

Captura de Moviment

Recerca i desenvolupament d'un sistema de captura de moviment amb Kinect i les seves aplicacions en l'animació tridimensional



Índex

1. Introducció	3
2. Introducció al 3D	4
2.1. Què és?.....	4
2.2. Possibles confusions.....	4
3. Captura de moviment.....	5
3.1. Què és i com funciona?	5
3.2. Quines utilitats té?	6
4. Vicon	10
4.1. Què és i com funciona?	10
4.2. Avantatges i inconvenients	12
5. Kinect.....	13
5.1. Què és i funció original	13
5.2. Funcionament	13
5.3. Funcions alternatives	14
5.4. Avantatges i inconvenients respecte als sistemes Vicon	16
6. Part pràctica	17
6.1. Rodatge.....	17
6.2. Codificació de vídeo	21
6.3. Edició inicial.....	24
6.4. Captura de moviment.....	25
6.5. Modelatge i texturització	30
6.6. <i>Motion tracking</i>	35
6.7. Composició 3D	37
6.8. Postproducció	43
6.9. Edició final.....	47
7. Conclusions	52
8. Agraïments	53
9. Programari utilitzat.....	54
10. Bibliografia.....	55

1. Introducció

Aquest treball de recerca està dedicat al món de la captura de moviment, ja que des de fa anys em crida l'atenció el món del cinema, en especial tota la part tècnica que hi ha darrere dels efectes visuals i especials.

Vaig tenir bastant clar des del principi que volia parlar d'alguna cosa relacionada amb aquest camp. Però el que no tenia clar era cap a on enfocar el treball o què explicar d'aquest camp. Al final em vaig decantar per un tema que m'encuriosia molt arran del seu ús massiu en les pel·lícules més actuals: la captura de moviment.

El primer objectiu al qual vull arribar és explicar què és la captura de moviment, com funciona i quines possibles aplicacions té.

El segon objectiu és la realització d'una captura de moviment 3D per animar un personatge generat per ordinador a través del dispositiu Kinect.

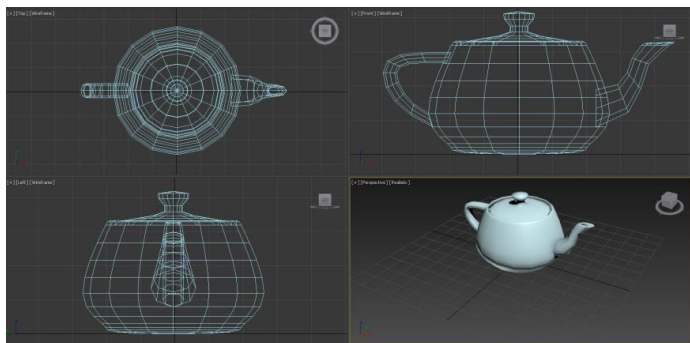
El tercer objectiu consisteix a desenvolupar una metodologia simplificada per aconseguir els mateixos resultats o similars als que s'obtenen als platós de captura de moviment, però amb el mínim pressupost i el màxim d'automatització i simplificació possible.

Finalment, es fa un curtmetratge d'un minut aproximadament posant en pràctica aquests coneixements.

2. Introducció al 3D

2.1. Què és?

Són gràfics que utilitzen una representació tridimensional de dades geomètriques que són emmagatzemades amb el propòsit de realitzar càlculs i exportar imatges 2D o imprimir-les en 3D.

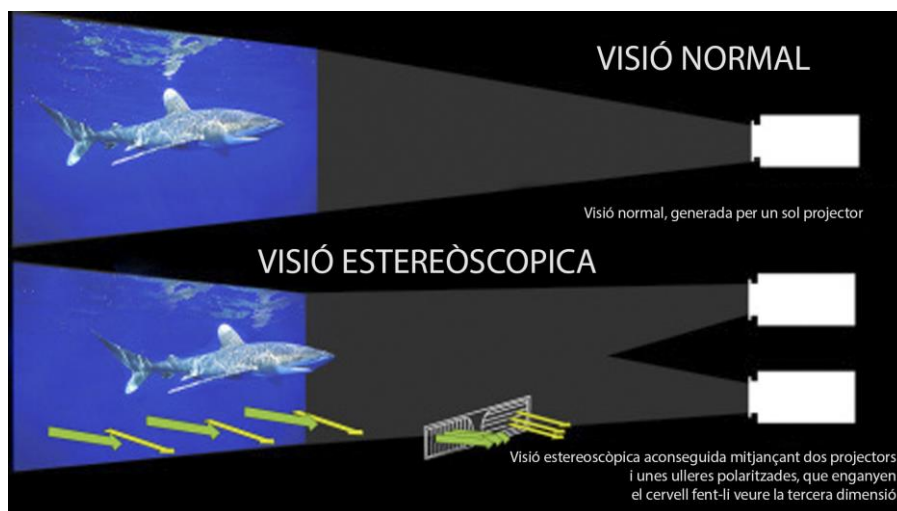


Per crear-les fa falta programari especialitzat en generar i modelar dades tridimensionals, tot i que actualment hi ha alguns editors 2D que ofereixen la possibilitat d'aplicar tècniques 3D o de "pseudo-3D" i donen un resultat bastant similar. També és recomanable una targeta de processament gràfic per poder calcular models tridimensionals més complexos (amb milions de polígons).

2.2. Possibles confusions

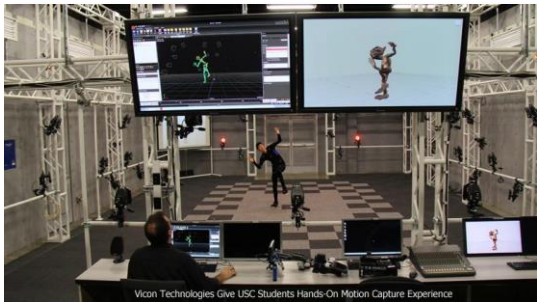


Cal diferenciar entre el 3D estereoscòpic i els gràfics 3D generats per ordinador. El 3D estereoscòpic és un efecte creat amb dues càmeres separades, amb una certa distància i un cert angle de divergència entre elles, les imatges se solapen i posteriorment es projecten amb dos projectors també separats. L'espectador ha de mirar la projecció a través d'unes ulleres polaritzades que enganyen la visió i creen una il·lusió de profunditat, però realment no existeix cap tercera dimensió. En canvi, els gràfics 3D generats per ordinador, tot i que després seran visualitzats en dues dimensions en la majoria de casos, sí que tenen una tercera dimensió. En aquest treball ens centrarem únicament en la captura de moviment que s'utilitza en els gràfics tridimensionals.



3. Captura de moviment

3.1. Què és i com funciona?



©Vicon Technologies

La captura de moviment (Motion Capture o Mocap) és el procés d'enregistrament de moviment d'objectes, persones o animals.

En cinema i videojocs, es refereix a la gravació de les accions dels actors (humans, animals, accessoris...) i utilitzar aquesta informació per animar models de personatges en dues o tres dimensions.

Durant les sessions de captura de moviment, es capturen només els moviments de l'actor, deixant de banda l'aparença visual de l'actor.

El seu funcionament és complicat. A un actor se li col·loquen uns marcadors a prop de cada articulació per identificar les posicions o angles entre ells.



Exemple d'un marcador

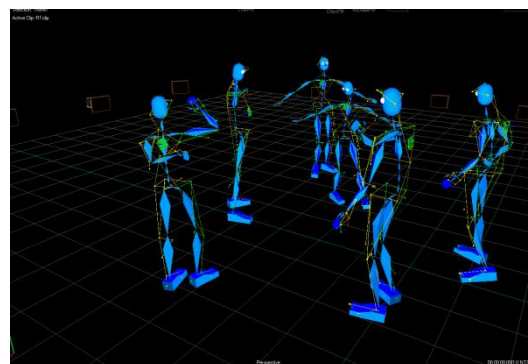
Aquests marcadors envien un senyal que permet localitzar-ne la posició exacta.



Exemple d'un sensor

Els marcadors poden ser: acústics, inercials, LED, magnètics o reflectants, o combinacions de qualsevol d'aquests.

Els senyals que els marcadors emeten són captats pels sensors, uns 50 aproximadament, que estan repartits per tota la sala i a través del programari són capaços de seguir el moviment de cada marcador i interpretar la seva posició 3D per tal de formar un esquelet entre els marcadors.



Esquelets formats mitjançant la informació rebuda pels sensors

3.2. Quines utilitats té?

Aquesta captura de moviment pot tenir diverses aplicacions, per exemple mèdiques, militars, en videojocs, cinema i alguns altres usos minoritaris.

En medicina se sol utilitzar amb persones que tenen alguna discapacitat motora per veure com i per què aquests moviments són diferents del normal. També s'utilitza en pacients postictus per monitoritzar la seva recuperació o per estudiar els moviments d'una persona per elegir i ajustar una pròtesi amb més facilitat.



En l'exèrcit té una utilitat més orientada a l'entrenament mitjançant simulacions per ordinador. Aquest tipus de simulacions són útils per als soldats perquè són capaços de veure's dintre d'una situació real de guerra sense córrer cap tipus de perill. També s'utilitza per a la investigació de robòtica no tripulada.

Tot i que tenen una utilització cada vegada major, els esmentats anteriorment no són els usos més freqüents de la captura de moviment. Els més freqüents són els videojocs i el cinema.

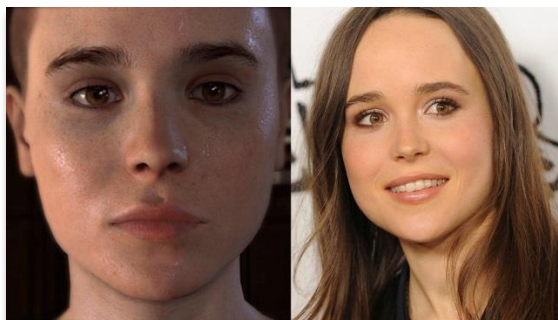
En videojocs s'utilitza per crear moviments més orgànics als personatges amb l'objectiu de crear una experiència més realista al jugador. Pràcticament s'utilitza en tots els videojocs. No només per la part de cinemàtiques (escenes no jugables del joc on els personatges fan les accions sols), sinó també per capturar un a un els infinits moviments que fa el personatge durant el joc.

Però algunes empreses han anat més enllà de la simple captura dels moviments i també han capturat les expressions i les emocions dels actors. Un dels videojocs més recents que ha revolucionat la indústria de la captura de moviment és un videojoc creat per Quantic Dream exclusivament per a PlayStation 3 anomenat Beyond: Two Souls. Per a aquest videojoc van ser contractats actors famosos per a la interpretació

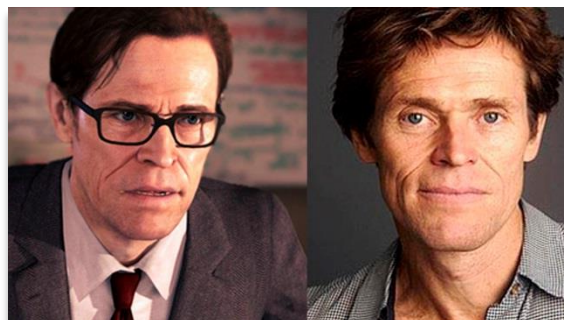
Captura de Moviment

dels personatges. Els personatges no només compartien els mateixos moviments sinó que també tenien la mateixa veu, les mateixes expressions facials i el mateix físic que els actors. Al tenir exactament la mateixa constitució i trets que els actors, es pot fer una captura de moviment facial i corporal molt més precisa i expressiva.

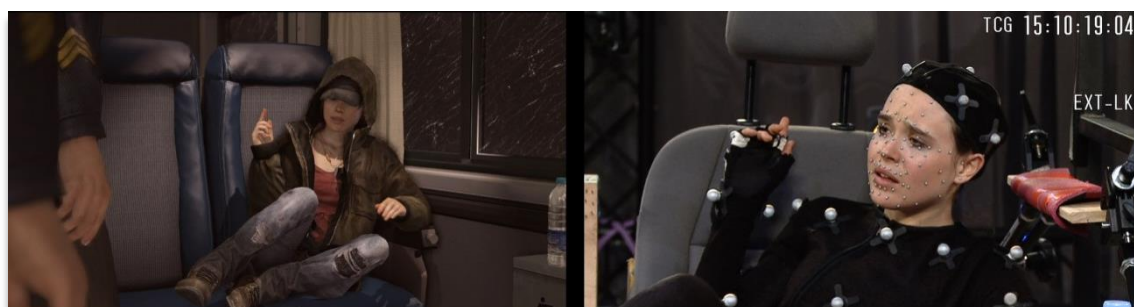
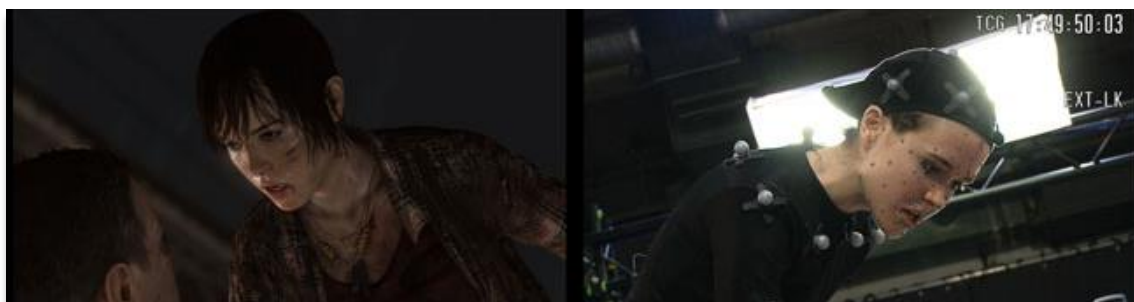
A continuació hi ha una mostra de com es veuen els actors dintre el joc i com són a la realitat. També hi ha diverses mostres del rodatge de la captura de moviment.



Ellen Page



Willem Defoe



Captura de Moviment

L'últim ús seria el cinema. En el cinema diferenciem clarament dos usos, el primer seria les pel·lícules d'animació on tot està fet per ordinador i el segon serien els efectes visuals per a les pel·lícules. En les pel·lícules d'animació no és tan comú perquè moltes vegades els personatges als quals es vol animar no tenen forma humana o es prefereixen uns moviments més exagerats i còmics abans que un moviment realista. Tot i això, hi ha pel·lícules, com per exemple *Les aventures de Tintin*, on s'utilitzen actors per interpretar els personatges mitjançant la captura de moviment.

És bastant similar a la captura de moviment per videojocs, es crea un plató on els personatges actuaran i se'ls capturarà el moviment, les expressions i la veu. Posteriorment, hi ha tot un equip de tècnics encarregats de traspasar aquest moviment als personatges de la pel·lícula.

A continuació teniu algunes mostres del rodatge de *Les aventures de Tintin*:



© Paramount Pictures



Captura de Moviment

El segon ús de la captura de moviment en el món del cinema és per als efectes visuals. Els efectes visuals són els processos on es barregen imatges que estan creades o manipulades a través de l'ordinador amb imatges gravades directament amb la càmera. Els efectes visuals s'utilitzen quan es volen crear ambients que semblin realistes, però realitzar-los seria molt costós, perillós o simplement impossible de capturar-los amb una càmera.

En aquest cas la captura de moviment s'utilitza per animar personatges creats per ordinador que després s'integraran en imatges gravades directament per la càmera.

A continuació hi ha una mostra de la pel·lícula *Ted* on es veu el procés de captura de moviment i la integració dins d'una imatge gravada directament des de la càmera.



© Universal Pictures

Aquí es mostra tot el procés utilitzat, des de la captura de moviment fins al *render* final. Aquest procés de *render* l'explicarem més endavant.

4. Vicon

4.1. Què és i com funciona?

És una empresa que fabrica un dels sistemes de captura de moviment més precisos i fiables de l'actualitat. Vicon ha revolucionat la indústria de la captura de moviment durant més de 25 anys, empenyent constantment els límits del possible. Fabrica dispositius de captura de moviment per a tot tipus d'aplicacions, des d'aplicacions mèdiques fins a aplicacions més encarades a l'entreteniment.

L'actor que estigui interpretant el personatge ha de dur un vestit especial, normalment de tela sintètica i elàstica, on s'enganxen uns petits marcadors que tenen forma de creu amb una petita esfera que sobresurt al mig. Es posen també marcadors als objectes d'*atrezzo* que porta el personatge i als objectes de l'entorn que interactuïn amb l'actor (portes, cadires, taules, etc.). Aquests marcadors acostumen a emetre una llum infraroja que és captada per un conjunt de sensors que estan repartits per tota la sala. Moltes vegades es munta com una espècie de càmera virtual que consta d'una pantalla, uns petits marcadors al voltant de la pantalla, dos mànecs per on poder agafar la pantalla i, a vegades, un adaptador per a l'espatlla. Des de la pantalla es veu l'animació que s'està duent a terme a temps real i a través d'aquesta



Un exemple de càmera virtual. En aquest cas, també s'han afegit uns comandaments de la consola Wii com a marcadors infraroig.

càmera l'operari pot elegir amb més llibertat quin angle o quins moviments vol filmar. Tot i que

és freqüent l'ús d'aquestes càmeres virtuals i que donen uns resultats més orgànics, moltes vegades les càmeres s'animen a través de l'editor 3D. Aquest conjunt de sensors estan connectats a un ordinador central o a diversos ordinadors connectats en xarxa des d'on es processa tota la informació i s'aplica el moviment dels actors capturat als personatges creats per ordinador prèviament. Encara que s'utilitzen els personatges creats per ordinador a més baixa qualitat del resultat final i encara no s'han aplicat les textures i els ombrejats finals, es necessiten ordinadors de gran potència de processador i processador gràfic per poder carregar-ho tot a temps real.

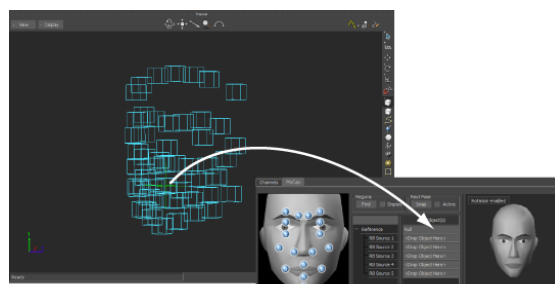
Captura de Moviment



Vicon Active Wand

Un cop està tot instal·lat i connectat comença el procés de calibratge abans de poder gravar les animacions. Per calibrar les càmeres s'utilitza una espècie de T, anomenada Active Wand, on hi han LEDs. Aquest dispositiu es mou pel mig de l'escena seguint una sèrie de moviments marcats pel programari de l'ordinador i l'ordinador va calculant la posició de les càmeres.

Després de calibrar totes les càmeres, es calibren els marcadors de l'actor amb el del personatge creat per ordinador. Normalment els vestits que porten ja estan calibrats amb els personatges, però en el cas de no estar-ho s'ha d'associar la posició del marcador respecte el cos de l'actor amb la posició d'un objecte anomenat *null* respecte al personatge generat per ordinador. Un objecte *null* és la representació virtual del marcador dins de l'editor 3D.



Aquí es mostra el procés de posicionament dels objectes *nulls* detectats amb el model

Un cop calibrats tots els dispositius es pot començar a gravar. Durant el procés de gravació el que succeeix és que l'actor, al moure's, mou el marcador que duu al cos, el qual és detectat pels sensors que passen la informació a l'ordinador. Un cop arribada tota la informació a l'ordinador, es transmet el moviment tridimensional del marcador a l'objecte *null*. Quan es mou l'objecte *null* arrossega la part del personatge que li pertoca per al marcador. Cada programa utilitza una representació diferent dels *nulls*, un nom diferent o, depèn del programa, utilitza en esquelet en comptes dels *nulls*, però el funcionament acostuma a ser similar.

4.2. Avantatges i inconvenients

Avantatges:

- Es pot capturar a una freqüència de 1.000 fps (fotogrames per segon)
- El marge d'error de la captura es mínim, té un marge d'error de 0,5 mm o inferior, depenent del model
- Capaç de realitzar captura facial i corporal al mateix temps
- Capaç de fer la captura a temps real

Inconvenients:

- Tenen un cost elevat, aproximadament entre uns 20.000 € i 50.000 € tot l'equip
- Necessitat d'un ordinador potent
- Necessitat de marcadors per captar el moviment
- Necessitat d'unes instal·lacions capaces de suportar tot l'equip
- Requereix una gran quantitat de temps per calibrar tots els aparells

5. Kinect

5.1. Què és i funció original

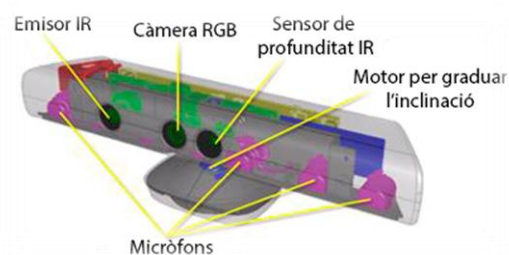
Kinect és un comandament de joc i entreteniment creat per Alex Kipman, desenvolupat per Microsoft per a la videoconsola Xbox 360, tot i que al juny del 2011 Microsoft va llençar els controladors per poder utilitzar-la a l'ordinador (només amb sistema operatiu Windows). Kinect



esta dissenyada perquè els usuaris puguin controlar i interactuar amb la consola sense la necessitat de mantenir contacte físic amb el comandament tradicional de les consoles. Es capaç de reconèixer gestos, controls de veu, objectes i imatges. L'objectiu era augmentar l'ús de la consola Xbox 360.

5.2. Funcionament

El sensor Kinect és una barra horitzontal connectada a una petita base amb un pivot motoritzat i està dissenyat per ser col·locat per sobre o per sota de la pantalla de televisió. El dispositiu compta amb una càmera RGB, un sensor de profunditat compost per un emissor i un receptor infraroig i per micròfon de múltiples



Parts de la Kinect

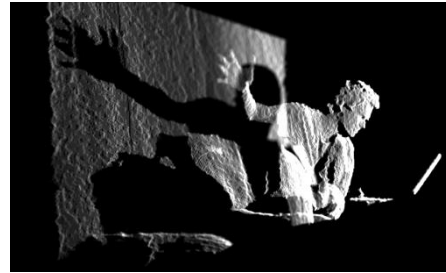
matrius que proporcionen la captura de moviment de tot el cos, reconeixement facial i reconeixement de veu .



Punts de llum infraroja projectats

El sensor de profunditat es compon d'un emissor infraroig combinat amb un sensor monocrom CMOS, que captura les dades de vídeo en 3D sota qualsevol llum ambiental. En un extrem hi ha un emissor de llum infraroja que projecta com un seguit de punts que topen amb tots els objectes dins de l'angle de visió de la Kinect. Aquests punts són captats per un sensor de profunditat infraroja que calcula la distància que hi ha entre el sensor i cada punt projectat. Dins de l'ordinador o la Xbox 360 la imatge és representada com un núvol de punts tridimensional. El programari del dispositiu és capaç d'identificar què es una persona dins el núvol de punts i capturar quin moviment fa. L'interval de detecció del sensor de profunditat és ajustable, i el programari Kinect

és capaç de calibrar automàticament el sensor basat en el joc i l'entorn físic del jugador, per acomodar la presència de mobles o altres obstacles. El flux de vídeo RGB predeterminat utilitza una resolució VGA de 8 bits (640 × 480 píxels). El sensor Kinect és capaç de capturar el moviment a 1,2 a 3,5 m de distància del jugador quan s'utilitza amb el programari de Xbox. L'àrea requerida per jugar amb la Kinect és d'aproximadament 6 m², tot i que el sensor pot mantenir el seguiment a través d'un interval d'aproximadament 0,7-6 m de distància amb el jugador.



Representació de l'espai mitjançant el núvol de punts utilitzat pel dispositiu

5.3. Funcions alternatives

Des del dia del seu llançament, la Kinect va intentar ser "hackejada" per tot tipus de usuaris amb la finalitat de millorar-la, descobrir el seu funcionament o per buscar funcions alternatives. Van començar a utilitzar-la per a jocs d'ordinador o altres usos minoritaris. No hi havien uns controladors estables i cadascú feia la seva pròpia versió de controladors depenent de la finalitat que volguessin (com a *webcam*, com a micròfon, com a comandament per a videojocs, etc.). Al novembre de 2010, Indústries Adafruit va oferir una recompensa per a qui trobés un controlador de codi obert per Kinect. El 10 de novembre es va anunciar l'espanyol Héctor Martín com el guanyador, que va usar mètodes d'enginyeria inversa amb Kinect i va desenvolupar un controlador per a GNU/Linux que permet l'ús de la càmera RGB i les funcions de profunditat. L'enginyeria inversa consisteix a obtenir informació o un disseny a partir d'un producte accessible al públic, amb la finalitat de determinar de què està fet, què el fa funcionar i com va ser fabricat. Aquest tipus d'enginyeria sovint és utilitzada per reprogramar el producte final i aconseguir una altra utilitat; malauradament aquesta enginyeria també és utilitzada per molts pirates informàtics per reprogramar el programari i així evitar els sistemes antipirateria programats dins el programari.

Un controlador de codi obert vol dir que les bases de programació que han utilitzat per fer-lo estan a l'abast de tothom sense protecció de la propietat intel·lectual per poder-lo modificar al gust de l'usuari. Gràcies a aquest controlador de codi obert van començar a aparèixer diferents programes que donaven diverses utilitats alternatives.

L'aparició d'aquest codi també va suposar que apareguessin nous fabricants amb dispositius que milloraven la qualitat. Asus va treure al mercat un dispositiu anomenat Xtion PRO Live, i PrimeSense va treure'n un altre anomenat Carmine. Tot i tenir millor maquinari els resultats no varien massa i la Kinect de Microsoft ofereix una millor qualitat de controladors per al dispositiu.

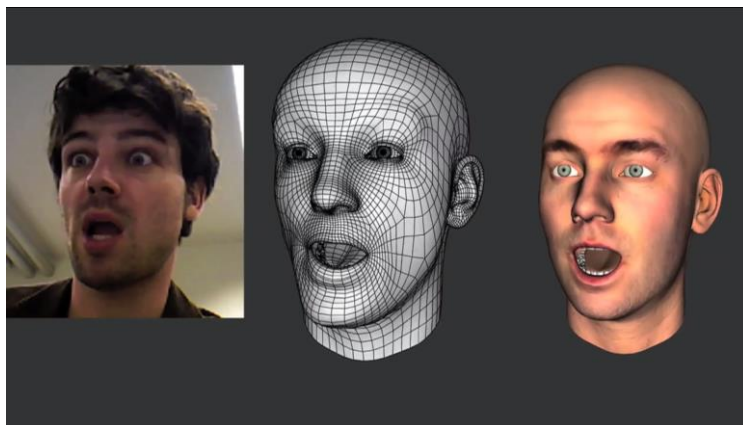
Captura de Moviment

Una de les utilitats va ser l'opció d'utilitzar-la com a escàner d'objectes tridimensionals: el sensor infraroig detecta la forma de l'objecte el qual s'ha de rotar davant del sensor fins que detecti tota la superfície i la càmera RGB detecta quin és el color de la figura i crea una textura per al model tridimensional. Els resultats de l'escanejat no són perfectes però tenen una precisió bastant bona, com els mostrats a la fotografia del lateral. Hi ha molts programes i moltes variants dels programes al mercat que realitzen aquest escanejat, els més populars són ReconstructMe o Skanect.



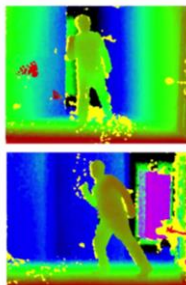
Resultats d'utilitzar la Kinect com a escàner tridimensional
©Skanect

Entrant en el tema d'animació, es van començar a crear programes que eren capaços de fer una captura de moviment facial bastant precisa a temps real. Funcionen amb qualsevol llum i són capaços de detectar inclús si tens els ulls



En la primera imatge es mostra el vídeo RGB natiu de la càmera, en la segona el resultat de l'animació sense aplicar-hi les textures i l'ombrejat, i en la tercera hi ha el resultat de l'animació amb les textures i l'ombrejat aplicat. ©Faceshift

tancats o oberts. Aquest tipus de programes van aconseguir un gran èxit i actualment són utilitzats en estudis de captura de moviment com el de LucasArts. Un dels programes més populars s'anomena Faceshift.



Les dues imatges de la dreta són les que extreu el sensor de profunditat. Es representen amb una escala de colors on blau és la distància més llunyana i vermell la més propera. La imatge de l'esquerra és el resultat final de l'animació.

Continuant amb el tema d'animació van començar a aparèixer programes que permetien una captura de moviment corporal amb una precisió bastant elevada. La majoria funcionen a temps real amb una Kinect, però n'hi ha que

accepten dues Kinects amb la conseqüència de no ser una captura a temps real. Un dels programes més coneguts i que té més bones crítiques

s'anomena iPi Studio. Aquest programa no només funciona amb Kinect/Xtion/Carmine, sinó que també és compatible amb dues Kinect/Xtion/Carmine, la càmera Sony PS Eye (fins a un màxim de 6), amb els comandaments Playstation Move o amb els comandaments de la popular consola Wii. També disposa de filtres de processament de l'animació per refinar els errors que pugui causar la Kinect.

5.4. Avantatges i inconvenients respecte als sistemes Vicon

Avantatges:

- Té un cost molt baix, el preu ronda entre 50 i 150 €
- Funciona amb la gran majoria d'ordinadors de gamma mitjana i baixa; els requisits són els següents:
 - CPU: compatible x86 (Intel Pentium 4 o superior, AMD Athlon o equivalent), de doble o quàdruple nucli preferiblement
 - Sistema operatiu: Windows 8, 7, XP SP3, Vista (x86 o x64)
 - USB: almenys dos ports USB 2.0 o USB 3.0
 - Sistema d'emmagatzematge: HDD o SSD o RAID amb una velocitat d'escriptura d'almenys 55 Mbytes/s
- Captura el moviment sense la necessitat de marcadors
- Capacitat de realitzar la captura en qualsevol lloc i amb qualsevol llum, l'espai recomanat és d'uns 5-6 m²
- Els dispositius es calibren en menys d'un minut

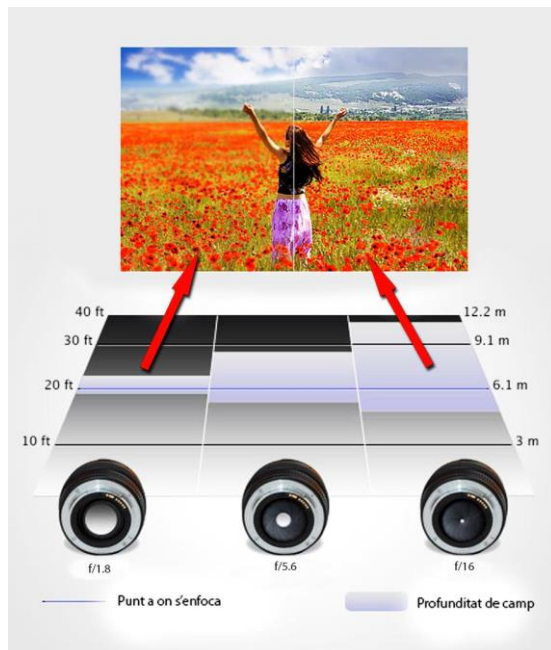
Inconvenients:

- Només es pot capturar amb una freqüència de 60 fps com a màxim
- El marge d'error no és tan baix com en els sistemes Vicon, però igualment tenen un marge d'error molt baix
- Tot i que la majoria de programes funcionen a temps real, els programes més potents i els que admeten dues Kinects simultàniament no són capaços de realitzar captura de moviment a temps real
- Només es poden connectar dues Kinects com a màxim

6. Part pràctica

6.1. Rodatge

És la part més important de totes, ja que si es grava mal material després fa falta més feina d'edició i postproducció. Cal tenir cura de tres factors, el primer és el desenfocament de la càmera, el segon és el desenfocament de moviment i el tercer és l'aparició de gra.



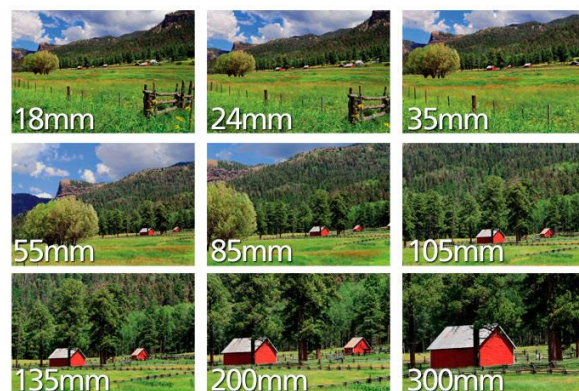
El desenfocament de la càmera es produeix a causa d'una baixa profunditat de camp. La profunditat de camp és una distància dins de la qual els objectes queden enfocats, si els objectes es troben fora d'aquesta distància queden desenfocats. Per reduir aquest efecte és necessari anar buscant un diafragma i una distància focal adequats.

El diafragma funciona de manera semblant a la pupil·la humana, com més obert estigui més llum entra dins el sensor, i com més tancat estigui menys llum entra. Per mesurar el diàmetre d'obertura del diafragma s'utilitza la següent divisió: $D = f/N$, on N és un nombre més gran o igual que 1 i expressa una constant, f es el diàmetre màxim d'obertura del sistema i D es el diàmetre d'obertura del diafragma. S'expressa en forma de relació entre f i N on f no s'expressa amb un nombre (ex: f/2,8, f/5,6, f/22, etc.) i es llegeix com "efa dos coma vuit", "efa cinc coma sis", "efa vint-i-dos", etc. Amb altres paraules, vol dir que si tenim un f/5 per exemple, l'obertura del diafragma és 5 vegades més petita que la total. Com més alt és el valor d'N, més profunditat de camp hi ha, i a l'inrevés. L'altre aspecte que hem de tenir en compte és la distància focal, que és la distància que hi ha entre la lent de l'òptica i el sensor, que és on queda projectada la imatge. Com més distància focal té una òptica, menys profunditat de camp té i la imatge que queda projectada al sensor és més llunyana, i a l'inrevés. S'ha de tenir en compte que no totes les càmeres porten la mateixa mida de sensor. Aquests variacions de la mida del sensor fan que la distància focal que percep el sensor variï. Com que la majoria de òptiques

El diafragma funciona de manera semblant a la pupil·la humana, com més obert estigui més llum entra dins el sensor, i com més tancat estigui menys llum entra.

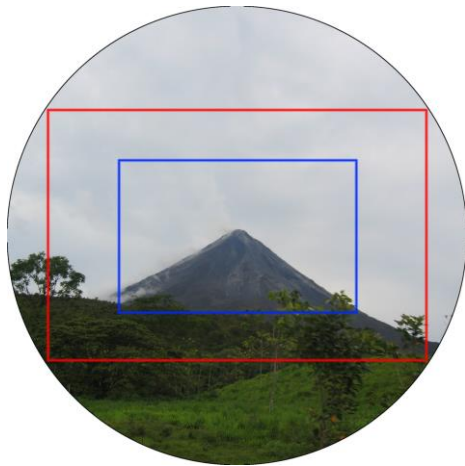
Per mesurar el diàmetre d'obertura del diafragma s'utilitza la següent divisió: $D = f/N$, on N és un nombre més gran o igual que 1 i expressa una constant, f es el diàmetre màxim d'obertura del sistema i D es el diàmetre d'obertura del diafragma. S'expressa en forma de relació entre f i N on f no s'expressa amb un nombre (ex: f/2,8, f/5,6, f/22, etc.) i es llegeix com "efa dos coma vuit", "efa cinc coma sis", "efa vint-i-dos", etc. Amb altres paraules, vol dir que si tenim un f/5 per exemple, l'obertura del diafragma és 5 vegades més petita que la total. Com més alt és el valor d'N, més profunditat de camp hi ha, i a l'inrevés. L'altre aspecte que hem de tenir en compte és la distància focal, que és la distància que hi ha entre la lent de l'òptica i el sensor, que és on queda projectada la imatge. Com més distància focal té una òptica, menys profunditat de camp té i la imatge que queda projectada al sensor és més llunyana, i a l'inrevés. S'ha de tenir en compte que no totes les càmeres porten la mateixa mida de sensor. Aquests variacions de la mida del sensor fan que la distància focal que percep el sensor variï. Com que la majoria de òptiques

hem de tenir en compte és la distància focal, que és la distància que hi ha entre la lent de l'òptica i el sensor, que és on queda projectada la imatge. Com més distància focal té una òptica, menys profunditat de camp té i la imatge que queda projectada al sensor és més llunyana, i a l'inrevés. S'ha de tenir en compte que no totes les càmeres porten la mateixa mida de sensor. Aquests variacions de la mida del sensor fan que la distància focal que percep el sensor variï. Com que la majoria de òptiques



Aquests variacions de la mida del sensor fan que la distància focal que percep el sensor variï. Com que la majoria de òptiques

Captura de Moviment



En aquesta imatge es mostra en vermell com es veuria aquesta imatge amb un sensor de format complet i en blau ens ensenya com es veuria aquesta imatge amb un sensor APS-C si els dos sensors duguessin la mateixa òptica.

marquen la distància focal per a un sensor de format complet (24 × 36 mm) és necessari utilitzar el factor de multiplicació de la distància focal per saber la distància focal real. Aquest factor de multiplicació és el valor numèric pel qual s'ha de multiplicar la distància focal d'un objectiu per determinar la distància focal equivalent a una càmera de format complet, per tal de saber quin objectiu seria necessari en aquest format. Com més petit és el sensor, el factor és més gran i a l'inrevés. És a dir, si tenim una càmera amb sensor APS-C i una òptica de 25 mm i el factor de multiplicació focal en aquest tipus de sensors és d'1,5, realment gravarem a una distància focal de 37,5 mm. Per fer aquesta classe de vídeos l'important és que hi hagi una profunditat de camp bastant alta per evitar

desenfocaments que provocarien errors de *motion tracking*, que explicarem més endavant. També s'ha de vigilar perquè una distancia focal molt baixa podria causar distorsions al voltant de la imatge que també provocarien errors de *motion tracking*.

El desenfocament de moviment és causat per un moviment bruscat de la càmera a una baixa velocitat d'obturador. La velocitat d'obturador és el temps que està l'obturador obert. L'obturador és una espècie de cortineta situada davant del sensor que baixa i puja cada vegada que es captura una imatge. La velocitat d'obturador es mesura en segons o en fraccions de segons, 1 s, 2 s, 1/20 s, 1/60 s, 1/250 s. Com més tardi a fer un cicle, entrarà més llum i crearà més desenfocament de moviment i a l'inrevés. El desenfocament de moviment en vídeo és pràcticament necessari sempre perquè l'ull humà està acostumat a veure-ho en les televisions, el cinema... i si s'elimina l'espectador pot notar una pèrdua de realisme en l'escena i que un moviment molt bruscat de càmera provoqui certa dificultat de visió o inclús mal d'ulls. Un altre problema és que aquesta velocitat d'obturador s'ha de compaginar amb la freqüència que té la càmera en capturar imatges. Si una càmera està gravant a



Diferències entre una imatge amb desenfocament de moviment (esquerra) i una altra sense (dreta).

25 fps i té una velocitat d'obturador de 1" el que passarà serà que les 25 imatges gravades durant aquest segon seran pràcticament iguals i donarà el mateix resultat que si es gravés a 1 fps, sense comptar l'excessiu desenfocament de moviment que produiria. El recomanable perquè no succeeixi això és utilitzar la següent fórmula per saber la velocitat d'obturador mínima per a que no hi hagin fotogrames duplicats: $\frac{1}{(fps \times 2)}$ segons. Un exemple pràctic d'aquesta fórmula seria en una càmera de 25 fps, que es la freqüència comuna de les càmeres de vídeo actuals, on la velocitat d'obturador mínima seria $\frac{1}{(25 \times 2)}$ segons, es a dir, 1/50 s. En aquest tipus de vídeos és necessari una velocitat d'obturador alta per evitar els desenfocaments de moviment que ens causaran errors de *motion tracking*. Posteriorment s'afegirà aquest desenfocament de moviment mitjançant la postproducció per evitar les repercussions mencionades anteriorment.



Diferències entre una imatge amb gra (esquerra) i una altra sense (dreta)

a mostrar defectes que s'expressen en forma de gra a la imatge, són com una espècie de granets que es mouen per dins la imatge. Aquests grans també produeixen errors de *motion tracking*, per tant, s'ha d'utilitzar una baixa sensibilitat per realitzar aquest tipus de vídeos. Un sensor més gran o de millor qualitat permet utilitzar sensibilitats més altes sense l'aparició de gra.

L'aparició de gra és un defecte del sensor quan hi actua una sensibilitat molt alta. Els sensors digitals són capaços de regular la sensibilitat que tenen a la llum que els arriba. Com més sensible sigui el sensor més llum tindrà el vídeo i. La sensibilitat es mesura amb dB (decibels). Quan hi està actuant una sensibilitat molt alta aquest guany artificial comença

Per fer aquest curtmetratge vam utilitzar una càmera Sony NEX-VG20 que té un sensor APS-C (s'ha de multiplicar la distància focal per 1,5) i grava a una definició de 1080p amb una freqüència de 25 fps. En aquesta càmera hi havia connectada una òptica que té una distància focal variable que va des de 16 mm (24 mm amb el nostre sensor) fins a 50 mm (75 mm amb el nostre sensor) amb una obertura màxima d' $f/2,8$ i una mínima

Captura de Moviment

d'f/22. En el cas particular del nostre curtmètratge, era un dia amb no gaire sol però prou com per poder gravar en condicions òptimes. En les escenes on s'havia de fer *motion tracking* vam utilitzar un diafragma f/9, amb una velocitat d'1/250 s i una sensibilitat de 12 dB (imatge de la dreta). En les escenes en què sortia només l'actor vam utilitzar un diafragma f/2'8 per crear una baixa profunditat de camp i ressaltar l'actor, una velocitat de 1/100 s i una sensibilitat de 0 dB (imatge de l'esquerra).



6.2. Codificació de vídeo

Tindre una noció de codificació és essencial a l'hora d'editar. Cada càmera utilitza el seu format que es podria dir que és com una espècie de contenidor, dins de cada format hi ha la informació sobre com és el vídeo. Els formats poden ser mov, mp4, mpeg, avi... Però els formats no controlen el llenguatge amb que està escrit aquest vídeo ni la compressió que té aquest vídeo. Aquesta és la tasca del còdec. El còdec (codificador i descodificador) és el tipus de llenguatge que utilitza l'arxiu per a la seva gravació i reproducció, també és l'encarregat de la compressió del vídeo. Un dels principals errors que es produeix al parlar és dir: "Els arxius .mov, .mp4... no me'ls llegeix el meu ordinador". Aquesta afirmació es errònia ja que qui conté la informació és el còdec i no el format. Poden existir arxius .mov, .mp4... que utilitzin el mateix còdec, per tant, el correcte seria dir: "El meu ordinador no llegeix aquest tipus de còdec". Els dos còdec que utilitzarem en aquest procés seran l'Apple Prores 442 o Avid DNxHD i l'H.264.

Els còdec Apple Prores 442 i l'Avid DNxHD són còdec bastant similars i per això els englobarem els dos junts. La principal diferència entre ells és que l'Apple Prores 442 només funciona amb sistemes operatius Mac i l'Avid DNxHD és compatible amb la majoria de sistemes operatius actuals. El seu funcionament és, bàsicament, guardar cada fotograma del vídeo com si fos una imatge com es mostra en la imatge següent.



L'avantatge és que ofereix molta qualitat i si l'ordinador ha de buscar la informació d'un fotograma en especial ja la té tota disponible en ell, però el principal inconvenient és que ocupa molt espai al disc dur.

El còdec H.264 és un dels més utilitzats actualment i un dels que ha revolucionat més la indústria del vídeo. Aquest còdec utilitza diferents formes de compressió per reduir l'espai al disc dur. La primera que explicarem és la compressió espacial. La compressió espacial és un tipus de compressió que fa el còdec amb els píxels d'una zona específica que tenen un color bastant similar. En comptes de guardar la informació de cada un dels píxels, agrupa tots els que tenen un color similar i crea un *chunk* on només hi ha un valor d'informació i s'aplica a tots ells, així aconseguim evitar la repetició d'informació. En aquesta imatge



que es mostra aquí es veu subratllat en vermell en quines zones s'ha generat un *chunk*. El següent tipus de compressió que explicarem es la compressió temporal. La compressió temporal consisteix a agafar les parts del fotograma que no experimenten

cap canvi respecte els fotogrames anteriors, eliminar-les d'aquell fotograma i dir-li al reproductor que en aquelles parts vagi a buscar la informació que hi pertoca als fotogrames anteriors. És a dir, si una part del vídeo es manté constant al llarg de 200 fps, només agafa la informació del primer fotograma i els altres 199 fotogrames van a



buscar la informació del primer. En aquesta imatge es mostra a quines parts del fotograma el còdec fa una compressió temporal ja que no hi ha res que canviï al llarg del vídeo. L'últim tipus de compressió que utilitza l'H.264 és l'anomenada compensació del moviment. El còdec és

capaç de detectar moviments simples de la càmera i agafar les parts del fotograma que estan repetides en fotogrames anteriors però en diferent lloc. En altres paraules, seria com una espècie de compressió temporal però que també té en compte si la regió que es repeteix s'ha mogut. En la següent imatge demostra que si se solapa el primer fotograma amb l'últim les gravadores estan repetides tot i que estiguin en diferent lloc del fotograma, per tant, en l'últim fotograma les gravadores no hi seran i agafaran la informació del primer fotograma.

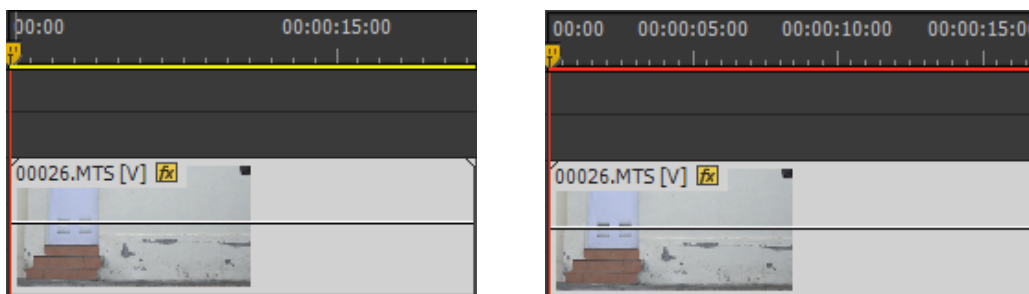
L'avantatge que ofereix aquest còdec és que a l'eliminar la informació repetida ocupa molt menys espai al disc dur, però l'inconvenient és que requereix un processador més potent per a la edició, ja que al mateix temps que li apliques un efecte ha de buscar la informació en fotogrames anteriors.



Cada vegada que insereixes un efecte, canvies l'opacitat del vídeo o qualsevol altra correcció que li apliquis al vídeo, l'ordinador et demana fer *render*. *Render* en l'edició vol dir que l'ordinador calcula com seran tots els fotogrames amb la correcció aplicada. Els programes no poden reproduir el vídeo sense aplicar-hi el *render* primer. Un cop ha fet el *render*, només que canviïs un valor als efectes provoca un altre *render*. Això a la llarga pot suposar una considerable pèrdua de temps. En còdecs com l'H.264, els temps de *render* són més extensos i més freqüents que en DNxHD o en Prores 442. Això causava que si es volia editar amb la mateixa rapidesa que es fa amb DNxHD o Prores 442 necessitaves tindre un processador més potent. Per contrarestar això, Adobe va aprofitar la tecnologia que ens ofereix la targeta gràfica. El processador calcula els processos pas per pas, mentre que la targeta gràfica pot calcular múltiples processos al mateix temps. Si utilitzem la targeta gràfica per fer *render*, es pot calcular la informació emmagatzemada en fotogrames anteriors i aplicar les correccions al mateix temps. El resultat és un temps de *render* molt inferior i la capacitat d'aplicar moltes més correccions al vídeo abans que ens demani fer *render*. En la imatge

Captura de Moviment

següent es mostra el mateix clip de vídeo però a la dreta utilitzem el processador per la línia de temps i a l'esquerra utilitzem la targeta gràfica. Si li apliquem el mateix canvi en l'opacitat dels dos clips de vídeo podrem començar a observar que en la que funciona per processador ja ens demana fer *render*, mentre que la que utilitza targeta gràfica no. Però per altra banda, el petit inconvenient que se li podria trobar és que aquesta línia de temps només funciona amb targetes gràfiques nVidia compatibles amb la tecnologia CUDA. CUDA és un tipus de controlador de la targeta que permet processar paral·lelament múltiples tasques al mateix temps.



La línia vermella de la imatge de la dreta simbolitza que és necessari fer un *render* per continuar mentre que la línia groga de la imatge de la dreta simbolitza que no és necessari.

6.3. Edició inicial

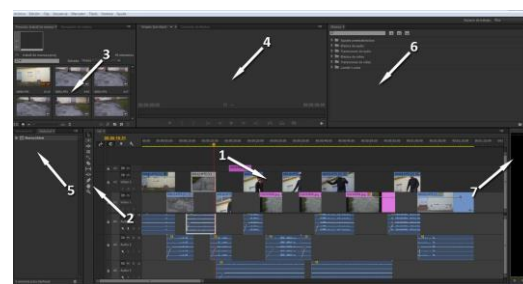
L'edició és la part del procés on s'agafen els clips gravats, s'elegeixen els millors i es fan els talls per seleccionar la part útil de cada clip. Actualment hi ha molts programes d'edició que ofereixen uns resultats molt bons amb poc temps. Alguns dels més utilitzats són el Final Cut i l'Adobe Premiere. Final Cut només



La icona dels programes Final Cut Pro X (esquerra) i Adobe Premiere CC (dreta)

és compatible amb sistemes operatius Macintosh i Adobe Premiere és compatible amb sistemes operatius Windows i Macintosh. Tot i que he tingut l'oportunitat de provar els dos, per realitzar aquest curtmetratge m'he decantat per l'Adobe Premiere. Un dels principals motius d'aquesta elecció ha sigut que actualment només dispo d'un ordinador amb sistema operatiu Windows. A més a més, l'antiga versió de Final Cut no està ben preparada per treballar amb el còdec H.264 ja que quan es va crear el programa aquest còdec encara no existia, això provoca que cada vegada que hagi d'editar he de convertir els arxius nadius de la meua càmera amb la consegüent pèrdua de temps i qualitat. D'altra banda la nova versió de Final Cut, que sí és compatible amb l'H.264, utilitza una línia de temps (explicarem què és més endavant) amb la qual trobo que és molt difícil treballar. A més a més, Adobe Premiere ofereix una perfecta compatibilitat amb Adobe After Effects i Adobe Photoshop, dos eines que utilitzarem molt per aquesta part pràctica; és compatible amb la majoria de còdecs actuals i ofereix l'opció d'utilitzar la targeta gràfica en la línia de temps en comptes del processador. En el nostre cas, la nostra càmera treballa amb H.264, per això elegirem l'opció d'utilitzar la targeta gràfica en la línia de temps.

El primer que farem serà fer una edició inicial que ens servirà de referència per insertar els clips de vídeo on hi hagin les captures de moviment. S'importen els clips de vídeo, es retallen les parts no desitjades i ja s'està a punt per començar a inserir els efectes. De moment deixarem aquesta línia de temps així i ja l'acabarem de perfeccionar més endavant. El resultat d'aquesta edició inicial quedarà com en la imatge que es mostra a continuació:



En aquest cas vam utilitzar una configuració de dues pantalles per poder gaudir d'una visualització del vídeo a mida real. Com es mostra en la imatge següent, en una pantalla hi haurà tota la línia de temps (1), la paleta d'eines (2), la paleta on s'importen els clips de vídeo (3), la paleta on s'editen els efectes (4), l'historial de tot el que s'ha fet (5), la biblioteca d'efectes (6) i els nivells d'àudio (7) i en l'altra pantalla hi ha una visualització del vídeo (8).

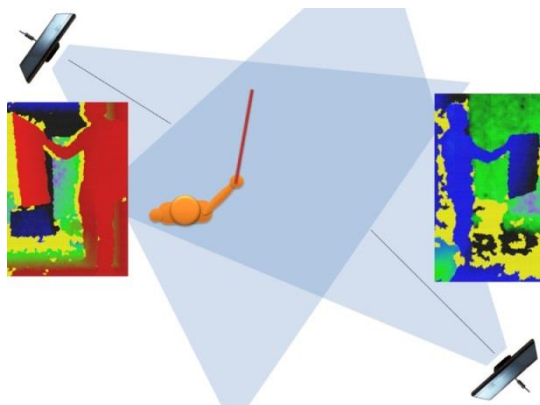
6.4. Captura de moviment

Per fer la captura de moviment utilitzarem el programa iPiSoft i una configuració de dues Kinects a 180° i una distància de cinc metres de separació entre elles. La configuració de 180° l'hem elegida després de moltes proves. Hem obtingut resultats més bons amb aquesta configuració ja que permet obtenir un núvol de punts de les dues bandes, mentre que amb una configuració de 90° només s'obté una imatge frontal i una de lateral, deixant una zona fora de la visió de la càmera. També s'ha de vigilar que no interfereixin els infrarojos emesos per les dues Kinects. Per contrarestar això vaig col·locar les Kinects a diferents altures. Una d'elles se situava damunt d'una tauleta d'uns 70 cm d'altura (esquerra) i l'altra a uns 120 cm d'altura situada damunt d'una estanteria (dreta).



Captura de Moviment

Un cop tenim les dues Kinects connectades a l'ordinador, toca fer el procés de calibratge. Per fer-ho només cal un cartró o qualsevol peça rectangular de mínim 0,5 m x 0,7 m i màxim 1 m x 1,5 m i preferiblement que no superi els 3 cm de gruix. En aquest cas, vaig utilitzar un pòster de 0,55 m x 0,85 m. Un cop tenim això, s'ha de fer una gravació movent-ho dins de l'angle de visió de les dues Kinects de forma que des de cada càmera es pugui veure una cara de la superfície rectangular, com es mostra en la imatge. El programa calibra l'escena basant-se en la posició de la superfície rectangular gravada amb cada càmera.



Un cop està fet això ja està tot calibrat, el programa calcula quin dels núvols de punts és l'actor i quin és el fons de la imatge. Per fer-ho és necessari que l'actor surti d'escena i que les càmeres gravin una seqüència de 10 s cada una per separat. El programa utilitza els núvols de punts diferents a les dues primeres seqüències per obtenir només els núvols de punts que representen l'actor aïllats.

En aquest moment ja es pot començar a gravar. El primer que s'ha de fer és que l'actor es col·loqui al centre de l'escena i es posi en forma de T durant 2 s, com es mostra en la imatge següent. Acte seguit ja pot començar a fer els moviments necessaris per a l'escena, sempre evitant sortir de l'angle de visió de les dues càmeres. En les imatges següents es pot apreciar la posició en T acabada d'esmentar i una imatge del procés de gravació.

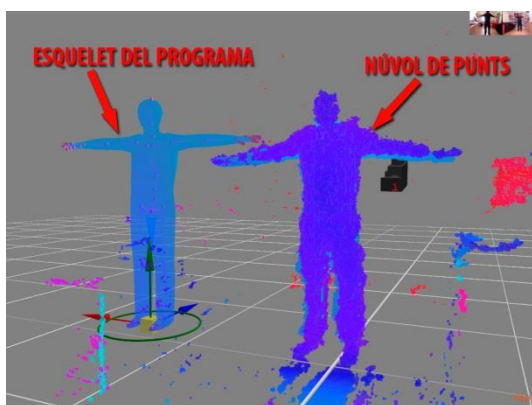


Captura de Moviment

En aquest cas també vaig poder utilitzar un micròfon de corbata com el que es mostra a continuació per gravar la veu del actor. Posteriorment utilitzarem aquest arxiu de veu per animar la boca del personatge generat per ordinador.

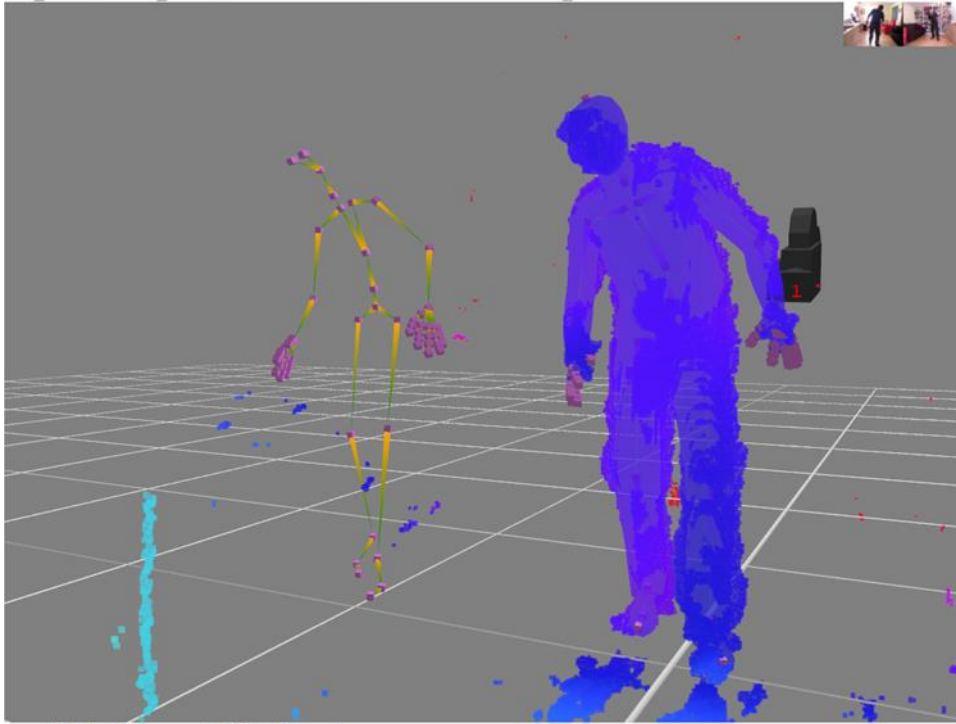


Un cop s'ha gravat l'escena s'agafen els núvols de punts generats per les càmeres i ja es posa l'esquelet que genera el programa dins del núvol de punts. L'esquelet per defecte està en forma de T, per això és tan important que l'actor comenci amb aquesta postura al principi de la gravació. El programa detecta automàticament el moviment d'una forma bastant precisa, però igualment es pot fer de forma manual en cas que el programa ho faci erròniament. Després, a aquest moviment se li poden aplicar correctors de moviment perquè sigui un moviment més fluid i suau. En aquestes imatges que podem veure a continuació es pot observar el procés de posar l'esquelet en forma de T dins del núvol de punts i un imatge de l'esquelet dins del núvol de punts mentre el programa està calculant el moviment.



Captura de Moviment

Un cop el programa ja ha realitzat la captura de moviment s'importa l'esquelet que necessita l'editor 3D, en aquest cas el Poser, i se li transmet el moviment. Un cop transmès el moviment s'exporta aquest esquelet i ja està a punt per integrar-ho al personatge que posteriorment crearem. En aquesta imatge podem observar l'esquelet ja animat creat pel programa i al costat l'esquelet que necessita el Poser amb el moviment aplicat.



Captura de Moviment

Si obrim aquest arxiu de captura de moviment amb un editor de text o en qualsevol consola d'ordres podem observar quina estructura segueix per guardar el moviment. Com es mostra en la imatge següent, el que fa és guardar la informació mitjançant coordenades i angles tridimensionals. Parteix de la base que l'abdomen es troba en les coordenades (0,0,0) i a partir d'aquí comença a agrupar totes les parts del cos en una espècie d'arbre on diu la posició de cada una respecte les altres. Si observem bé la següent imatge es pot veure com de la part anomenada "abdomen" surt una altra part anomenada "chest" (*pit* en català), i que d'aquesta part en surten dos més anomenades "neck" (*coll* en català), i "rCollar" (es podria entendre com *clavícula dreta* en català), i segur que si continuéssim llegint les 427 pàgines que té aquest document podríem trobar la clavícula esquerra i les parts del cos que falten. A més a més, dins d'aquestes tres parts pots anar veient les diferents parts que surten d'aquestes tres amb les seves coordenades corresponents.

```
GNU nano 2.0.6 File: again.bvh
HIERARCHY
ROOT hip
{
  OFFSET 0.0000 0.0000 0.0000
  CHANNELS 6 Xposition Yposition Zposition Xrotation Yrotation Zrotation
  JOINT abdomen
  {
    OFFSET -0.0679 2.1756 0.1870
    CHANNELS 3 Xrotation Yrotation Zrotation
    JOINT chest
    {
      OFFSET 0.0716 4.2985 -0.2866
      CHANNELS 3 Xrotation Yrotation Zrotation
      JOINT neck
      {
        OFFSET -0.2866 7.3791 0.3582
        CHANNELS 3 Xrotation Yrotation Zrotation
        JOINT head
        {
          OFFSET 0.2149 3.2239 0.4299
          CHANNELS 3 Xrotation Yrotation Zrotation
          JOINT leftEye
          {
            OFFSET 0.0734 1.0360 2.7436
            CHANNELS 3 Xrotation Yrotation Zrotation
            End Site
            {
              OFFSET 0.0000 0.0000 0.4288
            }
          }
          JOINT rightEye
          {
            OFFSET -0.0512 1.0259 2.7304
            CHANNELS 3 Xrotation Yrotation Zrotation
            End Site
            {
              OFFSET 0.0000 0.0000 0.4288
            }
          }
        }
      }
    }
  }
  JOINT rCollar
  {
    OFFSET -0.9313 5.8746 -0.8597
    CHANNELS 3 Xrotation Yrotation Zrotation
    JOINT rShldr
    {
      OFFSET -3.6537 -0.5015 0.9313
      CHANNELS 3 Xrotation Yrotation Zrotation
      JOINT rForeArm
      {
        OFFSET -8.2299 0.2635 -0.5022
        CHANNELS 3 Xrotation Yrotation Zrotation
        JOINT rHand
        {
          OFFSET -7.7345 -0.2595 2.8051
          CHANNELS 3 Xrotation Yrotation Zrotation
          JOINT rThumb1
          {
            OFFSET -0.5283 -0.1432 0.9612
            CHANNELS 3 Xrotation Yrotation Zrotation
            JOINT rThumb2
            {
              OFFSET -0.5318 -0.3786 0.9708
              CHANNELS 3 Xrotation Yrotation Zrotation
              JOINT rThumb3
              {
                OFFSET -0.5554 -0.0206 0.4283
                CHANNELS 3 Xrotation Yrotation Zrotation
                End Site
                {
                  OFFSET -0.7016 0.0000 0.4762
                }
              }
            }
          }
        }
      }
    }
  }
}

```

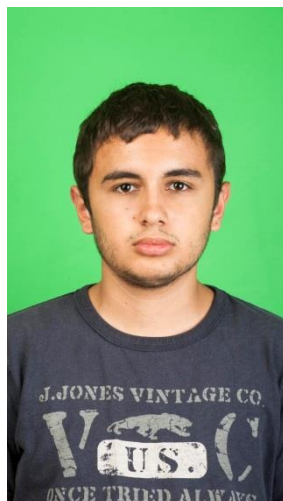
Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell

6.5. Modelatge i texturització

Modelatge és el procés en el qual es van generant polígons i es van ordenant de manera que quedi la forma desitjada i la texturització és el procés d'aplicar les textures desitjades al model creat anteriorment. Per fer el model i les textures utilitzarem el Poser i el Photoshop. El programa Poser per defecte ja té alguns models amb forma humana de molt alta qualitat. Per això partirem de la base d'un model fet i l'anirem personalitzant al nostre gust de manera que s'assembli al nostre actor. En aquest cas utilitzarem un personatge anomenat JamesCasual aprofitant que ja porta roba incorporada i així no ens caldrà modelar-la.

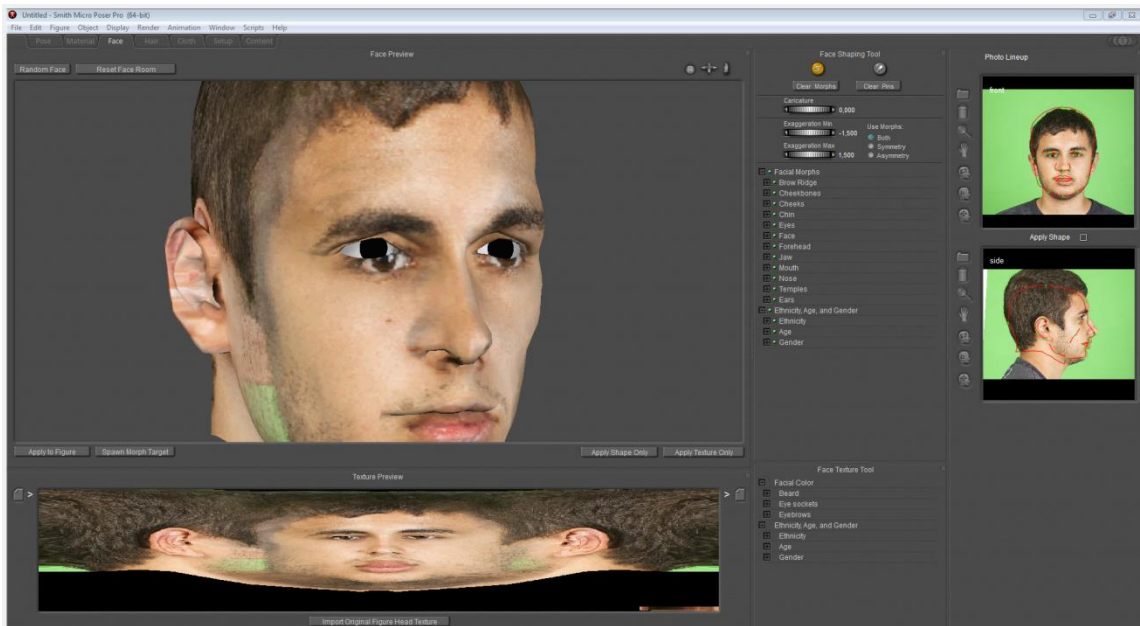


Un cop tenim importat aquest model des del programa, començarem a aplicar-hi les textures i la forma de la cara del actor. Per fer això aprofitarem un editor de cares tridimensional a partir de dues imatges integrat al programa. Agafarem dues imatges del actor (una frontal i l'altra de perfil) i les importarem al programa.



Captura de Moviment

Un cop tenim importades les fotografies hem d'intentar quadrar els punts marcats al lateral dret de la següent imatge amb la nostra fotografia.

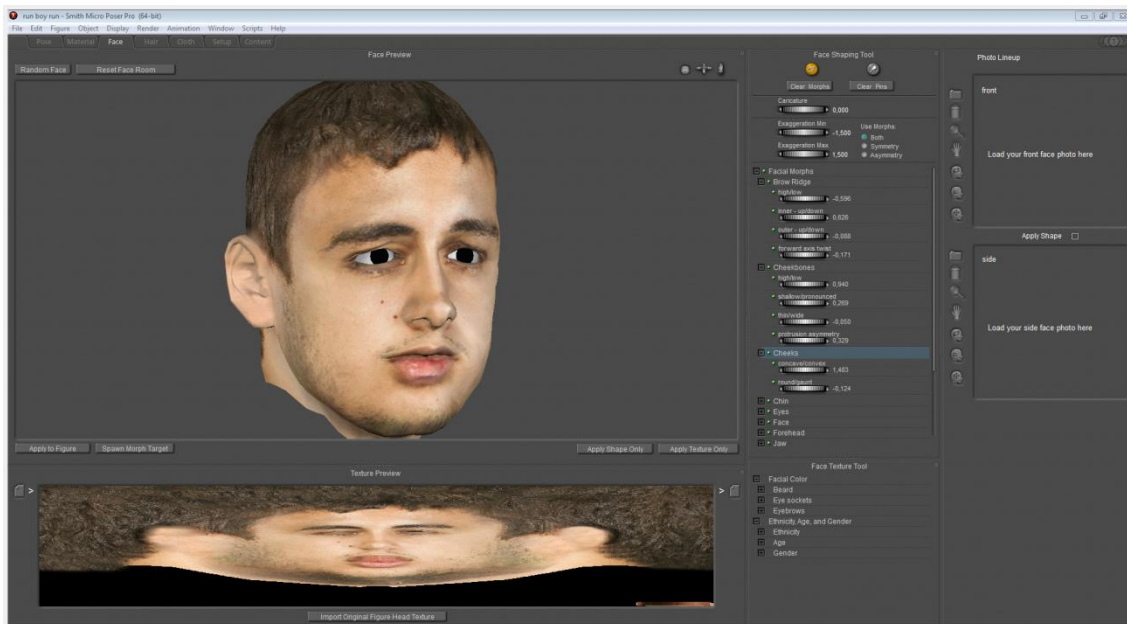


El següent que farem és agafar la imatge que hi ha en la part inferior de la fotografia anterior i l'importarem a Adobe Photoshop, on perfeccionarem la textura i eliminarem les zones on es veu color verd. Aquest procés d'agafar la textura d'una figura tridimensional i convertir-la en pla de dues dimensions per facilitar la feina de texturització s'anomena UV mapping. El resultat de l'UV mapping és com el que es mostra a continuació.



Captura de Moviment

L'últim pas consisteix a utilitzar les eines de deformació dels trets de la cara que ens ofereix aquest programa per acabar de modelar la cara i obtenir un resultat bastant similar al de l'actor. En la imatge següent es mostra el resultat final del modelatge.



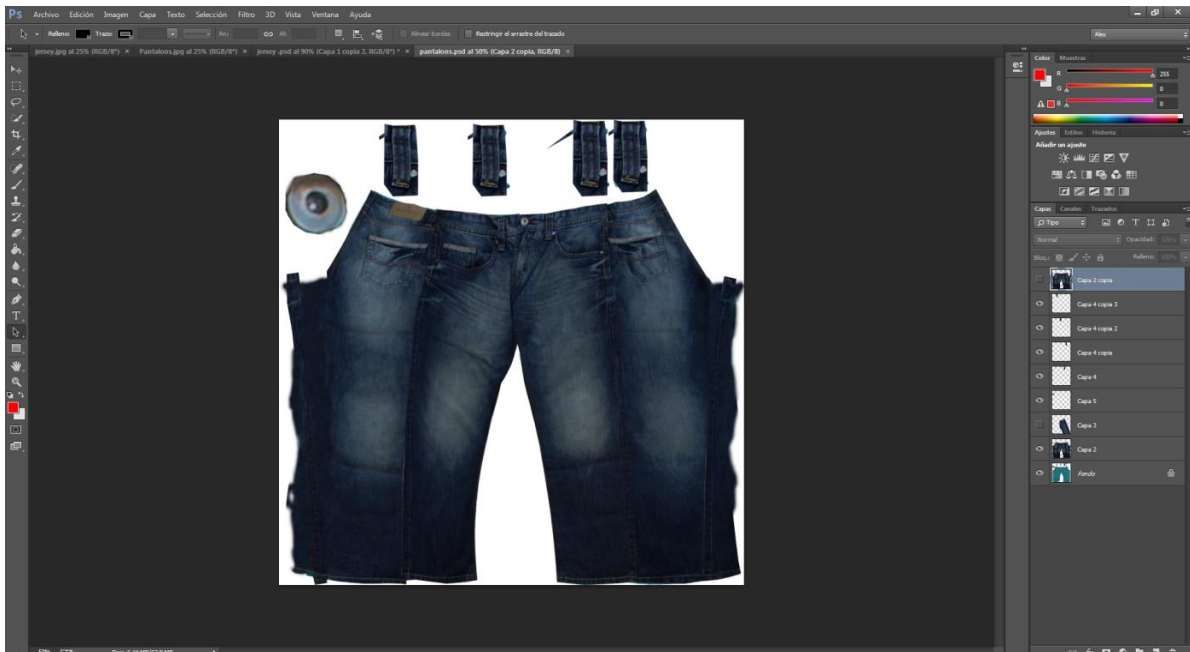
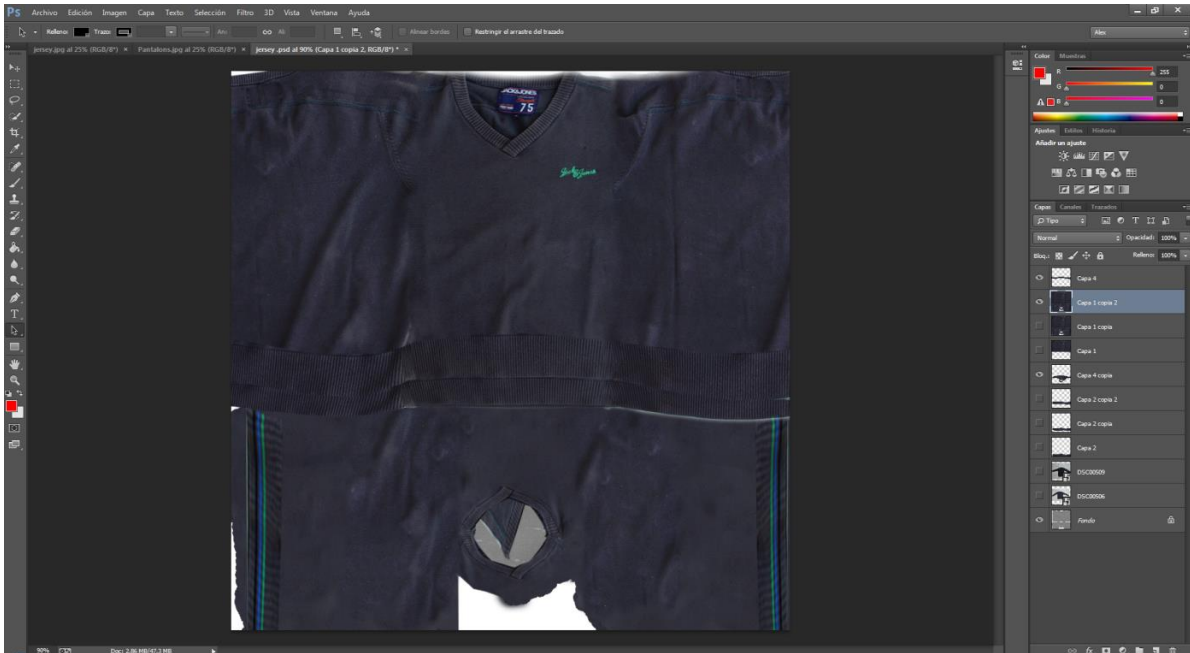
Pel que fa a la cara, el procés ja està, el següent pas és crear una textura per a la roba. Per fer-ho partirem de fotografies de la roba de l'actor.



Importarem aquestes fotografies a Photoshop i realitzarem UV mapping per a la textura de la roba. Mitjançant l'UV mapping i les fotografies realitzades anteriorment es dóna un cert realisme a la textura, ja que es poden observar les arrugues i els defectes del material.

Captura de Moviment

El resultat final quedarien com es mostra a continuació:

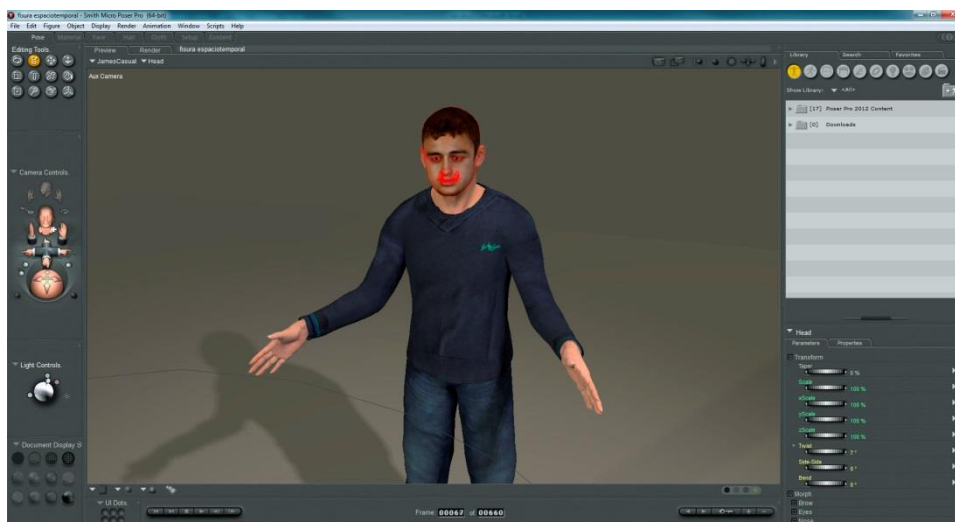


Captura de Moviment

Un cop realitzades les textures, ja està tot llest per tornar a importar les textures al Poser i acabar amb el procés de texturització. Un cop exportades les textures, el nostre personatge generat per ordinador queda així:

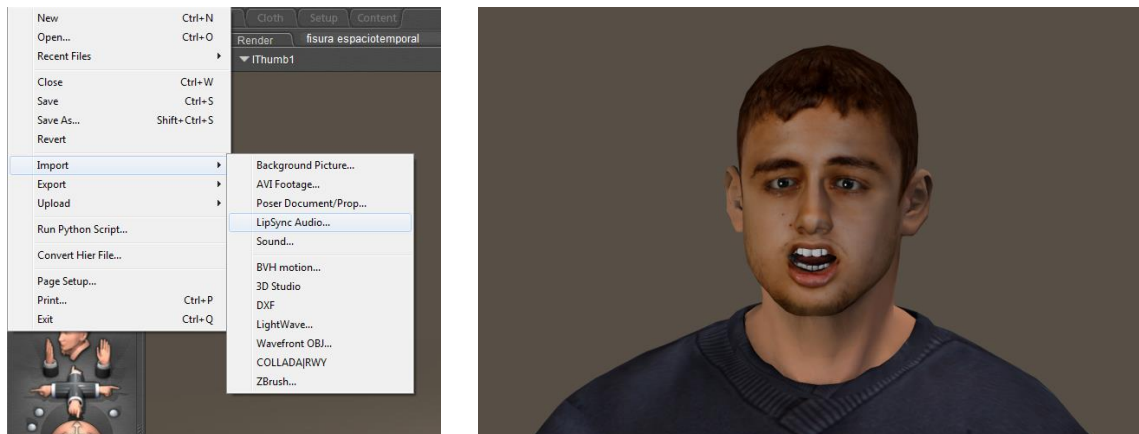


Aquest personatge ja està a punt per inserir-li les captures de moviment que havíem exportar anteriorment des del programa iPosoft. Importarem les animacions una per una en arxius de Poser diferents. Un cop importades les animacions haurem d'animar les mans i el cap de forma manual ja que la Kinect no és capaç de detectar el moviment en aquestes zones.



Captura de Moviment

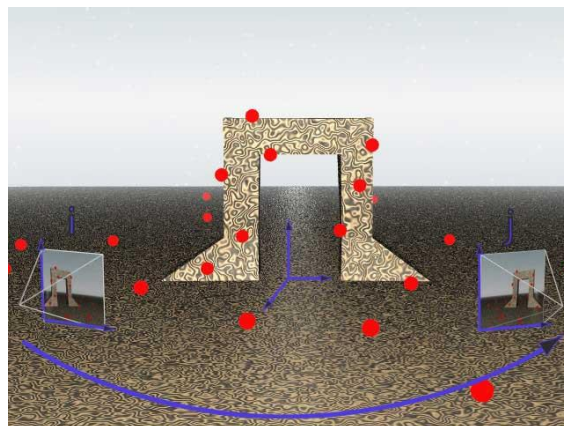
Posteriorment, mitjançant els arxius d'àudio gravats abans, procedirem a animar la cara del personatge. Per fer-ho només cal importar un arxiu d'àudio dins el programa, el qual ja calcula els moviments de la boca interpretant l'ona de l'àudio.



Un cop fet això ja tenim el nostre personatge modelat, texturitzat i animat.

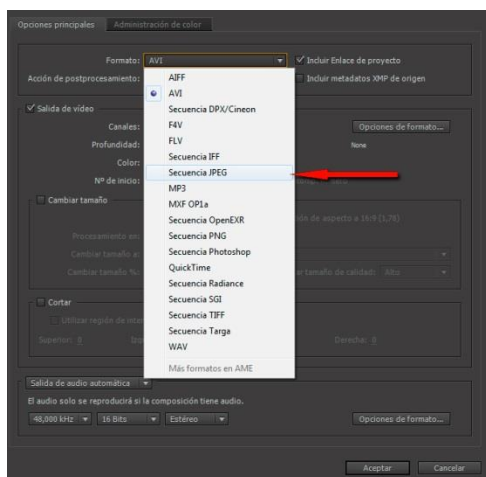
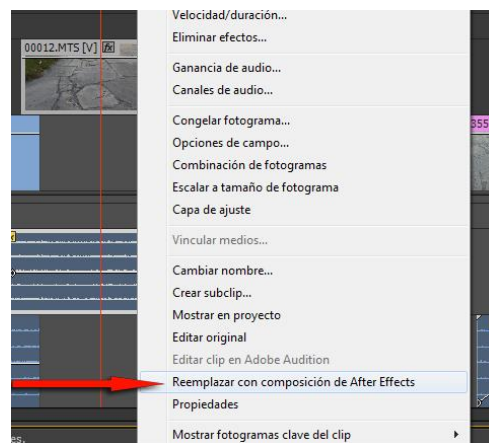
6.6. Motion tracking

Un cop tenim gravat el vídeo i tenim el nostre personatge ja animat, només ens fa falta inserir aquest personatge dins del nostre vídeo. El vídeo gravat anteriorment només disposa de dues dimensions i, si volem inserir el nostre personatge tridimensional, hem d'aconseguir transformar el nostre vídeo en un espai tridimensional per poder col·locar el personatge. A banda d'això, també hem de calcular la posició correcta, l'escala i el moviment de la càmera respecte a l'escena. Aquest procés és conegut amb el nom de *motion tracking*, *tracking 3D* o *match moving*. Per fer-ho el programa primer calcula punts de referència al llarg del vídeo, després triangula la posició entre ells mitjançant els canvis de perspectiva dels punts i automàticament calcula la geometria de l'escena, com es mostra en la següent imatge. Per fer-ho utilitzarem els programes After Effects i Boujou.



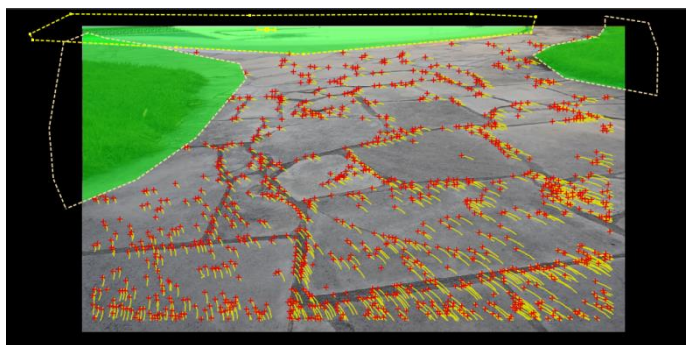
Captura de Moviment

El primer que hem de fer és anar a la línia de temps que havíem deixat apartada en la edició inicial i mirar quins són els clips en què volem inserir el personatge. Seleccionem quins volem i escollim l'opció d'obrir com una composició d'After Effects. After Effects és un programa del mateix fabricant que Premiere que ens permet afegir efectes al vídeo, crear animacions 2D i fer composicions d'imatges i vídeos.

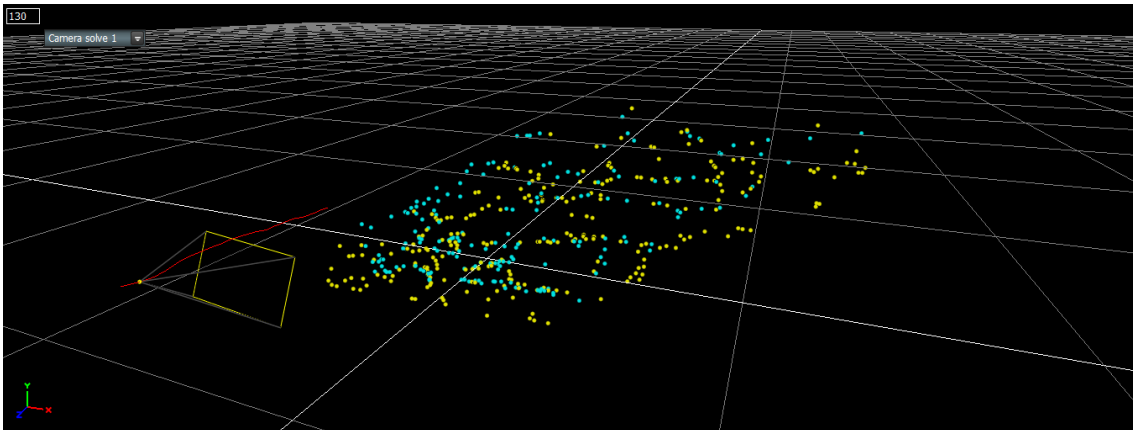


El programa de *motion tracking* no és capaç de reconèixer els punts de *tracking* d'un arxiu de vídeo, per això, un cop oberta la composició dins l'After Effects ho exportarem com a seqüència de fotografies.

S'importa la seqüència de fotografies al programa Boujou on es farà el *motion tracking*. Un cop importat es fa la primera part del procés, calcular els punts de referència del llarg del vídeo. També hem de tenir en compte que si a la nostra escena hi ha zones amb punts de referència mòbils (aigua, fum...), els hem d'eliminar perquè no causin errors de *tracking*. En aquest cas també eliminarem els punts de la zona d'herba ja que al ser una zona tan homogènia podria crear confusions al programa.



Un cop hem calculat els punts de referència calcularem els moviments de la càmera i la geometria de l'escena. Actualment els programes ja estan suficientment automatitzats com per poder calcular el moviment i la geometria de forma automàtica i precisa. En la imatge següent es mostra la geometria creada en aquesta escena simbolitzada amb punts blaus i grocs, el moviment de la càmera simbolitzat amb una línia vermella i la càmera, que està representada amb una piràmide on la base de la qual és on enfoca la càmera.

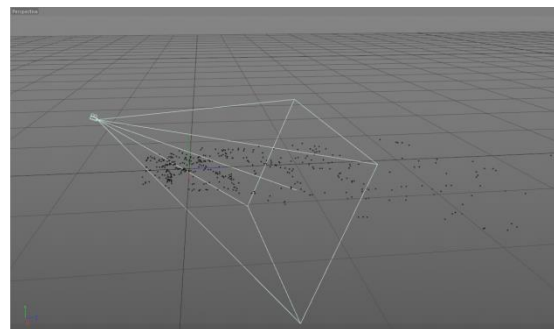


Un cop tenim el *motion tracking* acabat, l'exportarem per a Cinema 4D on realitzarem la composició 3D.

6.7. Composició 3D

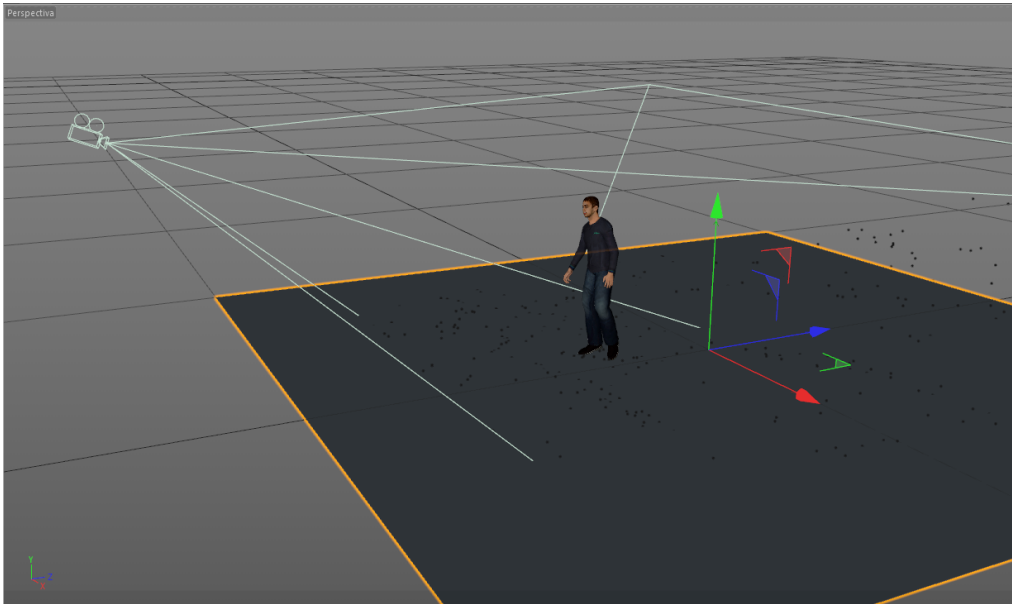
Un cop tenim el nostre personatge animat i hem calculat els moviments que fa la càmera, és el moment d'unir-ho tot dins el nostre vídeo. Per fer-ho utilitzarem el Cinema 4D ja que es un editor 3D que ens ofereix una compatibilitat perfecta amb els programes Poser i Boujou i també ens permet crear il·luminacions i ombres més realistes.

El primer que farem és obrir l'arxiu que hem exportat des de Boujou dins el Cinema 4D.



Captura de Moviment

El següent pas serà importar el nostre arxiu de Poser i crear un terra on es projectaran les ombres.



De fons col·locarem el vídeo on volem integrar el nostre personatge i farem que el terra on es projecten les ombres sigui una continuació de la imatge de fons.



Captura de Moviment

El següent que farem és col·locar un entorn a la nostra escena. L'entorn serveix per crear una il·luminació simple a l'escena i que el personatge pugui tenir reflexos. Per generar aquest entorn utilitzarem una fotografia de 360° realitzada el mateix dia de la gravació. Està feta amb una aplicació gratuïta per smartphones anomenada Photosynth. Tot i tenir alguns defectes en zones com l'herba i el fanal de l'esquerra, aquesta imatge es pot utilitzar perfectament com a entorn ja que aquests defectes no són apreciables en els reflexos. Aquesta fotografia es col·loca mitjançant l'UV Mapping a una esfera al voltant de la nostra escena.

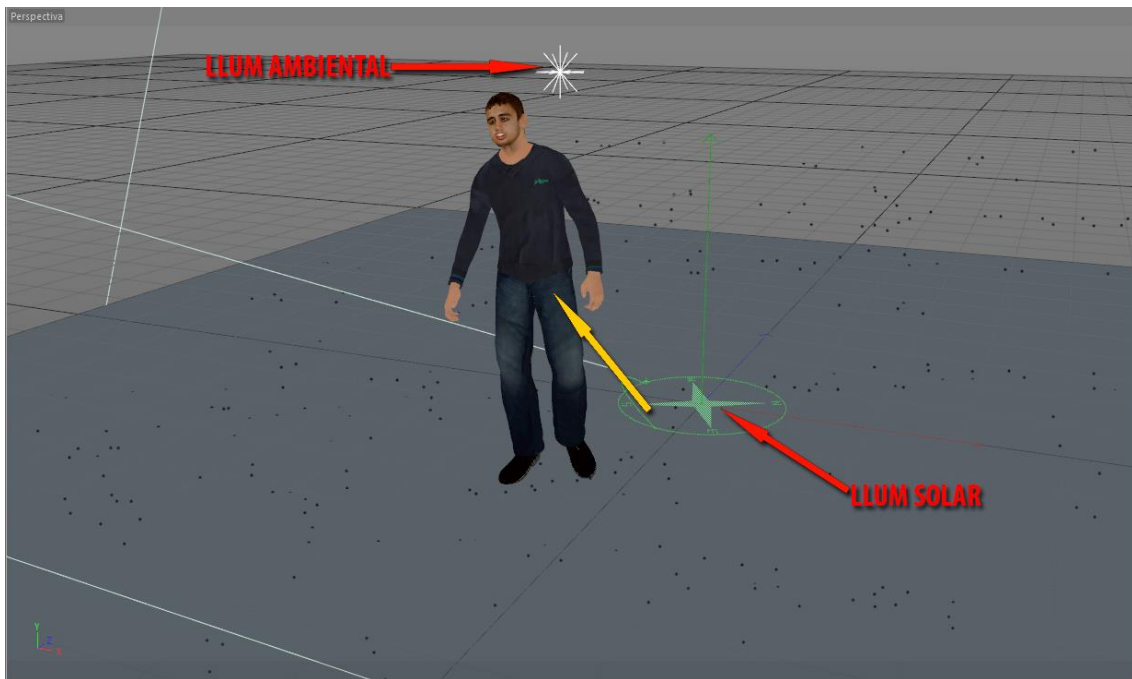


A continuació hi ha dues mostres del mateix model i amb la mateixa il·luminació però una amb entorn i l'altra sense. A simple vista ja es pot observar que la que no té entorn no mostra cap tipus de reflex a les zones on n'hi hauria i la que sí que en té reflecteix els objectes que té davant donant un resultat més realista a l'escena.



Captura de Moviment

Un cop tenim col·locats el nostre entorn, el personatge i la càmera del *motion tracking*, només ens fa falta una il·luminació realista per a l'escena. Per fer-ho vam realitzar un sistema de dues llums, una en forma de llum solar intentant coincidir amb la llum del sol que hi havia el dia de la gravació i una altra en forma de llum d'ambient per evitar ombres molt dures. En la següent imatge es pot observar la situació de les dues llums esmentades anteriorment i la direcció de la llum solar, simbolitzada amb una línia groga.



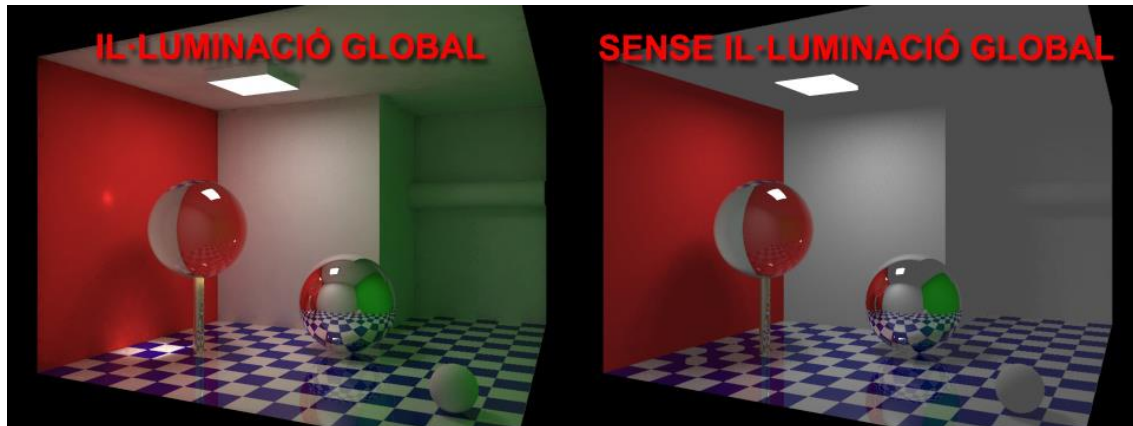
En un editor 3D com és el cas de Cinema 4D, el procés de *render* es un pèl diferent. A més a més de calcular com quedarien tots els fotogrames, també calcula la il·luminació i les ombres de cada fotograma.

També utilitza alguns algorismes i alguns mètodes per augmentar el realisme de l'escena. En comentarem dos, els anomenats il·luminació global i oclusió ambiental.

La il·luminació global o GI (de l'anglès *global illumination*) és un nom general per a un grup d'algorismes utilitzats en els gràfics per ordinador en 3D per afegir il·luminació més realista a les escenes 3D. Aquests algorismes tenen en compte no només la llum que prové directament d'una font de llum (il·luminació directa), sinó també els casos subsegüents en els quals els raigs de llum de la mateixa font es reflecteixen en altres superfícies en l'escena (il·luminació indirecta), ja sigui reflectant o no .

En les imatges següents es pot observar la diferència entre un *render* fet amb il·luminació global i un sense. La que no disposa d'il·luminació global només mostra les

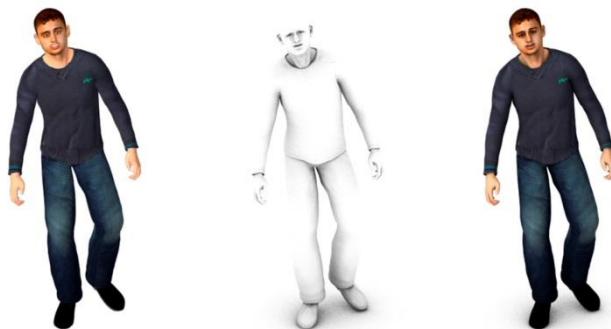
llums directes del focus de llum situat al sostre i les ombres resultats d'aquesta, mentre que la que disposa d'il·luminació global mostra els reflexos provocats pels rebots de llum sobre els objectes, els canvis de llum en rebotar sobre un objecte de color i les diferents ombres ocasionades per totes les llums.



L'oclusió ambiental o AO (de l'anglès *ambient occlusion*) s'intenta aproximar a la manera amb què la llum s'irradia a la realitat, sobretot fora del que normalment es consideren superfícies no reflectants.

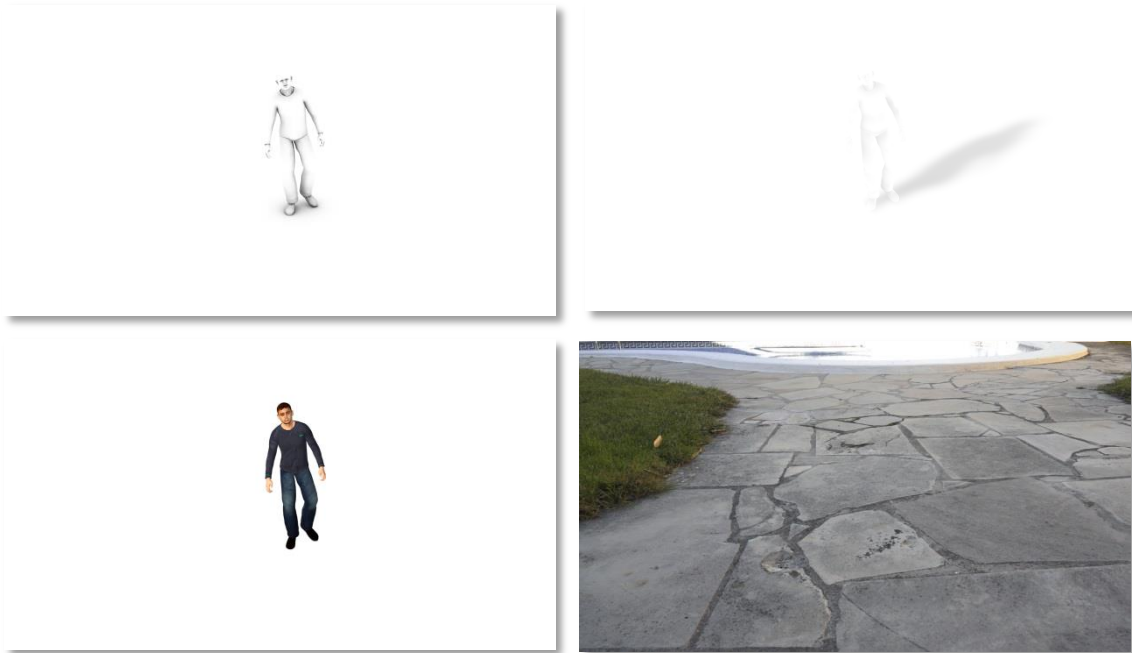
L'oclusió ambiental és un tipus d'ombreig que s'utilitza per donar realisme a l'escena. Oclusió ambiental és simplement una simulació de l'ombra causada per objectes que bloquegen la llum de l'ambient. A diferència d'altres tipus d'ombres, l'oclusió ambiental no depèn de la direcció de la llum, per la qual cosa pot ser precalculada per a objectes estàtics. L'aspecte suau aconseguit per oclusió ambiental per si sola és similar a la forma en què un objecte apareix en un dia ennuvolat.

En la imatge següent es mostra a l'esquerra el model sense l'oclusió ambiental, al mig només amb l'oclusió ambiental i a la dreta es mostren les dues primeres imatges fusionades per veure com queda el model amb l'oclusió ambiental aplicada.



Captura de Moviment

Per fer el *render* del nostre vídeo vam utilitzar l'oclusió ambiental i la il·luminació global. A l'hora d'exportar vam utilitzar el que s'anomena multipassada. Multipassada vol dir que en comptes d'exportar les ombres, l'oclusió ambiental... tot junt, fa diferents passades i exporta les diferents parts de la imatge en imatges diferents. Cinema 4D et deixa elegir quines parts vols exportar per separat i quines no. En aquest cas vam elegir un multipassada de quatre passades, una per a les ombres, una per a l'oclusió ambiental, una altra per al personatge i una altra per al fons. A continuació es mostren les diferents parts exportades:



6.8. Postproducció

Un cop tenim aquestes seqüències d'imatges de les diferents parts les hem de juntar totes en una sola imatge. El primer que farem és importar tots els arxius exportats dins l'After Effects. El fet d'haver utilitzat el *render* de multipassada dins el Cinema 4D ens permet que ara puguem modificar certs paràmetres com ara la quantitat d'ombra o l'oclusió ambiental de l'escena. El que farem amb l'After Effects serà fer coincidir les tonalitats de color, les ombres i l'oclusió ambiental del nostre personatge amb el color, les ombres i l'oclusió ambiental que hi ha en la imatge de fons. En el nostre cas hi havia un dia ennuvolat i les ombres no quedaven gaire marcades en el terra, per això reduïrem l'opacitat de les ombres i l'oclusió ambiental. En les següents imatges es poden veure unes mostres de com eren les ombres i l'oclusió ambiental al principi (imatge esquerra) i com queden al final, després de reduir les ombres i l'oclusió ambiental (imatge dreta).



A part d'això, si observem amb detall la imatge podem observar que les parts que estan generades per ordinador es veuen molt nítides i que les parts gravades per la càmera no se'n veuen tant. Això passa perquè el sensor de la càmera no és capaç d'oferir la realitat amb un cent per cent de precisió i el nostre editor 3D sí que és capaç de fer-ho. Per compensar aquesta diferència de precisió utilitzarem els efectes de desenfocament que ens ofereix l'After Effects. After Effects disposa de molts d'aquests efectes, però en aquest cas en especial utilitzarem un efecte anomenat *Desenfoque ràpid*. Anirem augmentant el grau de desenfocament del nostre personatge fins a obtenir un resultat que coincideixi amb l'escena. En les següents imatges es pot observar com en la primera el nostre personatge està molt nítid i el nostre fons no tant i en la segona el fons i el personatge tenen el mateix grau de nitidesa. Hem de tenir en

Captura de Moviment

compte que al passar el vídeo a fotografies pot ser que la diferència quedi menys ressaltada.



Un altre punt que hem de tenir en compte és el desenfocament per moviment. Cinema 4D té l'opció de generar aquest desenfocament en el moment del *render*. Ofereix uns resultats bastants similars als de la vida real però té l'inconvenient que augmenta molt el temps de *render*. Cada clip d'animació va tardar una mitjana d'entre quatre i cinc hores a fer *render* sense aquest desenfocament, amb aquest desenfocament aplicat amb el mateix ordinador el temps estimat de *render* era d'unes 26 hores per cada clip d'animació. Per evitar aquests temps extensos de *render* i aconseguir el desenfocament per moviment vam trobar un *plug-in* (una expansió del programa) per After Effects anomenat Reel Smart Motion Blur que ens permet crear el desenfocament per moviment amb acabats similars als que utilitza Cinema 4D però amb l'avantatge que el temps de *render* és inferior a 5 minuts per a cada clip. A continuació es pot observar la mateixa imatge, a l'esquerra sense el desenfocament per moviment i a la dreta amb el desenfocament per moviment amb el *plug-in* aplicat.



Captura de Moviment

En la primera escena en què surt el nostre personatge, a banda de fer el que hem comentat anteriorment, també vam fer una petita animació per fer l'aparició del personatge. El primer que vam fer va ser animar amb una màscara l'aparició del nostre personatge.



El següent pas que vam fer va ser aplicar un filtre que permet augmentar la mida i la quantitat dels píxels de la imatge o reduir-los al nostre gust. El que vam fer va ser que al principi la mida dels píxels fos molt gran i a mesura que anava apareixent anés disminuint la mida, de forma que semblés que s'estava generant el nostre personatge en aquell moment.



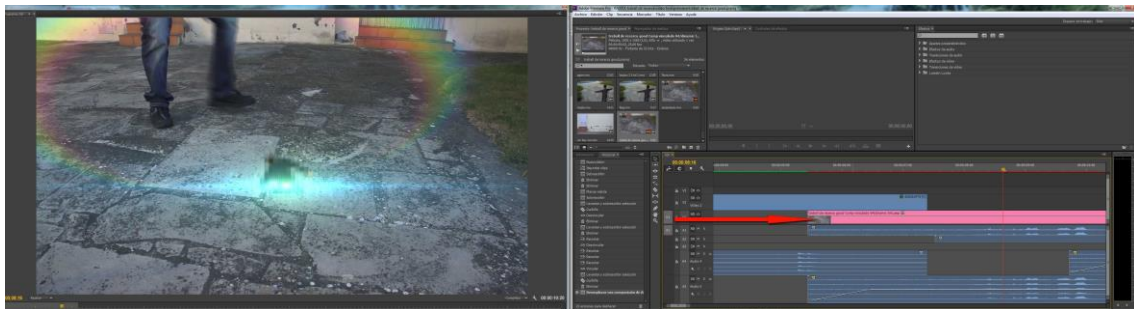
Captura de Moviment

També vam utilitzar un altre *plug-in* anomenat Optical Flares que serveix per simular els centelleigs que fa el sol a les òptiques de la càmera o per crear-ne d'abstractes. En la imatge de la dreta es mostra una imatge on el sol ha produït els centelleigs mencionats anteriorment. Vam agafar-ne un d'abstracte ja creat pel programa i vam animar-lo seguint el moviment de la màscara creada anteriorment per donar un toc de ciència-ficció a l'aparició del nostre personatge. En les següents imatges es pot observar l'animació final amb els centelleigs aplicats l'escena.

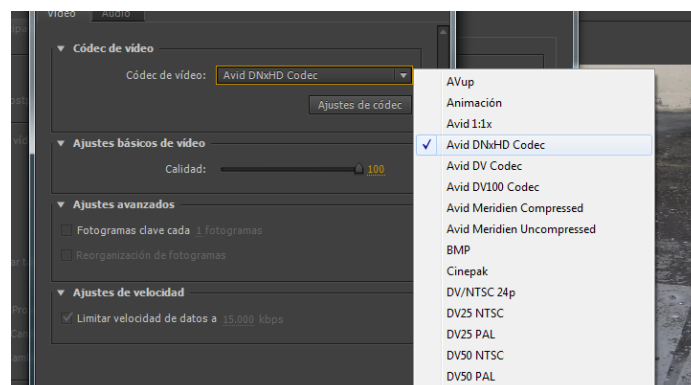


6.9. Edició final

Un cop tenim totes les escenes on surt el personatge ja podem tornar al nostre editor de vídeo, en aquest cas l'Adobe Premiere. Com que l'After Effects està fet pel mateix fabricant i que abans hem reemplaçat els nostres vídeos per composicions d'After Effects, ara no ens fa falta exportar un per un els nostres vídeos des de l'After Effects per importar-los a Premiere ja que ho fa automàticament. L'avantatge que ofereix aquesta opció és que si després canvies alguna cosa de l'animació feta desde l'After Effects automàticament està canviada a Premiere, l'inconvenient és que aquesta opció consumeix molts recursos de l'ordinador i, en el cas de no tenir un equip molt potent o tenir moltes animacions en un mateix vídeo, això pot causar un alentiment en el sistema. Si mirem la següent imatge del nostre editor de vídeo podem observar que els clips on hem inserit el nostre personatge, assenyalat amb una fletxa vermella, ja han estat importats automàticament a la línia de temps.



En el nostre cas tenim un ordinador de gamma mitjana però també tenim moltes escenes on hi ha les animacions fetes des de l'After Effects, per això ens vam decantar per l'opció d'exportar els clips un cop els tinguéssim fets i no tinguéssim intenció de canviar-los. A l'hora d'exportar-los vam elegir el còdec DNxHD per no consumir tants recursos de l'ordinador.



Captura de Moviment

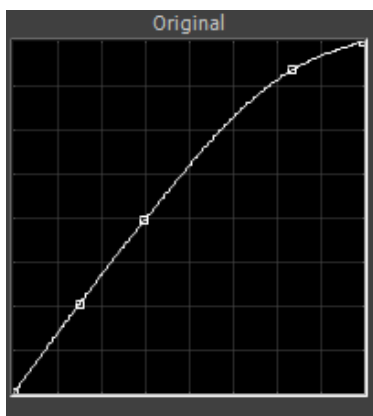
Un cop tenim tots els clips exportats només ens fa falta inserir-los a la línia de temps que havíem deixat aparcada al principi i arreglar la durada dels clips en cas que variï a l'inserir el nostre personatge. En aquesta imatge es pot observar la línia de temps ja acabada.



També cal arreglar els canvis de llum entre escena i escena perquè l'espectador es pensi que tot succeeix al mateix moment. Per fer-ho utilitzarem un filtre que ofereix el Premiere anomenat corbes i un *plug-in* anomenat Colorista II. El filtre anomenat *Curvas* serveix per enfosquir o il·luminar les imatges i el filtre Colorista II serveix per canviar les tonalitats de color de l'escena. En les següents imatges es pot observar com entre escena i escena la llum ha canviat (a dalt), i com després d'aplicar els filtres s'igualen les diferències de llum (a baix).



En les imatges següents es pot observar el funcionament dels filtres, a la imatge de l'esquerra es pot observar el filtre *Curvas* i a la imatge de la dreta el filtre *Colorista II*. El filtre corbes consta d'una línia dins una graella, l'extrem superior de la línia fa referència als colors més clars i l'extrem inferior als més foscos. En aquesta línia hi podem introduir punts i moure'ls al llarg de la graella. Si movem la línia en direcció al vèrtex superior esquerre de la graella augmentem la llum, si ho fem a l'inrevés la disminuïm. El filtre *Colorista II* consta de tres circumferències on has de moure el punt que cadascuna té al centre en direcció al color que vols a l'escena; la circumferència de l'esquerra representa els colors de les ombres, la circumferència del mig els tons mitjans i la de la dreta representa els colors més clars. En aquests dos casos el que hem fet ha sigut augmentar la llum de la segona imatge mitjançant el filtre anomenat corbes i després amb el filtre *Colorista II* hem augmentat el color groc en l'escena fins coincidir amb el groc de l'altra escena.



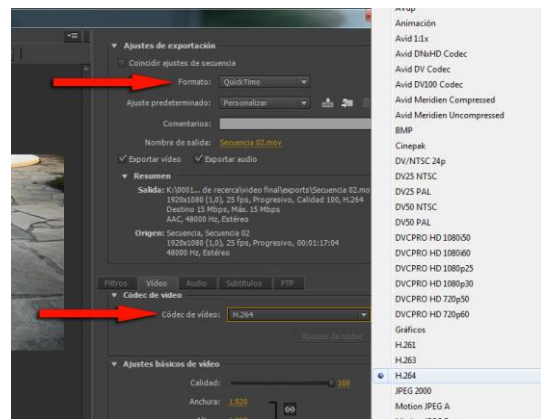
Un cop igualada tota l'escena, cal donar un toc cinematogràfic al nostre vídeo. Això s'aconsegueix mitjançant certes correccions de color. Per fer-ho utilitzarem dos *plugins*, un anomenat *Mojo* i un altre anomenat *Missfire Vignette*. El *Mojo* el que fa és donar detall en certes zones de la imatge i modificar el color per un de més fred per simular les càmeres que s'utilitzen en el cinema. El *Missfire Vignette* el que fa és enfosquir lleugerament els laterals de la imatge per simular un defecte que tenien les primeres òptiques de les càmeres cinematogràfiques. La gràcia d'aquesta correcció està en el fet que l'espectador no ho noti, per això cal no abusar d'aquests filtres per no carregar massa la imatge. En les següents imatges es pot apreciar el canvis que experimenta la imatge en aplicar els filtres, en la primera imatge es mostra el vídeo sense haver-hi aplicat cap filtre, en la segona es mostra el vídeo aplicant-hi el *Missfire Vignette* i es pot observar com ha enfosquit una mica els laterals, i en l'última imatge

Captura de Moviment

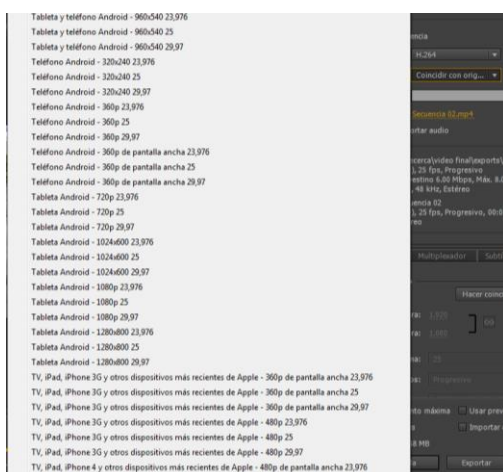
es pot observar el Missfire Vignette i el Mojo aplicats i es pot observar que el color de l'escena ha canviat i que el personatge té més detall.



Un cop tinguem el nostre vídeo editat i amb el color editat, ja estaria a punt per exportar-lo definitivament. A l'hora d'exportar cal tenir en compte per on s'ha de veure aquest vídeo, s'ha de reproduir en un DVD?, en un ordinador?, o en xarxes socials com YouTube, Vimeo o Facebook? Dependent del lloc de reproducció variarà el tipus de còdec i format del vídeo. En el nostre cas exportarem tres vídeos, un per a cada dispositiu. El primer l'exportarem per reproduir-lo en ordinadors i xarxes socials, per fer-ho utilitzarem el còdec H.264 amb qualsevol dels formats que suporta, els més comuns són el mov (Quicktime) o l'mp4. Tot i que Premiere ens deixa personalitzar el còdec H.264, en aquest cas el deixarem amb les opcions predeterminades. El segon que

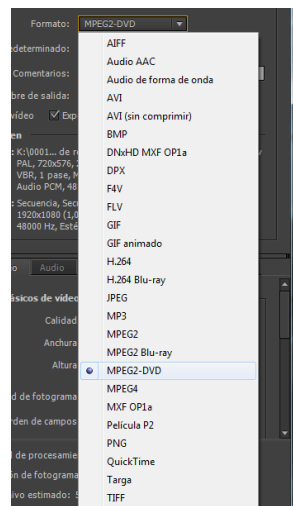


exportarem també serà amb H.264 però en aquest cas serà per utilitzar-ho en dispositius tipus *tablet*. Per fer-ho aprofitarem configuracions predeterminades d'H.264 que ens ofereix Premiere. Ens deixa elegir entre la majoria de *tablets* o *smartphones* del mercat, la resolució i la freqüència. Com que Premiere s'actualitza constantment, ràpidament es pot adaptar a qualsevol *tablet* o *smartphone* del mercat.



Captura de Moviment

L'última exportació que farem serà per poder reproduir el vídeo en un reproductor de DVD. Per fer-ho tornarem a utilitzar configuracions preestablertes a Premiere. En aquest cas no serà amb l'H.264, sinó amb un còdec anomenat MPEG-2. No entrarem en detall en el funcionament d'aquest còdec ja que és molt complex i és un còdec que actualment està perdent pes al mercat per culpa de l'H.264. Actualment aquest còdec només es fa servir en televisió, en reproductors de DVD o algun altre ús minoritari. També cal explicar que el còdec H.264 també és conegut com MPEG-4 ja que es considera que és una versió actualitzada de l'antic MPEG-2.



Un cop exportat el treball ja estaria la nostra part pràctica enllestida. El resultat d'aquesta part pràctica es pot trobar al final d'aquest document en format sòlid o en les següents webs:

<http://www.vimeo.com/alexsole/treballderecerca> (contrasenya: recerca)

http://youtu.be/_aLPCTZwgw

7. Conclusions

Pel que fa al primer objectiu, que era el d'explicar què és la captura de moviment, com funciona i quines possibles aplicacions té, podem concloure que hem investigat i explicat què és la captura de moviment, els seus usos i com es fa.

Pel que fa al segon objectiu, que era el de realitzar una captura de moviment 3D per animar un personatge generat per ordinador a través del dispositiu Kinect, podem concloure que hem aconseguit fer una captura de moviment amb resultats molt precisos i animar un personatge que prèviament havíem creat per ordinador. Com que treballem amb el format universal de captura de moviment, aquestes animacions servien per a la majoria de programes, ja siguin per a videojocs, cinema, finalitats mèdiques..., tot i que en aquest treball només l'haguem aplicat en els programes de cinema. El que fa més especial aquesta captura que hem aconseguit és que ho haguem fet amb dos dispositius que ens van costar 49,99 € cada un i que realment no estan dissenyats per fer això, sinó que estan dissenyats per fer-los servir de comandament. Deixant de banda el tema de captura de moviment, també hem pogut crear un personatge tridimensional i aplicar-hi textures UV mapping amb uns resultats molt bons.

Pel que fa a l'últim objectiu, que era el de desenvolupar una metodologia simplificada per aconseguir els mateixos resultats o similars als que s'obtenen als platós de captura de moviment, podem concloure que la metodologia que hem utilitzat en la part pràctica ha estat provada en diferents llocs amb diferents condicions lumíniques i amb diferents ordinadors i en tots els casos ha donat un resultat similar al mostrat en aquest treball. Per tant, podem dir que la nostra metodologia funciona en una àmplia varietat de condicions i plataformes. D'altra banda, mitjançant aquesta metodologia i aquests dispositius hem aconseguit realitzar un curtmetratge el qual es pot trobar al final d'aquest document en format sòlid o en les següents webs:

<http://www.vimeo.com/alexsole/treballderecerca> (contrasenya: recerca)

http://youtu.be/_aLPCTZwgw

Personalment, aquest treball m'ha servit per demostrar que no es requereix una gran quantitat de capital, eines o coneixements previs per realitzar una captura de moviment. Només fa falta tenir-ne ganes i ser una mica autodidacta per internet.

8. Agraïments

No voldria acabar aquest treball sense abans haver mostrat el meu agraïment a les persones que m'han donat suport d'una manera o altra. En primer lloc voldria agrair a l'Òscar Garcia per l'atenció i dedicació que m'ha ofert des del primer moment.

També voldria donar les gràcies al Josep Giribet, que es qui em va introduir en tot aquest món i m'ha ajudat sempre que li ha estat possible, al seu fill David Giribet, que em va ajudar a resoldre dubtes sobre el Cinema 4D, al Paco Almansa, que em va ajudar a buscar programari de modelatge, al professor Rubén de Diego, per la profitosa conversa que vam tenir aquest passat mes de desembre, al Joan Farran, per l'assessorament i correcció del text, a la Marina Rodríguez, per fer d'operari de càmera per al curtmètratge, i a l'Antoni Pont, que va transformar la veu del personatge generat per ordinador.

Finalment vull agrair als meus pares la paciència i l'ajuda oferta al llarg de la realització d'aquest treball.

9. Programari utilitzat

Adobe Creative Cloud (Adobe Premiere CC, Adobe After Effects CC, Adobe Photoshop CC)

Cinema 4D R14.014

Boujou v5.0.0 47534

Poser Pro 2012

IPi Recorder v2.2.1.25

IPi Mocap Studio v2.0.1.138

Microsoft Photosynth v1.6.1

10. Bibliografia

Comunicacions orals

Almansa, Paco (Departament Infografia 3D, Integrum). Comunicació personal. 22-7-13.

de Diego, Rubén (Departamento de Ingeniería y Arquitecturas Telemáticas, Universidad Politécnica de Madrid). Comunicació personal. 25-12-13.

Giribet, David (responsable de disseny videogràfic i multimèdia del Canal Super3, TVC). Comunicació personal. 12-6-13.

Giribet, Josep (realitzador multimèdia, CALIDOS.cat). Comunicació personal. 12-6-13.

Solé, Jaume (fotògraf i president de l'AFP Comarques de Lleida). Comunicació personal. 5-10-13.

Webs consultades

Adobe Systems Incorporated. *CC tutorials*. [Consultat: 7-10-13]. Disponible a Internet: <http://tv.adobe.com/channel/how-to/creative-cloud-tutorials/>.

Allied Vision Technologies. *Allied Vision Technologies*. [Consultat: 24-7-13]. Disponible a Internet: <http://www.alliedvisiontec.com/apac/products/application-markets/application-case-studies/application-case-study/article/motion-capture-with-avt-cameras-makes-orthopedic-diagnosis-easier.html>.

Gentilini, Mike. *Aerial Video, 4K, Visual Effects, Adobe Tutorials, VFX, VidMuze*. [Consultat: 7-10-13]. Disponible a Internet: <http://www.vidmuze.com/>.

Gorski, Sam; Pueringer, Niko. *Corridor Digital*. Collective. [Consultat: 3-6-13]. Disponible a Internet: <http://www.youtube.com/corridordigital>.

Gorski, Sam; Pueringer, Niko. *samandniko*. Collective. [Consultat: 3-6-13]. Disponible a Internet: <http://www.youtube.com/samandniko>.

iPi Soft LLC *Motion Capture for the Masses..* [Consultat: 25-8-13]. Disponible a Internet: <http://ipisoft.com/>.

iPi Soft LLC. *iPiSoft Wiki*. [Consultat: 25-8-13]. Disponible a Internet: <http://wiki.ipisoft.com/>.

Kramer, Andrew. *VIDEO COPILOT / After Effects Tutorials, Plug-ins and Stoc Footage for Post Production Professionals*. Video Copilot and Final Image Inc. [Consultat: 3-6-13]. Disponible a Internet: <http://www.videocopilot.net/>.

Lasenby, Joan. *Medical applications of optical motion capture*. Cambridge University Engineering Department. [Consultat: 12-7-13]. Disponible a Internet: [http://www-g.eng.cam.ac.uk/mmg/lifesciences/lasenby.html](http://www.g.eng.cam.ac.uk/mmg/lifesciences/lasenby.html).

Maxon. *CINEWARE by MAXON*. [Consultat: 19-10-13]. Disponible a Internet: <http://landingpage.maxon.net/?lang=es>.

Organic Motion, Inc. *Markerless Motion Capture System*. [Consultat: 12-8-13]. Disponible a Internet: <http://www.organicmotion.com/>.

Quantic Dream. *Quantic Dream*. [Consultat: 24-7-13]. Disponible a Internet: <http://www.quanticroam.com/en>.

Red Giant LLC. *Red Giant - Filmmaking & Motion Graphics Software*. [Consultat: 25-8-13]. Disponible a Internet: <http://www.redgiant.com/>.

Red.com, Inc. *WELCOME TO RED DIGITAL CINEMA*. [Consultat: 23-7-13]. Disponible a Internet: <http://www.red.com/>.

Wikipedia Foundation, Inc. *3D computer graphics*. [Consultat: 12-7-13]. Disponible a Internet: https://en.wikipedia.org/wiki/3D_computer_graphics.

Wikipedia Foundation, Inc. *Motion Capture*. [Consultat: 23-7-13]. Disponible a Internet: https://en.wikipedia.org/wiki/Motion_capture.