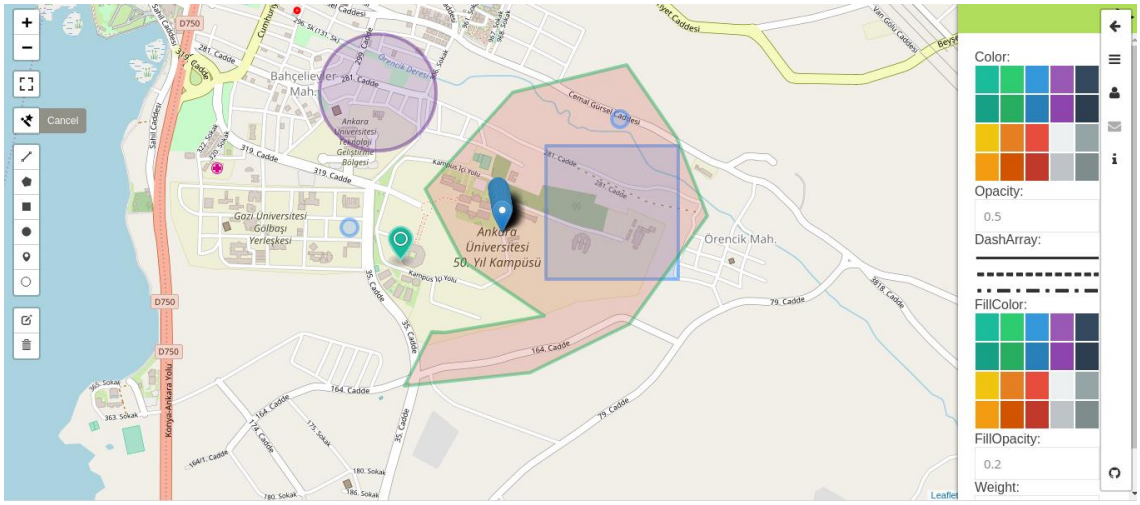
Leaflet FullScreen görüntüleme kütüphanesi ise bir buton ile tam ekranda harita ekranımızın kullanımına olanak tanır. Leaflet Realtime kütüphanesi ise koordinatların harita ekranını yenilenmeden değiştirilebilip, düzenlenmesine olanak tanır. Harita üzerinde alınan GPS koordinatları sürekli olarak değiştiği için değişen koordinatları haritayı yenilemeden görüntüleyebilme ihtiyacını gidermek için bu kütüphaneyi kullanıldı. Arka planda değişen koordinatlar gerçek zamanlı bir şekilde gösterildi.

Leaflet Pip kütüphanesi de aslında point in poligon probleminin çözümünün Leaflet.js ye uyumlu hale getirilmiş halidir. Çözümü Ray Casting (ışın döküm) algoritmasını kullanarak yapmıştır. Kısaca sorgulama noktasından bir ışın oluşturur ve sorgulanan point in poligon içerisinde olup olmadığını test eder. Bu kütüphane sayesinde kolaylıkla takip edeceğimiz bir GPS koordinatının geofence alanı içinde olup olmadığını sorgulanabilir.(https://github.com/mapbox/leaflet-pip, 2018j) Leaflet Pip kütüphanesi Leaflet Realtime kütüphanesi ile entegreli çalıştırıldığında ise ekranı yenilemeye, yeniden sorgulamaya gerek kalmadan gerçek zamanlı geofence işlemini gerçekleştirilmemize olanak tanır.



Şekil 4.3 Geofence uygulaması

Şekil 4.3 uygulama ekranında kırmızı renkte çizilmiş olan poligon geofence alanını göstermektedir. Yer belirteci olarak gösterilen pin de cihazın nerede olduğunu belirtmektedir. Mavi çizgi de daha önce gittiği güzergâhı göstermektedir. Yan kısımda ise Geofence içerisinde olup olmadığı koordinat verileri ile takip edilebilir. Ayrıca uygulama kapsamında sesli uyarı da çalışmaktadır. Konum değerleri ve çeşitli hesaplamalar da haritanın altında anlık olarak gösterilmektedir.



Şekil 4.4 Tüm kütüphanelerin kullanıldığı uygulama ekranı

Şekil 4.4 genel harita uygulama ekranımızdır. Poligon, çizgi, nokta, daire gibi şekiller çizilip, düzenlenip, silinebilir olacak şekilde yapılmıştır. Ayrıca sol kısımda görünen Leaflet Style editör sayesinde farklı renk, opaklık, çizgi şekli gibi ayarlamalar yapılabilmektedir. Geofence alanları renklerine göre gruplandırılabilir. Aktiflik durumuna göre opaklık ayarı yapılabilir. Uygulama ekranı hazırlanırken her şeyin kullanışlı ve güzel görünmesine özen gösterilmiştir. Uygulamanın daha da geliştirilebileceği düşünülerek ek özellikler için görünüm ayarlamaları yapılmıştır. Kullanım kolaylığı olması açısından farklı altlık haritaları eklenmiştir.

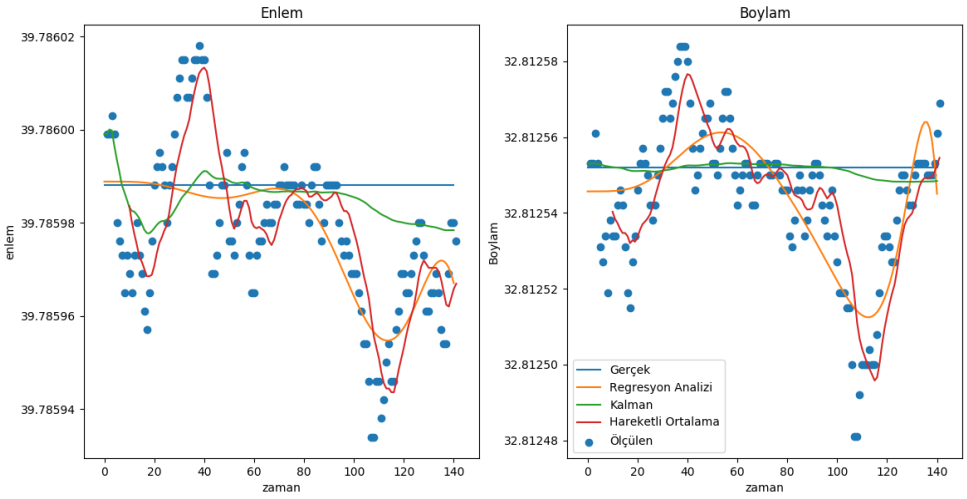
Leaflet.js GeoJSON şeklinde kendine ait bir veri türü kullanır. JSON verilerinin haritada kullanımına olanak sunan bir veri türüdür. Hızlı ve etkili bir kullanım sunduğu için tercih edilmiştir. Veritabanına kaydedilmiş olan veri GeoJSON formatına çevrilerek uygulamada kullanılmıştır. Arka planda kayıtlı olan bir poligonu geofence alanı olarak kullanım mümkün iken, online olarak çizilen poligonu alanı da veritabanına kayıt edilmeden geofence alanı olarak kullanılabilmektedir. Bunun yapılabiliyor olması projeye esneklik ve kullanım kolaylığı sağlamıştır. 2 farklı kullanım söz konusu olduğundan statik geofence ve dinamik geofence şeklinde isimlendirme yapılmıştır. Statik geofence, veritabanında kayıtlı olan poligonu geofence alanı olarak kullanımı iken, dinamik geofence kayıtlı olmayan poligonu geofence alanı olarak kullanılmasıdır.

GPS ile çalışırken 2 farklı problem karşımıza çıkmıştır. İlki GPS verilerinin hatalı olmasıdır. Veri hesaplanabilir ama bazı sebeplerden dolayı hata payına sahiptir. Bu sebepler tezin kaynak özetleri kısmında anlatılmıştır. Çözüm olarak veri tabanına kayıt ettiğimiz verilere bazı algoritmalar, teknikler kullanarak hata payını azaltma işlemi gerçekleştirilmiştir. Hareketli ortalama filtresi, lojistik regresyon filtresi ve kalman filtresi bu problem için çözüm olarak değerlendirilmiş olup uygulama sonuçları paylaşılmıştır. İkinci problem ise çeşitli sebeplerden dolayı verilerin bazı zaman değerlerinde alınmamasıdır. Bu durum saniye başına alınan bir veride yok sayılabilir ancak, saatte bir örnek alınan bir göçmen hayvanın GPS verisinin alınmaması sorun teşkil eder. Buna yönelik hata azaltma amacı gütmeden bir sonraki adımın konumu tahmini edilmeye çalışılmıştır.

Bunun için de ARIMA, LSTM ve GRU algoritması kullanılmıştır. Kalman Filtresi de yapısı gereği bir sonraki noktayı tahmin etmesine karşın o hata azaltan özelliği daha ağır bastığından dolayı ilk kısımda değerlendirilmiştir.

4.2 GPS Hata Azaltma

Algoritmaların genel bilgileri tezin önceki bölümlerinde anlatılmıştır. Bu bölümde deney ortamından ve analiz sonuçlarından bahsedilecektir. Öncelikle GPS verileri ile çok sayıda ölçüm yapılıp algoritma kullanılabilir, ancak tüm yapılan bu ölçümler hatalı olacağı için kıyaslanabilecek bir doğru veriye ihtiyaç vardır. Bu amaçla ölçüm hatası sadece el hassasiyeti kadar olan bir cihazla alınan konum verisi ile projede kullandığımız GPS cihazı ile ölçmüş olduğumuz veriler karşılaştırılmıştır. 3 adet konum belirlenip yaklaşık olarak 100 tane hatalı veri alınıp analiz yapılmıştır. Alınan veriler sürekli hatadan dolayı sıçradığı için zaman serisi analizi olarak temellendirilmiştir.



Şekil 4.5 Konum 1 GPS hata azaltma

Şekil 4.5'de konum 1 in grafiği görülmektedir. Mavi olan çizgi gerçek koordinat değerini, mavi nokta şeklinde olanlar ise ölçülmüş hatalı değerleri gösterir. Hareketli ortalama, kalman ve lojistik regresyon karşılaştırmalı olarak verilmiştir. Görüldüğü üzere hata oranını gittikçe azaltan kalman filtresidir. Hata oranları daha sonra karşılaştırmalı olarak verilecektir.

Konum 1 enlem gerçek değeri: 39.785988
Konum 1 boylam gerçek değeri 32.812552'dir.

Hareketli ortalama filtresinde parametre olarak 10 kullanılmıştır. Yani ilk 10 verinin ortalaması 11. Verinin değerini vermektedir. Hareketli bir pencere gibi düşünürsek böyle verinin sonuna kadar devam eder. Grafikten anlaşılacağı üzere ilk 10 değer hesaplaması yapılmamıştır. Bu hareketli ortalama filtresinin bir dezavantajıdır. Yeni tahmin değeri geçmişteki verilere çok bağımlıdır. Verilerin çok dağınık olduğu göz önüne alındığında yine de bir miktar iyileştirme yaptığı söylenebilir.

GPS verisindeki hataları azaltmak için lineer regresyon analizinin kullanıldığı bazı makaleler vardır. Lineer regresyon düz bir çizgi şeklinde bir sonuç verdiğinden problemimize uygun çözüm vermemiştir. Sebebi ise her zaman düz bir yolda lineer ilerleme yerine dönüşler, hız ve ivme değişimlerine göre değişkenlik göstermemesidir.

Bu amaçla bağımlı değişkenlere değişiklik göstermesi açısından lojistik regresyon tercih edilmiştir. Farklı parametre değerleri denenmiş olup en az hata oranı parametre değeri 9 iken olduğu tespit edilmiştir. Lojistik regresyon ölçülen değerlere en yakın şekilde doğru elde etmeyi hedefler. Lakin veriler çok dağınık olduğu için verilerin daha yoğun olduğu yere doğru yoğunlaşır. Bu durumda grafiklerden görüldüğü üzere daha az hataya sahip bir veri dizisi elde etmemize sebep olmuştur.

Kalman filtresi ise minimum varyans tahmini yapmayı amaçlayan bir filtredir. Varyans, verilerin aritmetik ortalamadan sapma miktarlarının kareleri toplamıdır. Yani standart sapmanın karekök alınmış halidir. Sisteme verilen girişe ve ölçülen gürültülü çıkışa bakarak sitemin hatasız gerçek durumunu tahmin etmeye çalışır. Projede elimizdeki veri gereği bir tane zamana göre sıralı değer vardır. Bir boyutlu lineer kalman filtresi uygulanmıştır. Kalman filtresinin parametre değerleri GPS verisine uygun olarak ayarlanması gerekmektedir. Daha önceki bölümlerde kalman filtresinin formülleri detaylı bir şekilde açıklanmıştır. Projede kullanılan parametre değerleri aşağıdadır. Kodlar python dilinde yazılmıştır.

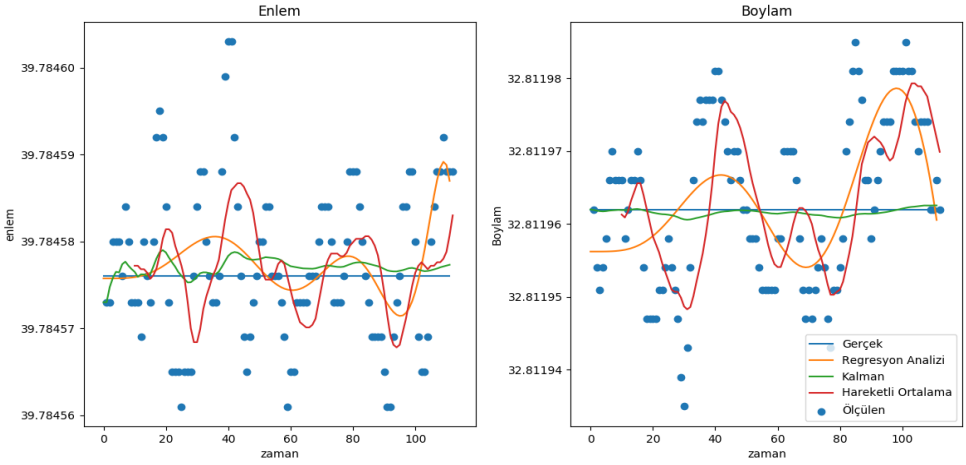
$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad R = \begin{bmatrix} 10000 & 0 \\ 0 & 10000 \end{bmatrix} \quad H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad P = \begin{bmatrix} 10000 & 0 \\ 0 & 10000 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Kalman filtresinde parametre değerleri belli düzenlemeler yapması amacıyla kullanılır. A matrisi durum fark eşitliğini ifade ederken R matrisi ölçüm kovaryansını ifade eder. H matrisi durum eşitlik ölçümlerini, P matrisi başlangıç durum kovaryansını, B matrisi ise durum kontrol eşitliğini ifade eder.

Çizelge 4.5 Konum 1 hata oranı azaltma

Konum 1	Enlem	Boylam
MAE Hareketli Ortalama	14.369697	14.485606
MAE Lojistik Regression	9.985147	12.593720
MAE Kalman	4.320978	1.303553
MSE Hareketli Ortalama	0.339551	0.400679
MSE Lojistik Regression	0.232372	0.297516
MSE Kalman	0.030419	0.003212
RMSE Hareketli Ortalama	0.582710	0.632992
RMSE Lojistik Regression	0.482049	0.545451
RMSE Kalman	0.174412	0.056675



Şekil 4.6 Konum 2 GPS hata azaltma

Konum 2 hata azaltma grafiği şekil 4.6'da verilmiştir. En iyi sonucu veren bu grafikte kalman filtresidir. Ondan sonra da lojistik regresyon analizidir.

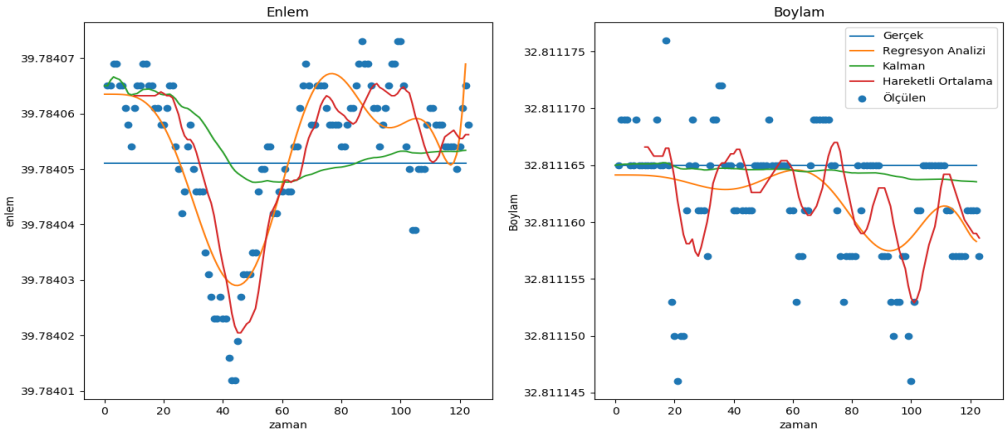
Konum 2 enlem gerçek değeri: 39.784576

Konum 2 boylam gerçek değeri: 32.811962

Çizelge 4.6 Konum 2 hata oranı azaltma

Konum 2	Enlem	Boylam
MAE Hareketli Ortalama	3.599029	7.734951
MAE Lojistik Regression	2.488031	5.809960
MAE Kalman	1.197186	0.508281
MSE Hareketli Ortalama	0.020859	0.082780
MSE Lojistik Regression	0.013332	0.051854
MSE Kalman	0.001970	0.000382
RMSE Hareketli Ortalama	0.144425	0.287715
RMSE Lojistik Regression	0.115463	0.227715
RMSE Kalman	0.044382	0.019543

Çizelge 4.6'dan görüldüğü üzere konum 2 için de en iyi sonucu kalman filtresi vermiştir.



Şekil 4.7 Konum 3 GPS hata azaltma

Konum 3 ün grafiği ise şekil 4.7'deki gibidir. Enlem değerlerinde alınan veri daha fazla hatalı olduğundan dolayı kalman filtresi bile çok iyi sonuç vermemiştir. Boylam değerinde kalman filtresinin çok iyi sonuç verdiği gözlemlenebilir.

Konum 3 enlem gerçek değeri: 39.784051

Konum 3 boylam gerçek değeri: 32.811165

Çizelge 4.7 Konum 3 hata oranı azaltma

Konum 3	Enlem	Boylam
MAE Hareketli Ortalama	10.379825	3.475439
MAE Lojistik Regression	10.377440	2.926829
MAE Kalman	4.912363	0.586339
MSE Hareketli Ortalama	0.159223	0.020673
MSE Lojistik Regression	0.139857	0.013697
MSE Kalman	0.047036	0.000515
RMSE Hareketli Ortalama	0.399028	0.143782
RMSE Lojistik Regression	0.373974	0.117033
RMSE Kalman	0.216877	0.022696

Çizelge 4.7'de görüldüğü üzere nokta 3 için de en iyi sonucu yine kalman filtresi vermiştir. Coğrafi bilgi sistemlerinde koordinat mesafe hesabı 2 şekilde yapılır. Haversine hesaplama yöntemi, Dünya'yı tam bir küre varsayarak hesaplama yapar (Robusto 1957). Vincenty yöntemi ise kutuplardan basık bir küre şeklinde modeller (Vincenty 1975). İki yöntem ile de hata payı mesafeleri hesaplanmıştır.

Çizelge 4.8 Kullanılan yöntemlerin gerçek değer ile uzaklık farkları

	Nokta 1	Nokta 2	Nokta 3
Gerçek Uzaklık Haversine	2.28621203268	1.27326984591	1.40801789288
Regression Haversine	1.61315272884	0.62778070682	0.55635836932
Kalman Haversine	0.49971380215	0.14702156021	1.21700171837
Hareketli Ortalama Haversine	2.0936958237	0.82879824123	1.24026963602
Gerçek Uzaklık Vincenty	2.28621985815	1.27397321428	1.40678048780

Çizelge 4.8 Kullanılan yöntemlerin gerçek değer ile uzaklık farkları (devamı)

	Nokta 1	Nokta 2	Nokta 3
Regression Vincenty	1.61387943262	0.62858035714	1.21563414634
Kalman Vincenty	0.49909929078	0.146821428571	0.55556097560
Hareketli Ortalama Vincenty	2.0939090909	0.82979611650	1.23897368421

Çizelge 4.8'de belirtildiği üzere Haversine ve Vincenty formülleri ile hesaplama yapılmış ve iki koordinat arasındaki mesafe metre cinsinden bulunmuştur. Deney yapılan yer bina yoğunluğunun nispeten az olduğu bir yer olduğundan binalardan kaynaklanan hatalar en aza indirgemiştir. Bulutsuz bir günde yapılan ölçüm de hava durumundan etkilenen GPS hatalarını en aza indirgemiştir. Nokta 1 için ortalama hata 2.28 metredir. Nokta 2 için 1.27 metre, Nokta 3 için 1.40 metredir. Kullanılan algoritmaların ortalama olarak hatayı ne kadar azalttığı çizelgede belirtilmiştir. Kalman Filtresi için yaklaşık 50 santimetre kadar bir hata payı ile çalıştığı söylenebilir. Bu değer kalman filtresinin parametrelerine göre değişmektedir. Farklı parametreler denenip hata payını en aza indirebilen parametreler tercih edilmiştir.

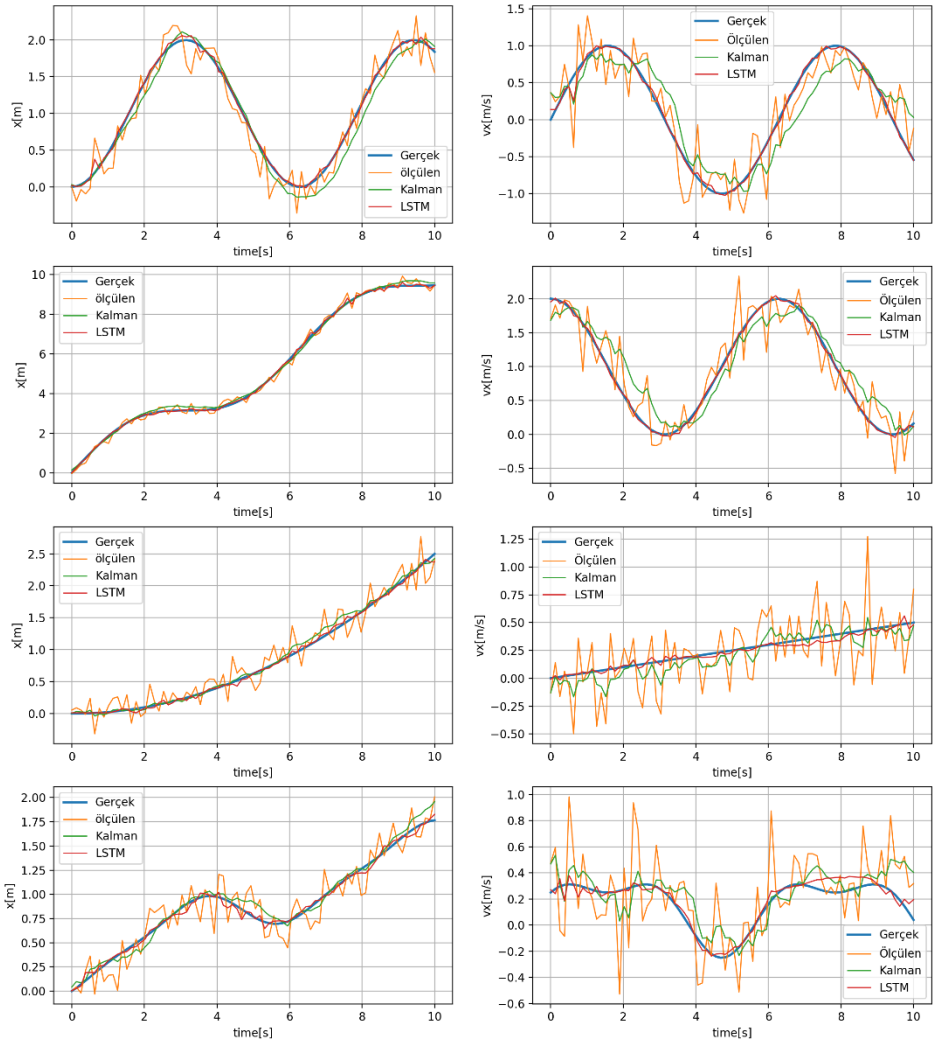
4.3 Bir Sonraki Konum Tahmini

Hata azaltma amacı gütmeden verilerin eksik ve yetersiz kaldığı noktalarda konum verilerini tahmin edebilmek oldukça önem arz etmektedir. GPS verisi yapısı itibarı ile oldukça değişken yapıdadır. Sinyalin bir binaya çarpıp sekmesi ile bile veri o zaman zarfında elde edilmeyebilir. Hava durumu değişiminde, arada alınamayan veriler olmaktadır. Kapalı bir mekâna girilince veriler alınamamaktadır. Bu sebeple veri tamamlama çok önemlidir. Zaman serisi analizi yaparak bir sonraki saniyedeki veriyi doğru tahmin etmeye yönelik uygulamalar yapılmıştır.

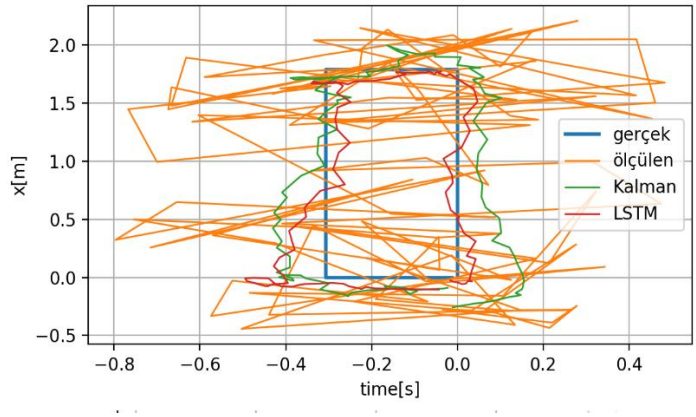
LSTM bir RNN yani geri beslemeli ağların türüdür. Daha önce açıklandığı gibi RNN türünde meydana gelen belirli bir zaman sonra unutma ve geçmiş verileri kullanmadaki

yetersiz kalması üzerine üretilmiştir. Makine öğrenmesi algoritmalarındadır ve günümüzde oldukça popülerdir. LSTM yapısı hata azaltma amacı ile kullanınca oldukça güzel sonuçlar vermektedir. Kalman filtresinden bile daha güzel sonuçlar verdiği uygulamalar vardır. Ancak yeterinde hatalı ve gerçek veri elimizde olmadığından ağın eğitilmesi sağlanamamıştır. Örnek vermek gerekirse: elimizde bir sinüs sinyali olsun. Bu sinyalin çevresinde hatalı bir ölçüm sinyali daha tanımlamış olursak ve LSTM ağına yeterince doğru sinüs sinyali ve hatalı sinüs sinyalini verdiğimizde ağın eğitimi sağlanır. Sonuç olarak da hatalı verilen sinüs sinyalinin gerçek değerini tahmin ettiği gözlemlenmiştir.

Şekil 4.8 de farklı sinyallerin gerçek değerleri, ölçülmüş olan hatalı değerleri kalman filtresi ve LSTM algoritması ile karşılaştırmalı olarak gösterilmiştir. Uygun değerler verildiği zaman LSTM yapısının hata azaltma konusunda kalman filtresinden daha iyi sonuçlar verdiği gözlemlenmiştir. Bu örnekteki LSTM filtresinde batch değeri 4, giriş boyutu 2, çıkış boyutu 1, gizli katman sayısı 10, veri boyutu 80, tahmin edilmeden önce kaç verinin değerlendirmeye alınacağına dair değişken ise 12 olarak belirlenmiştir. Son olarak da epoch sayısı 800 olarak belirlenip test işlemimiz gerçekleştirilmiştir. Farklı sinyal çeşitlerinde hata oranları farklılaştırılıp bu analiz çeşitlendirilmiştir.

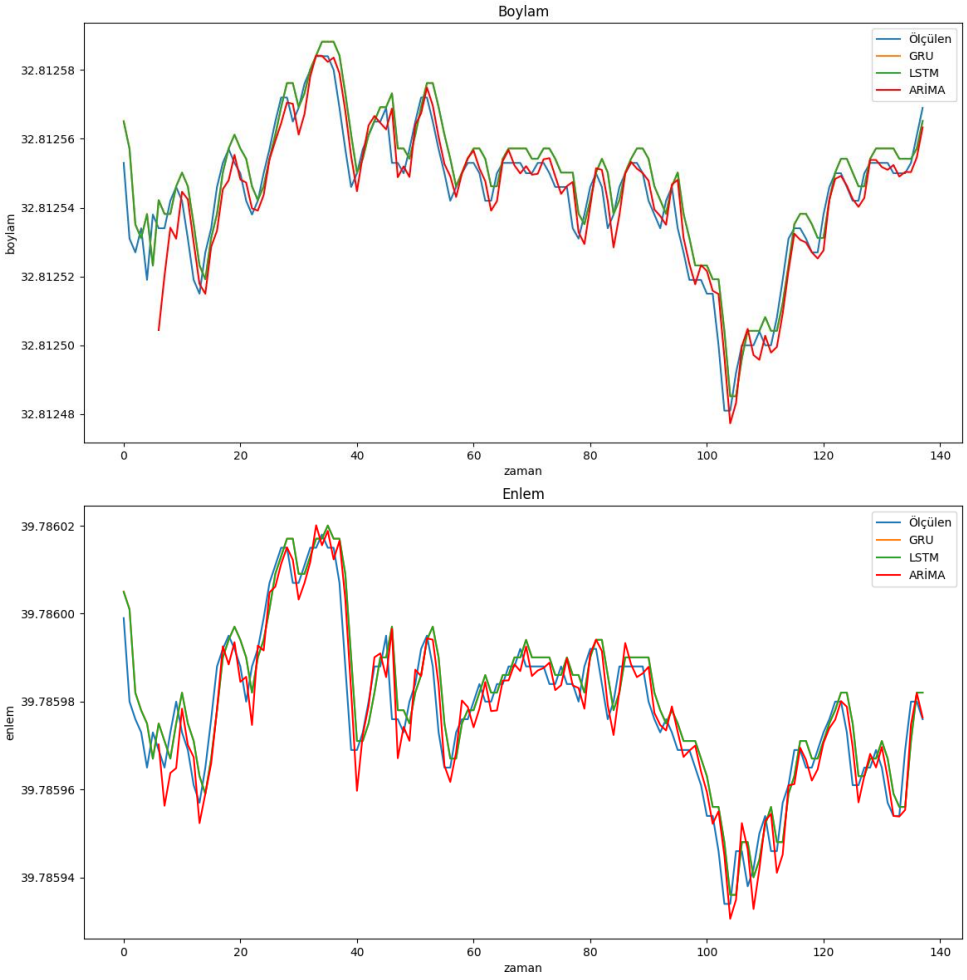


Şekil 4.8 Çeşitli sinyaller için LSTM ve kalman filtresi uygulanması



Şekil 4.9 2 boyutlu veriler için kalman ve LSTM uygulaması

Biraz daha karmaşık bir veri setine sahip şekil 4.9'daki örnekte de gerçek değere yaklaşıldığı gözlemlenmektedir. Daha öncede bahsedildiği gibi elimizde sadece çok sayıda hatalı veri vardır. Doğru ile karşılaştırılabilecek bir durum söz konusu olmadığından kalman filtresi ve LSTM karşılaştırması yapılamamıştır. Düzgün bir test ortamında yapılmak üzere tez sonrasına ertelenmiştir. LSTM yapısına sadece gürültülü veri girişi sağlandığında da beklenen tahmini sonucu vermektedir. Bu şekilde çeşitli kur tahmini uygulamalarında kullanılmaktadır. GPS verisine uygulayıp yine aynı şekilde 3 konum için sonuçları analiz ettik. Konum 1 için grafiğimiz şekil 4.13'de gösterilmiştir. Hata oranları karşılaştırılması da çizelge 4.9'da verilmiştir.



Şekil 4.10 Konum 1 tahmin algoritmaları

Çizelge 4.9 Konum 1 tahmin algoritmaları hata karşılaştırması

Konum 1	Enlem	Boylam
MAE LSTM	0.528	0.612
MAE GRU	0.528	0.613
MAE ARIMA	0.477	0.452
MSE LSTM	0.464	0.635
MSE GRU	0.465	0.637
MSE ARIMA	0.369	0.354
RMSE LSTM	6.812	7.971
RMSE GRU	6.820	7.980
RMSE ARIMA	6.077	5.951

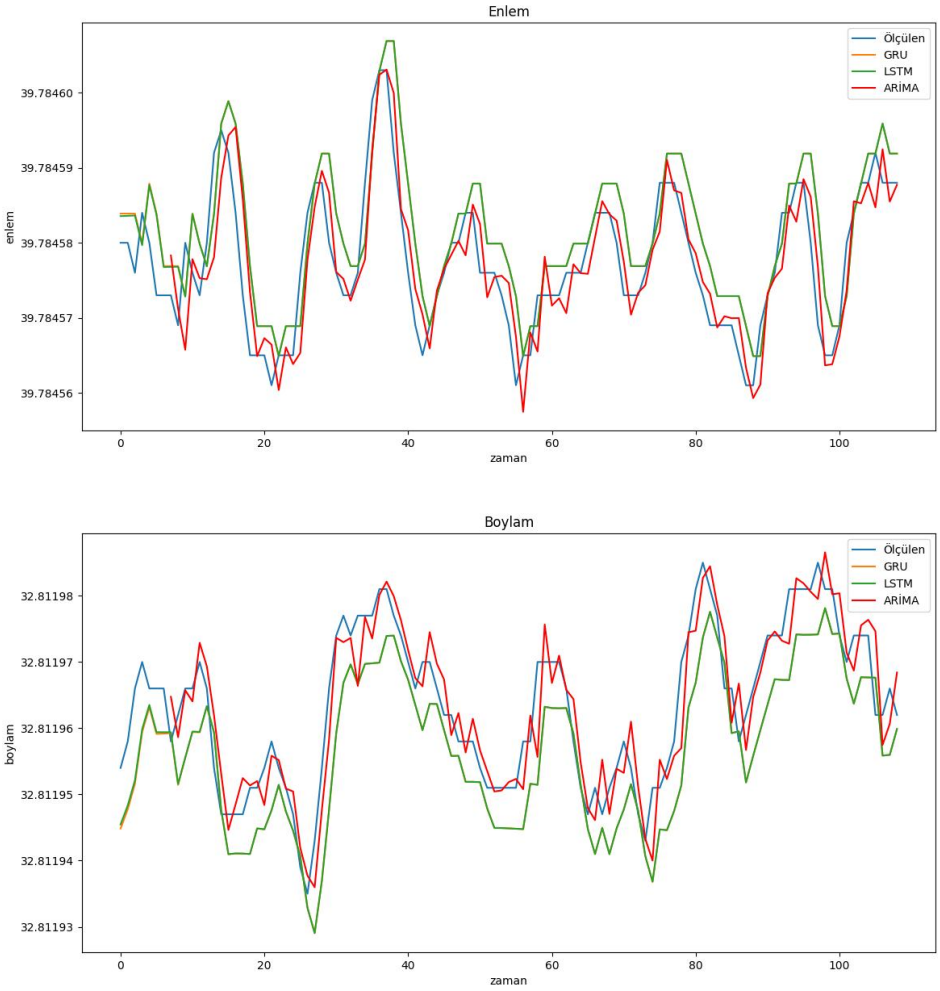
Konum 1 için enlem ve boylam değerlerine karşılaştırmalı olarak LSTM, GRU ve ARIMA uygulanmıştır. Hepsi bir sonraki konumu en doğru şekilde tahmin etmeyi amaçlar. Görüldüğü üzere GRU ve LSTM filtresinin sonuçları yaklaşık olarak aynıdır. İkisi de içerik ve yöntem olarak benzer olmalarına rağmen GRU daha hızlı çalışır. GRU bünyesinde hafıza bloğunu barındırmaz. Farklı bir yaklaşımla tahmin işlemini gerçekleştirir. Daha az parametre kullanır. GRU daha küçük boyutlu verilerde daha hızlı ve etkili sonuç verirken, büyük boyutlu verilerde LSTM kullanmak daha güzel sonuçlar alınmasına sebep olur. ARIMA(p,d,q) şeklinde ifade edilir ve giriş değeri olarak aldığı parametreler tahmin işlemi için önem arz etmektedir. ARIMA(4,1,0) olarak seçilip analizler o şekilde yapılmıştır.

Çizelge 4.10 Konum 2 tahmin uygulamaları hata karşılaştırması

Konum 2	Enlem	Boylam
MAE LSTM	0.394	0.371
MAE GRU	0.393	0.371
MAE ARIMA	0.359	0.367
MSE LSTM	0.252	0.235

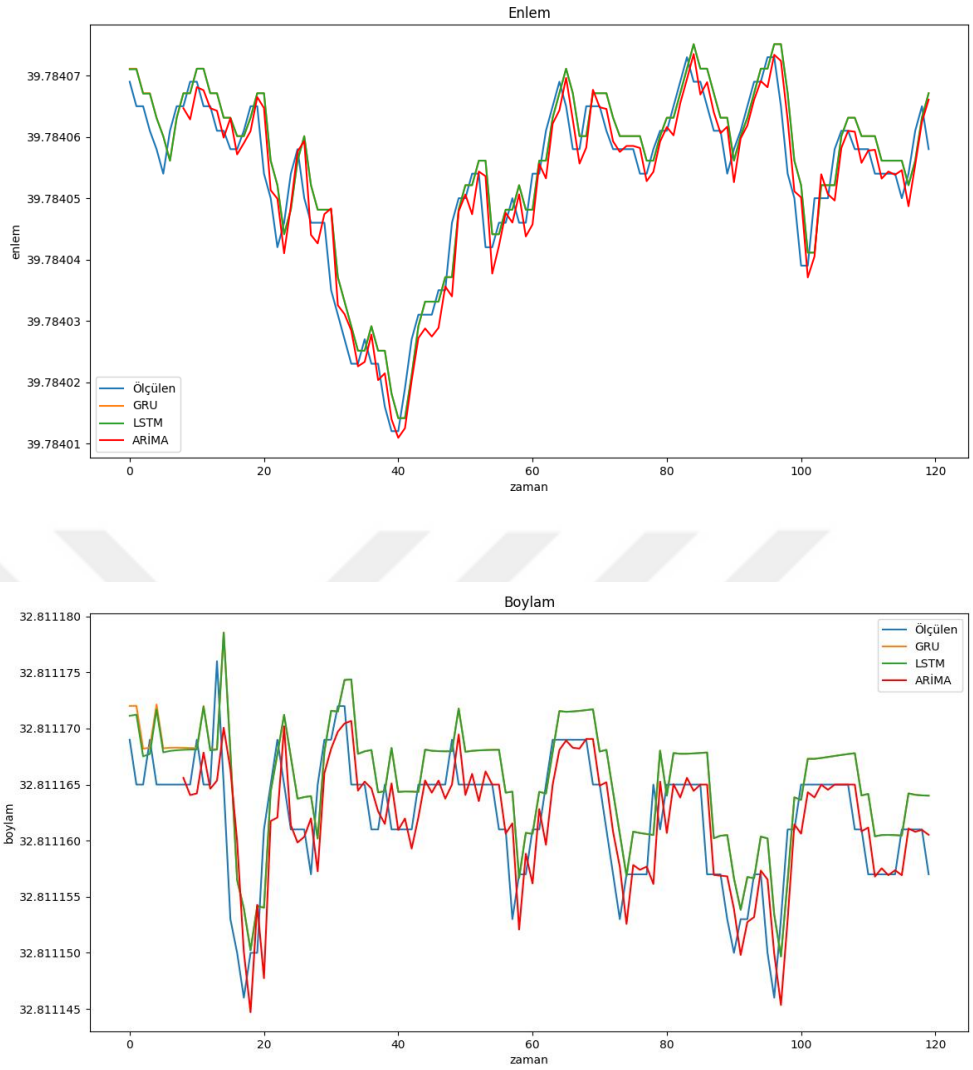
Çizelge 4.10 Konum 2 tahmin uygulamaları hata karşılaştırması (devamı)

Konum 2	Enlem	Boylam
MSE GRU	0.252	0.236
MSE ARIMA	0.235	0.228



Şekil 4.11 Konum 2 tahmin uygulamaları

Konum 2 için bir sonraki nokta tahmin grafikleri ve hata analizleri şekil 4.11'de verilmiştir. Hata oranları da çizelge 4.10'da belirtildiği gibidir. Enlem için GRU LSTM den daha iyi sonuç verirken, boylam için aynı çıkmıştır. ARIMA her iki algoritmadan da daha iyi sonuç vermiştir.



Şekil 4.12 Konum 3 tahmin uygulamaları

Çizelge 4.11 Konum 3 tahmin uygulamaları hata karşılaştırması

Konum 3	Enlem	Boylam
MAE LSTM	0.378	0.254
MAE GRU	0.381	0.252
MAE ARIMA	0.373	0.272
MSE LSTM	0.240	0.130
MSE GRU	0.239	0.129

Çizelge 4.11 Konum 3 tahmin uygulamaları hata karşılaştırması (devam)

Konum 3	Enlem	Boylam
MSE ARIMA	0.228	0.157
RMSE LSTM	4.894	3.606
RMSE GRU	4.886	3.597
RMSE ARIMA	4.770	3.968

Konum 3 için bir sonraki nokta tahmin grafikleri ve hata analizleri şekil 4.12'de verilmiştir. Hata oranları da çizelge 4.11'de belirtildiği gibidir. Enlem değerlerinde hata oranı daha fazla iken boylamda azdır.

Çizelge 4.12 Konum tahmini mesafe ölçümleri

	Nokta 1	Nokta 2	Nokta 3
LSTM Haversine	0.83185518011	0.90371193850	0.61518974113
GRU Haversine	0.83235170497	0.90665778343	0.61736415065
ARIMA Haversine	0.71276156981	0.57181780938	0.83273357391
LSTM Vincenty	0.83213043478	0.90417431192	0.61521666666
GRU Vincenty	0.83260144927	0.90712844039	0.61739166666
ARIMA Vincenty	0.71278625954	0.57195098039	0.83233333333

Çizelge 4.12'de tahmin edilen bir sonraki noktanın gerçekte o konum ile arasındaki uzaklıkların ölçümleri metre cinsinden verilmiştir. Haversine ve Vincenty formülleri kullanarak iki şekilde de hesaplama yapılmıştır. Nokta 1 için en iyi sonucu 0.712 ile ARIMA algoritması vermiştir. Nokta 2 ve nokta 3 için de ARIMA diğer yöntemlerden daha iyi bir sonuca sahiptir. Vincenty ve Haversine yöntemleri arasında çok az fark vardır. Mesafe uzaklığı arttığı zaman hata oranı daha da artacağından Vincenty yönteminin Haversine yönteminden daha doğru sonuç verdiği söylenebilir.

5. SONUÇ

Çalışmada gerçek zamanlı takip ve geofence işlemi gerçekleştirilmiştir. Literatür taramasında anlatıldığı üzere daha önce yapılmış çalışmalar sadece gerçek zamanlı takip sistemi gerçekleştirmektedir. Ek olarak bazılarında basit çapta geofence işlemi yapılmıştır. Gerçekleştirilmiş olan geofence işlemi de işin donanım tarafında koordinatlar arası mesafe hesabı yaparak bir dairenin dışına çıkıp çıkılmadığı kontrol edilerek yapılmaktadır. Gerçekleştirilmiş olan bu çalışmada ise geofence işlemi veriler sunucu tarafına aktarıldıktan sonra harita uygulama ekranında yapılmaktadır. Sadece daire değil kullanıcının çizimine bağlı poligonlar geofence olarak kullanılabilmektedir. Gerçek hayat uygulamalarında düzgün daire şeklinde geofence alanı oldukça kullanışsızdır. Bu konuda iyileştirme yapılmıştır.

Uygulama çocukların takibi için kullanılabilir. Çocuk evinin bahçesinde oynuyor olsun ve geofence alanı evinin çevresini kapsayacak şekilde belirlenirse, çocuk evinden uzaklaştığı zaman uyarı verecektir. Bu senaryoya göre geofence alanı statik ve dinamik olarak 2 şekilde özelleştirilmiştir. Statik ön tanımlı olarak verilmiş olan poligondur. Çocuğun yaşadığı ev, her zaman gittiği okul değişmez alanlar olduğu için statik olarak tanımlanır. Yeni gidilen bir alanda gerçek zamanlı olarak çizilen poligonun kaydetmeden geofence olarak kullanılabilmesi dinamik geofence olarak tanımlanmıştır ve projeye oldukça esnek bir kullanım sağlamıştır.

Web sistemi olarak ise NodeJS kullanımı tercih edilmiştir. Hem sunucu tarafı hem de kullanıcı tarafı javascript dilinde yazılmıştır Kullanıcı kısmına sadece html çıktısı gönderdiği için diğer sistemlerden daha güvenlidir. Leaflet kütüphanesi de açık kaynak kodlu bir harita yazılım kütüphanesidir. Farklı kütüphanelerle birlikte kullanarak kullanıcı dostu bir web uygulaması yazılmıştır.

GPS verisi çeşitli nedenlerle hataya sahiptir. Sistemin daha hızlı çalışması ve istikrarlı olması açısından verilerdeki hata oranının azaltılması hedeflenmiştir. Kaynak özetleri kısmında belirtildiği gibi bu konuya çok fazla odaklanılmış ve çeşitli algoritmalar geliştirilmiş. Bunların en çok tercih edileni Kalman filtresi olmuştur. Çalışmada

uygulanan 3 yöntemin karşılaştırmalı olarak sonuçları verilmiştir. Hareketli ortalama, lojistik regresyon ve kalman filtresi hata azaltma amacı ile uygulanmıştır. Hareketli ortalamada hatayı en çok azaltan parametre değerinin 10 olduğu tespit edilip bu şekilde kullanılmıştır. Bu algoritmada ilk 10 veri için tahmin değeri elde edilmemesi negatif bir durumdur. Kalman filtresinin giriş parametreleri de GPS verisi ile kullanım için önem arz etmektedir. Tezin bulgular ve tartışma bölümünde paylaşılmıştır. Lojistik regresyon analizinde de kullanılan parametre değeri 9 dur. Algoritma sonuçlarımızın doğru çalışıp çalışmadığını anlamak için test edilen GPS verisinin hatalı ve gerçek değerlerine ihtiyaç vardır. Bu sebeple 3 nokta belirlenip bu noktalarda ölçüm yapılmıştır. Alınan konumun gerçek değeri 1-2 cm hata payı olan daha gelişmiş bir GPS cihazı ile yapılmıştır. Bu değer ile karşılaştırarak yapılan çalışmada tüm konumlar için en çok hatayı azaltan kalman filtresidir. Açık havada ve bina yoğunluğunun az olduğu bir ortamda deney yapılmıştır. Bu sebeple ortalama hata değeri verilerimiz için yaklaşık 2 metre civarındadır. Haversine uzaklık hesaplama yöntemi ile hesaplanan Nokta 1 değeri 2,28 metre iken, lojistik regresyon 1,61m, kalman 0,49m ve hareketli ortalama filtresi 2,09 metre ortalama hata payına sahiptir. Karşılaştırmalı grafikler ve hata oranları bir önceki bölümde detaylı anlatılmıştır

GPS verisinde karşılaşılan bir diğer sorun ise kapalı mekânlarda, bina aralarında, hava durumunun kötü olduğu durumlarda alınamamasıdır. Böyle bir durumda verideki hata oranını azaltmak yerine hatalı da olsa verinin gelecek değerini tahmin etmek önem arz etmektedir. Bu bakış açısı ile LSTM, GRU ve ARIMA yöntemleri kullanarak gelecek verinin tahmini yapılmıştır. LSTM ve GRU bir geri beslemeli ağlar türüdür. Yapıları birbirine çok benzediğinden ve ortak parametreler kullanıldığı için analiz sonuçları birbirine çok yakındır. Bazen LSTM daha iyi sonuç verirken bazen GRU daha düşük hata payına sahiptir. Ama genel anlamda test edilen 3 konum için de ARIMA en iyi sonuca sahiptir, yani en az hata payına sahip olacak şekilde sonuç elde edilmiştir. Nokta 1 için Haversine yöntemi ile hesaplanan sonuçta LSTM 0,83 metre, GRU 0,83 metre ve ARIMA 0,71 metre ortalama hata ile bir sonraki konumu tahmin etmiştir.

Bu çalışma çocukların takibi amacı ile kullanılabileceği gibi farklı amaçlar için de özelleştirilebilir. Alzheimer hastaları, zihinsel engelli insanlar için kullanımı gibi

örnekler çoğaltılabilir. Açık hapishane sistemlerinde suçlunun takibi amacı ile bileklik şeklinde kullanımı sağlanabilir. Geofence uygulaması sayesinde ev hapsi istenen suçluların evden uzaklaşma durumunda polise haber vermesi şeklinde bir tasarım yapılabilir.

KAYNAKLAR

Almomani, I. M., Alkhalil, N. Y., Ahmad, E. M. and Jodeh, R. M. 2011. Ubiquitous GPS Vehicle Tracking and Management System.

Anonymous. 2018a. Web Sitesi: https://thesciencegeek.org/2017/01/29/gps/, Erişim Tarihi: 27.11.2018.

Anonymous. 2018b. Web Sitesi: https://tr.gearbest.com/blog/how-to/what-is-gprs-and-what-are-the-advantages-of-it-2849, Erişim Tarihi: 27.11.2018.

Anonymous. 2018c. Web Sitesi: http://diyot.net/arduino-uno-uzerinde-neler-var/, Erişim Tarihi: 27.11.2018.

Anonymous. 2018d. Web Sitesi: https://www.cooking-hacks.com/ documentation/ tutorials/geolocation-tracker-gprs-gps-geoposition-sim908-arduino-raspberry-pi, Erişim Tarihi: 27.11.2018.

Anonymous. 2018e. Web Sitesi: https://www.npmjs.com/, Erişim Tarihi: 27.11.2018.

Anonymous. 2018f. Web Sitesi: https://expressjs.com/, Erişim Tarihi: 27.11.2018.

Anonymous. 2018g. Web Sitesi: https://gulpjs.com/, Erişim Tarihi: 27.11.2018.

Anonymous. 2018h. Web Sitesi: http://colah.github.io/posts/2015-08-Understanding-LSTMs/, Erişim Tarihi: 27.11.2018.

Anonymous. 2018ı. Web Sitesi: https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21, Erişim Tarihi: 27.11.2018.

Anonymous. 2018i. Web Sitesi: https://github.com/Leaflet/Leaflet.draw, Erişim Tarihi: 27.11.2018.

Anonymous. 2018j. Web Sitesi: https://github.com/mapbox/leaflet-pip, Erişim Tarihi: 27.11.2018.

Beeco, J. A., Hallo, J. C., and Brownlee, M. T. J. 2014. GPS Visitor Tracking and Recreation Suitability Mapping: Tools for understanding and managing visitor use. Landscape and Urban Planning, 127, 136–145.

Çayıroğlu, İ. 2012. Kalman Filtresi ve Programlama. Fen ve Teknoloji Bilgi Paylaşımı, 1–6.

El-Medany, W., Al-Omary, A., Al-Hakim, R., Al-Irhayim, S. and Nusaif, M. 2010. A cost effective real-time tracking system prototype using integrated GPS/GPRS module. Proceedings - 6th International Conference on Wireless and Mobile Communications, ICWMC 2010, 521–525.

Feldmann, T., Bauch, A., Piester, D., Esteban, H., Palacio, J., Galindo, F. J., … Schn, S. 2009. GPS carrier phase and precise point positioning time scale comparisons using different software packages. Içinde 2009 IEEE International Frequency Control Symposium Joint with the 22nd European Frequency and Time Forum.

Fleischer, P.B., Nelson, A.Y., Sowah, R.A. and Bremang, A. 2012. Design and development of GPS/GSM based vehicle tracking and alert system for commercial inter-city buses. Proceedings of the 2012 IEEE 4th International Conference on Adaptive Science and Technology, ICAST 2012, 1–6.

Gunnar, H. 2003. GSM Networks: Protocols, Terminology and Implementation. Sensor

Hu, J. M., Li, J. and Li, G. H. 2012. Automobile anti-theft system based on GSM and GPS module. Proceedings - 5th International Conference on Intelligent Networks and Intelligent Systems, ICINIS 2012, 199–201.

Huang, J.Y. and Tsai, C.H. 2008. Improve GPS positioning accuracy with context awareness. Içinde Proceedings - 2008 the 1st IEEE International Conference on Ubi-Media Computing and Workshops, U-Media2008.

Iter, D., Kuck, J. and Zhuang, P. 2016. Target Tracking with Kalman Filtering, KNN and LSTMs. Tarihinde adresinden erişildi http://cs229.stanford.edu/proj2016/report/IterKuckZhuang-TargetTrackingwithKalmanFilteringKNNandLSTMs-report.pdf

Kalman, R.E. 1960. A New Approach to Linear Filtering and Prediction Problems. Journal of Basic Engineering.

Kaplan, E.D. and Hegarty, C. 2005. Understanding GPS: Principles and Applications, Second Edition. Artech House.

Koyuncu, B. and Özdemir, Z. 2016. Real Time Position Detection by Using GPS + GSM + GPRS and Arduino Mega Based Telit GL865, 7(1), 44–48.

Koyuncu, B. and Özdemir, Z. 2016. Real Time Localization and Leaflet Map Geofencing by using Sim900 based GPS / GSM / GPRS, 5(2), 47–52.

Le-Tien, T. and Phung-The, V. 2010. Routing and tracking system for mobile vehicles in large area. Içinde Proceedings - 5th IEEE International Symposium on Electronic Design, Test and Applications, DELTA 2010 297–300.

Lee, S., Tewolde, G. and Kwon, J. 2014. Design and Implementation of Vehicle Tracking System Using GPS / GSM / GPRS Technology and Smartphone Application, (March), 353–358.

Monti, G., Tarricone, L., Corchia, L. and De Benedetto, E. 2016. Wearable logo-antenna for GPS–GSM-based tracking systems. IET Microwaves, Antennas & Propagation, 10(12), 1332–1338.

Singh, A. 2017. Anomaly Detection for Temporal Data using Long Short-Term Memory (LSTM).

Thomas, B., Holland, J.D. and Minot, E.O. 2011. Wildlife tracking technology options and cost considerations. Wildlife Research, 38(8), 653–663.

Verma, P. and Bhatia, J.S. 2013. Design and development of GPS-GSM based tracking system with google map based, 3(3), 33–40.

ÖZGEÇMİŞ

Adı Soyadı **:** Zeynep ÖZDEMİR

Doğum Yeri **:** Sultandağı / AFYONKARAHİSAR

Doğum Tarihi **:**25.03.1993

Medeni Hali **:** Bekar

Yabancı Dili **:** İngilizce

Eğitim Durumu

Lise : Sinop Fen Lisesi

Lisans : Ankara Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü (2016)

Yüksek Lisans : Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı (Eylül 2016 – Ocak 2019)

Çalıştığı Kurumlar

Araştırma Görevlisi, Niğde Ömer Halisdemir Üniversitesi, Ocak 2018 – Aralık 2018

Araştırma Görevlisi, Ankara Üniversitesi, Aralık 2018 – Devam ediyor

Yayınlar

Koyuncu, B. and **Özdemir, Z.** 2016. Real Time Position Detection by Using GPS + GSM + GPRS and Arduino Mega Based Telit GL865, 7(1), 44–48.

Koyuncu, B. and **Özdemir, Z.** 2016. Real Time Localization and Leaflet Map Geofencing by using Sim900 based GPS / GSM / GPRS, 5(2), 47–52.