

# Sistema Estereoscòpic de Baix Cost

Víctor Blanch Vicens, Antoni Jaume-i-Capó

*Enginyeria en Informàtica*

**Resum**— El present treball mostra el procés d'investigació portat a terme per tal de desenvolupar un sistema binocular de baix cost capaç de realitzar captura estereoscòpica. S'analitzen els requeriments necessaris per a portar a terme el projecte, així com el procés de disseny i construcció d'un prototipus, tant en les vessants del software com del hardware. A continuació s'introdueix el concepte de correspondència estèreo i la seva aplicació en forma d'algoritmes dins l'àmbit del dispositiu. Finalment, es mostren i s'analitzen els resultats obtinguts.

## I. INTRODUCCIÓ

En el mercat existeixen dispositius capaços de realitzar captura estereoscòpica. Tot i això, aquests acostumen a tenir un preu molt elevat. Aquest factor dificulta l'experimentació i utilització d'aquesta mena de sistemes. El principal objectiu del present projecte és donar resposta a aquest problema, proposant una solució basada en el desenvolupament d'un dispositiu de baix cost fent servir webcams, capaç de portar a terme la captura d'imatges i el seu processament posterior. Es fa servir com a sistema de referència el model BumbleBee 2 de Point Grey Research [1], mostrat a la figura 1.



Fig. 1 Sistema BumbleBee de Point Grey Research.

Els principals objectius del projecte són:

- Identificar dispositius de captura de baix cost.
- Anàlisi, disseny i construcció del prototipus.
- Transformació de coordenades de 2D a 3D i a la inversa.
- Estudi i implementació d'algoritmes de correspondència estèreo.
- Desenvolupament d'una API d'estereovisió per operar el sistema.
- Desenvolupament d'un programa de prova.

## II. ARQUITECTURA DEL SISTEMA

Resulta necessari definir clarament quines són les característiques desitjables del software i del hardware que s'ha de desenvolupar per al projecte. Per tal de fer-ho, es

defineixen els requeriments que ha de complir el sistema. Un cop fet això, és possible adquirir els elements necessaris per a la construcció del prototipus, i l'inici del desenvolupament del software associat.

### A. Requeriments del Sistema

Les imatges capturades han de tenir una certa qualitat. Per això es defineix una grandària d'imatge de com a mínim 320x240 píxels. La velocitat de captura mínima és de 30 quadres per segon, ja que segons Boulic et al. [2], aquest és el mínim a un sistema d'estereovisió en temps real per tal d'oferir resultats prou bons. La codificació de les imatges ha de ser RGB, per tal de fer-les més fàcils manipular.

Com que es tracta d'un dispositiu de baix cost, el preu dels components és un factor important a tenir en compte. S'ha de vigilar especialment la relació qualitat/preu dels dispositius de captura adquirits. Tenint en compte que el sistema de referència BumbleBee2 té un preu aproximat de 1212€, es fixa un pressupost màxim per al projecte de 200€, el qual ha d'ésser suficient per a portar-lo a terme.

### B. Sincronització

La sincronització d'un sistema estereoscòpic consisteix en la captura de les dues imatges amb la distància temporal més petita possible. D'aquest temps dependrà la qualitat dels resultats obtinguts posteriorment. Existeixen diverses opcions a tenir en compte a l'hora de realitzar-la.

1) *Sincronització per Software*: La sincronització basada en software consisteix en la captura de dues imatges tan properes en el temps com sigui possible [3]. La distància temporal entre imatges és de  $1/f$  segons, essent  $f$  la quantitat de quadres per segon a la qual és capaç d'arribar el dispositiu de captura en qüestió. Destaca per la seva senzillesa, ja que no requereix hardware addicional, però la seva qualitat depèn de la velocitat de captura de les càmeres utilitzades.

2) *Sincronització per Hardware*: Existeixen diverses alternatives dins aquest àmbit:

- Sincronització per controladora USB: consistiria en modificar una controladora USB amb la intenció de manipular el moment de captura de les dues imatges.
- Genlock: Sistema implementat en diferents dispositius de captura comercials, basats en la interconnexió dels dos dispositius mitjançant un cable, que envia un senyal de sincronisme d'un a l'altre, com indica Meijer [4].
- Interconnexió de càmeres: Fent servir els esquemes de la circuiteria interna dels dispositius de captura, es tracta

d'enviar els senyals de sincronisme des d'un d'ells a l'altre, amb la finalitat d'anul·lar els senyals propis d'aquest i substituir-los pels entrants.

### C. Disseny de l'Arquitectura

Els principals problemes a resoldre en quant a l'arquitectura del sistema són dos: la selecció de les càmeres a fer servir i la selecció del tipus de sincronització entre elles.

En quant a les càmeres, després de fer un estudi de mercat, s'arriba a la conclusió que les webcams són dispositius adients per al projecte, pel seu baix cost i la qualitat que han arribat a assolir en els darrers anys. Dins del mercat d'aquest tipus de càmeres, Philips [5] és la marca que fabrica els millors dispositius de captura per al projecte.

D'entre els models que fabrica la marca, es fa recomanable adquirir el model SPC900NC per diverses raons. És capaç d'arribar als 90 quadres per segon, i a més hi ha molta informació a Internet sobre modificacions, circuits i òptica del model, cosa que facilita en part la manipulació i identificació del seu hardware. Gràcies a això, es coneix el circuit integrat intern, i es poden trobar explicacions detallades sobre maneres de modificar el seu sincronisme.

S'ha de prendre una decisió respecte de l'alternativa a triar per tal de portar a terme el sincronisme entre les captures d'imatge. Amb aquest objectiu, s'han estudiat les possibilitats que ofereix el dispositiu, i analitzat detalladament els avantatges i desavantatges de cadascuna de les opcions possibles per tal d'escollir la millor.

La sincronització per hardware és sempre preferible, ja que no requereix intervenció del processador del sistema base, i s'executa sobre el propi hardware de la càmera. Per tant és molt més eficient i ràpida que els mètodes basats en software, ja que la distància temporal entre les imatges resulta molt petita. Tot i aquests avantatges, es descarta aquesta mena de sincronització per diverses raons. Per una banda, tant el mètode de sincronització mitjançant una controladora USB, com el d'interconnexió de les dues càmeres per mitjà dels seus circuits interns, resulten arriscats, i són necessaris coneixements avançats d'electrònica per portar-los a terme. A més les càmeres només permeten l'entrada d'un senyal de sincronització vertical, que indica quan s'ha d'actualitzar la imatge del *buffer*, però no horitzontal, que té la funció d'indicar quan s'ha de capturar una nova línia d'imatge. La falta d'aquest senyal produeix nombroses distorsions. Finalment, donat que les càmeres no disposen de sistema Genlock, tampoc resulta possible sincronitzar amb aquesta tècnica.

Donades les limitacions exposades, s'opta per un model de sincronització per software, mostrat a la figura 2, i basat en la captura de les dues imatges més properes en el temps. Fent servir aquest mètode, en el millor dels casos s'obté una distància temporal entre imatges de 1/90 quadres per segon, és a dir, 0.011 segons. Suposant que les càmeres superen els 30 quadres per segon com a mínim, la distància temporal màxima és de 1/30 segons, és a dir, 0.033 segons, el qual resulta acceptable segons els requeriments de qualitat fixats per al projecte.

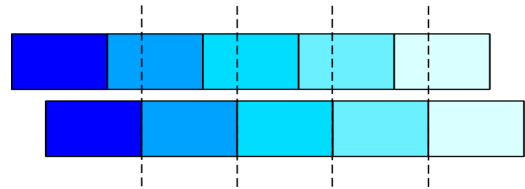


Fig. 2 Il·lustració de la sincronització de captures basada en software.

## III. SISTEMA BINOCULAR

Un cop definida l'arquitectura, es passa a la següent fase de desenvolupament del projecte. Aquesta etapa engloba tant el procés de construcció del sistema, com l'especificació del funcionament de les operacions bàsiques que haurà de portar a terme el prototipus mitjançant una llibreria, desenvolupada especialment per a tasques d'estereovisió.

### A. Construcció

El sistema posseeix una base sobre la qual se situen les dues càmeres. La distància entre els dos dispositius és de 120 mm, similar a l'existent en el sistema BumbleBee2, la qual permet una apreciació de la profunditat similar a la que té un ésser humà. Per altra banda es decideix que la base sigui de 170 mm, suficient per a sostenir-les separades per la distància descrita anteriorment. Aquesta configuració es pot apreciar a la figura 3, i el resultat del procés de construcció es pot observar a la figura 4.

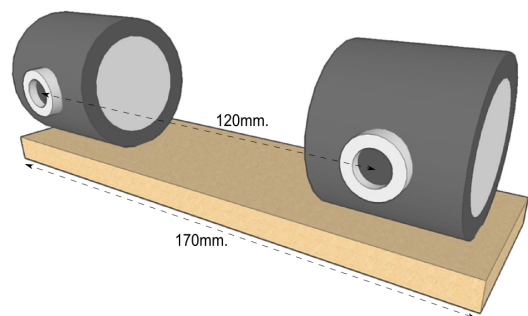


Fig. 3 Disseny en 3D del prototipus on es mostren les distàncies proposades.



Fig. 4 Resultat final del procés de construcció.

### B. Captura d'Imatge

En primer lloc és necessari un petit calibratge del sistema, comprovant que les distàncies són correctes i els eixos òptics

es troben completament alineats i en paral·lel. D'aquesta manera es facilita el processament estereoscòpic de les imatges, així com les equacions matemàtiques subjacents que fan possible l'obtenció de bons resultats.

La figura 5 mostra com per tal d'accedir a les imatges capturades per les webcams, es fa servir la llibreria OpenCV [6]. A més, sobre aquesta es construeix una llibreria de programació, anomenada SVAPI (*Stereo Vision Application Programming Interface*), que fa d'interfície entre les aplicacions i OpenCV, facilitant la programació i el processament de les imatges.

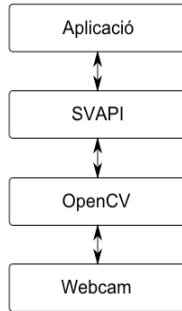


Fig. 5 Etapes del procés de captura.

### C. Projectió i Reconstrucció de Coordenades del Món

El model de càmera *pinhole* que segueixen les càmeres es basa en el pas de la llum a través d'un petit orifici, i defineix matemàticament la projecció d'un punt del món sobre la imatge. La imatge projectada obtinguda apareix girada verticalment, i quan més petit és el forat o *pinhole*, major resulta la seva definició. Per tal de poder transformar coordenades entre imatge i món, cal conèixer els seus fonaments matemàtics.

Consideri's la configuració òptica de la figura 6, representant el sistema binocular que segueix el model estàndard descrit per Iocchi [7]. Es pot apreciar com un punt P del món es projecta sobre el pla de projecció, el qual es troba separat de la lent per una distància anomenada focal. Suposant els eixos òptics en paral·lel, s'aprecia com la coordenada  $x$  de la projecció és la distància entre el punt projectat i el centre de la lent de la càmera corresponent. Aquest resultat es pot extrapolar a la coordenada  $y$  de la projecció.

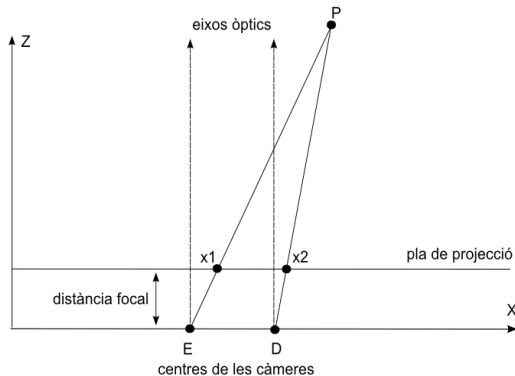


Fig. 6 Projeccions d'un punt sobre el pla de projecció del dispositiu binocular.

Per tal de transformar les coordenades, és necessari aplicar les fórmules del model *pinhole*. Les restriccions del sistema binocular, amb els eixos òptics paral·lels i les càmeres alineades, simplifiquen molt les equacions necessàries, i a més fan que el component a l'eix Y de les dues projeccions 2D resulti ser el mateix. Aleshores, la transformació de coordenades imatge-món s'expressa així:

$$\begin{aligned} zw &= (b * f) / (x1 - x2) \\ xw &= x1 * zw / f \\ yw &= y * zw / f \end{aligned} \quad (1)$$

essent  $(xw, yw, zw)$  les coordenades del punt reconstruït en 3D,  $b$  la distància entre els centres de les càmeres,  $f$  la distància focal, i  $(x1, y)$ ,  $(x2, y)$  les coordenades de les projeccions de P sobre els plans de projecció de les càmeres.

Conegudes aquestes equacions, realitzar el pas invers resulta senzill. Es tracta d'aïllar els termes que interessin de les equacions anteriors, de manera que la transformació de coordenades món-imatge es pot expressar com:

$$\begin{aligned} y &= yw * f / zw \\ x1 &= xw * f / zw \\ x2 &= x1 - (b * f / zw) \end{aligned} \quad (2)$$

on les diferents variables tenen el mateix significat que en l'equació 1.

## IV. CORRESPONDÈNCIA

Amb el sistema binocular muntat i correctament configurat, és el moment de passar a realitzar la captura de les imatges estereoscòpiques i portar a terme l'anàlisi d'aquestes, per així extreure'n tota la informació 3D de l'escena que sigui possible mitjançant la llibreria desenvolupada per al projecte.

### A. Correspondència Estèreo

La correspondència estèreo consisteix en comparar dues imatges amb l'objectiu d'obtenir un mapa de disparitat, que indica la distància entre un punt d'una imatge i el seu homòleg a l'altra. Aquesta representació se sol fer mitjançant una imatge en escala de grisos, de manera que el nivell de disparitat de cada píxel està representat pel seu nivell de gris. Una disparitat major suposa un color més clar, mentre que una menor disparitat representa el contrari. Per tal de generar-la, es duu a terme un procés de *matching* o comparació, que consisteix en cercar, per a cada píxel d'una de les imatges, el píxel corresponent a l'altra.

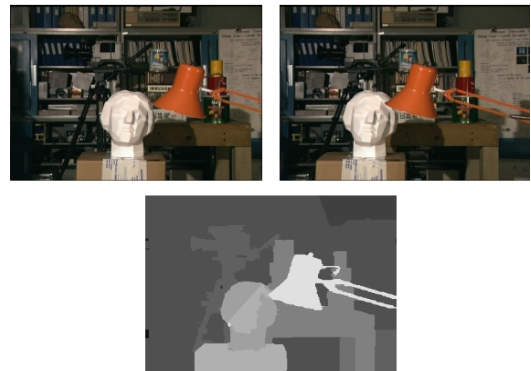


Fig. 7 Exemple de parell estèreo i resultat mitjançant *matching*.

Aplicada a un parell d'imatges estereoscòpiques, la correspondència indica la distància dels objectes de l'escena respecte del sistema de captura. La figura 7 mostra com a partir d'un parell d'imatges estereoscòpiques i el corresponent processament, es pot obtenir aquest mapa, on els objectes més propers apareixen en un color més clar que els llunyans.

Existeixen diversos enfocaments de la disparitat. En aquest projecte s'utilitza el de la correspondència densa, basat en comparar àrees de píxels, contraposat al de correspondència dispersa, basat en comparar contorns o altres característiques distintives de l'escena.

### B. Algoritmes Utilitzats

Es fan servir tres algoritmes de disparitat en el present projecte. Aquests algoritmes són:

3) *Birchfield-Tomasi*: Aquest algoritme [8] es troba implementat a OpenCV, la qual cosa facilita la seva utilització. Es basa en realitzar comparacions entre *scanlines* de les dues imatges dins la mateixa línia epipolar. Fa servir programació dinàmica, i es tracta d'un algoritme molt ràpid del qual s'obtenen bons resultats, tot i que produeix cert renou horitzontal.

4) *Finestra amb SAD*: L'algoritme de finestra compara regions rectangulars de píxels, anomenades finestres, d'una imatge per a cada píxel de l'altra, tal i com descriu Kuhl [9]. Per realitzar aquesta comparació es fa servir el criteri SAD (*Sum of Absolute Differences*), consistent en la suma de les diferències absolutes entre dues finestres de píxels:

$$SAD(f, g)_{i,j,w,h,d} = \sum_{\substack{\forall k \in [i-w/2, i+w/2], \\ l \in [j-h/2, j+h/2]}} |f(k, l) - g(k+d, l)| \quad (3)$$

on  $(i, j)$  són les coordenades del píxel de referència de la finestra als eixos X i Y,  $f$  i  $g$  són la imatge esquerra i dreta, respectivament,  $d$  és el desplaçament dins l'eix X de la finestra dreta respecte de l'esquerra. En darrer terme,  $w$  i  $h$  són l'amplada i l'alçada de la finestra, respectivament. Resulta ràpid i dona bons resultats amb imatges riques en textures, però s'obtenen mals resultats a les regions molt uniformes.

5) *Rectangular Subregioning amb CC*: Es basa en els treballs de Changming Sun [10]. Consisteix en dividir la imatge en subregions, de manera que es calcula la correspondència per a cadascuna d'elles. El criteri per a realitzar el procés de *matching* és la correlació creuada (*Cross Correlation*) expressada com:

$$C(i, j, d) = \frac{cov_{i,j,d}(f, g)}{\sqrt{var_{i,j,d}(f)} \cdot \sqrt{var_{i,j,d}(g)}} \quad (4)$$

on  $f$  i  $g$  són les imatges a tractar,  $(i, j)$  les coordenades del centre de la finestra o subregió,  $d$  el desplaçament en píxels de la finestra a comparar respecte de la coordenada  $i$ . Per altra banda,  $cov_{i,j,d}(f, g)$  és la covariància existent entre les dues finestres,  $var_{i,j,d}$  la variància existent entre els píxels

dins la finestra de coordenades  $(i, j)$ , i essent  $var_{i,j,d}$  la variància de la finestra situada a  $(i+d, j)$ . Aquest algoritme estalvia comparacions, però el procés de subdivisió resulta costós des del punt de vista computacional.

### C. Comparativa

Es volen comparar els algoritmes de correspondència per tal de determinar quin d'ells resulta millor per a un sistema de temps real. Per fer-ho, es fa necessari definir un entorn de proves. En aquest cas, es tracta d'un PC de sobretaula amb processador Pentium IV a 2.4 GHz, i el parell d'imatges estereoscòpiques "Tsukuba", molt utilitzat en estereovisió per a la realització de comparacions entre algoritmes.

Comparar els resultats dels diferents algoritmes, així com els seus temps d'execució, resulta molt difícil donada la diferent manera de calcular la disparitat i la dificultat per trobar un criteri que determini la correcció del resultat. A mode demostratiu, la taula I mostra els millors i pitjors resultats obtinguts per cadascun dels algoritmes estudiats en el transcurs de les proves realitzades dins aquest treball.

TAULA I  
COMPARATIVA ENTRE ALGORITMES

Algoritme	Valor mínim	Valor màxim
Birchfield-Tomasi	137,5 ms	273,5 ms
Finestra + SAD	17,1 ms	20079,6 ms
RSR + CC	1178 ms	6308 ms

Un cop realitzada la comparació dels algoritmes variant diversos paràmetres, es conclou que els algoritmes de Birchfield-Tomasi i el de finestra amb SAD són els que tenen temps d'execució més adequats per a un sistema de temps real, essent capaços de calcular diversos mapes de disparitat per segon. Per tant, es fan servir aquests per a proves posteriors, quedant descartat l'algoritme RSR amb CC pel seu elevat cost.

## V. RESULTATS

La darrera etapa a portar a terme en el desenvolupament del sistema estereoscòpic consisteix en la realització d'un conjunt de proves que permetin avaluar els resultats que es poden obtenir a partir del prototipus construït. Això implica analitzar totes les tasques que el sistema ha d'ésser capaç de realitzar. És necessari crear un programa específic capaç de comprovar tots aquests aspectes, i a més plantejar diversos jocs de prova que permetin obtenir, analitzar i comparar els resultats.

### A. Aplicació DemoEstereo

Amb la finalitat de supervisar el funcionament del sistema binocular i de l'API desenvolupada per a realitzar el processament estereoscòpic, es crea una aplicació anomenada DemoEstereo, capaç de portar a terme tant la captura com el càlcul de la disparitat, així com de transformar punts del món a coordenades del dispositiu i viceversa. L'aplicació reconstrueix l'escena en 3D, facilitant a l'usuari l'observació de la correspondència estereo.



El programa, mostrat a la figura 8, es troba escrit en el llenguatge C++, i fa servir la llibreria GTK+ per a implementar la interfície d'usuari. A més, l'aplicació utilitza la llibreria OpenGL per a portar a terme la reconstrucció 3D de l'escena processada. El codi font de l'aplicació està, com l'API, sota la llicència GPL, el qual permet la seva lliure distribució. L'entorn de programació utilitzat per a desenvolupar l'aplicació és Dev-C++, i es porta a terme sota el sistema operatiu Windows XP.

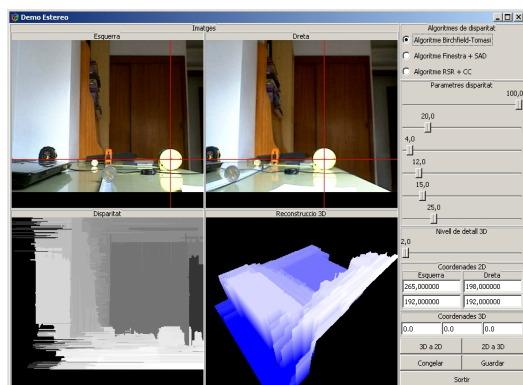


Fig. 8 Captura de pantalla de l'aplicació DemoEstereo.

## B. Jocs de Prova

Amb la finalitat de provar la capacitat del sistema per a reconstruir coordenades 3D i calcular la correspondència estèreo, es munta una escena senzilla sense elements mòbils. Dins aquesta escena, les posicions respectives de diferents punts són mesurades prèviament. Així es podrà comprovar fins a quin punt és correcta la distància calculada pel sistema.



Fig. 9 Escena estereoscòpica utilitzada per a la prova estàtica.

A l'escena es mostra una habitació i una taula, sobre la qual es troben situats diversos objectes col·locats a profunditats diferents. Per facilitar el càlcul de les distàncies, tres dels elements són cintes mètriques que indiquen amb precisió les distàncies existents respecte de la càmera i entre elles mateixes. A més, també hi ha altres elements, com diversos mobles o dues monedes, i s'hi troben objectes llunyans com diferents armaris i portes, que permeten avaluar el rendiment de l'aplicació a distàncies més grans. La figura 9 mostra el parell estèreo utilitzat, així com els diferents objectes dels quals s'han mesurat les distàncies, assignant un número identificable a cadascun d'ells.

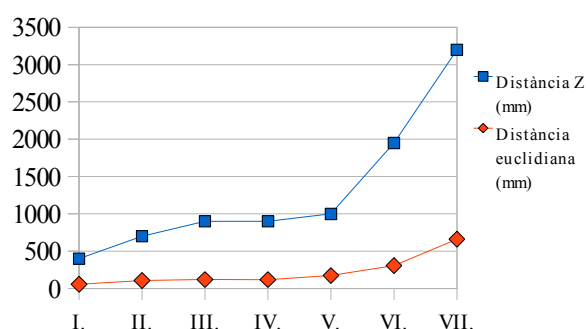


Fig. 10 Resultats de la prova de reconstrucció de coordenades.

La figura 10 mostra una gràfica on es compara la profunditat del punt amb la distància euclidiana existent entre les coordenades calculades (reconstruïdes) i reals. Als resultats obtinguts s'observa un increment de l'error a mida que els punts són més llunyans, indicant que el càlcul a distàncies grans resulta menys precís. Això passa també amb sistemes de visió biològics, com el dels éssers humans, essent més difícil apreciar la profunditat d'objectes allunyats.

Un cop feta la prova de la reconstrucció de coordenades, es passa a provar els algorismes de disparitat. En primer lloc, es fa servir l'algorisme de Birchfield i Tomasi, fent servir aquells paràmetres capaços de realitzar el càlcul de la correspondència 4 cops per segon. Els resultats obtinguts per aquest algorisme es mostren a la figura 11.

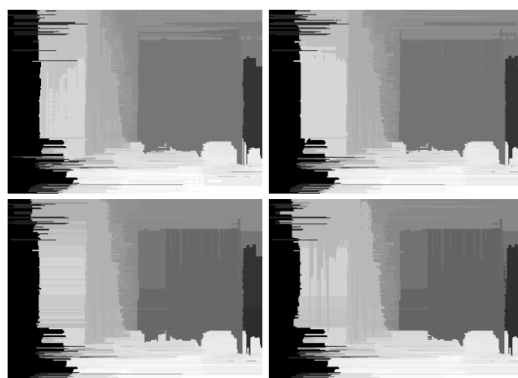


Fig. 11 Resultats de l'algorisme de Birchfield i Tomasi.

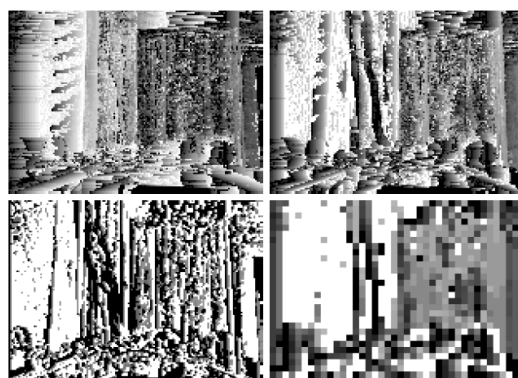


Fig. 12 Resultats de l'algorisme de finestra amb SAD.

A continuació, és necessari posar a prova l'altre algorisme de correspondència candidat. Els resultats obtinguts a partir de l'algorisme de finestra amb SAD es mostren a la figura 12, fent servir també els paràmetres per a aquest algorisme capaços de mostrar 4 mapes de disparitat per segon.

Un cop comparats els resultats obtinguts amb els dos algorismes, es considera millor aquell que, globalment, s'aproxima més a la realitat. Així doncs, el mapa de disparitat obtingut mitjançant l'algorisme de Birchfield i Tomasi presenta una major continuïtat en les àrees de gris calculades i un menor nivell de renou, és a dir, menys errors de càlcul expressats en forma de punts i zones de color blanc o negre. Aquest és, doncs, més recomanable a l'hora de calcular la correspondència estèreo mitjançant el dispositiu, i per tant es descarta l'algorisme de finestra amb SAD per a aquesta tasca.

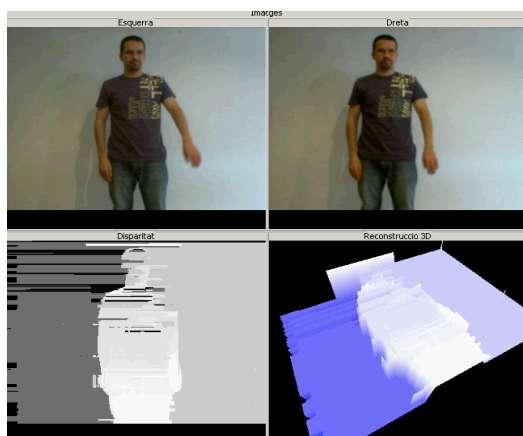


Fig. 13 Imatge de la prova amb una escena en moviment.

A continuació es duen a terme proves mitjançant una escena en moviment. L'algorisme utilitzat és el de Birchfield-Tomasi, fent servir els paràmetres que, d'entre aquells capaços de calcular 4 mapes de disparitat per segon, han donat mapes més precisos. Els resultats es mostren a la figura 13.

Dins d'aquest joc de prova es poden observar els mateixos problemes en els mapes de disparitat que amb els obtinguts mitjançant les proves amb imatges estàtiques. Es produeix renou vertical, però sobre tot horitzontal en forma de segments que no es corresponen amb la disparitat real, distorsionant el mapa i també la reconstrucció 3D. Es tracta d'un error típic de l'algorisme de Birchfield i Tomasi, que pot ésser compensat mitjançant el post-processament del mapa de correspondència, tenint el compte el cost d'aquesta operació.

## VI. CONCLUSIONS

En finalitzar aquest projecte s'ha aconseguit desenvolupar un prototipus operatiu de sistema estereoscòpic capaç de realitzar la captura d'imatges, trobar la correspondència, reconstruir l'escena en 3D i fer la transformació de coordenades imatge-món i viceversa. A més, l'API d'estereovisió desenvolupada per a realitzar aquestes operacions és independent dels dispositius de captura utilitzats, i per tant resulta útil en sistemes binoculars diferents al mostrat.

Les principals dificultats trobades durant el projecte han estat la tria del model de càmera a fer servir i les limitacions del software utilitzat per fer la captura, poc preparat per realitzar captures simultànies amb dos dispositius.

Aquest treball obre possibilitats per a nous projectes d'investigació, principalment en dues vessants:

- Experimentació amb millors models de càmera, com el recentment aparegut Minoru de Novo, cosa que permetrà millorar la qualitat dels resultats mitjançant una millor sincronització i una major velocitat de captura.
- Millores en el software emprat pel sistema, especialment dels algorismes de correspondència i de la llibreria OpenCV, així com l'ampliació i optimització de l'API desenvolupada. Això permetrà obtenir mapes de correspondència i transformacions de coordenades de major precisió.

## REFERÈNCIES

- [1] (2008) Point Grey Research Inc. [Online]. Disponible: <http://www.ptgrey.com>.
- [2] R. Boulic, J. Varona, L. Unzueta, M. Peinado, A. Suescun, F. Perales. Evaluation of on-line analytic and numeric inverse kinematics approaches driven by partial vision input, *Virtual Reality*, 2006.
- [3] S. M. Szymczyk (2006). *Webcam Synchronization* [Online]. Disponible: <http://cogville.cog-lab.com>.
- [4] P. Meijer (1996). *Stereoscopic vision for the blind* [Online]. High Tech Campus, E.A. Eindhoven. Disponible: <http://www.seeingwithsound.com/binocular.htm>.
- [5] (2008) Royal Philips [Online]. Disponible: <http://www.philips.com>.
- [6] (2008) Intel, *OpenCV: Open Source Computer Vision Library* [Online]. Disponible: <http://www.intel.com/technology/computing/opencv/>.
- [7] L. Iocchi (1998). *Stereo Vision: Triangulation* [Online]. Università di Roma "La Sapienza". Disponible: <http://www.dis.uniroma1.it/~iocchi/stereo/triang.html>.
- [8] S. Birchfield, C. Tomasi (1999). *Depth Discontinuities by Pixel-to-Pixel Stereo*. Department of Computer Science, Stanford University. Disponible: [http://www.ces.clemson.edu/~stb/publications/p2p\\_ijcv1999.pdf](http://www.ces.clemson.edu/~stb/publications/p2p_ijcv1999.pdf).
- [9] A. Kuhl (2005). *Comparison of Stereo Matching Algorithms for Mobile Robots*. The University of Western Australia Faculty of Engineering, Computing and Mathematics. Disponible: <http://robotics.ee.uwa.edu.au/theses/2005-Stereo-Kuhl.pdf>.
- [10] C. Sun (2002). *Fast Stereo Matching Using Rectangular Subregioning and 3D Maximum-Surface Techniques*. CSIRO Mathematical and Information Sciences. Disponible: <http://extra.cmis.csiro.au/IA/changs/docsun02ijcv.pdf>.



**Víctor Blanch Vicens** (Palma de Mallorca, 1981). [victor.blanch.v@gmail.com](mailto:victor.blanch.v@gmail.com). Enginyer en Informàtica per la UIB, actualment treballa com a tècnic a la Unitat de Gràfics i Visió per Ordinador, i Intel·ligència Artificial de la mateixa universitat.



**Antoni Jaume i Capó** (Santa Maria del Camí, 1979). [antoni.jaume@uib.es](mailto:antoni.jaume@uib.es). Enginyer en Informàtica i professor del Departament de Ciències Matemàtiques i Informàtica a la Universitat de les Illes Balears. Realitza la seva tasca de recerca a la Unitat de Gràfics i Visió per Ordinador, i Intel·ligència Artificial.