

FACULTATEA ELECTRONICA,
TELECOMUNICATII SI TEHNOLOGIA
INFORMATIEI
MASTER INGINERIA INFORMATIEI SI A
SISTEMELOR DE CALCUL

Driver pentru dispozitivele de intrare-iesire
3D

Coordonator:
Conf. dr. ing. *Stefan STANCESCU*

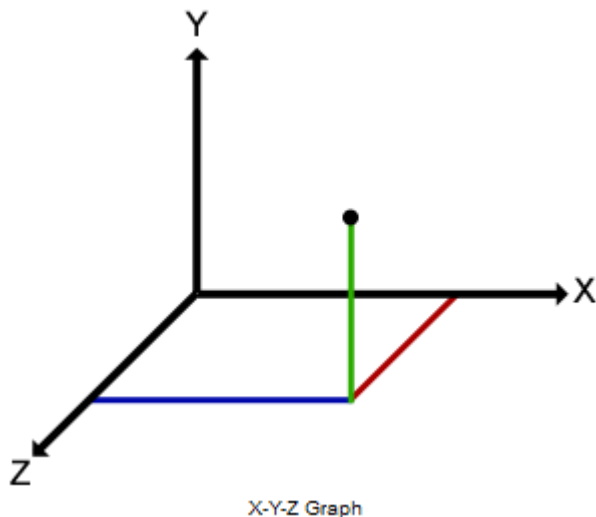
Student:
ing. Mengheris Ioana
An II, Master IISC

Cuprins

1. Introducere
2. Tehnologia 3D
3. Interactiune 3D
 - 3.1. Interactiune 3D cu utilizatorul
 - 3.1.1. Dispozitive de intrare
 - 3.1.2. Dispozitive de iesire
 - 3.1.3. Dispozitive de intrare-iesire (hibride)
 - 3.2. Tehnici de interactiune 3D
4. Driveri pentru dispozitivele 3D
 - 4.1. Driveri pentru dispozitive de intrare-iesire
 - 4.2. Dezvoltarea driverelor
 - 4.3. Modalitatea de functionare a driverelor
 - 4.4. Programarea driverelor
 - 4.5. Driverii pentru dispozitivele 3D
5. Provocari ale tehnologiei 3D
6. Concluzii
7. Bibliografie

1. Introducere

Spatiul tridimensional (3D) este un model geometric tri-parametric al universului fizic (fara a considera timpul) in care exista toata materia cunoscuta. Aceste trei dimensiuni pot fi etichetate printr-o combinatie de trei termeni alesi dintre lungime, grosime, inaltime, adancime si largime. Oricare trei directii pot fi alese, cu conditia sa nu stea in acelasi plan.[1]



Sursa: http://en.wikipedia.org/wiki/Three-dimensional_space#mediaviewer/File:Coord_planes_color.svg

Oamenii pot percepe relatiile spatiale dintre obiecte prin simpla observare a lor deoarece detine perceptie 3D, cunoscuta si sub termenul de perceptie in adancime. Pe masura ce privim, retina din fiecare ochi formeaza o imagine bidimensionala a mediului inconjurator iar creierul proceseaza cele doua imagini intr-o experienta vizuala 3D.[2] Totusi, faptul de a detine o astfel de viziune in ambii ochi (viziune stereoscopica sau binoculara), nu este singura modalitate de a percepe 3D. Oamenii care pot vedea doar cu un singur ochi (viziune monoculara) pot percepe lumea in 3D.



Sursa:

http://en.wikipedia.org/wiki/Visual_perception

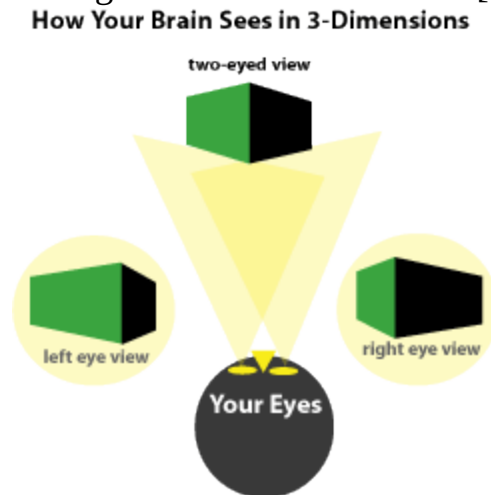
Instrumentele pe care oamenii le folosesc pentru a folosi perceptia 3D sunt:

- viziunea stereoscopică – cei doi ochi ofera imagini ușor diferite; obiectele apropiate apar separate față de obiectele departate.
- acomodarea – pe măsura ce omul focalizează asupra unui obiect apropiat sau departat, lentilele din ochi își modifică fizic mărimea, oferind un indiciu asupra distanței obiectului.
- paralaxă – pe măsura ce capul omului se mișcă dintr-o parte în alta, obiectele apropiate par a se mișca față de obiectele îndepărtate.
- familiaritatea mărimei – dacă omul cunoaște mărimea aproximativă a obiectului, atunci poate determina cu aproximație distanța obiectului în funcție de mărimea percepută a acestuia. În mod similar, dacă două obiecte au aceeași mărime dar unul este perceput ca fiind mai mare decât celălalt, atunci omul va presupune că obiectul mai mare este mai aproape.
- perspectiva aeriană – deoarece lumina este împrăștiată aleatoriu de către aer, obiectele departate detin un contrast mai mic decât obiectele apropiate.[3]

Pentru a reprezenta lumea tridimensională pe o suprafață plată (2D), cum ar fi un ecran, este de dorit de a simula cât mai multe din instrumentele de percepție prezentate. Deși în acest moment nu există o modalitate de a simula toate aceste instrumente în același timp, se folosește o combinație. De exemplu, perspectiva aeriană și familiaritatea mărimei sunt capturate automat pe camera video. Adăugarea imaginilor stereoscopice (o imagine separată pentru fiecare ochi) este o îmbunătățire semnificativă pentru percepția 3D a filmelor stereoscopice.[4]

2. Tehnologia 3D

Tehnologia 3D (tridimensională) oferă o multitudine de posibilități în aproximativ orice domeniu, în special în cele de divertisment. Aceasta se bazează pe instrumentele de achiziție 3D. Acestea sunt simulate în cadrul afișării 3D prin crearea unei iluzii de adâncime prin prezentarea de imagini diferite fiecărui ochi.[2]



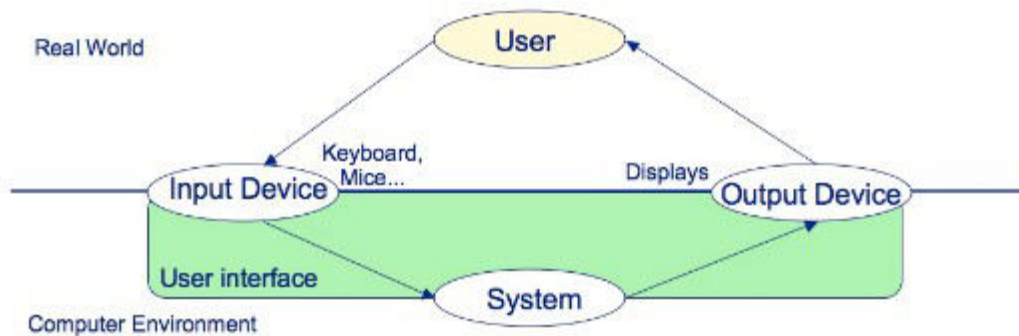
Sursa: <http://www.thebrainfactory.com/learn-about-3d.html>

Pornind de la acest concept, standarde 3D au fost dezvoltate, ce codează fluxul 3D în mod similar cu codarea unui flux normal, cu excepția separării lor pentru fiecare ochi. Aceste standarde diferă în funcție de modul de afișare a fluxului 3D. Când cele două imagini sunt capturate de cele două lentile și apoi generate pe un dispozitiv de afișare electronic 3D, afișarea se face fie prin prezentarea a două perspective ochilor la momente de timp diferite (folosind ochelarii acestea se pot vedea ca o imagine singulară), fie prin prezentarea ambelor perspective în același timp printr-un filtru special. În practică se folosesc dispozitive de intrare-iesire 3D pentru a realiza o interacțiune 3D a utilizatorului cu calculatorul (mașina).[5]

Aplicațiile și dispozitivele 3D au două componente importate și intercorelate. Prima componentă este tehnologia privind producerea luminii, a doua este tipul și forma geometrică a suprafeței pe care lumina este afișată. Există diferite metode pentru producerea luminii pe o suprafață geometrică. În timp ce tuburile catodice (CRT) și ecranele cu cristale lichide (LCD) sunt deja cunoscute și folosite, noi produse apar, cum sunt dispozitivele digitale optice DMD.[6]

3. Interactiune 3D

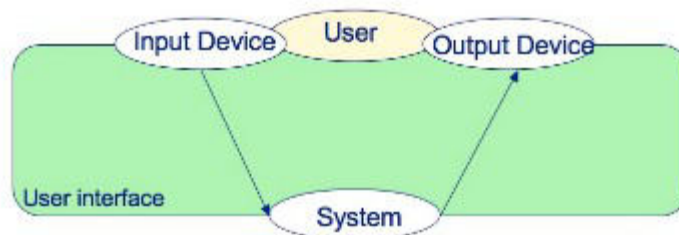
Interfetele cu utilizatorul sunt un mediu prin care utilizatorii umani pot comunica cu calculatoarele sau sistemele. Aceste interfete traduc actiunea utilizatorului intr-o reprezentare pe care masina o poate intelege si pentru care poate actiona si invers. Dispozitivele de intrare-iesire sunt instrumente fizice ce ofera acest tip de interactiune, astfel incat devin componente importante in construirea interfetelor cu utilizatorul. Modul in care se petrece interactiunea este implementat de diverse tehnici de interactiune. In interactiunea om-masina traditionala, utilizatorul interactioneaza cu mediul calculatorului din afara prin intermediul interfetelor.[5] Aceasta interactiune poate fi compusa intr-un circuit de interactiune continuu:



[5]

Utilizatorul percepe starea actuala a sistemului si isi transfera scopul cu ajutorul dispozitivelor fizice de intrare. Sarcina interfetei cu utilizatorul este de a traduce actiunea utilizatorului in semnale pe care sistemul le poate intelege. Astfel, sistemul percepe scopul utilizatorului si actioneaza in consecinta. Mai departe, interfata transcrie scopul utilizatorului intr-o reprezentare perceptibila uman si o transfera catre utilizator cu ajutorul dispozitivelor de iesire.

Datorita avansarii tehnologiei partilor standard ale interfetei, interactiunea om-masina poate fi efectuata intr-un context spatial 3D. Aceasta inseamna ca utilizatorul interactioneaza din mediul calculatorului. Astfel, in termenii unei interactiuni 3D, utilizatorul nu interactioneaza cu ajutorul unei interfete, ci el poate interactiona ca o parte a interfetei.[5]



[5]

In domeniul stiintific, interactiunea 3D este o forma de interactiune om-masina, in

care utilizatorii sunt capabili de a se misca si de a interactiona in spatiul 3D. Atat oamenii cat si masinile proceseaza informatii in care pozitia fizica a elementelor in spatiul 3D este relevanta. Spatiul 3D folosit pentru interactiune poate fi spatiul fizic real, o reprezentare spatiala virtuala simulata pe calculator, sau o combinatie a amandurora. Cand spatiul real este folosit pentru datele de intrare, oamenii efectueaza actiuni sau ofera comenzi masinii folosind un dispozitiv de intrare ce detecteaza pozitia 3D a actiunii umane. Cand spatiul real este folosit pentru datele de iesire, scena 3D simulata virtual este proiectata in mediul real folosind un dispozitiv de iesire sau o combinatie de astfel de dispozitive.[6] Literatura stiintifica deseori delimiteaza clar distinctia dintre intrare si iesire; astfel, oamenii de stiinta denumesc ecranul drept un dispozitiv de iesire pasiv iar mouse-ul drept un dispozitiv de intrare pur.[7] Totusi, aproape toate interactiunile om-masina necesita ambele tipuri de dispozitive, intrare si iesire. De exemplu, la ce ar folosi un mouse fara feedback-ul corespunzator oferit de catre cursorul de pe ecran si fara sunetul si senzatiile de atingere a butoanelor atunci cand este apasat? Astfel, distinctia dintre dispozitive de intrare si de iesire devine dificila. O foaie de hartie poate fi folosita pentru a nota idei (intrare) cat si pentru a reprezenta idei (iesire). Dispozitivele de intrare si de iesire produc o fisura in lumea de biti a calculatorului si in lumea reala a utilizatorului. Dispozitivele de intrare pentru calculatoare constau in informatie senzitiva despre mediul fizic. Printre exemple se afla mouse-ul, care simte miscarea pe suprafata sa si tastatura care detecteaza contactul la apasarea tastelor de catre utilizator. Dispozitivele de iesire ale calculatorului pot compria orice emisie sau modificare a mediului fizic, cum ar fi ecranul, sistemul audio sau dispozitivele de feedback tactil. O tehnica de interactiune este fuzionarea dispozitivelor de intrare si de iesire, ce constau din elemente hardware si software, ce ofera o metoda utilizatorului de a efectua o sarcina.

Sarcina fundamentala a unei interactiuni 3D om-masina este de a transfera informatii intre creierul utilizatorului si lumea calculatorului. Progresul acestei arii incearca sa creasca largimea de banda folositoare a interfetei prin cautarea rapida, naturala si metode conveniente pentru utilizatorii sa transmita informatii catre calculatoare, cat si mecanisme placute de a oferi feedback catre utilizator. Pe partea canalului de comunicatie a utilizatorului, interactiunea este constransa de natura atentiei umane, de cunoasterea si abilitatile motorii si perceptuale. Pe partea canalului de comunicatie al calculatorului, interactiunea este constransa doar de tehnologie si de metodele ce se pot inventa. Cercetarea in centrele de dispozitive de intrare-iesire se invarte in jurul ambelor capete ale canalelor de comunicatie: dispozitivele si tehnicile pe care calculatoarele le pot folosi pentru comunicarea cu utilizatorii, precum si abilitatile si procesele perceptuale pe care utilizatorii le pot folosi pentru a comunica cu calculatoarele.[8] Cercetarile de baza cauta teorii si principii care detin informatii despre facilitatile perceptuale si cognitive umane, precum si modele ce pot prezice performanta utilizatorului in sarcinile computationale.

Tranzactiile interfetei cu utilizatorul sunt compuse din urmatoarele sarcini:

- selectia – alegerea obiectelor dintr-un set de alternative.

- pozitia – specificarea pozitiei dintr-un interval. Aceasta include alegerea unei coordonate a ecranului cu un dispozitiv de indicare.
- orientarea – specificarea unui unghi sau orientare tridimensionala.
- traiectoria – specificarea unei serii de pozitii si/sau orientarii in timp.
- cantitatea – specificarea unei valori numerice exacte.
- text – introducerea unor date simbolice.

Aceste sarcini definesc majoritatea interactiunilor 3D om-masina.[8]

3.1.1. Dispozitive de intrare

Dispozitivele de intrare sunt componente fizice folosite pentru traducere scopului utilizatorului in comenzi sau informatii pe care calculatorul le poate percepe. Contrar tehnicilor de interactiune, dispozitivele de intrare sunt instrumente fizice ce efectueaza obiectivul interactiunii. Dispozitivele de intrare sunt instrumente cu care utilizatorul poate controla un sistem sau un program.

[9]http://people.cs.vt.edu/bowman/3dui.org/course_notes/siggraph2001/io.pdf

Dispozitivele de intrare pot fi caracterizate drept dispozitive de intrare discrete sau continue, depinzand de tipul de eveniment pe care il genereaza. Intrarea discreta, cum ar fi apasarea unui buton, genereaza un singur semnal, in timp ce intrarea continua genereaza un sir continuu de evenimente. Tastatura si mouse-ul sunt doua astfel de dispozitive de intrare, insa tehnologiile noi permit date de intrare dintr-un context spatial 3D.[9]

Un exemplu de dispozitiv de intrare este masa de pictat, dispozitiv 3D discret, folosit in aplicatii CavePaintng, ce reprezinta un sistem de pictare scene 3D intr-un mediu virtual.[10]



[10]

Tracker-ul este un dispozitiv de intrare 3D in care pozitia si orientarea obiectelor aflate in miscare sunt urmarite ca un sir de evenimente.



[12]

Astfel, doar parti ale corpului utilizatorului sunt urmarite, cum ar fi mana dintr-o manusa (Dataglove) sau ochii, sau chiar intregul corp al utilizatorului poate fi urmarit. Folosind manusele de date (Dataglove) drept dispozitive de intrare pentru interactiunea cu mediul virtual este o abordare intuitiva, deoarece manipularea manuala este o modalitate de interactiune majora in mediile fizice naturale. [11]



[11]

Un alt exemplu de dispozitiv de intrare continuu este dispozitivul bioelectric ce citeste semnalele nervoase musculare ce se emit din antebrat. Aceste semnale nervoase sunt capturate de un electrod positionat pe brat. Semnalele nervoase sunt analizate folosind un software de recunoastere a modelelor si sunt apoi rutate catre calculator pentru a elabora comenzi relevante interfetei.[12]



[9]

Alegerea dispozitivului de intrare este puternica legata de cea a dispozitivului de iesire, deoarece acesta din urma ofera feedback utilizatorului, adica o parte esentiala in intelegerea interactiunii din contextul spatial. Manusile de date pot fi atat dispozitive de intrare cat si de iesire. Pentru a oferi o mapare naturala si eficienta intre tehnicile de interactiune si partile hardware, designer-ul interfetei cu utilizatorul trebuie sa aiba o intelegere corespunzatoare a avantajelor si limitarilor dispozitivelor folosite.[9] Modalitatile spatiale de intrare cum sunt gesturile si vorbirea castiga din ce in ce mai multe capacitati pentru interactiunea 3D.

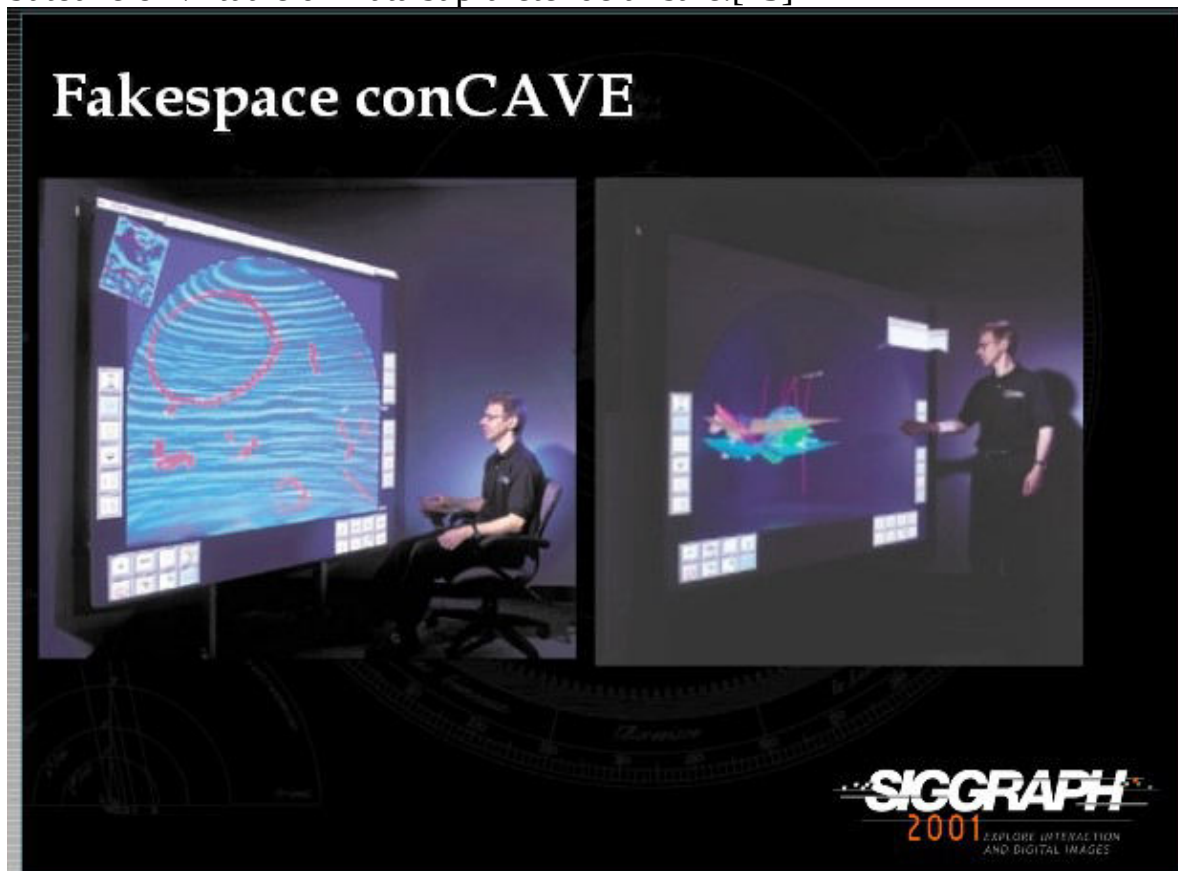
3.1.2. Dispozitive de iesire

Dispozitivele de iesire in interfetele cu utilizatorul sunt componentele fizice ce prezinta informatia generata de calculator si perceputa de utilizator. Informatia de iesire reprezinta feedback-ul pentru utilizator. Informatia oferita de dispozitivele de iesire intr-o interactiune om-masina sunt caracterizate drept vizuale, auditorii, tactile si olfactive.

3.1.2.1. Dispozitive de iesire vizuale

Dispozitivele de iesire vizuale sunt cele mai comune, deoarece informatia vizuala este cel mai bine perceputa de catre utilizatorul uman. Dispozitivele de afisare full-imersiva absorb lumea reala intr-o reprezentare grafica a lumii virtuale. Dispozitivele de afisare semi-imersiva permit utilizatorului sa observe atat lumea reala cat si cea virtuala. Mai mult, aspectele precum mobilitatea si campul vizual sunt implementate in lumea virtuala.[8]

Dispozitivele de afisare surround-screen (cum ar fi CAVE) constau din 3 pana la 6 planuri de proiectie, `aranjate in forma de cub in care se afla utilizatorul astfel incat i se ofera acestuia o vedere stereoscopica. Dispozitivul CAVE creeaza o viziune tunel corecta spatial 3D prin imagini de date volumetrice ce se extind de la podea pana la tavan. Perspectiva 3D este imbunatatita in adancime, generand o imagistica stereoscopica fara nevoia de ochelari speciali 3D. Dispozitivul ofera utilizatorului atat interactiune 3D prin folosirea unui paddle cat si interactiune 2D prin folosirea butoanelor virtuale din fata suprafetei de afisare.[13]



[13]

Dispozitivele de afisare head-mounted (HMD) sunt dispozitive ce contin lentile incorporate intr-o casca, in ochelari sau intr-un vizor. Prin afisarea unei imagini fiecarui ochi, dispozitivul poate fi folosit pentru a reda imagini stereoscopice. O combinatie a acestora cu dispozitive ce urmaresc miscarea capului, ii permit utilizatorului sa isi intoarca capul pentru a vedea lumea inconjuratoare virtuala. Aceasta tehnologie HMD poate fi folosita pentru dispozitivele full-immersive dar si pentru cele semi-immersive, astfel incat realitatea este marita prin imagini virtuale. Dispozitivele de afisare virtuala pe retina includ dispozitive ce urmaresc miscarea ochilor astfel incat permit proiectarea informatiei cu ajutorul unui laser direct in ochi.[14]

HMDs and BOOMs



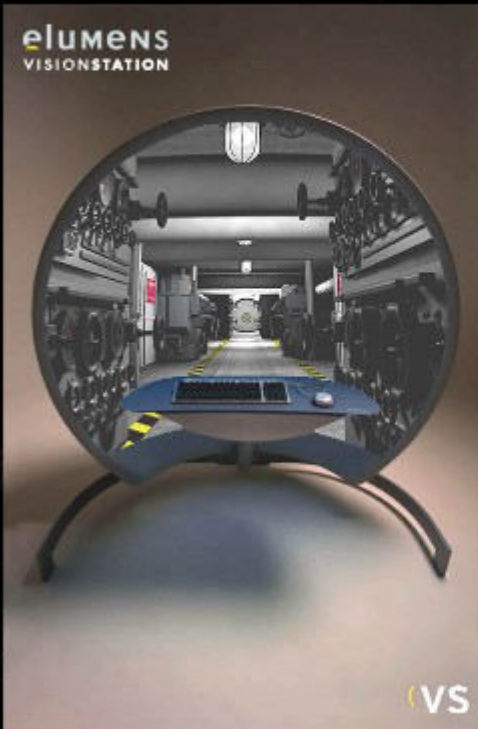
SIGGRAPH
2001
EXPLORE IMMERSIVE FILM
AND DIGITAL IMAGES

[15]

Dispozitivele de afisare head-up (HUD) sunt dispozitive de afisare transparente ce prezinta datele fara a restrictiona vederea utilizatorului.[15] Acestea sunt folosite in cadrul avioanelor sau a automobilelor pentru a prezenta informatie simbolica, cum ar fi indicatoarele de viteza, sau pot fi folosite intr-o combinatie cu un sistem de urmarire a miscarii capului pentru a afisa informatie conforma.

Dispozitivele VisionStation folosesc o suprafata de afisare sferica proiectata in fata utilizatorului. Acestea folosesc software si optica specifica pentru lentilele de proiectare si afiseaza imagini intr-un camp vizual de 180 pe 180 grade. Utilizatorul este pozitionat in fata unei mese si poate interactiona cu aplicatiile 3D folosind tastatura, mouse, cat si dispozitive de intrare 3D.[8]

VisionStation from Elumens



SIGGRAPH
2001
EXPLORE INTERACTION
AND DIGITAL IMAGES

[8]

Dispozitivele VRD (Virtual Retina Display) folosesc o sursa de fotoni pentru a genera o raza de lumina coerenta ce permite sistemul sa proiecteze o pata de difractie limitata pe retine. Raza de lumina este modulata in intensitate pentru a se potrivi cu intensitatea imaginii ce va fi proiectata. Raza este apoi scanata pentru a plasa fiecare imagine in punctul sau pixelul la pozitia potrivita a retinei.[16]

Virtual Retinal Display



SIGGRAPH
2001
EXPLORE INTERACTION
AND DIGITAL IMAGES

[16]

Imprimarea 3D este una dintre diversele procese pentru a realiza un obiect tridimensional. In imprimarea 3D, sunt folosite procese aditive, in care straturi succesive de material sunt stabilite sub control computerizat. Aceste obiecte pot fi de aproape orice forma sau geometrie, si sunt fabricate dintr-un model 3D sau alte surse de date electronice. O imprimanta 3D este un tip de robot industrial.[31]

Termenul de imprimare 3D in sens original si tehnic precis se refera la procesele care depoziteaza secvential materialul pe un pat de pulbere cu capete de imprimante cu jet de cerneala. Recent, sensul termenului s-a extins pentru a cuprinde o varietate mai larga de tehnici, cum ar fi procesele bazate pe extrudare si sinterizare.[31]

Procesul de modelare manuala de pregatire a datelor geometrice pentru grafica 3D pe calculator este similar cu artele plastice, cum ar fi sculptura. Scanarea 3D este un proces de analiza si colectare a datelor digitale pe forma si aspectul unui obiect real. Pe baza acestor date, modele tridimensionale ale obiectului scanat poate fi astfel produse.

[31]

3.1.2.2. Dispozitive de iesire auditive

Dispozitivele de iesire auditive ofera un canal de comunicatie secundar atractiv, informational sau se semnalare a alarmelor ce poate fi transferat acustic, astfel incat sa reduca incarcarea cognitiva. Semnalele auditorii pot fi folosite pentru ghidarea atentiei utilizatorilor intr-o directie specifica pentru a sustine locatia obiectelor in mediul inconjurator sau substituirea unei alte modalitati senzoriale precum apasarea unui buton. Scopurile mari ale unei astfel de interfete 3D este generarea de sunete 3D si sonificari,

transformarea informatiei in sunete. Mai mult, efectele ambientale acustice, precum ciripitul pasarilor, poate oferi un simt al realismului in mediile virtuale.[17]

3.1.2.3. Dispozitive de iesire tactile

Dispozitivele de iesire tactile ofera un simt al fortei si al atingerii. Cu astfel de dispozitive se pot integra efecte atractive la interfetele cu utilizatorul, deoarece simtul si atingerea face mediul virtual mai realist. Dispozitivele tactile pot fi folosite pentru atragerea atentiei utilizatorului sau pentru oferirea de feedback si indicii informationale. Pentru a transfera informatie tactila, dispozitivele de iesire trebuie sa fie conectate fizic la utilizator. Aceasta se poate realiza fie prin instrumente fixe la sol, cum sunt simulatoarele de zbor sau banda rulanta de alergat, fie prin instrumente fixe pe corp, prin care se ofera mai multa libertate utilizatorului.[18]

3.1.2.4. Dispozitive de iesire multimodale

Dispozitivele de iesire multimodale sunt o combinatie de diferite tipuri de dispozitive, astfel incat se elimina dezavantajele tipice fiecarui tip. In lumea reala, interactiunea diferitelor simturi detine un rol esential pentru ca oamenii sa devina constienti de mediul inconjurator. Oamenii percep informatie vizuala, acustica si tactila simultan, astfel incat este eficienta combinarea diferitelor tipuri de dispozitive pentru a oferi un mai mare simt de realism.[8]

3.1.3. Dispozitive de intrare-iesire (hibride)

Dispozitivele de intrare-iesire (hibride) au abilitatea de a genera atat evenimente discrete cat si continue, fiind mult mai flexibile. Doua din cele mai comune dispozitive hibride sunt joystick-ul si mouse-ul. Un alt dispozitiv din aceasta categorie este tableta cu stilou. Tablete cu stilou devin din ce in ce mai populare in aplicatiile cu mediu virtual deoarece ofera abilitatea de a interactiona 2D ce ofera o combinatie folositoare in anumite interfate.[19]

Mouse-ul spatial (Magelland) este un dispozitiv de intrare originat din manipularea telerobotica. Presiunea usoara asupra capului mouse-ului va genera deflectii usoare in X, Y si Z, ce poate misca obiecte in spatiul 3D. De asemenea, detine o serie de butoane ce vor genera evenimente discrete. Mouse-ul inel este un mic dispozitiv purtat pe degetul utilizatorului ce foloseste urmarirea ultrasonica. Detine doua butoane pentru generarea de evenimente discrete. Avantajele majore ale acestui dispozitiv sunt functionarea wireless si costul mic. Fly Mouse este un alt mouse 3D ce foloseste urmarirea ultrasonica. Acest dispozitiv detine cinci butoane in loc de doua, si poate fi

folosit si ca microfon.[19]

Mai exista un tip de dispozitiv de intrare ce detine functionalitatile unui mouse, inasa este atasat mainii asemenea unei manusi pentru o interactiune mai buna si mai rapida cu utilizatorul.

3.2 Tehnici de interactiune 3D

Calitatea interfetelor 3D cu utilizatorul depinde nu doar de iesirea furnizata dar si de multimea posibilitatilor de interactiune. Astfel, scopul unei interfete 3D este de a realiza o experienta de interactiune simpla si intuitiva pentru utilizator. Tehnicile de interactiune sunt metode folosite pentru a dobandi o sarcina impusa prin intermediul interfetei. In acest context, componentele hardware si software trebuie luate in calcul. Tehnicile de interactiune sunt puternic dependente de cerintele particulare astfel incat nu au fost definite standarde. Categorizarea interactiunii spatiale in termeni generali ofera principii de ghidare si de design pentru interactiune. Pentru a face interactiunea om-masina cat mai naturala si intuitiva posibil, sunt folosite metafore. Metaforele sunt modele comprehensive mental ce permit utilizatorului sa aplice cunostintele generale in mediul virtual.[20]

3.2.1. Manipulare si selectie

Sarcinile de manipulare sunt fundamentale pentru mediile fizice si virtuale astfel incat este indispensabila furnizarea de tehnici pentru selectie, achizitionare a unui obiect particular, pozitionare, mutarea unui obiect sau pentru rotatie, schimbarea orientarii unui obiect. Exista cateva abordari pentru aceste tehnici, inasa eficacitatea lor depinde de o multitudine de variabile precum scopul aplicatiei, marimea obiectelor, distanta dintre utilizator si obiect si starea utilizatorului. O mana virtuala poate fi o cale intuitiva de interactiune deoarece manipularea manuala este o modalitate majora de interactiune in mediul fizic natural. Inasa doar obiectele din imediata apropiere a utilizatorului pot fi manipulate. Pentru a depasi aceasta problema, tehnicile de selectare radiala sau conica ofera o modalitate usoara si independenta de distanta pentru selectarea obiectelor. Atunci cand un vector sau o raza, ce emana dintr-o mana virtuala, intersecteaza un obiect virtual, acesta poate fi ales folosind un trigger, precum o comanda vocala sau un buton. Astfel de abordari ofera doar pozitionari sau tehnici de rotatie slabe, obiectele pot fi manipulate doar in miscari radiale din jurul utilizatorului si rotite in jurul axei, definite de un vector. Mai mult, apar probleme de blocare. Obiectele din spatele altora nu pot fi selectate si prin urmare nu pot fi manipulate.[21]

O alternativa la tehnicile prezentate anterior este selectia folosind doi vectori ce emana din fiecare mana a utilizatorului. Obiectul aflat la intersectia a doi vectori poate fi selectat, astfel incat si cele blocate pot fi manipulate.[21]

3.2.2. Navigarea

Navigarea este o sarcina umana fundamentala in mediul fizic, cat si in cel virtual, in care utilizatorul trebuie sa navigheze in jocuri sau in Internet cu ajutorul browser-elor. Pentru a oferi miscare eficienta, utilizatorul trebuie sa isi cunoasca pozitia fata de alte obiecte din mediul inconjurator. Utilizatorul are nevoie de suport pentru constientizarea spatiala. Navigarea poate fi impartita in calatorie si gasirea caii de navigatie. Calatoria este componenta motorie a navigarii, reprezinta sarcina de miscare dintr-o locatie curenta intr-o locatie dorita. Gasirea caii de navigatie reprezinta procesul cognitiv de definire a traiectoriei intr-un mediu, astfel cunostintele spatiale trebuie folosite pentru a construi harti mentale. Folosirea si achizitionarea cunostintelor spatiale sunt furnizate de catre indicii naturali si artificiali. Gasirea caii de navigatie include miscarea vizuala, miscarea reala si indici auditivi.[21]

3.2.3. Controlul sistemului

Controlul sistemului este o sarcina a utilizatorului de a accesa functionalitatea sistemului computational prin elaborarea de comenzi. Comenzile sunt folosite pentru a-i cere sistemului de a realiza functii particulare, pentru a schimba modul de interactiune si pentru a schimba starea sistemului. Utilizatorul elaboreaza ce ar trebui sa efectueze sistemul, dar ii permite sistemului sa aleaga modalitatea de realizare a comenzii. Dispozitivele de control precum ferestrele, pictogramele, meniurile si pointerii (WIMP) sunt, in mod uzual, tipuri de interactiune bidimensionale. Acestea pot fi adaptate interfetelor 3D cu utilizatorii. Problema acestei adaptari este plasarea in spatiul 3D a dispozitivelor de control. O alternativa este controlul sistemului prin gesturi, dar aceasta este folositoare doar atunci cand aceste gesturi sunt in relatie cu gesturi naturale bine-definite.[21]

4. Drivere pentru dispozitivele 3D

4.1. Drivere pentru dispozitive de intrare-iesire

In domeniul calculatoarelor, un driver pentru dispozitive este un program de calculator ce opereaza sau controleaza un tip particular de dispozitiv atasat unui calculator sau sistem computational. Un driver ofera o interfata software dispozitivelor hardware, permitandu-le sistemelor de operare si altor programe de calculator sa acceseze functii hardware fara nevoia de a cunoaste detaliile precise ale hardware-ului folosit.[22]

Un driver comunica cu dispozitivul prin magistrala calculatorului sau a sistemului computational sau prin subsisteme de comunicatie la care se conecteaza hardware-ul.

Cand un program de apelare invoca o rutina din driver, driverul initiaza comenzi catre dispozitiv. Odata ce dispozitivul trimite date inapoi catre driver, driverul poate invoca rutine din programul initial. Drivererele sunt dependente de hardware si specifice sistemului de operare. Acestea ofera manevrarea intreruperilor necesara pentru orice interfata hardware asincrona dependenta de timp.[24]

Scopul driverelor este de a simplifica programarea. Ele actioneaza ca un translator intre un dispozitiv hardware si aplicatiile sau sistemele de operare ce il folosesc.[22] Programatorii pot scrie cod de aplicatie la un nivel inalt in mod independent pentru orice dispozitiv hardware specific pe care utilizatorul final il va folosi. De exemplu, o aplicatie de nivel inalt pentru interactionarea cu un port serial poate avea doua functii pentru trimiterea datelor si pentru primirea datelor. Un driver de dispozitiv ce implementeaza aceste functii va comunica cu un controlor particular al portului serial intalat in calculatorul utilizatorului. Comenzile necesare pentru a controla UART 16550 sunt cu mult mai diferite fata de comenzile necesare pentru a controla un convertor de port serial FTDI, dar fiecare driver de dispozitiv abstractizeaza aceste detalii in interfata software.[25] In domeniul computational, o interfata este o limita comuna pentru doua componente separate a unui sistem computational ce efectueaza schimb de informatie. O interfata software se poate referi la interfete de diferite tipuri: un sistem de operare poate detine o interfata cu diferite tipuri hardware. Aplicatiile sau programele ce ruleaza pe un sistem de operare pot necesita sa interactioneze prin fluxuri, si in programe obiect-orientate, obiectele dintr-o aplicatie pot necesita sa interactioneze prin metode.[25]

4.2. Dezvoltarea driverelor

Pentru dezvoltarea driverelor este necesara o intelegere profunda a modalitatii de functionare atat a componentelor hardware cat si software pentru o platforma de functii data. Deoarece drivererele necesita acces de nivel scazut la functiile hardware pentru a fi operabile, drivererele opereaza de obicei intr-un mediu privilegiat la nivel inalt si poate cauza probleme operationale de sistem daca exista ceva in neregula. Pe de alta parte, majoritatea componentelor software la nivelul utilizatorului ale sistemelor de operare moderne pot fi oprite fara a afecta major restul sistemului. Chiar si drivererele ce executa in user mode pot face ca un sistem sa se opreasca daca dispozitivul este programat eronat. Acesti factori fac ca diagnosticarea problemelor sa fie dificila.[26] Sarcina de a scrie drivere revine inginerilor software sau inginerilor de calculator ce lucreaza in companiile de dezvoltare hardware, deoarece detin mai multe informatii privind designul propriilor componente hardware. De asemenea, este considerat ca este in interesul producatorului hardware de a garanta ca clientii lor pot folosi dispozitivele hardware produse de catre ei in modalitatea optima. De obicei, driverul de dispozitiv logic (LDD) este scris de catre producatorul de sistem de operare, iar driverul de dispozitiv fizic (PDD) este implementat de catre producatorul de dispozitiv. In ultimul timp, non-producatorii au scris numeroase drivere de dispozitive, in special pentru folosirea in sistemele de operare gratuite si de tip open-source. In astfel de cazuri, este

important ca producatorul de software sa ofere informatii despre modul in care dispozitivul comunica. Desi aceasta informatie poate fi invata prin inginerie inversa, acest lucru este mult mai dificil de invatat pentru hardware decat pentru software.[26]

Microsoft a incercat sa reduca instabilitatea sistemului din cauza programarii slabe a driverelor prin creare unui nou framework pentru dezvoltarea driverelor, denumit Windows Driver Foundation (WDF). Aceasta include User-Mode Driver Foundation (UMDF), care incurajeaza dezvoltarea anumitor tipuri de drivere – in primul rand cele care sa puna in aplicare un protocol bazat pe mesaje pentru comunicarea cu dispozitivele lor – cum ar fi drivere de tip user-mode. Daca aceste drivere prezinta la un moment dat o anumite defectiune, acestea nu provoaca instabilitatea sistemului. Modelul Kernel-Mode Driver Framework (KMDF) continua sa permita dezvoltarea de drivere de dispozitive kernel-mode, dar incearca sa ofere implementari standard de functii care sunt cunoscute de a provoca probleme, inclusiv anulara operatiunilor I / O, de gestionare a energiei, si suport pentru dispozitivele de tip plug & play.[26]

Apple detine un framework de tip open-source pentru dezvoltarea driverelor pentru sistemul de operare Mac OS X, numit I/O Kit. In mediile Linux, programatorii pot construi drivere de dispozitive ca parte din kernel, separabile ca module de incarcare, sau ca drivere de dispozitive de tip user-mode (pentru anumite tipuri de dispozitive unde exista interfetele kernel, cum sunt dispozitivele USB).[26]

Drivererele de dispozitiv, in special pe platformele Microsoft Windows moderne, se pot executa in kernel-mode (Ring 0 pe procesoare x86), sau in user-mode (Ring 3 pe procesoare x86). [27] Beneficiul principal al executarii unui driver in user-mode este stabilitatea imbunatatita, deoarece un driver de dispozitiv scris gresit in user-mode nu poate prabusi sistemul prin suprascrierea memoriei kernel. [28] Spatiul Kernel pot fi accesat prin user-mode numai prin utilizarea de apeluri de sistem. Programele utilizatorilor finali, cum ar fi shell UNIX sau alte aplicatii bazate pe GUI sunt parte a spatiului utilizatorului. Aceste aplicatii interactioneaza cu hardware-ul prin intermediul functiilor acceptate de kernel.[27]

Datorita diversitatii sistemelor moderne hardware si de operare, drivererele functiona in multe medii diferite [29] Drivererele pot interfata cu:

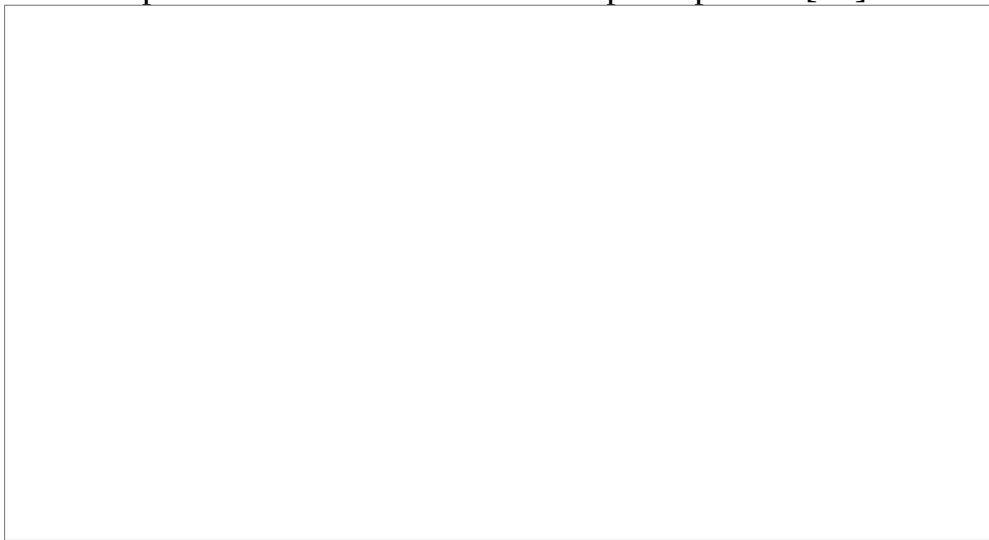
- Imprimante
- Adaptoare video
- Placi de retea
- Placi de sunet
- Magistrale locale de diverse tipuri
- Magistrale I / O de largime de banda mica de diverse tipuri (pentru dispozitive de indicare, cum ar fi mouse, tastatura, USB, etc.)
- Dispozitive de stocare de calculator, cum ar fi hard disk, CD-ROM, si magistralele floppy disk (ATA, SATA, SCSI)
- Suport pentru implementarea diferitelor sisteme de fisiere
- Scanere

- Aparate foto digitale
Nivelurile comune de abstractizare pentru driverele de dispozitiv includ:
- Pentru hardware:
 - interfatare directa
 - scriere sau citire de la un registru de control al dispozitivului
 - folosirea unor interfete de nivel superior (de exemplu, Video BIOS)
 - utilizarea unui alt driver de dispozitiv de nivel inferior (de exemplu driverele de sistem de fisiere ce folosesc drivere de disc)
- Pentru software:
 - permiterea accesului sistemului de operare direct la resursele hardware
 - implementarea doar a primitivelor
 - implementarea unei interfete pentru software ce nu necesita driver (de exemplu, TWAIN)
 - implementarea unui limbaj (de exemplu, PostScript)

Deci, alegerea si instalarea driverelor de dispozitiv corecte pentru hardware-ul dat este de multe ori o componenta cheie a configuratiei sistemului calculatorului.[29]

4.3. Modalitatea de functionare a driverelor

Drivere de dispozitiv opereaza in stratul kernel al sistemului de operare. Kernel este parte a sistemului de operare care interactioneaza direct cu structura fizica a sistemului. in loc de a accesa un dispozitiv direct, un sistem de operare incarca driverele de dispozitiv si solicita functiile specifice in software-ul de driver pentru a executa sarcini specifice pe dispozitiv. Fiecare driver contine codurile specifice de dispozitiv necesare pentru efectuarea actiunilor de pe dispozitiv.[29]



[29]

Daca se considera exemplul unei imprimante, atunci cand este conectata la calculator si driverul de dispozitiv specific este instalat, un obiect dispozitiv este creat pe computer, care este proiectat pentru a controla dispozitivul. Acest obiect dispozitiv

reprezinta imprimanta si modurile sale de structura fizice care permit sistemului de operare al computerului sa controleze functiile sale. Cand se alege o operatie (cum ar fi Control + P pentru a imprima un document) de pe imprimanta, aceasta comanda se duce la driverul de dispozitiv prin kernel al sistemului de operare. Apoi un program de apelare invoca o rutina in driverul de dispozitiv si driverul emite comenzile corespunzatoare pentru microcontrolere din cadrul imprimantei. Mai mult, aceste microcontrolere controleaza componentele imprimantei pentru a incepe imprimarea documentului.[31]

4.4. Programarea driverelor

Un driver de dispozitiv este nimic mai mult decat o entitate care implementeaza functiile de baza I / O - citire, scrie, primire configurare, si stabilire configurare. De obicei un driver de dispozitiv foloseste si gestioneaza intreruperile din dispozitiv. in timp ce interfata este generica, iar driverul de dispozitiv este independent, implementarea driverului tine cont doar de designerul driverului.[27]

Acetia fiind spuse, motivul pentru utilizarea un driver de dispozitiv este de a oferi acces la un dispozitiv din codul aplicatiei. Cei mai multi dintre cei care programeaza drivere sunt, de asemenea, preocupati de a face acest acces cat mai simplu posibil, si in acelasi timp cat mai eficient posibil.[29]

Cele mai multe drivere de dispozitiv au in vedere miscarea informatiei, de exemplu a datelor de bytes de-a lungul unei interfete seriale, sau a pachetelor intr-o retea. in scopul de utiliza cat mai eficient resursele sistemului, sunt folosite intreruperile. Acest lucru va permite altor operatiuni de procesare de aplicatii sa aiba loc in timp ce transferurile de date sunt in curs de desfasurare, cu intreruperi folosite pentru a indica cand au avut loc diverse evenimente. De exemplu, un port serial genereaza in mod tipic o intrerupere dupa ce un caracter a fost trimis "down the wire" si interfata este gata pentru un altul. Este logic de a permite prelucrarea altor aplicatii in timp ce datele sunt trimise deoarece acest lucru poate dura destul de mult timp. intreruperea poate fi folosit pentru a permite driverului sa trimita un caracter imediat ce trimiterea caracterului actual este finalizata, fara nici o participare activa a codul de aplicatie.[29]

intreruperile permit dispozitivelor sa notifice CPU atunci cand detin date pentru tranfer sau atunci cand o operatiune este completa, ceea ce permite ca CPU sa indeplineasca alte atributii atunci cand transferurile de I / O nu au nevoie de atentie imediata.

Procesorul are o linie de cerere a intreruperilor, cerere care este detectata dupa fiecare instructiune.[30]

- Controlerul unui dispozitiv cere o intrerupere prin afirmarea unui semnal pe linia de cererere de intrerupere.
- Procesorul efectueaza apoi o stare de salvare, apoi transfera controlul catre rutina

de tratare a intreruperilor la o adresa fixata in memorie.

- Rutina de tratare a intreruperii determina cauza intreruperii, efectueaza prelucrarile necesare, efectueaza o stare de restaurare, si executa o intoarcere de la instructiunea de intrerupere pentru a returna controlul la CPU.

Urmatoarea figura ilustreaza procedura tratarii intruperilor I/O:



[30]

Descrierea de mai sus este adecvata pentru intreruperile I / O simple, dar exista trei necesitati in sistemele de calcul modern, care complica aceasta imagine:[30]

1. Necesitatea de a amana manevrarea intreruperilor in timpul prelucrarilor critice.
2. Nevoia de a determina ce rutina de tratare a intreruperilor trebuie invocata, fara a fi nevoie de un sondaj al tuturor dispozitivelor pentru a vedea care are nevoie de atentie.
3. Nevoia de intreruperi multi-nivel, astfel incat sistemul sa poata face diferenta dintre intreruperile cu prioritate mare si cele cu prioritate mica pentru a oferi raspunsul adecvat.

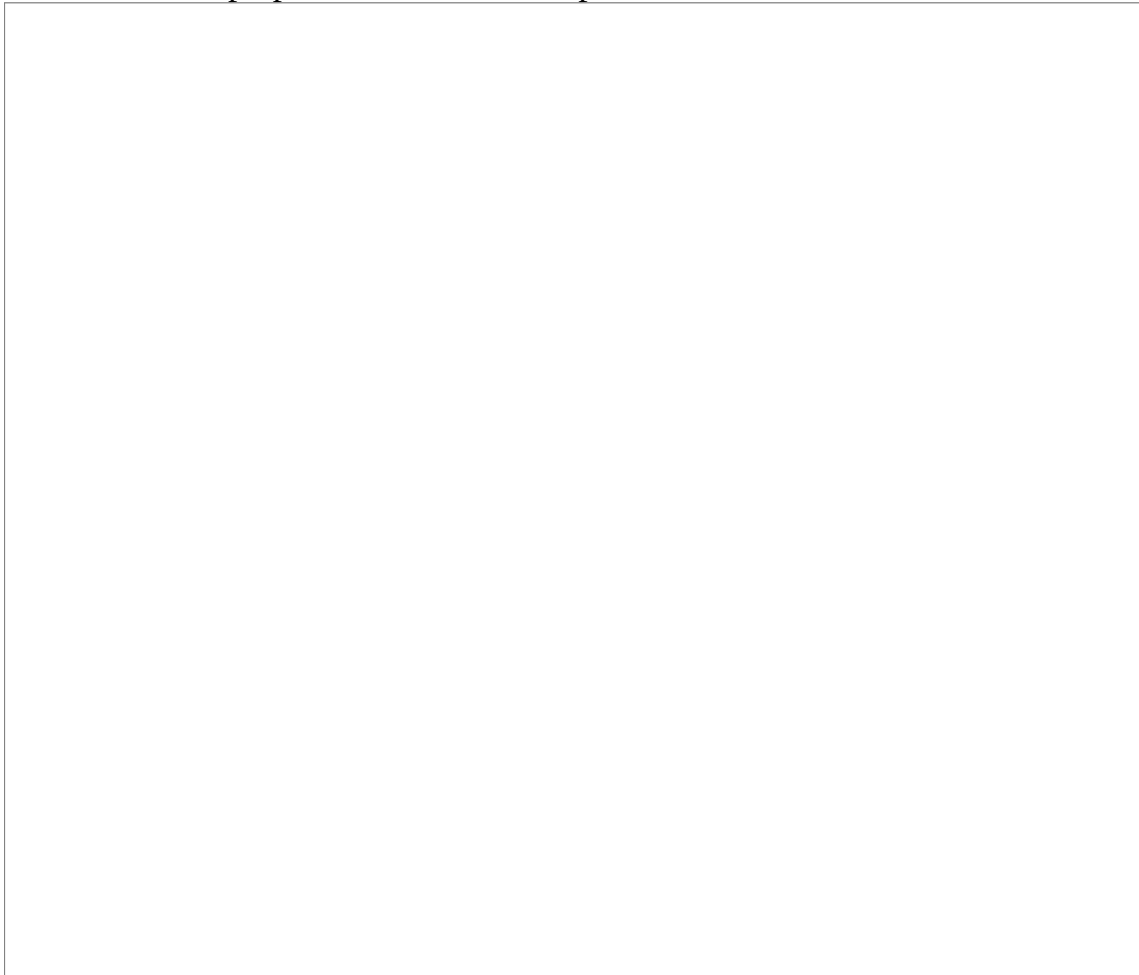
Aceste aspecte sunt manevrate in arhitecturile de calculatoare moderne, prin

componente hardware de control al intreruperilor.[29]

- Cele mai multe procesoare au acum doua linii de cerere a intreruperilor: una care nu este mascata pentru conditiile de eroare critice si una care este mascata, astfel incat procesorul poate ignora temporar in timpul prelucrarilor critice.
- Mecanismul de intrerupere accepta o adresa, care este de obicei una dintre un mic set de numere pentru un offset intr-un tabel numit vectorul de intrerupere. Acest tabel detine adresele de rutine pregatite pentru a procesa intreruperi specifice.
- Numarul de posibile rutine de tratare a intreruperilor depaseste inca intervalul de numere definite pentru intreruperi, astfel incat mai multe rutine pot avea intreruperi inlantuite. Efectiv, adresele continute in vectorii de intrerupere sunt indici (pointeri) catre liste inlantuite de rutine de intrerupere.

Urmatoarea figura prezinta un vector de intrerupere Intel Pentium. intreruperile de la 0 la 31 nu sunt mascat si sunt rezervate pentru erori grave de hardware si alte erori. intreruperile mascate, inclusiv intreruperile normale de dispozitive I / O incep de la intreruperea 32.[29]

Componentele hardware moderne detin suport pentru nivele de prioritate pentru intreruperi, ceea ce permite sistemului sa mascheze intreruperile de prioritate mica, in timp ce servesc intreruperile de prioritate mare, sau sa permita unui semnal cu prioritate mare sa intrerupa procesarea unuia cu prioritate mica.[29]



In momentul de boot al sistemului se determina care sunt dispozitivele prezente, si se incarca adresele rutinei corespunzatoare in tabelul de intreruperi. In timpul functionarii, dispozitivele semnaleaza erori sau finalizarea comenzilor prin intreruperi. Exceptiile, cum ar fi impartirea la zero, accesul invalid la memorie, sau incercarile de a accesa instructiuni in modul kernel, pot fi semnalizate prin intreruperi. Timp felierea si context switch-uri pot fi, de asemenea, puse in aplicare prin mecanismul de intrerupere.[29]

Apelurile de sistem sunt implementate prin intreruperi software. Cand un program necesita procesare in mod kernel, stabileste informatii de comanda si, eventual, adrese de date in anumite registre, iar apoi ridica o intrerupere software. (De exemplu, 21 hex in DOS.) Sistemul salveaza starea si apoi cheama rutina de intrerupere corespunzatoare pentru a procesa cererea in modul kernel. Intreruperile software au, in general, prioritate scazuta, deoarece acestea nu sunt la fel de urgente precum dispozitivele cu spatiu limitat de tamponare. intreruperile sunt, de asemenea, folosite pentru a controla operatiunile kernel, si sa programeze activitati pentru performante optime. De exemplu, finalizarea unei operatiuni de citire de pe disc implica doua intreruperi:[29]

1. O intrerupere de mare prioritate ce recunoaste finalizarea actiunii asupra dispozitivului, si emite urmatoare cerere de disc, astfel ca hardware-ul nu sta inactiv.
2. O intrerupere de prioritate inferioara transfera datele din spatiul de memorie kernel in spatiul utilizator, si apoi transfera procesul din coada de asteptare la coada ready.

Sistemul de operare Solaris foloseste un kernel multi-thread si threaduri cu prioritate pentru a atribui threaduri diferite pentru diferite rutine de intrerupere. Acest lucru permite manipularea "simultana" a mai multor intreruperi, si asigura ca intreruperile de inalta prioritate vor avea prioritate in fata celor cu prioritate scazuta si peste procesele utilizatorului.[27]

4.5. Drivele pentru dispozitivele 3D

Drivele pentru dispozitivele de intrare-iesire sunt programe de calculator ce opereaza si controleaza un tip particular de dispozitiv ce este atasat unui system. Un driver 3D ofera o interfata software unui dispozitiv hardware 3D, ce ii permite sistemului de operare sa acceseze functiile dispozitivului fara a cunoaste detaliile specifice ale dispozitivului folosit.[22]

Drivele pentru tehnologia 3D sunt in plina cercetare, dezvoltare si standardizare pe masura dezvoltarii tehnologiei 3D.



[14]

Cele mai comune driver de afisare 3D sunt cele pentru dispozitivele de iesire 3D, cum sunt monitoarele, proiectoarele sau dispozitivele head-mounted (HMD).

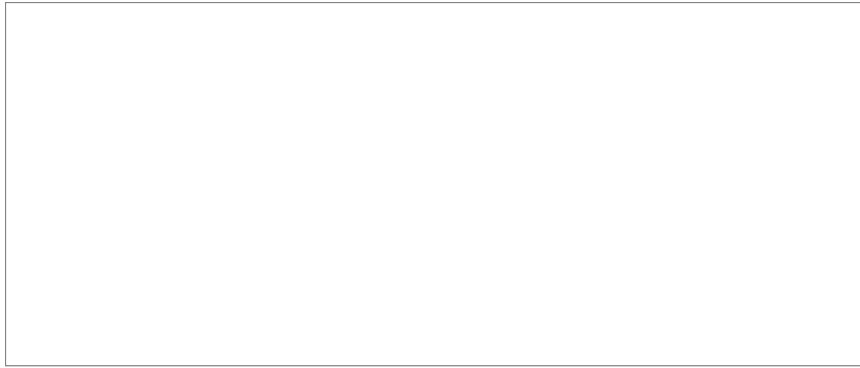
Un driver de afisare 3D ce foloseste ca dispozitiv de iesire HMD creeaza imaginea pentru utilizator in diferite culori pentru fiecare ochi: rosu pentru ochiul stang, cyan pentru ochiul drept. Sursa de semnal 3D ofera astfel o imagine compusa din imaginea stanga transmisa intr-o nuanta, iar cea dreapta transmisa in alta nuanta.[14]

Redarea continutului stereoscopic 3D este activata numai pe sistemele care au toate componentele care sunt stereoscopic 3D-capabile. Aceste componente includ hardware capabil de redare continut vizual 3D, componente hardware grafice, periferice, precum si aplicatii software. Designul stereo din stiva grafica este astfel incat vizualizarea particulara sau tehnologia de afisare care este utilizata, este agnostica pentru sistemul de operare. Driverul comunica direct cu dispozitivul de afisare si are cunostinte despre capacitatile de afisare prin structura standardizata Extended Display Identification Data (EDID). Driverul enumera capabilitati stereo numai cand recunoaste ca un astfel de dispozitiv de afisare este conectat la sistem.[29]

Pentru a pune in aplicare capabilitatile stereo din miniportul de afisare si driverele in user-mode, se folosesc dispozitive DDI:[29]

- D3D11DDIARG_CREATERESOURCE
- D3DDDI_ALLOCATIONINFO
- D3DKMDT_VIDPN_SOURCE_MODE_TYPE
- D3DKMT_PRESENTFLAGS
- DXGI_DDI_ARG_ROTATE_RESOURCE_IDENTITIES
- DXGK_PRESENTFLAGS
- DXGK_SETVIDPNSOURCEADDRESS_FLAGS
- **DXGKARG_OPENALLOCATION**

Setarea de afisare stereoscopica se poate face din panoul de control al rezolutiei ecranului, dupa cum se arata aici:



5. Provocari ale tehnologiei 3D

Interfetele 3D dobândesc noi posibilitati, noile tehnici de interactiune permit o interactiune mai rapida comparativ cu interfetele standard. In consecinta, apar noi provocari si probleme. Incorporarea informatiei vizuale intr-un HUD integrat in parbrizul masinii este un exemplu de folosire a interfetelor 3D pentru sarcini critice in timp.[15] Aceasta poate oferi suport dar poate si conduce catre un fenomen psihologic de distragere a atentiei.

5.1. Supraincercarea informationala

Supraincercarea informationala se refera la starea de a detine prea multa informatie la un interval de timp, astfel incat utilizatorul nu poate percepe informatiile importante. De exemplu, in dispozitivele de afisare Head-Up integrate intr-o masina, ce ofera informatii vizuale semnificate utilizatorului, cum sunt limitarile de viteza sau informatiile privind distanta de siguranta fata de o alta masina.

5.2. Orbirea la schimbare

Orbirea la schimbare se refera la esecul in detectarea a ceea ce ar trebui sa fie o schimbare evidenta. Aceasta apare in interactiunile om-masina atunci cand intr-un dispozitiv de afisare exista mai mult de o schimbare in acelasi timp iar atentia utilizatorului este tulburata. Utilizatorul trebuie sa memoreze starea inainte de schimbare si trebuie sa detecteze noua schimbare pentru a recunoaste schimbarea ce a avut deja loc. O incercare de a reduce posibilitatea orbirii la schimbare este de a face schimbarile explicit.

5.3. Blocarea si perceptia in adancime

Dispozitivul HUD integrat in parbrizul masinii ofera nu doar posibilitatea de a prezenta informatie simbolica, ci si prezentarea sagetilor de navigatie simbolice in adancimea corespunzatoare, astfel incat acestea sunt aliniate cu strada si se comporta ca obiecte integrate in mediul real. Aceasta abordare se afla inca in stadiul de cercetare deoarece un astfel de sistem de navigatie este foarte intuitiv, iar utilizatorul nu detine posibilitatea de a transforma mental perspectiva sa in sistemele computationale.[15]

5.4. Tunelare perceptuala

In afara de supraincarcarea infomationala, schemele animate de semnalare integrate in dispozitivul HUD al masinii poate conduce la tunelarea perceptuala. Acest fenomen este recunoscut in domeniul aviatiei si se refera la starea in care utilizatorul este concentrat asupra unui stimul specific, cum ar fi un semnal de alarma clipitor, si neglijeaza acordarea atentiei altor sarcini importante sau informationale cum ar fi condusul, zburatul sau traficul. Utilizatorul este de obicei atras de un stimul specific, dar nu consuma o incarcare cognitiva asupra stimulului, ci pur si simplu il priveste fara a gandi.[23] Dar atractia vizuala se poate schimba usor in atractie cognitiva si poate duce astfel la un alt fenomen denumit captura cognitiva.

5.5. Captura cognitiva

Captura cognitiva se refera la situatia in care utilizatorul poate fi distras complet ceea ce poate duce la o pierdere totala a constiintei situationale. Acest efect este cunoscut in situatiile zilnice. De exemplu, cand o persona intr-un cinematograful se uita dupa un loc liber si este pierdut in sarcina lui, incatt nu isi vede prietenul care incearca sa ii atraga atentia. In acest caz, tunelarea perceptuala nu poate cauza consecinte importante, insa in situatiile critice de timp, poate cauza efecte primejdioase.[21]

6. Concluzii

Interfetele 3D castiga din ce in ce mai multe capabilitati si astfel ofera o interactiune mai rapida si intuitiva comparativ cu interfetele standard. Noi dispozitive, tehnici si metafore promit noi posibilitati pentru interactiunile 3D om-masina, in mod special pentru interactiunile critice de timp. Faptul de a nu detine standarde ridica probleme majore. Design-ul interfetelor 3D este un domeniu interdisciplinar de cercetare, deoarece nu doar aspectele tehnice sunt importante ci si situatiile fizice si psihologice ale utilizatorului trebuie luate in considerare. Astfel, cunostintele despre perceptie, cunoastere, lingvistica, factori umani, etnografie, design grafic si multe altele sunt foarte importante. Mai mult, pentru a evita fenomenele cognitive, testele de utilitate sunt indispensabile.

In alegerea dispozitivelor de intrare-iesire 3D pentru crearea de sisteme si aplicatii cu medii virtuale, factorul monetar este foarte important. Insa, cele mai scumpe dispozitive de intrare-iesire nu garanteaza si folosirea corespunzatoare a acestora. Selectia acestora trebuie sa depinda de modul in care utilizatorul va folosi dispozitivul si de tipul de interactiune necesar. Mai mult, nici unul din dispozitivele de intrare-iesire prezentate nu sunt perfectionate, majoritatea inca se afla in stadiul de cercetare. Creativitatea este un factor major in dezvoltarea dispozitivelor de intrare-iesire 3D. Astfel, daca dispozitivul de intrare-iesire 3D necesar utilizator nu exista intr-o forma comerciala, cea mai buna modalitate de achizitionare a lui este construirea lui.

7. Bibliografie

1. S. Lipschutz, M. Lipson, *Linear Algebra (Schaum's Outlines)* (4th ed.), ed. McGraw Hill, 2009, p.22
2. Megumi Iizuka, „Welcome to the wonderful world of 3D”, *Optics and Photonics News*, Vol. 18, Nr. 4, aprilie 2007, p. 8
3. Kaufman, Lloyd. *Sight and Mind*, New York: Oxford University Press, 1974, p.139–141
4. Posted on *Understanding Requirements for High-Quality 3D Video: A Test in Stereo Perception*, 3droundabout.com, 19 December 2011
5. Bowman, D., Kruijff, E., LaViola, J., Poupyrev, *An Introduction to 3-D User Interface Design*, Presence, 10(1), 2001, February, p. 96–108
6. Yoder, Lars. *The Digital Display Technology of the Future*, INFOCOMM'97, Iulie 1997
7. Lewis, J., K. Potosnak & R. Magyar, *Keys and Keyboards. Handbook of Human-Computer Interaction*, Amsterdam, North-Holland, 1997, p. 1285-1316
8. D.A. Bowman, E. Kruij, J. J. LaViola, I. Poupyrev, *3D User Interfaces: Theory and Practice*, Addison Wesley, 2004
9. MacKenzie, I. S., *Input Devices and Interaction Techniques for Advanced Computing. Virtual environments and advanced interface design*, Oxford, UK, Oxford University Press, 1995, p. 437-470
10. Keefe, D., Acevedo, D., Moscovich, T., Laidlaw, D., LaViola, J., *CavePainting: A Fully Immersive 3D Artistic Medium and Interactive Experience*, Proceedings of the 2001 Symposium on Interactive 3D Graphics, 2001, p. 85-93
11. Sturman, D.J., Zeltzer, D., "A survey of glove-based input". *IEEE Computer Graphics and Applications*, January 1994 p. 30–39
12. Jorgensen, Charles, Kevin Wheeler, Slawomir Stepniewski, *Bioelectric Control of a 757 Class High Fidelity Aircraft Simulation*, <http://ic.arc.nasa.gov/publications/index.html>, 1999
13. Cruz-Neira, Carolina, Daniel Sandin, Thomas Defanti. *Surround-Screen*

- Projection-Based Virtual Reality: The Design and Implementation of the CAVE*, SIGGRAPH 1993, p. 135-142
14. Melzer and Moffitt, *Head Mounted Displays: Designing for the user*, McGraw Hill, 1997
 15. Spitzer, Cary R., *Head-Up Displays*, ed. Digital Avionics Handbook, FL: CRC Press, 2001
 16. Michael Tidwell, Richard S. Johnston, David Melville, Thomas A. Furness III, *The Virtual Retinal Display – A Retinal Scanning Imaging System*, Ph.D. Human Interface Technology Laboratory, University of Washington
 17. Begault, Durand R., *3D Sound For Virtual Reality and Multimedia*, ed. Academic Press, 1994
 18. Burdea, Grigore C., *Force and Touch Feedback for Virtual Reality*, ed. Wiley Interscience, 1996
 19. LaViola, Joseph, Robert Zeleznik, *Flex and Pinch: A Case Study of Whole-Hand Input Design for Virtual Environment Interaction*, Proceedings of the IASTED International Conference on Computer Graphics and Imaging 1999, p. 221-225
 20. Youngblut, C. R.E. Johnson, S.H. Nash, R.A. Wienclaw, C.A. Will, *Review of Virtual Environment Interface Technology*, Technical Report IDA Paper P-3186, Log:H96-001239. Institute for Defense Analysis. 1996
 21. Shamus P. Smith, Jonathan Hart, *Evaluating Distributed Cognitive Resources for Wayfinding in a Desktop Virtual Environment 3D User Interfaces (3DUI'06)*, 2006
 22. *What is a device driver?*, *The purpose of device drivers*, PC Gesund, Retrieved November 8, 2012
 23. Carolina Cruz-Niera, *Applied Virtual Reality*, Course #14, Siggraph 1998
 24. EMC Education Services, *Information Storage and Management: Storing, Managing, and Protecting Digital Information*, John Wiley & Sons, 2010
 25. The Java Tutorials, *What Is an Interface*, Oracle, 2012-05-01
 26. Burke, Timothy, *Writing device drivers: tutorial and reference*, Digital Press, 1995
 27. Microsoft, *User-mode vs. Kernel-mode Drivers*, 2003-03-01
 28. Microsoft, *Introduction to the User-Mode Driver Framework (UMDF)*, 2006-10-10
 29. Deborah Morley, *Understanding Computers 2009: Today and Tomorrow*, Cengage Learning, 2009
 30. Abraham Silberschatz, Greg Gagne, Peter Baer Galvin, *Operating System Concepts*, Editia 8, Capitolul 13
 31. *Exploring the 3D printing opportunity*, The Financial Times, 2012-08-30