



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

[PROTOTIP DE REALITAT AUGMENTADA PER UNA EXPOSICIÓ MUSEÍSTICA]

**A Degree Thesis
Submitted to the Faculty of the
Escola Tècnica d'Enginyeria de Telecomunicació de
Barcelona
Universitat Politècnica de Catalunya
by
Elisabet Gutiérrez Oliveira**

**In partial fulfilment
of the requirements for the degree in
*AUDIOVISUAL SYSTEMS ENGINEERING***

Advisor: Josep Ramón Casas Pla

Barcelona, June 2016



Abstract

In this project we examine the concept of augmented reality (AR), where virtual 3D objects are integrated in a real environment, so they keep their qualities of orientation, rotation and scale. This report describes applications for education, focusing on Mesozoic era for the children to learn more about the age of dinosaurs. This document describes how we explored the capabilities of augmented reality using an RGB viewer to move it into a functional application for a museum with the RGBD sensor. We evolve from the conception of the idea, the exploration of programs and most used RGBD sensors, to the result of several functional application prototypes, addressing issues such as registration or integration of the AR object by an RGBD sensor.

Resum

En aquest projecte s'examina el camp de la realitat augmentada (en anglès, Augmented Reality), on els objectes virtuals en 3D estan integrats en un entorn real, pel que mantenen les seves qualitats d'orientació, rotació i escala. Aquí, es descriuen aplicacions destinades a l'educació, centrades en l'època Mesozoica per a que els nens aprenguin més sobre l'era dels dinosaures. Aquest document descriu com hem exportat les capacitats de la realitat augmentada mitjançant càmeres RGB per traslladar-ho en una aplicació funcional per un museu amb un sensor RGBD. L'informe analitza des de la concepció de la idea, la exploració dels programes i sensors RGBD més utilitzats, fins al resultat de diferents prototips d'aplicacions funcionals, tractant temes com el registre o la integració de l'objecte de realitat augmentada a través d'un sensor RGBD.

Resumen

En este proyecto se examina el campo de la realidad aumentada (en inglés Augmented Reality), donde los objetos virtuales en 3D están integrados en un entorno real, por lo que mantienen sus calidades de orientación, rotación y escala. En él, se describen aplicaciones destinadas a la educación, centradas en la época Mesozoica para que los niños aprendan más sobre la era de los dinosaurios. Este documento describe cómo hemos explorado las capacidades de la realidad aumentada mediante una cámara RGB para trasladarlo a una aplicación funcional para un museo con un sensor RGBD. El informe analiza desde la concepción de la idea, la exploración de los programas y sensores RGBD más usados, hasta el resultado de varios prototipos de aplicaciones funcionales, tratando temas como el registro o la integración del objeto de realidad aumentada mediante un sensor RGBD.

Agraïments

M'agradaria agrair tot el suport i seguiment donat pel meu tutor, Josep R. Casas, que tot i ser un projecte de caire més exploratori ja que hem utilitzat programes atípics al grau de sistemes audiovisuals.

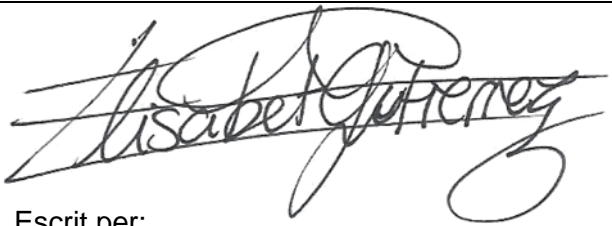
També voldria agrair a l'Eduard Pallàs, estudiant del grau de Ciències i Tecnologies de Telecomunicació, ja que gràcies als resultats del seu projecte hem pogut combinar ambdós projectes per desenvolupar una aplicació molt interessant i creativa.

Historial de revisió i registre d'aprovació

Revisió	Data	Propòsit
0	15/04/2016	Creació del document
1	26/06/2016	Revisió del document
2	27/06/2016	Entrega del document

LLISTA DE DISTRIBUCIÓ DE DOCUMENTS

Nom	e-mail
Elisabet Gutiérrez Oliveira	elisabet.gutierrez@alu-etsetb.upc.edu
Josep Ramon Casas Pla	josep.ramon.casas@upc.edu

 Escrit per:		Revisat i aprovat per:	
Data	25/06/2016	Data	27/06/2016
Nom	Elisabet Gutiérrez	Nom	Josep R. Casas
Posició	Autor/a del projecte	Posició	Supervisor del projecte

Taula de continguts

[PROTOTIP DE REALITAT AUGMENTADA PER UNA EXPOSICIÓ MUSEÍSTICA]	0
Abstract	1
Resum	2
Resumen	3
Agraïments	4
Historial de revisió i registre d'aprovació	5
Taula de continguts	6
Llista de Figures:	8
Llista de Taules:	9
1. Introducció	10
1.1. Objectius	11
1.2. Requeriments i especificacions	11
1.3. Pla de treball	11
1.4. Incidències	12
2. Estat de l'art	13
2.1. Actualitat	13
2.1.1. Quiver	13
2.1.2. Google Glass	14
2.2. Aplicacions	14
2.2.1. Microsoft Hololens	14
2.2.2. Occipital Bridge Engine	14
2.2.3. Project TANGO	14
2.2.4. Magic Leap	15
3. Metodologia	16
3.1. Definir la idea del projecte	16
3.2. Exploració de Software 3D	17
3.3. Aprenentatge de Unity 3D	17
3.4. Exploració de sensors pel registre	19
3.4.1. Structure Sensor	20
3.4.2. Kinect	20
3.5. Creació / Modelat d'una figura 3D	20
3.5.1. Registre i Integració: Unity 3D + ARToolkit	21

3.5.2. Registre i Integració: Unity 3D + Kinect	22
3.5.3. Figura 3D: Model final	23
3.6. Integració amb un detector de punts d'ancoratge	25
4. Resultats	26
4.1. Resultats preliminars	26
4.2. Resultats del prototip d'aplicació	30
5. Pressupost	33
6. Conclusions i línies futures:	34
Bibliografia:.....	35

Llista de Figures:

Figura 1 Exemple de realitat augmentada complexa [2]	10
Figura 2 Exemple de realitat augmentada elemental [3]	10
Figura 3 Digrama Gantt del projecte	12
Figura 4 Dos exemples d'AR de l'aplicació QuiverVision [12]	13
Figura 5 Objectiu que preten crear Magic Leap [17]	15
Figura 6 Esquema del projecte	16
Figura 7 Escena inicial de l'aprenentatge amb Unity.....	18
Figura 8 Structure Sensor - Com s'utilitza.....	20
Figura 9 Sensor RGBD: Kinect	20
Figura 10 Funcionament intern d'ARToolkit a Unity [26]	21
Figura 11 Exemple d'esquelet detectat amb Kinect	22
Figura 12 Visualització del triceratops com a malla per veure el bípede intern	23
Figura 13 Eixos de coordenades de la càmera d'un Kinect [30].....	25
Figura 14 Escanejat sala reconstrucció ICP	27
Figura 15 Escanejat taula de treball del ICP	27
Figura 16 Escanejat de l'esquelet del triceratops del ICP	28
Figura 17 Escanejat del crani del triceratops del ICP.....	28
Figura 18 Resultat de Unity amb ARToolkit amb el model cedit pel ICP	29
Figura 19 Resultat intermedi. Detecció de l'esquelet	30
Figura 20 Resultat primera aplicació. Triceratops ens camina per la mà	31
Figura 21 Resultat aplicació final. Triceratops a la sala	32



Llista de Taules:

Taula 1 Milestones del projecte	11
Taula 2 Efecte de partícules	18
Taula 3 Possibles unions d'elements amb Unity	19
Taula 4 Com aplicar textura a una figura a partir d'arxius .tga	24
Taula 5 Pantalles del resultat final de l'aprenentatge amb Unity	26
Taula 6 Resultat de Unity amb ARToolkit amb l'escanejat del ICP	29
Taula 7 Resultat segona animació. Som un triceratops a la jungla	32

1. Introducció

En els últims anys, cada vegada es sent més a parlar d'una tecnologia anomenada realitat augmentada [1]. Coneixem per realitat augmentada (o AR de l'anglès *Augmented Reality*) la visió a través d'un dispositiu tecnològic del món físic real combinat amb elements virtuals, obtenint així una realitat mixta que no suprimeix la realitat sinó que l'augmenta amb nous elements que aporten nova informació.

El concepte de realitat augmentada és molt discutit, ja que per abús del llenguatge o per tècniques comercials, s'anomena realitat augmentada a diferents tecnologies mitjançant dispositius tecnològics tant siguin més elementals o més complexes. Les tècniques més complexes de realitat augmentada tenen en compte la geometria de l'escena real on es registren i són dotades d'escala, rotació i orientació. D'aquesta manera si movem el nostre dispositiu veurem des d'una altra perspectiva l'element virtual però aquest mantindrà la seva relació d'escala, rotació i orientació amb l'escena. En canvi, altres aplicacions més elementals només tenen en compte el registre ja sigui auditiu, visual o a partir de coordenades GPS i superposen l'element virtual, sovint sense tenir en compte l'escena real i es produeix l'efecte que l'element virtual sembla sobreposat a la pantalla del dispositiu.

Observem doncs les dues imatges que trobem a continuació:

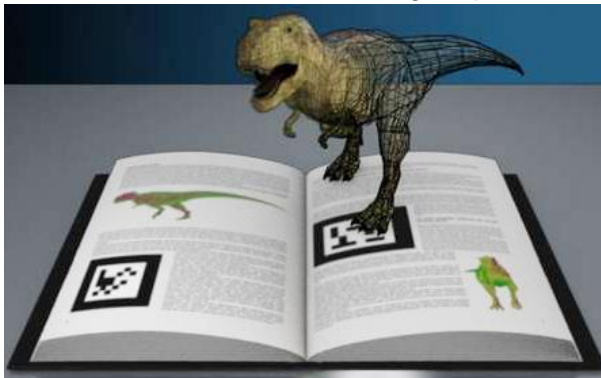


Figura 1 Exemple de realitat augmentada complexa [2]



Figura 2 Exemple de realitat augmentada elemental [3]

En la Figura 1, quan es detecta el marcador (quadre de vora negra amb fons blanc i formes negres al interior) el sistema de realitat augmentada fa aparèixer un element nou, en aquest cas, un dinosaure amb escala, rotació i orientació concretes. Si poguéssim el dispositiu, el veuríem des d'un altre angle. En canvi, en la Figura 2, els elements virtuals són 2D i es sobreposen a la pantalla i, independentment de la posició en què veiem l'edifici, sempre veurem la icona amb la mateixa orientació.

A mesura que hem desenvolupat el coneixement en el hardware dels dispositius també ho hem fet en el software per crear noves aplicacions. Aquest és doncs, el cas de la realitat augmentada que a dia d'avui té utilitats en mons tan diversos com educació, entreteniment, publicitat o turisme. Es es va decidir utilitzar aquesta tecnologia, perquè ofereix aquesta qualitat polifacètica tant en els aspectes de geometria 3D com en els de creativitat de l'animació. D'aquí va sorgir l'objectiu del projecte, **crear una aplicació de realitat augmentada per una exposició museística.**

Amb aquest objectiu, es van obrir diverses vies de comunicació amb diferents museus i finalment va sorgir una col·laboració amb l'Institut Català de Paleontologia (ICP) [4] amb el que ens vam reunir i presentar l'esquema principal que constituïa aquest TFG: crear

un prototip de realitat augmentada per que els nens petits i els no tan petits s'acostessin al món dels dinosaures i aprenguessin nous conceptes de manera divertida.

1.1. Objectius

Els objectius que es van proposar per aquest projecte són els següents:

- Explorar les eines de treball disponibles per captura d'escenes i generació d'animacions AR
- Crear / dissenyar una figura en 3D (temàtica del museu)
- Obtenir el model d'una sala a través d'un sensor RGBD.
- Registrar una figura virtual en un punt determinat del espai
- Veure els resultats de la inserció de realitat augmentada en aplicacions per sensors RGBD (online).
- Dotar de perspectiva i escala la figura per tal que es vegi correctament a través del sensor a partir de la integració amb un detector de punts d'ancoratge. (Combinació del projecte de l'Eduard Pallàs)
- Veure els resultats de la inserció de realitat augmentada en la sala (offline).

1.2. Requeriments i especificacions

- Dispositiu RGBD com a visor de l'aplicació generada.
 - o Kinect Sensor de Microsoft amb les especificacions donades pel fabricant:
 - Sensibilitat de profunditat: 512 x 424
 - Framerate: 30Hz
 - Field of View: 70 x 60
- Creació d'una figura 3D amb temàtica museu escollit:
 - o Partim del model cedit pel museu, es tracta d'un triceratops [5] en 3D a mida real amb la seva textura.
- Generació de l'aplicació:
 - o Plataforma on integrarem les dades obtingudes del sensor RGBD amb la figura 3D del dinosaure.

1.3. Pla de treball

Els diferents Workpackages en els que hem dividit el nostre treball es poden consultar a l'Annex A. En aquest apartat llistem només les fites (milestones) marcats en el projecte i el diagrama de Gantt per donar una visió de perspectiva.

Milestones:

WP#	Task#	Short title	Milestone / deliverable	Date (week)
1	3	Aprentatge Unity3D	"Hello World" Unity3D	20/03/2016
2	1	Extreure App	Plataformes / SO	30/03/2016
3	5	Model sala	Núvol de punts de la sala	05/04/2016
4	5	Figura 3D	Prototip figura 3D amb sensor RGB	26/04/2016
4	5	Figura 3D	Figura 3D final amb sensor RGBD	02/06/2016
5	3	Registre	Detecció de punt d'ancoratge	02/06/2016
6	2	Integració	Inserir figura al núvol de punts	09/06/2016

Taula 1 Milestones del projecte

El diagrama de Gantt obtingut a partir dels workpackages de l'Annex A.

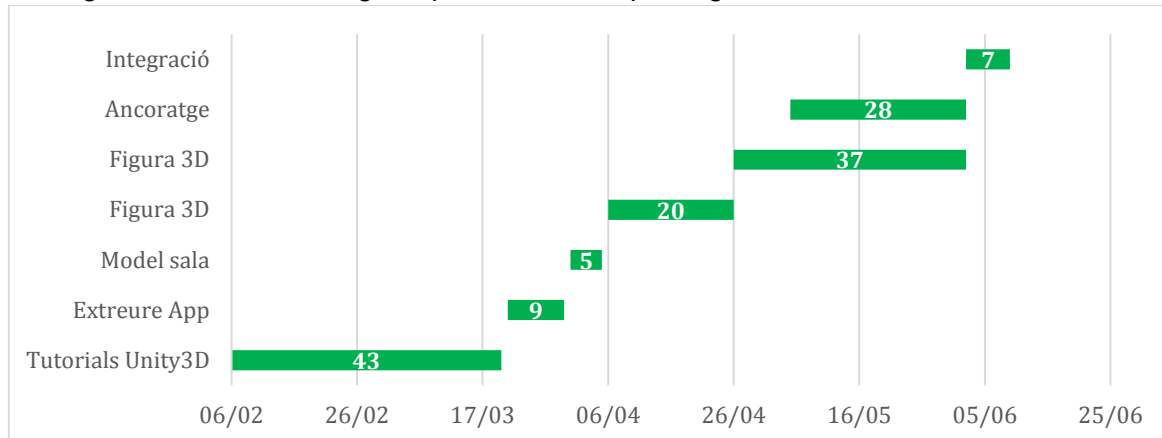


Figura 3 Digrama Gantt del projecte

1.4. Incidències

En aquest projecte ens hem trobat dues incidències molt diferenciades que exposem a continuació.

La primera, va ser aprendre l'ús del model complet d'un triceratops en 3D, que el museu ICP ens va proporcionar per treballar com a objecte de realitat augmentada.

El model complet d'objecte del ICP, és un model de triceratops de carn i ós a mida real, no com el que està disponible a l'exposició que és només l'esquelet, també a escala real.

El model està compost de diferents formats:

- Triceratops_animations.ma – Model amb esquelet i animació
- Triceratops.tga – Textura
- Triceratops_spec.tga – Textura
- Triceratops_normal.tga – Textura
- Triceratops.3DS – Model amb esquelet associat
- Triceratops.blend – Model sense esquelet associat
- Triceratops.obj - Model sense esquelet associat
- Triceratops.mtl – Textura de l'arxiu obj
- Triceratops_animations.fbx – Model amb esquelet i animació

Vam començar a treballar amb el model en format .3DS (3DS MAX) ja que era el primer model que vam trobar amb esquelet associat i que per tant el podíem animar. No obstant, més tard vam veure que el model en format .ma (Maya) era més complet, ja que, a part del model amb esquelet associat, ja tenia pre-definides unes animacions i només havíem d'extreure els *frames* desitjats de l'animació completa. Per altra banda, el model en format .fbx (Filmbox) era una exportació prèvia del document original Maya però l'exportació no s'havia fet correctament i només teníem l'ós de la pelvis amb animació en comptes del cos complet.

La segona incidència que ens vam trobar, va ser el pas de integració. És a dir, vam passar per un pas entremig on veiem una aplicació generada amb Unity (animació d'un triceratops) creada conjuntament amb ARToolkit, gràcies al *package* existent d'ARToolkit per Unity, que permet veure les figures 3D a través d'una càmera RGB gràcies a un marcador. Aquesta sinèrgia entre Unity i ARToolkit s'havia d'intentar extrapolar a un sistema que treballés amb un sensor RGBD.

Donada la complexitat que generava el model triat (Structure), i el fet d'haver de programar per iOS, es va optar per treballar amb Windows i utilitzar Kinect.

2. Estat de l'art

Primerament, abans d'explicar la metodologia d'aquest projecte és necessari analitzar l'estat actual de realitat augmentada i explicar breument les aplicacions i els dispositius que trobem actualment al mercat.

2.1. Actualitat

Actualment comptem amb aparells físics que mitjançant un software fan visible la informació virtual existent registrada en l'entorn, o bé permeten crear-la, per gaudir d'una experiència de realitat augmentada des dels nostres ordinadors, tauletes digitals o smartphones.

Aquesta tecnologia porta molts anys en funcionament tot i que el seu recent auge és gràcies, en gran part, al Mobile World Congress, que ha mediatitzat les presentacions de noves aplicacions i prototips. No obstant, trobem els seus inicis l'any 1962, quan Morton Heilig va crear un simulador, el Sensorama [6] amb imatges, sons, vibracions i olors. Tot i que fins el 1992 no se li va atorgar el terme de realitat augmentada com a tal.

Des d'aleshores podem remarcar el KARMA [7], de l'any 1994, un dispositiu per arreglar màquines complexes sobreimprimint el manual de l'equip sobre els motors i que va desenvolupar en una eina per troballes arqueològiques. També podem destacar la contribució de Hirokazu Kato del 1999, en desenvolupar el ARToolKit al HitLab [8] que més tard es va integrar amb Adobe Flash per gaudir de realitat augmentada en navegadors web.

2.1.1. Quiver

De HIT Lab en va néixer Puteko Limited [9] una spin-off anomenada HIT Lab NZ [10]. Aquesta spin-off va desenvolupar una aplicació de realitat augmentada pel cas de càmeres 2D anomenada QuiverVision [11]. Aquesta aplicació genera animacions de realitat augmentada d'alta qualitat artística i tècnica en un entorn d'entreteniment per a infants. L'aplicació s'executa sobre una tauleta i aprofita la càmera integrada per mostrar la realitat augmentada. A la **¡Error! No se encuentra el origen de la referencia.**a 4 es mostren algunes imatges d'aquesta aplicació. El descobriment d'aquesta aplicació ha sigut l'increment que va motivar el projecte.



Figura 4 Dos exemples d'AR de l'aplicació QuiverVision [12]

No obstant, és ara quan la realitat augmentada vol assolir el seu major repte: ser part de la vida diària de les persones, abaratir costos i reduir la mida dels dispositius portàtils a més de simplificar el seu ús. Com al 2012 que Google es va llançar al disseny d'unes ulleres que crearien la primera realitat augmentada comercialitzada.

2.1.2. Google Glass

Google Glass [13] és un dispositiu de visualització de realitat augmentada 2D en forma d'ulleres desenvolupat per Google. El llançament d'aquest dispositiu va ser molt mediàtic pel gran ventall de possibilitats que obrien davant seu, fomentant la creativitat i obtenint com a resultat noves aplicacions.

El seu propòsit era el de mostrar informació disponible pels usuaris de telèfons intel·ligents sense utilitzar les mans, permetent també l'accés a Internet mitjançant ordres de veu.

No obstant, els exemples que hem explicat utilitzen com a visor de realitat augmentada una càmera RGB pel que obtenim la realitat augmentada en un entorn projectat, 2D. El nostre projecte vol fer un pas més enllà i utilitzar com a visor una càmera RGBD, com ho són el Kinect o el Structure Sensor per exemple.

2.2. Aplicacions

La investigació i el coneixement més profund dels sensors RGBD ha permès que els desenvolupadors creïn aplicacions molt creatives i enfocades a un entorn comercial per tal que la societat tingui coneixement de la realitat augmentada.

2.2.1. Microsoft HoloLens

Les HoloLens són unes smartglasses de realitat mixta desenvolupades per Microsoft. La seva popularitat ve donada per ser el primer ordinador en córrer la plataforma de Windows Holographic sota el sistema operatiu de Windows 10. La diferència essencial respecte les Google Glass, vistes anteriorment, és que les HoloLens integren un sensor RGBD sent les successores del Kinect.

2.2.2. Occipital Bridge Engine

De la mateixa manera que HoloLens són les successores del Kinect per Windows per iOS també tenim Bridge Engine [14] desenvolupat per Occipital. Consisteix en un pont que ofereix capacitats innovadores de realitat mixta desenvolupat en un senzill software. Compatible amb Structure Sensor i Structure Core SDK, per la majoria de hardware i software d'experiències de realitat mixta convincents. [15]

2.2.3. Project TANGO

El projecte Tango [16] és una plataforma de Google que utilitza la visió per ordinador per permetre que els dispositius mòbils, tals com smartphones o tauletes digitals, detectin la seva posició en relació al món que els envolta sense necessitat de GPS o altres senyals. Això permet als desenvolupadors d'aplicacions crear experiències pels usuaris com navegació d'interiors, cartografia 3D, el mesurament dels espais físics, reconeixement d'entorns coneguts, realitat augmentada i finestres a mons virtuals en 3D.

GLASS



occipital



2.2.4. Magic Leap



Magic Leap [17] també l'hem inclòs en aplicacions tot i que no està enfocat en un entorn comercial on venguin un producte, sinó que, es tracta d'una start-up dedicada a la recerca per aconseguir el seu producte objectiu mitjançant un finançament extern. No obstant, el producte que proposen és molt interessant, original i creatiu basat en càmeres plenòptiques que des del punt de vista de la realitat augmentada, com es veu a la imatge que hi ha a continuació, la figura 5, deixa entreveure que serà espectacular:



Figura 5 Objectiu que preten crear Magic Leap [17]

3. Metodologia

Per explicar la metodologia seguida en aquest projecte hem preparat un gràfic a continuació. L'explicació anirà passant les diferents quadres per seguir un ordre lògic de l'adquisició dels coneixements.

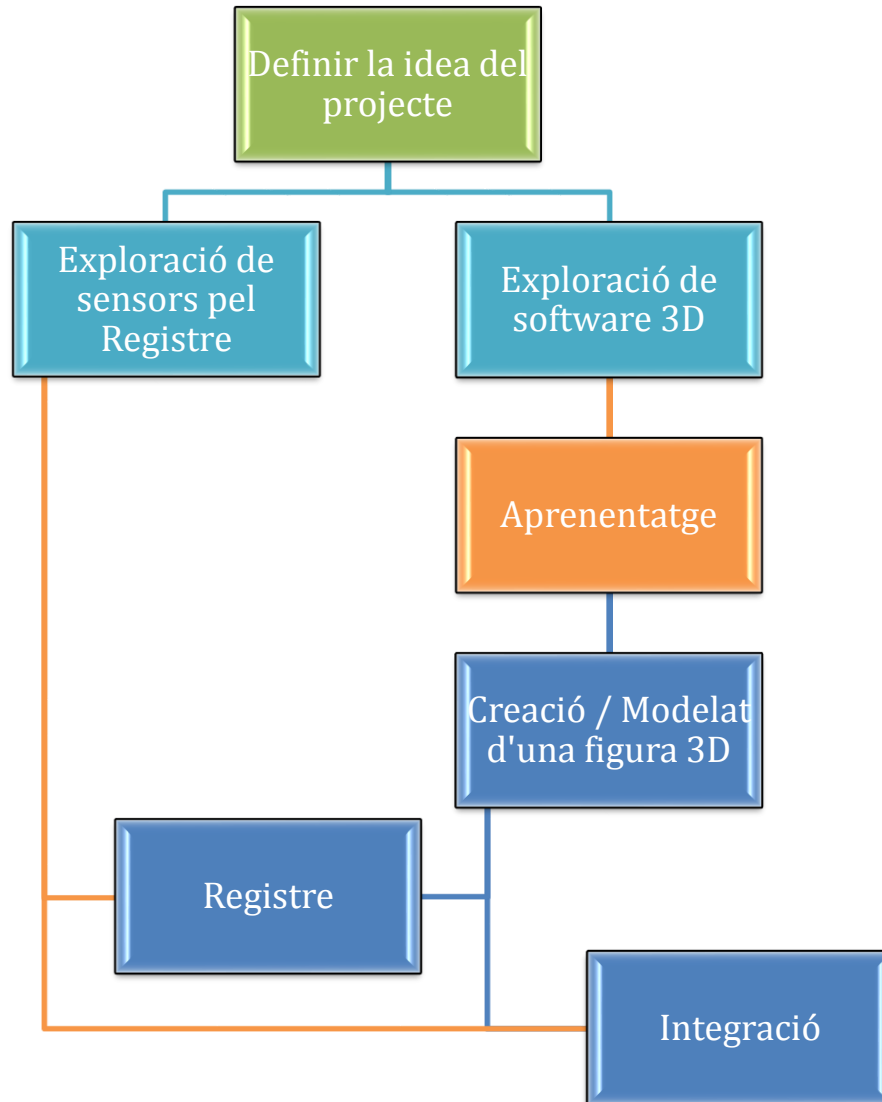


Figura 6 Esquema del projecte

3.1. Definir la idea del projecte

La idea d'aquest projecte té un factor motivador que és l'aplicació Quiver. Es va voler recrear un prototip amb el seu funcionament però amb un sensor RGBD en comptes d'una càmera RGB com fa Quiver. Aquesta innovació suposaria dotar l'element virtual de coneixement de l'entorn pel que l'animació es restringiria a posicions disponibles on no hi haguessin obstacles. Amb una càmera RGB, seria molt difícil interpretar la geometria de

l'escena per a evitar obstacles. Dit d'una altra manera, la càmera RGB només detecta el marcador sobre el que s'ha d'inserir l'objecte virtual, i no permet analitzar fàcilment la resta de l'escena. En canvi, un sensor RGBD permet capturar de manera nativa la geometria 3D de l'escena, i tenir-la en compte (per exemple, considerar la presència d'obstacles o transicions en la geometria) per inserir i animar l'objecte 3D en un entorn llunyà d'un marcador definit.

3.2. Exploració de Software 3D

Software per dissenyar 3D n'hi ha molts i cadascun potencia diferents aspectes, pel que havíem de decidir quin era el més adient pel projecte i podria aportar més facilitats per fer la posterior integració.

Es van posar sobre la taula els següents programes:

- OGRE 3D (Object-Oriented Graphics Rendering Engine): És un motor de renderitzat 3D orientat a escenes, escrit en C++ i software lliure.
 - o Va ser descartat ja que es va detectar poc versàtil alhora de treballar en un entorn iOS, pel que seria impossible treballar amb el sensor de Structure, per exemple.
- Blender: És un programa informàtic multi plataforma, dedicat especialment al modelat, il·luminació, renderitzat, animació i creació de gràfics tridimensionals.
 - o Va ser descartat en un primer moment, per ser un programa d'alt nivell, ja que requereix invertir molt temps per aprendre a utilitzar-lo al màxim rendiment.
- Unity 3D: És un motor de videojoc multi plataforma, permet desenvolupar jocs per Windows, OS X, Linux, iPad, Android, entre altres. Respecte altres programes de creació de videojocs ofereix un gran avantatge: la integració per Nvidia que suporta en temps real malles arbitràries i sense pell, ray casts gruixuts i capes de col·lisió, a part, és compatible amb 3DS MAX, Maya i Blender.
 - o S'escull per ser el més versàtil i el que proporciona un millor entorn per animacions. Es convertirà en l'eina principal on fer la integració del nostre projecte.

3.3. Aprentatge de Unity 3D

Donat que es va escollir Unity com a plataforma per desenvolupar les nostres aplicacions, primer havíem d'aprendre a utilitzar les seves funcionalitats, extrapolar els coneixements i obtenir així el nostre objectiu. Per tant, el primer pas va ser un treball exploratori, consistent en un seguit de vídeos explicatius d'aprenentatge per Unity 3D. L'autor del quals, és Asbjørn Thirslund [18] o anomenat Brackeys com a desenvolupador. [19]

L'objectiu pràctic d'aquest aprenentatge és crear un videojoc on nosaltres controlem una pilota i l'hem de portar per diferents escenes, esquivant els obstacles i recollint monedes que es transformen en punts fins acabar el joc.

El videojoc d'aprenentatge, s'ha desenvolupat seguint els següent punts. Cada punt es centra en un concepte nou per aprendre:

1. Comencem amb les parts més bàsiques del videojoc. Creem l'entorn (pilota i plataforma) i dotem de qualitats físiques la pilota.

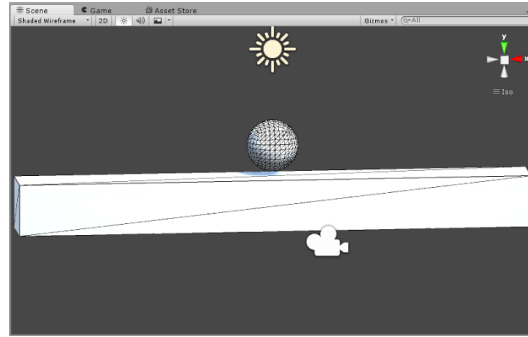





Figura 7 Escena inicial de l'aprenentatge amb Unity

2. Creació d'un script per fer saltar la pilota. Els scripts són funcions escrites en Java-Script o c-Sharp que s'associen a objectes de l'entorn perquè tinguin les habilitats desitjades. En el nostre cas, hem programat en Java-Script.
3. Script pel control de la càmera. Que segueixi la pilota en comptes de ser fixa.
4. Com aplicar textures als elements. Com fer les textures més realistes.
5. Il·luminació i joc d'ombres de l'escena. Script per reiniciar el joc quan la pilota cau de la plataforma.
6. Qualitat de l'ús. Unity, al ser un entorn multi plataforma que permet desenvolupar per la gran majoria de sistemes operatius i dispositius, permet escollir la qualitat en la que s'ha de mostrar en cada cas. És a dir, si l'obrim des d'un sobre taula jugarem en alta qualitat, però si juguem des d'un smartphone serà una qualitat mitja.
7. Disseny de col·leccionables. En la majoria de videojocs ens trobem amb elements que podem anar agafant. Com per exemple al Mario Bros les monedes. Recreació d'aquest col·leccionables pel nostre videojoc.
8. Afegir animació al element col·leccionable per tal que giri sobre si mateix.
9. Partícules. Igual que podem afegir esferes o cubs hi ha l'opció d'afegir partícules. Les partícules són una font de llumetes o espurnes editable, la qual podem dissenyar per tal d'aconseguir qualsevol efecte. En el nostre cas, volem que quan passi la pilota per la moneda es generin les partícules al fer desaparèixer la moneda.



Taula 2 Efecte de partícules

10. Creació d'un marcador que informi al jugador quantes monedes ha recol·lectat.
11. Dotar de musicalitat el videojoc. So quan agafem una moneda. So de "game over" quan cau la pilota de la plataforma, música ambiental, so de la pilota en contacte amb la plataforma.
12. Optimitzar els scripts per eliminar possible redundància i fer-los més eficients. Importació d'un nou model de plataforma a partir d'un model de Blender.
13. Creació del objecte "Music Manager" amb un script per tal que quan es reinici el joc la música segueixi per no escoltar sempre el mateix fragment.
14. Millorar la il·luminació, la rotació de la pilota sobre la plataforma i explicació de les restriccions que existeixen.

Fixed: Fixes	Spring: en game mode es veuen separades, però segueixen unides, efecte cadena.	Hinge: en game mode es veuen juntes, però segueix tenint l'efecte cadena, aquest cas, sense separació.
		

Taula 3 Possibles unions d'elements amb Unity

15. 16. Nova escena, per configurar el menú principal del joc.
17. CheckPoints. Punts de control, elements del joc amb script assignat que al passar a través seu amb la pilota guarda la posició, i en cas de caure de la plataforma en comptes de reiniciar el joc, començar a l'últim CheckPoint guardat.
18. 19. Obstacles. Creació d'un element enemic a la pilota.
20. Nova versió. Unity5 Què canvia? Capacitat d'arreglar automàticament el codi per adaptar-lo a la nova versió.
21. Millora de l'enemic a partir d'un model de Blender amb el mateix aspecte que el Goomba [20] de Mario Bros. En cas que la pilota col·lidís contra el *collider* del cos, es reinicia el joc. Si col·lideixi contra el *collider* del cap l'enemic mor.
 - a. Importància de Blender en quan a modelat. Hem de recuperar Blender, tot i haver-lo descartat per fer tasques concretes.
22. ¿Què passa quan la pilota "mor"? Efecte gerro trencat amb Blender: Crear una malla per l'esfera i escollir en quantes parts es divideix. Guardar-ho en format .blend i passar-ho directament a Unity.
23. EndPoints. Elements del jocs que al ser travessats per la pilota fan que passi a l'escena següent.
24. Com importar el joc. Com crear l'arxiu .exe per executar-lo des de sistema operatiu Windows.
25. Arxius per executar el joc des de Windows i Mac.
26. Executar el joc des de un entorn web (browser). WebGL [21] vs UnityWebPlayer.
 - a. Dóna prioritat a WebGL ja que UnityWebPlayer utilitza el NPAPI que Google Chrome va anunciar que retirar del cercador el passat 1 de setembre del 2015.

3.4. Exploració de sensors pel registre

Com hem comentat anteriorment, pel registre volem utilitzar sensors RGBD. Per tant cam estudiar els dos més utilitzats: el Structure Sensor, per iOS, i el Kinect per Windows.

3.4.1. Structure Sensor



Figura 8 Structure Sensor - Com s'utilitza

El Structure Sensor és un sensor RGBD desenvolupat per Occipital, amb la gran particularitat que la seva correcció de soroll és molt més ajustada. Per altra banda, s'ha concebut per iOS pel que només es pot utilitzar a través d'iPad i amb les diferents aplicacions disponibles al AppStore que ens permeten:

- Escanejat i mesura d'una habitació
- Creació de models 3D
- Jocs / Aplicacions de realitat augmentada

Es va triar com a plataforma d'ús per les opcions que oferia pel nostre projecte. Ja que el seu petit disseny ofereix més facilitat d'integració pensant en l'aplicació final. No obstant, el desenvolupament per Structure només es pot fer a través Xcode [22] i la programació amb iOS, un nou programa i un nou llenguatge que donada la longitud del projecte eren inviablés d'aprendre i dur a terme.

3.4.2. Kinect

KINECT



Figura 9 Sensor RGBD: Kinect

El dispositiu Kinect va sortir al mercat l'any 2015 i originalment va se concebut com a "controlador de joc lliure i entreteniment". El va desenvolupar Microsoft com a accessori perifèric per la videoconsola Xbox 360 i permet als usuaris controlar i interactuar amb la consola sense necessitat de tenir contacte físic amb un controlador de videojocs tradicional.

No obstant, és l'únic dispositiu electrònic amb més vendes en l'entorn de recerca i investigació que comercials (de fet, té un rècord Guinness [23]). En l'entorn de recerca es sol emprar independentment de la consola Xbox, gràcies al seu software lliure OpenNI que proporciona controladors (drivers) i una llibreria de reconeixement de gestos (SDK). Això el fa un producte versàtil i útil per desenvolupar aplicacions interactives que requereixen un sensor de profunditat.

3.5. Creació / Modelat d'una figura 3D

Un cop apresos els conceptes bàsic de Unity comencem amb el modelat d'una figura 3D. El primer pas és fer un prototip inicial amb visor RGB per comprendre el funcionament de registre de figures.

3.5.1. Registre i Integració: Unity 3D + ARToolkit

Unity té l'opció d'afegir més prestacions al programa a través del Asset Store. Aquí vam trobar que existia un Asset d'ARToolkit i el vam incloure. Un asset consisteix en un *package* per Unity que al descarregar-lo ens permet importar-lo directament en el programa i ens apareix a la pestanya d'assets disponible per utilitzar en aquell projecte. Si obrim un nou projecte, l'hem de tornar a importar.

A partir del *package* ARToolkit el que ens interessa són els scripts que associarem als objectes.

Pel nostre primer prototip ens hem basat en l'aprenentatge d'ARToolkit per Unity disponible en la documentació [24]. Seguint les instruccions que dona configurem una escena amb un controlador (ARToolkit), origen de visió de la càmera (Scene Root), Marcador (en el nostre cas, hem utilitzat el típic, Hiro pattern [25]), la figura virtual que volem veure a través de la càmera i la càmera.

Els scripts s'associen als elements esmentats seguint el funcionament descrit a continuació per tal que quan es mostra el patró de Hiro a la càmera aparegui el nostre element virtual.

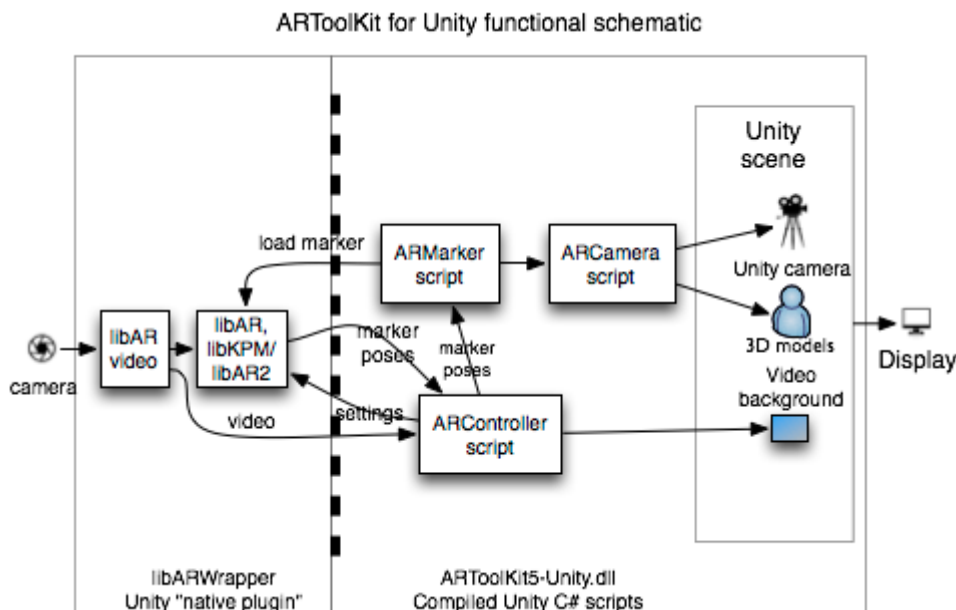


Figura 10 Funcionament intern d'ARToolkit a Unity [26]

- Controlador, anomenat: ARToolkit, li assignem la capa: AR Background i li associem els següents scripts:
 - o ARController Script: agafa la informació de la càmera a través del libAR vídeo i gràcies a les llibreries: libAR i libKPM/libAR2 llegeix la posició del marcador per passar-li al ARMarker la informació d'orientació, escala i rotació.
 - o ARMarker Script: Carrega el marcador que ha de veure la càmera. I obté la posició del marcador del ARController. Després passa els paràmetres d'orientació, rotació i escala per la figura virtual.
 - Hem d'assignar una etiqueta al marcador que haurà de coincidir amb l'etiqueta de l'element marcador. També hem d'assignar el patró Hiro que es el que veurà la càmera.

- Origen de visió de càmera, anomenat: Scene Root li assignem capa: AR Foreground, així tots els elements que descendeixin d'ell tindran la mateixa qualitat de capa, i li associem el script:
 - o AROrigin Script: Marca el punt (0,0,0) de l'escena per registrar l'element virtual.
- Marcador, dins del Scene Root i l'anomenem: Marker i l'hi assignem el següent script:
 - o ARTrackedObject Script: Aquest script ha de tenir la mateixa etiqueta de marcador que la que hem posat al script ARMarker. Aquest script permet registrar l'objecte virtual sobre el patró i que mantingui la posició per tal que si canviem la posició de la càmera puguem veure l'objecte virtual des d'un altre angle.

Aquí, dins de l'element marcador, col·loquem els objectes 3D per veure per pantalla a través del marcador.

- Dins del Scene Root també, inserim un element càmera, anomenat: Camera i amb el següent script associat:
 - o ARCamera Script: ha de permetre mostrar l'element virtual pel que hem de configurar una màscara de tall per tal que només mostri els elements amb la capa AR Foreground configurada.

3.5.2. Registre i Integració: Unity 3D + Kinect

El model final de la figura es va dur a terme amb Unity, però aquesta vegada requeríem un sensor RGBD pel que vam buscar a l'Asset Store de Unity i vam trobar un *package* anomenat *Asset Kinect MS-SDK*. Aquest *package* inclou quatre exemples d'aplicacions amb Kinect per Unity amb els seus corresponents scripts. Entre ells en destaquem l'aplicació d'*overlay*. *Overlay* prové de l'anglès que vol dir incrustar, permetent que al detectar una part del cos de referència (per exemple una mà), insereixi un objecte virtual en aquella posició amb l'escala i la rotació adequades.

Primer de tot, per entendre el funcionament bàsic de Kinect, quan et poses davant del sensor el primer que fa es detectar el cos de la persona i assignar-li un esquelet anomenat cubeman, com la Figura 11 que veiem a continuació, que consisteix en un seguit de cubs i unions. De manera que cada cub té un nom concret com right hand, head, spine,... i pertany a una part del nostre cos.

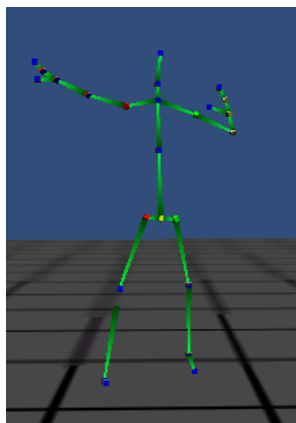


Figura 11 Exemple d'esquelet detectat amb Kinect

D'aquesta manera, assignem una part del cos on volem fer el *overlay*, la mà dreta per exemple, així quan la detecti incrustarà l'element virtual que serà el nostre model final.

3.5.3. Figura 3D: Model final

Aquesta vegada en comptes de treballar amb el model d'ossos escanejat del museu vam utilitzar el model cedit pel museu. Un model 3D d'un triceratops de mida real amb cos complet i textura.

Vam començar a treballar amb l'arxiu de format 3DS ja que l'arxiu en format .obj, .blend o el .fbx no tenien el cos del dinosaure connectat amb el seu esquelet intern pel que moure l'esquelet no implicava moure el cos.

Autodesk 3DS MAX



Autodesk 3DS MAX és un programa de la col·lecció d'Autodesk Inc. de creació de gràfics i animació 3D. Ens hem interessat per aquest programa ja que ens permet animar de forma senzilla models 3D a partir de figures bípedes, com és el cas del nostre Triceratops. Ja que si canviem la visualització de la pell i la passem a malla, l'esquelet de l'interior del triceratops és un bípede en la posició de quatre potes per a que encaixi com veiem a continuació a la Figura 12:

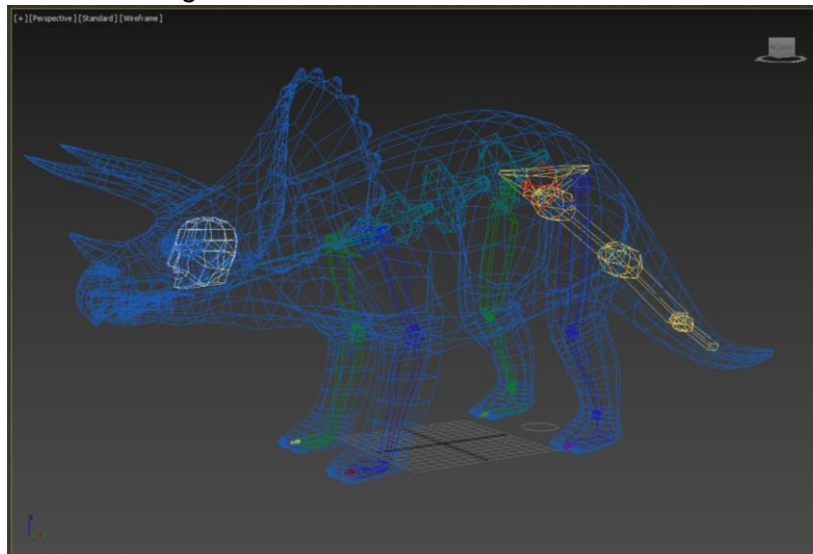


Figura 12 Visualització del triceratops com a malla per veure el bípede intern

Mitjançant aquest programa podem moure el model, movent l'esquelet intern i creant "keys" que són posicions clau per generar l'animació. És a dir, nosaltres generem una keyA on està el peu esquerre darrera i el peu dret davant, i una keyB on estan al revés. Aleshores quan es genera l'animació la figura passa de la posició keyA a la posició keyB. Quantes més keys entre A i B fem, més suau serà l'animació al reproduir-la.

No obstant, la memòria que requereix el programa per moure l'esquelet i el fet que la figura és massa gran, origina errors inesperats al programa, pel que conseqüentment, finalitza de forma inesperada sense possibilitat de recuperar la feina feta.

Donada aquesta situació i fen impossible treballar amb el programa es va explorar treballar amb un altre arxiu disponible, l'arxiu en format .ma, d'Autodesk Maya.

Autodesk Maya



Autodesk Maya és un programa informàtic dedicat al desenvolupament de gràfics 3D per ordinador, efectes especials i animació. El principal motor d'aquest programa i utilitzar-lo

per aquest projecte és la part d'animació. Permet crear efectes d'animació d'alt nivell i aconseguir suavitat en els moviments.

Al utilitzar el programa Maya i obrir el document .ma ens vam adonar que era l'arxiu original amb les animacions pre-definides. És a dir, quan vam executar l'animació des de Maya mateix teníem el triceratops amb tot d'animacions: caminant, rugint, estirant-se, etc. El que vam fer va ser extreure l'animació dels primers 80 *frames* que corresponen a l'animació del triceratops caminant. L'extracció la vam fer en format .fbx, ja que aquest format es compatible amb Unity, on desenvoluparem l'aplicació.

SDK Kinect

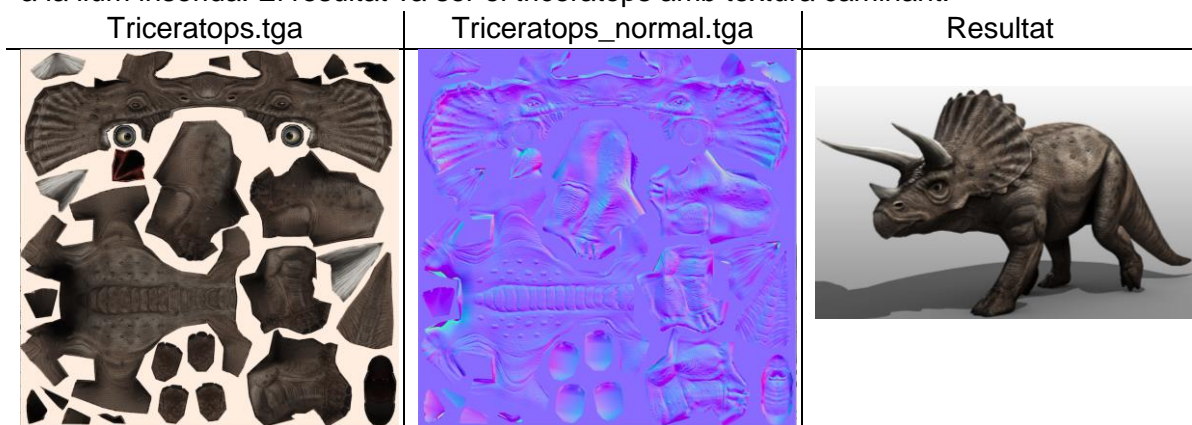
Abans d'anar a Unity per fer l'aplicació per a Kinect, com volem treballar en sistema operatiu Windows, necessitem el SDK de Microsoft per Kinect, ja que els scripts de Unity utilitzen llibreries i funcions definides al SDK per la detecció del cos. En buscar el SDK per descarregar-lo i instal·lar-lo al ordinador vam provar amb l'última versió disponible a la pàgina de Microsoft, la versió 2.0, però no va funcionar. Al llegir les línies de codi dels scripts de Unity vam veure que el *path* definit era per versions 1.X.

Vam provar amb la versió 1.0 amb el mateix resultat. Buscant informació sobre l'error que obteníem vam trobar que altres usuaris ho havien solucionat utilitzant la versió 1.8. Així que vam instal·lar el SDK de Microsoft, versió 1.8 amb el dispositiu desconnectat del ordinador per tal d'evitar altres errors interns.

Textura

Un cop a Unity amb l'animació importada de Maya en format .fbx i el SKD correctament instal·lat i funcionant amb l'asset de Kinect vam generar l'aplicació final.

Primer de tot vam haver de definir la textura a partir dels arxius en format .tga [27]. Primer vam crear un material nou a Unity que era la pell del triceratops amb l'arxiu tga i el normal, ja que Unity dóna l'opció d'afegir la normal de la textura per fer-la més realista a la llum inserida. El resultat va ser el triceratops amb textura caminant.



Taula 4 Com aplicar textura a una figura a partir d'arxius .tga

El segon pas, al ser l'animació massa curta, la vam doblar assignant nous valors de l'eix x per tal que el dinosaure avancés.

Vam partir de l'exemple de l'asset de Kinect anomenat Overlay i vam canviar l'objecte virtual pre-definit per l'animació del nostre dinosaure amb textura. Així que quan ens posem davant del sensor i ens detecta la mà dreta es col·loca el triceratops a la mà caminant.

3.6. Integració amb un detector de punts d'ancoratge

Finalment, agafant el resultat de l'animació obtinguda hem extret diferents posicions del triceratops caminant. Generant varis *frames* en format .ply [28] (únic format permès per registre en núvols de punts). Inserim els *frames* en diferents posicions per generar l'efecte que el triceratops camina, al combinar-los amb el projecte de l'Eduard Pallàs. El seu projecte consisteix en partir d'un sensor RGBD, com a visor de l'escena, per tal de detectar una cantonada amb el seu codi programat gràcies a la llibreria PCL [29]. Detectar una cantonada ens aporta molta informació ja que definim cantonada com la unió de tres plans. Gràcies a aquesta detecció podem trobar la direcció dels eixos XYZ en l'entorn real fent que coincideixin amb els eixos XYZ de la càmera.

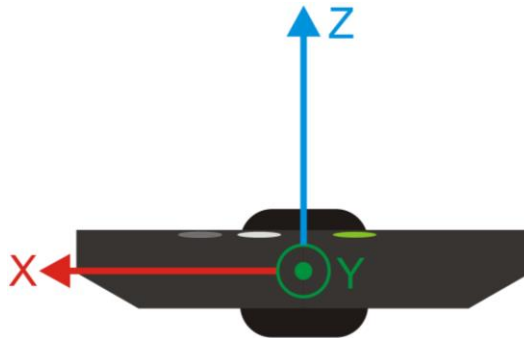


Figura 13 Eixos de coordenades de la càmera d'un Kinect [30]

A més a més, l'objectiu és fer que la cantonada sigui el punt de l'espai (0,0,0), així és molt més senzill inserir una figura 3D ja que aquest punt actuaria com de marcador.

4. Resultats

4.1. Resultats preliminars

- Resultat de l'aprenentatge de Unity

A continuació, a la Taula 5, mostrem diverses imatges obtingudes del vídeo joc final de l'apartat 3.3. Aprenentatge de Unity 3D:



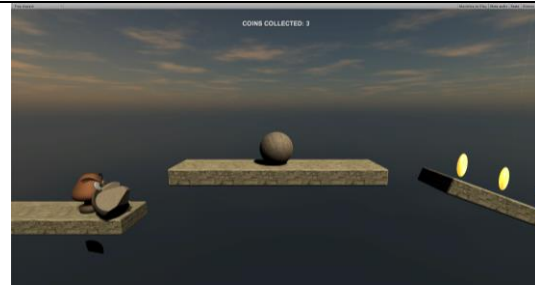
Escena principal, menú per iniciar el joc, sortir del joc o regular el volum de la música. Té el valor afegit que es pot controlar el moviment de la pilota.



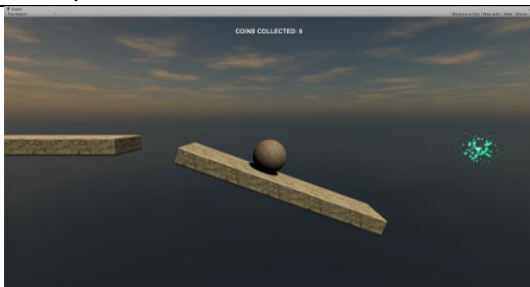
Primera pantalla del joc. El joc consisteix en recollir monedes, cada moneda, augmenta el comptador en un.



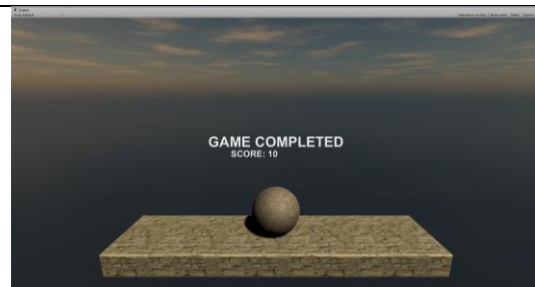
Enemics. Què passa quan ens toca un enemic? La pilota es fragmenta i escoltem una veu de "game over". Tornem a començar des del principi o des de l'últim check point.



Check Point. Quan un enemic ens toca abans de que el matem, tornem a començar des de l'últim check point. Conviu la pilota fragmentada amb la nova per uns instants.



Com canviar d'escena. Quan toquem les espurnes blaves canviem d'escena per passar a la següent.



Final del joc. Un cop ens hem passat totes les pantalles, si hem aconseguit 10 monedes, vol dir que hem fet la puntuació màxima.

Taula 5 Pantalles del resultat final de l'aprenentatge amb Unity

- Resultat d'escanejar amb Structure la sala del ICP
Amb la intenció de tenir una recreació de la sala per no interferir en el treball dels professionals es va escanejar la sala de reconstrucció del ICP.

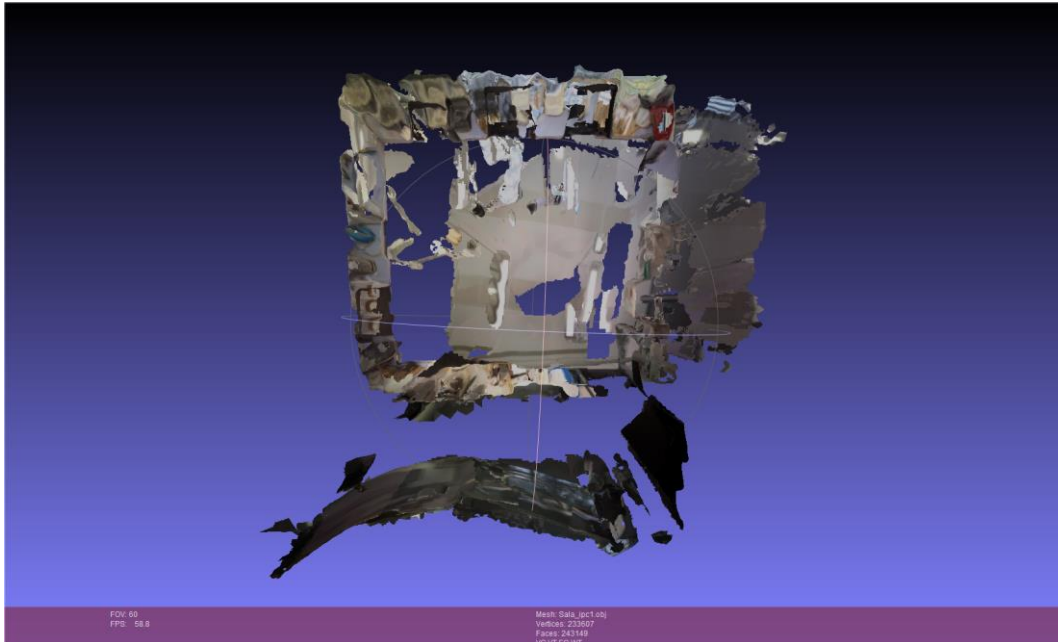


Figura 14 Escanejat sala reconstrucció ICP

Com observem a la Figura 14, és el resultat d'un escaneig total a la sala de reconstruccions del ICP. El resultat és molt bo, tot i que no vam poder evitar tenir punts cecs. D'igual manera, hi ha una part de la sala, que veiem que es troba fora del perímetre. Això es degut a la presència d'un vidre, el vidre al ser reflectant provocava un error al sensor i fa que no detecti correctament la distància a la que es troben els punts de la finestra.



Figura 15 Escanejat taula de treball del ICP

En aquest cas, al fer l'escaneig ens vam centrar en escanejar la taula de treball on els arqueòlegs netegen els fòssils.

- Resultat d'escanejar el model del triceratops al ICP
- Per altra banda, vam escanejar el model disponible en el museu per tenir una figura amb la que començar a treballar mentre no teníem el model de cos sencer.



Figura 16 Escanejat de l'esquelet del triceratops del ICP

En aquest cas, vam escanejar l'esquelet disponible al museu ICP. Es tracta d'un esquelet a mida real d'un triceratops. És una reconstrucció de l'original que es troba als Estats Units, al Museu americà d'història natural. [31]

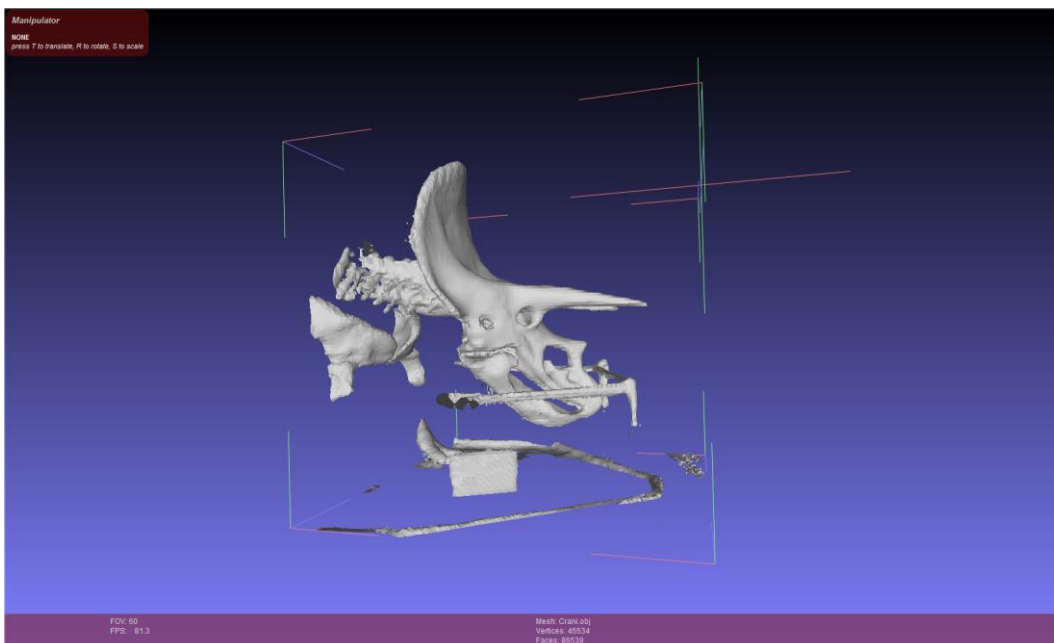


Figura 17 Escanejat del crani del triceratops del ICP

En la Figura 17, ens vam centrar únicament en el crani per tenir-lo amb el màxim detall possible.

- Resultat del model 3D del ICP

A la Taula 6 i la Figura 18, observem els resultats obtinguts del procés explicat al apartat 3.5.1. Registre i Integració: Unity + ARToolkit:



Partint de la posició inicial vam arreglar el model, allisant la superfície i omplint forats. També vam desencaixar la mandíbula esborrant els punts corresponents a la unió.



I vam crear l'animació del moviment de la part corresponent únicament a la mandíbula, recreant com si l'esquelet parlés.

Taula 6 Resultat de Unity amb ARToolkit amb l'escanejat del ICP



Figura 18 Resultat de Unity amb ARToolkit amb el model cedit pel ICP

Gràcies al ICP que ens va cedir un model a mida real però de carn i óssos. Un cop vam aconseguir extreure l'animació del triceratops caminant, abans de treballar amb sensors RGBD vam provar de registrar-lo en l'aplicació ja funcionant de l'esquelet.

- Resultat intermedi. Com Kinect detecta l'esquelet?

A la Figura 19, tenim el resultat pràctic del que explicàvem de la detecció d'esquelet.

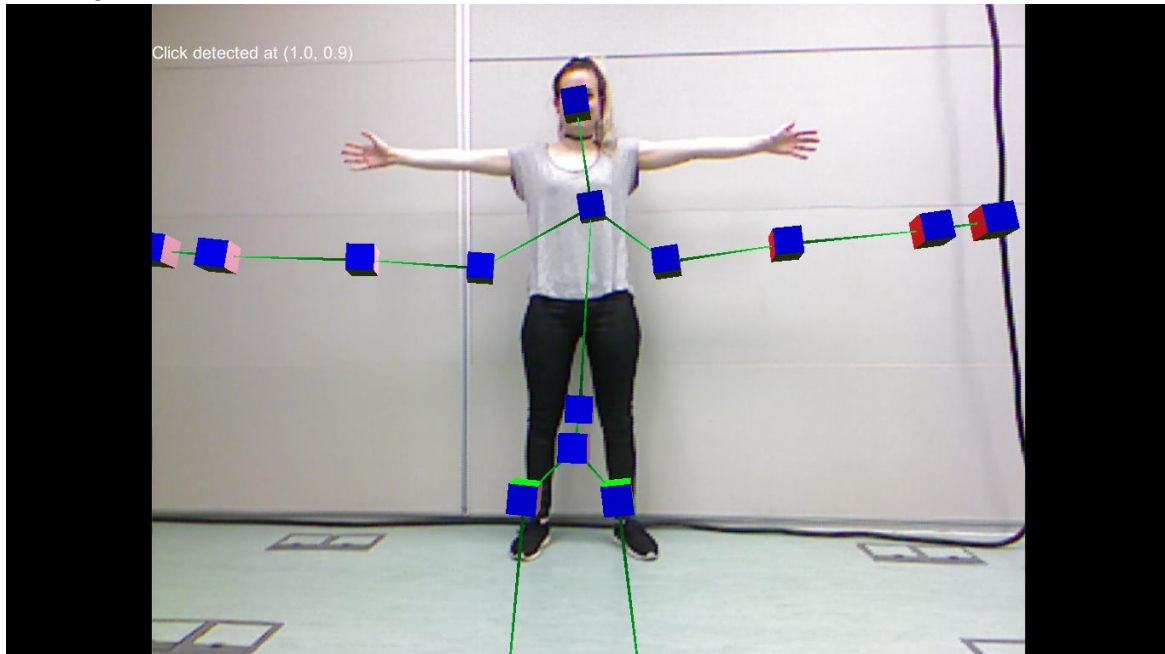


Figura 19 Resultat intermedi. Detecció de l'esquelet

Per comprendre millor els resultats següents calia una explicació prèvia per ensenyar com Kinect ens detecta una vegada ens col·loquem davant. Kinect ens detecta com un esquelet compost de cubs i unions, els cubs representen articulacions del cos: espatlles, colzes, canells, genolls,... I extremitats finals com les mans, els peus i el cap.

Detecta tot el cos, tot i que a la imatge, es desquadrés la representació i no es vegi de forma completa.

4.2. Resultats del prototip d'aplicació

A continuació, es mostren unes captures de pantalla de diferents prototips d'aplicacions obtingudes gràcies a tot el procés descrit a l'apartat 3.5.2. Registre i Integració: Unity + Kinect:

- Resultat primera aplicació:

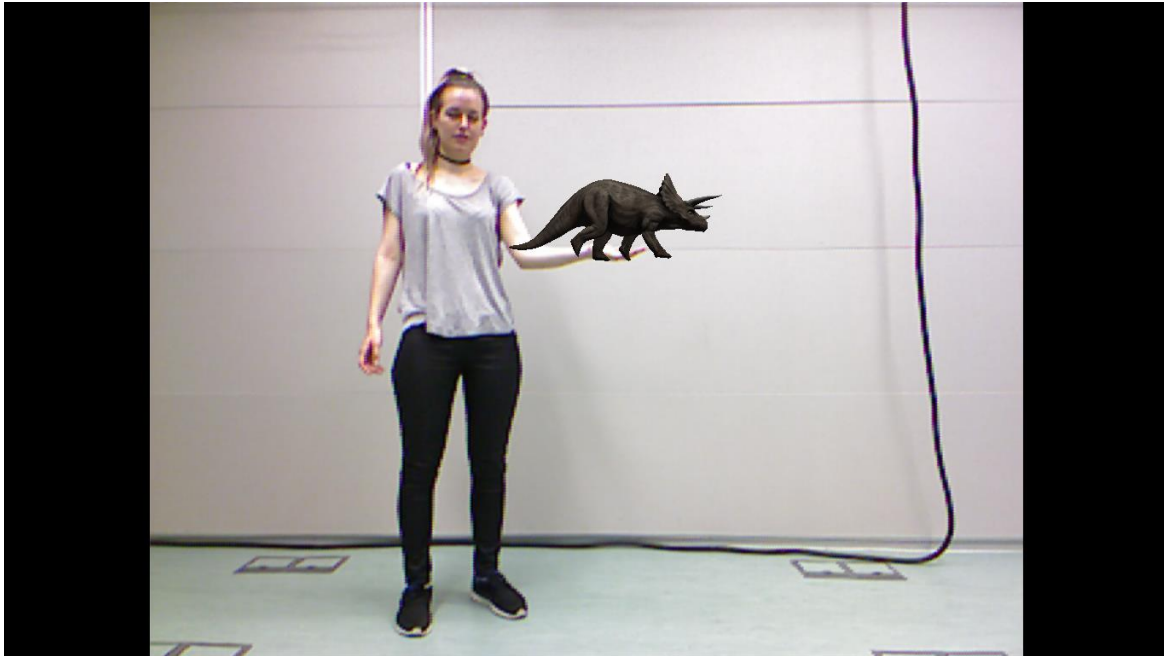
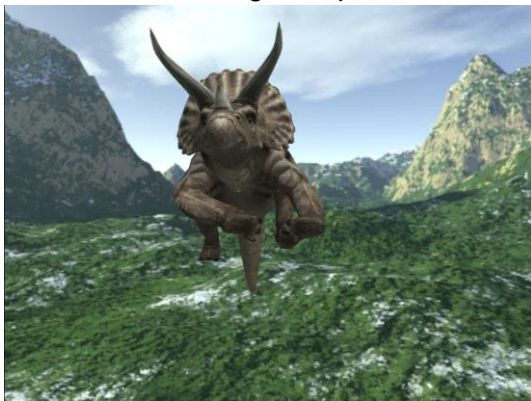


Figura 20 Resultat primera aplicació. Triceratops ens camina per la mà

Mitjançant la detecció de l'esquelet creem una aplicació que detecti la mà dreta i un cop sigui detectada col·loqui el nostre dinosaure caminant. L'efecte on el triceratops camina per la mà va subjecte a la precisió del Kinect per detectar la mà dreta.

- Resultat segona aplicació:



Per pantalla observem el nostre triceratops que es mou per la jungla.



Mentre nosaltres fem els moviments davant d'un Kinect.



Hi ha vegades que per pantalla els moviments no són del tot fidels (degut a les restriccions de moviment que té el triceratops en traslladar els moviments d'un esquelet bípede a un cos quadrúpede).



Nosaltres ens col·loquem amb un braç aixecat, no obstant, el triceratops no pot aixecar-lo més d'un punt.

Taula 7 Resultat segona animació. Som un triceratops a la jungla

- Resultat aplicació final:

Com hem comentat a l'apartat 3.6. Integració amb un sensor detector de punts d'ancoratge, mostrarem tot seguit el resultat de la combinació del nostre projecte amb el projecte de l'Eduard Pallàs.

El seu projecte ens aporta nous avantatges:

- És independent de que el sensor vegi o no el marcador (cantonada) per inserir l'element virtual.
- És robust davant canvis en la il·luminació ja que no té en compte les components RGB, només les coordenades en l'espai
- L'element virtual, no és un *overlay*, sinó que s'integra amb l'escena com es veu en la següent Figura 21.

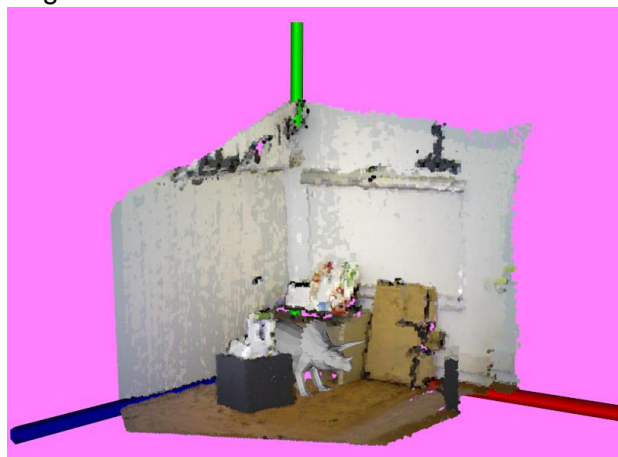


Figura 21 Resultat aplicació final. Triceratops a la sala

Partint dels nous avantatges que ha obtingut, hem pogut inserir el triceratops en un núvol de punts, procedent d'una captura d'un sensor RGBD. Mitjançant les diferents posicions del triceratops caminant generem l'efecte que el triceratops camina per la sala.

5. Pressupost

En aquest projecte hem necessitat els següents components:

- 1 x iPad Pro = 1070€
- 1 x Structure = 285€
- 1 x Bracket = 164,5€ + 32,5€ (enviament) = 197€
- 1 x Llicència de Unity Pro = 57€/mes x 5 mesos = 285€
- 1 x Llicència Autodesk 3DS MAX = 242€/mes x 2 mesos = 484€
- 1 x Llicència Autodesk Maya = 242€/mes x 3 mesos = 726€

SUBTOTAL: 3.047€

I hem invertit les següents hores en desenvolupar el projecte total:

- Temps invertit en supervisió:
 - o En hores: 17h
 - o Hores tutor: 17h
 - o Subtotal: 17h x 8€/h + 17h x 30€/h = 646€
 - o Total: 646€ + 21% (overhead UPC *) = 782€
- Temps d'aprenentatge amb Unity:
 - o En hores: 545h
 - o Fer-los: 545h
 - o Total: 1090h x 8€/h = 8.720€
- Temps de desenvolupament 3DS MAX:
 - o En hores: 50h
 - o Total: 50h x 8€/h = 400€
- Temps de desenvolupament Maya:
 - o En hores: 100h
 - o Total: 100h x 8€/h = 800€
- Temps de desenvolupament Unity per l'aplicació:
 - o En hores: 100h
 - o Total 100h x 8€/h = 800€

SUBTOTAL: 11.502€

TOTAL: 14.549€

*Overhead UPC: es tracta d'un percentatge d'un projecte desenvolupat a la UPC que, si genera ingressos per facturació, serveix per cobrir despeses indirectes de infraestructura i espais de la UPC com ho són ocupació de sales de treball i despatxos, despeses de subministrament i serveix com llum, aigua, neteja, serveis de seguretat i consergeria etc.

6. Conclusions i línies futures:

Un cop hem vist els resultats és hora d'extreure conclusions del projecte.

Un cop acabat el nostre projecte i en haver explorat diferents eines de treball per capturar escenes a partir del Structure Sensor o del Kinect podem concloure que som capaços de generar diferents prototips d'aplicacions per generar animacions de realitat augmentada, com hem vist en els resultats obtinguts.

Tal i com s'ha explicat a la metodologia, a partir del model base cedit pel museu, hem aconseguit dissenyar una figura 3D compatible amb les nostres aplicacions per tal que s'integrés amb l'entorn.

Tot i que finalment la captura de la sala no va resultar útil pel projecte. Vam aprendre a extreure un model de sala gràcies al Structure Sensor, igual que el model de la taula de reconstrucció com hem mostrar en els resultats. D'haver tingut accés més continuat a aquesta sala del museu, podríem haver generat l'augmentació en aquesta sala de la mateixa manera que s'ha fet per una sala de la UPC.

Podem concloure que hem registrat amb èxit els elements virtuals en l'espai físic i que hem aconseguit dues aplicacions de realitat augmentada que funcionen a temps real i una aplicació final, on hem aconseguit registrar la figura virtual en una sala amb rotació, escala i orientació. Tot i que hem assolit l'objectiu d'aplicació aquesta no és a temps real ja que amb el temps del que es disposava no era un objectiu factible.

Les línies futures d'aquest projecte serien varies.

La primera de totes, seria desenvolupar l'aplicació final per tal que funcionés en temps real i veure a través de la pantalla i d'un sensor RGBD com el dinosaure es passeja per la sala. Explorant més opcions de la llibreria PCL o aprofitant llibreries de robòtica com MRPT [32] (Mobile Robotic Programming Toolkit) són unes llibreries en C++ on podem trobar llibreries molt interessants per Kinect i núvols de punts.

Un altre possible continuació d'aquest projecte, seria programar en iOS a partir de l'aplicació Xcode per aconseguir que l'aplicació funcionés amb el sensor RGBD Structure Sensor, ja que pel moment només és executable per Kinect o Asus. [33]

Una millora del projecte actual seria dotar de més animació el dinosaure, actualment, només vam poder extreure els frames del dinosaure caminant però hi ha tot d'altres animacions pre-definides que podrien configurar una animació més complexa i interessant.

L'última línia futura que podem presentar, és una combinació de les tres anteriors. És a dir, aconseguir l'aplicació en temps real, compatible amb Structure Sensor per tal d'estar més a prop de l'aplicació final pel museu. Ja que si fos compatible amb el Structure Sensor, podríem cedir tauletes amb el sensor als visitants del museu i que a través de les seves tauletes veiessin reconstruïts els fòssils. Així, veurien per la sala els dinosaures caminant o estirats o rugint,... alhora que també veurien a les pantalles informació important dels dinosaures per explotar la seva vessant educativa. A partir d'aquí, caldria que els responsables de comunicació del museu proposessin un fil conductor per a la visita amb realitat augmentada amb guió, paraula i comunicació amb el visitant, de manera que el dinosaure no fos només un element decoratiu sinó que es comportés com un guia que informés del que estem veient a les diferents sales d'exposició.

Bibliografia:

- [1] Wikipedia, «Realidad Aumentada,» 8 Juny 2016. [En línia]. Available: https://es.wikipedia.org/wiki/Realidad_umentada.
- [2] HolaApp, «Realidad Aumentada – Augmented Reality,» WordPress, 20 Novembre 2013. [En línia]. Available: <http://www.realidad-aumentada.eu/es/tag/tecnicas-realidad-aumentada/>.
- [3] OhMyPhone!, «La Realidad Aumentada Móvil se triplicará en 2018,» Orange, 07 Novembre 2013. [En línia]. Available: ohmyphone.orange.es/mas-smartphones/curiosidades/la-realidad-aumentada-movil-se-triplicara-en-2018.html.
- [4] M. Crusafont, «Institut Català de Paleontologia,» Generalitat de Catalunya, Universitat Autònoma de Barcelona, 2015. [En línia]. Available: <http://www.icp.cat/index.php/ca/>.
- [5] Dinosaurios, «Triceratops,» 16 Setembre 2014. [En línia]. Available: <http://dinosaurios.org/triceratops/>.
- [6] M. Heilig, «Sensorama,» Wikipedia, 27 Gener 2015. [En línia]. Available: <https://es.wikipedia.org/wiki/Sensorama>.
- [7] S. Feiner, B. MacIntyre i D. Seligmann, «KARMA (Knowledge-based Augmented Reality for Maintenance Assistance),» Columbia University, [En línia]. Available: <http://monet.cs.columbia.edu/projects/karma/karma.html>. [Últim accés: 2016].
- [8] I. Healthcare Innovation and Technology Lab, «HitLab,» Healthcare Innovation and Technology Lab, Inc., 2016. [En línia]. Available: <http://www.hitlab.org/#about-1>.
- [9] «Puteko Limited,» 2012. [En línia]. Available: <http://colarmix.com/>.
- [10] H. L. NZ, «Human Interface Technology Laboratory, New Zealand,» University of Canterbury, [En línia]. Available: <http://www.hitlabnz.org>.
- [11] QuiverVision, «Quiver - 3D Coloring App,» PUTEKO Limited, [En línia]. Available: <http://quivervision.com>.
- [12] P. Roldán, «Plástica con Quiver,» CEIP, 10 Febrer 2016. [En línia]. Available: <http://proferoldi.blogspot.com.es/2016/02/plastica-con-quiver.html>.
- [13] Google, «Google Glass,» Google Developers, 15 Abril 2016. [En línia]. Available: <https://developers.google.com/glass/>.
- [14] «Develop With Depth,» Occipital, 2016. [En línia]. Available: <http://structure.io/developers#bridge>.
- [15] Occipital, «Bridge Engine Demo: Occipital at CES 2016,» Youtube, 14 Gener 2016. [En línia]. Available: <https://www.youtube.com/watch?v=cEnnbCSbijo>.
- [16] Google, «Tango,» Google, 05 Juny 2014. [En línia]. Available: <https://get.google.com/tango/>.
- [17] R. Abovitz, «Magic Leap,» Magic Leap Inc., 2010. [En línia]. Available: <https://www.magicleap.com/#/home>.
- [18] A. Thirlund, «Brackeys,» 2016. [En línia]. Available: <http://brackeys.com/about/>.
- [19] A. Thirlund, «Make A Game,» Brackeys, 6 Agost 2013. [En línia]. Available: <https://www.youtube.com/watch?v=g5QFW12utdU&list=PLPV2KyIb3jR4u6zeBY77WPj0KuFdmv84g>.
- [20] S. M. Bros, «Goomba,» Nintendo, 16 Novembre 2007. [En línia]. Available:

- <http://es.mario.wikia.com/wiki/Goomba>.
- [21] M. Foundation, «WebGL,» Wikipedia, 15 Març 2016. [En línia]. Available: <https://es.wikipedia.org/wiki/WebGL>.
- [22] i. S.a.r.l., «Xcode,» Apple, 03 Maig 2016. [En línia]. Available: <https://itunes.apple.com/es/app/xcode/id497799835?mt=12>.
- [23] «Fastest-selling gaming peripheral,» Guinness World Records, 03 Gener 2011. [En línia]. Available: <http://www.guinnessworldrecords.com/world-records/fastest-selling-gaming-peripheral/>.
- [24] ARToolkit, «Getting Started with ARToolKit for Unity,» DAQRI, 2016. [En línia]. Available: http://artoolkit.org/documentation/doku.php?id=6_Unity:unity_getting_started#AdvancedTopics.
- [25] «Hiro Pattern,» [En línia]. Available: <http://www.neo-tech-lab.co.uk/arsensor/Article.files/HiroPattern.png>.
- [26] ARToolkit, «ARToolKit for Unity Scripting and Low-Level API,» DAQRI, 2016. [En línia]. Available: http://artoolkit.org/documentation/doku.php?id=6_Unity:unity_low_level_api.
- [27] «Truevision TGA,» Wikipedia, 09 Març 2015. [En línia]. Available: https://en.wikipedia.org/wiki/Truevision_TGA.
- [28] «PLY (file format),» Wikipedia, 29 Abril 2016. [En línia]. Available: [https://en.wikipedia.org/wiki/PLY_\(file_format\)](https://en.wikipedia.org/wiki/PLY_(file_format)).
- [29] W. Garage, «Point Cloud Library,» Creative Commons Attribution 3.0, Març 2010. [En línia]. Available: <http://pointclouds.org/>.
- [30] «Capture for Kinect,» Depth Biomechanics, 15 Gener 2013. [En línia]. Available: <http://www.depthbiomechanics.co.uk/?p=2496>.
- [31] «Triceratops,» American Museum of Natural History, [En línia]. Available: <http://www.amnh.org/exhibitions/permanent-exhibitions/fossil-halls/hall-of-ornithischian-dinosaurs/triceratops>.
- [32] «MRPT,» GitHub, 2016. [En línia]. Available: <http://www.mrpt.org/>.
- [33] «Xtion PRO LIVE,» ASUS, 12 Setembre 2011. [En línia]. Available: http://www.asus.com/3D-Sensor/Xtion_PRO_LIVE/.