

**KÜLTÜREL MİRAS ALANLARI İÇİN UZAKTAN
ARTIRILMIŞ GERÇEKLİK SİSTEMİ**

**DISTANT AUGMENTED REALITY SYSTEM FOR
CULTURAL HERITAGE SITES**

METEHAN ÜNAL

Doç. Dr. SÜLEYMAN TOSUN

Tez Danışmanı

Yrd. Doç. Dr. GAZİ ERKAN BOSTANCI

Yardımcı Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Bilgisayar Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2017

METEHAN ÜNAL' ın hazırladığı "**Kültürel Miras Alanları İçin Uzaktan Artırılmış Gerçeklik Sistemi**" adlı bu çalışma aşağıdaki jüri tarafından **BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI'** nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Doç. Dr. Suat ÖZDEMİR

Başkan

Doç. Dr. Süleyman TOSUN

Danışman

Yrd. Doç. Dr. Mehmet Serdar GÜZEL

Üye

Yrd. Doç. Dr. Adnan ÖZSOY

Üye

Yrd. Doç. Dr. Gönenç ERCAN

Üye

Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından **YÜKSEK LİSANS TEZİ** olarak onaylanmıştır.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU

Fen Bilimleri Enstitüsü Müdürü

YAYINLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

- Tezimin/Raporumun tamamı dünya çapında erişime açılabilir ve bir kısmı veya tamamının fotokopisi alınabilir.

(Bu seçenekle teziniz arama motorlarında indekslenebilecek, daha sonra tezinizin erişim statüsünün değiştirilmesini talep etmeniz ve kütüphane bu talebinizi yerine getirirse bile, tezinin arama motorlarının önbelleklerinde kalmaya devam edebilecektir.)

- Tezimin/Raporumun 28/07/2017 tarihine kadar erişime açılmasını ve fotokopi alınmasını (İç Kapak, Özet, İçindekiler ve Kaynakça hariç) istemiyorum.

(Bu sürenin sonunda uzatma için başvuruda bulunmadığım takdirde, tezimin/raporumun tamamı her yerden erişime açılabilir, kaynak gösterilmek şartıyla bir kısmı ve ya tamamının fotokopisi alınabilir)

- Tezimin/Raporumuntarihine kadar erişime açılmasını istemiyorum, ancak kaynak gösterilmek şartıyla bir kısmı veya tamamının fotokopisinin alınmasını onaylıyorum.

- Serbest Seçenek/Yazarın Seçimi

28 / 07 / 2017

(İmza)



Öğrencinin Adı Soyadı

Metehan ÇINAR

ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

25/07/2017


İMZA

Metehan Ünal

ÖZET

KÜLTÜREL MİRAS ALANLARI İÇİN UZAKTAN ARTIRILMIŞ GERÇEKLİK SİSTEMİ

Metehan ÜNAL

Yüksek Lisans, Bilgisayar Mühendisliği

Tez Danışmanı: Doç. Dr. Süleyman TOSUN

Yardımcı Tez Danışmanı: Yrd. Doç. Dr. Gazi Erkan BOSTANCI

TEMMUZ 2017, 90 sayfa

Artırılmış Gerçeklik kavramı, gerçek dünya görüntüsü ile bilgisayarda üretilmiş ses, metin, resim ya da 3B modellerin birleştirilmesi olarak tanımlanabilir. 90lı yıllarda ortaya çıkmış bu kavram o yıllarda sadece akademik alana hitap ederken, mobil cihazların yaygınlaşmasından sonra toplumun her kesimine hitap eder hale gelmiştir.

Kültürel miras alanları için yapılan Artırılmış Gerçeklik çalışmaları, tarihi alanlara ilgiyi artırmak adına önemli bir konudur. Kadim medeniyetlerden izler taşıyan, zamanın yıkıcı etkisi ile harabe haline gelmiş ve çok az bir kısmı ayakta kalabilmiş tarihi binaların aslına uygun olarak inşa edilmesi oldukça zahmetli ve masraflı bir iştir. Artırılmış Gerçeklik ile tarihi binaların fiziksel olarak yeniden inşasına gerek duyulmadan, buldukları alan üzerinde sanal olarak bindirilmesi mümkündür.

Bu tez çalışmasında Ankara Roma Hamamları için mobil konum tabanlı artırılmış gerçeklik uygulaması prototipi geliştirildi ve bir uzaktan artırılmış gerçeklik sistemi önerildi. Bu proje kapsamında, Ankara Roma Hamamları alanında çalışmak için istenilen izinler alınmadığı için tüm çalışmalar Ankara Üniversitesi Gölbaşı yerleşkesinde yapılmıştır. Bu amaçla öncelikle mobil cihazlar üzerinde konum alma işlemleri yapıldı ve

konum verilerinin tutarlılıkları test edildi. Konum verilerinin kayan pencere yöntemi ile filtrelenmesine karar verildi. İkinci aşamada 3B modellerin elde edildi ve bunların üzerinde iyileştirmeler yapıldı. Sonraki adımda Artırılmış Gerçeklik yazılım geliştirme araçları işaretçi tabanlı olarak test edildi. Yazılım geliştirme araçlarından konum tabanlı sistemlere desteği olanlarla uygulama geliştirildi ancak projenin Unity 3D oyun motoru kullanılarak tasarlanmasına karar verildi. Unity ile geliştirilen konum tabanlı uygulamadan sonra uzaktan artırılmış gerçeklik sistemi anlatıldı. Son olarak da dronedan görüntü alınmasını sağlayacak olan Android uygulaması geliştirildi. Uygulama yardımıyla dronedan görüntü alındı ve kaydedildi. Kaydedilen video Unity'e aktarıldı ve 3B hamam modeli video üzerine bindirildi.

Anahtar Kelimeler: Artırılmış Gerçeklik, Kültürel Miras, GPS, Drone.

ABSTRACT

DISTANT AUGMENTED REALITY SYSTEM FOR CULTURAL HERITAGE SITES

Metehan ÜNAL

Master of Science, Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Süleyman TOSUN

Co-Supervisor: Asst. Prof. Dr. Gazi Erkan BOSTANCI

July 2017, 90 Pages

The concept of Augmented Reality can be described as the combination of real-world imagery with sound, text, images, or 3D models produced on a computer. This concept, which emerged in the 1990s, while addressing only the academic field in those years, appealed to all segments of the society after the spread of mobile devices.

Studies of Augmented Reality for cultural heritage is important issue to enhance interest in historical sites. It's a troublesome and costly task to reconstruct historical buildings that have traces of ancient civilization, ruined by the destructive effect of time, and few survived. With Augmented Reality, It's possible to virtually overlay the historical buildings in situ, without requiring physical reconstruction.

In this thesis an mobile location based augmented reality application prototype was developed and a distant augmented reality system was presented for Roman Baths in Ankara. Within the scope of this project, all the work was done in Ankara University Gölbaşı Campus because the required permissions could not obtained to work in the area of Ankara Roman Baths. Firstly, location operations were performed on mobile devices and location data was tested. It's decided to filter the location data with sliding window model. Secondly, 3D model was obtained and restorations were made on the model. Later augmented reality software development kits were tested for marker based applications.

Applications were developed with software development kits which supports location based systems but it's decided to design the project with only using Unity 3D game engine. After the location based application developed with Unity, the distant augmented reality system was introduced. Finally, an Android application was developed for taking the drones video stream. With the help of the application, the video from drone's camera is taken and recorded. The recorded video was transferred to Unity and the 3D model of bath was overlaid on top of the video.

Key Words: Augmented Reality, Cultural Heritage, GPS, Drone.

TEŞEKKÜR

Tez çalışmam süresince beni yönlendiren, bilgi ve tecrübeleri ile destekleyen, sonsuz sabırla beni her zaman çalışmaya teşvik eden ve zor zamanlarda yardımlarını asla eksik etmeyen danışmanlarım Sayın Doç. Dr. Süleyman TOSUN'a (Hacettepe Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) ve Sayın Yrd. Doç. Dr. Gazi Erkan BOSTANCI'ya (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) en derin sevgi ve saygılarımla teşekkür ederim.

Çalışmalarım süresince desteklerini esirgemeyen Ankara Üniversitesi Bilgisayar Mühendisliği Bölüm başkanı Sayın Prof. Dr. İman ASKERBEYLİ'ye (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) ve Hacettepe Üniversitesi Bilgisayar Mühendisliği Bölüm Başkanı sayın Mehmet Önder EFE'ye en derin sevgi ve saygılarımla teşekkür ederim.

Hayatım boyunca karşılaştığım tüm zorluklarda desteği ile yanımda olan, lisansüstü eğitimime başlamam konusunda telkinleri ile akademik hayata atılmamı sağlayan ve tüm eğitim hayatım boyunca benden maddi, manevi desteklerini asla esirgemeyen babama, anneme ve kardeşime en derin sevgi ve saygılarımla teşekkür ederim.

Birlikte geçirdiğimiz zaman içinde iyi ve kötü günü gördüğümüz, bu süreçte her zaman bana destek olan eşime, maddi ve manevi her anlamda yanımda olan değerli ailesine en derin sevgi ve saygılarımla teşekkür ederim.

Ayrıca bu tez çalışmamı 3001:215E156 numaralı “Kültürel Miras Alanları için Mobil ve Uzaktan Artırılmış Gerçeklik Sistemi Geliştirilmesi” isimli proje kapsamında destekleyen Türkiye Bilimsel ve Teknolojik Araştırma Kurumuna (TÜBİTAK) teşekkür ederim.

İÇİNDEKİLER

ÖZET.....	i
ABSTRACT	iii
TEŞEKKÜR.....	v
ŞEKİLLER	viii
ÇİZELGELER	x
SİMGELER VE KISALTMALAR.....	xi
1. GİRİŞ	1
2. KURAMSAL TEMELLER VE KAYNAK ÖZETLERİ.....	3
2.1. Artırılmış Gerçeklik	3
2.1.1. Artırılmış Gerçeklik ve Sanal Gerçeklik	5
2.1.2. Artırılmış Gerçeklik Uygulama Alanları.....	5
2.2. Kullanıcı İzleme Yöntemleri.....	11
2.2.1. Kapalı Alan.....	12
2.2.2. Açık Alan	16
2.3. Görüntüleme Yöntemleri.....	18
2.3.1. Video-Görme	18
2.3.2. Optik-Görme	19
2.3.3. Yansıtmalı-Görme.....	20
2.4. Kültürel Miras Alanları İçin Artırılmış Gerçeklik	22
3. KULLANICI TAKİBİ	24
3.1. Kullanılan Sensörler.....	24
3.1.1. Küresel Konumlama Sistemi	24
3.1.2. Jiroskop.....	25
3.2. Koordinat Sistemi Dönüşümü	26
3.3. Konum Testleri.....	28
3.4. Konum Bilgilerini Filtreleme.....	31
4. MODELLERİN OLUŞTURULMASI.....	33
4.1. Modelleme	33
4.2. Optimizasyon	35
5. GELİŞTİRİLEN İŞARETÇİ TABANLI UYGULAMALAR.....	37
5.1. Artırılmış Gerçeklik Yazılım Geliştirme Araçları.....	37
5.2. Yazılım Geliştirme Ortamı	39
5.3. Yazılım Geliştirme Araçları ile İşaretçi Tabanlı Olarak Geliştirilen Uygulamalar	40

5.3.1. ARToolKit ile Geliştirilen Uygulama	40
5.3.2. Vuforia ile Geliştirilen Uygulama	43
5.3.3. Wikitude ile Geliştirilen Uygulama	46
5.3.4. Kudan AR SDK ile Geliştirilen Uygulama	48
6. GELİŞTİRİLEN KONUM TABANLI MOBİL ARTIRILMIŞ GERÇEKLİK UYGULAMALARI	51
6.1. Projenin Planı	51
6.2. Ankara Roma Hamamı	52
6.3. Artırılmış Gerçeklik Yazılım Geliştirme Araçları ile Geliştirilen Uygulamalar	53
6.3.1. Wikitude ile Geliştirilen Uygulama	53
6.3.2. Vuforia ile Geliştirilen Uygulama	59
6.4. Geliştirilen Asıl Mobil Yerde Artırılmış Gerçeklik Uygulaması	64
6.4.1. Giriş Sahnesi	65
6.4.2. Artırılmış Gerçeklik Sahnesi	66
6.5. Uzaktan Artırılmış Gerçeklik Sistemi	76
6.5.1. Geliştirilen Uygulama	77
7. SONUÇ	81
KAYNAKLAR	83
ÖZGEÇMİŞ	90

ŞEKİLLER

Sayfa

Şekil 2.1. Kamera görüntüsünün üzerine model bindirilerek AG elde edilmiş hali.....	3
Şekil 2.2. (a) 2B AG uygulaması [6] ve (b) 3B AG uygulaması örnekleri [84].	4
Şekil 2.3. Milgram ve Kishino'ya göre Sanallık Süreci [3].	5
Şekil 2.4. VWRG ve Metaio tarafından geliştirilen uygulama [13].....	7
Şekil 2.5. Motor Tasarım Kritiği [12]	7
Şekil 2.6. Pokemon Go [19].....	8
Şekil 2.7. Google Translate ile İngilizceden Türkçeye Çeviri	9
Şekil 2.8. Snapchat uygulamasından alınmış 3 görüntü.....	10
Şekil 2.9 Kullanıcı İzleme Yöntemleri [28]	11
Şekil 2.10. Dıştan-içe (Sol) ve İçten-dışa (Sağ) yöntemler [36]	13
Şekil 2.11. TaPuMa (Tangible Public Map) Sistemi[39].....	14
Şekil 2.12. Bostancı ve arkadaşlarının [42] çalışması.....	14
Şekil 2.13. Model tabanlı açık alan AG mimarisi [78].	17
Şekil 2.14. Video-Görme Kafaya Monte Ekran mimarisi [82].	19
Şekil 2.15. Optik-görme kafaya monte ekran sistemi [82].....	20
Şekil 2.16. Kafaya Monte Yansıtıcı[88].....	21
Şekil 2. 17.Kafaya monte yansıtıcı aracılığı ile yansıtılmalı görme [88].	21
Şekil 2.18. Angelopoulou ve arkadaşlarının çalışmalarından işaretçi tabanlı AG örneği [96].	23
Şekil 3.1. Dünya Koordinat Sistemi	25
Şekil 3.2. Yerel Koordinat Sistemi.....	26
Şekil 3.3. GPS verisinin ECEF formatına dönüşümü [51].....	27
Şekil 3.4. (a) Kapalı alanda kablosuz internet bağlantısı kullanıldı. (b) Açık alanda bir mobil cihaz hücrel veri noktası olarak internete bağlanmak için kullanıldı.	28
Şekil 3.5. Geliştirilen uygulamanın arayüzü	28
Şekil 3.6. (a) Kapalı alanda ve (b) açık alanda izlenen yol.....	29
Şekil 3.7. Kapalı alanda 3 sabit kullanıcı ile yapılan testin sonuçları	29
Şekil 3.8. Kapalı alanda 3 hareketli kullanıcı ile yapılan testin sonuçları.....	30
Şekil 3.9. Açık alanda 3 sabit kullanıcı ile yapılan testin sonuçları	30
Şekil 3.10 Açık alanda 3 hareketli kullanıcı ile yapılan testin sonuçları.....	31
Şekil 3.11. Kayan Pencere filtreleme yöntemi	32
Şekil 4.1. Ankara Roma Hamamı Çizimi [53]	33
Şekil 4.2. Tasarlanan Model.....	34
Şekil 4.3. Kullanılan dokular : (a) Binaların ortasındaki alan, (b) mermer alanlar, (c) bina duvarları, (d) bina çatıları.	34
Şekil 4.4. Hamamın 3B modeli	34
Şekil 4.5. (a) Tasarlanan bir bina modeli ve (b) multires ile köşe sayısı %50 azaltılmış hali.....	35
Şekil 5.1. Geliştirilen Uygulamalarda Kullanılan Model [71]	40
Şekil 5.2. ARToolKit için varsayılan işaretçiler (a) Hiro ve (b) Kanji kalıbı	41
Şekil 5.3. ARToolKit ile geliştirilen işaretçi tabanlı AG uygulamasından alınan görüntü.	42
Şekil 5.4. (a) Sistemde 2 yıldız alan bir resim ve (b) 5 yıldız alan bir resim.	43

Şekil 5.5. Ankara Üniversitesi logosunun özellikleri çıkartılmış hali.....	44
Şekil 5.6. Vuforia ile geliştirilen işaretçi tabanlı AG uygulamasından alınan bir görüntü.	46
Şekil 5.7. Wikitude ile geliştirilen işaretçi tabanlı AG uygulamasından alınan bir görüntü.....	48
Şekil 5.8. Kudan ile geliştirilen işaretçi tabanlı AG uygulamasından alınan bir görüntü.	49
Şekil 6.1. Yerde Artırılmış Gerçeklik Uygulamasının Konsepti.....	51
Şekil 6.2. Roma Hamamı'nın yukarıdan görünüşü (70).....	52
Şekil 6.3. Hamamın bulunduğu alandan görüntüler (70).	53
Şekil 6.4. Konum tabanlı AG uygulamasının menü planı.....	54
Şekil 6.5. Wikitude ile geliştirilen konum tabanlı uygulamanın giriş menüsü.	55
Şekil 6.6. Wikitude ile geliştirilen konum tabanlı uygulamanın ayarlar menüsü.	55
Şekil 6.7. Wikitude ile geliştirilen konum tabanlı uygulamanın kamera görüntüsü alanı.....	56
Şekil 6. 8. Uygulamanın AG kısmının sınıfları arası ilişkiler.	57
Şekil 6.9. Wikitude ile geliştirilen konum tabanlı AG uygulamasından bir görüntü.	59
Şekil 6.10. Vuforia ile geliştirilen uygulamanın giriş menüsü.	60
Şekil 6.11. Başlat butonunun çalışması.....	60
Şekil 6.12. Konum güncelleme işleminin akış şeması.	61
Şekil 6.13. Yeterli sayıda güncelleme alındıktan sonra giriş ekranı.	62
Şekil 6.14. Kamera sahnesinin tasarımı.	62
Şekil 6. 15. Vuforia ile geliştirilen uygulamadan bir görüntü.....	63
Şekil 6.16. Uygulamanın giriş menüsü GPS güncellemesi alırken.	65
Şekil 6.17. Çıkış butonuna tıklandığında ön plana gelen ara menü.	66
Şekil 6.18. Artırılmış Gerçeklik ekranı tasarımı.	67
Şekil 6.19. Kamera görüntüsünün alınması.....	69
Şekil 6.20. Gyro Manager sınıfının görevi.....	70
Şekil 6.21. Çalışan uygulamadan alınan görüntü.	74
Şekil 6. 22. Modelin tamamı eklenmiş uygulamadan alınan bir görüntü.....	75
Şekil 6.23. Uzaktan artırılmış gerçeklik sistemi.....	76
Şekil 6.24. DJI Phantom 3 Pro drone ve uzaktan kumandası.....	77
Şekil 6.25. Drone için geliştirilen uygulamanın giriş menüsü.	78
Şekil 6.26. Drone için geliştirilen uygulamanın kamera menüsü.....	79
Şekil 6.27. Droneun uçuşu sırasında alınan bir görüntü.....	79
Şekil 6.28. Dronedan alınan video Unity'e aktarılıp modelin üzerine bindirilmiş hali.	80

ÇİZELGELER

	<u>Sayfa</u>
Çizelge 4.1. Modelin içindeki bir binaya multires uygulandığında geometrik şekil sayısı.....	36
Çizelge 5.1. Yazılım Geliştirme Kitleri Karşılaştırması [54].....	37
Çizelge 5.2. ARToolKit ile geliştirilen uygulamadaki nesnelere.....	42
Çizelge 6.1. Location sınıfının alanları ve metotları.	71
Çizelge 6.2. Cartesian Coordinate sınıfının alanları ve metotları.	72
Çizelge 6.3. Tek Bina Modeli için saniye başına kare sayısı değişimi.	74
Çizelge 6.4. Hamamın tamamı eklendiğinde saniye başına kare sayısı değişimi.	75

SİMGELER VE KISALTMALAR

Simgeler

φ	enlem
λ	boylam
h	yükseklik
a	Dünya'nın ekvatordaki yarıçapı
e^2	Dünya'nın dış merkezliliği

Kısaltmalar

AG	Artırılmış Gerçeklik
SG	Sanal Gerçeklik
HMD	Head Mounted Display (Kafaya Monte Ekran)
USB	Universal Serial Bus (Evrensel Seri Yolu)
GPS	Global Positioning System (Küresel Konumlama Sistemi)
2B	İki Boyutlu
3B	Üç Boyutlu
ABD	Amerika Birleşik Devletleri
LLA	Latitude - Longitude - Altitude (Enlem - Boylam - Yükseklik)
ECEF	Earth Centered Earth Fixed
MB	Megabyte
SDK	Software Development Kit (Yazılım Geliştirme Aracı)
IOS	iPhone Operating System (iPhone İşletim Sistemi)
API	Application Programming Interface (Uygulama Programlama Arayüzü)
SLAM	Simultaneous Localization and Mapping (Eşzamanlı Yerelleştirme ve Planlama)

1. GİRİŞ

Tarihçiler Anadolu'da insan izlerinin Üst Paleolitik Çağ'a (MÖ 50.000-10.000) kadar uzandığını bulmuşlardır¹. Tarihi on binlerce yıl geriye dayanan bu topraklarda modern insanın ataları yaşamlarını sürdürürken gelecek nesillere de aktarılacak binlerce eser yapmıştır. Anadolu'nun hemen her köşesinde bu eserlerin izlerine rastlamak mümkündür. Ancak zamanın aşındırıcı etkisi, savaşlar, doğal afetler gibi nedenlerle bu yapıtların bir çoğu yok olma noktasına gelmiştir. Bazı alanlarda yapıların kalıntıları üzerine fiziksel geriçatım uygulanmıştır.

Bu kadar eski tarihe sahip topraklar üzerine kurumuş olan ülkemizde turizm endüstrisi, doğal olarak, en önemli sektörlerin başında gelmektedir. Türkiye İstatistik Kurumu'nun verilerine göre 2016 yılında Türkiye'ye yaklaşık 32 milyon ziyaretçi gelmiş ve ülke 22 milyar dolarlık bir gelir sağlamıştır.

Kültürel miras alanları açısından oldukça zengin olan Türkiye'de turizm endüstrisinin önemli bir kısmını kültür turizmi oluşturmaktadır. Ülkedeki kültürel miras alanlarını korumanın ve geleceğe aktarmanın bu açıdan önemi büyüktür.

Kültürel miras alanlarının yeniden inşası oldukça zaman alan ve masraflı bir işlemdir. Ayrıca inşa sırasında kalıntılara zarar verilmesi olasılığı da göz önünde bulundurulmalıdır.

Bu bilgiler ışığında daha önce büyük medeniyetler tarafından inşa edilen fakat günümüze çeşitli nedenlerden dolayı ancak kalıntıları ulaşabilmiş eserleri gelecek nesillere aktarmak oldukça önemli bir problemdir. Artırılmış Gerçeklik (AG) teknolojisi ile fiziksel olarak geriçatım gerektirmeden kültürel miras alanlarındaki kalıntılar üzerine asıllarına uygun olarak tasarlanmış 3B modelleri yerleştirme mümkündür.

AG, gerçek dünya görüntüsü üzerine sanal resimler, nesnelere ve sesler bindirerek gerçekte ortamda bulunmayan bilgilerle insana daha ilgi çekici bir ortam sunabilir. Kültürel miras alanları için bu teknolojiyi kullanmak hem kalıntılara zarar vermemek hem de yeniden inşa işleminin alacağı zaman ve maliyetten kurtulmak adına oldukça avantajlı bir işlem olacaktır.

AG uygulamalarında karşılaşılan temel problemlerden biri uygulamanın bulunduğu mobil cihazın baktığı açının sağlıklı olarak algılanabilmesidir. Ayrıca gerçek zamanlı olarak

¹ Early Upper Paleolithic shell beads at Üçağızlı Cave I (Turkey): Technology and the socioeconomic context of ornament life-histories - Mary C. Stiner a,*, Steven L. Kuhn a, Erksin Güleç b

cihazın konumunun ve yöneliminin algılanması ve uygulamada kullanılması da AG adına önemli bir konudur.

Bu tez kapsamında, Ankara'da bulunan Roma Hamamı kalıntılarının üzerine uzman arkeologlar tarafından oluşturulmuş çizimlerden hareketle tasarlanan 3B modeller bindirilmesi amaçlanmıştır. Gösterim için Android tabanlı tablet ve mobil telefonlar kullanılmıştır.

Uygulama için kullanılacak modeller tasarlandıktan sonra uygulamanın mobil cihazlar üzerinde GPS (Global Positioning System, Küresel Konumlama Sistemi) verileri kullanılarak modellerin dünya üzerinde bir konuma yerleştirilmesi için testler yapılmıştır. Bu kapsamda mobil cihazların GPS verileri gözlemlenmiş ve filtre uygulanmıştır.

İkinci aşamada AG yazılım geliştirme araçları (SDK, Software Development Kit) test edildi. Öncelikle işaretçi tabanlı sonra konum tabanlı olarak uygulamalar geliştirildi.

Sonraki aşama olarak mobil cihazlarda görüntünün alınacağı ve üzerine modellerin bindirileceği ortam için çalışmalar yapıldı ve uygulamanın Unity 3D oyun motoru üzerinde herhangi bir AG kütüphanesi kullanmadan geliştirilmesine karar verildi. Bu aşamada Ankara Roma Hamamları üzerine çalışılması için Kültür ve Turizm Bakanlığında gerekli izin alınamadı bu sebeple bu çalışmadaki tüm deneyler Ankara Üniversitesi Gölbaşı Yerleşkesi'nde yapılmıştır.

Unity 3D kullanılarak bir konum tabanlı uygulama prototipi geliştirildi ve deneyler yapıldı. Sonuçlar gözlemlendi ve AG'nin modern cihazlar üzerinde uygulanabilirliğine dair testler yapıldı.

Projenin son aşamasında "Uzaktan Artırılmış Gerçeklik" kavramını gerçekleştirebilmek adına drone (quadkopter) çeşitleri üzerinde araştırma yapıldı. DJI firmasının Phantom 3 Professional modelli droneu üzerinde karar kılındı.

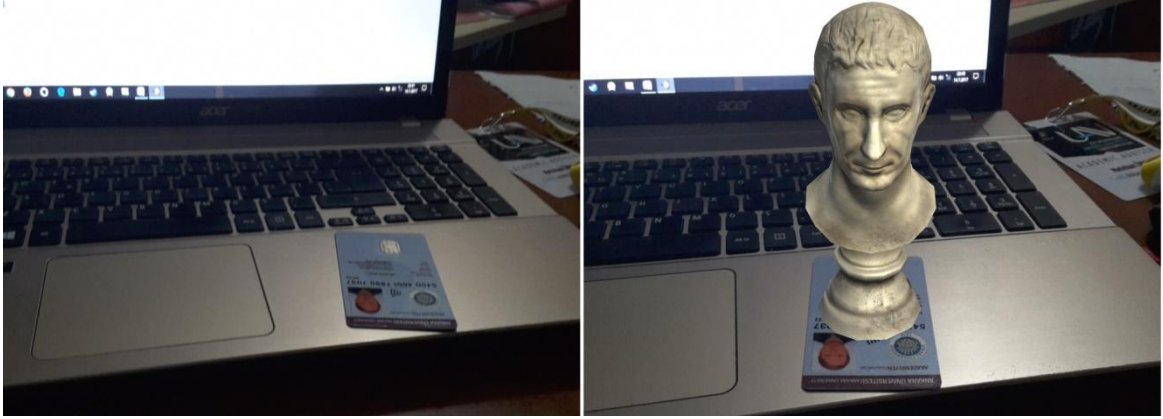
Dronedan görüntü alınması için uygulama geliştirildi. DJI SDK'nın Unity desteği bulunmadığı için video kaydedilerek Unity'e aktarıldı ve üzerine model bindirilmesi çalışması yapıldı.

2. KURAMSAL TEMELLER VE KAYNAK ÖZETLERİ

Bu bölümde tez kapsamında çalışılan konular üzerinde daha önceki çalışmalar özetlenmiştir ve dört alt başlıktan oluşmaktadır. İlk başlıkta önce Artırılmış Gerçeklik kavramı açıklanmış, sonra Sanal Gerçeklik ile ilişkisi incelenmiş, ilk bölümün son kısmında ise uygulama alanları özetlenmiştir. İkinci kısımda ise AG teknolojisinde kullanıcı takip yöntemleri açık ve kapalı alanlar olmak üzere iki başlık altında tartışılmıştır. Üçüncü bölümde AG teknolojisinde görüntüleme yöntemleri üç bölüme ayrılmış ve incelenmiştir. Son bölümde ise literatürdeki kültürel miras alanları için artırılmış gerçeklik çalışmaları incelenmiştir.

2.1. Artırılmış Gerçeklik

AG, bilgisayarda üretilmiş sanal içerik ile gerçek dünya görüntülerini birleştirerek kullanıcıya daha ilgi çekici bir içerik sunar. Kullanıcının gerçekte gördüğünden daha fazlasını algılamasına yardımcı olur ve karmaşık yapıların anlaşılmasını kolaylaştırır [1]. Gerçek dünya görüntüsü üzerine 3B nesnelere, video, resim, ses veya metin yerleştirilerek kullanıcının algısı artırılır.



Şekil 2.1. Kamera görüntüsünün üzerine model bindirilerek AG elde edilmiş hali.

Azuma'ya göre [2] bir AG uygulaması aşağıdaki özelliklere sahip olmalıdır:

- Gerçek ve sanalı birleştirmeli.
- Gerçek zamanda etkileşimli olmalı.
- 3 boyutta da çalışabilmeli.

Bu tanım ile birlikte AG kavramı sadece HMD (Head Mounted Display, Kafaya Monte Ekran) ile sınırlı bırakılmamış üretilebilecek olan yeni mobil sistemlere de uygun olabileceği belirtilmiştir. Günümüzde AG teknolojisi mobil telefonlardan tabletlere ve akıllı gözlüklere kadar kullanılabilir ve uygulanabilir haldedir.

AG uygulamalarında gerçek görüntü üzerine bindirilecek sanal verinin türü de uygulamalarda çeşitliliği artırmaktadır. Görüntü üzerine bindirilecek veri bir arayüzün parçası, bir işaretleyici ya da bir metin ise 2B bir sanallık sağlanır ve derinliğe gerek duyulmaz. Zhihan ve arkadaşları [5], dokunma gerektirmeyen etkileşimli bir AG oyunu tasarlamıştır. Buna göre ekranda kameradan alınan görüntüdeki el mimikleri üzerinde işlem yapılmış ve istenilen durumlar yakalandığında ekran üzerinde yakalanan alanı içine alan bir dikdörtgen çizdirilmiştir (Şekil 2.2 (a)). Aynı şekilde Liu ve arkadaşları [6] da bir barkod okuma sistemi yapmış, barkod ekran üzerinde belirlendiği anda etrafını saracak şekilde bir kare çizilmiştir.



Şekil 2.2. (a) 2B AG uygulaması [6] ve (b) 3B AG uygulaması örnekleri [84].

2B uygulamalara göre daha ilgi çekici, popüler ve belki de daha zorlayıcı olan 3B AG uygulamaları da mevcuttur. Bu tür uygulamalarda genellikle 3B nesnelere görüntü üzerine bindirilir ve çoğu zamanda gerçek dünyada bulunan 3B nesnelere iletişim kurulabildiği gibi bindirilen sanal nesnelere de iletişim kurulabilir. Bostancı ve arkadaşları [84] çalışmalarında, kullanıcı izleme yöntemlerini test etmek için 3B hazine avı oyunu geliştirmişlerdir. Buna göre sistemde kullanıcı kafaya monte bir kamera ile belli bir alanda gezinir ve belli aralıklarla yerleştirilmiş olan sanal paraları toplayarak puan kazanır.

2.1.1. Artırılmış Gerçeklik ve Sanal Gerçeklik

AG kavramı genel olarak Sanal Gerçeklik (SG) kavramıyla karıştırılabilmektedir. SG teknolojisinde kullanıcı tamamen bilgisayarda üretilmiş bir ortam içindedir ve gerçek dünya ile iletişimi beklenmez. Bu ortam gerçek dünyanın sanal bir kopyası ya da tamamen hayal ürünü bir çevre de olabilir. Milgram ve Kishino [3] gerçeklikten sanallığa bir süreç tanımlamış (Şekil 2.3), AG ve SG'yi bu süreçte nerede olacağını göstermiş ve gerçeklik ve sanallığın bir arada bulunduğu sisteme Karma Gerçeklik (KG) adını vermiştir.



Şekil 2.3. Milgram ve Kishino'ya göre Sanallık Süreci [3].

2.1.2. Artırılmış Gerçeklik Uygulama Alanları

AG'nin kullanım alanları çok geniş ve kapsamlı olabilir. Azuma [2], 2000li yılların başında AG'in uygulama alanlarını oldukça detaylı olarak sıralamıştır. Azuma'ya göre AG; Tıp, Üretim ve Tamirat, Görselleştirme, Robot Yol Planlama, Eğlence ve Askeri Hava Araçlarında kullanılabilir ve uygulanabilir.

Papagiannakis ve arkadaşları [4], mobil AG sistemlerinin uygulama alanları olarak Sanal Karakterler, Kültürel Miras, Eğitici Oyunlar, Navigasyon ve Yol Bulma, Montaj ve İnşa, Bakım ve Denetleme olarak sıralamıştır.

Eğitim alanında AG kullanımına ilişkin Hsiao ve arkadaşları [7], Tayvan'da öğrencilerin öğrenirken alıştırma yapmasını sağlayan bir uygulama geliştirmiştir. Ecosystems AR Learning System (EARLS) adını verdikleri projeye göre öğrencilere ders, "okuyarak öğrenme" ya da "dinleyerek öğrenme" teknikleri ile değil de "yaparak öğrenme" tekniğiyle verilebilecektir. Webcam yardımıyla öğrencilerin canlı görüntüsü bilgisayar aktarılır, bilgisayarda sanal nesnelere görüntü de doğru yerlere yerleştirilir ve yansıtıcı yardımıyla oluşan görüntü duvara yansıtılır. Öğrencilerin alandaki gerçek nesnelere dokunması sonucu görüntüde tepkiler oluşur.

Dunleavy ve arkadaşları [8], çalışmalarında yine eğitim alanında kullanılacak bir AG uygulaması tasarlamıştır. Alien Contact! adı verilen uygulama ile öğrencilere matematik, sanat ve bilim kültürü eğitimi verilmesi amaçlanmıştır. Dell Axim X51 cep bilgisayarı üzerinde cihazın GPS verisini kullanarak çalışan uygulamada öğrencilerin gerçek dünyadaki konumları alınarak uygulamanın sanal dünyası üzerine uygun konuma yerleştirilir. Öğrenciler gerçek dünyada yürüdükçe uygulamanın sanal evreninde de hareket ederler ve etrafta çeşitli sanal nesnelere karşılaşırlar. Bu nesnelere yaklaşıncaya uygulamanın senaryosu gereği çeşitli video, metin yada ses dosyalar çalışır.

AG teknolojisi inşaat sektörü için de oldukça kullanışlıdır. Wang [9], çalışmasında; AG'nin, inşaat sahası planlama konusunda 3B bir perspektif sunduğu için, 2B çizimlerden çok daha tutarlı ve faydalı olacağı belirtmiş ve AG tabanlı bir inşaat alanı planlama programı prototipi tasarlamıştır.

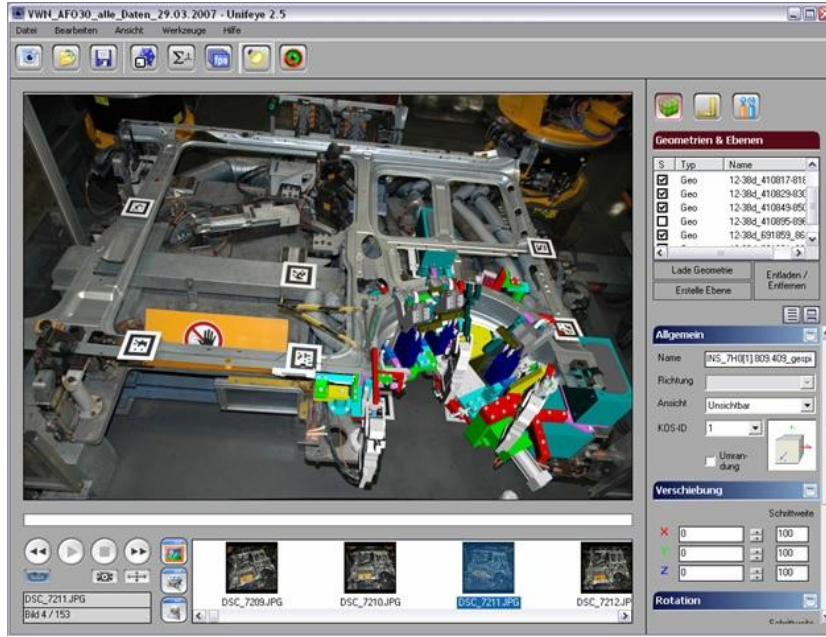
İnşaat alanında AG kullanımına örnek olabilecek bir diğer çalışma ise Yabuki ve arkadaşlarının [10] güzel manzaraları korumak adına bina ve inşaatlara uygulanan yükseklik kısıtlamaları için görünmez yükseklik değerlendirme sistemidir. Yeni yapılacak bir binanın herhangi bir tarihi bina, köprü, tapınak ya da doğa harikasının görünmesini engelleyip engellemeyeceğini ölçecek AG kullanan bir metot geliştirilmiştir.

Wang ve Dunston [14], KG tabanlı iki birlikte-çalışma tasarım uygulaması geliştirmiş ve uygulanabilirliği test etmişlerdir. İlki, yan yana bulunan bir grup insanın bir plan üzerinde ortak çalışması sağlayan KG uygulamasıdır. İkincisi ise birbirinden coğrafi olarak farklı yerlerde bulunan insanlar için bir KG tabanlı çevre uygulamasıdır.

Literatürdeki bir diğer tasarım uygulaması örneği ise Luh ve arkadaşlarının [15] AG kullanılarak çocuklar için ayakkabı tasarımı çalışması olarak görülmektedir. Çalışmada tasarlanan ayakkabılar Microsoft Kinect ile alınan ayak görüntüsü üzerine bindirilmekte ve AG ortamı oluşturulmaktadır.

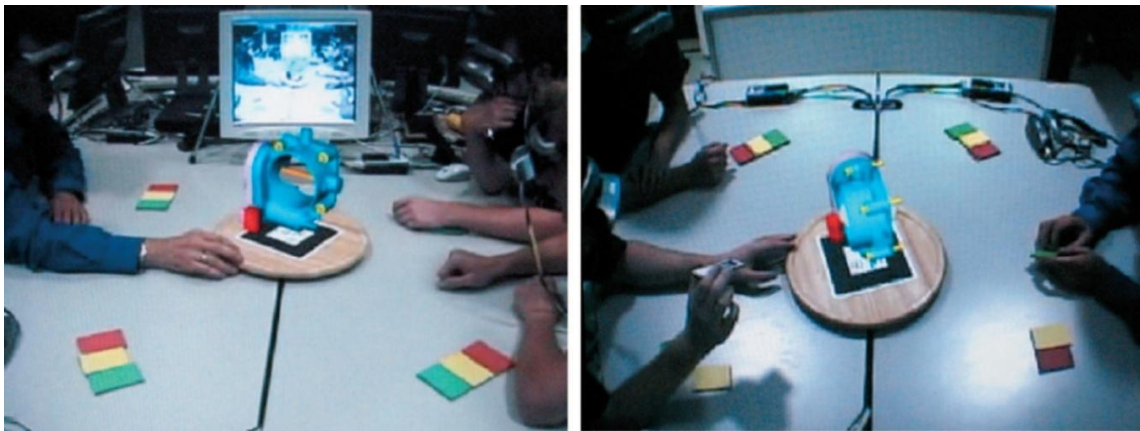
Üretim ve fabrikasyon alanında AG'nin ilk örneklerinden biri Boeing uçak fabrikasında uçak yapımı sırasında elektrik kablolarını kurmaya yardım edecek bir AG uygulamasıdır [11]. 1990 yılında yapımına başlanan bu projeye göre sistemin çalışması için üç bileşen gereklidir: Bir şeffaf kafaya monte ekran, bir kafa pozisyon/oryantasyon izleyici, bir giyilebilir bilgisayar. Pilot proje sonunda elektrik kablolarının AG kullanılarak monte edilebilir olduğu görülmüştür.

2003 yılından başlayarak Volkswagen Group Research ve metaio GmbH birlikte çalışmış ve AG destekli fabrika planlama uygulaması geliştirmişlerdir [13] (Şekil 2.4).



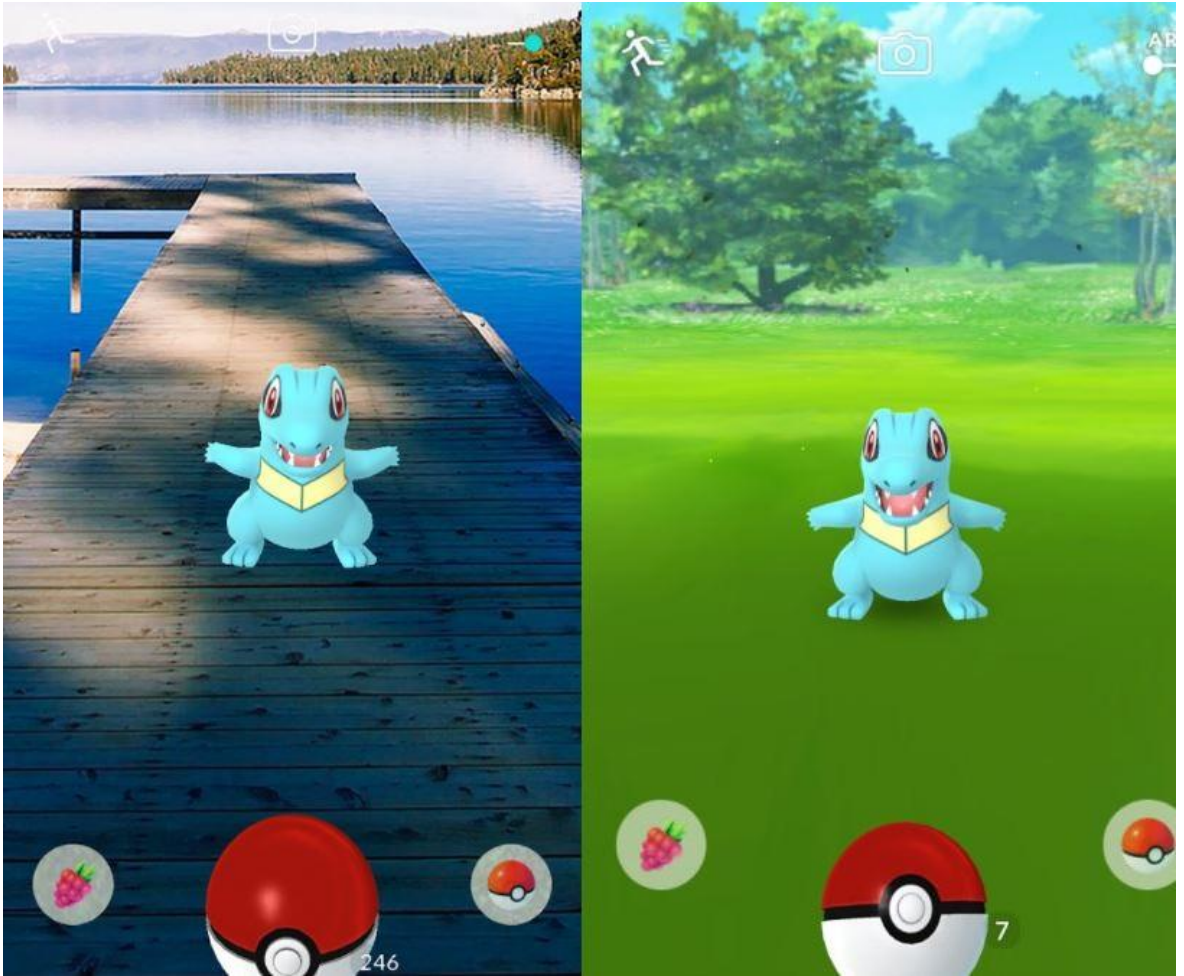
Şekil 2.4. VWRG ve Metaio tarafından geliştirilen uygulama [13]

Regenbrecht ve arkadaşlarının [12] makalelerinde Almanya'daki Daimler-Chrysler ekibi tarafında çeşitli alanlarda kullanmak için geliştirilen AG uygulamaları anlatılmaktadır. Buna göre ekip beş yıl boyunca Uzay İstasyonu Filtre Değişimi, Motor Tamiri, Tramvay Arıza Tespiti, Uçak Kabin Tasarımı, Birlikte Tasarım Kritiği, Kokpit Planı, Sigorta Yerleştirme, Seçme ve Kabloleme gibi başlıklar altında on AG uygulaması geliştirmiş ve test etmiştir. Tüm uygulamaların sonunda karşılaşılan sorunlar, tasarım problemleri ve gereksinimleri tartışılmıştır.



Şekil 2.5. Motor Tasarım Kritiği [12]

Bir başka ilgi çekici AG uygulaması ise 2016 yılında mobil aygıtlar için piyasaya çıkan ve bir anda tüm oyun dünyasında konuşulur hale gelen Pokemon Go'dur [16]. The Pokémon Company tarafında piyasaya sürülen oyun mobil cihazların GPS özelliğini kullanarak kullanıcıların sanal bir dünyada gezinebilmesini sağladı. Haritada rastgele yerlerde beliren pokemonları (oyun içindeki yaratıklar) yakalamak için onlara yaklaşıldığında oyun, sıradaki sahneye atlayarak kullanıcıyı, arka planda cihazın gerçek zamanlı kamera görüntüsü ve ön planda karşılaşılan pokemonun 3B animasyonu ile karşı karşıya bırakır. Oyun beta aşamasında olmasına rağmen haklarına sahip şirket olan Nintendo'nun hisselerinin bir anda %50 oranında artmasını sağlamıştır [17]. Oyun aynı zamanda hem Android hem de iOS tabanlı cihazlarda en hızlı çıkış yapan oyun rekorunu kırmıştır. En çok kar edenler listesinde birinci sıraya oturması App Store'da (iOS) bir günden az, Google Play'de 4 gün sürmüştür[18]. Pokemon Go'nun bu hızlı yükselişi bir anda gözleri AG teknolojisine çevirmiştir.



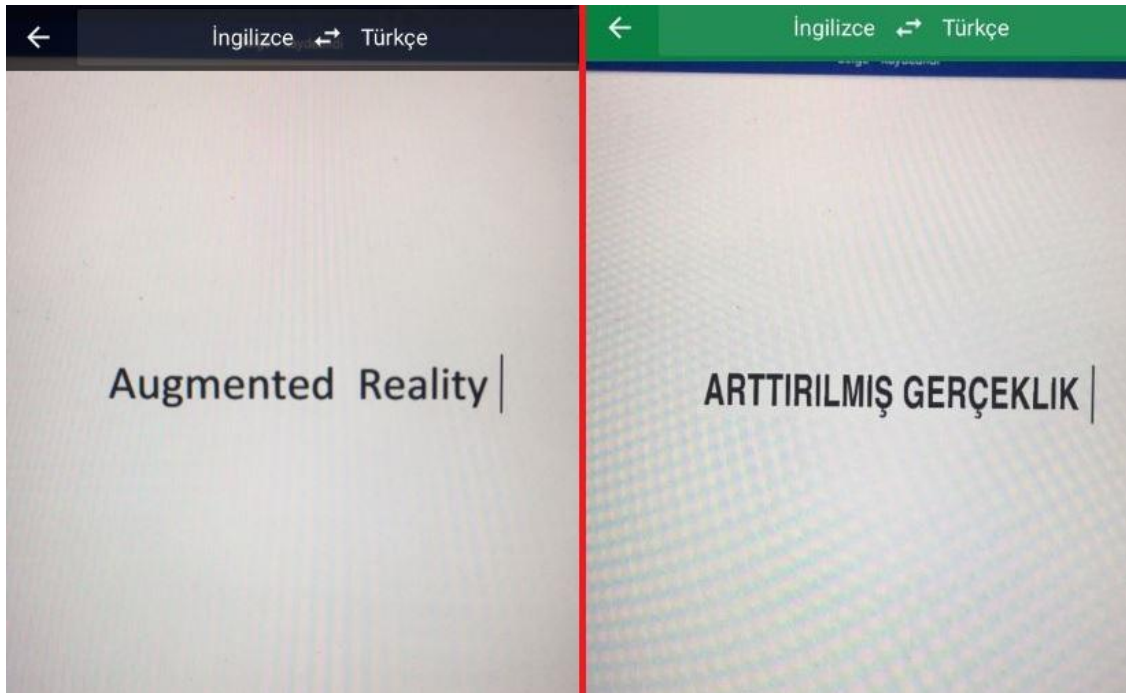
Şekil 2.6. Pokemon Go [19]

Tıp alanında da AG teknolojisinin önü oldukça açıktır. Moulton ve arkadaşları [20], faset sendromu hastalarına iğne tedavisi uygulanması için tasarlanan bir AG uygulaması tartışmışlardır. Perk Tutor adlı AG destekli iğne enjeksiyon eğitim sisteminin kullanıldığı çalışmada daha önce hiç iğne enjeksiyonu yapmamış tıp hazırlık öğrencileri kullanılmıştır.

AG çalışmaları genellikle görme duygusuna hitap eden metin, resim ve video veya nesnelere kullanılarak yapılırsa da duyma ya da dokunma duygusuna hitap eden çalışmalar da yapılmıştır. Wei ve arkadaşları [21], dokunma ve duyma duygularına yönelik bir AG uygulaması tasarlamıştır. Tarihsel motifleri temsil eden 3 farklı ritim ve dönemleri temsil eden 3 farklı genlik (ses için alçaklık veya yükseklik, dokunma için yoğunluk) hazırlanmış ve 30 kişiden bu motif ve dönemi ayırt etmesi istenmiş ve sonuçlar not edilmiştir.

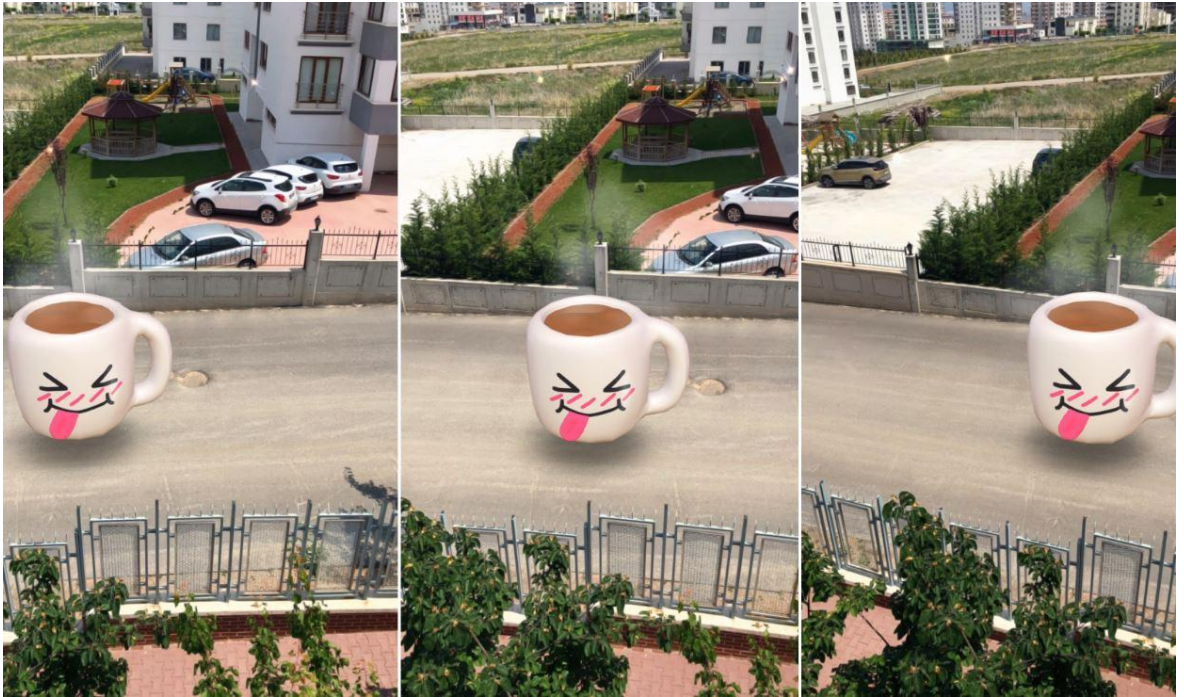
Tayvan'da öğrencilere kütüphane kültürünün öğretilmesini amaçlayan Wang ve arkadaşları [22] tasarladıkları etkileşimli AG oyunu ile 116 öğrencinin katıldığı deneyde, öğrencilerin öğrenme performansının arttığını gözlemlemişlerdir. Uygulama işaretçi tabanlı olarak ARToolkit çatısı kullanılarak geliştirilmiştir.

AG alanında yapılmış en iyi çalışmalardan birisi Google Translate [97] mobil uygulamasının AG tabanlı çevirme seçeneğidir. Desteklenen dillerden herhangi birinde yazılmış metinden bir kelimeye odaklanıldığında o kelimenin üzerine çevirilmiş hali bindirilir.



Şekil 2.7. Google Translate ile İngilizceden Türkçeye Çeviri

Eğlence alanına yönelik AG uygulamalarına en iyi örneklerden biri Evan Spiegel, Bobby Murphy ve Reggie Brown [23] tarafından geliştirilen Snapchat [24] adı verilen görüntülü mesajlaşma uygulamasının filtreleridir. Uygulamada fotoğraf yada videolar karşıdaki kişiye geçici bir süreliğine gönderilir. Süre sona erdiğinde gönderilen mesaj silinir ve bir daha görüntülenemez. Uygulamanın asıl ilgi çekici olan yanı ise gerçek zamanlı olarak mobil cihazın kamerasından alınan görüntüye AG uygulanmasıdır. Kullanıcının yüz hatları belirlendikten sonra yüzünün çevresine eklenen çeşitli animasyonlar veya gözler, kulaklar, ağız gibi yüz hatlarında yapılan deformasyonlar ile uygulama, mesajlaşma konusunda milyonlarca insanın tercihi haline gelmiş ve bir çok sosyal platform uygulamasının da AG teknolojisini kullanmaya başlamasına öncülük etmiştir. Bunlardan en önemlisi ve bir fotoğraf ve video paylaşım platformu olan Instagram'dır [25]. Instagram kendi sistemine Snapchat tarzı yüz filtreleri ve hikaye getirince popülerliği oldukça arttı ve bu insanların Snapchat'ten uzaklaşmasına neden oldu [26]. Bunun üzerine Snapchat 2017 nisan ayında 3D World Lenses duyurdu. 3B animasyonlu modellerin gerçek zamanlı olarak video görüntüsüne sabitlenebildiği bu sistemle birlikte, mobil uygulamalarda AG kullanımını bir üst seviyeye taşıdı [27].



Şekil 2.8. Snapchat uygulamasından alınmış 3 görüntü

Google Translate ile başlayan ve Snapchat ve Instagram ile devam eden mobil uygulamalarda AG teknolojisinin eklenmesi, AG'nin günlük hayatın bir parçası olmasını sağladı.

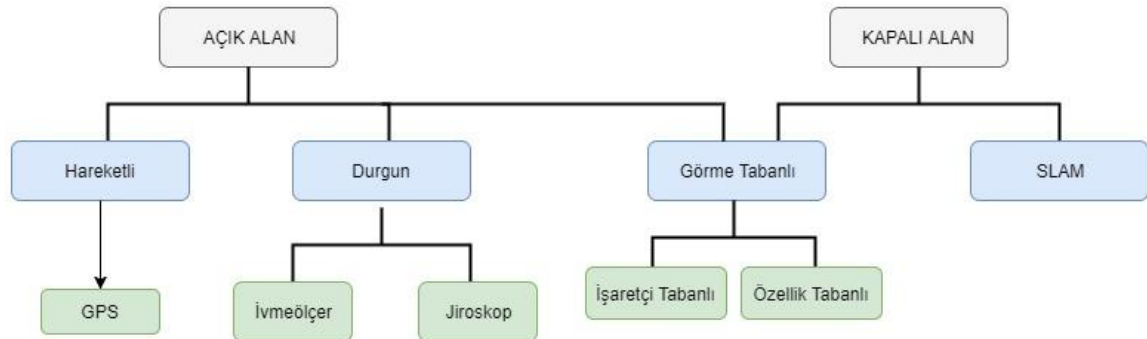
2.2. Kullanıcı İzleme Yöntemleri

Kullanıcı İzleme Yöntemleri denince akla kullanıcının o anda bulunduğu konum, yükseklik ve bakış açısının takibi gelmelidir. Kullanıcının doğru ve tutarlı şekilde izlenmesi AG uygulamaları için en temel konulardan biridir. İzleme işlemi, ekrana yerleştirilecek olan nesne ya da videonun doğru yerde gözükmesi adına yeterince hassas olmalıdır. Gerçek zamanlı olarak izlemede yaşanabilecek gecikmeler minimuma indirilmelidir.

"Kullanıcı İzleme" tabiri kullanıcının kendisinden ziyade daha çok kullandığı cihazı izlemek anlamına gelmektedir. Kullanıcı bir mobil telefon, tablet yada taşınabilir bilgisayar kullanıyorsa cihazların bakış açısı ve konumu önemlidir. Öte yandan kullanıcı eğer bir kafaya monte ekran kullanıyorsa bu durumda kullanıcının kafasını izlemek daha tutarlı sonuçlar verecektir.

Zhou ve arkadaşları [32], AG sistemleri için kullanıcı izleme tekniklerini üç ana başlık altında toplamıştır: Çevrede güçlü sinyal yayıcılar ve sensörlerin bulunduğu Sensör-Tabanlı, kameranın aldığı görüntüler üzerinde hesaplamalar yapılarak izleme işleminin gerçekleştirildiği Görme-Tabanlı ve bu iki sistemin avantajlı yanlarının kullanıldığı karma Karma teknikler.

Bostancı ve arkadaşları[28] ise kullanıcı izleme yöntemlerini; kapalı alanlar ve açık alanlar için yöntemler olmak üzere ikiye ayırmıştır. Bu çalışmada da sınıflandırma bu şekilde yapılacaktır.



Şekil 2.9 Kullanıcı İzleme Yöntemleri [28]

2.2.1. Kapalı Alan

Kapalı alanlarda yer genellikle sınırlı ve belli başlı nesnelere çok belirgin olduğundan kullanıcı hareketlerini izlemek daha kolaydır [30]. Kullanıcı izlemede ultrason yada manyetik sensörlerin kullanılabilir ancak bu sensörlerin GPS sensörlerine göre maliyeti göreceli olarak yüksektir. Bu durumda GPS sinyali kapalı alanlarda izleme işlemi için elverişli olduğu düşünülebilir. Ancak sinyalin uydulardan gelmesi ve kapalı alanın duvarların bu uyduların sinyalini büyük ölçüde engellemesi sonucunda kapalı alanlar için GPS kullanılması izleme için yeterli kaynak sağlamaz. Bu durumda kapalı alanlar için en uygun izleme seçeneği görme tabanlı izleme olarak öne çıkmaktadır [28].

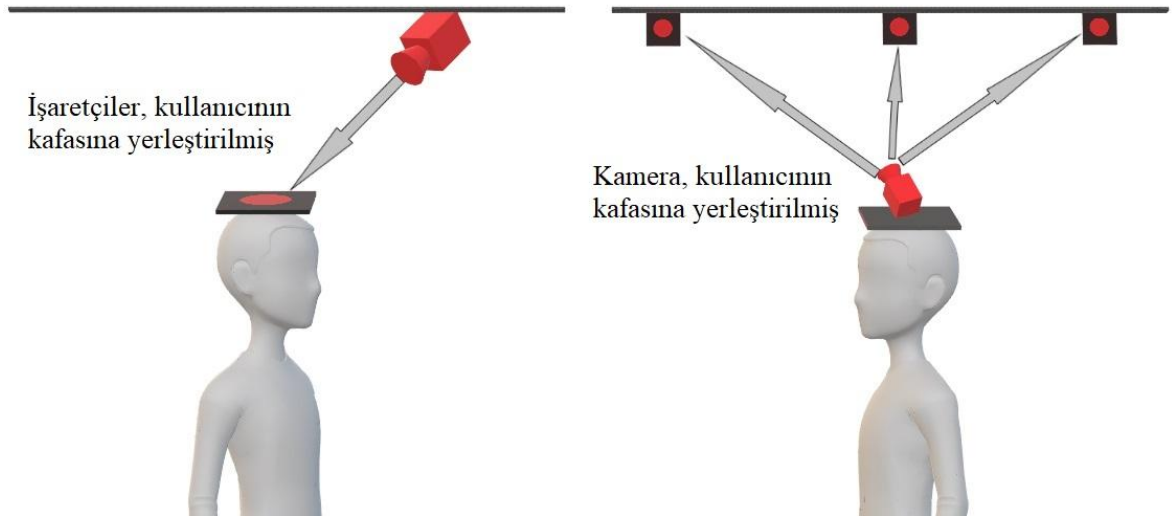
Livingston [34], manyetik alan oluşturan bir cihaz ve izlenen cihaza yerleştirilen bir alıcı yardımıyla bu manyetik alanı algılayan bir izleme sistemi kullanmıştır. Sistemde çevredeki metal ve elektromanyetik cihazların manyetik alanı etkilediğini ve dolaylı olarak izleme işleminin de etkilendiğini belirtmiştir.

Başka bir çalışmada ultrason-tabanlı iğne biyopsisi adlı tıbbi prosedürü AG teknolojisi ile kolaylaştırmayı amaçlayan Livingston [35], yine manyetik algılayıcıları kullanmıştır. Bulduğu çevrede oldukça fazla metal bulunması sebebiyle manyetik aygılayıcıların etkilendiğini görmüş ve sistemini görüntü tabanlı işaretçi izlemeyle birleştirmiştir.

RFID (radio-frequency identification - radyo frekanslı tanımlama) teknolojisi ile radyo frekanslı kullanılarak, önceden etiketlenmiş nesnelere tanımlanabilir. RFID teknolojisi barkod teknolojisine alternatif olarak gelmiş olsa da çevredeki nesnelere izleme konusunda elverişlidir. Zhang ve arkadaşları [37] çalışmalarında, bir AG montaj kılavuz sistemini geliştirmişlerdir. Çalışmada AG, kafaya monte ekran aracılığı ile gerçekleştirilmiş, bu sayede kullanıcının elleri montaj işlemi için boş bırakılmıştır. Montaj işlemi yapılırken ele takılan bir çip ile el takip edilmiş ve burada takip sistemi olarak RFID teknolojisi kullanılmıştır.

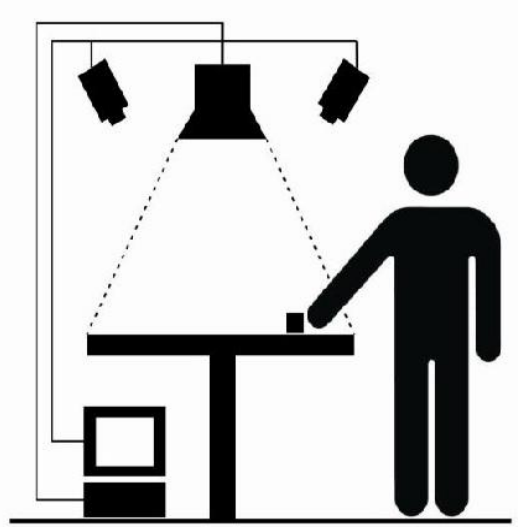
Ultrasonik takip sistemleri ise akustik sensörler ve ultrasonik ileticilerden oluşur. Sensörler genellikle çalışma alanında sabit olarak bulunurken kullanıcı bir ultrasonik sinyal yayıcı kullanır. Sesin sensöre ulaşma zamanını baz alarak kullanıcının uzaklığı ve pozisyonu hesaplanır. Sesin hızının çevredeki nem, sıcaklık, basınç gibi etkenlerle hızının değişmesi ve kullanıcının yerini tespit etmek için yayılan sinyallerin yeterince sıklıkla gelmemesi gibi nedenlerle bu sistem çok kullanışlı değildir [38].

Kapalı alanlarda kullanıcı izleme teknikleri genel olarak sensörlerin nereye yerleştirildiğine göre iki farklı şekilde uygulanabilir. Dıştan-içe (Outside-in) veya İçten-dışa (Inside-out) [36]. Dıştan-içe teknikte sensörler, çalışmanın yapıldığı alana sabitlenmiş şekildedir. Yani kullanıcının "dışındadır". İşaretçiler ise kullanıcıya takılmıştır. Sabit sensörler işaretçilerin takılmış olduğu hareketli kullanıcıları izlerler. İçten-dışa yöntemde ise yine adından anlaşılacağı üzere sensörler, kullanıcıya takılmış yani kullanıcının "içindedir". İşaretçiler ise çalışma alanına sabitlenmiş şekildedirler.



Şekil 2.10. Dıştan-içe (Sol) ve İçten-dışa (Sağ) yöntemler [36]

Mistry ve arkadaşları [39] çalışmalarında TaPuMa (Tangible Public Map - Hissedilebilir Ortak Harita) adını verdikleri gerçek zamanlı, çok kullanıcı, etkileşimli bir bilgi ve yer arama haritası prototipi tasarlanışlardır. TaPuMa sistemi tüm haritanın ve etkileşim sonunda oluşturulacak dinamik ve sanal içeriğin yansıtılacağı bir masa üzerinde çalışmaktadır. Yüzeydeki nesnelere tanıyıp ve takip etmeye yarayan kameralar, haritayı ve oluşturulacak sanal bilgiyi yansıtacak olan yansıtıcı (projektör) masanın üzerine yerleştirilmiş ve ikisi biden tüm mantıksal işlemleri yapacak olan bir bilgisayara bağlanmıştır. Bu sistemde kullanıcılar her gün yanlarında taşıdıkları bir nesneyi haritanın üzerine koyarlar, sistemdeki yazılım nesneyi tanıyıp ve takip etmeye başlar. Yazılım nesneyi yorumlayarak bilgi yada konum gönderir. Örneğin; bir öğrenci okul kimlik numarasını haritanın üzerine koyduğunda, sistem onun bir okul kartı olduğunu tanımlar, öğrencinin numarasını okur ve onun ders programı bilgisini öğrenciye döndürür. Prototip sistem bir havaalanı için geliştirilmiş ve dıştan-içe yöntem uygulanarak tasarlanmıştır (Şekil 2.11).



Şekil 2.11. TaPuMa (Tangible Public Map) Sistemi[39]

Dıştan-içe yöntemi ile geliştirilen bir başka uygulama ise Cooper ve arkadaşlarının [40] çalışmasında görülmektedir. Bu çalışmada yine masaüstü çalışan bir AG Çin daması oyunu tasarlanmıştır. Tavana yerleştirilen kameralar ile üzerine işaretçi yerleştirilmiş nesnelere takip edilir ve oyun ilerletilir. Çalışmada 4 farklı etkileşim türü test edilmiş ve sonuçlar gösterilmiştir.

Bostancı ve arkadaşları [42], Microsoft Kinect ile kullanıcı hareketlerini izleyerek kültürel miras alanlarına yönelik AG uygulaması tasarlamışlardır. Çalışmada Kinect'in derinlik sensörü yardımıyla kameranın duruşu hesaplanmış ve AG'in uygulanacağı nesnelere görüntü üzerinde tespiti yapılmıştır. Önceden 3B tasarımı yapılmış tarihi bir kemerin modeli, kamerada tespit edilen dikdörtgenel görüntülerin üzerine bindirilmiştir. Çalışmada ayrıca insan hareketleri de Kinect yardımıyla izlenmiştir. Yine tasarlanan tarihi asker kıyafet ve kılıç modeli kullanıcının görüntüsü üzerine bindirilmiştir.



Şekil 2.12. Bostancı ve arkadaşlarının [42] çalışması.

AG çalışmalarda içten-dışa yöntem diğer yöntemlere nazaran çok daha fazla tercih edilmektedir. Chen ve arkadaşları [41], çalışmalarında ameliyatın güvenliğini ve güvenilirliği artırmak için Şeffaf kafaya monte ekran ile AG tabanlı cerrahi yönlendirme sistemi geliştirmişlerdir. Sistem için bir adet yeni nesil bilgisayar, bir adet kafaya monte ekran, bir adet monitör ve bir adet takip cihazı kullanılarak çalışmaktadır. Ameliyattan önce hastanın tomografi sonuçları elde edilir ve buna göre damarlar ve sinirler de dahil olmak üzere ameliyat edilecek kısmın 3B modeli tasarlanır. 3B model üzerinde takip cihazının da yardımıyla, tüm aletlerin birlikte, gerçek zamanlı olarak çalışması ve sanal yüzey ile model yüzeyi eşleşmesi sağlanır. Daha sonra ameliyatta yapılacak işlem model üzerinde yapılır. En sonunda da hastanın üzerinde yapılacak ameliyat için render edilmiş plan kafaya monte ekrana aktarılır ve ekran hastanın pozisyonu ile kalibre edilerek ameliyat yapılır.

Tsai ve Hsu [43] yaptıkları çalışmalarında bluetooth destekli bir "Odadan Kaçış" oyunu tasarlamışlardır. Normalde bir grup insanın, bir odadaki ipuçlarını toplayarak oradan çıkmaya çalıştıkları bir oyun olan Odadan Kaçış oyununun tamamen sanal hali de mevcuttur. Bu çalışmada ise AG kavramına da uygun olarak gerçek ve sanal oyun birleştirilmiştir. Kullanıcıları ellerinde tabletler ile bir odada dolaşmakta ve yerleri çevrede bulunan bluetooth cihazları yardımı ile tablet tarafından hesaplanmaktadır. Kullanıcı belli alanlarda bulunan ipuçları ile çıkışa yada problemin çözümüne doğru yaklaşır. Burada ipuçları tablette beliren çeşitli sanal nesnelere dir. Kullanıcı çözüme ilerken odadaki belli resmin yada işaretçileri tablet aracılığı ile tarar, uygulama bu resmi tanır ve sıradaki ipucunun kullanıcıya sunar. Bu uygulama hem kapalı alanlar için farklı bir kullanıcı izleme sistemi sunmakta (bluetooth), hem de etrafta sabit olarak bulunan işaretçiler ve gezen izleme mekanizması (tablet) ile içten dışa yöntemine iyi bir örnek olmaktadır.

Juan ve arkadaşları [45], bir psikolojik rahatsızlık olan hamamböceği korkusunun tedavisinde kullanılmak üzere bir AG uygulaması geliştirmişlerdir. Çalışmada USB kamera ile alınan görüntü üzerine animasyonlu hamamböceği modelleri bindirilmiştir. Uygulama daha önce hamamböcekleri korkusu yaşayan bir hasta üzerinde test edilmiş ve hastanın korkusunu yenmeye başladığı gözlenmiştir. Bu çalışma da USB kamera kullanılması ve işaretçilerin serbest olması sebebiyle içten dışa yöntem kullanılarak geliştirildiği söylenebilir.

2.2.2. Açık Alan

Açık alanlarda kullanıcının; izleyebileceği yolun, bakabileceği yönün ve de çevredeki nesnelerin çeşidinin bir sınırı yoktur. Kapalı alanlara göre oldukça değişken olabilen ışık koşulları ve hatta kapalı alanlara herhangi bir etkisi olmayan hava koşulları açık alanlarda mevcuttur. Bu yüzden bir çok kapalı alan tekniği açık alanlarda, çok fazla değişkene maruz kaldığından, oldukça verimsiz çalışır ve istenilen sonucu vermeyebilirler.

Azuma ve arkadaşları [46] açık alanlar için AG uygulaması geliştirmenin kapalı alanlara göre çok daha zor olacağını belirtmişlerdir. Açık alanlarda sensörler, görüntüleme cihazları ve bunların depolayabileceği güçlerinin sınırlı olmasının yanında kullanıcı bunları taşımak da zorundadır. Diğer zorlayıcı neden ise kullanıcının çevre kontrolünün oldukça düşük olmasıdır. Kapalı alanlarda görüntüye sadece istenilen nesnelere yerleştirilebilir, en uygun işaretçiler alanda en çok verimi sağlayacak noktalara yerleştirilebilir ve ışık şartları kontrol edilebilir. Açık alanlarda tüm bunları kontrol etmeye akılcı bir yaklaşım olmaktan çok uzaktır. Açık alanları zorlayıcı kılan bir diğer neden ise çalışacak uygulamanın tüm şartlar altında çalışması zorunluluğudur. Örneğin; uygulama gece veya gündüz, karlı, yağmurlu, güneşli havalarda çalışabilmelidir.

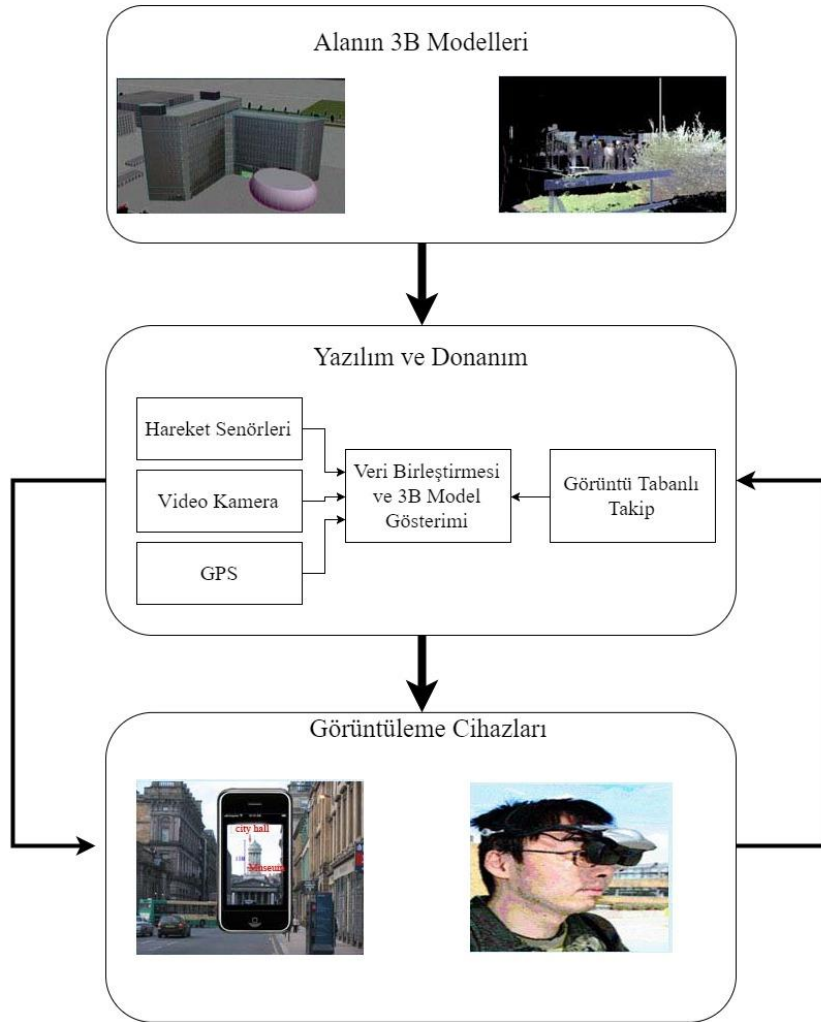
GPS tabanlı teknikler verinin yeterince sık aralıklarla geldiği açık alanlarda oldukça etkilidir. AG uygulamalarında GPS kullanımının ilk örnekleri Azuma ve arkadaşlarının [46] çalışmasında görülebilir.

Bazı durumlarda GPS sinyali yeterince sık olarak gelmeyebilir bu durumda kullanıcının pozisyonunun farklı yöntemleri kullanarak daha doğru şekilde belirlenmesi gerekir. Park ve Park [47], gökdelenler gibi uzun binaların GPS sinyalini engellediği alanlar için kullanıcının pozisyonunu ver oryantasyonunu belirlemek adına GPS sinyalinden alınan verilerin yanında bilgisayarlı görme teknikleri de kullanmışlardır. Haritadaki binaların köşe pozisyonları ve görüntüdeki binaların dikey kenarları kullanılarak geliştirilen yöntem yeterince bina kenarı görüntülenmediğinde yada GPS veri hatası oldukça fazla olduğunda tutarlı çalışmamıştır.

Alappanavar ve arkadaşları [81], mobil AG uygulamalarında cihazın bakış açısını hesaplamak için GPS kullanmışlardır. Geliştirilen uygulamada resimlerden oluşan bir veritabanı oluşturulmuş ve cihazın kamerasından alınan görüntü ile bu veritabanındaki resimler karşılaştırılmıştır. En yakın bulunan resimle kamera görüntüsü eşleştirilerek

cihazın yönelimi hesaplanmıştır. 3B nesnelere görüntünün üzerine bindirmek için cihazın konumunu bulma işlemi ise GPS sensörleri ile yapılmıştır.

Li ve Fan [78] çalışmalarında, 3B şehir modelleri ve 3B oyun motoru kullanarak yapılan model tabanlı açık alan AG izleme çalışmalarına katkı yapmayı amaçlamışlardır. Bu çalışmada kameranın pozisyonu ve bakış açısı, canlı kamera görüntüsü ve 3B şehir modelinin karşılaştırılması ile hesaplanmaktadır. Kullandıkları yöntemde öncelikle alanın 3B modelinin özellik noktaları çıkartılır. Aynı anda kameradan alınan görüntülerin de 2B özellik noktaları çıkartılır. Bu noktalara göre sanal nesnenin noktaları ve gerçek video görüntüsün noktaları hizalanır. Noktalar karşılaştırılarak kameranın bakış açısı hesaplanır. Son olarak da sanal nesnelere kameranın pozisyonuna ve pozuna göre görüntüye yerleştirilir.



Şekil 2.13. Model tabanlı açık alan AG mimarisi [78].

Forsman ve arkadaşları [79], daha önceden bilinmeyen açık alanlar için görüntü tabanlı bir izleme yöntemi olan panoramik takip algoritmasını genişletmişlerdir. Panorama takipçisi Unity 3D oyun motoruna entegre edilmiş ve GPS sistemi oluşturularak genişletilmiştir. Bu sistem daha sonra test edildiğinde açık alanlar için takip sisteminin tutarlılığı artırdığı görülmüştür.

Langlotz ve arkadaşları [80] çalışmalarında, önceden hazırlık yapılmamış alanlar için mobil AG içerik oluşturma sistemi tasarlanmışlardır. Bu sistemde hem açık alanlar hem de kapalı alanlar için takip sistemleri tasarlanmıştır. Açık alanlar için tasarlanan takip sisteminde sabitlenmiş bir mobil cihaz üzerinde panorama tabanlı görüntü takibinin yanında, ivmeölçer ve pusula gibi algılayıcı birlikte kullanmışlardır.

2.3. Görüntüleme Yöntemleri

Bu bölümde AG için görüntüleme yöntemleri incelenecektir. AG için görüntüleme yöntemleri genel olarak 3 bölüme ayrılabilir. Bu bölümler Video-Görme , Optik-Görme ve Yansıtılmalı-Görme olarak sıralanabilir [29].

2.3.1. Video-Görme

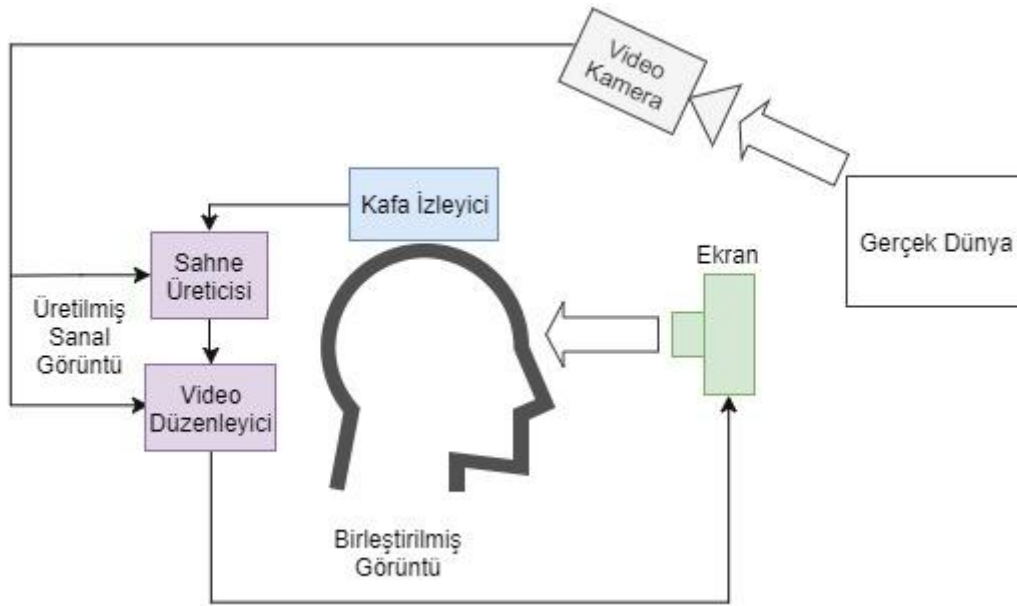
Video-Görme (Video See-Through) Yöntemi AG sistemlerinde en çok kullanılan görüntüleme yöntemidir. Video-görme yönteminde kullanıcı direk olarak çevreyi görmektense, çevrenin kamera görüntüsünü görür. Bu yöntem en ucuz olmasa da gerçekleştirmesi en kolay olan yöntemdir [29]. Ayrıca tüm görüntünün dijital halde olması sanal nesnelere video üzerine bindirme işlemini kolaylaştırır. Dezavantajlar, video görüntüsünün doğal olarak gerçek görüntüden her zaman daha az çözünürlüğe sahip olması ve göz ile görme cihazı arasındaki uzaklıktan dolayı kullanıcının hissedeceği uyumsuzluk olarak sıralanabilir.

Video-görme yönteminde genel olarak kullanılan araçlar; kafaya monte ekran (head mounted display, HMD), akıllı telefon ve tablet bilgisayar olabilir. Hatta bazı durumlarda taşınabilir bilgisayarlarda bir tür video-görme aracı olarak kullanılabilir.

Video-görme yöntemiyle yapılan AG uygulamalarına örnek olarak Hsieh ve Lee'nin [83] çalışması verilebilir. Bu çalışmada tıp alanında kullanılacak işaretçisiz bir AG

uygulaması tasarlanmıştır. Bu uygulamada hedef yüzey verilerini oluşturmak için stereo kamera kullanılmıştır.

Video-görme yöntemine farklı bir örnek de Bostancı ve arkadaşlarının [84] açık alanlarda kullanıcı takibi yöntemlerini test etmek adına geliştirdiği AG uygulamasında kullandığı sistem gösterilebilir. Kasıtlı olarak en basit teknoloji kullanılarak yapılmış bu sistemde bir bisikletçi kaskına bağlı web kamera ve görüntünün aktarılıp AG uygulandığı taşınabilir cihazdan oluşmaktadır. Bu çalışmada tüm AG işlemi video üzerinde yapıldığı için bu başlık altında değerlendirilmiştir.

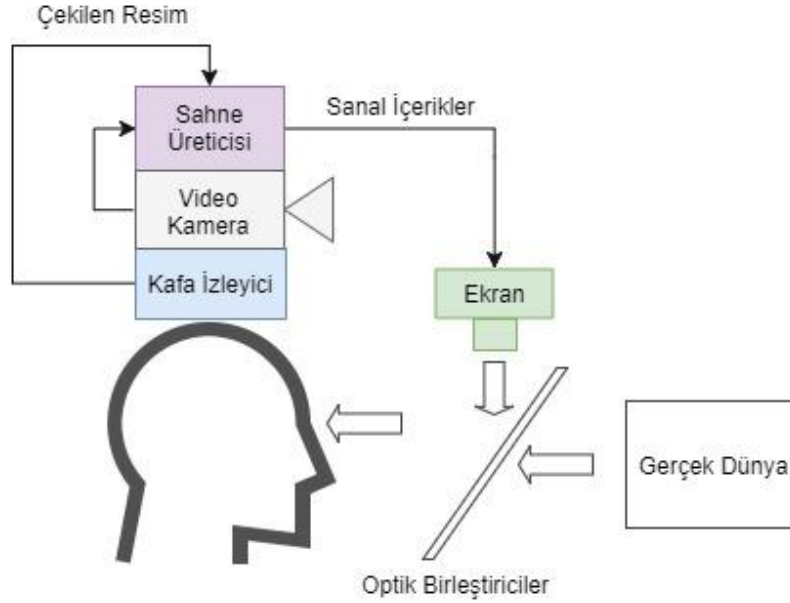


Şekil 2.14. Video-Görme Kafaya Monte Ekran mimarisi [82].

2.3.2. Optik-Görme

Optik-Görme sisteminde kullanıcı direk olarak gerçek dünyayı görürken sanal bilgiler ve nesnelere optik bir ekran üzerine bindirilir. Bu yöntemin avantajı, gerçek dünya görüntüsünün çözünürlüğü sabit kalır ve kullanıcı herhangi bir oryantasyon sorunu yaşamaz. Bu yöntem ayrıca sistem güç kaybına uğradığında çevre hala görülebildiği için daha güvenlidir [29].

Optik görme yönteminde kullanılan araçlar genellikle akıllı gözlüklerdir. Bunlara örnek olarak Epson Moverio [85] ve Google Glass [86] verilebilir.



Şekil 2.15. Optik-görme kafaya monte ekran sistemi [82].

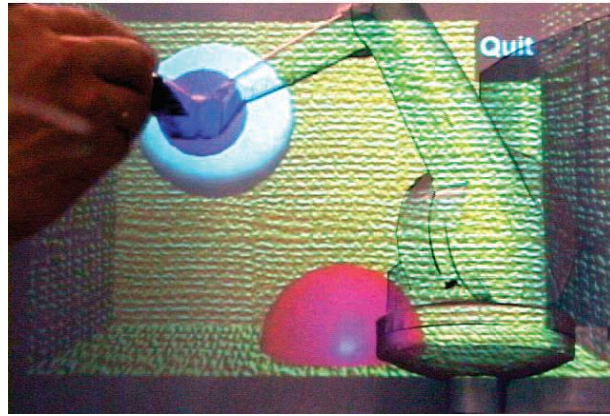
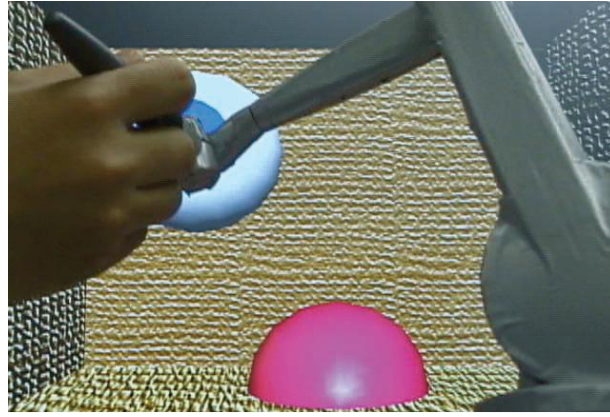
Optik-görme yöntemine örnek olarak Kaufmann [44] çalışması verilebilir. Bu çalışmada matematik ve geometri eğitimi için çok kullanıcılı bir AG uygulaması tasarlanmıştır. Uygulamada kullanıcılar optik-görme kafaya monte ekran kullanarak ortak bir alanda çalışabilirler. Nokta, doğru, düzlem, küp, küre, silindir ve koni gibi geometrik şekilleri uygulamada mevcuttur. Kullanıcılar gerçek zamanlı olarak bu şekilleri değiştirebilir yada büyütüp küçültebilir.

2.3.3. Yansıtma-Görme

Yansıtma-görme yaklaşımında sanal bilgi direk olarak AG uygulanacak nesnenin üzerine bindirilir [88]. Bu nesne, düz bir duvar yada yüzey şekilleri karmaşık bir nesne olabilir [87]. Bu yöntemde yansıtıcı ile yapılan sistemlerde herhangi bir gözlüğe ihtiyaç yoktur ancak sistemle iletişim dış aygıtlarla sağlanabilir. Bazı durumlarda kafaya monte yansıtıcılar da kullanılabilir (Şekil 2.16). Bu durumda hedef nesnelere ışığı yansıtan malzeme ile kaplanmalıdır.



Şekil 2.16. Kafaya Monte Yansıtıcı[88].



Şekil 2. 17.Kafaya monte yansıtıcı aracılığı ile yansıtmalı görme [88].

2.4. Kültürel Miras Alanları İçin Artırılmış Gerçeklik

Kültürel miras alanları için AG, tarihi alanların ve yapıların daha iyi anlaşılması geleceğe aktarılması adına oldukça ilgi çekici bir konudur. Müzelerde yada direk olarak tarihi alanlarda yapılan AG uygulamaları bu alanlara olan ilgiyi artırır. Bu da toplumların tarihsel bilgisinin ve bilincinin artmasına yardımcı olur.

Kültürel miras AG uygulamaları Azuma [88] ve Papagiannakis [4] tarafından AG'nin önemli uygulama alanları arasında gösterilmiştir.

AG, harabe haline gelmiş antik yapıların 3B modellerinin alan üzerinde gösterilmesini sağlayarak kültürel mirası daha eğlenceli bir hale getirir [89]. Antik uygarlıklar tarafından inşa edilmiş tarihi yapılar çoğu zaman savaşlar, depremler ve diğer doğal afetler sebebiyle yıkılmış ve harabe haline gelmiştir. Bu yapıların ait oldukları tarihi dönemdeki hallerine geri döndürmek için fiziksel olarak yeniden inşa çalışmaları yapılmalıdır. Ancak bu oldukça zahmetli bir işlemdir. Bunun yerine bilgisayar ortamında tasarlanmış sanal modellerin tarihi yapıların üzerine bindirilmesi daha kolay bir işlemdir.

Stricker ve arkadaşları [90], açık alanlar için referans fotoğrafları yardımıyla konumunun tespit edildiği bir AG uygulaması gerçekleştirmişlerdir. Bu uygulamada bilgisayarda üretilen tarihi bina 3B modeli referans fotoğraflar yardımı ile doğru konuma bindirilmiştir.

Turizm deneyimini daha eğlenceli hale getirmek için mobil bir AG oyunu tasarlayan Hu ve Tsai [91], tarihi anıtları kullanarak turistleri hem eğlendirmeyi hem de onlara anıtlarla ilgili bilgiler vermeyi amaçlamışlardır. Buna göre tarihi bir şehir olan Tainan'da kullanıcılar mobil aygıtlarıyla hazine avı şeklinde ilerleyerek tarihi anıtları keşfeder ve bilgi edinirler.

Vlahakis ve arkadaşları [92], ARCHEOGUIDE (Augmented Reality-based Cultural Heritage On-site GUIDE, Artırılmış Gerçeklik Tabanlı Alanda Kültürel Miras Rehberi) projesi adını verdikleri çalışmalarında kültürel alanları ziyaret eden insanlar için bir kişisel elektronik rehber hazırlamışlardır. Sistem kullanıcının pozisyonuna ve duruşuna göre 3B sanal nesnelere tarihi alan üzerine yerleştirir. Ayrıca sistem çevrimiçi olarak alanın bilgilerini alır ve ziyaretçiye sunar.

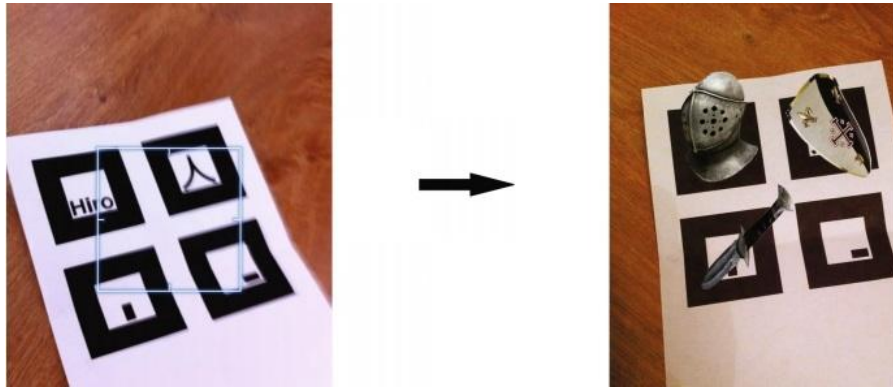
Bostanci ve arkadaşları [42] çalışmalarında, Microsoft Kinect yardımıyla görüntüdeki insanları tespit ederek üzerlerine Roma dönemine uygun tasarlanmış sanal giysiler giydirmişlerdir.

Papagiannakis ve arkadaşları [93], tarihi Pompeii şehrinde bulunan antik resimlerin AG ile sanal olarak hikaye anlatımı şeklinde canlandırılması üzerinde çalışmışlardır. Buna göre tarihi alan üzerinde antik resimlerden yararlanılarak oluşturulmuş insan modelleri gerçek zamanlı olarak bindirilmektedir. Ayrıca bu modeller gerçek zamanlı konuşma, mimik ve hareket animasyonlarına sahiptirler.

Haugstvedt ve Krogstie [94] ise çalışmalarında tarihi alanların fotoğraflarını ve bilgilerini kullanarak bir konum tabanlı mobil AG uygulaması tasarlamışlar ve bu tarz uygulamaların toplumda kabul görüp görmeyeceğini tartışmışlardır.

Müzeler için bir sanal ve artırılmış gerçeklik sistemi tasarlayan Wojciechowski ve arkadaşları [95], müzelerde sanal sergiler yapılmasını amaçlamışlardır. Bu doğrultuda tamamen sanal bir ortam oluşturarak SG sistemini, işaretçi tabanlı bir izleme sistemi oluşturarak da AG sistemini tasarlanmıştır.

Angelopoulou ve arkadaşları [96] çalışmalarında, müzeler ve arkeolojik alanlarda kullanılmak üzere bir mobil AG yaklaşımından bahsetmişlerdir. Bu yaklaşım IOS tabanlı mobil telefonlar için eğitici bir oyun geliştirilerek anlatılmıştır. Sutton Hou arkeolojik alanında 7. yüzyılda bir Anglosakson kralına ait olduğu düşünülen gemi hazineleri ile karaya oturmuştur. Ancak bugün geminin yerinde aslı değil yeniden inşa edilmiş hali mevcuttur. Geminin aslı ve birlikte getirdiği hazineler British Museum'da saklıdır. Bu sebeple Sutton Hou'ya gelen ziyaretçiler sınırlı sayıda kalıntı ile karşılaşır. Bu sebeple bu çalışmada anlatılan oyun bu alan hakkında ziyaretçilere bilgi vererek ve alanda bulunmayan hazineleri AG ile gerçek dünya görüntüsü üzerine bindirerek alana olan ilgiyi artırmayı amaçlamaktadır.



Şekil 2.18. Angelopoulou ve arkadaşlarının çalışmalarından işaretçi tabanlı AG örneği [96].

3. KULLANICI TAKİBİ

AG uygulamalarında kullanıcının bulunduğu pozisyonun sürekli olarak takibi ana problemlerin başında gelir. Kullanıcı hareket halindeyse, bu hareket hızına bağlı olarak takip sistemi hızlı çalışmalı ki görüntü üzerine bindirilecek sanal veriler doğru alanlara yerleştirilebilsin.

Bölüm 2.2'de kullanıcı takip yöntemleri kapalı ve açık alanlar için kullanılan yöntemler olarak iki sınıfta incelenmişti. Bu çalışmada geliştirilen uygulama da bir kültürel miras alanında çalışması gerektiğinden açık alanlar için uygun olan yöntemlerden yararlanılmıştır.

Android işletim sistemine sahip mobil cihazlar kullanımı 2010ların başından bu yana oldukça artış gösterdi. Bu cihazlar zamanla telefon özelliğinin yanında kamera, radyo, kablosuz internet bağlantısı (wi-fi), GPS ve jiroskop gibi bir çok özelliğe de varsayılan olarak sahip oldu. Android cihazların konum (GPS) ve oryantasyon (jiroskop) özelliklerine sahip olması, onları konum bazlı açık alan AG uygulamaları için oldukça avantajlı hale getirmiştir.

3.1. Kullanılan Sensörler

Kullanıcının dünya üzerinde konumunun ve bakış açısının belirlenmesi konum tabanlı AG uygulamalarında önemli problemlerdendir. Günümüzde Android işletim sistemli ortalama bir mobil aygıtla görüntü kaydetmek veya kullanıcının konumunu belirlemek mümkündür. Ayrıca bir çok cihaz kullanıcının yönünün belirlenebilmesini sağlayan jiroskop özelliğine de sahiptir. Sonuç olarak günümüz Android işletim sistemli mobil aygıtları konum tabanlı AG uygulamalarının minimum gereksinimlerini karşılamaktadır.

3.1.1. Küresel Konumlama Sistemi

Küresel Konumlama Sistemi (Global Positioning System, GPS), Amerika Birleşik Devletleri'nin sahibi olduğu ancak son kullanıcıların da yararlanabildiği uzay ve uydu tabanlı konumlandırma, yönlendirme ve zamanlama hizmetlerini sağlayan bir radyo-navigasyon sistemidir[48]. Sistem genel olarak 3 kısma ayrılır : Uzay Bölümü, Kontrol Bölümü ve Kullanıcı Bölümü. Uzay ve Kontrol bölümlerinin geliştirilmesi ve sürdürülmesi Birleşik Devletler Hava Kuvvetleri tarafından yapılır. Kullanıcı bölümü ise GPS alıcılarından oluşur. Uydulardan alınan veriler ile kullanıcının 3B konumunu hesaplar.

GPS ařağıdaki adımları gerekleřtirerek alıřır [48]:

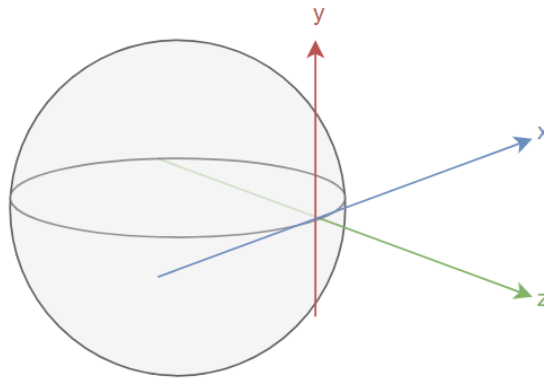
1. GPS uyduları; kendi konumlarını, durumlarını ve zamanlarını radyo sinyalleri yayınlarlar.
2. Radyo sinyalleri uzayda neredeyse ışık hızında hareket ederler.
3. Bir GPS cihazı gönderilen radyo sinyalini algılar, tam olarak vardığı zamanı kaydeder ve bu bilgileri yakındaki diğerk uydulardan olan uzaklığını hesaplamak için kullanır.
4. Cihaz yakındaki en az 4 uyduya olan uzaklığı hesapladıktan sonra Dünya üzerindeki 3B konumunu bu uzaklıklar yoluyla bulur.

Kullanıcının Dünya üzerindeki 3B konumundan kasıt enlem (latitude), boylam (longitude), yükseklik (altitude) olarak sıralanabilir. Günümüzde bir mobil telefonlardan, arabalara pek çok araç üzerinde GPS alıcılar mevcuttur.

3.1.2. Jiroskop

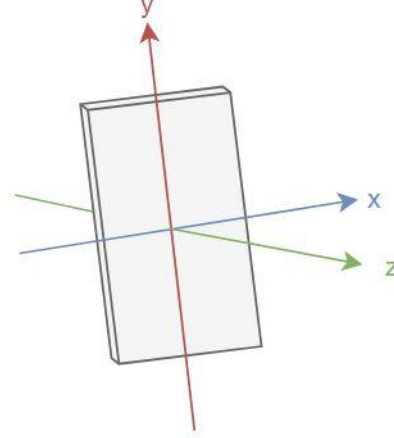
Jiroskop, cihazın yönlendirmesine karar vermek için Dünya'nın yerçekimini kullanan bir cihazdır. Jiroskop ile kullanılan cihazın yönlenimi ve bakış açısı bulunabilir. Günümüzde mobil cihazlarda, 3-eksenli MEMS-tabanlı (mikroelektromekanik sistem) jiroskoplar kullanılmaktadır.

Android tabanlı cihazlarda bulunan jiroskoplar genel olarak titreşim tabanlıdır[49]. Bir çift titreşen kol yardımıyla cihazının dönüşünü hesaplar. Jiroskop için Android cihazlarda kullanılan bir ok koordinat sistemi mevcuttur. Bunlardan ilki Dünya Koordinat Sistemi olarak adlandırılır. Bu sistem Android cihazlarda Doęu-Kuzey-Yukarı (East-North-Up) olarak alıřır (řekil 3.1).



řekil 3.1. Dünya Koordinat Sistemi

Bir diğ er koordinat sistemi ise Yerel Koordinat Sistemi'dir. Bir cihazın varsayılan pozisyonunda tutulduğ unda Z eksenini cihazdan çıkacak şekilde, Y eksenini dikey olarak yukarıyı iş aret edecek şekilde ve X eksenini yatay olarak ve sağı iş aret edecek şekildedir. Buradaki en önemli nokta cihazın yönelimi değı ştirilse bile koordinatlar yer değı ştirmeyecektir [49].



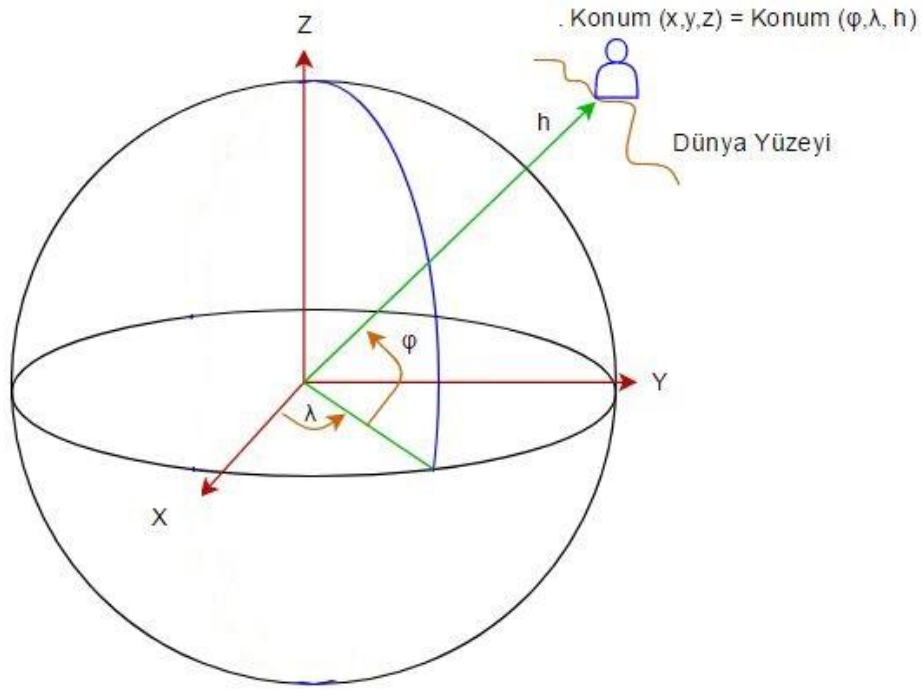
Ş ekil 3.2. Yerel Koordinat Sistemi

Geliştirilen AG uygulamasında kullanıcının dünya üzerindeki konumu ve kullandığı cihazın baktığı yönünün tespiti için bu iki sensörden gelen veriler kullanılmıştır.

3.2. Koordinat Sistemi Dönüşümü

Bu bölümün başında da belirtildiği üzere GPS verileri alıcılara enlem, boylam ve yükseklik verilerini iletir. GPS sistemi ile kullanıcının dünya üzerindeki konumunu uygun şartlar altında oldukça tutarlı olarak belirlemek mümkündür. Ancak Dünya'nın şekli nedeniyle iki nesne arasındaki asıl uzaklığı bulmak için GPS koordinatları yeterli değildir. Bu hesabı doğruya en yakın şekilde yapabilmek için; enlem, boylam ve yükseklik olarak gelen verilerin ECEF (Earth-Centered, Earth-Fixed) kartezyen koordinat formuna çevrilmesi gerekir.

ECEF, kullanıcının dünya üzerindeki konumunu göstermek için 3B kartezyen XYZ koordinat sistemini kullanır[50]. Burada (0,0,0) konumu Dünya'nın kütle merkezine denk gelir. Z-ekseni Kuzey Kutbu üzerine denk gelirken X-ekseni ana meridyen ve ekvatorun birleştiği noktadan geçer. Y-ekseni ise sağ el kuralına göre iş aret parmağı Z-eksenini, avuç içi X-eksenini gösterecek şekilde olduğ unda başparmağın gösterdiği noktadadır. Y-Eksenini Dünya üzerinde yaklaşık Hint Okyanusu'na denk gelir.



Şekil 3.3. GPS verisinin ECEF formatına dönüşümü [51]

Sonuç olarak AG sistemi için GPS verisi enlem (φ), boylam (λ) ve yükseklik (h) formatından 3B kartezyen x, y, z formatına aşağıdaki gibi çevrilebilir:

$$\begin{aligned} x &= (N + h) \cos(\varphi) \cos(\lambda) \\ y &= (N + h) \cos(\varphi) \sin(\lambda) \\ z &= ((1 - e^2)N + h) \sin(\varphi) \end{aligned} \quad (3.1)$$

Ayrıca burada

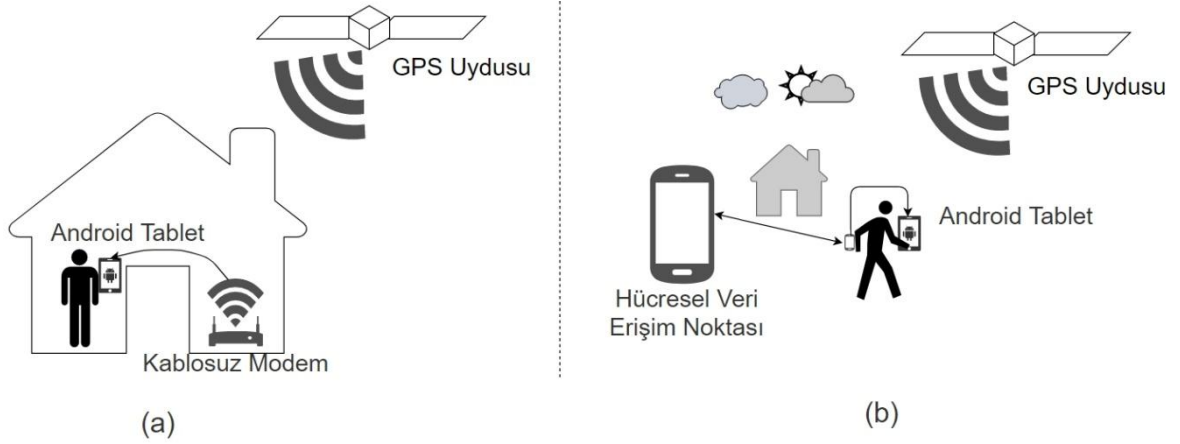
$$N = \frac{a}{\sqrt{1.0 - e^2 \sin^2(\varphi)}} \quad (3.2)$$

olarak yazılabilir. Burada a Dünya'nın ekvatordaki yarı çapı [52] (6,378,137 m), e^2 ise Dünya'nın dış merkezliliğidir [51] ($6.69437999 \times 10^{-3}$).

Kartezyen koordinat formatına çevrilen konum bilgisi artık kullanıcının sanal nesneyi kamera görüntüsüyle kontrol etmesi için uygundur. Geliştirilen uygulamada Unity 3D oyun motoru kullanıldığından ve oyun motoru nesnelere yerleştirmek için 3B bir sanal ortam sağladığından, kullanıcının konumu kartezyen koordinatlara çevrildiğinde, nesnelere kullanıcının doğru uzaklığı konumlandırılması mümkün olacaktır.

3.3. Konum Testleri

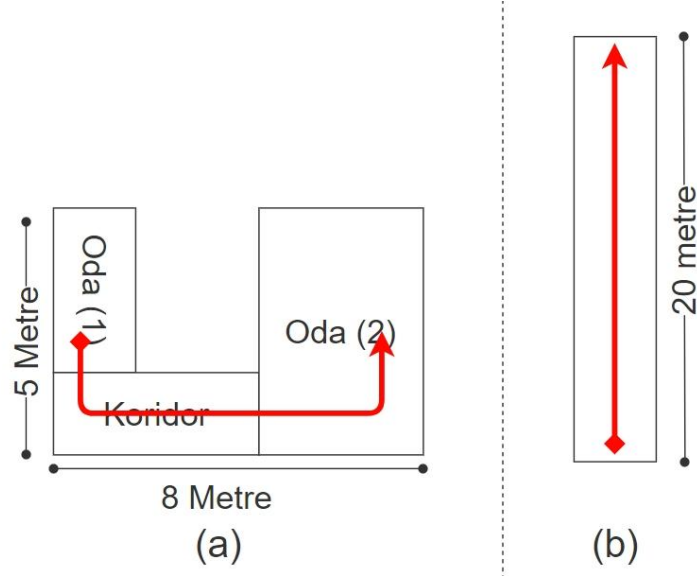
Konum tabanlı sistemlerde GPS verisinin gelme sıklığı ve tutarlılığı oldukça önem arz etmektedir. Bu bağlamda kullanılacak cihazın da önemi büyüktür. Uygulamanın mobil cihazlar üzerinde geliştirildiği düşünüldüğünde bu cihazların GPS verisinin önceden test edilmesinde fayda vardır. Projede kullanılması düşünülen Samsung Galaxy Tab A (SM-P550 - Android 6.0) ve Samsung Galaxy Tab S2 Android işletim sistemli tabletler için bir yazılım geliştirildi. Yazılım belli aralıklarla cihazın konum bilgilerini (enlem, boylam, yükseklik) GPS yardımıyla güncelleyecek ekrana verileri iletecek şekilde tasarlandı. . Kapalı alanda GPS verisini test etmek için kablosuz modem aracılığı ile tabletin internet bağlantısı sağlandı. Açık alanda ise kablosuz internete erişilemeyeceği için, hücresel veri erişimine sahip bir mobil aygıt yardımıyla erişim noktası oluşturuldu. Tabletin internet erişim bu cihaza bağlanılarak halledildi.



Şekil 3.4. (a) Kapalı alanda kablosuz internet bağlantısı kullanıldı. (b) Açık alanda bir mobil cihaz hücresel veri noktası olarak internete bağlanmak için kullanıldı.

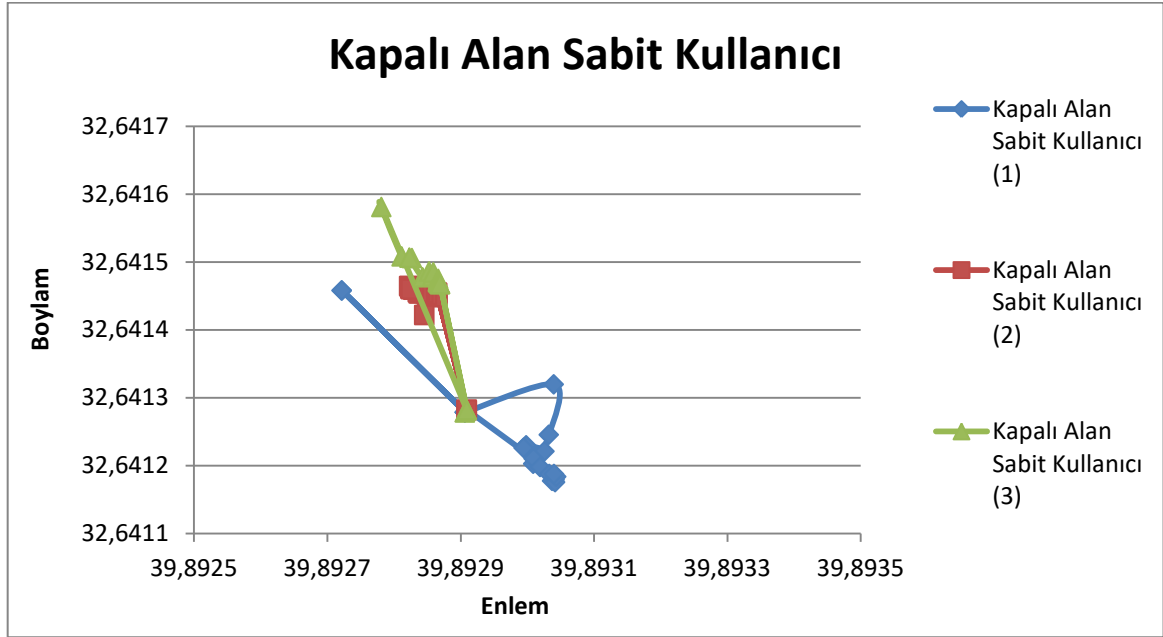
Enlem	Boylam
39.8929066	32.6412785
Güncelleme Sayısı : 4	
Enlem	Boylam
1) 39.8929094	1) 32.6412819
2) 39.8929094	2) 32.6412819
3) 39.8929094	3) 32.6412819
4) 39.8929066	4) 32.6412785

Şekil 3.5. Geliştirilen uygulamanın arayüzü

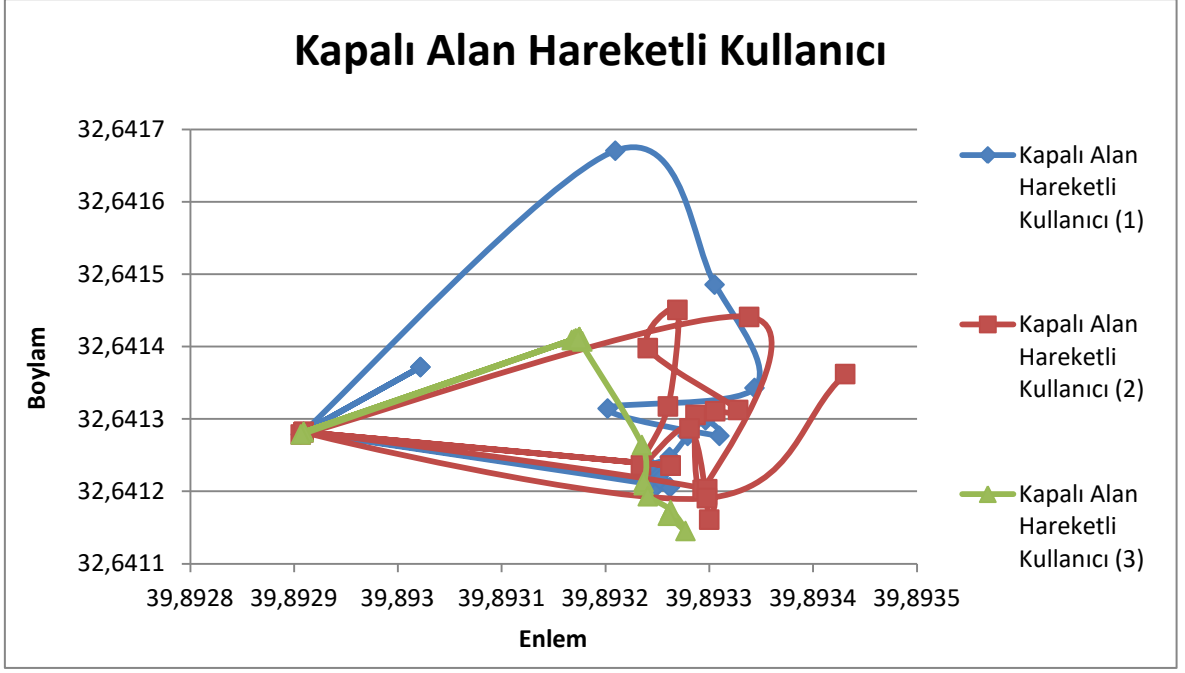


Şekil 3.6. (a) Kapalı alanda ve (b) açık alanda izlenen yol

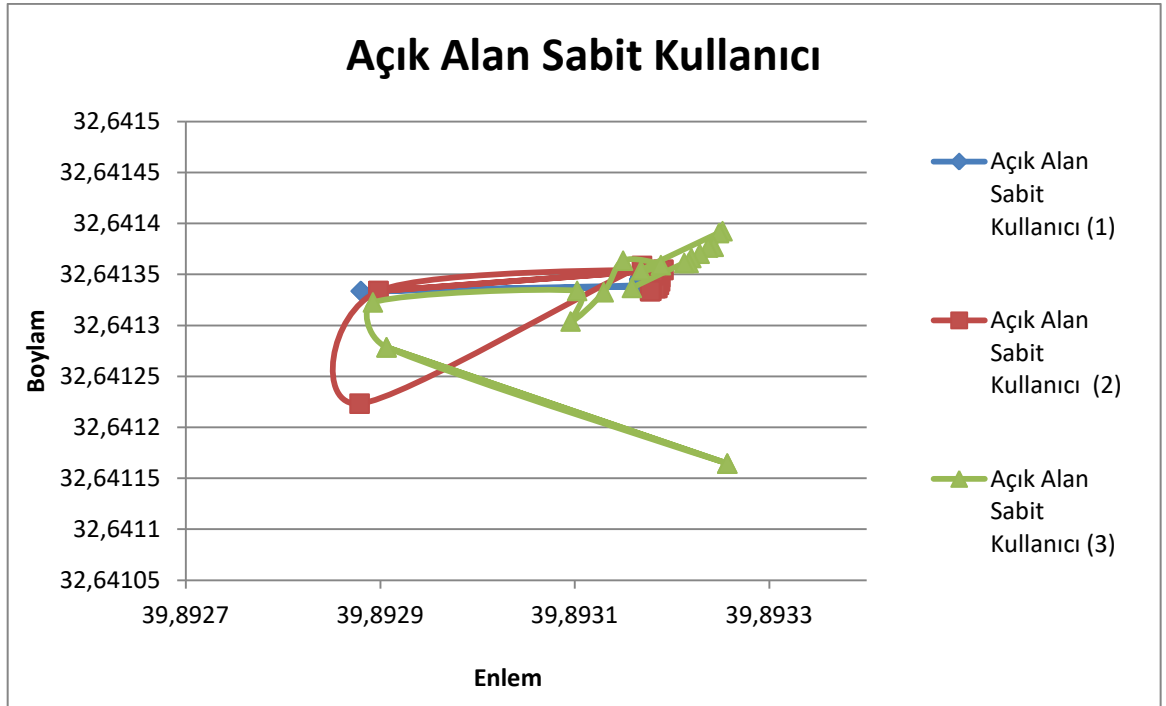
Bağlantılar yapıldıktan sonra kapalı ve açık alan için de testler yapıldı, her iki alan için hareketli kullanıcı 3 kez aynı alanda yürüdü.(Şekil 3.6.) Sonuçların Enlem-Boylam grafikleri çıkartıldı (Şekil 3.7, Şekil 3.8, Şekil 3.9, Şekil 3.10).



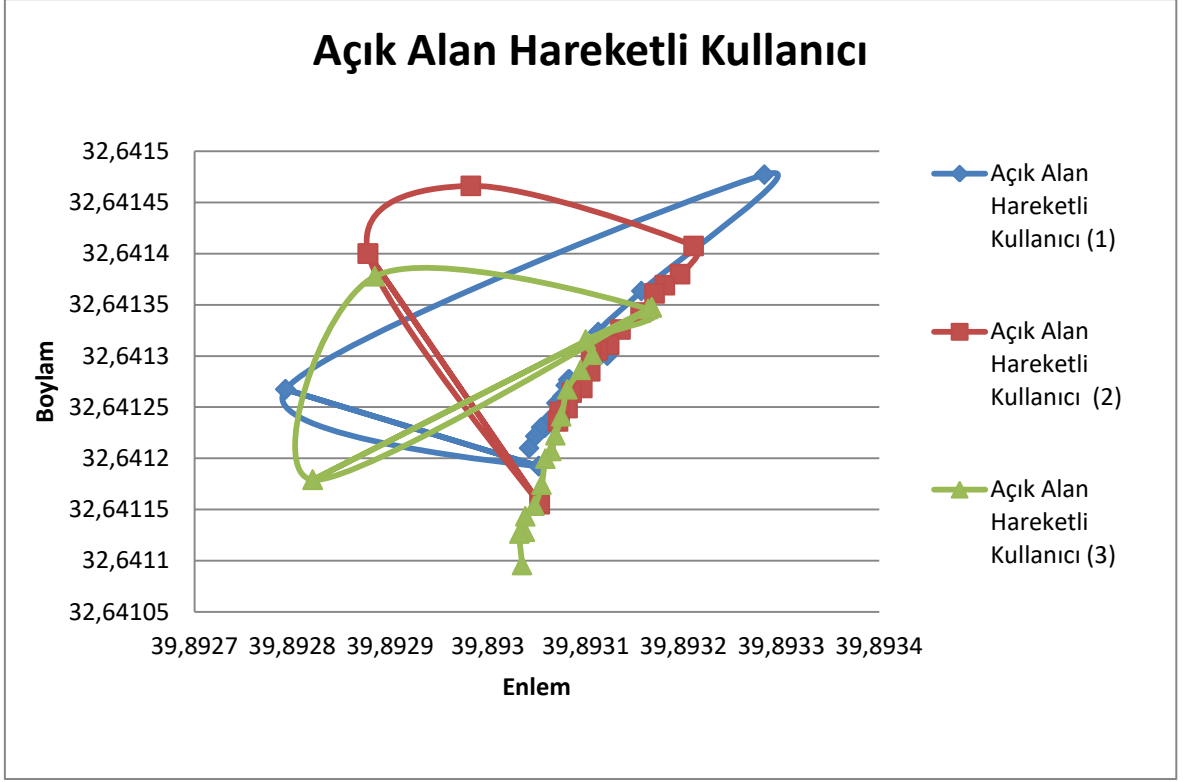
Şekil 3.7. Kapalı alanda 3 sabit kullanıcı ile yapılan testin sonuçları



Şekil 3.8. Kapalı alanda 3 hareketli kullanıcı ile yapılan testin sonuçları



Şekil 3.9. Açık alanda 3 sabit kullanıcı ile yapılan testin sonuçları

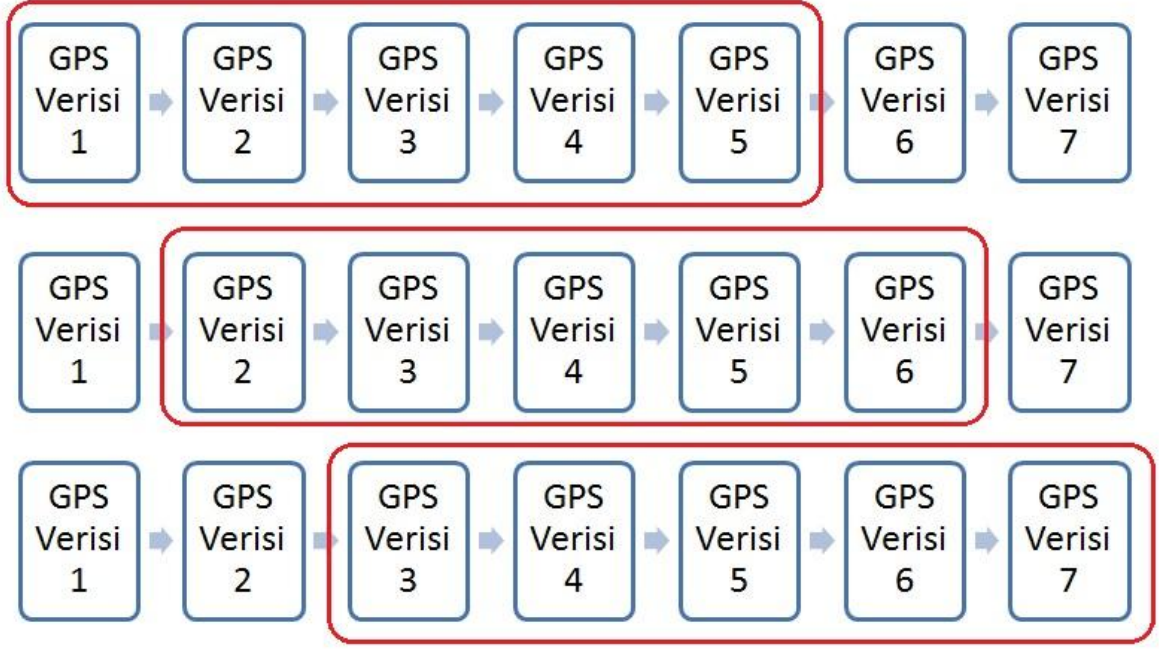


Şekil 3.10 Açık alanda 3 hareketli kullanıcı ile yapılan testin sonuçları

Sonuçlardan hareketle özellikle ilk gelen 5 veri güncellemesinde tutarsızlıklar görülmesine rağmen 10 güncelleme sonrası verinin tutarlılığı göze çarptı. Ayrıca bazı durumlarda verilerde atlamalar görüldü. Konum tabanlı AG uygulamaları bu tarz beklenmedik atlamalar uygulamada istenmedik sonuçlara neden olabilmektedir. Bunun için gelen konum verilerini hiçbir işleme tabi tutmadan kullanmaktansa bir tür filtrelemeden geçirip kullanmak daha doğru sonuçlar üretecektir.

3.4. Konum Bilgilerini Filtreleme

Daha önce de bahsedildiği üzere konum bilgilerini filtrelemek, özellikle verilerin düzensiz gelmesine karşı uygulama için bir kalkan işlevi görmesi adına önemli bir konudur. Bu çalışmada GPS verilerini filtrelemek için Kayan Pencere yöntemi gerçekleştirilmiştir. Buna göre gelen konum verileri direk olarak kullanılmaktansa belirli sayıda gelen verinin ortalamasının kullanılması daha iyi sonuçlar vereceğinden bu yöntem seçilmiştir. Uygulamaya gelen en son 5 verinin ortalaması alınır ve kullanıcının konumu burası kabul edilir (Şekil 3.11). Bu sayede gelen uyumsuz bir veri ya da atlama, ortalamaya katılacağından, etkisi küçültülmüş olur. Bu da ekrana yerleştirilecek sanal nesnenin daha tutarlı davranmasını sağlar.



Şekil 3.11. Kayan Pencere filtreleme yöntemi

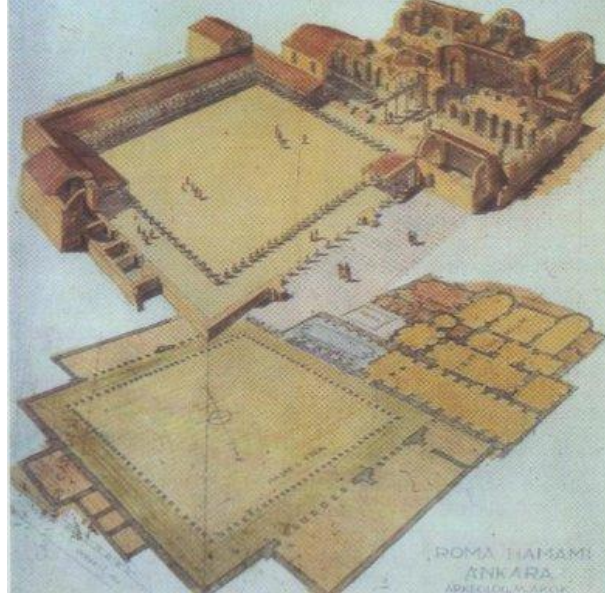
Kayan pencere modelinin alternatifi olarak Kalman Filtresi de uygulanabilir. Kalman Filtresi, sistemin önceki bilgilerinden hareketle sonraki adımını tahmin etmeye çalışır. Sistem, kendi tahmin ettiği ve çıkan sonuç arasındaki farkı sürekli ölçer ve bu farkı sonraki tahminler iyileştirmek için kullanır. Kalman Filtresi teorik olarak GPS verileri filtrelemek için oldukça uygundur. Ancak Kalman Filtresi uygulandığında, kayan pencere yöntemine göre çok daha fazla hesap yapılır. Yazılacak uygulama mobil cihazlar üzerinde çalışacağı için; Kalman Filtresinden, uygulamanın ve modelin ağırlığı ile filtrenin hesapları birleşince sistemi çok fazla yavaşlatacağı düşünülerek vazgeçildi.

4. MODELLERİN OLUŞTURULMASI

AG uygulamalarında kullanılacak 3B modellerin tasarlanması kullanıcıda oluşturulacak gerçeklik hissi adına önemli bir konudur. Özellikle bu çalışmada olduğu gibi kültürel miras alanları için tasarlanan modeller, tarihi yapının özelliklerini mümkün olduğunca iyi yansıtması gelecek nesillere aktarılacak bilginin doğruluğu açısından büyük önem arz etmektedir.

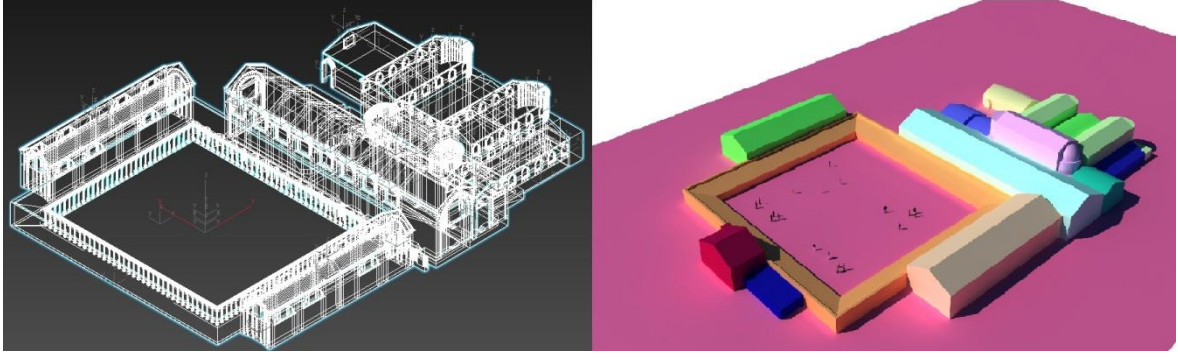
4.1. Modelleme

Modelin tasarımı için, Ankara Roma Hamamı'nın arkeologlar tarafında yapılmış çizimleri incelendi. Tasarımlar için profesyonel bir tasarım uygulaması olan 3D Studio Max uygulaması kullanıldı.

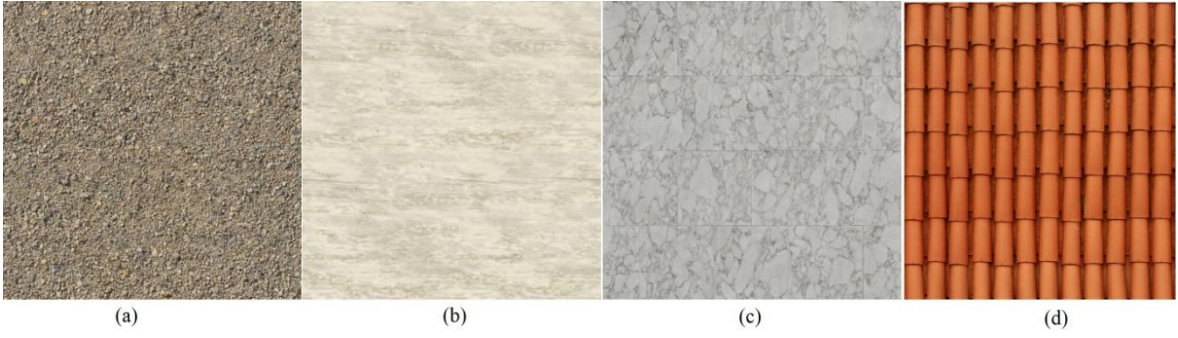


Şekil 4.1. Ankara Roma Hamamı Çizimi [53]

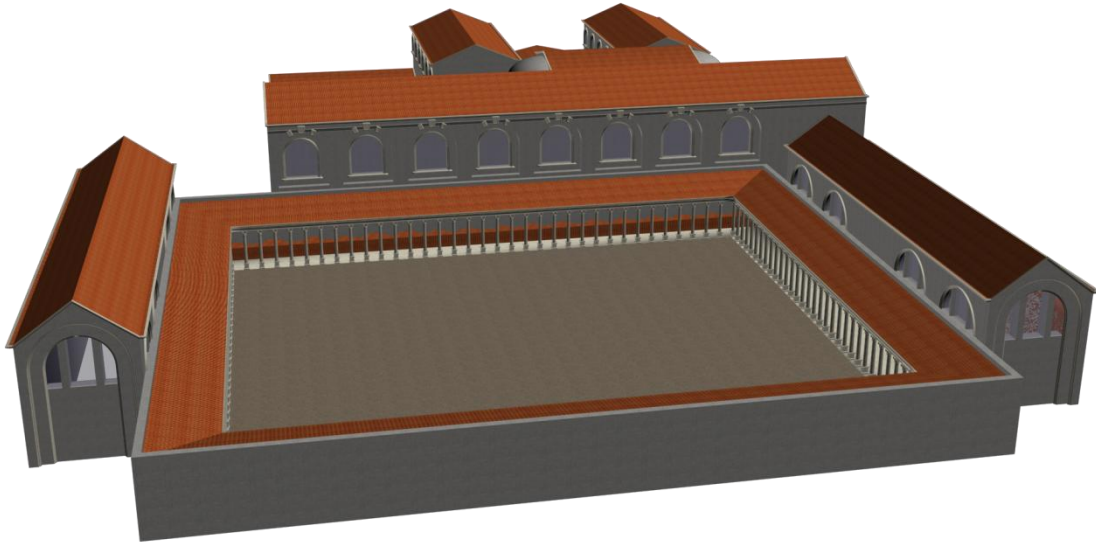
Tasarlanması düşünülen modelin ön çizimleri yapıldı (Şekil 4.2). Uygulamanın araçlarını kullanarak ve çizimlerden yola çıkarak hamamın 3B modeli tasarlandı. Modelin gerçekçi görünmesi için kullanılacak dokular araştırıldı ve düzenlendi. Bunun için tasarlanan modelde binaların; çatıları, duvarları ve yerleri için ayrıca tüm binaların ortasındaki alan için dokular bulundu ve ataması yapıldı (Şekil 4.3). Sonuç olarak ortaya oldukça gerçekçi bir model çıktı (Şekil 4.4). Tasarlanan model geliştirme ortamında kullanılabilmesi için 3D Studio Max programının ile ".fbx" formatına dönüştürüldü. Ortaya çıkan model dosyasının boyutunun 300 MB aştığı görüldü. Bu da mobil uygulamaya eklenecek bir model için oldukça büyüktür.



Şekil 4.2. Tasarlanan Model



Şekil 4.3. Kullanılan dokular : (a) Binaların ortasındaki alan, (b) mermer alanlar, (c) bina duvarları, (d) bina çatıları.

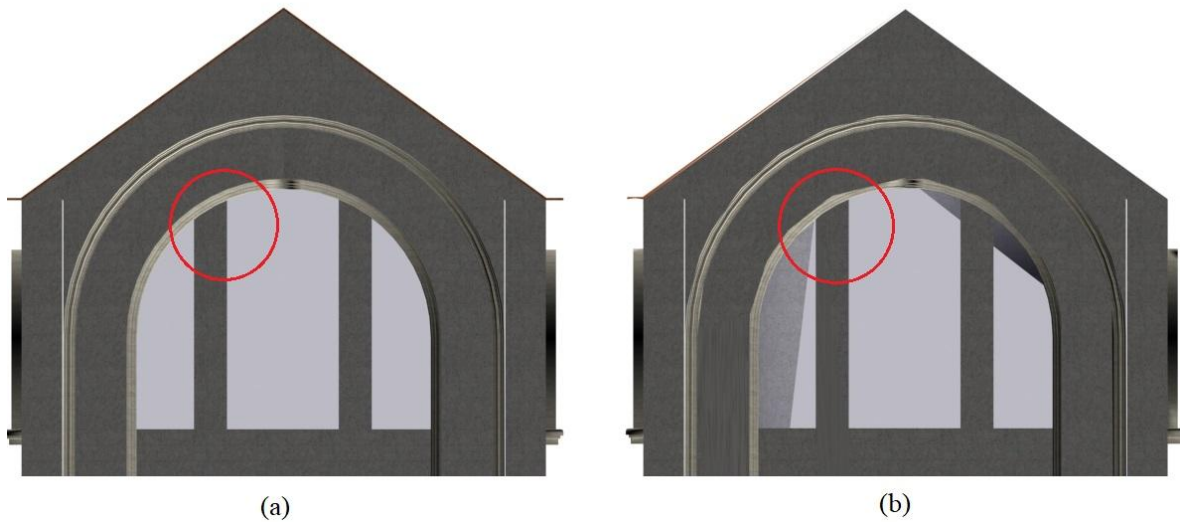


Şekil 4.4. Hamamın 3B modeli

4.2. Optimizasyon

Önceki bölümde de bahsedildiği üzere tasarlanan modeli boyutları, özellikle mobil uygulama geliştirmede kullanılacaksa, uygulamanın çalışması adına önem taşır. Mobil aygıtların en önemli avantajlarından olan taşınabilirlik, bazı dezavantajları da -doğal olarak- yanında getirmiştir. Taşınabilir olarak tasarlanan aygıtların hafif olabilmesi için içindeki donanım özelliklerinden feragat etmektedirler. Bu da onları taşınamaz aygıtlardan daha zayıf kılar. Sonuç olarak mobil aygıtlar için tasarlanmış uygulamalar, masaüstü bilgisayar vb. aygıtlar için tasarlanmış uygulamalara göre daha "hafif" olmalıdırlar.

Tasarlanan modelin yukarıda bahsedilen nedenlerle boyutunun düşürülmesi gerekmektedir. Bunun için modelin genellikle detay sayısı azaltılmaya çalışılır. Modelin tasarlandığı uygulama olan 3D Studio Max'in multires aracı ile köşeleri ve yüzeyleri azaltmak mümkündür. Şekil 4.5'te multires uygulanmış bir modeldeki değişim görülebilmektedir.



Şekil 4.5. (a) Tasarlanan bir bina modeli ve (b) multires ile köşe sayısı %50 azaltılmış hali

Şekil 4.5'te de görülebilen bina modeline multires uygulandığında verteks yüzdesi oranı düşürüldüğünde oluşan değişiklikler Çizelge 4.1'de görülebilmektedir. Verteks yüzdesi hiç düşürülmemişken çok detaylı olan model verteks yüzdesi düşükçe detaylarını kaybetmekte ancak çalışacak sistem için daha uygun hale gelmektedir.

Gereksiz detayları silmek adına oldukça faydalı olan multires özelliği verteks yüzdesi fazla düşürüldüğünde şeklin bozulmasına yol açmaktadır. Bu sebeple detaylar azaltılırken istenmeyen sonuçları doğuracak sınır geçilmemelidir.

Çizelge 4.1. Modelin içindeki bir binaya multires uygulandığında geometrik şekil sayısı.

	Verteks Oranı (%)		
	%100	%75	%50
Çokgen Sayısı	79.220	59.223	39.318
Üçgen Sayısı	79.220	59.223	39.318
Kenar Sayısı	237.660	177.669	117.954
Verteks Sayısı	40.144	30.109	20.072

Bina modelinin verteks yüzdesi azaltılırken bir yandan da şekildeki bozulmalar gözlemlendi. Verteks yüzdesi %50'nin altına düştüğünde şekildeki bozulmalar estetik açıdan görmezden gelmenin çok mümkün olamayacağı noktaya geldi. Bu sebeple projede kullanılacak modelin verteks yüzdesinin en fazla %50 düşürülmesine karar verildi.

5. GELİŞTİRİLEN İŞARETÇİ TABANLI UYGULAMALAR

Önceki bölümlerde de anlatıldığı üzere AG video görüntüsü üzerine sanal görüntü, nesne veya herhangi bir türde bilgi bindirme işlemidir. Bu işlem gerçek zamanlı olarak kullanıcıya sunulduğunda oldukça ilgi çekici bir içerik ortaya çıkar. Bu içerikleri oluşturmanın bir yolu işaretçi tabanlı sistemler iken diğeri de konum tabanlı sistemlerdir.

Bu kısımda, proje kapsamında araştırılan AG Yazılım Geliştirme Araçları (Software Development Kit - SDK) ve bu kütüphanelerin karşılaştırılması, geliştirilen ortamı ve geliştirilen işaretçi tabanlı uygulamalar anlatılacaktır.

5.1. Artırılmış Gerçeklik Yazılım Geliştirme Araçları

AG teknolojisinin, özellikle Android ve IOS tabanlı akıllı telefonların ortaya çıkmasıyla popülaritesi artmış ve bu alanda bir çok yazılım geliştirme aracı ortaya çıkmıştır. Yazılım geliştirme araçları AG uygulamalarının geliştirme aşamasını oldukça hızlandırmaktadır. Genel olarak dezavantajları ise geliştiricinin özgürce hareket etmesini engellemektedirler.

Bu çalışmada yazılım geliştirme araçlarında genel olarak aranan özellikler son dönemin en popüler oyun motorlarından olan Unity 3D'ye olan desteği, konum tabanlı çalışması ve ücretsiz olmasıdır.

Pek çok yazılım firması AG teknolojisinin geleceğinin parlak olacağını düşünmüş ve AG uygulamaları geliştirmeyi kolaylaştıracak yazılımlar üzerinde çalışmıştır. Günümüzde bir çok AG SDK'sı bulunsa da bunların en gelişmiş ve en popüler olanları Çizelge 5.1'de listelenmiştir.

Çizelge 5.1. Yazılım Geliştirme Kitleri Karşılaştırması [54]

Yazılım Geliştirme Kiti	Teknolojiler (Desteklenenler √ ile işaretlenmiştir.)			
	Tür	Unity 3D	GPS	Yazılım Diller
Metaio*	Ücretsiz**	√	√	-
Vuforia	Ücretsiz**	√	-	C# (Unity 3D)
Wikitude	Ücretsiz**	√	√	Javascript, Java
Kudan AR SDK	Ücretsiz**	√	-	Java
ARToolKit	Açık Kaynak	√	-	C++, C# (Unity 3D)

*Metaio, Apple Inc. tarafından satın alınıp geliştirmeleri durdurulmuştur.
** Ücretli sürümleri de bulunmaktadır

AG teknolojinin çok fazla popülerleşmediği 2000li yılların başında kurulan Metaio[55], ilk projelerinde Volkswagen ile çalışmıştır [56]. Daha sonraları AG yazılımlarının mobil telefonlar için geliştirilmeye başlandığı ilk yıllarda Metaio SDK oldukça popülerleşmiş ve işaretçisiz izleme yapabilen teknolojiler sunmuştur. Ancak 2015 yılında Apple Inc tarafından satın alınmış ve geliştirmeleri son kullanıcıya kapatılmıştır [57].

Bir diğer önemli SDK Vuforia[58] olarak öne çıkmaktadır. Vuforia Android, IOS ve Windows tabanlı sistemler için çözümler sağlamakla birlikte Unity 3D oyun motoru ile de geliştirme seçeneği mevcuttur. Resim tanıma, çoklu hedef tanıma, metin tanıma ve VuMark (Kullanıcı tanımlı yeni nesil bar kod) tanıma gibi bir çok izleme özelliği sunmaktadır.

Wikitude SDK [59] ise Vuforia gibi Android, IOS ve Unity 3D'yi desteklemesinin yanında Epson Moverio gibi akıllı gözlükler için AG uygulamaları geliştirmeye yarayacak çözümler sunmaktadır. Wikitude'un Android tabanlı sistemler için yerel Java API (Application Programming Interface - Uygulama Programlama Arayüzü) bulunmaktadır. Ancak tüm özelliklerini kullanmak Java API ile mümkün değildir. JavaScript API, Wikitude'un özellikle konum tabanlı AG uygulamaları geliştirmek için gerekli özelliklerini bulundurur. Konum tabanlı özelliklerinin yanında Wikitude'u ilgi çekici kılan bir diğer özelliği ise SLAM çözümdür. SLAM, konum belirleme ve haritalama işlemlerinin eş zamanlı olarak yapılması işlemidir. Bu özellik ile sayesinde açık kapalı alanlarda oldukça tutarlı bir nesne yerleştirme yapılabilmektedir.

Hem işaretçili hem de işaretçisiz izleme sunan bir diğer SDK ise Kudan AR SDK'dır [60]. Kudan'ın motoru C++ ile yazılmış ve sistemin hızlı çalışması adına optimizasyonu assembly ile yapılmıştır. Diğerlerine göre daha yeni olan Kudan resim izleme, SLAM gibi özellikleri de sunmaktadır. Kudan ile uygulama geliştirmek tamamen ücretsizdir. Uygulamayı yayınlama kısmında ücretli seçenekler mevcuttur.

Listedeki tek açık kaynak SDK olan ARToolKit [61] kendisini dünyanın en çok kullanılan AG izleme kütüphanesi olarak tanımlamaktadır. Görsel izleme algoritması üzerinde geliştiricinin tümüyle kontrol sahibi olduğu kütüphanede, Android, IOS, Windows, Linux, Mac OS X tabanlı sistemlere uygulama geliştirilebilir. Kütüphanenin ayrıca akıllı gözlükler için de desteği bulunmaktadır. Mobil aygıtlar için GPS ve pusula kontrolü de mevcuttur. Tüm bu özellikleri bir yana ARToolKit'i popüler yapan en önemli özelliği açık kaynak olmasının getirdiği özgürlüktür.

5.2. Yazılım Geliştirme Ortamı

Android tabanlı bir sisteme yazılım geliştirme için bir çok araç mevcuttur. Android tabanlı sistemler ilk ortaya çıktığında yazılım geliştiriciler tarafından en çok Eclipse [62] kullanılmaktaydı. Mayıs 2013'te Google, Android'e yazılım geliştirmek için Android Studio [63] platformunu duyurdu ve Eclipse'e desteğini çekti. Bunun sonucu olarak Android Studio, Android tabanlı sistemler için en popüler ortam haline geldi.

Bu tez çalışmasında Android tabanlı mobil sistemlere yazılım geliştirirken kullanılan araçlardan ilki Android Studio'dur. Android Studio, Android İşletim Sistemi'ne sahip mobil cihazlar için Java tabanlı uygulamalar geliştirmeyi sağlar. Android Studio, IntelliJ IDEA tabanlı olarak geliştirilmiştir. Gerçek bir Android cihaz olmadan, geliştirilen uygulamanın sanal cihazlar üzerinde test edilmesine olanak sağlar.

Çalışmada kullanılan bir diğer ortam ise Unity 3D [64] oyun motorudur. Unity 3D bilgisayarlar, mobil cihazlar ve konsollar (Playstation [65], Xbox [66], Nintendo WiiU [67]) için genel olarak oyun geliştirmeye yarayan, oyun motorudur. Unity 3D oyun motorunun popüler olmasını sağlayan en önemli iki özelliği, geliştirilen oyunun; bilgisayarda kurulum gerektirmeden çalışabilmesi ve istenilen platformun (bilgisayar, mobil aygıt yada konsol) formatına kolayca çevrilerek o platformlarda çalışabilmesidir. Unity 3D'nin kişisel versiyonu ücretsizdir. Kişisel versiyon ile geliştiriciler istedikleri uygulamayı geliştirip yayımlayabilirler. Ancak kişisel versiyonda uygulama açılırken 3 saniye kadar "Made with Unity" yazısı ve Unity 3D logosu gözükür ve geliştirilen uygulama yayınladıktan sonra uygulamadan kazanç 100.000\$'ı geçerse ücretli seçeneklerden birinin satın alınması gerekir.

Unity 3D bir oyun motoru olmasına rağmen sağladığı özgürlük ve kolaylıklarla çapraz platformlarda çalışacak ve farklı uygulamalar geliştirecek kişiler için de oldukça uygun bir araçtır. Unity 3D ile oyun dışında simülasyon, animasyon filmler, SG ve AG uygulamaları geliştirilebilir [68]. Çizelge 5.1'de de görülebileceği gibi bir çok AG SDK'sı da Unity 3D destekli olarak gelmektedir. Unity 3D'nin çalışma ortamı 3B olması sebebiyle özellikle konum bazlı AG uygulamalarında sanal nesnelerin yerleştirilmesinde kolaylıklar sağlar.

Sonuç olarak bu çalışmada ortaya çıkan yazılımlar Android Studio ve Unity 3D kullanılarak geliştirilmiştir.

5.3. Yazılım Geliştirme Araçları ile İşaretçi Tabanlı Olarak Geliştirilen Uygulamalar

Bölüm 5.1'de incelenen AG Yazılım Geliştirme Araçlarının gücünü ve tutarlılığını test etmek adına hepsi ile ayrı ayrı işaretçi tabanlı uygulamalar geliştirildi. Uygulama geliştirme ortamı olarak Unity 3D kullanıldı. Geliştirilen uygulamalarda model olarak fazla karmaşık olan Roma Hamamı modeli yerine Unity Asset Store'dan bir Sezar büstü modeli [71] kullanıldı.

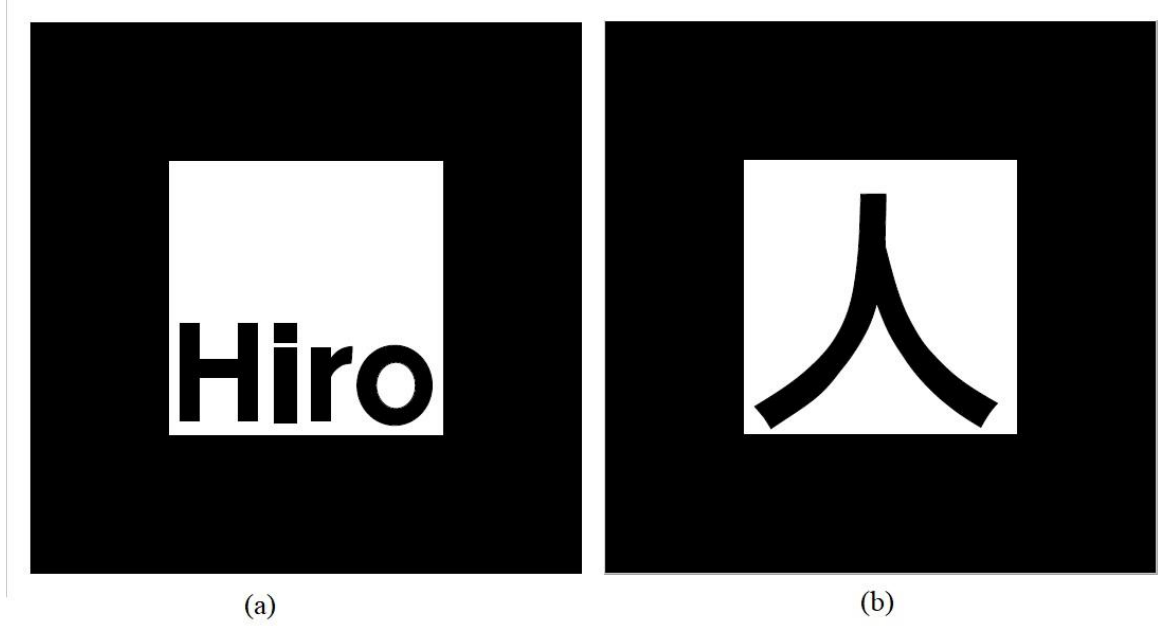


Şekil 5.1. Geliştirilen Uygulamalarda Kullanılan Model [71]

5.3.1. ARToolKit ile Geliştirilen Uygulama

ARToolKit ile geliştirilmiş uygulamalarda kullanılan işaretçiler belli standartlara sahip olmalıdırlar. SDK varsayılan olarak 2 adet işaretçi sunar (Şekil 5.2). Bu işaretçiler dışında geliştiriciler kendilerine özel başka işaretçiler de tanımlayabilirler. Ancak bir şeklin SDK tarafından işaretçi olarak tanımlanabilmesi için bazı özellikleri taşıma gerekmektedir. ARToolKit'de işaretçi olarak kullanılacak şekiller aşağıdaki özellikleri taşımalıdır.

- Kare şeklinde olmalıdır.
- Dış sınırları arlıksız olmalı ve mümkün olduğunca arkaplan renginin zıttı olmalıdır.
- İşaretçi resmi olarak adlandırılan ve sınırlar içinde kalan bölgedeki resim asimetrik olmalıdır.



Şekil 5.2. ARToolKit için varsayılan işaretçiler (a) Hiro ve (b) Kanji kalıbı

Geliştirilen uygulamada kullanılmak üzere Hiro kalıbı (Şekil 5.2) seçildi. ARToolKit'in Unity için hazırlanmış olan paketi geliştirme ortamına aktarıldı. Bu paket içinde bulunan ARController, ARMarker, AROrigin, ARCamera ve ARTrackedObject adlı kod parçaları geliştirilen uygulamada kullanılması gerekmektedir.

Uygulamanın çalışabilmesi için SDK'nın belirlediği bazı işlemler gerçekleştirilmek zorundadır. Bunun için öncelikle Unity üzerinde bir boş nesne oluşturuldu ve nesneye "ARToolKit" adı verildi. ARToolKit nesnesi geliştirilen uygulamadaki en dış parça olarak kullanıldı. Bu nesneye ARController ve ARMarker kod parçaları yerleştirildi. Ayrıca nesnenin katmanı "AR background" olarak atandı. Bu işlemten sonra ARToolKit nesnesinin alt nesnesi (çocuğu) olacak şekilde yeni bir boş nesne oluşturuldu ve ismi "Root" (Kök) olarak atandı. AROrigin kod parçası Root nesnesi üzerine atandı ve nesnenin katmanı "AR foreground" haline getirildi. Root nesnesinin çocuğu olacak şekilde bir kamera nesnesi oluşturuldu. Kamera nesnesinin katmanı atası Root nesnesi gibi "AR foreground" olarak atandı. Kamera nesnesine kod parçası olarak ARCamera eklendi. Yine Root nesnesinin çocuğu olacak şekilde boş bir nesne oluşturuldu, adı "Marker" katmanı "AR foreground" olarak atandı. Marker nesnesi üzerine ARTrackedObject kod parçası atandı. Sonraki aşamada Sezar Büstü modeli Marker nesnesinin çocuğu olarak atandı ve katmanı "AR foreground" olarak değiştirildi.

Çizelge 5.2.ARToolkit ile geliştirilen uygulamadaki nesnelere.

Nesne	Katman	Ata	Kod
ARToolkit	AR background	-	ARController, ARMarker
Root	AR foreground	ARToolkit	AROrigin
Camera	AR foreground	Root	ARCamera
Marker	AR foreground	Root	ARTracked
CaesarModel	AR foreground	Marker	-

Sonraki aşamada işaretçi olarak kullanılacak Hiro kalıbı, ARToolkit nesnesinin içine atanmış ARMarker ve Marker nesnesinin içine atanmış ARTracked kod parçalarında seçildi. Unity'de kamera nesnesinin, sadece seçilen objeleri ekranda göstermesine (render) yarayan culling mask özelliğinde etiket olarak "AR foreground" seçildi. Son olarak uygulamanın Android cihaz üzerinde çalışması için ayarlamaları yapıldı ve uygulama test edildi.



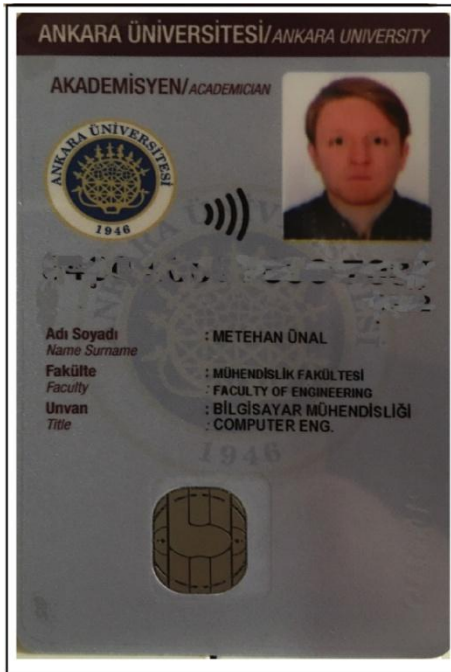
Şekil 5.3. ARToolkit ile geliştirilen işaretçi tabanlı AG uygulamasından alınan görüntü.

Burada nesnelere eklenen kod parçalarının yaptığı işlemler önemlidir.

- ARController, AG ortamının oluşturulması ve izleme işleminin yönetilmesini kontrol eder. Ayrıca arka planda video görüntüsünün oluşturulmasını sağlar.
- ARMarker, işaretçilerin bulunduğu yeri, türünü ve sayısını yönetir. AROrigin, sahnedeki dinamik nesnelere kontrol eder.
- ARTrackedObject, ARMarker kodu aracılığı ile bulunan işaretçilerden hangisinin geçerli olacağını yönetir. ARTrackedObject kodunun atandığı nesne, işaretçinin görüntüde bulunduğu yere göre sahnede yer değiştirir.
- ARCamera, AG içeriği ve kamera görüntüsünü ilişkilendirir.

5.3.2. Vuforia ile Geliştirilen Uygulama

ARToolKit'ten farklı olarak Vuforia'nın açık kaynak olmaması bazı durumlarda kısıtlamaları da beraberinde getirmiştir. Vuforia'nın Unity 3D için oluşturulmuş paketini kullanmak için sitesinden [58] üye olarak ücretsiz (yada ücretli) lisans anahtarı alınması gerekmektedir.



(a)



(b)

Şekil 5.4. (a) Sistemde 2 yıldız alan bir resim ve (b) 5 yıldız alan bir resim.

Vuforia ile geliştirilen işaretçi tabanlı AG uygulamasında kullanılacak işaretçiyi önceden Vuforia'nın sitesinde oluşturulan veritabanına kaydetmek gerekmektedir. ARToolKit işaretçilerinden farklı olarak burada normal resimler kullanılabilir. Veritabanına kaydedilen resimler site tarafından izlemeye ne kadar elverişli olduğunu göstermek adına yıldız şeklinde puanlanmaktadır. 5 yıldız alan bir resmin işaretçi olarak tespit ve takibi çok rahat yapılabilirken 1 yıldız alan bir işaretçinin tespiti ve takibi göreceli olarak daha zordur. Şekil 5.4'te görülebileceği üzere soldaki kimlik resmi 2 yıldız almışken sağdaki Ankara Üniversitesi logosu 5 yıldız almıştır. Sonuç olarak sağdaki resmin tespiti ve takibi daha kolaydır. Yine de sistem soldaki resimle de çalışabilir.

Veritabanına eklenen resimler önce siyah-beyaz hale getirilmekte sonra da özellikleri çıkartılmakta ve puanlandırılmaktadır. Ankara Üniversitesi logosunun özellikleri çıkartılmış hali Şekil 5.5'te görülebilir.



Şekil 5.5. Ankara Üniversitesi logosunun özellikleri çıkartılmış hali.

Bu işlemten sonra veritabanı Unity'e aktarılmak üzere siteden indirildi. Vuforia geliştiricisi, Unity'nin sağladığı kamera yerine kendi paketinden çıkan kamerayı kullanmaya zorlamaktadır. Bir prefab (önceden hazırlanmış nesne ve kodların birleştirilip

paketlenmiş hali) olarak Vuforia paketinin içinden çıkan kamerayı kullanmak için ekranda varsayılan olarak gelen kamerayı silip, bu nesneyi ekrana aktarmak gerekmektedir.

ARCamera adı verilen bu nesne sadece kamera görüntüsü alınmasına değil ayrıca uygulama için lisans anahtarının alınması, eğer kullanılacaksa akıllı gözlüğün türü, aynı anda kaç işaretçinin takip edileceği gibi bir çok özelleştirme yapılmasına olanak sağlar.

Vuforia kullanılarak geliştirilen AG uygulamalarının çalışabilmesi için Vuforia'nın sitesinden alınan lisans anahtarını kullanmak gerekmektedir. Lisans anahtarı almak için Lisans Yöneticisi bölümüne girildiğinde 3 seçenek mevcuttur. Bu seçenekler geliştirme ve araştırma amaçlı "Development", uygulamanın geliştirildiği şirketin müşterilerine yönelik "Consumer" ve şirket içi çalışanlara yönelik "Enterprise" olarak sunulmaktadır. Development (Geliştirme) seçeneği ücretsizken diğer iki seçenek değişken olarak ücretlendirilmektedir.

Geliştirme seçeneği ile oluşturulan lisans anahtarı siteden kopyalanıp uygulamadaki ayarlar kısmına yapıştırıldı. Site üzerinde daha önce oluşturulan işaretçi veri tabanı Unity seçeneği tercih edilerek indirildi. İndirilen ve bir Unity paketine dönüşmüş olan hedef işaretçiler, Unity'nin içine aktarıldı.

Vuforia paketiyle birlikte gelen bir diğer prefab olan ImageTarget sahneye aktarıldı. Bu prefabın üzerindeki "Image Target Behaviour" kodu üzerindeki veritabanı seçeneği daha önce yüklenen işaretçi veritabanı olarak seçildi. Bu veritabanı üzerinde birkaç işaretçi birden olabileceği için alt seçenekte bulunan işaretçi olarak kullanılacak resim seçildi.

Kamera görüntüsünün üzerinde işaretçin tespit edildiği yerde bindirilecek olan Sezar Büstü modeli yine Unity Asset Store'dan indirildi ve projeye aktarıldı. Vuforia ile geliştirilen bir uygulamada modelin işaretçinin üzerinde görüntülenebilmesi için daha öncede bahsedilen ve sahneye aktarılan ImageTarget nesnesinin çocuğu olarak atanması gerekmektedir. Bu amaçla indirilen modelin prefabı sahneye atıldı ve ImageTarget nesnesinin çocuğu haline getirildi. Modelin boyutları ve işaretçi üzerinde oluşması gereken açısı ayarlandı.

3 boyut hissini daha iyi verilmesi ve projeden alınan resimlerin özgünlüğü adına modelin üstüne 3B metin eklendi.

Son olarak Android tabanlı sistemlere aktarılabilmesi için ayarlamaları yapıldı ve aktarıldı. Geliştirilen uygulamada dikkat çeken yanlardan ilki sol altta gözüken Vuforia logosu olurken diğeri kamera görüntüsünün çözünürlüğünün kalitesi olmuştur.



Şekil 5.6. Vuforia ile geliştirilen işaretçi tabanlı AG uygulamasından alınan bir görüntü.

5.3.3. Wikitude ile Geliştirilen Uygulama

AG yazılım geliştirme araçlarından en popülerlerinden olan Wikitude, Vuforia gibi ücretli ve ücretsiz versiyonlara sahiptir. Ücretsiz versiyonu daha çok deneme versiyonu tarzında olan Wikitude; geliştiricileri, projelerde kullanılacak işaretçiler için kendi işaretçi yöneticisi ve Unity dışında kullanılacak modeller için o modellerin formatını değiştirmeye yarayan 3D Encoder gibi yan uygulamaları da kullanmaya zorlamaktadır.

Uygulama geliştirmeye başlamadan önce işaretçiler belirlendi. Şekil 5.4'te görülen ve Vuforia için de kullanılan 2 resim işaretçi olacak şekilde Wikitude Target Manager'a [72] yüklendi. Tıpkı Vuforia'da olduğu gibi Wikitude da işaretçileri takip edilebilirliğine göre puanladı. Wikitude'un işaretçileri puanlama sistemi 3 yıldızdan 0 yıldıza kadar olacak şekilde ayarlanmıştır. 3 yıldız alan bir işaretçi hemen hemen her koşulda rahatça takip edilebilirken 0 yıldız alan bir işaretçinin takibi yapılamamaktadır [72]. Buna göre Şekil

5.4'te görülen iki resimden Ankara Üniversitesi logosu olan 3 yıldız olarak takip için oldukça iyi olu bir işaretçi olarak ön plana çıkarken diğer resim 2 yıldız almıştır.

İşaretçiler oluşturulduktan sonra projeye yüklenebilmesi için indirildi. İndirme işlemi sonunda formatı WTC (Wikitude Target Collection) olan bir dosya oluştu ve bu dosya Unity'de oluşturulan proje içine aktarıldı.

Sonraki aşamada Wikitude Unity paketi indirildi ve projeye aktarıldı. Vuforia'da olduğu gibi Wikitude da Unity'nin varsayılan kamerasını kullanmaz. Bu amaçla ekrandaki kamera silindi ve paketle birlikte gelen WikitudeCamera prefabı sahneye eklendi. WikitudeCamera prefabının Unity kamerasından farkı sadece üzerinde atanmış olan WikitudeCamera kod parçası ve AG uygulanmasını sağlayabilecek ayar düzenlemeleri olarak gözlemlenmiştir. Burada WikitudeCamera kod parçası lisans anahtarının alınması ve test edilmesini yönetmesinin yanında kameranın çözünürlüğü, saniyede gözükecek resim sayısı gibi ayarları da yönetir. Çözünürlük ve saniyede gözükecek resim sayısı ayarlarının otomatik olarak ayarlanması uygulamanın daha verimli çalışması adına en iyi seçeneklerdir.

Sahneye bir kamera eklendikten sonraki adımda yine paketin içinde bulunan prefablardan ImageTracker prefabı sahneye aktarıldı. ImageTracker aslında içinde aynı adlı kod parçası bulunan boş bir nesneden ibarettir. İçindeki kod parçası ile izlenecek işaretçiler koleksiyonu seçilebilir. Ayrıca ImageTracker nesnesine bağlı Tracable adlı boş bir nesne daha bulunmaktadır. Bu nesnenin içinde de işaretçi koleksiyonundaki resimlerden hangilerinin izleneceğinin seçilebildiği ve işaretçilerin görüntüde tespit edildiklerinde programın tepkilerinin düzenlenebildiği TracableBehavior kod parçası bulunur. Eğer işaretçiler koleksiyonu projeye doğru şekilde aktarıldıysa, koleksiyonun içindeki resimler burada ayrı ayrı görüntülenir.

İşaretçi ekranda tespit edildiğinde görüntülenecek Sezar Büstü modeli yine Unity Asset Store'dan indirildi ve projeye aktarıldı. Modelin işaretçinin üzerinde görüntülenebilmesi için Trackable nesnesinin çocuğu olarak atanması gerekmektedir. Bu atama işlemi yapıldıktan sonra modelin boyutları ve duruş açısı düzenlendi. Vuforia projesinde yapıldığı gibi 3B metin oluşturuldu ve o da Traceable nesnesinin çocuğu olacak şekilde ayarlandı. Wikitude'un internet sitesinden lisans anahtarı indirildi ve projeye kopyalandı. En son aşama olarak da Android cihaz üzerinde çalışabilmesi için paket ismi, gerekli minimum Android versiyonu gibi ayarlamalar yapıldı ve cihaza yüklendi. Sonuçlar Şekil 5.7'de görülebilir.



Şekil 5.7. Wiktude ile geliştirilen işaretçi tabanlı AG uygulamasından alınan bir görüntü.

Sonuç olarak takip gücü oldukça yüksek olsa da Wiktude'un deneme versiyonunda ekranda beliren "Trial" filigranların estetik olarak görüntüyü bozduğu görüldü.

5.3.4. Kudan AR SDK ile Geliştirilen Uygulama

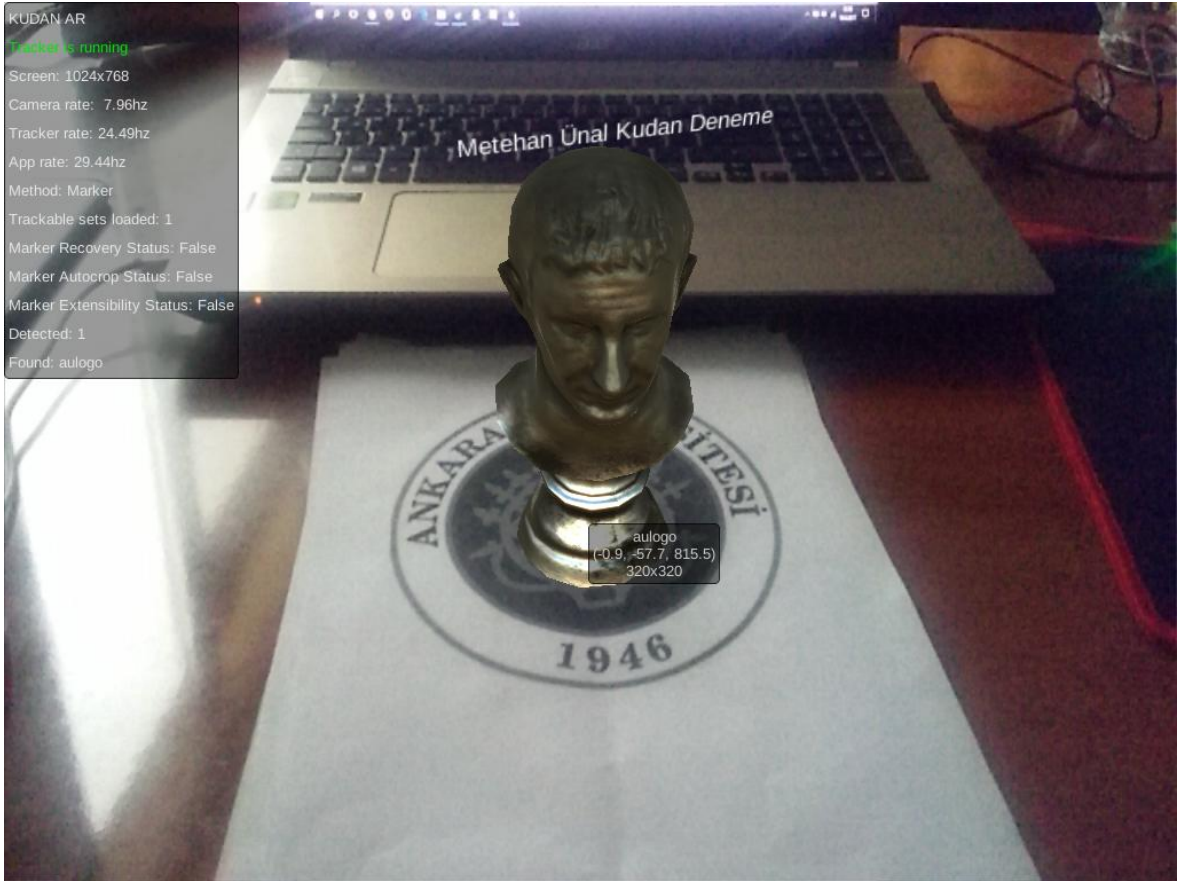
Kudan AR SDK da açık kaynaklı olmayan diğer yazılım geliştirme araçları gibi ücretli ve ücretsiz versiyonları bir arada sunmaktadır. Ücretsiz lisans ile geliştirilen uygulamanın ekranında Kudan filigranı gözükse de bu lisans ile AG motorunun tüm özellikleri elde edilebilmektedir.

Kudan ile geliştirilecek işaretçi tabanlı bir AG uygulamasında kullanılacak işaretçiler diğer yazılım geliştirme araçlarında da olduğu gibi kullanıcı tanımlı olabilmektedir. Ancak bu SDK ile Unity üzerinde geliştirilen uygulamalarda kullanıcı tanımlı işaretçilerin kullanılabilmesi için işaretçilerin, Kudan AR Toolkit adlı yazılım kullanılarak uygun formata dönüştürülmesi gerekmektedir. Bu yazılım aynı zaman Unity dışındaki çalışma ortamlarında kullanılacak 3B modellerinde uygun formata dönüştürülmesinde de kullanılmaktadır.

Geliştirilecek yazılımın ilk adımı olarak daha önce kullanılan ve Şekil 5.4'te görülen Ankara Üniversitesi logosu işaretçi olarak belirlendi. Bu amaçla logo Kudan AR Toolkit yazılımı yardımıyla uygun formata dönüştürüldü ve Unity'e aktarıldı.

Kudan'ın Unity için hazırlanmış paketi de çalışma ortamına aktarıldı. Açık kaynak kodlu olmayan diğer yazılım geliştirme araçlarında olduğu gibi Kudan da geliştiriciyi, Unity'nin varsayılan kamerası yerine kendi kamera prefabını kullanmaya zorlamaktadır. Bu amaçla yine sahnede bulunan kamera silindi ve paketle birlikte gelen "Kudan Camera" adlı prefab ekrana taşındı.

"Kudan Camera" prefabı normal bir kamera olmasının yanında üstünde Kudan Tracker kod parçası ile de birlikte gelmektedir. Bu kod parçası lisans anahtarının alınması, izleme yönteminin belirlenmesi gibi bir çok işlevi yerine getirir. Ayrıca prefabın çocuğu olacak şekilde 2 nesnesi daha vardır. Bu nesnelere ilki arkaplanda görüntünün oluşmasını sağlayan "BackgroundRender" ve izleme metodunu belirten "MarkerTracking".



Şekil 5.8. Kudan ile geliştirilen işaretçi tabanlı AG uygulamasından alınan bir görüntü.

Modelin görüntüde oluşan işaretçi üzerine bindirilebilmesi için sahne üzerinde boş bir nesne oluşturuldu ve üstüne pakette birlikte gelen "Marker Transform Driver" kod parçası yerleştirildi. Daha önceden projeye aktarılmış işaretçi bu kod parçası üzerinde seçildi. Son aşama olarak Sezar Büstü modeli ve 3B metin bu boş nesnenin çocuğu olacak şekilde ayarlandı. Android tablette çalışması için düzenlemeler yapıldı ve cihaza aktarıldı.

Sonuç olarak işaretçi takibi oldukça iyi olan Kudan'ın dokümantasyon konusunda eksikleri olduğu görüldü.

İncelenen 4 yazılım geliştirme aracı da işaretçi tabanlı sistemler için oldukça verimli çalıştı. İşaretçi tabanlı sistemler genel olarak kapalı alanlarda kullanılmak için uygun olduğundan açık alanlarda ve özellikle kullanıcının bakış açısının sıklıkla değiştiği bir sistemin tasarlanacak olması sebebiyle işaretçi olmadan bir AG uygulaması geliştirilen proje için daha uygun bir yöntem olacağı görüldü.

6. GELİŞTİRİLEN KONUM TABANLI MOBİL ARTIRILMIŞ GERÇEKLİK UYGULAMALARI

Bu bölümde daha önce işaretçi tabanlı olarak denemeleri yapılan AG yazılım geliştirme araçları ile geliştirilen konum tabanlı AG uygulamaları anlatılacaktır. İlk alt bölümde planlanan sistem ana hatları anlatılacak, sonra Ankara'daki Roma Hamamı'ndan bahsedilecek, sonraki kısımda AG yazılım geliştirme araçları kullanılarak geliştirilen yazılımlar anlatılacaktır.

Bu çalışmada Ankara Ulus'ta bulunan Roma Hamamları'nın uzman arkeologlar tarafından oluşturulmuş çizimlerinden hareketle tasarlanan 3B modeli mobil cihazlardan alınan kamera görüntüsü üstüne bindirilecek bir konum tabanlı AG yazılımı geliştirilmiştir. Buna göre model sanal olarak dünya üzerinde belli bir enlem ve boylam üzerine yerleştirilecek ve kullanıcılar bu alanda gezindiklerinde tarihi yapının modelini kamera görüntüsünün üzerinde görebileceklerdir.

6.1. Projenin Planı

Geliştirilen uygulama bir yerinde gerçeklik uygulamasıdır. Buna göre ziyaretçiler Roma Hamamı'nın harabelerinin bulunduğu alana giderler. Mobil aygıtlarını zaman içinde yıkılmış yada çok az kısmı ayakta kalabilmiş yapıların olduğu alana doğru döndürüp uygulamayı çalıştırdıklarında yıkık harabelerin yerinde 3B modeli görürler (Şekil 6.1).



Şekil 6.1. Yerinde Artırılmış Gerçeklik Uygulamasının Konsepti

Kullanıcılar alanda yürüdüklerinde 3B modelinde içinde yürürler ve tarihi yapıları inşa edildikleri ilk halleri ile görebilirler. Burada sistem birden çok kullanıcı için de çalışır. Kültürel miras alanına gelen herkes alanın üzerine bindirilmiş sanal modeli görebilir. Bu sayede kültürel miras alanları ziyaretçiler için daha ilgi çekici bir hale gelir.

6.2. Ankara Roma Hamamı

Ankara Roma Hamamı, Ulus Meydanı'nda Çankırı Caddesi üzerinde bulunur. 3. yüzyılda Septimius Severus'un oğlu Roma İmparatoru Caracalla tarafından Sağlık Tanrısı Asklepion adına yaptırılmıştır [69]. Hamam taştan ve tuğladan inşa edilmiş 80 x 130 metre boyutlarında bir yapıdır.



Şekil 6.2. Roma Hamamı'nın yukarıdan görünüşü (70)

Günümüzde hamamın büyük bölümü yıpranmış olsa da sütunların ve duvarların yerleri korunmuştur. Üzerinde restorasyon çalışmaları yapılan hamamın alanına müze kartıyla giriş yapılabilmektedir (Şekil 6.3).

Bu proje kapsamında hamam alanında çalışma yapmak için Kültür ve Turizm Bakanlığına başvurulmuş ancak izin alınamamıştır. Bu sebeple deneyler Ankara Üniversitesi Gölbaşı Yerleşkesi'nde park alanında yapılmıştır.



Şekil 6.3. Hamamın bulunduğu alandan görüntüler (70).

6.3. Artırılmış Gerçeklik Yazılım Geliştirme Araçları ile Geliştirilen Uygulamalar

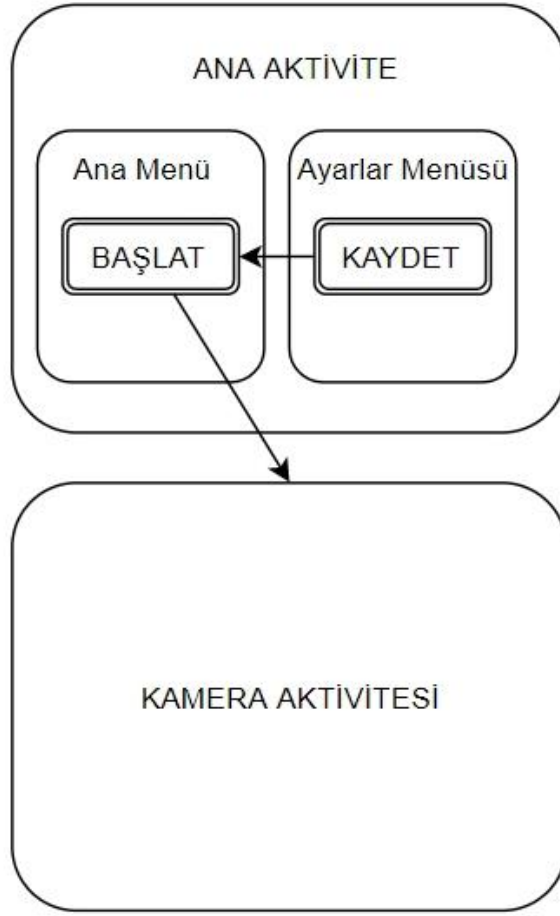
AG gerçeklikte konum tabanlı uygulama geliştirmek işaretçi tabanlı geliştirmeye göre oldukça zordur. AG yazılım geliştirme araçlarından hemen hepsi işaretçi tabanlı izlemeyi varsayılan olarak sunarken, işaretçisiz yada konum tabanlı izlemeyi sunan oldukça azdır. Bazı yazılım geliştirme araçları ise konum tabanlı sistemlerle ilgili herhangi bir destek sunmasa da Unity 3D oyun motorunun sağladığı imkanlarla bu araçları kullanarak konum tabanlı bir sistem geliştirme mümkündür. Bu bölümde iki SDK ile uygulama geliştirilecek, projenin bu SDKlar ile geliştirilebilirliği test edilecek ve varılan sonuçlar gösterilecektir.

6.3.1. Wikitude ile Geliştirilen Uygulama

Bölüm 5.1'de anlatılan yazılım geliştirme araçlarından konum tabanlı uygulama geliştirmeyi destekleyen Wikitude ilk denenen araç oldu. Wikitude GPS verisi ile konum tabanlı bir uygulama geliştirme sistemi sunar ancak bu destek Android tabanlı sistemler için sadece Unity oyun motorunda yoktur. Bu yüzden Wikitude'un konum altyapısını kullanarak uygulama geliştirmek isteniyorsa Unity kullanılmaz. Bu sebeple konum tabanlı uygulama, Android Studio ile geliştirilmiştir. Uygulamanın menü tasarım planı ana hatları ile Şekil 6.4'de görülmektedir.

Uygulamanın ilk aşamasında çeşitli ayarların sağlanması ve ilk konumun alınması için AG işlemi öncesi bir menü oluşturuldu. Android'in TabLayout düzeninde ana aktivite içinde 2 alt menü oluşturuldu.

İlk menüde kamera görüntüsünün alınmaya başlanmasını sağlayacak olan Başlat butonu bulunmaktadır (Şekil 6.5). Bu buton uygulama ilk açıldığında ayarlar menüsünde kaydetme işlemi yapmadan çalışmamaktadır.

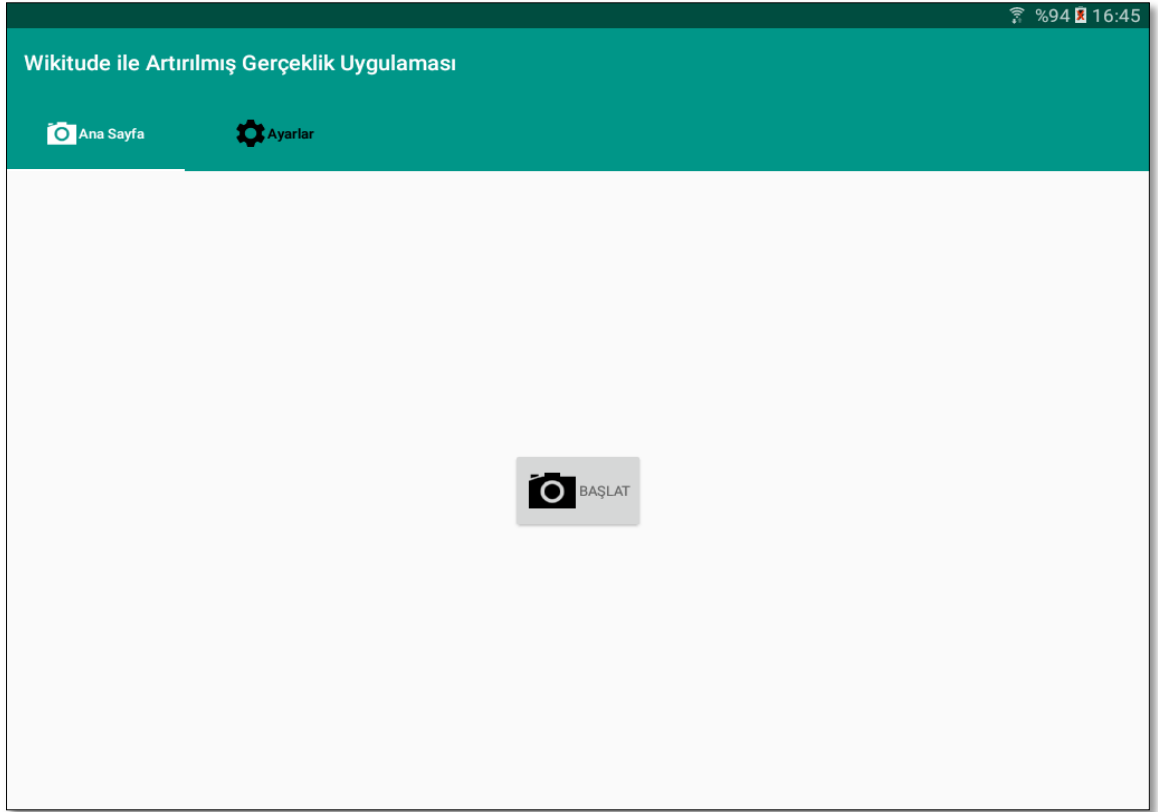


Şekil 6.4. Konum tabanlı AG uygulamasının menü planı.

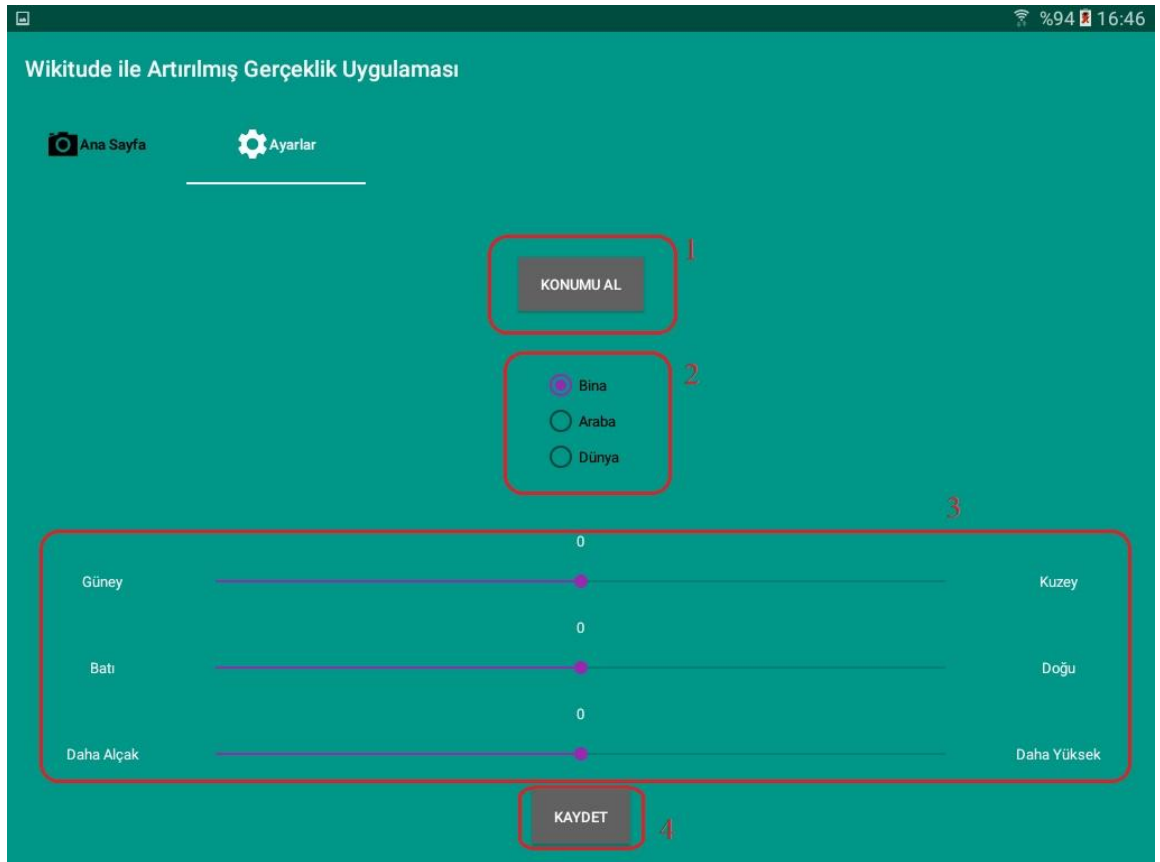
Diğer alt menü ise Şekil 6.6'da görülen Ayarlar menüsüdür. Bu kısımda uygulama başlatılmadan önce, menünün başlığından da anlaşılacağı üzere, çeşitli ayarlar yapılmasını sağlar. Bu alt menüde;

- ilk konumun alınmasını sağlayan "Konumu Al" butonu (1),
- ekranda görüntülenecek farklı modellerin seçimini sağlayacak seçenek butonları (2),
- modelin konumunu kullanıcının konumuna göre belirlemeye yarayan kayan menüler (3),
- tüm yapılan ayarları kaydetmeye yarayan Kaydet butonu (4) bulunmaktadır.

Ana sayfadaki Başlat butonunun çalışması için burada konum alınmalı, diğer ayarlar yapılmalı ve kaydet butonuna tıklanmalıdır.



Şekil 6.5. Wikitude ile geliştirilen konum tabanlı uygulamanın giriş menüsü.

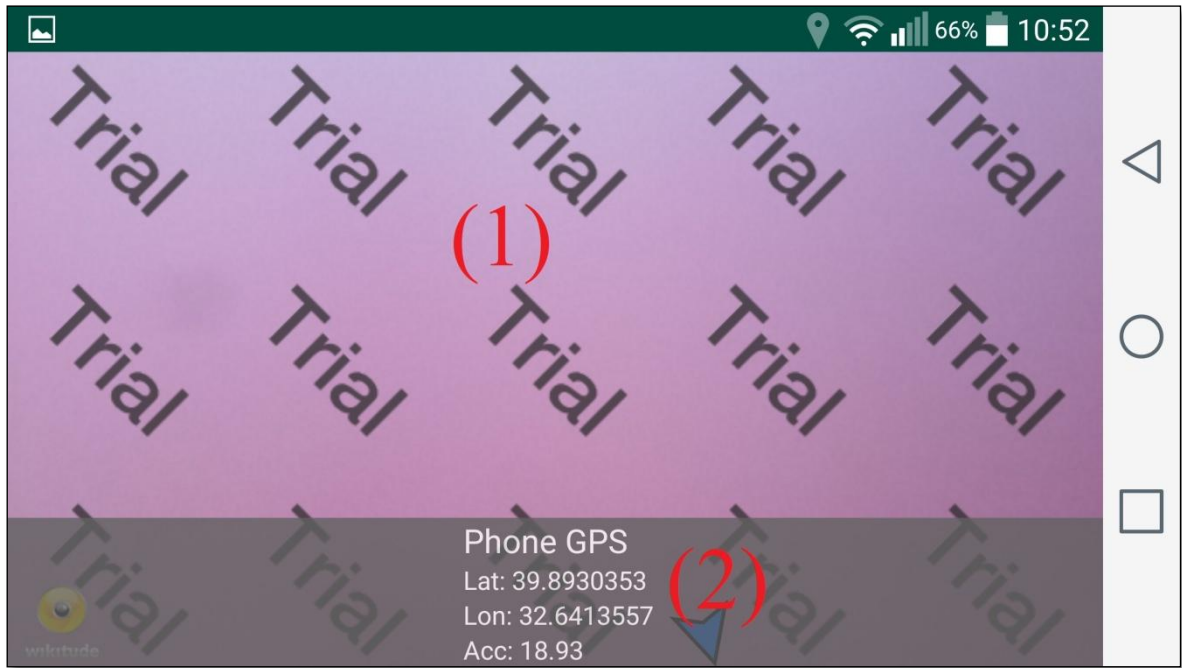


Şekil 6.6. Wikitude ile geliştirilen konum tabanlı uygulamanın ayarlar menüsü.

Ana Aktivite tasarlandıktan ve işlev kazandırıldıktan sonra kamera görüntüsünün alınacağı menü geliştirildi. Buna göre kamera görüntüsü tüm ekranı kaplayacak şekilde olacakken ekranın en alt kısmında ise mobil cihazın konum bilgileri yer alacaktır.

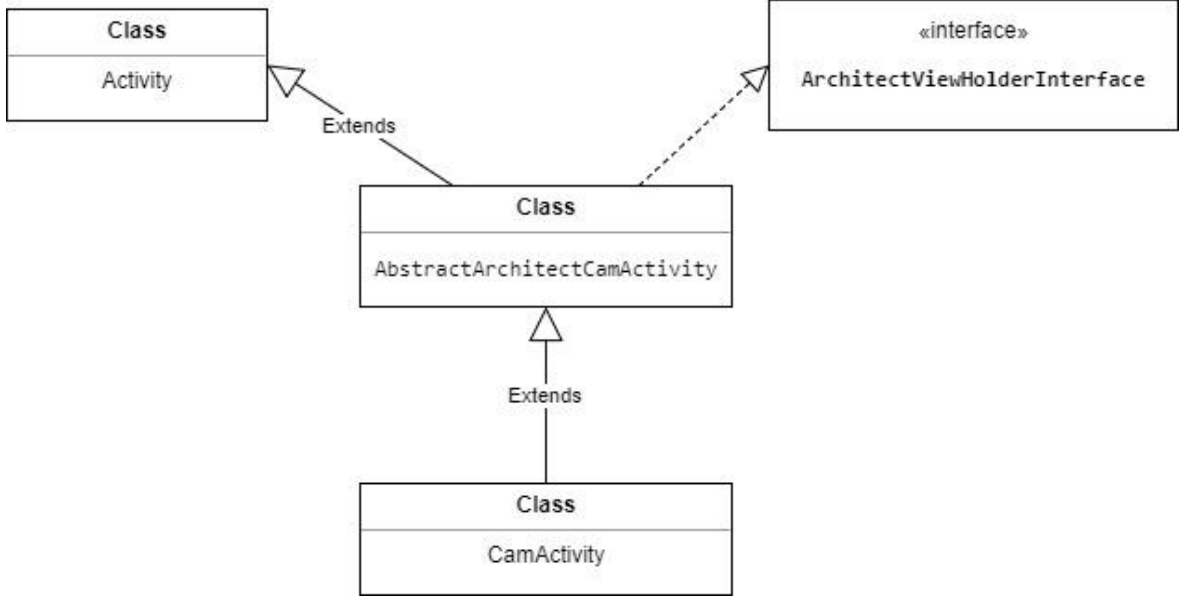
Wikitude ile tasarlanan bir uygulamada kullanılacak modellerin Wikitude 3D Encoder [73] kullanılarak .wtc formatına çevrilmesi gerekmektedir. Bu amaçla Wikitude 3D Encoder indirildi ve tasarlanan hamam modelinin tamamı burada uygun formata çevrilmeye çalışıldı. Modelin çok detaylı olması ve bu uygulamanın boyutu 150 MB'ı aşan modellerde hatalı çalışması [74] sebebiyle dönüştürme işlemi başarısız oldu. Bunun yerine en azından konum tabanlı sistemi denemek adına Wikitude'un kendi modelleri kullanıldı.

Sonraki aşamada kamera görüntüsünün oluşacağı alan tasarlandı (Şekil 6.7). Buna göre Android cihazın ekranında, kameradan alınacak görüntü (1) ve altında kullanıcının enlem ve boylam bilgisi yer alacak (2) şekilde bir görüntü oluştu .



Şekil 6.7. Wikitude ile geliştirilen konum tabanlı uygulamanın kamera görüntüsü alanı.

Tüm tasarım işlemleri tamamlandıktan sonra kamera görüntüsünün alınması ve modelin üstüne bindirilmesi için Wikitude'un sınıflarından da yararlanılarak yeni sınıflar oluşturuldu. Sınıflar arası ilişki Şekil 6.8'de görülmektedir.



Şekil 6. 8. Uygulamanın AG kısmının sınıfları arası ilişkiler.

Burada ArchitectViewHolderInterface adından da anlaşılacağı üzere bir arayüzdür ve AbstractArchitectCamActivity sınıfı tarafından gerçekleştirilir. AbstractArchitectCamActivity en önemli sınıftır ve Activity sınıfından türer. Bu sınıf içerisinde bir aktivitenin tüm yaşam döngüsü gerçekleştirilir, kullanıcının görüntülediği arayüz katmanları atanır, Android cihazın konumu alınır ve daha sonra bahsedilecek olan ve modelin dünya üzerinde bir konuma yerleştirilmesini sağlayan javascript kodu çağırılır. CamActiviy sınıfın ise AbstractArchitectCamActivity sınıfından türer. Bu sınıf içinde genel olarak üst sınıfta tanımlanan işlevler bu uygulamaya özel hale getirilir. Çağırılacak javascript kodu ve java kodu arasındaki iletişimi sağlayan ara kodun ismi ve yeri, uygulamanın başlığı, kullanılacak düzen, Wikitude'un lisans anahtarı burada belirlenir.

Görüntü üzerine bindirilecek model verilerini tutmak için bir sınıf kodu yazıldı. MyObjectValues adlı sınıf;

- nesnenin türü,
- enlem,
- boylam,
- yükseklik,
- göreceli kuzey uzaklığı,
- göreceli doğu uzaklığı gibi değerleri tutmaktadır.

Sahne de gözükecek sadece bir model olacağı ve bu modelin de verilerine bir çok yerden erişileceği için verileri tutan sınıftan sadece tek bir nesne oluşturulmasını sağlamak ve daha fazla nesneye izin vermemek adına Singleton kodu yazıldı. Bu kod aynı zamanda uygulamanın farklı yerlerinden verilere erişilmek veya değiştirilmek istendiğinde tutarlılık sağlayacaktır.

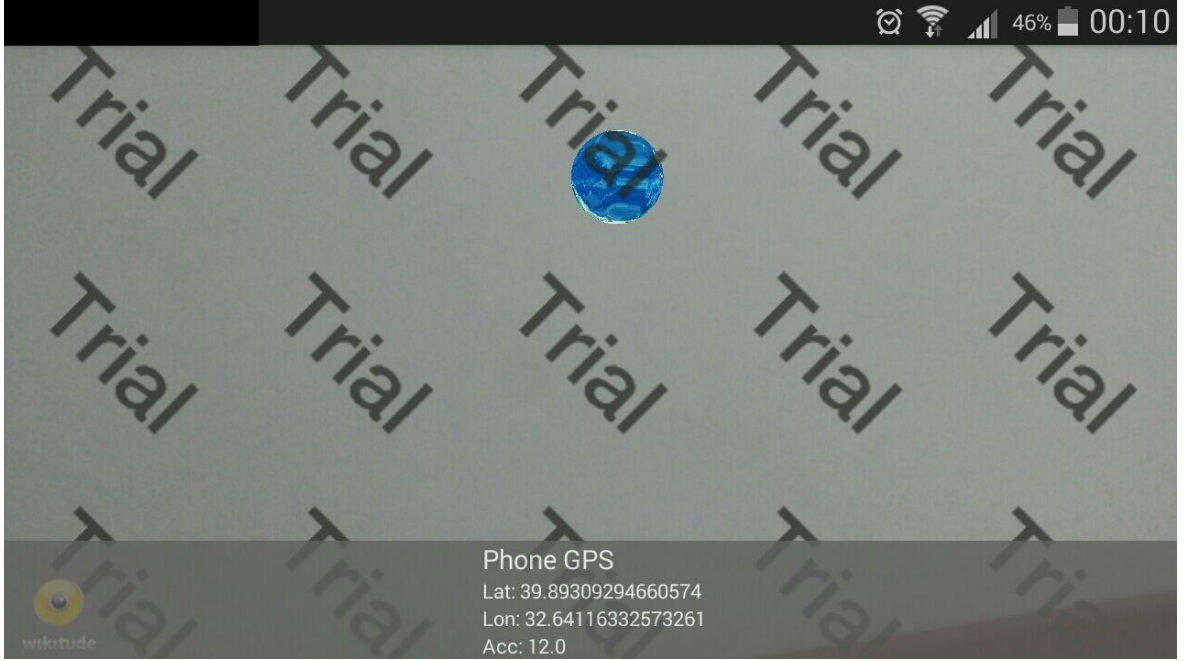
Daha önce de belirtildiği üzere Wikitude'un Unity 3D'de konum tabanlı AG uygulamaları için bir çözümü yoktur. Zaten bu sebeple uygulama geliştirme ortamı olarak Android Studio tercih edilmişti. Wikitude'un konum tabanlı çözümü Android Studio için var olsa da Wikitude, Android'in yerel geliştirme dili olan Java ile konum tabanlı geliştirmeye izin vermez kendi yerel dili olan javascript dili ile geliştirmeye zorlar. Bu sebeple cihazın konum bilgisi Java ile elde edilse de, bu konum bilgilerinin kullanılması, modelin doğru konuma yerleştirilmesi işlemi javascript ile yapılmaktadır.

Bu amaçla Wikitude'un örnek kodundan da hareketle modeli konuma yerleştirecek kod yazıldı. Kod nesne parametrelerini almasının yanında, kamera ekranına giriş yapılırken çeşitli uyarılar göstermek ve modeli kullanıcının konumuna göre yerleştirmek gibi işlevleri yerine getirmektedir.

Tamamlanan uygulama aşağıdaki gibi çalışmaktadır:

- Ayarlar ekranından ilk konum, modeli, modelin göreceli olarak uzaklığını belirlenir ve kaydedilir.
- Ana ekrandan Başlat tuşuna tıklayarak kamera görüntüsünün alındığı aktiviteye geçiş yapılır.
- Nesne kullanıcıdan belirli uzaklıkta oluşturulur.
- Kullanıcının konumu cihazdan alınan bilgiler ile sürekli güncellenir.

Çalışan uygulama Şekil 6.9'da görülebilmektedir. Sonuç olarak uygulama geliştirildikten sonra Wikitude adına varılan olumlu tek sonuç konum tabanlı yazılım geliştirmeye yerel olarak destek vermesi oldu. Bu geliştirme sürecinde olumsuz olarak göze çarpan yanlar ise arkaplandaki "Trial" filigranı, yazılımın çoğunun Java ile geliştirilmesine rağmen konum tabanlı sistemin ancak javascriptle geliştirilebilmesi, Wikitude'un uygulamada kullanılması planlanan 3B modellerin kendi formatına çevrilmesini zorunlu kılması ve bu çevirme işlemini yapan 3D Encoder'ın tutarlı çalışmaması gibi sebepler sıralanabilir.



Şekil 6.9. Wikitude ile geliştirilen konum tabanlı AG uygulamasından bir görüntü.

6.3.2. Vuforia ile Geliştirilen Uygulama

Bir önceki bölümde sıralanan sebeplerle projenin Wikitude üzerinde inşa edilmemesine karar verildi. Bu amaçla Bölüm 5'te incelenen diğer yazılım geliştirme araçlarından biri olan Vuforia ile birlikte Unity 3D oyun motoru ile konum tabanlı uygulama yapılmasına karar verildi.

Bunun için öncelikle bir giriş menüsü oluşturulması planlandı. Giriş menüsü yapılamasındaki amaç kullanıcının ilk konumunun en doğru şekilde belirlemektir. Günümüz mobil cihazlarında bazı durumlarda GPS konumu elde edilemediğinde en son elde edilen konumu kullanmaktadır. Bazı durumlarda da yeterince güçlü olmayan GPS sinyali normalde bulunan konumunda oldukça uzak konumları işaret edebilmektedir. Bu gibi durumların önüne geçilebilmesi adına cihazın, uygulama başlamadan önce belli sayıda GPS sinyali güncellemesi alarak onu kaydetmesi planlandı.

Yukarıda anlatılan sebeplerle bir giriş ekranı tasarımı yapıldı. Öncelikle Unity'in kullanıcı arayüzü özelliklerinden biri olan Canvas kullanılarak butonlar ve metinler oluşturuldu. Arayüz Şekil 6.10'da görülebildiği gibi; kullanıcıya uygulamanın durumu hakkında bilgi veren metin kısmı (1), uygulamanın, yeterince sinyal güncellemesi aldığı anda kamera

görüntüsü alınacak sahneye geçmesini sağlayan giriş butonu (2) ve uygulamadan çıkışı sağlayan çıkış butonu olarak tasarlanmıştır.



Şekil 6.10. Vuforia ile geliştirilen uygulamanın giriş menüsü.

Giriş menüsünün tasarım aşaması tamamlandıktan sonra işlevselliği sağlayacak olan kod kısmına geçildi. Vuforia'nın yerel olarak konum tabanlı AG sistemine bir çözümü olmadığı için cihazdan GPS sinyalinin alınması işlevi tamamen Unity'nin sağladığı olanaklarla gerçekleştirilmesine karar verildi. Bu amaçla öncelikle butonların işlevselliğini kazanması için çalışma yapıldı.

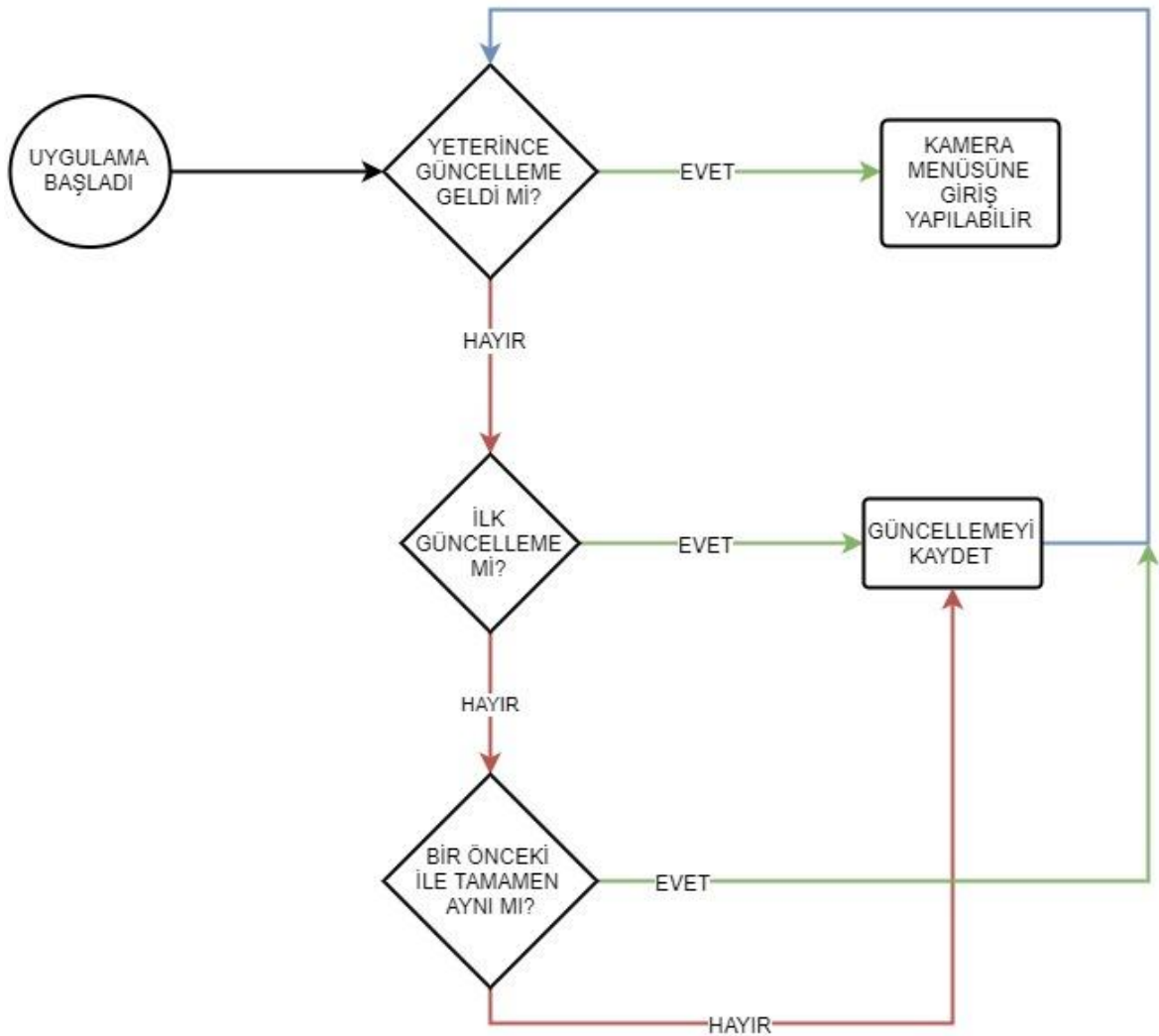


Şekil 6.11. Başlat butonunun çalışması.

Başlat butonun çalışması için öncelikle güncelleme sayısının yeterli olup olmadığı test edildi. Şekil 6.11'de görüldüğü üzere Başlat butonuna basıldığında güncelleme sayısını konum bilgisinin alındığı kod yardımı ile öğrenildi. Eğer yeterli sayıda güncelleme

geldiyse kamera görüntüsünün alınacağı sahneye geçildi aksi takdirde aynı sahnede kalındı. Çıkış butonunun tek görevi ise adından da anlaşılacağı üzere uygulamadan çıkmak olarak ayarlandı.

Sonraki adımda sistemin GPS sinyalleri alması için gereken geliştirmeler yapıldı. Şekil 6.12'de konum güncelleme işleminin akış şeması görülebilmektedir. Buna göre uygulama başlamasından itibaren GPS sinyalleri elde edildi. Her sinyal güncellemesinde bir önceki güncelleme ile değerleri karşılaştırıldı ve aynı değilse kaydedildi. Bu sayede tüm sinyallerin farklı olması sağlandı. Yeterli sayıda güncelleme geldiğinde uygulama kamera menüsüne girişe izin verdi. Bu sırada da kaydedilen güncellemelerin ortalaması alındı ve bu değerler sıradaki sahneye aktarıldı.



Şekil 6.12. Konum güncelleme işleminin akış şeması.



Şekil 6.13. Yeterli sayıda güncelleme alındıktan sonra giriş ekranı.

Bu aşamadan sonra AG'nin uygulanacağı sahne tasarlandı. Bunun için öncelikle Unity'nin Canvas özelliği kullanılarak arayüz üzerinde GPS konumunun ve cismin hesaplanan uzaklığının yazılacağı metinler oluşturuldu (Şekil 6.14).



Şekil 6.14. Kamera sahnesinin tasarımı.

Tasarım aşaması tamamlandıktan sonra Vuforia'nın Unity için hazırlanmış paketi indirildi ve sahneye aktarıldı. Sahnede varsayılan olarak oluşan kamera silindi ve paketin içinden gelen kamera prefabı sahneye aktarıldı. Prefab üzerindeki World Center Mode ayarı üzerinde "Device Tracking" seçeneği seçilir. Bu ayarlama mobil cihazı, geliştirilen ortamın merkezi haline getirir. Yine kamera prefabı üzerindeki "Enable device pose tracking" seçeneğinin işaretlenmesi ile cihazın sağa, sola, yukarı ve aşağı dönüşleri sahneye aynen yansıtılabilir.

Sonraki aşamada giriş sahnesinde olduğu gibi bu sahnede için de GPS verilerinin alınması için kod yazıldı. GPS verilerinin düzgün bir şekilde alındığı test edildikten sonra deneme amaçlı olarak hamam modelinin bir kısmı sahneye kamera da görünecek şekilde sahneye yerleştirildi.



Şekil 6. 15. Vuforia ile geliştirilen uygulamadan bir görüntü.

Bu aşamada GPS sinyallerinin alımı, uzaklığın hesaplanması, modelin sahnede istenilen alana yerleştirilmesi gibi bu projenin en önemli kısımlarının Vuforia'dan bağımsız olarak Unity ile yapılabileceği görüldü.

6.4. Geliştirilen Asıl Mobil Yerde Artırılmış Gerçeklik Uygulaması

Bölüm 6.3'te geliştirilen uygulamalarda yazılım geliştirme araçları kullanılmış ancak istenilen sonuçlara ulaşılamamıştır. Özellikle Vuforia ile geliştirilen uygulamada, gerekliliklerden çoğunun Unity tarafından karşılanabileceği görülmüştür. Bu amaçla geliştirilen uygulamalardan elde edilen tecrübeyle bu tez çalışmasında ortaya çıkması planlanan projenin sadece Unity oyun motoru kullanılarak geliştirilmesine karar verilmiştir. Bu amaçla bir önceki bölümdeki uygulamalarda olduğu gibi Giriş Sahnesi ve Ana Sahne olmak üzere 2 sahne oluşturulmuştur. Bu bölümde uygulama sahnelere ayrılarak detaylı bir biçimde anlatılacaktır.

Projenin detaylarına giriş yapmadan önce Unity'nin kod sisteminin anlatılmasında fayda vardır. Unity birden fazla dilde kod yazımını desteklemektedir. Bu diller C# ve Javascript'tir [75]. Bu projede geliştirilecek tüm kodlar C# kullanılarak yazılmıştır.

Unity'de bir kodun çalışması için, o kodun sahnede oluşturulmuş bir nesnenin bileşeni olması gerekmektedir. Oluşturulan kodun bir nesneye atanabilmesi için ise Unity'nin gömülü sınıfı olan MonoBehaviour'dan türemesi gerekmektedir.

MonoBehaviour sınıfından türeyen bir sınıfta da 2 önemli ana metod çalışmaktadır [75]. Bunlardan ilki Update metodudur. Update metodu uygulama çalışırken sürekli olarak çağrılır. Bu metod ile çalışma zamanındaki tüm aksiyonlar kontrol edilebilir ve kullanıcı girdilerine karşı cevaplar döndürülebilir. Diğer metod ise Start metodudur. Start metodu adından da anlaşılacağı üzere uygulama başlarken bir defa çalışır. Start metodunun çalışması bitmeden Update metodu çalışmaz. Start metodunda tüm ilk atama işlemleri gerçekleştirilip uygulama çalışmaya hazır hale getirilir.

Unity de tüm işlemler bir sahne üzerinde yapılır. Uygulamanın çalışması için gerekli olan tüm nesnelere, modeller ve kodlar sahne üzerinde bulunmalıdır. Yukarıda da bahsedildiği üzere çalışacak olan kodlar bir nesne üzerine aktarılmış olmalıdır. Bu amaçla çoğu zaman ekranda gözükmeyen ancak hiyerarşi de gözükken boş nesnelere oluşturulur ve kodlar bu nesnelere içine atılır. Herhangi bir kod sahne üzerinde herhangi bir nesneye erişmeye çalışabilir. Bunun için kodun erişmek istediği nesne üzerinde bulunduğu nesne değilse bunu nesnenin etiketini veya ismini arayarak yapabilir. Ayrıca çalışma zamanında kod yardımıyla yeni nesnelere oluşturulabilir ya da yok edilebilir.

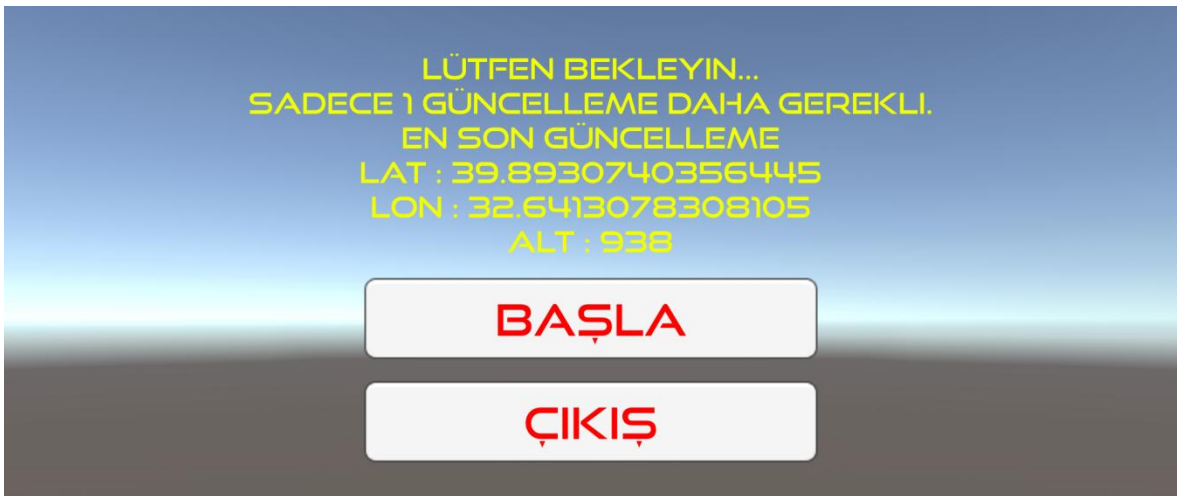
6.4.1. Giriş Sahnesi

Giriş sahnesinin çalışma prensibi ve tasarımı Bölüm 6.3.2'de anlatılan ve Vuforia ile geliştirilen uygulamanınki ile aynı olacak biçimde planlanmıştır. Bu amaçla Unity'nin Canvas özelliği kullanılarak arayüz oluşturulmuş, arayüze bir bilgi metni ve Başla, Çıkış butonları eklenmiştir.

Android cihaz üzerinden ilk konumun alınması ve butonları çalışması adına sırasıyla Location Manager ve Button Manager isimli sınıflar yazılmıştır. İki sınıf da Unity'nin MonoBehaviour sınıfından türetilmiştir. Bu sebeple iki sınıf da Start ve Update metotlarına sahiptir.

Location Manager sınıfının Start fonksiyonu içerisinde cihazın pusulası aktif edildi, ekrandaki metin nesnesine, daha önce kendisine atanan etiketi yardımıyla, erişildi. Daha sonra cihazın konum verisinin açık olup olmadığı test edildi ve açıksa ilk konum verisini almak için deneme yapıldı. Uygulama çalışırken bu işlemleri yaptıktan sonra Location Manager sınıfı için Start fonksiyonun görevi sona erecektir.

Location Manager sınıfının Update fonksiyonunda ise konum alma işlemi devam ettirildi. Her alınan konum verisi, eğer ilk veri değil ise, bir önceki veri ile karşılaştırıldı ancak farklı ise kaydedildi. Bu sayede uygulama başlamadan istenilen sayı kadar farklı konum verisi alınacaktır. Bu işlemler yeterli sayıda konum verisi alınıncaya kadar devam edecek şekilde düzenlendi ve yeterli sayıya ulaşıncaya Button Manager sınıfına diğer sahneye gidilebileceğine dair sinyal gönderildi. Ayrıca Update metodunun içerisinde her güncelleme verisi alındığında ve yeterli sayıya ulaşıldığında ekrandaki metin güncellendi.



Şekil 6.16. Uygulamanın giriş menüsü GPS güncellemesi alırken.

6.4.2. Artırılmış Gerçeklik Sahnesi

Uygulamanın giriş sahnesinde yeterli güncelleme sayısına ulaşıldığında Başlat butonu çalışır hale gelir. Kullanıcı bu butona tıkladığında AG sahnesi açılacak ve kamera görüntüsü alınacaktır. Aynı zamanda Giriş sahnesinde elde edilen konum verilerinin ortalaması bu sahnede modelin yerleştirilmesi için kullanılacak ve kamera görüntüsünün önünde bulunan model ile AG sağlanacaktır.

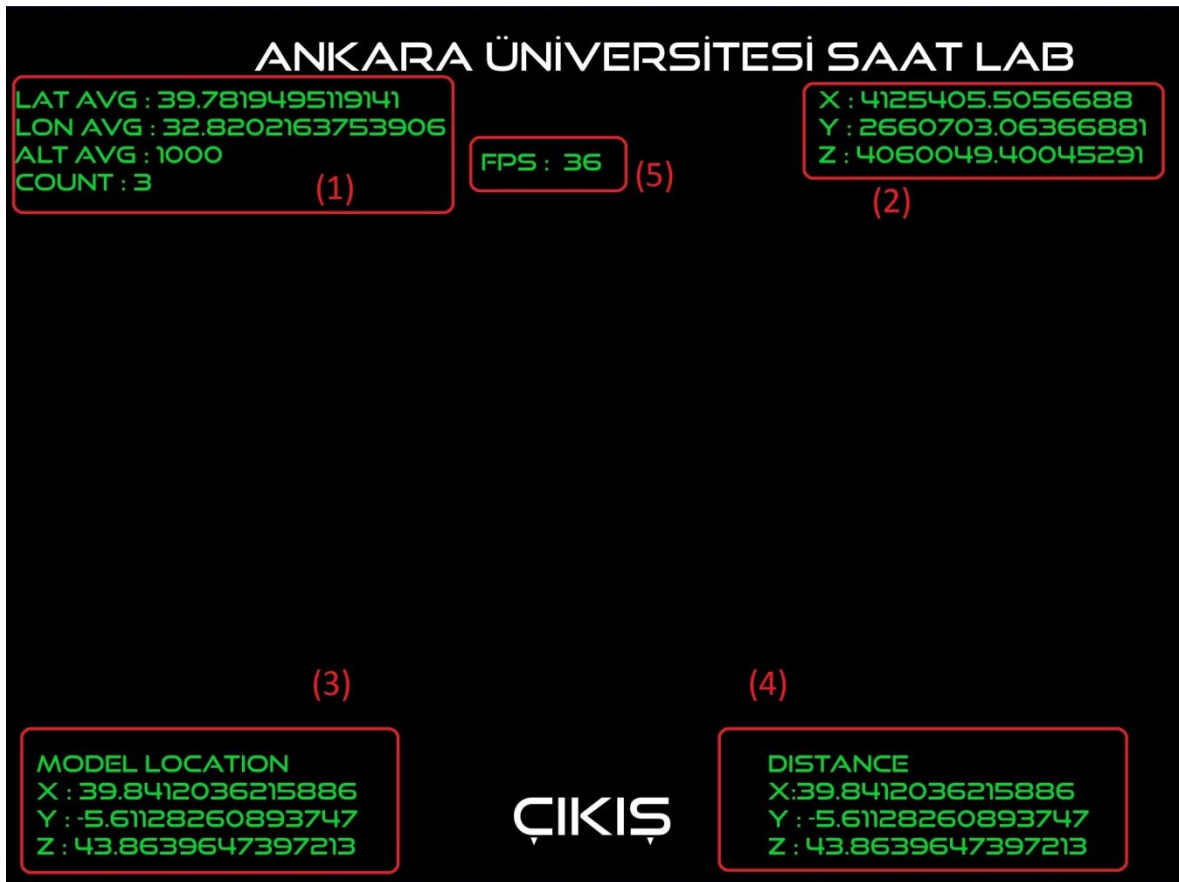
Bu amaçla öncelikle sahnenin arayüz tasarımı yapıldı. Daha önce geliştirilen uygulamalarda olduğu gibi arayüz oluşturmak için Unity'nin Canvas özelliği kullanıldı. Öncelikle ekranın en üst kısmına projenin çalışıldığı laboratuvarının ismi sabit bir metin olacak şekilde yazıldı. Daha sonra ekranın en alt kısmına uygulamadan istenildiği zaman çıkış yapılabilmesi için "Çıkış" metni eklendi. Bu metin tıklanabilir hale gelmesi için Unity'nin buton özelliği bileşen olarak eklendi.

Kullanıcıların çıkış butonuna yanlışlıkla basma durumları göz önüne alınarak Canvas özelliği kullanılarak bir ara menü daha oluşturuldu (Şekil 6.17). Bu ara menü üzerinde bir soru metni ve "Evet" ile "Hayır" olmak üzere toplam üç metin oluşturuldu. Evet-Hayır metinlerinin üzerine yine tıklanabilir olmaları için buton özelliği eklendi.



Şekil 6.17. Çıkış butonuna tıkladığında ön plana gelen ara menü.

Ara menünün tasarımından sonra kamera ekranını asıl bileşenlerinin tasarımına geçildi. Bu amaçla ekranda 5 adet metin nesnesi oluşturuldu (Şekil 6.18). Bu nesneler; kullanıcının enlem boylam ve yükseklik bilgisini içeren GPS verilerinden alınan ortalama konumunu (1), ortalama konumdan hesaplanan kartezyen konumunu (2), sahneye giriş yapıldığında modelin cihaz merkez alınarak yerleştirildiği konumu (3), sürekliği güncellenen verilerle modelin cihaza göre konumunu (4) ve saniyede oluşan kare sayısını (5) göstermesi için ayarlandı. Ayrıca bu metin nesnelerinin kod içerisinde bulunabilmesi ve güncellenebilmesini sağlayabilmek için her biri için etiketler oluşturuldu ve bu etiketlerin ataması yapıldı.



Şekil 6.18. Artırılmış Gerçeklik ekranı tasarımı.

Tüm arayüz bileşenleri tasarlandıktan sonra kameranın arka plana yerleştirilmesi işlemi yapıldı. Bu amaçla üzerine doku olarak cihazın kamerasından gelen görüntünün aktarılacağı ve Unity'nin bir arayüz bileşeni olan RawImage nesnesi oluşturuldu. Nesneye "Arkaplan" adı verildi. Uygulamanın aktarılacağı cihazda nesnenin boyutlarının ayarlanabilmesi için Aspect Ratio Fitter bileşeni nesneye eklendi. En-Boy oranının kontrolünü sağlayan bu bileşenin modu genişlik yüksekliği kontrol edecek şekilde

ayarlandı. Ayrıca Arkaplan nesnesi oluşturulduğunda onun atası olarak Canvas nesnesi oluştu. Bu nesnenin adı diğer oluşan Canvas nesneleri ile karışmaması adına Camera Canvas olarak değiştirildi. Ayrıca kameranın görüntüsünün tüm ekrana yayılmasını sağlamak için Camera Canvas nesnesinin üzerinde bulunan Canvas Scaler özelliğindeki UI Scale Mode bileşeni ekran genişliği ile ölçeklenecek şekilde ayarlandı.

Arkaplan nesnesi daha öncede belirtildiği gibi bir RawImage nesnesidir. RawImage ise Unity'nin UI bileşeninin bir alt parçasıdır. Unity'e de tüm UI bileşenleri ekranda her şeyin önünde gösterilir. Bu uygulamada istediğimiz bunun tam tersidir. Yani kamera görüntüsü her zaman nesnelere arkasında kalmalıdır ki nesnelere görülebilsin. Bu amaçla; Arkaplan nesnesi üzerine cihazdan alınan kamera görüntüsünü iletecek olan arkaplan kamerası ve tüm sahnede olanları görüntüleyecek olan ana kamera ile birlikte 2 adet kamera sahnede oluşturulmuştur.

Unity'de geliştirilen uygulamada 2 kamera olduğunda hangisinin neleri göstereceği ve hangisinin üstün olacağını ayarlanması gerekir. Bu doğrultuda öncelikle ana kamera arkaplan kamerasında üstün hale getirildi. Bu noktada 2 kameranın farklı işlevleri yerine getirebilmesi için her ikisinde de farklı olarak ayarlanacak olan Clear Flags ve Culling Mask özelliklerinden bahsetmekte fayda vardır.

Clear Flags özelliği her karede bir önceki kareden kalan hangi özelliklerin temizleneceğini belirtir. Clear Flags özelliğinde seçenekler; sadece derinliği olan nesnelere bir sonraki karede temizlenip yeniden çizileceği anlamına gelen Depth Only, kameranın görüş açısının dışında kalan her şeyi belirli bir renge çizen Solid Color, yine kameranın görüş açısının dışında kalan her şeyi gökyüzü olarak gösteren Skybox ve çizilen her karenin temizlenmeden önceki karenin üzerine çizilmesini sağlayan Don't Clear'dır. Ana kamera Clear Flags özelliği sadece derinliği olan nesnelere ekranda göstermesi için ayarlandı. Arkaplan kamerasının görevi sadece video görüntüsünü ve arayüz bileşenlerini göstermek olduğu için bu özelliği Solid Color olarak değiştirildi.

Culling Mask özelliği ise kameranın görüntüsünde katmanlardan hangilerinin gözükmeyeceğini belirler. Daha önce de bahsedildiği gibi arkaplan kamerasının tüm amacı arayüz bileşenlerini ve video görüntüsünü ekran üzerinde göstermektir. Bu yüzden onun Culling Mask özelliğinde sadece UI katmanı olarak seçildi. Arayüz dışındaki kalan her nesneyi göstermek de ana kameranın görevidir. Bu yüzden onun Culling Mask özelliği ise UI dışında tüm katmanlar olacak şekilde seçildi.

Bu aşamalardan video görüntüsünün üzerinde oluşacağı nesne olan Arkaplan nesnesinin atası Camera Canvas nesnesin sadece tek bir kamerada gözükmesi adına Render Mode özelliği Screen Space Camera haline getirildi.

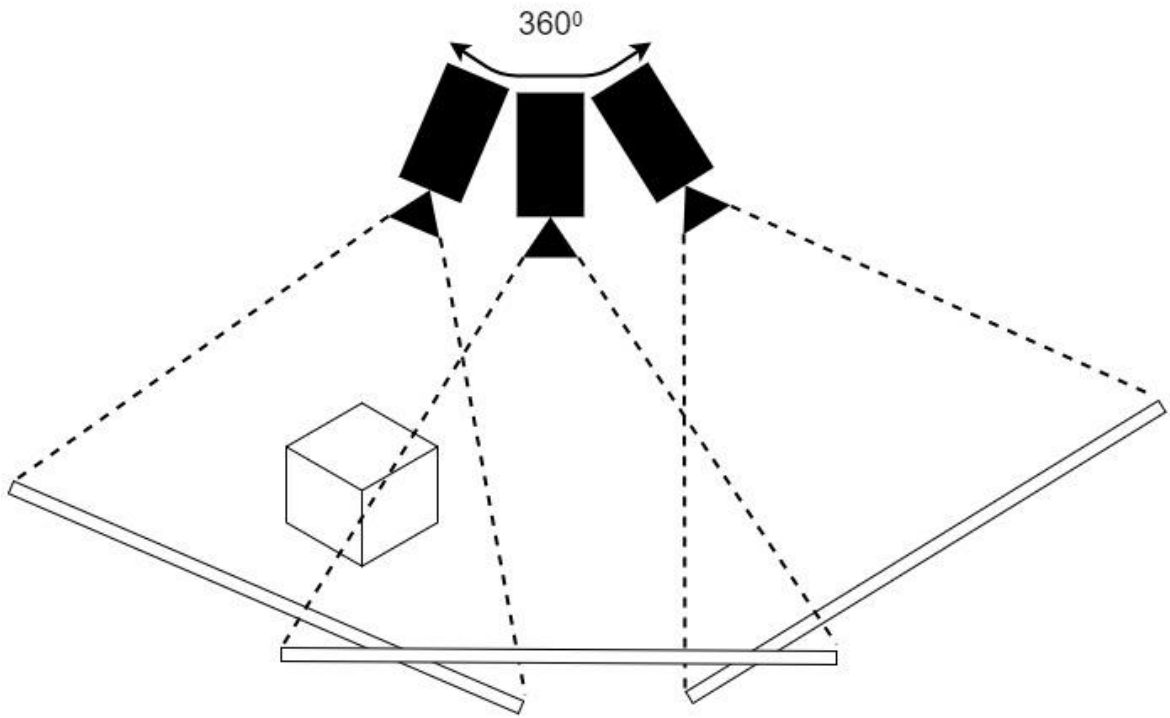
Bu işlemlerden sonra kamera görüntüsünün alınıp Arkaplan nesnesinin üzerine doku olarak atanmasını sağlayacak Background Manager sınıfı yazıldı. Sınıf içerisindeki Start fonksiyonunda öncelikle RawImage türünde bir nesne olan Arkaplan nesnesi bulundu. Unity'de kamera görüntüsünü almaya yarayan sınıf olan Web Cam Texture sınıfı oluşturuldu. Bu sınıftan gelen video görüntüleri Arkaplan nesnesinin dokusu olarak atandı ve video alımı başlatıldı. Update metodunda ise Web Cam Texture nesnesinden gelen video akımının en boy oranına göre Arkaplan nesnesinin en boy oranı sürekli güncellenedi.

Background Manager sınıfı sadece Arkaplanı yönettiği ve başka bir işlevi yerine getirmediği için Arkaplan nesnesinin içine aktarıldı ve kamera görüntüsünün alınıp alınmadığı test edildi. Kamera görüntüsünün başarılı bir şekilde alındığı görüldü (Şekil 6.19).



Şekil 6.19. Kamera görüntüsünün alınması.

Kamera görüntüsünün alınmasından sonra sahneye cihazın jiroskopundan alınan veriler ile ana kameranın dönüşlerinin senkronizasyonu işlemine geçildi. Bu aşamada sahneye 3B bir nesne aktarılsa bile cihaz döndürüldüğünde nesnenin cihazda görülen konumu asla değişmemektedir. Cihazın gerçek dünyada döndürülmesi veya bakış açısının değiştirilmesi ile sahnede bulunan ana kameranın yön değişimleri aynı olmalıdır ki AG hissi doğru olarak verilsin.



Şekil 6.20. Gyro Manager sınıfının görevi.

Gyro Manager sınıfının amacı kameranın ve video görüntüsünün cihazın hareketlerine göre sahnede dönmesini sağlamaktır (Şekil 6.20). Bu amaçla Gyro Manager sınıfı, Monobehavior sınıfından türetilerek yazıldı. Öncelikle sınıfın Start fonksiyonu içerisinde yüklendiği cihazın jiroskopu olup olmadığı denetlendi. Eğer jiroskop cihaz tarafından destekleniyorsa Unity'nin Input sınıfından gyro bilgisi alınır. Ayrıca yine Start fonksiyonunda kameranın dönüş işlemlerini daha rahat yapabilmek için dinamik olarak boş bir nesne oluşturulur ve ana kamera bu nesnenin çocuğu haline getirilir. Daha sonra Update fonksiyonu içerisinde jiroskoptan alınan bilgiler kameranın dönüş açısı olarak ana kamera nesnesine aktarıldı. Update fonksiyonu sürekli olarak çağırıldığı için dönüş bilgileri de sürekli olarak güncellenecek ve kameranın bakışı açısı doğru olarak ayarlanabilecektir. Son işlem olarak da Gyro Manager sınıfı ana kameraya aktarıldı.

Bu işlemden sonra ana ekrandaki butonlar için Button Manager sınıfı oluşturuldu. Daha önce buton özelliği eklenmiş olan "Çıkış", "Evet" ve "Hayır" metinleri için tıklanmaları halinde çalışacak birer fonksiyon yazıldı. Buna göre uygulamanın kamera sahnesi ilk açıldığında normalde çıkış butonuna tıklanınca görünmesi gereken menü görünmez hale getirildi. Çıkış butonuna tıkladığında ise bu menü görünür ve etkileşilir hale getirildi. Evet butonuna tıkladığında çıkışın tamamlanması hayır butonuna tıkladığında ise sahneye geri dönülmesi için metotlar yazıldı.

Bundan sonraki aşamada konum bilgilerinin saklanması ve erişiminin düzenlenmesi için Location sınıfı yazıldı. Location sınıfı bu projedeki Monobehaviour sınıfından türemeyen sınıflardan birisidir. Location sınıfının tüm amacı enlem, boylam ve yükseklik bilgilerinin uygun şekilde ve birbirine bağlı olarak tutulmasını sağlamaktır. Ayrıca bu sınıf enlem, boylam ve yükseklik verileri, eklenen get ve set metotları ile dışarıdan direk müdahaleye kapatılmıştır.

Çizelge 6.1. Location sınıfının alanları ve metotları.

Location
- latitude : double - longitude : double - altitude : double
+ Location () : + Location (double, double, double) : + getLat () : double + setLat (double) : + getLon () : double + setLon (double) : + getAlt () : double + setAlt (double) :

Location sınıfı ile enlem, boylam ve yükseklik verileri tutulduktan sonra bu verilerin Unity'nin sahnesine uygun olması ve dünya üzerinde hareketlerin doğru hesaplanması için çevrileceği kartezyen koordinatları tutacak olan Cartesian Coordinate sınıfı yazıldı. Bu sınıf da tıpkı Location sınıfı gibi tüm amacı kartezyen X Y Z koordinat verilerini tutmak ve düzenlemektir.

Çizelge 6.2. Cartesian Coordinate sınıfının alanları ve metotları.

Cartesian Coordinate
- X : double
- Y : double
- Z : double
+ CartesianCoordinate () :
+ CartesianCoordinate (double, double, double) :
+ getX () : double
+ setX (double) :
+ getY () : double
+ setY (double) :
+ getZ () : double
+ setZ (double) :

Bu işlemlerden sonra yine Monobehavior sınıfından türemeyen bir sınıf olan Converter sınıfı yazıldı. Bu sınıfın amacı ise Bölüm 3.2'de anlatılan koordinat sistemi dönüşümlerini uygulamaktır. Sınıfın içerisinde convert adında tek metot bulunur. Bu metot aldığı enlem boylam ve yükseklik verilerini daha önce anlatılan yöntemle göre dönüştürür ve geriye 3 boyutlu bir dizi döndürür.

Uygulamaya koordinat sistemi dönüşümü işlevi de kazandırıldıktan sonra modeli ilk konumuna yerleştirecek olan ModelReplacer sınıfı yazıldı. Monobehaviour sınıfından türetilen sınıfın Start fonksiyonunda ekranda görülecek olan ve modelin yerini ve yakınlık uzaklık verilerini gösterecek olan metinlerin nesnelere ve modelin kendisine etiketleri yardımıyla erişildi. Bu sınıf modelin yerleştirileceği enlem ve boylam verilerini aldıktan sonra Converter sınıfına modelin kartezyen koordinatlarını hesaplatır. Daha sonra kullanıcının girişte aldığı ortalama ilk konum bilgileri yardımıyla modelin sahne üzerindeki yerini hesaplar ve modeli oraya yerleştirir. Bu işlemi Update fonksiyonu içerisinde yaparsa da bir kez yapar.. Ancak yine Update fonksiyonu içerisinde kullanıcının kartezyen konumunu sürekli olarak alır ve modele olan uzaklığını hesaplar. Bulduğu değerleri ekranın sağ alt kısmında yer alan metin nesnesine yerleştirir.

Kameranın sahne üzerinde hareketini sağlamak için Movement Manager sınıfı yazıldı. Bu sınıfın için Update fonksiyonunda bir defaya mahsus olmak üzere, daha önce çalışma zamanında kameranın atası olacak şekilde oluşturulan nesneyi bulur. Bu sınıf Update fonksiyonunda başka hiçbir işlem yapmaz sadece kendisine ait statik Move metodu ile daha sonra açıklanacak olan Location Manager sınıfının kamera nesnesinin yerinin güncelleyebilmesini sağlar. Bu yer değiştirme işlemi atlamalar olmaması adına iki konum bilgisi arasında yavaş yavaş geçilerek yapılır.

Bu işlemden sonra konum verilerinin alınmasını yönetecek olan Location Manager sınıfı oluşturuldu. Bu sınıf içerisinde Start fonksiyonunda öncelikle ekrandaki GPS konumlarının yazılacağı metinlerin nesnelere, etiketleri yardımıyla bulundu. Daha sonra aynı metot içerisinde cihazın pusulası aktif hale getirildi ve ilk konum alma işlemi yapıldı. Sınıfın Update fonksiyonunda ise sürekli olarak konum isteğinde bulunuldu.

Her konum isteği çalıştığında, Şekil 6.12'deki çalışma prensibine benzer olarak aşağı işlemler sırası ile yapıldı.

1. Gelen enlem ve boylam değerleri bir önceki değerlerle aynı ise hiç bir şey yapma.
2. Gelen enlem ve boylam değerleri bir önceki değerlerden farklı ise dizideki sıradaki yere kaydet.
3. Bölüm 3'de anlatılan kayan pencere yöntemi ile son 10 verinin ortalamasını al.
4. Alınan ortalamayı Converter metoduna gönder ve kartezyen koordinatları al.
5. Alınan GPS ve kartezyen koordinat verilerini ekrandaki metinlere yaz.
6. Movement Manager sınıfının Move metodunu çalıştırarak kameranın yerini güncelle.

En son aşama olarak ise ekrandaki saniye başına kare sayısını hesaplayacak olan FPS Calculator sınıfı yazıldı. Bu sınıf Start fonksiyonunda ekranda bulunan FPS değerlerinin yazılacağı metnin nesnesini bulur ve Update fonksiyonunda sürekli olarak güncel saniye başına kare sayısını bu metne yazar.

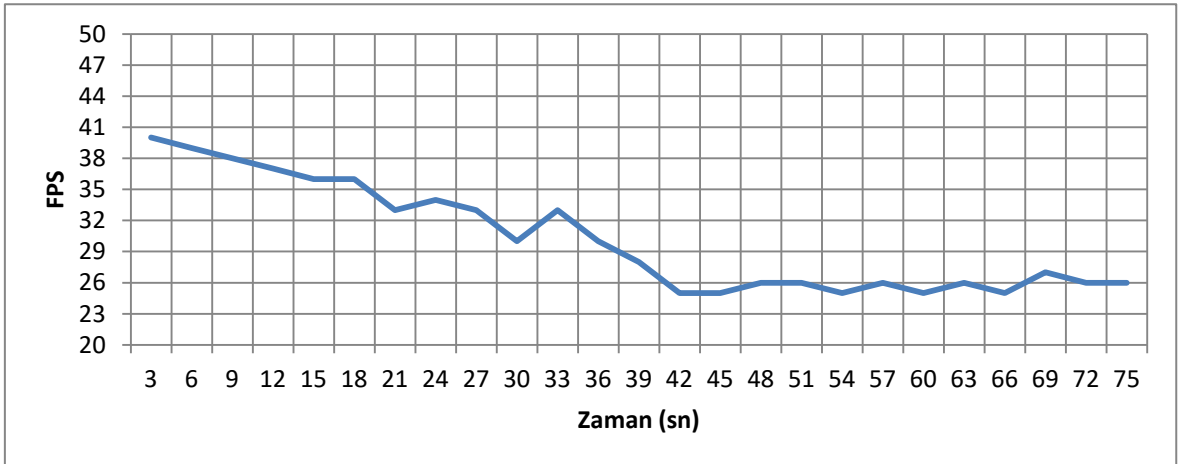
Bu işlemlerden sonra kodlar sahne üzerindeki nesnelere aktarıldı. Tüm kod işlemleri tamamlandıktan sonra deneme amaçlı olarak hamam modelinin bir kısmı sahneye aktarıldı. Modelin boyutları ayarlandıktan sonra Model Replacer sınıfı tarafından bulunabilmesi için modele etiket verildi.

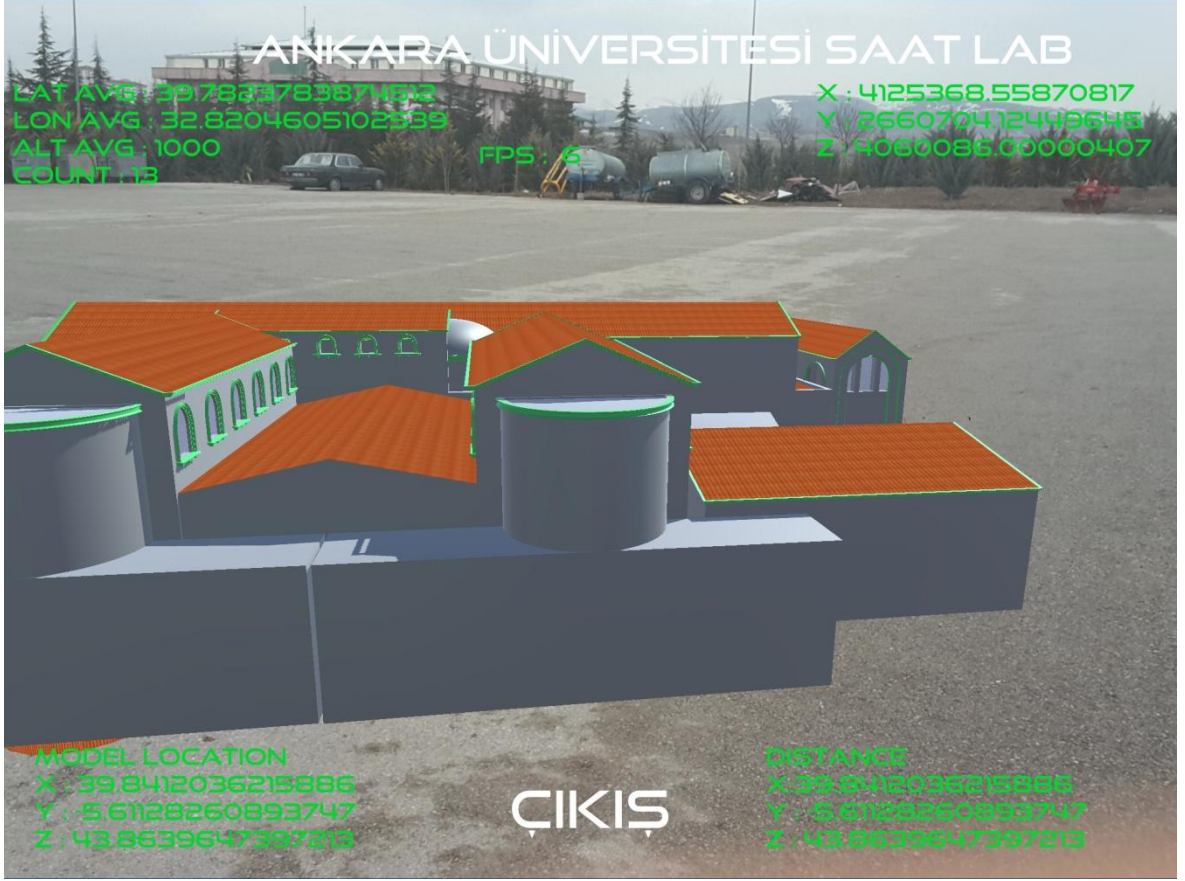


Şekil 6.21. Çalışan uygulamadan alınan görüntü.

Son olarak da Android tabanlı cihazlarda çalışabilmesi için gerekli olan id ve minimum API gibi ayarlar düzenlendi. Uygulama Samsung Galaxy Tab S2'ye aktarılarak denendi. Modelin bir parçası eklenerek yapılan denemelerden bir görüntü Şekil 6.21'de görülebilir. Modelin tek parçası aktarıldıktan sonra saniye başına kare sayısı değişimleri gözlemlendi. Modelin doğru konuma yerleştirildiği görüldü.

Çizelge 6.3. Tek Bina Modeli için saniye başına kare sayısı değişimi.

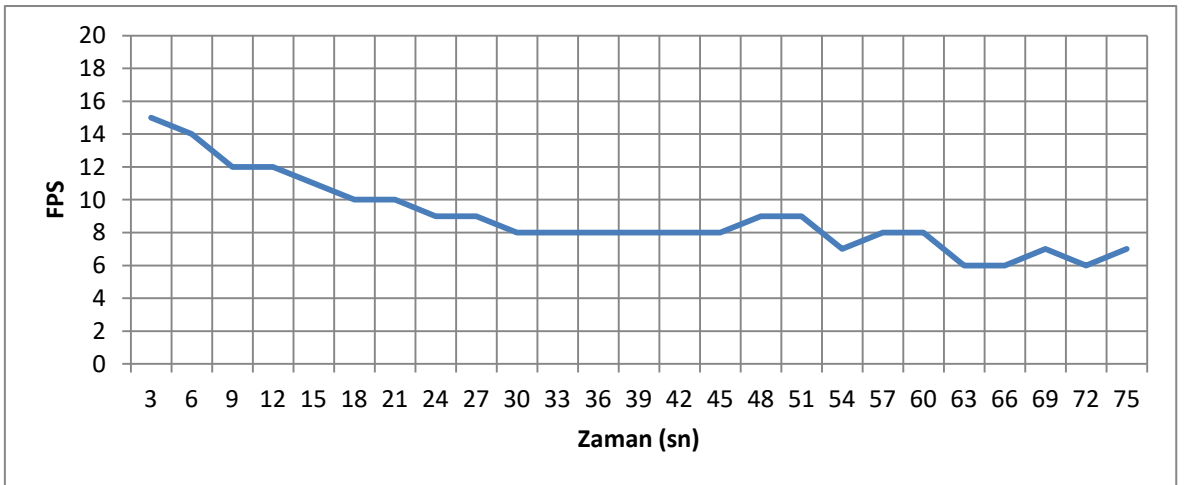




Şekil 6. 22. Modelin tamamı eklenmiş uygulamadan alınan bir görüntü.

Sonraki aşamada ise Roma Hamamı modelinin tamamı uygulamaya eklendi (Şekil 6.22). Uygulama bu modeli konumda gösterecek şekilde ayarlandı. Bu halde uygulama çalışırken saniye başına kare sayısı hesaplandı. Bu model eklendikten sonra uygulamanın boyutunun oldukça arttığı görüldü.

Çizelge 6.4. Hamamın tamamı eklendiğinde saniye başına kare sayısı değişimi.

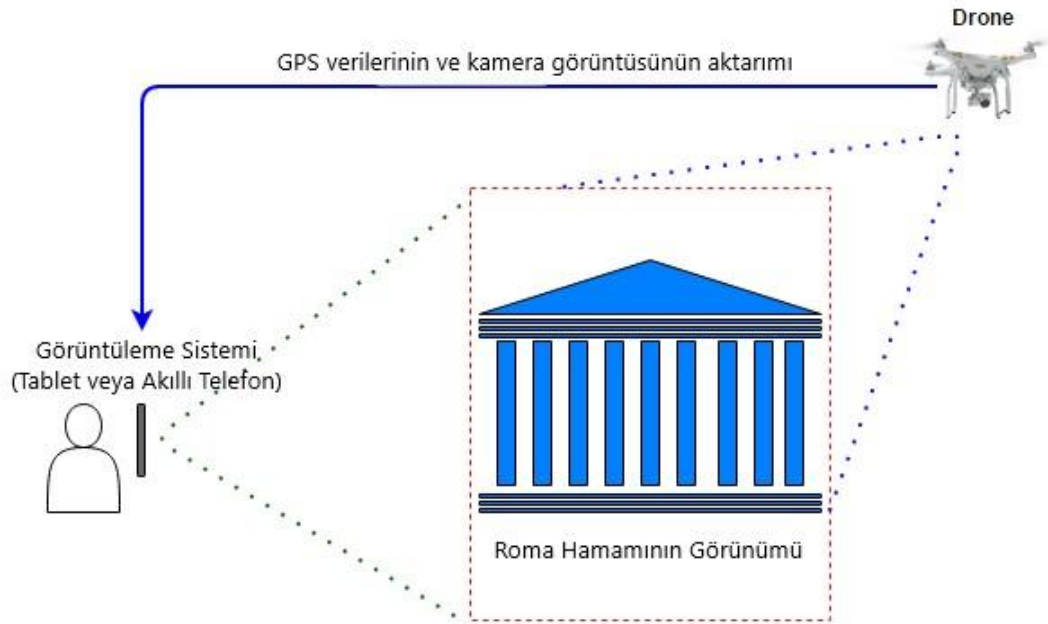


Sonuç olarak saniye başına düşen kare sayısında özellikle doğru konuma yaklaşıldığında düşüşler gözlemlenmiştir. Modelin detaylarının oldukça fazla olması ve arkaplanda yapılan hesaplar bunda etken oluşturmaktadır.

Tüm hesaplamalar ve testler sona erdikten sonra yerinde konum tabanlı mobil artırılmış gerçeklik uygulaması tamamlanmış oldu.

6.5. Uzaktan Artırılmış Gerçeklik Sistemi

Mobil artırılmış gerçeklik sisteminin tamamlanmasından sonra uzaktan artırılmış gerçeklik için bir sistem tasarlandı. Bu sisteme göre kullanıcı harabe halinde bulunan hamam alanı üzerinde uçan dronelerin çektiği görüntüleri alacak ve mobil cihaz üzerinde hamam modelini bu görüntülerin üzerine bindirecek. Bu sayede kültürel miras alanları için oldukça ilgi çekici bir içerik oluşacaktır.



Şekil 6.23. Uzaktan artırılmış gerçeklik sistemi.

Bu sistemde DJI [76] firmasının ürettiği Phantom 3 Pro (Şekil 6.24) modellenen dronenin kullanılmasına karar verilmiştir. Bu dronenin seçiminde etkili olan faktörler; Android cihazlara geliştirme yapılabilmesi için bir SDKsı olması, GPS, 4K (Ultra HD) kalitede video çekebilme özelliği, uçuş sırasında kendisini dengelemesi gibi sıralanabilir. Dronenin ağırlığı yaklaşık 1280 gram iken, maksimum uçuş süresi yaklaşık 23 dakikadır [77].



Şekil 6.24. DJI Phantom 3 Pro drone ve uzaktan kumandası.

Phantom 3 modellenli dronelar, onu yönetmek için tasarlanmış bir uzaktan kumanda ile birlikte gelir. Bu uzaktan kumanda ile droneu havalandırıp indirmek, havadayken yönetmek dışında kamerasının bakış açısını değiştirmek, çeşitli uçuş modları arasında geçiş yapmak ve kalkış konumuna geri döndürmek gibi bir çok eylem gerçekleştirilebilir. Ayrıca uzaktan kumanda ile dronedan alınan canlı video görüntüsü ona bağlı mobil cihaza iletilir.

6.5.1. Geliştirilen Uygulama

DJI, geliştiriciler için üç farklı SDK sunmaktadır. Bunlardan bu projede gerekli olan ve dronedan video görüntüsü alıp kaydetmeye yarayacak olan Mobile SDK seçeneğidir. Bu SDK ile video görüntüsü alınması dışında, drone kontrolü, önceden tanımlı görevler belirleme ve kalan şarjı görüntüleme gibi işlemler yapılabilmektedir.

Bu proje kapsamında dronedan video görüntüsü almak ve kaydetmek için DJI SDK kullanılmıştır. Bu SDK, Android tabanlı cihazlar için sadece Android Studio ile geliştirmeye izin vermektedir. Bu sebeple geliştirme aşamasında Android Studio kullanılmıştır.

Öncelikle DJI SDK kullanan bir uygulamanın çalışması için bir lisans anahtarı alınması gerekmektedir. Bu lisans anahtarı uygulama çalışmaya başladığında DJI tarafından internet bağlantısı üzerinden test edilir. Eğer uygulama bu testi geçerse droneun kontrolünü ele alabilir. Bu amaçla öncelikle lisans anahtarını test edecek menü, SDKnın örnekleri baz alınarak tasarlandı (Şekil 6.25).



Şekil 6.25. Drone için geliştirilen uygulamanın giriş menüsü.

Buna göre öncelikle uygulama lisans anahtarını test eder, eğer çalışıyorsa başarılı olduğuna dair bir bilgi gönderir. Bu aşamadan sonra herhangi bir dronenun bağlı olup olmadığı kontrol edilir. Eğer bulunursa Başlat seçeneği aktif hale gelir. Tasarımdan sonra ekrandaki menü bileşenleri kontrol eden ve uygulamanın çalışması için gerekli lisans anahtarının test edilmesini sağlayacak olan kod yazıldı ve test edildi.

İkinci aşama olarak video görüntüsünün alınacağı ekran SDK örnekleri aracılığı ile tasarlandı (Şekil 6.26). Bu tasarımda 4 buton soldan sağa sırasıyla fotoğraf modunda iken fotoğraf çekimi, video modundayken video kaydı başlatma veya durdurma, fotoğraf moduna geçiş ve video moduna geçiş işlemlerini yapmaktadır. Video görüntüsü butonların üstündeki alanda oluşacaktır.

Kamera menüsünün tasarımından sonra butonların çalışması ve video görüntüsünün alınması işlemleri için kod yazıldı. Bu koda göre uygulama kamera ekranına ilk girdiğinde öncelikle butonları tespit eder. Daha sonra canlı kamera görüntüsü sağlamak için dronenun kamerasından H264 video verisi alır ve DJI'nın codec yöneticisine gönderir. Bu sayede ekranda canlı video görüntüsü oluşur. Kullanıcı isterse bu görüntülerden butonlar yardımıyla fotoğraf yada video çekebilir.



Şekil 6.26. Drone için geliştirilen uygulamanın kamera menüsü.



Şekil 6.27. Droneun uçuşu sırasında alınan bir görüntü.

Kodlama işleminden sonra droneun havalandırılması ve görüntü alınması için test işlemlerine başlandı. Bu aşamada öncelikle uygulamanın yüklendiği Android cihaz dronenun uzaktan kumandasına USB girişi aracılığı ile takıldı. Drone çalıştırıldı ve uygulama çalıştırıldı. Uygulamanın lisans anahtarının başarılı şekilde kabul edildiği görüldü. Sonraki aşamada drone uzaktan kumanda ile havalandırıldı ve görüntü çekimi yapıldı (Şekil 6.27).

DJI SDK drone kontrolü için geliştirilecek Android uygulamalarının yalnızca Android Studio ile geliştirilmesine izin vermektedir. Ayrıca gerçek zamanlı olarak dronedan gelen görüntüleri bir mobil cihaz aracılığı ile Unity'e aktarmak ve aynı cihazda AG uygulamak mümkün değildir. Bu sebeple daha önceden drone ile alınmış ve kaydedilmiş video görüntüsü, sistemin gerçek zamanlı olarak çalıştığında nasıl olabileceğini göstermek adına Unity ortamına aktarılmış ve model üzerine bindirilmiştir. Zaten mobil cihazlar üzerinde çalışan bir uygulama protatipi geliştirildiğinden, DJI SDK'nın Unity desteği yayınlandığında bu işlem gerçek zamanlı olarak da yapılacaktır.



Şekil 6.28. Dronedan alınan video Unity'e aktarılıp modelin üzerine bindirilmiş hali.

7. SONUÇ

Gelişen teknoloji ile birlikte daha detaylı tasarlanan modeller, daha güçlü aygıtlarda çok daha gerçekçi sonuçlar verecek ve belki yakın bir gelecekte bildiğimiz "gerçeklik" insanlara yeterli gelmemeye başlayacaktır.

Sanal Gerçeklik kavramı ile birlikte oldukça popülerleşen Artırılmış Gerçekliğin, geçmişte uygulanması zor olsa da günümüzde artan mobil cihazlar ile hem uygulanabilirliği hem de kullanılabilirliği oldukça artmıştır. Artırılmış Gerçeklik ile insanlara bildiğimiz gerçekliği üstünde bir kavram sunulur. Bu da insan ilgisini belli bir konuya çekmek için oldukça iyi bir yöntemdir.

2010lu yılların başında, üzerinde uygulama geliştirmeye oldukça elverişli olan akıllı cihazların artması ile birlikte geliştirilen mobil uygulama sayısı da inanılmaz bir artış göstermiştir. Bu artıştan Artırılmış Gerçeklik teknolojisi de nasibini almıştır. Bu teknoloji ile bir çok uygulama geliştirilmiş, bir kısmının başarısı ise global anlamda ses getirmiştir.

Kültürel miras alanları için de pek çok Artırılmış Gerçeklik uygulaması geliştirilmiştir. Bu alanda yapılan akademik çalışmalarda izleme yöntemi olarak işaretçi tabanlı, görüntü tabanlı yada konum tabanlı sistemler tercih edilmiştir.

Ankara Roma Hamamı, Anadolu'daki binlerce tarihi yapıdan sadece bir tanesidir. Bu gün sadece kalıntıları kalmış olsa da Roma tarihine dair pek çok iz barındırır. Geçmişteki toplumların yaşayış şekillerini anlamak adına bu tür eserlerin toplum tarafından bilinmesi ve gelecek nesillere aktarılması önem arz eder.

Tez kapsamında, 3.yüzyılda Roma medeniyeti tarafından inşa edilen fakat günümüzde çoğunlukla kalıntılar halinde bulunan Ankara Roma Hamamı'nın AG teknolojisi ile gezilebilmesini sağlayan bir yazılım geliştirilmesi planlanmıştır. Bu amaç doğrultusunda öncelikle AG yazılım geliştirme araçlarının işaretçi tabanlı sistemleri test edilmiştir. Konum tabanlı uygulama için alanda çalışmak üzere bakanlıklardan gerekli izinler alınamamış bu sebeple sistem prototipi Ankara Üniversitesi Gölbaşı Yerleşkesi'nde gerçekleştirilmesine karar verilmiştir. Sonraki aşamada daha önce bahsedilen yazılım geliştirme araçları ile konum tabanlı uygulama geliştirilmiştir, bu sırada sadece Unity oyun motoru kullanılarak konum tabanlı sistemin daha özgürce gerçekleştirilebileceği görülmüştür. Bu anlamda bir kültür mirası olan Ankara Roma Hamamı'nın 3B modeli,

konum tabanlı bir sistemde, mobil cihazlardan alınan görüntü üzerine bindirilmiştir. Son aşamada ise bir Uzaktan Artırılmış Gerçeklik sistemi tasarlanmış ve anlatılmıştır. Bu amaçla drone ile görüntü alınmış ve görüntünün üzerine 3B hamam modeli bindirilmiştir.

Bu tez kapsamında sunulan sistem gelecekte yapılabilecek bazı çalışmalar ile çok daha kullanışlı ve ilgi çekici hale gelebilir. Gelecekte;

- kalıntılar üzerine görme tabanlı olarak mobil uygulama geliştirilebilir,
- dronedan alınan görüntüye gerçek zamanlı olarak AG uygulanabilir,
- daha önce anlatılan uzaktan artırılmış gerçeklik sistemine tablet ve dronenun yanında akıllı gözlükler de eklenebilir,
- akıllı gözlükler yardımıyla drone kontrolü sağlanabilir,
- kültürel miras alanına gelen ziyaretçilere, tarihi döneme uygun sanal rehberler eşlik edebilir,
- alandaki diğer ziyaretçiler üzerine AG teknolojisi ile döneme uygun kıyafetler giydirilerek alan çok daha ilgi çekici hale getirilebilir.

KAYNAKLAR

- [1] Ribo, M., Lang, P., Ganster, H., Brandner, M., Stock, C., & Pinz, A. Hybrid tracking for outdoor augmented reality applications. *IEEE Computer Graphics and Applications*, 22(6), 54-63 **2002**.
- [2] Azuma, R., A survey of augmented reality, *Presence:Teleoperators and Virtual Environments*, vol. 6, pp. 355-385, **1997**.
- [3] Kishino, P. F. A Taxonomy of Mixed Reality Displays, *IEICE Transactions on Information Systems*, Vol E77-D, No.12 **1994**.
- [4] Papagiannakis, G., Singh, G. and Magnenat-Thalmann, N. A survey of mobile and wireless technologies for augmented reality systems. *Comp. Anim. Virtual Worlds*, 19: 3–22. **2008**
- [5] Zhihan Lv, Alaa Halawani, Shengzhong Feng, Shafiq Ur Réhman, and Haibo Li. Touch-less interactive augmented reality game on vision-based wearable device. *Personal Ubiquitous Comput.* 19, 3-4 ,551-567. **2015**
- [6] Liu, T. Y., Tan, T. H., & Chu, Y. L. 2D barcode and augmented reality supported english learning system. In *Computer and Information Science*, **2007**
- [7] Hsiao, K. F., Chen, N. S., & Huang, S. Y. Learning while exercising for science education in augmented reality among adolescents. *Interactive Learning Environments*, 20(4), 331-349. **2012**
- [8] Dunleavy, M., Dede, C., & Mitchell, R. Affordances and limitations of immersive participatory augmented reality simulations for teaching and learning. *Journal of Science Education and Technology*, 18(1), 7-22. **2009**
- [9] Wang, X. Using augmented reality to plan virtual construction worksite. *International Journal of Advanced Robotic Systems*, 4(4), 42. **2007**
- [10] Yabuki, N., Miyashita, K., & Fukuda, T.. An invisible height evaluation system for building height regulation to preserve good landscapes using augmented reality. *Automation in Construction*, 20(3), 228-235. **2011**
- [11] D. Mizell, “Boeing’s Wire Bundle Assembly Project,” *Fundamentals of Wearable Computers and Augmented Reality*, W. Barfield and T. Caudell, eds., Lawrence Erlbaum and Assoc., pp. 447-467. **2001**
- [12] Regenbrecht, H., Baratoff, G., & Wilke, W. Augmented reality projects in the automotive and aerospace industries. *IEEE Computer Graphics and Applications*, 25(6), 48-56. **2005**
- [13] Pentenrieder, K., Bade, C., Doil, F., & Meier, P. Augmented Reality-based factory planning-an application tailored to industrial needs. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on* (pp. 31-42). IEEE. **2007**

- [14] Wang, X., & Dunston, P. S. Comparative effectiveness of mixed reality- based virtual environments in collaborative design. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(3), 284-296. **2011**
- [15] Luh, Y. P., Wang, J. B., Chang, J. W., Chang, S. Y., & Chu, C. H.. Augmented reality-based design customization of footwear for children. *Journal of Intelligent Manufacturing*, 24(5), 905-917. **2013**
- [16] www.pokemongo.com **Erişim Tarihi : 25/05/2017**
- [17] <http://www.bbc.com/news/business-36791275> **Erişim Tarihi : 25/05/2017**
- [18] <https://venturebeat.com/2016/07/11/pokemon-go-outpaces-clash-royale-as-the-fastest-game-ever-to-no-1-on-the-mobile-revenue-charts/> **Erişim Tarihi : 25/05/2017**
- [19] Google Play Pokemon Go Pictures
- [20] Moul, E., Ungi, T., Welch, M., Lu, J., McGraw, R. C., & Fichtinger, G. Ultrasound-guided facet joint injection training using Perk Tutor. *International journal of computer assisted radiology and surgery*, 8(5), 831-836. **2013**
- [21] Wei, S., Ren, G., & O'Neill, E.. Haptic and audio displays for augmented reality tourism applications. In *Haptics Symposium (HAPTICS)*, 2014 IEEE (pp. 485-488). IEEE. **2014**
- [22] Wang, Y. S., Chen, C. M., Hong, C. M., & Tsai, Y. N. Interactive augmented reality game for enhancing library instruction in elementary schools. In *Computer Software and Applications Conference Workshops (COMPSACW)*, 2013 IEEE 37th Annual (pp. 391-396). IEEE. **2013**
- [23] <https://www.bloomberg.com/news/articles/2014-09-09/snapchat-settles-reggie-brown-suit-credits-him-with-original-idea> **Erişim Tarihi : 27/05/2017**
- [24] <https://www.snapchat.com/> **Erişim Tarihi : 27/05/2017**
- [25] <https://www.instagram.com/> **Erişim Tarihi : 27/05/2017**
- [26] <https://www.forbes.com/sites/baldwincunningham/2017/04/25/snapchat-vs-instagram-stories-what-business-owners-need-to-know/> **Erişim Tarihi : 27/05/2017**
- [27] <http://www.cnn.com/2017/04/18/snapchats-3d-lenses-filters-now-work-in-the-real-world.html> **Erişim Tarihi : 27/05/2017**
- [28] Bostanci, E., Kanwal, N., Ehsan, S., & Clark, A. F. Tracking methods for augmented reality. In *The 3rd international conference on machine vision* (pp. 425-429). **2010**
- [29] Van Krevelen, D., & Poelman, R. *Augmented Reality: Technologies, Applications, and Limitations*. **2007**
- [30] Behzadan, A. H., Timm, B. W., & Kamat, V. R. General-purpose modular hardware and software framework for mobile outdoor augmented reality

- applications in engineering. *Advanced Engineering Informatics*, 22(1), 90-105. **2008**
- [31] Neumann, U., You, S., Cho, Y., Lee, J., & Park, J.. Augmented reality tracking in natural environments. In *International symposium on mixed realities (Vol. 24)*. Tokyo: Ohmsha Ltd and Springer-Verlag. **2009**
- [32] Zhou, F., Duh, H. B. L., & Billingham, M. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality* (pp. 193-202). IEEE Computer Society. **2008**
- [33] Yang, P., Wu, W., Moniri, M., & Chibelushi, C. C. A sensor-based SLAM algorithm for camera tracking in virtual studio. *International Journal of Automation and Computing*, 5(2), 152-162. **2008**
- [34] Livingston, M. A. Magnetic tracker calibration for improved augmented reality registration. *Presence: Teleoperators and Virtual Environments*, 6(5), 532-546. **1997**
- [35] Livingston, M. A., Garrett, W. F., Hirota, G., Whitton, M. C., Pisano, E. D., & Fuchs, H. Technologies for augmented reality systems: Realizing ultrasound-guided needle biopsies. In *Proceedings of the 23rd annual conference on computer graphics and interactive techniques* (pp. 439-446). **1996**
- [36] Welch, G., Bishop, G., Vicci, L., Brumback, S., Keller, K., & Colucci, D. N. High-performance wide-area optical tracking: The hiball tracking system. *presence: teleoperators and virtual environments*, 10(1), 1-21. **2001**
- [37] Zhang, J., Ong, S. K., & Nee, A. Y. C. RFID-assisted assembly guidance system in an augmented reality environment. *International Journal of Production Research*, 49(13), 3919-3938. **2011**
- [38] Ullah, S. Multi-modal Assistance for Collaborative 3D Interaction: Study and analysis of performance in collaborative work. Unpublished doctoral thesis). Université d'Évry-Val d'Essonne, Évry, France. **2011**
- [39] Mistry, P., Kuroki, T., & Chang, C.. TaPuMa: tangible public map for information acquirement through the things we carry. In *Proceedings of the 1st international conference on Ambient media and systems* (p. 12). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). **2008**
- [40] Cooper, N., Keatley, A., Dahlquist, M., Mann, S., Slay, H., Zucco, J., ... & Thomas, B. H.. Augmented reality chinese checkers. In *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology* (pp. 117-126). **2004**
- [41] Chen, X., Xu, L., Wang, Y., Wang, H., Wang, F., Zeng, X., ... & Egger, J. Development of a surgical navigation system based on augmented reality using an optical see-through head-mounted display. *Journal of biomedical informatics*, 55, 124-131. **2015**

- [42] Bostanci, E., Kanwal, N., & Clark, A. F. Augmented reality applications for cultural heritage using Kinect. *Human-centric Computing and Information Sciences*, 5(1), 20. **2015**
- [43] Tsai, C. Y., & Hsu, K. H. An Application of Using Bluetooth Indoor Positioning, Image Recognition and Augmented Reality. In *e-Business Engineering (ICEBE)*, 2016 IEEE 13th International Conference on (pp. 276-281). IEEE. **2016**
- [44] Kaufmann, H. Construct3D: an augmented reality application for mathematics and geometry education. In *Proceedings of the tenth ACM international conference on Multimedia* (pp. 656-657). ACM. **2002**
- [45] Juan, M. C., Botella, C., Alcaniz, M., Banos, R., Carrion, C., Melero, M., & Lozano, J. A. An augmented reality system for treating psychological disorders: application to phobia to cockroaches. In *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on* (pp. 256-257). IEEE. **2004**
- [46] Azuma, R., Hoff, B., Neely, H., & Sarfaty, R. A motion-stabilized outdoor augmented reality system. In *Virtual Reality, 1999. Proceedings.*, IEEE (pp. 252-259). IEEE. **1999**
- [47] Park, J., & Park, J. 3DOF tracking accuracy improvement for outdoor Augmented Reality. In *Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on* (pp. 263-264). IEEE. **2010**
- [48] <http://www.gps.gov/systems/gps/> **Erişim Tarihi : 11/6/2017**
- [49] <http://www.kircherelectronics.com/blog/index.php/11-android/sensors/15-android-gyroscope-basics> **Erişim Tarihi : 11/6/2017**
- [50] <https://microem.ru/files/2012/08/GPS.G1-X-00006.pdf> **Erişim Tarihi : 11/6/2017**
- [51] Bostanci, G. E. User tracking methods for augmented reality applications in cultural heritage (Doktora Tezi, University of Essex). **2014**
- [52] http://maia.usno.navy.mil/NSFA/NSFA_cbe.html#EarthRadius2009 **Erişim Tarihi : Haziran 2017**
- [53] <http://www.anadolumedeniyetlerimuzesi.gov.tr/TR,77792/ankara-romahamami.html> **Erişim Tarihi : Haziran 2017**
- [54] <http://socialcompare.com/en/comparison/augmented-reality-sdks> **Erişim Tarihi : Haziran 2017**
- [55] <http://www.metaio.eu/> **Erişim Tarihi : Haziran 2017**
- [56] <http://www.reuters.com/article/us-augmentedreality-idUSBRE85K0X020120621> **Erişim Tarihi : Haziran 2017**
- [57] <https://blogs.wsj.com/digits/2015/05/28/apple-buys-german-augmented-reality-firm-metaio/> **Erişim Tarihi : Haziran 2017**
- [58] <https://developer.vuforia.com/> **Erişim Tarihi : Haziran 2017**

- [59] <https://www.wikitude.com/> **Erişim Tarihi : Haziran 2017**
- [60] <https://www.kudan.eu/> **Erişim Tarihi : Haziran 2017**
- [61] <https://artoolkit.org/> **Erişim Tarihi : Haziran 2017**
- [62] <https://www.eclipse.org/> **Erişim Tarihi : Haziran 2017**
- [63] <https://developer.android.com/studio/> **Erişim Tarihi : Haziran 2017**
- [64] <https://unity3d.com> **Erişim Tarihi : Haziran 2017**
- [65] <https://www.playstation.com/> **Erişim Tarihi : Haziran 2017**
- [66] <http://www.xbox.com/> **Erişim Tarihi : Haziran 2017**
- [67] <http://www.nintendo.com/wiiu/> **Erişim Tarihi : Haziran 2017**
- [68] <https://unity3d.com/showcase/gallery/non-games> **Erişim Tarihi : Haziran 2017**
- [69] <http://www.muze.gov.tr/tr/muzeler/roma-hamami> **Erişim Tarihi : Haziran 2017**
- [70] <https://earth.google.com/> **Erişim Tarihi : Haziran 2017**
- [71] <https://www.assetstore.unity3d.com/en/#!/content/25088> **Erişim Tarihi : Temmuz 2017**
- [72] <https://targetmanager.wikitude.com/> **Erişim Tarihi : Temmuz 2017**
- [73] <https://www.wikitude.com/products/wikitude-sdk-features/wikitude-3d-encoder/> **Erişim Tarihi : Temmuz 2017**
- [74] www.wikitude.com/developer/developer-forum/-/message_boards/message/690959 **Erişim Tarihi : Temmuz 2017**
- [75] <https://docs.unity3d.com/Manual/CreatingAndUsingScripts.html> **Erişim Tarihi : Temmuz 2017**
- [76] <https://www.dji.com/> **Erişim Tarihi : Temmuz 2017**
- [77] <http://www.dji.com/phantom-3-pro/info#specs> **Erişim Tarihi : Temmuz 2017**
- [78] Li, J., & Fan, X. Outdoor augmented reality tracking using 3d city models and game engine. In Image and Signal Processing (CISP), 2014 7th International Congress on (pp. 104-108). IEEE. **2014**
- [79] Forsman, M., Arvo, J., & Lehtonen, T. (2016, October). Extended panorama tracking algorithm for augmenting virtual 3D objects in outdoor environments. In Virtual System & Multimedia (VSM), 2016 22nd International Conference on (pp. 1-8). IEEE. **2016**
- [80] Langlotz, T., Mooslechner, S., Zollmann, S., Degendorfer, C., Reitmayr, G., & Schmalstieg, D. Sketching up the world: in situ authoring for mobile augmented reality. Personal and ubiquitous computing, 16(6), 623-630. **2012**

- [81] Alappanavar, P. B., Kurvey, B., Karad, M., & Bhagwatkar, S. Location Based Augmented Reality. *International Journal of Scientific & Engineering Research*, 4(5), 566-568. **2013**
- [82] Daponte, P., De Vito, L., Picariello, F., & Riccio, M. State of the art and future developments of the Augmented Reality for measurement applications. *Measurement*, 57, 53-70. **2014**
- [83] Hsieh, C. H., & Lee, J. D. Markerless Augmented Reality via Stereo Video See-Through Head-Mounted Display Device. *Mathematical Problems in Engineering*, **2015**
- [84] Bostanci, E., Clark, A. F., & Kanwal, N. Vision-based user tracking for outdoor augmented reality. In *Computers and Communications (ISCC), 2012 IEEE Symposium on* (pp. 000566-000568). IEEE. **2012**
- [85] <https://epson.com/moverio-augmented-reality> **Erişim Tarihi : Temmuz 2017**
- [86] <https://developers.google.com/glass/> **Erişim Tarihi : Temmuz 2017**
- [87] Bimber, O., & Raskar, R.. *Spatial augmented reality: merging real and virtual worlds*. CRC press. **2005**
- [88] Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., & MacIntyre, B. Recent advances in augmented reality. *IEEE computer graphics and applications*, 21(6), 34-47. **2001**
- [89] Bostanci, E., Unal, M., *Making Visits to Museums More Fun with Augmented Reality using Kinect, Drones and Games*. *International Journal Of Computers And Communications* 10, 7-10 **2016**
- [90] Stricker, D., & Kettenbach, T. Real-time and markerless vision-based tracking for outdoor augmented reality applications. In *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on* (pp. 189-190). IEEE. **2001**
- [91] Hu, P. Y., & Tsai, P. F. Mobile outdoor augmented reality project for historic sites in Tainan. In *Advanced Materials for Science and Engineering (ICAMSE), International Conference on* (pp. 509-511). IEEE. **2016**
- [92] Vlahakis, V., Karigiannis, J., Tсотros, M., Gounaris, M., Almeida, L., Stricker, D., ... & Ioannidis, N. Archeoguide: first results of an augmented reality, mobile computing system in cultural heritage sites. *Virtual Reality, Archeology, and Cultural Heritage*, 9. **2001**
- [93] Papagiannakis, G., Schertenleib, S., O'Kennedy, B., Arevalo-Poizat, M., Magnenat-Thalmann, N., Stoddart, A., & Thalmann, D. Mixing Virtual and Real scenes in the site of ancient Pompeii. *Computer animation and virtual worlds*, 16(1), 11-24. **2005**
- [94] Haugstvedt, A. C., & Krogstie, J. Mobile augmented reality for cultural heritage: A technology acceptance study. In *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on* (pp. 247-255). IEEE. **2012**

- [95] Wojciechowski, R., Walczak, K., White, M., & Cellary, W. Building virtual and augmented reality museum exhibitions. In Proceedings of the ninth international conference on 3D Web technology (pp. 135-144). ACM. **2004**
- [96] Angelopoulou, A., Economou, D., Bouki, V., Psarrou, A., Jin, L., Pritchard, C., & Kolyda, F. Mobile augmented reality for cultural heritage. Mobile wireless middleware, operating systems, and applications, 15-22. **2011**
- [97] <https://play.google.com/store/apps/details?id=com.google.android.apps.translate>
Eriřim Tarihi: Temmuz 2017

ÖZGEÇMİŞ

Kimlik Bilgileri

Ad, Soyad: Metehan Ünal

Doğum Yeri: Adapazarı

Medeni Hali: Evli

E-posta: u.metehan@gmail.com

Adres: Ankara Üniversitesi Gölbaşı Yerleşkesi, I Blok Bilgisayar Mühendisliği,
Gölbaşı, ANKARA

Eğitim

Lisans: Ankara Üniversitesi, Bilgisayar Mühendisliği, Ankara, 2014

Lise: Mustafa Şeker Anadolu Lisesi, Bilecik, 2009

Yabancı Diller

İngilizce

İş Deneyimi

2015 - : Ankara Üniversitesi, Bilgisayar Mühendisliği Bölümü, Araştırma Görevlisi

Deneyim Alanları

Artırılmış Gerçeklik, Bilgisayar Grafikleri, Yapay Zeka

Tezden Üretilmiş Projeler ve Bütçesi

TÜBİTAK 3001:215E156, Kültürel Miras Alanları için Mobil ve Uzaktan Artırılmış Gerçeklik Sistemi Geliştirilmesi, 41.476TL, 2016-2018.

Tezden Üretilmiş Yayınlar

Bostancı, E., Ünal, M., “Making Visits to Museums More Fun with Augmented Reality using Kinect, Drones and Games”, The International Conference on Circuits, Systems, Signal Processing, Communications and Computers, Vienna, Austria, 2016.

Tezden Üretilmiş Tebliğ ile Katıldığı Toplantılar

Ünal, M., Bostancı, E., Sertalp, E. and Tosun, S., “A Distant Augmented Reality System for Cultural Heritage Sites using Drones”, 3rd Global Summit and Expo on Multimedia and Artificial Intelligence, Lisbon, 2017.



HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/DOKTORA TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI BAŞKANLIĞI'NA

Tarih: 23/07/2017

Tez Başlığı / Konusu: KÜLTÜREL MİRAS ALANLARI İÇİN UZAKTAN ARTIRILMIŞ
GERÇEKLIK SİSTEMİ

Yukarıda başlığı/konusu gösterilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler ve d) Sonuç kısımlarından oluşan toplam 104... sayfalık kısmına ilişkin, 23/07/2017 tarihinde şahsım/tez danışmanım tarafından Turnitin... adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 2 'tür.

Uygulanan filtrelemeler:

- 1- Kaynakça hariç
- 2- Alıntılar hariç/dâhil
- 3- 5 kelimedenden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Tez Çalışması Orjinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

Tarih ve İmza

Adı Soyadı: Metehan ÜNAL
Öğrenci No: N14126557
Anabilim Dalı: Bilgisayar Mühendisliği
Programı:
Statüsü: Y.Lisans Doktora Bütünleşik Dr.

23/07/2017

DANIŞMAN ONAYI

UYGUNDUR.

Doç. Dr. Süleyman Tosun
(Unvan, Ad Soyad, İmza)