UNIVERSITATEA ALEXANDRU IOAN CUZA IAŞI

**FACULTATEA DE INFORMATICĂ**



LUCRARE DE LICENŢĂ

**Driving Assistant**

**propusă de**

***Leuștean Andrei***

**Sesiunea:** luna, anul

**Coordonator ştiinţific**

**Drd. Colab. Olariu Florin**

UNIVERSITATEA ALEXANDRU IOAN CUZA IAŞI

**FACULTATEA DE INFORMATICĂ**

LUCRARE DE LICENŢĂ

**Driving Assistant**

***Leuștean Andrei***

**Sesiunea:** luna, anul

**Coordonator ştiinţific**

**Drd. Colab. Florin Olariu**

Avizat,

Îndrumător Lucrare de Licență

Titlul, Numele și prenumele \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Data \_\_\_\_\_\_\_\_\_\_\_\_ Semnătura \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DECLARAȚIE privind originalitatea conținutului lucrării de licență**

Subsemntatul(a) ………………………………………………………………………………………

domiciliul în …………………………………………………………………………………………………..

născut(ă) la data de ………………..…., identificat prin CNP ………….……………..………………..., absolvent(a) al(a) Universității „Alexandru Ioan Cuza” din Iași, Facultatea de ………………………. specializarea …………………………………………………………, promoția …………………………., declar pe propria răspundere, cunoscând consecințele falsului în declarații în sensul art. 326 din Noul Cod Penal și dispozițiile Legii Educației Naționale nr. 1/2011 art.143 al. 4 si 5 referitoare la plagiat, că lucrarea de licență cu titlul: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_elaborată sub îndrumarea dl. / d-na \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, pe care urmează să o susțină în fața comisiei este originală, îmi aparține și îmi asum conținutul său în întregime.

De asemenea, declar că sunt de acord ca lucrarea mea de licență să fie verificată prin orice modalitate legală pentru confirmarea originalității, consimțind inclusiv la introducerea conținutului său într-o bază de date în acest scop.

Am luat la cunoștință despre faptul că este interzisă comercializarea de lucrări științifice in vederea facilitării fasificării de către cumpărător a calității de autor al unei lucrări de licență, de diploma sau de disertație și în acest sens, declar pe proprie răspundere că lucrarea de față nu a fost copiată ci reprezintă rodul cercetării pe care am întreprins-o.

Dată azi, ………………………… Semnătură student …………………………

DECLARAȚIE DE CONSIMȚĂMÂNT

Prin prezenta declar că sunt de acord ca Lucrarea de licență cu titlul „*Driving Assistant*”, codul sursă al programelor și celelalte conținuturi (grafice, multimedia, date de testetc.) care însoțesc această lucrare să fie utilizate în cadrul Facultății de Informatică.

De asemenea, sunt de acord ca Facultatea de Informatică de la Universitatea „Alexandru Ioan Cuza” din Iași, să utilizeze, modifice, reproducă și să distribuie în scopuri necomerciale programele-calculator, format executabil și sursă, realizate de mine în cadrul prezentei lucrări de licență.

Iași, *26.06.2018*

Absolvent *Andrei Leuștean*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(semnătura în original)

Cuprins

[Introducere 10](#_Toc518329735)

[Structura aplicației 11](#_Toc518329736)

[Contribuții 12](#_Toc518329737)

[Capitolul 1. Tehnologii folosite 13](#_Toc518329738)

[1.1 Unity 13](#_Toc518329739)

[1.2 Vuforia 14](#_Toc518329740)

[1.3 Microsoft Paint 14](#_Toc518329741)

[1.4 Concluzii 15](#_Toc518329742)

[Capitolul 2. Problemele abordate 16](#_Toc518329743)

[2.1 Problema manualului de utilizare 16](#_Toc518329744)

[2.2 Problema diversificării automotivelor 16](#_Toc518329745)

[2.3 Soluția 17](#_Toc518329746)

[Capitolul 3. Implementare 18](#_Toc518329747)

[3.1 Baza de date Vuforia 18](#_Toc518329748)

[3.2 Aplicația 20](#_Toc518329749)

[Capitolul 4. Secvențe relevante de cod 22](#_Toc518329750)

[Scriptul btn\_back.cs 22](#_Toc518329751)

[Scriptul btn\_click.cs 23](#_Toc518329752)

[Scriptul btn\_virtual\_click.cs 24](#_Toc518329753)

[Scriptul CameraFocusController.cs 25](#_Toc518329754)

[Scriptul menu\_controller.cs 26](#_Toc518329755)

[Scriptul text\_blinkerdetection.cs 27](#_Toc518329756)

[Scriptul text\_makeDetection.cs 28](#_Toc518329757)

[Posibilități de dezvoltare ulterioară 29](#_Toc518329758)

[Concluzii finale 30](#_Toc518329759)

[Biliografie 31](#_Toc518329760)

[Anexe 32](#_Toc518329761)

# Introducere

Driving Assistant este o metodă de descoperire și evidențiere a funcționalităților unui autoturism. Aplicația este o alternativă a manualului de utilizare, fiind mai simplă de folosit și cu rezultate mai rapide prin reducerea semnificativă a timpului de căutare pentru o anumită funcționalitate. Aplicația utilizează camera unui dispozitiv Android pentru a scana anumite comutatoare și/sau controller-e dintr-un autoturism (ex: semnalizator, schimbător de viteze, etc.) și afișarea unor panouri ce descriu funcționalitatea comutatorului respectiv la momentul recunoașterii. Astfel, programul este util atât pentru cei ce urmează o școală de șoferi, cât și pentru posesorii unor autoturisme cu foarte multe funcționalități (ex: mașini de lux).

Scanarea și recunoașterea comutatoarelor se realizează utilizând Realitatea Augmentată.

Am ales ca lucrare de licență Realitatea Augmentată deoarece este un domeniu nou, iar aplicațiile în acest domeniu pot fi foarte practice. Deși prima aplicație pentru Realitate Augmentată a fost dezvoltată în 1992 pentru forțele aeriene ale Statelor Unite, popularitatea acestui domeniu a explodat în anul 2009, când ARToolKit, biblioteca utilizată pentru aplicații AR dezvoltată inițial în 1999, a fost portată pe Adobe Flash ca FLARToolKit.

Am ales această temă de licență deoarece am întâmpinat dificultăți în găsirea unor funcționalități în propriul autoturism, urmând a constata existența acestei probleme și la alți conducători auto.

Obiectivul general al lucrării este de a facilita familiarizarea șoferilor cu autoturismul pe care îl conduc.

# Structura aplicației

Aplicația conține un meniu principal prin intermediul căruia se poate alege modul de interacțiune cu butoanele la momentul recunoașterii, submeniul de opțiuni și ieșirea din aplicație.

Opțiunile meniului sunt:

1. Steering wheel scan – pornește camera în modul de interacțiune „partial AR”, butoanele ce apar fiind acționate de pe ecranul dispozitivului mobil.
2. Features scan – pornește camera în modul de interacțiune ”full AR”, butoanele ce apar fiind acționate în fața camerei dispozitivului.
3. Options – deschide un meniu de opțiuni, unde poate fi schimbată limba aplicației.
4. Quit – închide aplicația.

# Contribuții

Pe baza celor exprimate în Introducere, și anume, construirea unei aplicații care este o alternativă a unui manual de utilizare, folosind o cameră a unui smartphone și VR (realitatea virtuală), am implementat o aplicație, folosind și integrând următoarele API-uri deja existente:

* Unity API
* Vuforia SDK

Consider că ideea mea este cea mai bună, deoarece:

* Alternativa manualului de utilizare integrată pe un smartphone este mai modernă, având în vedere faptul că majoritatea oamenilor în general dețin un astfel de dispozitiv
* Aplicația este mult mai ușor de folosit pentru informarea șoferului, întrucât acesta nu mai este nevoit să parcurgă manualul de utilizare, uneori chiar în mod repetat, pentru a afla funcționalitatea unui comutator
* Aplicația este mult mai accesibilă prin suportul multi-lingvist, fiind evitată bariera limbii în care este scris manualul fizic

# Capitolul 1. Tehnologii folosite

Deși prima aplicație pentru Realitate Augmentată a fost dezvoltată în 1992 pentru forțele aeriene ale Statelor Unite, popularitatea acestui domeniu a explodat în anul 2009, când ARToolKit, biblioteca utilizată pentru aplicații AR dezvoltată inițial în 1999, a fost portată pe Adobe Flash ca FLARToolKit.

Realitatea Augmentată este o experiență interactivă a unui mediu real, al cărui elemente sunt „augmentate” prin informație perceptuală generată de computer. Aceasta aduce în percepția unei persoane despre lumea reală componente ale lumii digitale, nu doar ca un simplu afișaj de date, ci prin integrarea de senzații imersive ce sunt percepute ca părți naturale ale mediului. Realitatea Augmentată este folosită pentru intensificarea mediilor sau situațiilor naturale și oferirea de experiențe îmbogățite perceptual.

În dezvoltarea aplicației „Driver’s Assistant” am utilizat următoarele tehnologii:

* Unity game engine
* Vuforia framework
* Microsoft Paint

## 1.1 Unity

Un game engine este un framework pentru dezvoltarea de jocuri ce suportă și îmbină diverse zone esențiale.

Unity este un game engine ce suportă grafică 2D și 3D, funcționalitate drag-and-drop și scripting utilizând C#. Acesta a fost creat în 2005 ca mediu de dezvoltare pentru OS X, urmând ca în timp să se extindă la 27 de platforme, întrucât ținta celor de la Unity Technologies a fost accesibilitatea universală a dezvoltării jocurilor. Prin facilitarea procesului de game development, concentrarea atât pe platforma mobile cât și dezvoltarea de aplicații AR/VR, Unity a devenit cel mai popular mediu de dezvoltare de jocuri video. În prezent, Unity controlează 45% din market-ul global de game engines. Printre lucrurile care susțin această statăstică, se regăsesc și următoarele afirmații de pe pagina Unity:

* 2.4 miliarde de dispozitive mobile unice rulează jocuri dezvoltate în Unity
* 34% din top 1000 cele mai bune jocuri gratuite sunt dezvoltate în Unity
* 50% din jocurile mobile sunt dezvoltate în Unity Game Engine

Dominanța pe piață a acestui mediu de dezvolare se datorează unui model de bussiness foarte puternic, bazat pe modelul SAAS (Software as a service). Modelul SAAS are 3 categorii de prețuri, observabile în Anexa 1 de la pagina 22.

Am ales să folosesc Unity, deoarece Vuforia nu are suport pentru Unreal, iar Android Studio nu poate fi utilizat pentru crearea aplicațiilor pentru iOS.

## 1.2 Vuforia

Vuforia este un framework pentru dispozitivele mobile utlizat în dezvoltarea aplicațiilor AR, utilizând tehnologia Computer Vision pentru a recunoaște și urmări imagini planare (Imagini Țintă) și obiecte simple 3D în timp real. Capabilitatea de a înregistra imagini permite dezvoltatorilor să poziționeze și să orienteze obiectele virtuale în relație cu imagini din lumea reală când acestea sunt văzute prin camera unui dispozitiv mobil. Obiectul virtual apoi urmărește poziția și orientarea imaginii în timp real pentru ca perspectiva observatorului asupra obiectului să corespundă cu perspectiva imaginei țintă, astfel încât obiectul virtual pare a fi parte din scena reală Vuforia suportă o varietate de tipuri de țintă 2D și 3D, inclusiv imagini țintă „markerless”, configurații 3D multi-țintă și o formă de Fiducial Marker adresabil, cunoscut ca VuMark. Alte caracteristici ale framework-ului includ detecția de ocluziune localizată utilizând butoane virtuale, runtime image target selection, și abilitatea de a crea și reconfigura programabil seturi de ținte la runtime.

Vuforia furnizează API în limbajele C++, Java, Objective-C++ și .Net printr-o extensie la Unity. Astfel, framework-ul suportă dezvoltarea nativă pentru iOS și Android, permițând în același timp dezvoltarea de aplicații AR în Unity care sunt ușor portabile pe ambele platforme. Drept consecință, aplicațiile dezvoltate folosind Vuforia sunt compatibile cu o gamă largă de dispozitive mobile, inclusiv: iPhone, iPad și dispozitivele și ce rulează Android OS versiunea 2.2 sau mai mare.

## 1.3 Microsoft Paint

Deoarece dimensiunea imaginilor ce pot fi încărcate în baza de date Vuforia este limitată la 2 MB iar fotografiile realizate cu camera de la telefon au o dimensiune cuprinsă între 2 și 5 MB, este nevoie de un program third-party pentru reducerea dimensiunilor imaginii, fie prin scalarea la o rezoluție mai mică, fie prin decuparea unei imagini. Pentru realizarea acestui proces, am ales Microsoft Paint deoarece este cea mai accesibilă aplicație de redimensionare a imaginilor din sistemul de operare Windows.

## 1.4 Concluzii

Realitatea Augmentată a crescut în popularitate în ultimii ani, iar hardware-ul recent dezvoltat pentru acest domeniu – Google Glasses, Microsoft Hololens – au avut un impact imens asupra practicabilității acestor aplicații (precum utilizarea Hololens pentru învățarea operațiilor chirurgicale). Astfel, Realitatea Augmentată este considerată calea către viitor, prin interactivitatea intensificată, intuitivă, cu obiectele din lumea virtuală.

În urma observării acestui domeniu și constatării problemei manualului de utilizare a autovehiculelor, am decis să dezvolt o aplicație AR pentru dispozitive mobile care să faciliteze înțelegerea funcționalităților unui autoturism.

# Capitolul 2. Problemele abordate

Aplicația rezolvă două probleme care s-au dovedit a fi comune în privința condusului unui autoturism cu care un șofer nu este foarte familiarizat. Aceste probleme sunt descrise mai jos.

## 2.1 Problema manualului de utilizare

La cumpărarea unui autoturism de la un furnizor sau producător, acesta (autoturismul) vine însoțit de un manual de utilizare, în limba țării în care a fost cumpărat. După o perioadă de timp, proprietarul decide să vândă autoturismul, iar noul cumpărător locuiește într-o altă țară, în care se vorbește o limbă diferită, iar cumpărătorul nu cunoaște limba în care este scris manualul de utilizare. Din această cauză, viitorul proprietar va avea dificultăți în a înțelege toate comenzile din autoturism, în ciuda deținerii manualului de utilizare.

## 2.2 Problema diversificării automotivelor

Pentru a se remarca pe piață și pentru a putea oferi o experiență de condus unică sau diferită, producătorii auto au reorganizat comenzile autoturismelor, de la design-ul acestora (un buton pe care este desenat un fulg de nea în locul unui buton pe care scrie „AC” pentru aerul condiționat) la poziția și forma unui controller (pentru aprinderea farurilor, în funcție de model, se rotește maneta de semnalizare în sens trigonometric de 2 ori, sau se rotește întrerupătorul pentru faruri, situat sub gura de ventilație din stânga volanului, în sensul acelor de ceas de 2 ori). Astfel, conducătorii auto care au condus un anumit autoturism, fie și doar în școala de șoferi, nu vor reuși să folosească comenzile fără consultarea manualului de utilizare. Dacă manualul de utilizare lipsește sau este scris într-o limbă diferită, cu care șoferul să nu fie familiar, întâmpinăm problema descrisă anterior.

## 2.3 Soluția

Deoarece ambele probleme se referă la comutatoarele unui autoturism, iar comutatoarele pot fi observate fizic, am decis să folosesc Realitatea Augmentată pentru a recunoaște comutatoarele respective și afișarea unei descrieri referitoare la funcționalitatea acestora; astfel, având la dispoziție un telefon, șoferii vor putea recunoaște funcționalitățile importante din autoturismul pe care îl vor conduce. Driver’s Assistant devine astfel o alternativă a manualului de utilizare, fiind mult mai ușor de folosit și rezolvând problema descrisă la punctul 2.1 prin oferirea de suport multi-lingvist. Problema descrisă la punctul 2.2 este rezolvată prin afișarea funcționalităților din autoturism, șoferul fiind responsabil doar cu căutarea comutatoarelor.

# Capitolul 3. Implementare

În cadrul acestui capitol va fi descrisă implementarea, alături de detalii importante precum: baza de date Vuforia, punctele de contrast folosite pentru recunoaștere, butoanele virtuale.

## 3.1 Baza de date Vuforia

Pentru a putea recunoaște un anumit comutator dintr-un autoturism, avem nevoie întâi de o poză cu comutatorul respectiv. După realizarea pozei, redimensionăm fișierul dacă are o dimensiune mai mare de 2MB (limita impusă de Vuforia) și apoi o vom încărca în baza de date online. După încărcarea unei imagini, aceasta va primi un rating de la 0 la 5 stele reprezentând gradul de recunoaștere al unei imagini, fapt ce poate fi observat în Figura 1 de mai jos.

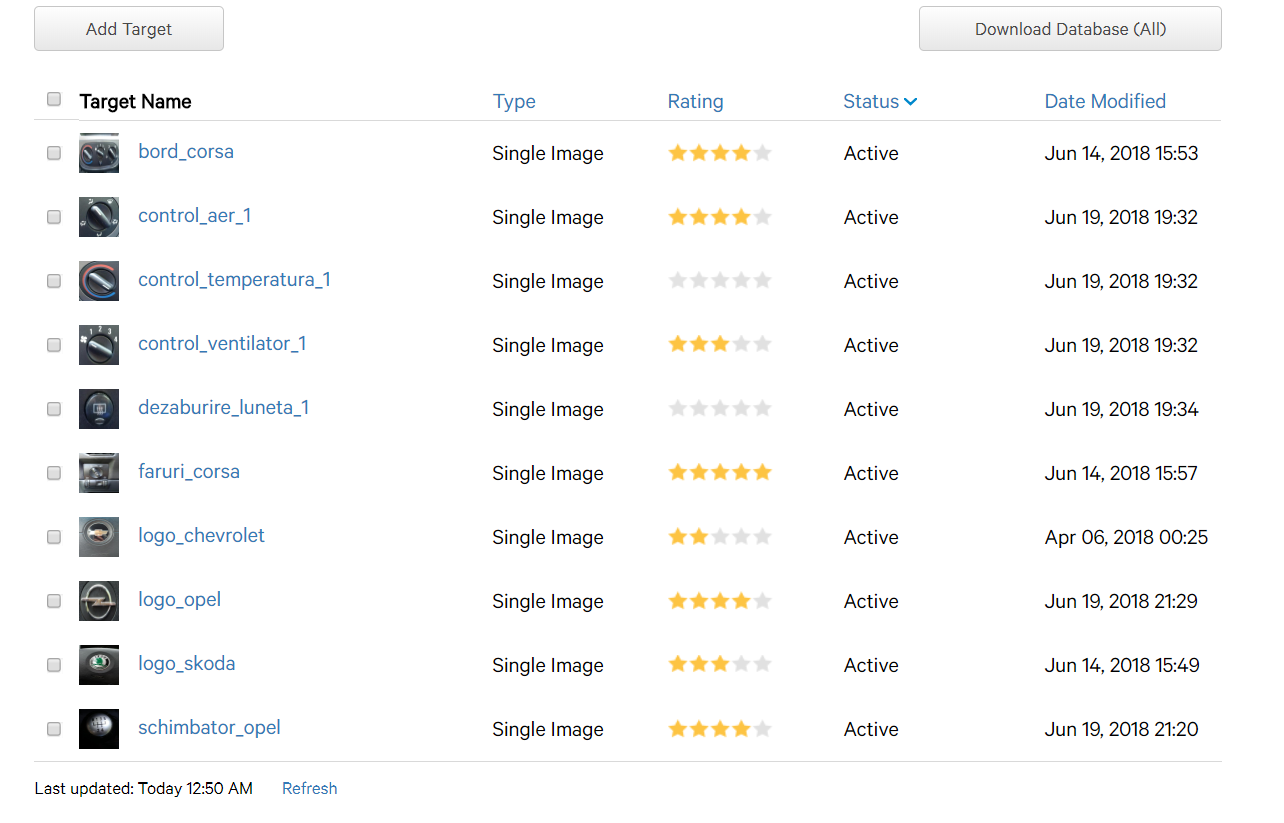


Fig. 1 Baza de date Vuforia

Rating-ul este acordat în funcție de numărul de puncte de contrast lumină-întuneric sau alb-negru (nu se ține cont de culorile prezente în imagine)

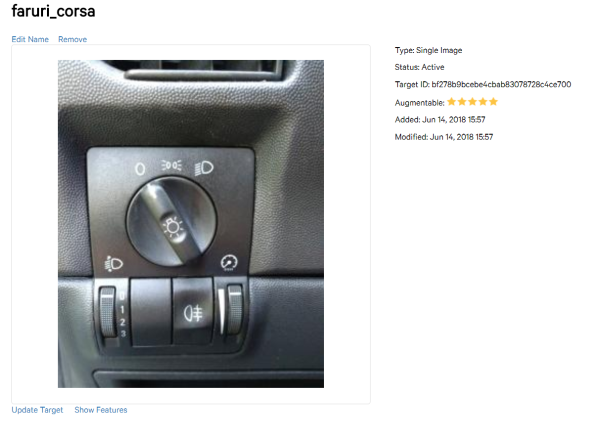
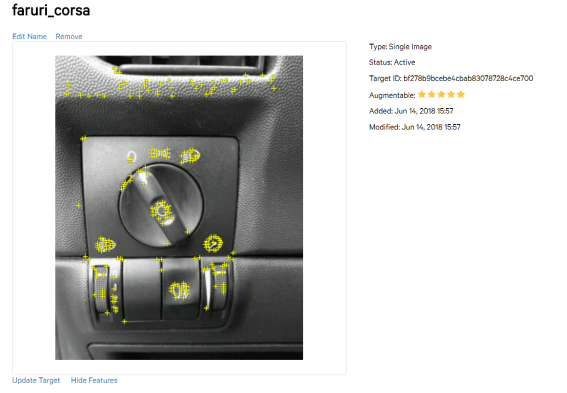
 

Fig. 2 Imagine cu rating de 5 stele Fig. 3 Punctele de contrast ale unei imagini de 5 stele

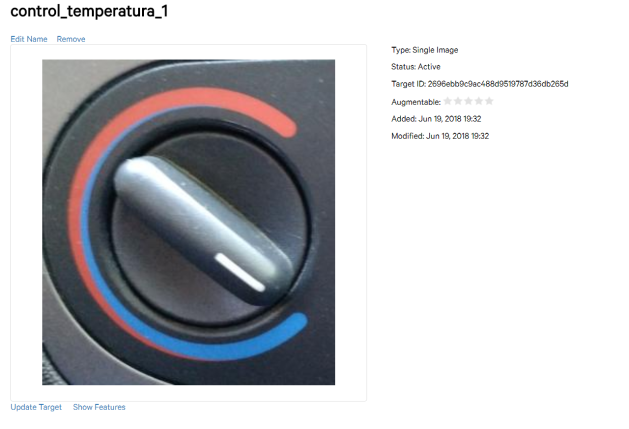
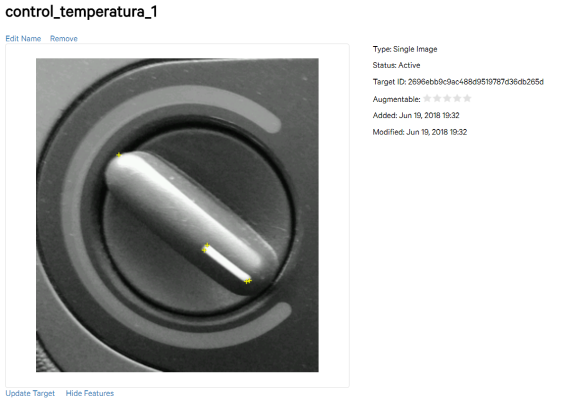
 

Fig. 4 Imagine cu rating de 0 stele Fig. 5 Punctele de contrast ale unei imagini de 0 stele

După cum se observă și în figura anterioară, o imagine cu foarte puține puncte de contrast (în cazul de față sunt 5) va fi recunoscută foarte greu sau chiar deloc, caz în care fie renunțăm la recunoașterea comutatorului respectiv, fie înlocuim imaginea, dacă este posibil, alta mai bună.

## 3.2 Aplicația

Aplicația întâmpină utilizatorul cu meniul principal de la figura 6.

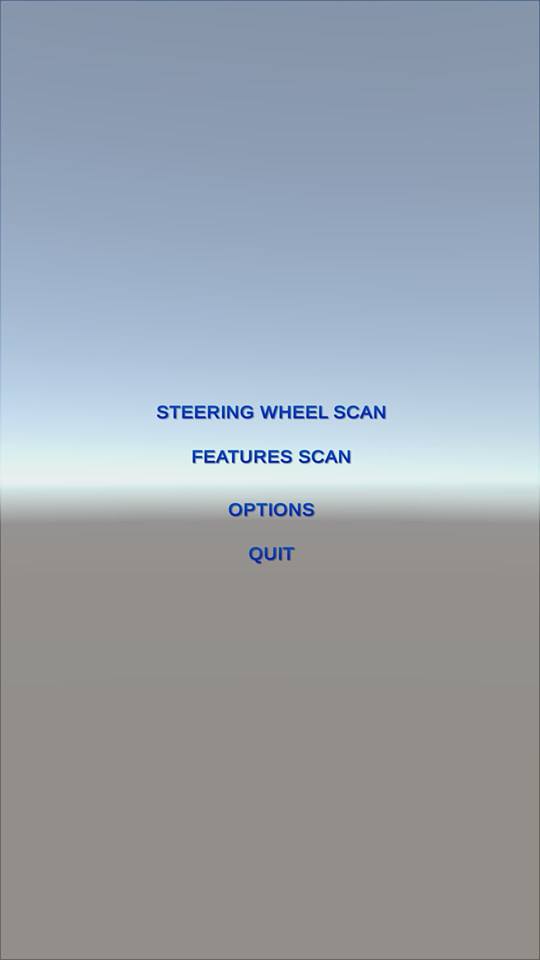


Fig. 6 Meniul Principal

Steering wheel scan porneste modul de scanare „Partial AR”, unde recunoașterea unui comutator poate fi observată prin culoarea interfeței grafice (roșu dacă nu a fost găsit niciun comutator, albastru altfel). În acest mod de scanare, interacțiunea cu butoanele și panourile descriptive se realizează prin touch screen-ul telefonului.

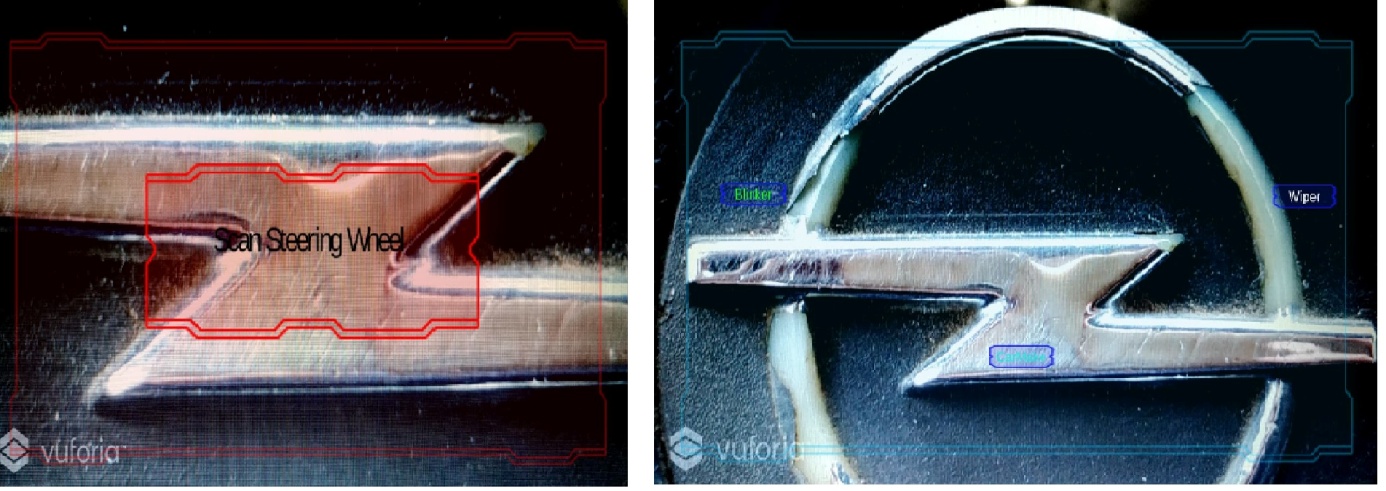


Fig. 7 Comparație între stările scanare/scanat ale modului Partial AR

Features scan inițiază modul de scanare „Full AR”, unde, la momentul recunoașterii unui comutator, pe camera va putea fi observat un buton. Acest buton poate fi acționat în mediul real, acoperind butonul virtual cu degetul, pentru a afișa panoul descriptiv pe cameră. Astfel, dacă telefonul este folosit ca pereche de ochelari utilizând Google Cardboard, sau prin utilizarea unui Head-up Unit (Google Glasses, Microsoft Hololens), utilizatorul poate avea parte de o experiență imersivă în mediul AR. Detecția ocluziunii localizate este realizată utilizând punctele de contrast descrise

la 3.1, motiv pentru care se recomandă plasarea butoanelor virtuale într-un loc bogat în astfel de puncte.

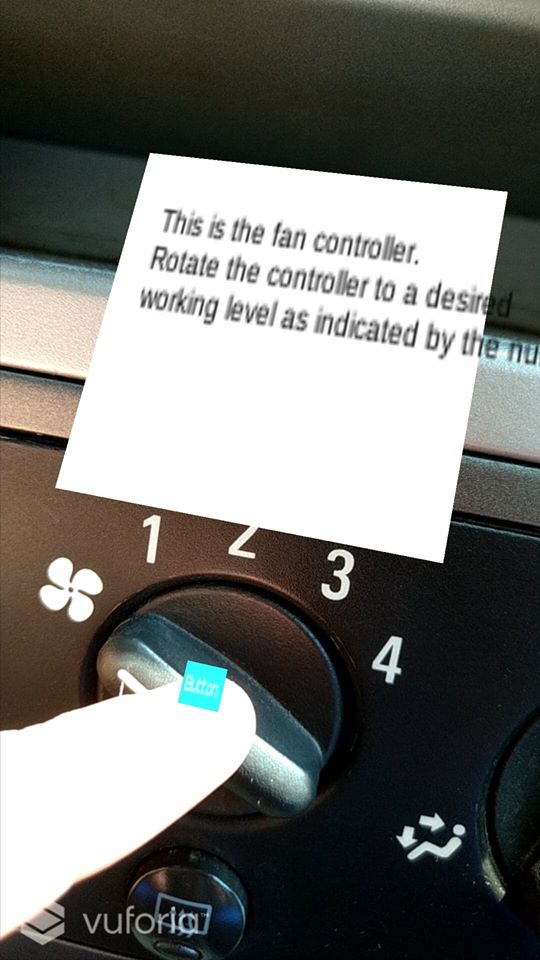


Fig. 8 Exemplu de utilizare al unui buton virtual

O consecință a utilizării punctelor de contrast în implementarea butoanelor virtuale este posibilitatea ca acestea să fie acționate de lumină și/sau umbre.

# Capitolul 4. Secvențe relevante de cod

Întrucât Unity și Vuforia asigură majoritatea funcționalităților de care am avut nevoie, contribuția proprie în ceea ce privește codul nu este foarte mare, script-urile necesare aplicației fiind: *btn\_back.cs, btn\_click.cs, btn\_virtual\_click.cs, CameraFocusController.cs, menu\_controller.cs, text\_blinkerdectection.cs* și *text\_makeDetection.cs*.

## 4.1 Scriptul btn\_back.cs

Pentru ca utilizarea aplicației să fie mai intuitivă, am folosit acest script pentru a permite întoarcerea la meniul principal dintr-un mod de scanare, sau închiderea aplicației dacă ne aflăm în meniu, la momentul apăsării butonului „Back” de pe dispozitivele Android, întrucât această funcționalitate nu există în mod implicit în Unity.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.SceneManagement;

public class btn\_back : MonoBehaviour {

void Update()

{

if (Input.GetKeyDown(KeyCode.Escape))

SceneManager.LoadScene("scene\_menuResponsive");

}

}

## 4.2 Scriptul btn\_click.cs

Acest script are rolul de a afișa un anumit panou descriptiv la momentul apăsării unui buton în modul Partial AR.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

public class btn\_click : MonoBehaviour {

public GameObject imagePanel;

public Text text;

public int counter=0;

// Use this for initialization

void Start () {

imagePanel.SetActive(false);

}

public void swap\_context()

{

counter++;

if (counter % 2 == 1)

{

imagePanel.SetActive(true);

}

else

{

imagePanel.SetActive(false);

}

}

}

## 4.3 Scriptul btn\_virtual\_click.cs

Acest script, similar celui precedent, are rolul de a afișa un anumit panou descriptiv la acționarea unui buton în modul Immersive AR.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using Vuforia;

public class btn\_virtual\_click : MonoBehaviour, IVirtualButtonEventHandler {

public GameObject VirtualButton;

public GameObject Panel;

// Use this for initialization

void Start () {

VirtualButton.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);

Panel.SetActive(false);

}

public void OnButtonPressed(VirtualButtonAbstractBehaviour vb)

{

Panel.SetActive(true);

Debug.Log("VB Pressed");

}

public void OnButtonReleased(VirtualButtonAbstractBehaviour vb)

{

Panel.SetActive(false);

Debug.Log("VB Released");

}

}

## 4.4 Scriptul CameraFocusController.cs

Deoarece am întâmpinat dificultăți cu focalizarea camerei în perioada de început a dezvoltării aplicației, a fost necesar un astfel de script care să permite focalizarea camerei în mod automat.

using UnityEngine;

using Vuforia;

public class CameraFocusController : MonoBehaviour

{

void Start()

{

var vuforia = VuforiaARController.Instance;

vuforia.RegisterVuforiaStartedCallback(OnVuforiaStarted);

vuforia.RegisterOnPauseCallback(OnPaused);

}

private void OnVuforiaStarted()

{

CameraDevice.Instance.SetFocusMode(

CameraDevice.FocusMode.FOCUS\_MODE\_CONTINUOUSAUTO);

}

private void OnPaused(bool paused)

{

if (!paused) // resumed

{

// Set again autofocus mode when app is resumed

CameraDevice.Instance.SetFocusMode(

CameraDevice.FocusMode.FOCUS\_MODE\_CONTINUOUSAUTO);

}

}

}

## 4.5 Scriptul menu\_controller.cs

Acesta este scriptul responsabil cu funcționarea meniului principal.

using UnityEngine;

using UnityEngine.SceneManagement;

public class menu\_controller : MonoBehaviour {

public void FullAR()

{

SceneManager.LoadScene("scene\_5Responsive");

}

public void PartialAR()

{

SceneManager.LoadScene("scene\_4Responsive");

}

public void Quit()

{

Debug.Log("Successfully quit");

Application.Quit();

}

private void Update()

{

if (Input.GetKeyDown(KeyCode.Escape))

Application.Quit();

}

}

## 4.6 Scriptul text\_blinker\_detection.cs

Utilizând câteva modificări ale scriptului *DefaultTrackableEventHandler.cs* din Vuforia pentru a obține numele mărcii autoturismului, acest script modifică textul din panoul descriptiv asociat butonului de semnalizator din modul Partial AR pentru a ilustra conducătorului auto, pe baza mărcii autoturismului, dacă comutatorul pentru faruri se află pe maneta de semnalizare, iar dacă nu, să îi faciliteze găsirea comutatorului respectiv.

using UnityEngine;

using UnityEngine.UI;

using Vuforia;

public class text\_blinkerdetection : MonoBehaviour {

public Text text;

void Update()

{

if (DefaultTrackableEventHandler.carMake == "chevrolet")

text.text = "This is the blinker switch. Push it " +

"upwards to signal the right side or push it " +

"downward to signal the left side. This" +

"is also the headlights switch; in order" +

"to turn them on, rotate the switch " +

"counter-clock wise twice. To turn " +

"them off, rotate the switch clock-wise" +

"twice.";

else

text.text = "This is the blinker switch. Push it " +

"upwards to signal the right side or push it " +

"downward to signal the left side. The " +

"headlights switch is placed below the vent " +

"that is placed to the left of this switch";

}

}

## 4.7 Scriptul text\_make\_detection.cs

Acest script este utilizat în modificarea textului despre marca autoturismului în funcție de logo-ul recunoscut.

using UnityEngine;

using UnityEngine.UI;

using Vuforia;

public class text\_makeDetection : MonoBehaviour {

public Text text;

// Update is called once per frame

void Update () {

text.text = "This " + DefaultTrackableEventHandler.carMake + " has \nbeen detected";

}

}

## 4.8 Exemplu de instanțiere a variabilelor publice în Unity

Unity permite instanțierea la runtime a variabilelor publice, prin asignarea componentelor la variable utilizând drag-and-drop, după cum se poate observa cu panoul pentru volanul marca Chevrolet și panoul ce indică faptul că nu a fost recunoscut niciun obiect în figura 9.

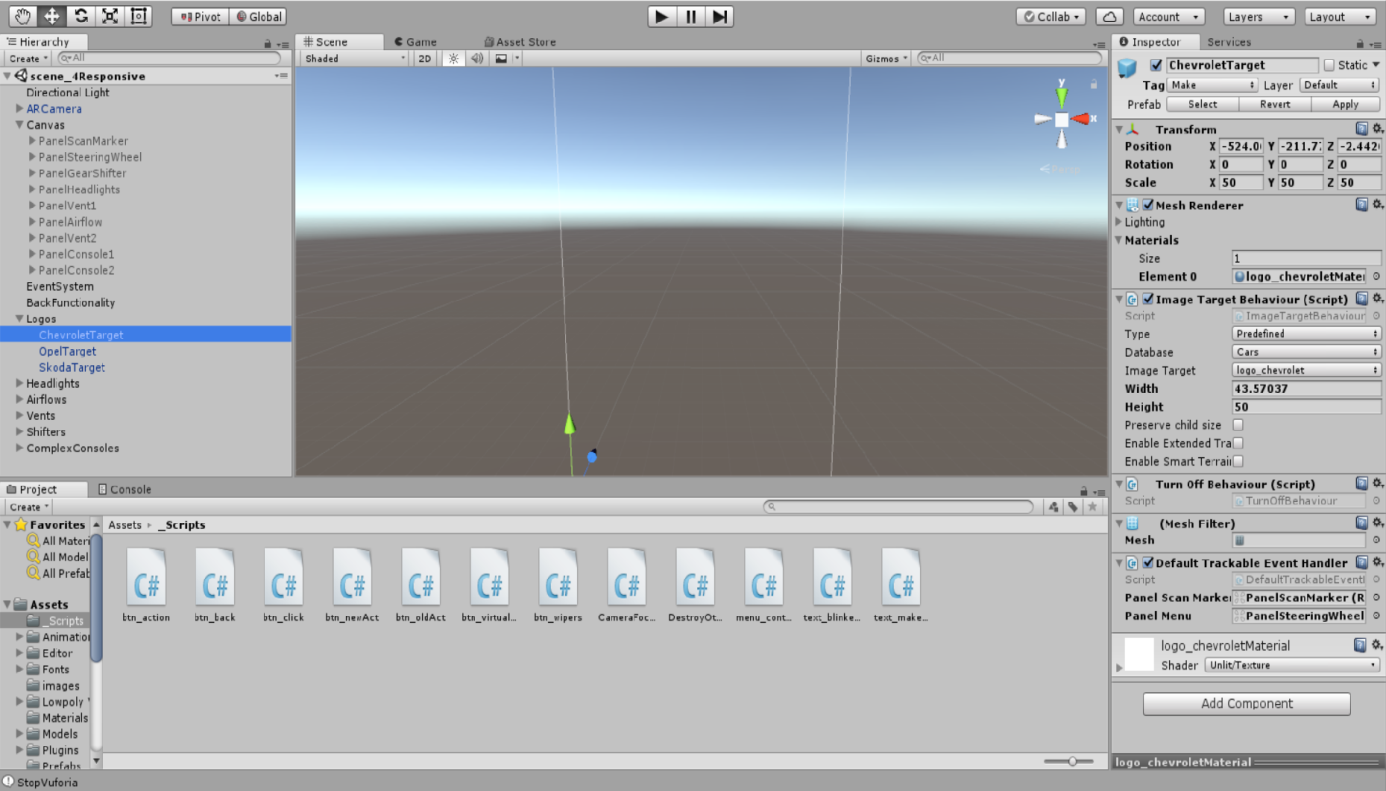


Fig. 9 Exemplu de instanțiere în Unity

# Posibilități de dezvoltare ulterioară

Consider că, pentru dezvoltarea ulterioară a aplicației, cele mai importante îmbunătățiri ce pot fi aduse sunt următoarele:

* portarea către iOS pentru ca aplicația să poate fi folosită și pe iPhone
* creșterea numărului de comutatoare ce pot fi recunoscute
* posibilitatea ca, la recunoașterea mărcii autovehiculului, utilizatorul să poate introduce anul, modelul și anul de fabricație al mașinii și să primească o listă cu toate funcționalitățile existente în autoturismul respectiv

# Concluzii finale

Având în vedere hardware-ul recent dezvoltat pentru AR, precum și interesul ridicat în acest domeniu, Realitatea Augmentată poate fi considerată drept următorul pas în evoluția

tehnologiei. În opinia mea, consider că aplicația Driving Assistant, în forma pe care o are la momentul prezentării proiectului, este o exemplificare a posibilităților acestui domeniu și în același timp un punct de plecare pentru diversificarea AR prin implementarea de noi funcționalități precum interacționarea cu hologramele (în acest moment, Hololens este printre singurele dispozitive hardware care permite interacționarea cu holograme)**,** plecând de la interacționarea cu butoanele virtuale ilustrate în acest proiect.

De asemenea, întrucât aplicația a fost gândită astfel încât orice funcționalitate sau abordare nouă să fie ușor integrată, extinderea acesteia ar putea duce la progrese în dezvoltarea domeniului.

# Biliografie

AR

* <https://en.wikipedia.org/wiki/Augmented_reality>
* <https://trends.google.com/trends/explore?date=all&q=augmented%20reality>
* <https://en.wikipedia.org/wiki/ARToolKit>

Unity

* <https://unity3d.com/what-is-a-game-engine>
* <https://en.wikipedia.org/wiki/Unity_(game_engine)>
* <https://digit.hbs.org/submission/unity-engine-a-unicorn-powering-the-video-game-and-vr-ar-economy/>

Vuforia

* <https://en.wikipedia.org/wiki/Vuforia_Augmented_Reality_SDK>
* <https://library.vuforia.com/articles/Training/Extended-Tracking.html>
* <https://library.vuforia.com/articles/Training/Image-Target-Guide.html>
* <https://library.vuforia.com/articles/Solution/Camera-Focus-Modes>

HoloLens

* <https://en.wikipedia.org/wiki/Microsoft_HoloLens>
* <https://www.digitaltrends.com/computing/microsoft-research-hololens-virtual-touchscreen-mrtouch/>

.

# Anexe

Anexa 1

