

Indice

Introduzione	3
1 La sindrome <i>Locked-in</i>	5
1.1 Descrizione della sindrome <i>locked-in</i>	5
1.2 Il problema della comunicazione	7
1.3 Uso del computer e controllo ambientale	9
1.4 Alcuni casi particolari	10
2 Sistemi di comunicazione	13
2.1 Il problema dell' <i>eye-tracking</i>	13
2.2 Tecniche di <i>Eye-Tracking</i>	15
2.3 Tecniche basate sulla luce riflessa	17
2.3.1 Limbus Tracking	18
2.3.2 Pupil Tracking	18
2.3.3 Corneal and Pupil Reflection Relationship	19
2.3.4 Corneal Reflection and Eye image using an Artificial Neural Network	20
2.3.5 Purkinje Image Tracking	21
2.4 Una tecnica basata sul potenziale elettrico della pelle	22
2.5 Tecniche basate sull'utilizzo di lenti a contatto	22
2.6 Confronto tra le Tecniche di Tracking	23
2.7 EagleEyes Project	24
2.8 Eyegaze Communication System	26
2.9 Il Metodo di Zhu-Yang	27

2.10	Accuratezza del Gaze Tracking	29
2.10.1	Accuratezza ottimale	29
2.10.2	Barriere Tecniche	30
2.10.3	La scelta dell'approccio	31
2.11	Gaze Tracking con accuratezza di livello subpixel	33
2.11.1	Subpixel eye corner tracking	33
2.11.2	Ricerca del bordo e del centro dell'iride	35
2.11.3	Stima della direzione dello sguardo	38
2.12	Risultati sperimentali	39
3	Il Metodo di Kothari-Mitchell	43
3.1	La matrice dei <i>Bins</i>	43
3.2	Analisi spaziale dei <i>Bins</i>	46
3.3	Analisi temporale	46
3.4	Estensione del metodo	49
4	Implementazione del Primo Metodo di <i>Kothari-Mitchell</i>	51
4.1	Acquisizione dati	51
4.2	Costruzione della matrice dei <i>Bins</i>	52
4.3	Selezione dei <i>bins</i>	55
4.4	Analisi del lavoro	57
4.5	Codice	58
	Conclusioni	63
	Bibliografia	65
	Ringraziamenti	69

Introduzione

Durante i mesi di servizio civile svolti presso l'AIAS di Bologna (Associazione Italiana Assistenza Spastici) sono stato colpito da alcuni dei progetti sviluppati dall'Ausilioteca, in merito a sistemi di comunicazione per persone disabili. In particolare era incuriosito dall'esperienza dell'Ing. Massimiliano Malavasi, che ha sviluppato un sistema di comunicazione per un particolarissimo utente affetto da sindrome *Locked-in* (paralisi completa dei muscoli volontari esclusi, in genere, quelli che controllano il movimento dell'occhio): si tratta di un sistema di monitoraggio del movimento dell'occhio dell'utente collegato ad un software a scansione di accesso ai dati. Da qui l'idea di continuare le ricerche in merito all'*eye tracking* e l'*eye detection* per cercare delle soluzioni più efficienti e moderne ed adattabili a molteplici contesti. In questa tesi ho raccolto alcune delle tecniche più importanti di *eye-tracking* e implementato un algoritmo di *eye-detection*. Con *Eye-tracking* si intendono quei sistemi in grado di localizzare e seguire lo sguardo dell'utente. Il sistema ha molteplici utilizzi, tra i quali spiccano l'adattamento a sistemi di puntamento sviluppati per l'aviazione militare e *mouse pointer* guidati dallo sguardo. L'*eye-detection* è invece quel processo che permette di localizzare gli occhi all'interno di sequenze video o singole immagini. Di particolare importanza sono il *Metodo di Zhu-Yang* che permette di tracciare la direzione dello sguardo con la precisione del sub-pixel e il *Metodo di Kothari-Mitchell* per la localizzazione degli occhi. Ho implementato il secondo metodo usando come ambiente di lavoro il Matlab, utilizzando come input delle semplici immagini.

Il primo capitolo descrive in maniera accurata la sindrome *locked-in*, evidenziando le problematiche relative alla comunicazione e l'interazione con l'ambiente. Nel secondo capitolo ho raccolto alcuni risultati in merito all'*eye-tracking*, cioè tecniche sperimentate, sistemi presenti in commercio ed ho descritto il Metodo di *Zhu-Yang*. Nel terzo capitolo ho presentato il Metodo di *Kothari-Mitchell* per la localizzazione dello sguardo. Nel quarto ho descritto i punti salienti dell'implementazione del Metodo di *Kothari-Mitchell*.

Capitolo 1

La sindrome *Locked-in*

“La vista e l’udito erano i due soli sensi che animassero ancora, come due scintille, quella materia umana già più morta che viva; e anche di questi due sensi, uno solo rivelava la vita che animava internamente quella statua: la vista; e il suo sguardo somigliava perciò a uno di quei lumicini lontani che, durante la notte, dimostrano al viaggiatore sperduto che vi è ancora un essere vivente che veglia in quella oscurità.”

Alexandre Dumas, *Il conte di Montecristo*, 1845

1.1 Descrizione della sindrome *locked-in*

Con il termine sindrome *locked-in* (letteralmente *chiuso dentro*) si definisce l’insieme dei sintomi di una rara patologia neurologica caratterizzata dalla paralisi completa dei muscoli volontari esclusi, in genere, quelli che controllano il movimento dell’occhio.

Le patologie che la possono provocare sono diverse (trombosi dell’arteria basilare, emorragia, tumore oppure mielinolisi centrale del ponte, lesioni bilaterali della porzione ventrale del mesencefalo, infarti bilaterali della capsula interna, miastenia grave, sclerosi laterale amiotrofica, overdose da farmaci) ma si manifesta solo quando il danno colpisce in modo selettivo i neuroni del moto, lasciando invece indenne la sostanza reticolare ascendente.

Le persone che presentano la sindrome *locked-in* sono coscienti e mantengono

un livello molto elevato di funzioni cognitive, ma perdono la funzione vocale e la possibilità di compiere alcun gesto; solo nei casi più fortunati mantengono il controllo del movimento degli occhi e delle palpebre.

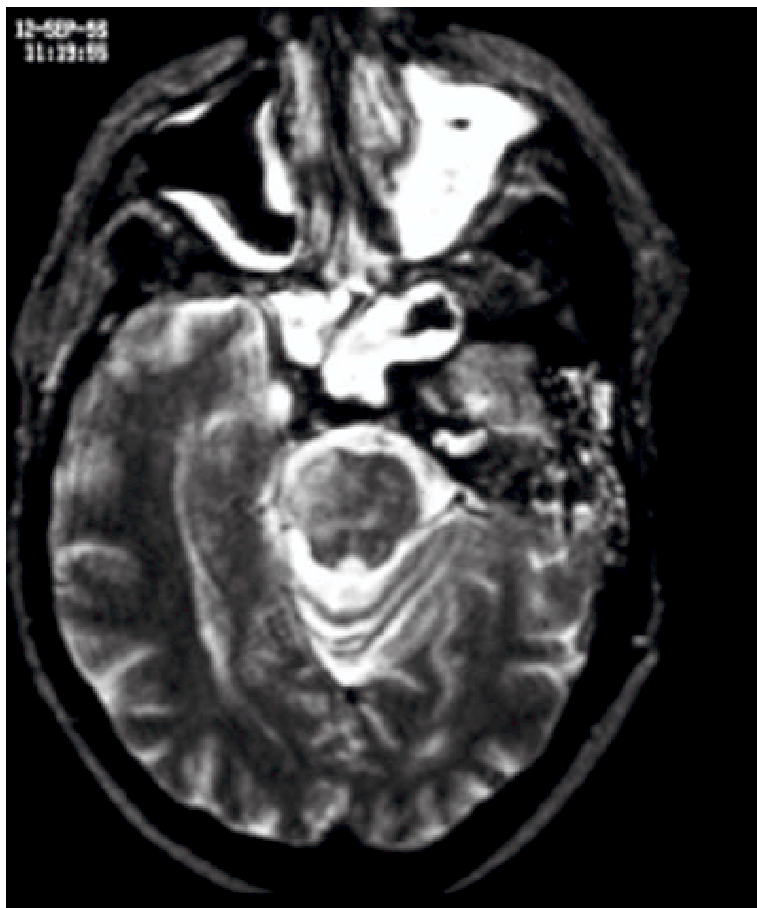


Figura 1.1: Trombosi dell'arteria basilare

Per alcuni pazienti questa sindrome può essere uno stato transitorio nella fase acuta post traumatica o post malattia, ma per altri rappresenta uno stato cronico. In questo caso non ci sono cure per la sindrome *locked-in* e nemmeno procedure standard per il trattamento dei pazienti che la manifestano.

L'esistenza di un certo numero di pazienti con questa sindrome è una situazione recente perché, fino ad alcuni anni fa, difficilmente si era in grado di

mantenere in vita persone che presentassero un quadro clinico così grave. Oggi le moderne tecniche di terapia intensiva consentono in questi casi maggiori prospettive di vita.

I pazienti *locked-in* necessitano di un trattamento sanitario continuo: la situazione di immobilità richiede periodiche aspirazioni, mediante canula, nell'apparato respiratorio, un monitoraggio continuo di alcune funzioni vitali, per esempio attraverso l'uso di un ossimetro, un'alimentazione artificiale tramite sonda, più tutte le cure necessarie ad un paziente immobilizzato.

Un problema molto grave, in Italia come in altri paesi, è la mancanza di strutture adibite ad accogliere queste persone; una volta finita l'emergenza sanitaria, spesso infatti le famiglie non sanno dove portare il paziente dopo la dimissione dall'ospedale. Oggi si è in grado di gestire queste persone anche in ambiente domestico, ma con costi e problematiche enormi, non ultima, tra queste, la necessità di avere a disposizione un medico in grado di intervenire in pochissimo tempo per gestire un'eventuale emergenza sanitaria.

L'esistenza di questi casi rende necessario trovare all'interno della società risorse da dedicare a queste situazioni che, aumentando di numero, sono sempre meno gestibili come singole emergenze.

1.2 Il problema della comunicazione

Per soggetti *locked-in*, contemporaneamente alla esigenze sanitarie, è fondamentale risolvere il problema della comunicazione, rompendo il muro di difficoltà che le tiene *chiuse dentro*.

Purtroppo è una barriera difficile da abbattere: l'impossibilità di parlare, di muoversi e la mancanza di mimica facciale tolgono all'interlocutore la possibilità di avere qualche indizio sul reale stato della persona *locked-in*, a meno che non sia questa a comunicarglielo espressamente, tramite un codice, attraverso i movimenti dell'occhio.

Il codice utilizzato per la comunicazione è normalmente di tipo sì/no (per esempio apertura dell'occhio = sì, non apertura o doppia apertura = No).

Usare codici più complessi, in questi casi, è risultato, nell'esperienza dell'*Ausilioteca* di Bologna, poco pratico: si potrebbe pensare, per migliorare la velocità di comunicazione, di fare corrispondere a varie posizioni dell'occhio altrettanti significati, ma ciò richiederebbe una grande concentrazione sia da parte della persona *locked-in*, sia da parte dell'interlocutore e comporterebbe una probabilità di fraintendimento così elevata da rendere ancora più difficile e lento il comunicare.

Grazie all'uso del *sì* e del *no* si possono comunque costruire sistemi di comunicazione complessi che utilizzano il metodo della scansione. Si basa essenzialmente sulla proposta in successione (sia da parte di un interlocutore umano, sia da parte di un computer) di scelte che la persona può confermare o no tramite il movimento dell'occhio. Se, per esempio, le scelte che vengono proposte sono le lettere dell'alfabeto, la persona *locked-in* può comunicare, anche se lentamente, qualunque cosa.

Un problema, in questi casi, è la forte tentazione per l'interlocutore di proporre frasi o domande a cui il paziente può rispondere solo sì o no rinunciando all'uso della scansione. Così facendo la comunicazione risulta troppo guidata e in massima parte unilaterale, permettendo al paziente solo di confermare o no le frasi che gli vengono proposte. Inoltre non è facile per persone di passaggio o non abituate usare immediatamente il sistema della scansione in modo efficiente e questo rende problematica la comunicazione in questi casi. Un altro grande problema è quello relativo al richiamo dell'attenzione: infatti se la persona *locked-in* ha necessità di comunicare qualcosa, deve comunque attendere che sia l'interlocutore a chiederlo o ad accorgersene attraverso il movimento dell'occhio. E' quindi importante rendere disponibile un dispositivo elettronico, controllabile dalla persona disabile, in grado di richiamare l'attenzione.

1.3 Uso del computer e controllo ambientale

Un passo molto importante per accrescere la qualità di vita delle persone *locked-in* è dotarle di un accesso ad ausili elettronici che permettano di richiamare l'attenzione in caso di necessità, di utilizzare un computer e di effettuare un controllo su una parte dell'ambiente circostante.

Il punto critico, nel caso di questi pazienti, è costituito dal sensore in grado di rilevare correttamente il movimento volontario dell'occhio; per effettuare le altre funzioni sopra citate esistono già efficaci e collaudati sistemi in grado di funzionare efficacemente, una volta sia disponibile il segnale proveniente da un sensore esterno.

Oggetto di questa tesi è la ricerca di tecniche per rilevare i movimenti della regione oculare tramite telecamera.

Un'alternativa che è stata sperimentata è l'uso di sensori posti sull'occhio come, ad esempio, testine all'infrarosso in grado di intercettare il movimento della palpebra e delle ciglia. Questi sistemi aggiungono però ulteriori oggetti sul corpo di un paziente che è già sottoposto a cateterismo e monitoraggio e sono in genere molto sensibili al modo con cui vengono posizionati: bastano piccoli movimenti per renderli inefficaci ed è quindi necessario un frequente intervento di personale in grado di posizionarli correttamente. Perciò questa soluzione presenta un elevato livello di criticità, soprattutto in ambito ospedaliero.

Nel mondo sono attualmente in sperimentazione sistemi invasivi basati su elettrodi per consentire a persone *locked-in* di comandare dispositivi elettronici (The Thought Translation Device). Nel caso in cui invece il paziente abbia conservato anche solo un unico piccolo movimento volontario (bastano pochi mm) sono invece disponibili sensori meccanici di vario tipo in grado di rilevarlo correttamente.

Qualunque sia il metodo utilizzato, una volta che un sensore renda disponibile un segnale generato volontariamente dal paziente, è possibile realizzare sistemi che consentano all'utente *locked-in* una certa possibilità di controllo dell'ambiente circostante. Attraverso sistemi di accesso facilitato al compu-

ter è possibile utilizzare un PC standard, mentre con telecomandi adattati o interfacce di controllo reperibili in commercio si possono comandare una vasta gamma di apparecchiature come ad esempio: campanelli o radio richiami a distanza, TV, stereo, luci, tapparelle o finestre motorizzate ecc.

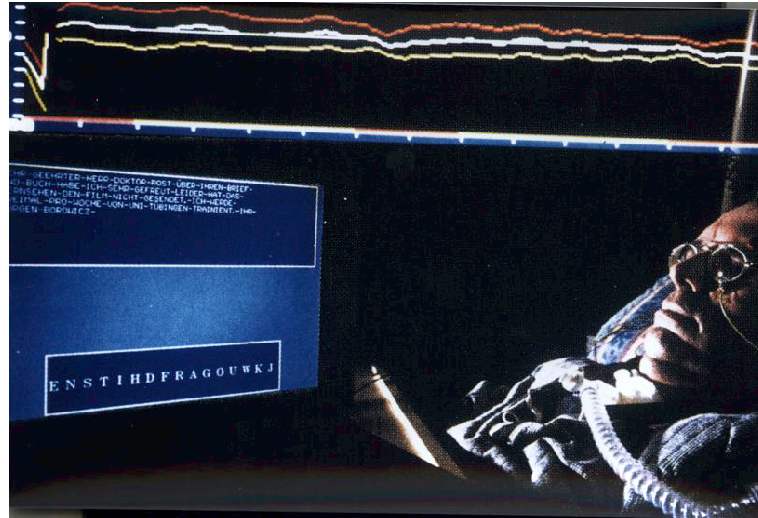


Figura 1.2: Paziente *locked-in* scrive usando il TTD (The Thought Translation Device)

1.4 Alcuni casi particolari

Casi estremi ma molto importanti sono rappresentati da Kurt e Huns, due pazienti di Niels Birbaumer, ricercatore del Dipartimento di psicologia medica e neurologica comportamentale dell'Università di Tubinga, in Germania. I due soggetti affetti dalla sindrome *locked-in* non controllano il movimento degli occhi ma riescono ugualmente a comunicare: Birbaumer ha sfruttato le più moderne tecniche neurofisiologiche e informatiche per trasformare le capacità motorie residue in un codice di comunicazione.

Il tracciato elettroencefalografico riflette la condizione di veglia del malato, e non solo: quando decide di compiere un gesto, tutto il suo corpo rimane immobile, ma gli elettrodi segnalano una variazione dell'ampiezza dei poten-

ziali corticali. Sfruttando questo residuo di movimento che non si traduce più in una variazione di posizione dei segmenti corporei nello spazio, ma solo in un evento elettrico, ha restituito ai due malati la capacità di comunicare. Il buon livello cognitivo ha permesso ai due pazienti di imparare a modificare il loro livello corticale per almeno due secondi; in questo modo il calcolatore, collegato ai due con degli elettrodi, segnalava l'aumento di ampiezza del potenziale attraverso la comparsa di un quadratino luminoso sulla parte inferiore dello schermo, e la riduzione con quadratino luminoso su quella inferiore. Questo perché l'idea iniziale dei ricercatori era quella di mettere a punto un codice assenso-dissenso in cui un aumento del potenziale corrispondesse ad una risposta affermativa e una riduzione ad una negativa.

Dopo una prima fase di sperimentazione si è constatato che entrambi i pazienti imparavano con facilità a comunicare solo le risposte di assenso. Per questo motivo il progetto è stato modificato introducendo un nuovo codice: sullo schermo del computer compariva metà alfabeto, se in esso era contenuta la lettera che il malato intendeva pronunciare egli esprimeva il suo assenso attraverso un aumento dell'ampiezza del potenziale corticale; allora il computer suddivideva la metà dell'alfabeto in due ulteriori metà, e così via fino a quando era possibile individuare la lettera corretta. Si ripeteva poi la stessa procedura per le lettere seguenti, fino a formare una parola. Kurt e Hans hanno reimparato a parlare in circa quarantacinque giorni, fino a riuscire ad individuare in un minuto due caratteri.

Non sono i soli malati di *locked-in* ad essere usciti dal loro isolamento. Per esempio Severino Mingroni è riuscito a scrivere una lettera a Repubblica in cinque giorni di sforzi. L'impresa è stata possibile grazie a un mouse speciale fornitogli da una ditta torinese: l'ausilio, unito a un apposito software, gli permette di digitare attraverso gli unici movimenti residui, quelli del capo.

Un altro caso che desta curiosità è quello di Jean Dominique Bauby, caporedattore del settimanale femminile Elle, che nel 1995 venne colpito da un ictus e, risvegliatosi dal coma, a causa della sindrome *locked-in* riesce a muovere solo la palpebra sinistra. Grazie al battito di ciglia il giornalista è stato in

grado di scrivere il libro “Lo Scafandro e la Farfalla”. “Nella mente - racconta - mescolo dieci volte ogni frase, tolgo una parola, aggiungo un aggettivo e imparo il testo a memoria. Al mattino mi si sgrana l’alfabeto finché con un battito di ciglia fermo il mio interlocutore sulla lettera che deve annotare”. Secondo Bauby il sistema di comunicazione impedisce risposte incisive: “La battuta più sottile si smorza e si spegne quando ci vogliono minuti per comporla. Ciò toglie alla conversazione la patina di argento vivo, e io annovero questa mancanza di humour tra gli inconvenienti del mio stato”.¹



Figura 1.3: Deborah Lanino, *Locked-in*

¹BAUBY J.D., *Lo scafandro e la farfalla*, Ponte alle grazie, Milano, 1997

Capitolo 2

Sistemi di comunicazione

In questo capitolo verrà trattato uno dei sistemi che permette alle persone affette da sindrome *Locked-in* di comunicare attraverso l'utilizzo del calcolatore: si tratta di sensori che rilevano il movimento dell'occhio e ne interpretano il significato. Questi sistemi si basano sull'*eye-tracking*, cioè l'*inseguimento* dell'occhio e dello sguardo.

Negli ultimi paragrafi verrà descritto il *Metodo di Zhu-Yang*, ovvero un approccio di tipo analitico al problema dell'*eye-tracking*

2.1 Il problema dell'*eye-tracking*

Ogni giorno usiamo i nostri occhi intensamente per una larga varietà di scopi: leggere, guardare spettacoli, raccogliere informazioni, pianificare le nostre azioni, percepire e apprendere nuove cose, guidare, ecc. Normalmente non ci accorgiamo del ruolo fondamentale che i nostri occhi hanno nei processi percettivi e quali vaste quantità di informazioni analizzano, ma siamo soliti pensare al nostro sistema visivo come “trasparente”: ad esempio, basti pensare alla possibilità di soffermarci su piccoli particolari, lasciando al nostro sistema visivo il compito di analizzare la mole di informazione che ricevono.

Questo non è un compito banale; gli impulsi luminosi dell'ambiente devono

essere continuamente campionati, integrati e immagazzinati nella conoscenza. A questo scopo i nostri occhi sono costantemente in movimento (così come la nostra testa e il corpo) per analizzare l'ambiente circostante. Siamo soliti pensare ai nostri occhi soprattutto come organismi di input, che si limitano a osservare quello che ci circonda, ma effettivamente operano anche come organismi di output. L'output che sono in grado di produrre è la *direzione*: nel processo di campionamento dell'apparato ottico, gli occhi sono puntati in una direzione, che indica quello che stanno focalizzando.

Sin dal 1936, Mowrer riuscì a fare delle registrazioni automatiche dell'orientazione degli occhi nella testa, e quindi della direzione dello sguardo (cfr. Scott & Findlay 1993¹). Le tecniche usate si sono gradualmente migliorate durante il XX secolo, ma recentemente hanno raggiunto sufficienti livelli di non invasività sul soggetto, tanto da essere usati anche nell'ambiente esterno. In particolare, usando una videocamera si può tracciare lo sguardo dell'occhio anche da distanza non ravvicinata.

Attraverso la possibilità di tracciare la direzione dello sguardo dell'utente è nata una ricerca parallela che vuole valorizzare la comunicazione con il calcolatore. Dall'avvento dei Personal Computer per uso domestico, sono stati fatti passi avanti soprattutto nell'interfaccia che lega il PC all'utente (presentazione grafica dei dati, finestre di sistema e l'uso dei suoni per presentare eventi), mentre il rapporto inverso (nella maggior parte dei casi) è rimasto confinato all'uso di tastiere, joysticks e mouse, strumenti attivati manualmente. Attraverso il tracking della direzione dello sguardo dell'utente, la banda della comunicazione dall'utente al PC - che è la potenziale quantità di informazione da trasferire - può essere accresciuta usando le informazioni che riguardano appunto quello che sta guardando l'utente, e la progettazione di *trackers*. Questo è solo l'inizio di un dibattito sull'incremento del rapporto tra utente e PC: attraverso il monitoraggio dell'intero utente il PC potrebbe reagire a tutti i tipi di gesti; un esempio è rappresentato da strumenti di

¹Scott, D. & Findlay (1993), J. M, Visual search, eye movements and display units, Human factors report, University of Durham, South Road, Durham DH1 3LE, UK

riconoscimento vocale esistenti in commercio.

2.2 Tecniche di *Eye-Tracking*

Oggi esistono vari modi per tracciare la direzione dello sguardo. Nessuna di queste tecniche è perfetta nel senso che non una singola tecnica riesce a soddisfare pienamente tutte le richieste di utilizzo proposte da Scott & Findlay (1993). L'ideale mezzo di tracking deve:

1. Offrire un campo visivo non ostruito con buon accesso al volto e alla testa;
2. Non avere contatto con il soggetto;
3. Andare in contro alla sfida pratica di essere in grado di stabilizzare artificialmente l'immagine della retina, se necessario;
4. Possedere un *accuratezza* di almeno l'un per cento o pochi minuti di arco; per esempio non approssimare 9° con 10° . L'accuratezza è limitata dagli effetti cumulativi della non linearità, distorsione, rumore ,ritardo e altre origini di errore;
5. Offrire una risoluzione di 1 minuto di arco/sec, e quindi essere in grado di trovare i più piccoli cambiamenti nella posizione dell'occhio; la risoluzione è limitata solo dal rumore strumentale;
6. Offrire un ampio raggio della dinamica di un minuto a 45° per la posizione dell'occhio e un minuto/sec a 800/sec per la velocità dell'occhio;
7. Offrire una buona dinamica temporale e velocità di risposta (buon guadagno e piccola fase vanno da 100Hz, o un buon passo di risposta);
8. Possedere una risposta real-time (permettendo manovre psicologiche);

9. Misurare tutti e tre i gradi della rotazione angolare e essere insensibile alla traslazione oculare;
10. Essere facilmente estendibile alla registrazione binoculare;
11. Essere compatibile con le rilevazioni della testa e del corpo;
12. Essere di semplice utilizzo;

Mentre molti concordano sulla desiderabilità di questi requisiti, notiamo che non sono tutti necessari per rendere accettabile l'interfaccia dell'*eye-gaze tracking*. Per esempio la richiesta (2) non risulta essere cruciale, in quanto spesso la costruzione fisica del mezzo dipende dall'utilizzo; un esempio è dato dai cartellini magnetici di riconoscimento. Bisogna considerare che i sistemi che utilizzano mezzi montati sulla testa del soggetto hanno almeno due vantaggi rispetto ad altre tecniche "a distanza": tracciano esattamente i raggi luminosi che attraversano gli occhi e un *tracker* può ricoprire un'intera stanza. Così, si può potenzialmente usare una videocamera invece di affidarsi ad una serie di trackers a distanza e calcoli geometrici sulle direzione dello sguardo per calcolare le coordinate dell'oggetto che si sta focalizzando. Questa distinzione diventa importante se c'è la possibilità che alcuni oggetti opachi (per esempio parti di fogli) oscurino lo schermo ma non il *tracking device* (vedi fig:2.1). La tecnica di tracking che si fonda sull'*immagine-Purkinje-duale* può ad esempio registrare l'adattamento del fuoco dello sguardo dell'utente, a prescindere da quanto lontano stia guardando, in modo da ridurre il problema dell'interferenza, ma richiede che la testa sia immobile in relazione all'equipaggiamento di misurazione e un modo per farlo è attraverso l'*head-mounting*.

La richiesta (10), pur concedendo una minore libertà di movimento al soggetto, permette ad un occhio di essere chiuso e generalmente produce un maggior numero di dati di tracking accettabili, ma non risulta essere essenziale.

Una classificazione delle tecniche basate sulla non-invasività è la seguente:

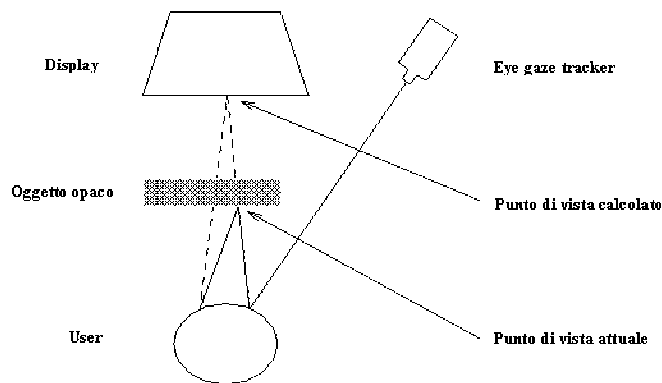


Figura 2.1: Un oggetto opaco oscura lo schermo ma non l'*eye-gaze tracker*, che calcola un errato punto di vista

1. Misurazione del riflesso di alcune luci che si vedono nell'occhio. Tipicamente, sono usate le luci infrarosse per distrarre l'utente il meno possibile, e non avere interferenze da altre sorgenti luminose, come i lampi;
2. Misurazione del potenziale elettrico della pelle intorno agli occhi;
3. Applicazione di una speciale lente a contatto che facilita il tracking della sua posizione;

Tra queste tecniche, la prima include quelle meno invasive, la terza quelle che tipicamente hanno un maggior contatto con l'utente.

Tutte le tecniche richiedono alcune forme di calibrazione prima di essere usate, o a volte anche durante l'utilizzo.

2.3 Tecniche basate sulla luce riflessa

Cinque tecniche di tracking usano la luce (soprattutto la luce infrarossa) riflessa dagli occhi (sulla cornea o più lontano nell'occhio): *limbus tracking*, *corneal tracking and pupil reflection relationship*, *corneal reflection* e *eye image using an artificial neural network and Purkinje image tracking*.

2.3.1 Limbus Tracking

Il *limbus* è il limite tra la sclera bianca e l'iride nero dell'occhio. Dato che la sclera è, normalmente, bianca e l'iride è scuro, questo limite può facilmente essere trovato otticamente e tracciato. Questa tecnica si basa sulla posizione e sulla forma del limbus relative alla testa, che deve essere quindi completamente ferma o l'apparato deve essere fissato sul di essa.

Dato che l'eventualità di coprire il limbus con le palpebre è più o meno rara, "è molto probabile osservare che il tracking del limbus appare come un tracciato orizzontale" (Scott & Findaly 1993). Scott and Findlay affermano anche che il limbus tracking non soddisfa i requisiti (4), (5) e (7) ma sembra possibile migliorare le dinamiche temporali (req. 7) attraverso il raffinamento della tecnica.

2.3.2 Pupil Tracking

La tecnica usata nel *Pupil Tracking* è simile a quella del *Limbus Tracking*, con la differenza che è usato anche il più piccolo confine tra la pupilla e l'iride. Ancora una volta l'apparato deve essere completamente quieto in relazione alla testa.

I vantaggi di questa tecnica rispetto al *Limbus Tracking* sono:

- La pupilla è più lontana dall'essere coperta dalle palpebre rispetto al limbus, che è inefficace nel tracking verticale;
- Il bordo della pupilla è spesso più veloce del limbus e quindi il prodotto ha una risoluzione migliore;

Lo svantaggio è che la differenza di contrasto tra la pupilla e l'iride è più bassa rispetto a quella tra l'iride e la sclera, così trovare il bordo è più difficile.

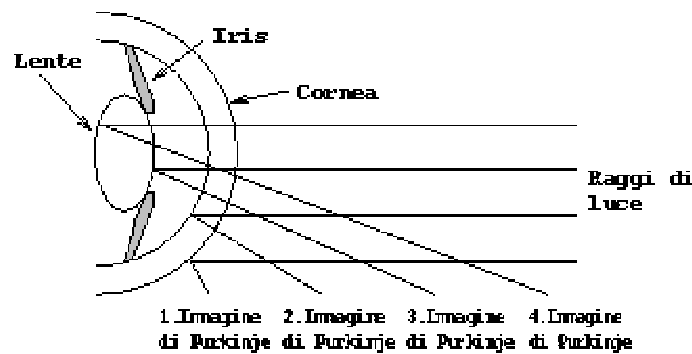


Figura 2.2: Le quattro immagini di Purkinje sono riflessioni della luce entrante sui bordi della lente e della cornea

2.3.3 Corneal and Pupil Reflection Relationship

Quando la luce infrarossa colpisce l'occhio dell'utente, vari riflessi si manifestano sui bordi della lente e della cornea, le così chiamate *immagini di Purkinje* (vedi fig:2.2). La prima immagine di Purkinje, anche chiamata *luccichio*, insieme alla riflessione di luce della retina (*occhio luminoso*), possono essere viste e registrate da una camera all'infrarosso e appaiono rispettivamente come un macchia luminosa e un disco meno luminoso. Quando l'occhio è in posizione orizzontale o verticale, la posizione relativa del *luccichio* e il centro dell'*occhio luminoso* cambiano di conseguenza, e la direzione dello sguardo può essere calcolata da questa posizione relativa.

I problemi associati a questa tecnica sono in primis quelli di ottenere una buona visione dell'occhio; il movimento laterale della testa può sfocare l'immagine video dell'occhio, o anche far uscire l'immagine dell'occhio fuori dall'inquadratura della camera. Il range oltre il quale la direzione dello sguardo può essere tracciato (req. 6 circa) da un semplice programma è $\pm 12^\circ$ - 15° , come asseriscono Scott & Findlay, dato dal fatto che il movimento dell'occhio fatto per focalizzare un punto più lontano porterà il *luccichio* fuori dalla parte sferica della cornea, richiedendo così calcoli più complessi per avere le necessarie correzioni geometriche.

Questi problemi in futuro potranno essere aggirati attraverso migliori pro-

grammi di correzione, migliori camere di tracking che si adattano alla posizione della testa dell'utente (come indicato da Jacob 1991²) e forse diverse camere che coopereranno per espandere il raggio d'azione oltre il quale la direzione degli occhi può essere registrata.

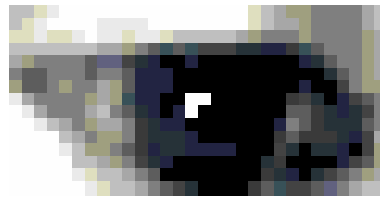


Figura 2.3: Un esempio di immagine 30x15 pixel in bassa risoluzione, estratta da una *wide angle* immagine dell'occhio dell'utente (Da Baluja & Pomerleau 1994)

2.3.4 Corneal Reflection and Eye image using an Artificial Neural Network

Una delle più recenti tecniche sviluppate si basa sulle Reti Neurali (Baluja & Pomerleau 1994³). Lo strumento utilizzato per l'*eye-gaze tracking* è ancora una video immagine digitale dell'utente, ma questa tecnica è basata su una più larga-angolata immagine del soggetto, in modo che l'intera testa rientri nel campo dell'inquadratura. Una luce stazionaria è puntata di fronte all'utente, e la procedura inizia cercandone l'occhio destro attraverso la ricerca del riflesso di questa luce (il *luccichio*) nella video immagine, distinguibile essendo un piccolo ma molto luminoso punto circondato da una regione scura.

²Jacob, R. J. K. (1991), *The use of eye movements in human-computer interaction techniques: What you look at is what you get*, ACM Transactions on Information Systems 9(3), 152-169.

³Baluja, S. & Pomerleau, D. (1994), Non-intrusive gaze tracking using artificial neural networks, Research Paper CMU-CS-94-102, School of Computer Science, Carnegie Mellon University, Pittsburgh PA, USA

Poi si estrae una più piccola, rettangolare parte della video immagine (tipicamente solo 40x15 pixels) centrata sul luccichio (fig:2.3) che sarà elaborata dalla Rete Neurale. L'output è un set di coordinate video.

La Rete Neurale richiede più della semplice calibrazione richiesta nelle altre tecniche; deve essere educata da un gruppo di immagini degli occhi e della testa dell'utente per almeno 3 minuti, mentre l'immagine muove un cursore sul display. Questo è fatto da una sessione automatica di addestramento che usa immagini continue di almeno 30 minuti, realizzate con la tecnologia corrente, ma che il sistema non richiede la volta successiva.

L'accuratezza della seguente tecnica non è la stessa delle altre; può essere migliorata leggermente con un algoritmo di calcolo basato sulle posizioni di cornea e pupilla relativamente al calcolo della posizione dell' *luccichio* nella cavità dell'occhio, ma è ancora limitato ad un'accuratezza di circa 1.5-2° (così non soddisfa il req. 4). Il grande vantaggio è che dato un *wide angle* della base dell'immagine, la mobilità della testa aumenta, tanto da poter muovere la testa oltre i 30 cm. Questo può fornire una soluzione accettabile per situazioni che non necessitano di un alto grado di accuratezza.

2.3.5 Purkinje Image Tracking

La prima e la quarta *Immagine di Purkinje* possono essere usate per il tracking attraverso la *Dual-Purkinje Image* tecnica (Muller 1993⁴), che usa le posizioni relative di questi riflessi per calcolare la direzione dello sguardo. Questa tecnica è generalmente più accurata delle altre, e la campionatura di frequenza è alta, fino a 4000Hz. Lo svantaggio è che la quarta immagine di Purkinje è piuttosto debole, così che la luce dell'ambiente circostante deve essere controllata adeguatamente (Cleveland & Cleveland 1992)⁵.

⁴Müller, P. U., Cavegn, D., d'Ydewalle, G. & Groner, R. (1993), A comparison of a new limbus tracker, corneal reflection technique, purkinje eye tracking and electro-oculography, in G. d'Ydewalle & J. V. Rensbergen, eds, *Perception and Cognition*, Elsevier Science Publishers, B.V., pp. 393-401.

⁵Cleveland, D. & Cleveland, N. (1992), Eyegaze eyetracking system, in *Imagina: Images Beyond Imagination. Eleventh Monte-Carlo International Forum on New Images*, LC

2.4 Una tecnica basata sul potenziale elettrico della pelle

L'*ellettro-oculografia* si fonda sull'esistenza del campo elettrostatico che ruota intorno all'occhio. Registrando le sufficienti piccole differenze nel potenziale della pelle intorno all'occhio, può essere ricavata la posizione dell'occhio; questo è realizzabile con l'ausilio di elettrodi posti sulla pelle intorno all'occhio.

Questa tecnica non richiede una chiara visione dell'occhio ed ha un largo e dinamico raggio di approssimazione $\pm 70^\circ$. Tra gli svantaggi notiamo che necessitando della presenza degli elettrodi risulta altamente invasiva e quindi non adatta per un uso quotidiano; non soddisfa i req. (4), (5) e (7).

2.5 Tecniche basate sull'utilizzo di lenti a contatto

Fornendo l'utente di particolari lenti a contatto, è possibile avere una rilevazione sufficientemente accurata della direzione dello sguardo. Esistono due differenti tecniche: incidendo uno o più piani visivi della superficie della lente, i raggi riflessi della luce possono essere usati per calcolare la posizione degli occhi. Inserendo un minuscolo rocchetto di induzione nelle lenti, l'esatta posizione delle suddette può essere registrata attraverso l'uso dell'alta frequenza elettromagnetica del campo posto intorno alla testa.

Questa tecnica difficilmente può essere usata quotidianamente: l'uso di lenti particolari possibilmente connesse ai fili, la presenza del campo elettromagnetico rendono questa tecnica troppo ingombrante per un uso non sperimentale. Forse un giorno saranno disponibili strumentazioni in grado di supplire queste difficoltà, ma è in dubbio la possibilità di usare questo sistema per un uso domestico.

Technique:	Limbus tracking	Pupil tracking	Corneal/pupil relationship	Artificial neural network	Dual Purkinje images	Electro-oculography	Contact lens search coil
Face access	poor to good	good	good	good	good	good	-
Subject contact	headmount/ chin rest	none	none	none	none	electrodes	contact lens
Image stabilising	-	-	-	-	yes	-	-
Accuracy	$h=0.5-7^\circ$ $v=1-7^\circ$	0.003°	$0.5-2^\circ$	1.5°	0.017°	$\pm 1.5-2^\circ$	0.08°
Resolution	0.1°	0.005°	good	-	0.25°	good	0.017°
Range	$h=\pm 15-30^\circ$ $v=\pm 15-20^\circ$	$h=\pm 30-40^\circ$ $v=\pm 20-40^\circ$	$h=\pm 12-40^\circ$ $v=\pm 12-50^\circ$	-	$\pm 20-60^\circ$	$\pm 70^\circ$	$\pm 25^\circ$
Sampling speed	200-4000Hz	50-250Hz	25-50Hz	15Hz	4000Hz	-	1000Hz
Real-time-response	-	6-12ms delay	-	-	1ms delay	yes	-
Rotational measurements	X/(Y)	X/Y	X/Y/Z	X/Y	X/Y	-	X/Y/Z
Pupil diameter measurement	no	yes	yes	-	-	-	no
Translation insensitivity	poor	good	-	up to 30cm	up to 25mm	-	-
Binocularly extendable	yes	yes	-	-	-	-	-
Subject variety	low	reasonable ⁶	reasonable ⁶	reasonable	-	-	-
Pricing	\$2,700-48,000	\$10,000-45,000	\$22,000-120,000	-	\$37,000-65,000	\$2700-	-

Tabella 2.1: Utilizzo di *eye-gaze tracking* attuali. Le informazioni sono tratte da diverse fonti, tra le quali Scott & Finflay 1993, Baluja & Pomerleau 1994, Wooding 1995, ecc. . .

2.6 Confronto tra le Tecniche di Tracking

Una descrizione di vantaggi e svantaggi delle tecniche descritte nei precedenti paragrafi può essere vista nella Tab: 2.1; i campi bianchi sono relativi alla mancanza di dati. I campi descrittivi sono quelli presentati a pag 15. E' stata omessa la proprietà (11), compatibilità di registrazione, in quanto grazie alle prestazioni di calcolatori potenti è un requisito spesso superato. In alcuni casi indichiamo l'accuratezza del raggio per entrambe le direzioni orizzontale(h) e verticale(v). Abbiamo anche indicato se la tecnica è in grado di misurare il diametro della pupilla, e infine abbiamo elencato il prezzo dell'apparecchiatura. Le informazione sono soprattutto una combinazione delle ricerche di Scott & Findlay 1993, Wooding 1995⁶ e Baluja & Pomerleau

⁶Wooding, D. (1995), *Eye-Movemen Monitoring Equipment*, Internet Document, University of Wales College of Medicine, Cardiff, UK. Version 2.0 of <ftp://mailbase.ac.uk/pub/lists/eye-movement/files/equipment.txt> or

1994. Nota che alcuni valori fruttati dalle strumentazioni potrebbero riferirsi a situazioni ideali. Inoltre alcune strumentazioni sono reperibili in commercio in differenti pacchetti, e così alcune dei dati elencati dipendono da quale tipo di tecnica è stata usata. Per esempio, la *Corneal and Pupil Reflection Relationship* normalmente richiede che l'utente sieda abbastanza lontano così che il tracciato dell'occhio sia nel campo di vista della lente dell'obiettivo; ma la stessa tecnica è anche disponibile in un pacchetto esteso, dove una seconda camera a largo-angolo è usata per localizzare la testa e in sequenza orienta la tele-lente correttamente (fig:2.4).

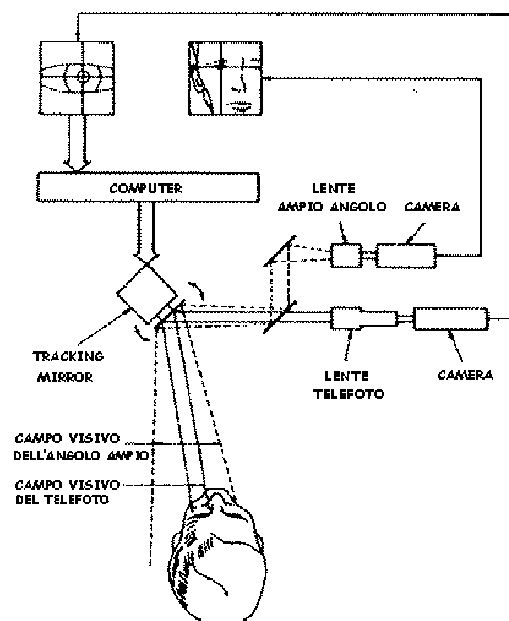


Figura 2.4: Eye Tracking che usa uno specchio di fronte alla camera.

2.7 EagleEyes Project

James Gips è il coordinatore di un gruppo di ricerca del Dipartimento di *Computer Science* di Boston che lavora da circa 6 anni al progetto "EagleEyes", che si occupa dell'uso dell'eye-tracking nell'ambito della disabilità.

<http://www.cf.ac.uk/uwcm/dr/groups/vision/em.txt>, dated 27.04.95.

Il loro obbiettivo è quello di educare persone di giovane età affette da profonda disabilità al controllo del PC attraverso il movimento dei propri occhi. La tecnologia *EagleEyes* rileva il potenziale elettro-oculografico (EOG) attraverso 5 elettrodi posti sulla testa. Vengono rilevati due canali: verticale e orizzontale. La prima fonte del segnale è la differenza di potenziale tra la cornea e la retina. I segnali cambiano di circa 20 microvolts per ogni grado di movimento dell'occhio. Gli elettrodi sono connessi a due Grass Instruments amplificatori elettrofisiologici. I segnali sono letti da un National Instruments board di acquisizione dati installato su piattaforma Windows o Macintosh. Particolari driver software ricevono il segnale dal board ogni 10 millisecondi e lo convertono nelle coordinate del mouse pointer. Ovviamente il variare della direzione dello sguardo determinerà lo spostamento del cursore del mouse. L'operatore può passare dall'EagleEyes control al mouse semplicemente con un tasto.

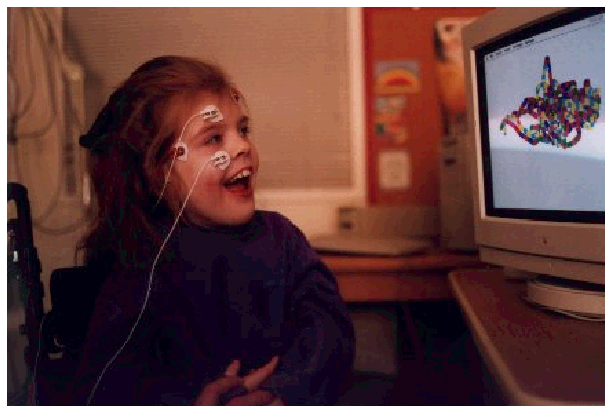


Figura 2.5: Alexa usa l'EagleEyes per disegnare

EagleEyes ha le stesse funzionalità di un mouse, ed è dotato di software educativi e addirittura giochi; in più è compatibile con molti software esistenti in commercio.

Un maggiore sviluppo del progetto è quello di insegnare alle persone il corretto utilizzo del puntatore. Persone non affette da disabilità imparano dopo 30 minuti scrivendo una lettera ogni 2,5 secondi. I bambini hanno più dif-

ficoltà. Le persone disabili hanno un approccio più lento e tormentato: in fondo questo strumento potrebbe cambiare il loro modo di relazionarsi col mondo. Così inizialmente preferiscono essere spettatori e verificare il corretto funzionamento dello strumento. Il momento della prova effettiva spesso è reso difficoltoso da tremori e spasmi involontari. Comunque in molti casi si è dimostrato uno strumento valido.

L'EagleEyes è un progetto del College di Boston di carattere umanitario e non a scopo di lucro. Viene testato in un campus interno per ragazzi con gravi disabilità e in altri due satelliti vicini alla città. Inoltre esistono altri EagleEyes system nel Connecticut e in Inghilterra.

2.8 Eyegaze Communication System

Un altro sistema esistente in commercio che si occupa dell'interazione tra calcolatore e persona disabile è l'*Eyegaze Communications System*.

Si tratta di uno strumento di comunicazione *direct-select-vision* che offre diverse funzioni:

1. Sintetizzatore vocale;
2. Controllo ambientale;
3. Controllo di accessori come telefono o televisore;
4. Controllo del PC;

E' corredato da diversi tipi di programmi, come i tutorial ,dove lo schermo è diviso in 4 o in due parti, o tastiere virtuali.

Il sistema si basa sul metodo *pupil-center/corneal-reflection* per determinare la direzione dello sguardo. Una videocamera a raggi infrarossi, montata sul monitor del sistema, rileva 60 immagini al secondo degli occhi. Un LED (*infrared light emitting diode*), montato nel centro della lente della videocamera illumina gli occhi. Questo riflette una piccola quantità di luce sulla superficie

della cornea dell'occhio che risplende attraverso la pupilla e riflette sulla retina, che appare bianca. Il processo di localizzazione del centro della pupilla è agevolato da questa tecnica ed a questo punto il calcolatore riesce a calcolare le coordinate del punto che sta focalizzando il soggetto. Questo sistema ha un'accuratezza di circa un quarto di pollice.

Il sistema prima di poter interpretare lo sguardo dell'utente deve essere calibrato attraverso una procedura durante la quale il soggetto fissa un piccolo circolo sul monitor per circa 15 sec.

Possono utilizzare questo strumento quei soggetti che hanno un adeguato controllo visivo, che riescono a mantenere una posizione stabile di fronte al monitor e che ovviamente hanno delle sufficienti abilità mentali per rapportarsi al calcolatore; il sistema è in grado di funzionare anche in presenza di occhiali.

L'*Eyegaze* è stato testato su persone affette da sclerosi laterale amiotrofica, traumi cerebrali, paralisi cerebrali, sclerosi multipla, distrofia muscolare, ecc. . . E' un prodotto distribuito dalla LC TECHNOLOGIES, INC©.



Figura 2.6: Eyegaze Communication System

2.9 Il Metodo di Zhu-Yang

Nei paragrafi precedenti sono stati descritti diversi sistemi di eye-gaze tracking, di carattere più o meno invasivo. Le ricerche degli ultimi anni sono

dirette nello sviluppo di tracking non-invasivi. Questi sistemi si dividono in due categorie:

- Approccio analitico, che calcola la direzione dello sguardo attraverso l'analisi della locazione di alcune caratteristiche facciali;
- Utilizzo delle *Reti Neurali*, che elaborano le immagini confrontandole con il proprio database.

Lo scopo è evitare all'utente qualsiasi attrezzatura *head-mounted*, e permettergli una certa libertà di movimento. In più un sistema di tracking real-time deve soddisfare le seguenti proprietà: velocità, accuratezza, robustezza.

Nell'approccio analitico, l'accuratezza del sistema dipende maggiormente dalla risoluzione dell'immagine. Imporre uno zoom ravvicinato limita i movimenti dell'utente; d'altra parte se si preserva la libertà di movimento, dobbiamo ricorrere a campi visivi più ampi che offrono una bassa risoluzione della zona oculare. Per queste ragioni sono state utilizzate le Reti Neurali che utilizzano le intense informazioni delle immagini con bassa risoluzione, ottenendo una buona accuratezza. Spesso però questi metodi non sono robusti, in quanto non sopportano i cambiamenti dell'ambiente circostante.

Da un'analisi più accurata dell'evoluzione delle Reti Neurali, si nota che queste si servono soprattutto delle regioni delle immagini che contengono informazioni relative ai bordi. Questa è la caratteristica fondamentale che ispira il *Metodo di Zhu-Yang* [8], ovvero l'analisi dei bordi e dei patterns locali per ottenere la rilevazione delle caratteristiche dell'occhio con una precisione del subpixel. In particolare in questi paragrafi verranno descritti i due algoritmi cardine del *metodo di Zhu-Yang* che rilevano robustamente gli angoli dell'occhio e il bordo dell'iride. A differenza dei metodi precedenti, che usano la trasformata di Hough [9] per approssimare il bordo dell'iride con una circonferenza, verrà proposto un efficiente e robusto algoritmo di carattere ellittico per cercare il centro dell'iride attraverso il rilevamento dei punti del bordo dello stesso. Un semplice ma accurato e robusto schema è usato per calcolare la direzione dello sguardo elaborando i dati relativi alle

caratteristiche analizzate. Questi approcci contribuiscono a rendere robusto ed efficace il sistema di tracking. Gli esperimenti dimostrano che il nuovo metodo ha un errore inferiore a $1,4^\circ$ usando una risoluzione normale, che è superiore ai risultati relativi a tutti gli altri metodi. In più è anche efficiente da un punto di vista computazionale, riuscendo ad essere real-time usando un calcolatore dotato di processore PIII da 900MHz.

2.10 Accuratezza del Gaze Tracking

2.10.1 Accuratezza ottimale

Definiamo la direzione dello sguardo come la direzione del vettore generato dal centro dell'iride e dal punto dello spazio osservato. Ci sono due tipi di fotorecettori nella retina dell'occhio umano: *bastoncelli* che sono responsabili della visione nella luce fioca e *coni retinici* responsabili della visione più accurata in situazioni di luminosità medie o alta. La fovea è una piccola area del cono nella retina responsabile della visione accurata con un diametro di circa $5,2^\circ$. Una piccola zona di diametro $1,7^\circ$, detta *rod-free* (priva di bastoncelli), nella fovea ha sempre un'alta risoluzione spaziale. Quando l'occhio guarda in una direzione ruota per permettere ai raggi luminosi di attraversare la fovea o la rod-free fovea. Per questo motivo $1,7^\circ$ rappresenta una buona accuratezza per un sistema di gaze tracking.

Dimensione del campo visivo di un occhio	$\sim 135^\circ \times \sim 160^\circ$
Raggio di rotazione dell'occhio	$\sim 70^\circ \times \sim 70^\circ$
Diametro della fovea	$\sim 5,2^\circ$
Diametro della rod-free fovea	$1,7^\circ \sim 2^\circ$
Raggio dell'occhio	1.3cm

Tabella 2.2: Alcuni parametri dell'occhi umani

2.10.2 Barriere Tecniche

Un sistema di tracking non-invasivo riesce difficilmente a ottenere l'accuratezza ottimale. Questa difficoltà deriva principalmente dall'insufficiente risoluzione dell'immagine degli occhi. Supponiamo che la dimensione dell'immagine dell'occhio sia 40x25 pixels (situazione tipica) e il raggio dell'iride sia di circa 20x20 pixels. A livello pixel la migliore risoluzione è di circa $70^\circ/20 \approx 3,5^\circ$ in entrambe le direzioni e gradi di elevazione. Più precisamente, consideriamo un metodo di calcolo della direzione dello sguardo che deriva dalla posizione dell'iride e dal centro degli occhi, o equivalentemente una qualsiasi caratteristica facciale che serva come punto di riferimento nei calcoli. La fig:2.7 mostra la struttura dell'occhio. Siano:

$$|l| \in [0, 10]; r \approx 20; |\Delta l| \geq 1$$

espresse in pixels. Allora:

$$\theta = \arcsin l/r \quad (2.1)$$

$$\Delta\theta = \frac{\partial\theta}{\partial l}\Delta l + \frac{\partial\theta}{\partial r}\Delta r = \frac{1}{r\sqrt{1-(l/r)^2}}\Delta l - \frac{1}{r^2\sqrt{1-(l/r)^2}}\Delta r \quad (2.2)$$

Supponiamo che $\Delta r = 0$, allora per $|l| \in [0, 10]$, si ha che: $|\Delta\theta| \in [2.9^\circ, 3.3^\circ]$. Così, la più piccola unità di θ è circa 3° . Questo è consistente con quanto affermato fin'ora: se nel rilevamento dell'iride si ha un errore minimo di 1 pixel, si avrà un errore di circa 3° nella direzione dello sguardo. In più, altri fattori determinano l'accuratezza del tracking, come un'intrinseco errore nella costruzione del modello dell'occhio. Alcuni ricercatori hanno schematizzato l'occhio con una sfera. Questo può dare un errore in Δr nell'equazione (2.2), che può causare un errore superiore a 6° se l'errore della stima del raggio è di circa 0.2cm.

Molti risolvono questi problemi di approssimazione usando una zoomed-in camera o semplicemente avvicinando la camera all'utente. Questi approcci richiedono che il soggetto o la sua testa siano immobili per mantenere gli occhi nel campo della camera con visibilità limitata.

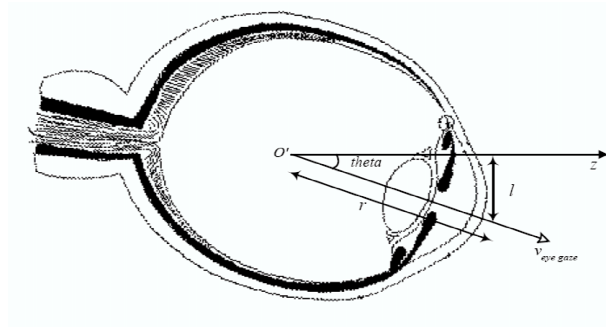


Figura 2.7: Struttura dell'occhio per l'analisi dell'errore

2.10.3 La scelta dell'approccio

Come abbiamo accennato un approccio analitico porta a dei margini di errore superiori rispetto all'utilizzo delle Reti Neurali. I dati sperimentali confermano infatti per questo approccio un'accuratezza di $1,7^\circ$. Una Rete Neurale prende le immagini degli occhi come input e "impara" i modelli dagli esempi. Raggiungono un livello di accuratezza alto perché sfruttano l'intensità delle informazioni dei pixel dell'intera immagine. Di conseguenza, le possibili variazioni non sono di 20×20 differenti posizioni del centro, ma $256^{40 \times 25}$ differenti immagini, assumendo una scala di grigio di 8-bit. Anche se le reti neurali permettono di avere un'alta accuratezza nelle valutazioni, pochi sistemi sfruttano questo approccio, perché si ha una maggiore sensibilità rispetto al cambiamento del soggetto preso in analisi, delle condizioni luminose e ogni spostamento dell'utente. Ciò che causa questi problemi è la proprietà che impedisce loro di sormontare la barriera della risoluzione (l'utilizzo delle informazioni dei pixel). I cambiamenti oltre che nella direzione dello sguardo possono anche avere grande effetto sull'intensità dei pixel. Ancora, non riesce a distinguere i pixels in uso da quelli esterni all'area analizzata.

Anche se le Reti Neurali hanno problemi inerenti con le applicazioni reali, la loro proprietà di codificare più informazioni può essere usata per migliorare l'accuratezza dell'approccio analitico. Vogliamo capire meglio come le reti neurali decodificano le informazioni. La fig:2.8 mostra due immagini dell'oc-

chio (a), (b) e la loro differenza. La direzione dello sguardo rispetto alle due immagini è minima. Quando trasmettiamo le due immagini alle Reti Neurali, ciò che rende l'output differente sono principalmente quei pixel che hanno grandi differenze. Quindi anche se la Rete Neurale prende delle immagini intere come input, ciò che realmente contribuisce alla valutazione dell'output sono soltanto alcuni pixel, quelli cioè posti sul bordo dell'iride e dello sclera. Gli altri pixels possono rappresentare del rumore e rendere il sistema instabile. Da questa osservazione nasce l'idea di stimare la direzione dello sguardo attraverso le caratteristiche locali e patterns anziché attraverso l'analisi dei singoli pixels. In questo modo un approccio analitico può ottenere un'accuratezza del subpixel attraverso la codifica dell'intensità dei pixels del bordo.

La procedura analitica usa almeno un punto mobile dell'occhio e uno fisso



Figura 2.8: Due immagini di un occhio e la loro differenza

del viso come riferimento per calcolare la direzione dello sguardo. Molte ricerche usano il centro dell'iride come punto mobile e l'angolo dell'occhio, cioè l'imbocco del dotto lacrimale, o altri marks speciali come punti fissi. Questi sistemi permettono un'accuratezza di livello pixel. Invece di usare punti, usiamo patterns locali e caratteristiche locali che permettono l'accuratezza del subpixel. Nel *Metodo di Zhu-Yang* viene scelto l'iride come parte mobile per il suo alto contrasto con lo sclera e l'angolo dell'occhio come punto fisso perché è una caratteristica molto stabile del viso.

Poiché la forma e la distribuzione di intensità dell'iride nell'immagine dell'occhio cambiano drammaticamente col movimento, se ne tracciano i punti del bordo e poi vengono interpolati con un'ellisse dinamico che serve per cercare il centro dell'iride. Vengono utilizzate caratteristiche locali per determinare l'angolo dell'occhio con l'accuratezza del subpixel.

2.11 Gaze Tracking con accuratezza di livello subpixel

Tecniche sub-pixel sono state sviluppate in varie applicazioni: image registration [10][11], stereo vision [12], edge and corner detection [13][14][15][16][17][18], and feature detection [4]. Tra i lavori precedenti quelli più inerenti al *Metodo di Zhu-Yang* sono: *subpixel edge and corner detection* e *feature detection*. Sfortunatamente non possono essere applicati direttamente ad un sistema real-time per le seguenti ragioni:

- **Eccessivo costo computazionale:** quasi tutti gli algoritmi richiedono pesanti calcoli per ogni singola immagine, che non è accettabile in una applicazione real-time;
- **Modelli inappropriati:** il popolare paraboloide lavora male con caratteristiche asimmetriche e immagini piccole. Quasi tutte le procedure di livello subpixel di rilevazione del bordo presuppongono un bordo diritto o altri modelli del bordo, che risultano inadeguati per un'operazione d'inseguimento;
- **Dati insufficienti:** questi algoritmi sono stati sviluppati per immagini ad alta qualità.

Per risolvere questi problemi è stato sviluppato un nuovo algoritmo di *eye corner tracking* e un algoritmo per rilevare informazioni sull'intensità locale basato sul bordo dell'iride.

2.11.1 Subpixel eye corner tracking

Le problematiche relative all'utilizzo di un algoritmo che abbia prestazioni real-time sono risolte con *l'eye corner tracking*: la forma e l'orientazione degli angoli degli occhi è molto stabile in molte persone e nelle varie espressioni facciali. Quindi non dobbiamo sviluppare uno strumento che cerca gli angoli

in ogni orientazione. Fatte queste considerazioni proponiamo il seguente algoritmo:

1. Definire una piccola area (SA) di ricerca nell'immagine che abbia come punto di partenza la posizione anteriore dell'occhio ;

2. Convolvere SA con il filtro CF (fig:2.9)

$$FO = SA \otimes CF$$

3. Definire una più piccola area SA' come l'area intorno ai pixels di FO con valori più alti. Filtrare SA':

$$FO' = SA' \otimes CF$$

4. Applicare l'interpolazione bi-cubica a FO' per avere una griglia più fine;
5. I punti di FO' interpolato con i valori più alti sono i *subpixel point* dell'angolo.

Questo algoritmo presenta le seguenti caratteristiche:

- Ha un costo computazionale basso;
- Attraverso l'uso di un semplice filtro *eye corner*, non una maschera estratta dall'immagine, si adatta ai cambiamenti dell'user, della luce ecc. . .
- Il filtro è fatto per bilanciare valori positivi e negativi, che aiutano a eliminare la possibilità di rumori indesiderati.

La fig:2.10 mostra un esempio di FO' e la sua interpolazione.

$$\begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Figura 2.9: Coppia di *corner filters*

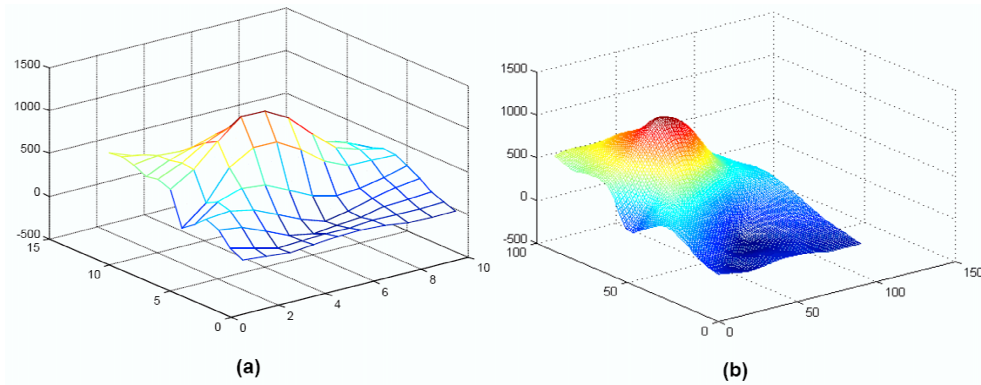


Figura 2.10: Risultato del *corner filtering* e la sua interpolazione

2.11.2 Ricerca del bordo e del centro dell'iride

In questo sottoparagrafo vengono presentate le procedure per il calcolo dei punti del bordo dell'iride, il calcolo del centro dell'iride e un'ulteriore algoritmo per distinguere i punti di confine tra iride e sclera da quelli di confine tra palpebra e iride.

Rilevazione del bordo dell'iride

Essendo alto il contrasto tra lo sclera e l'iride, i pixels del bordo dell'iride possono essere individuati facilmente. Per ottenere la precisa locazione del centro dell'iride abbiamo bisogno dei punti del bordo a livello subpixel. L'algoritmo è simile a quello precedente:

1. Ad ogni pixel del bordo dello sclera-iride applichiamo gli operatori orizzontali e verticali di Sobel⁷ e consideriamo i due valori u e v ;
2. Si calcola la direzione del gradiente di Sobel così:

$$d(e) = \arctan(u/v) \quad (2.3)$$

3. Si calcola il valore dell'operatore Sobel per ogni punto adiacente ai pixels lungo la direzione del gradiente;

⁷Trucco E., Verri A., *Introductory techniques for 3-D Computer Vision*, Prentice-Hall, USA, 1998

4. I valori dell'operatore di Sobel vengono interpolati con una cubica 1-dimensionale;
5. Si cercano i punti di massimo;

Nell'applicazione real-time, poiché i punti di contorno dell'iride-sclera che non sono occlusi dalle palpebre si estendono verticalmente, la procedura è semplificata assumendo la direzione orizzontale del gradiente di Sobel. La coordinata y dei punti subpixels del bordo è fissata nel punto medio del bordo e la coordinata x è calcolata con l'interpolazione. Questa è un'approssimazione della procedura che da buoni risultati pratici.

Stima del centro dell'iride

Una volta calcolati i punti del bordo, si può trovare il centro dell'iride. Un approccio comune è usare la trasformata di Hough⁸ per approssimare questi punti con un cerchio. Questo approccio ha diversi svantaggi:

- Per un ampio raggio della direzione dello sguardo, l'iride in un'immagine non può essere approssimata con un cerchio;
- Le immagini dell'occhio disponibili sono in genere piccole nel formato. Così il numero di punti del bordo sono molto limitati. Questo fa della trasformata di Hough una procedura discutibile.
- Il diametro dell'iride è spesso assunto come noto, mentre varia da persona e persona e dipende inoltre dalla distanza della camera dagli occhi.

Nel *Metodo di Zhu-Yang* si utilizza un efficiente ed accurato algoritmo di tipo ellittico. I punti del bordo sono quindi considerati come punti di un'ellisse e il calcolo dei parametri dell'ellisse diventa un problema di minimi quadrati. Questo approccio ha i seguenti vantaggi:

⁸Trucco E., Verri A., *Introductory techniques for 3-D Computer Vision*, Prentice-Hall, USA, 1998

- Nella pratica l'ellisse è una migliore approssimazione dell'iride rispetto alla circonferenza. I risultati empirici dimostrano che è un approccio abbastanza buono per variazioni della direzione dello sguardo che vanno fino a $\pm 20^\circ$ in azimuth e in altezza;
- Anche se l'algoritmo lavora meglio quando ci sono molti punti di input, può dare buoni risultati con almeno 6 punti. Questa è un'ottima proprietà nella pratica. Infatti il caso peggiore riscontrato ha portato una disponibilità di 7 punti;
- Non sono richieste ipotesi sulla dimensione dell'iride. L'ellisse può sempre essere trovato.

La precedente discussione sulla determinazione del bordo nel subpixel è anche essenziale in situazioni dove ci sono molto pochi punti di contorno. Se si usassero solo punti discreti incontreremmo una matrice singolare nel trovare l'ellisse perché i punti di confine che dovrebbero avere differenti coordinate in ascisse potrebbero essere nella stessa colonna. Con l'interpolazione il problema è risolto.

Distinzione tra punti della palpebra e dello sclera

Grazie all'occlusione dell'iride tramite le palpebre, ci sono due gruppi di punti del bordo intorno all'area dell'iride: punti tra l'iride e lo sclera e punti tra l'iride e la palpebra. Per il calcolo dell'ellisse bisogna prendere in considerazione solo quelli tra l'iride e lo sclera. Per distinguere tra i due tipi usiamo il seguente metodo: tracciare il punto più elevato del bordo superiore della palpebra ed il punto del bordo più basso della palpebra. Questi punti e gli angoli dell'occhio sono connessi da segmenti di retta per definire un profilo approssimativo del contorno dell'occhio. Soltanto i punti del bordo che cadono all'interno di questo profilo sono considerati come punti del bordo dell'iride.

2.11.3 Stima della direzione dello sguardo

Il calcolo della direzione dello sguardo non è sicuramente un problema banale. La sua risoluzione implica l'introduzione di modelli geometrici e assunzioni di comodo, come la dimensione e il movimento dell'occhio. Questi fattori non solo riducono la robustezza del procedimento, ma spesso introducono anche errori delle stime.

La procedura che sta per essere descritta si ispira ai sistemi di tracking già presenti in commercio; la sua struttura di calcolo è semplice e garantisce una certa robustezza e accuratezza. Questi sono i passi fondamentali:

1. Lo sguardo dell'utente deve fissarsi su una serie di punti noti del piano, come angoli e altri punti di ancoraggio. Si registrano le posizioni relative dell'angolo dell'occhio e del centro dell'iride. Questi sono punti di calibrazione;
2. Si costruisce una mappa bidimensionale dal vettore (*corner-iris*) tra l'angolo dell'occhio e il centro dell'iride e gli angoli generati dalla direzione dello sguardo. Le direzioni successive vengono calcolate attraverso l'interpolazione. Per esempio, supponiamo che l'angolo dello sguardo e il vettore dall'angolo dell'occhio al centro dell'iride per i punti di calibrazione P_1 e P_2 siano rispettivamente $\{(\alpha_1, \beta_1), (x_1, y_1)\}$ e $\{(\alpha_2, \beta_2), (x_2, y_2)\}$; se osserviamo il vettore corner-iris (x, y) , il corrispondente angolo dello sguardo è così calcolato:

$$\alpha = \alpha_1 + \frac{x - x_1}{x_2 - x_1}(\alpha_2 - \alpha_1) \quad (2.4)$$

$$\beta = \beta_1 + \frac{y - y_1}{y_2 - y_1}(\beta_2 - \beta_1) \quad (2.5)$$

Lo schema descritto ha i seguenti vantaggi:

- E' computazionalmente semplice ed efficiente;
- E' estremamente accurato. Questo potrebbe sorprendere, ma se esaminiamo lo schema (fig:2.11) scopriamo che l'unica approssimazione che

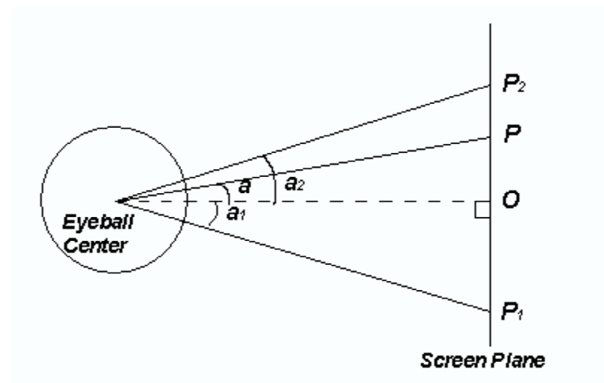


Figura 2.11: Calcolo della direzione dello sguardo

impiega è una semplificazione dell'angolo di rotazione dell'occhio come proiezione del movimento del centro dell'iride. Come è ben noto, l'approssimazione $\theta = \sin \theta$ è molto esatta per θ piccolo. Per angoli grandi fino a $\pm 35^\circ$ l'errore è di soli $1,2^\circ$. In genere però lo sguardo varia tra $\pm 15^\circ$ e l'errore in questi casi si aggira intorno a $0,17^\circ$. Metodi meno semplici hanno un'accuratezza minore.

- E' veloce, facile e malleabile. In molte applicazioni, per esempio inseguire un punto sul monitor, non è necessario conoscere l'angolo dello sguardo, ma α e β sono semplicemente le coordinate sullo schermo. Questo semplifica il processo di calibrazione che, in fase sperimentale, ha una breve durata.

2.12 Risultati sperimentali

Il metodo descritto è stato utilizzato dagli stessi autori per la realizzazione di un *eye-gaze tracking real-time*. Per valutarne l'accuratezza il sistema è stato testato usando diverse sequenze di immagini. Nei dati, al soggetto viene chiesto di seguire il cursore sul desktop del monitor usando il proprio sguardo e mantenendo la testa immobile. Il cursore si muoveva a zigzag orizzontalmente e verticalmente. Una video camera è stata posta nella parte

inferiore del monitor, con risoluzione 320x240. Il sistema ha salvato le immagini relative agli occhi. La lunghezza delle sequenze è tra i 1700 e 1200 fotogrammi. La fig:2.12 mostra un esempio di immagine di input, dalla quale si nota che l'utente ha sufficiente spazio per muoversi liberamente; un'immagine di dimensioni maggiori, per esempio 640x480, può essere usata per fornire maggiore libertà di movimento in applicazioni più generali.



Figura 2.12: Esempio di immagine di input

Sono stati confrontati i dati relativi a questo sistema e quelli generati da metodo di livello pixel e un altro di livello subpixel. L'errore medio delle prove è mostrato nella tab:2.3. L'errore medio del nuovo metodo è inferiore al diametro della rod-free fovea. Come detto prima un metodo con livello pixel, se funziona perfettamente, genera un errore maggiore dovuto all'effetto di quantizzazione. Questa è una prova del beneficio dell'uso di algoritmi di livello subpixel.

Sequence Algorithm	1	2	3
New Subpixel	1.2°	1.4°	1.1°
Old Subpixel	2.8°	3.3°	2.7°
Pixel Level	3.1°	3.7°	3.3°

Tabella 2.3: Errore medio del *gaze tracking* calcolato in tre sequenze

L'angolo di azimuth risultato del tracking su una sequenza di segmenti (200 fotogrammi) è mostrato nella fig:2.13. Nella fig:2.13(a) si può confrontare il sistema di Zhu-Yang con un altro di livello subpixel. La discontinuità dei risultati è ovvia. Nella fig:2.13(b) il risultato del metodo è confrontato con il risultato di un classico algoritmo subpixel template matching. Il metodo proposto è superiore.

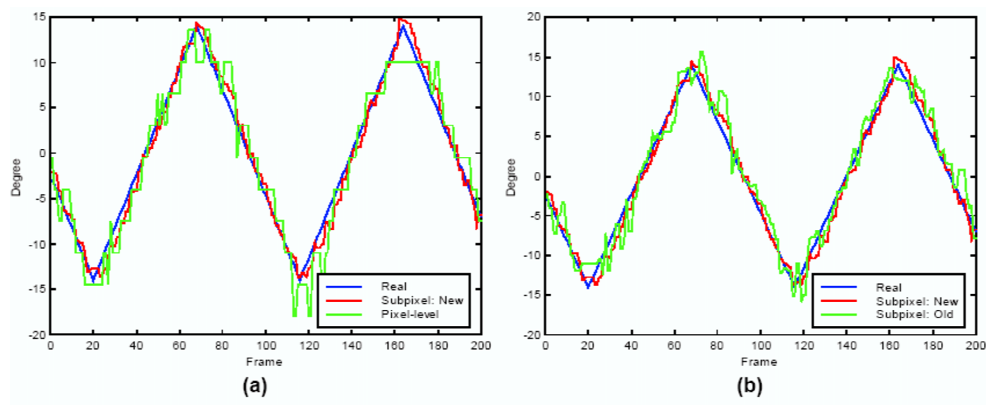


Figura 2.13: Angolo di azimuth risultante di una sequenza di segmenti

Capitolo 3

Il Metodo di Kothari-Mitchell

In questo capitolo verrà descritto un approccio computazionale per determinare in maniera accurata la locazione degli occhi dell'uomo in immagini monocromatiche; l'implementazione di questo procedimento sarà trattata nel capitolo successivo. Il metodo proposto si basa sullo sfruttamento delle caratteristiche dei gradienti calcolati nei punti di confine tra l'iride e lo sclera, cioè in quella zone dove è elevato il contrasto tra i livelli di grigio.

Un'importante caratteristica del procedimento è quella di sfruttare sia le informazioni spaziali dei pixels che quelle temporali dei fotogrammi. Il processo spaziale utilizza le informazioni relative alla variazione dei livelli di grigio per selezionare un gruppo di potenziali candidati come coordinate degli occhi. Il processo temporale usa il principio di continuità per filtrare l'attuale posizione dal gruppo di potenziali candidati. Questo metodo può essere utilizzato per la determinazione dell'angolo dello sguardo e per l'eye-tracking.

3.1 La matrice dei *Bins*

Associamo all'immagine monocromatica una matrice I di dimensione $n \times m$, dove ogni elemento della matrice rappresenta il valore di grigio del pixel corrispondente. Sia allora $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, tale che $f(i, j) = I_{ij}$, $i = 1, \dots, n$ e $j = 1, \dots, m$.

Le caratteristiche più importanti degli occhi, in un'immagine a scala di grigi, sono la forma ellissoidale dell'occhio e il grande contrasto tra l'iride e lo sclera. Un'area circolare scura, come l'iride, su uno sfondo chiaro, produce un campo di vettori uscenti dal centro dell'area descritti dalla direzione del gradiente associato alla funzione f . La fig. 3.1 mostra la direzione del gradiente locale in ogni punto dell'immagine ed è un tipico esempio del flusso del campo intorno all'iride. Se quindi consideriamo il campo dei vettori con

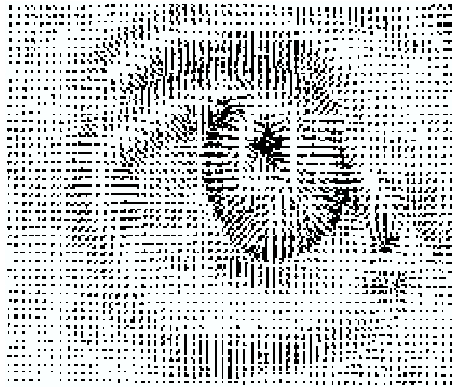


Figura 3.1: Direzioni dei gradienti in un immagine

verso opposto rispetto a quello del gradiente, questi si intersecheranno in un punto (i punti del gradiente seguono la direzione dell'incremento della funzione, cioè dalla zona più scura dell'iride a quella più chiara dello sclera). Consideriamo quindi una nuova matrice B di dimensione $k \times h$, che chiameremo matrice dei *bins*, inizializzata alla matrice nulla che useremo come accumulatore. Ogni *bin* B_{ij} verrà incrementato di uno se è attraversato da uno dei vettori generati da $-\nabla f$. Quindi i *bins* sono incrementati ogni volta che una linea li attraversa. Questo processo è mostrato per due arbitrari gradienti locali nella fig. 3.2. Ovviamente il *bin* posto nell'intersezione dei due gradienti locali ha un valore maggiore (mostrato da un colore più chiaro). Questo mostra come il *bin* posto nell'intersezione delle direzioni dei gradienti locali sarà incrementato di un largo numero di volte rispetto agli altri *bins* dell'immagine. Questo è alla base del nostro algoritmo. E' chiaro che il flusso del campo radiante posto intorno all'iride causa un simile effetto additivo nel

$bin(s)$ posto approssimativamente nel centro dell'iride.

Ci sono due variabili nel processo sopra descritto che hanno un impatto

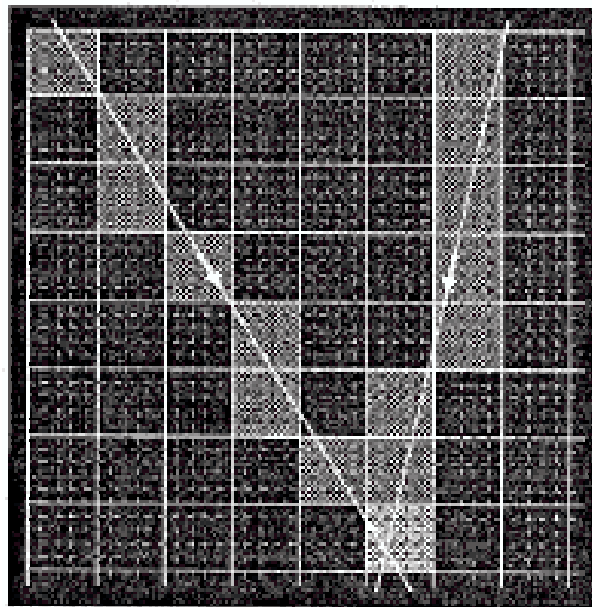


Figura 3.2: Illustrazione dello schema dell'incremento dei *bins* per due arbitrari gradienti locali. Le linee tratteggiate separano i pixels, mentre quelle continue i *bins*. La luminosità dei *bins* indica il loro livello di accumulazione.

in termini di efficienza e efficacia. La dimensione di ciascun *bin* (in termini di pixels) è uno di questi. *Bins* più grandi richiedono meno accuratezza computazionale per la localizzazione. I dati sperimentali dimostrano che un bin di 5 pixels fornisce una buona accuratezza nella localizzazione e efficacia computazionale per una largo raggio della distanza tra la camera e il soggetto (circa 3-2 piedi). Il secondo fattore che aggiunge efficienza computazionale è considerare solo gradienti di norma considerevole. Questo serve per ridurre i calcoli e privilegia i gradienti calcolati lungo i contorni dell'iride che hanno norma considerevole.

In una situazione ideale ci si aspetta che i due *bins* con maggiore accumulazione corrispondano ai due occhi. Questa previsione è però tradita dalle forme presenti nell'immagine, come oggetti sullo sfondo o vestiti del soggetto, che

fungono da rumore. Questo problema è stato sviato attraverso l'analisi delle caratteristiche spaziali e temporali dell'immagine. Il primo di questi metodi usa ulteriori proprietà dei *bins* e il secondo usa informazioni temporali per accertare la corretta locazione degli occhi.

3.2 Analisi spaziale dei *Bins*

Per ottenere la corretta locazione degli occhi da un gruppo di potenziali candidati, usiamo due assunzioni:

1. Il valore dei *bins* relativi agli occhi deve essere simile;
2. Gli occhi devono avere *ordinata* simile.

Entrambe le condizioni sono ragionevoli ma solo per un soggetto posto in posizione verticale. Sia B_{ij} il valore del *bin* di coordinate (i, j) ; sia $D = \{(x, y) \in \mathbb{R}^2 \mid 0 \leq x \leq k, 0 \leq y \leq h\}$, allora definiamo la funzione J come:

$$J : D \times D \rightarrow \mathbb{R}$$

$$J(i, j, i', j') = (B_{ij} - B_{i'j'})^2 + \lambda(j - j')^2 \quad (3.1)$$

dove (i, j) e (i', j') denotano le coordinate di due *bins* e λ controlla il relativo peso delle due condizioni. I due *bins* che minimizzano la J sono quelli cercati. Per limitare la ricerca è opportuno considerare i *bins* con valori più alti.

3.3 Analisi temporale

Il secondo metodo della scelta della corretta locazione dell'occhio da un gruppo di futuri candidati si basa sul principio di continuità della locazione degli occhi da un fotogramma al successivo. Sia $A^{(s+1)} \in M^{k \times h}(\mathbb{R})$, la matrice associata al fotogramma $(s+1)$ così definita:

$$A_{ij}^{(s+1)} = \sum g(A_{uv}^{(s)}) e^{-r} + B_{ij}^{(s+1)} \quad (3.2)$$

dove:

- u e v delimitano un intorno del punto (i, j) ;
- $g : \mathbb{R}^2 \rightarrow \mathbb{R}$ è una funzione continua che decresce rapidamente all'aumentare della distanza dal $bin (i, j)$;
- r controlla il contributo dei $bins$ del fotogramma precedente;

Così, la totale accumulazione del bin del fotogramma considerato è generato dal contributo dell'intorno dei $bins$ del fotogramma precedente e da quello attuale. La ricerca dei massimi andrà fatta dunque non sulla matrice B ma su A . Questo metodo presenta due vantaggi:

1. Non permette contraddizioni tra i risultati di un singolo fotogramma per eliminare la corretta locazione dell'occhio;
2. Fornisce un metodo più sofisticato per evidenziare le aree del fotogramma successivo alle quali deve essere ristretto il processo (quelle aree con la più alta accumulazione).



Figura 3.3: Immagini testate da Kothari & Mitchell

I risultati di questo secondo metodo sono particolarmente robusti se viene fatta una calibrazione che individua la locazione dell'occhio. Effettivamente

una combinazione dei due metodi può essere usata se le assunzioni del primo metodo risultano soddisfatte.

La fig. 3.3 mostra i risultati di questi algoritmi su un set di immagini semplici scelte per massimizzare le variazioni tipiche nelle caratteristiche facciali (per esempio occhiali, barba, capelli lunghi ecc...). I risultati del processo di incrementazione del bin sono mostrati nella fig. 3.4. Conteggi bassi dei *bins* sono rappresentati da pixels scuri mentre *bins* con valori alti hanno colori chiari.

La fig. 3.5 mostra i risultati di processi basati sul metodo spaziale. Sono

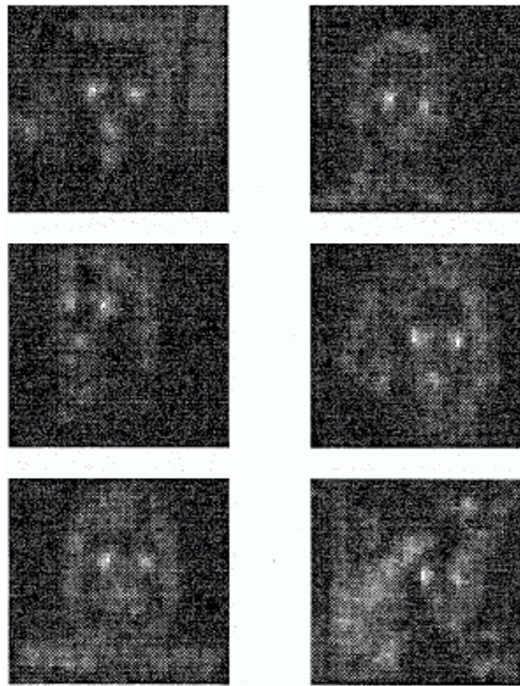


Figura 3.4: Visualizzazione delle matrici dei *bins*

evidenziati i due *bins* che minimizzano il costo funzionale dell'equazione (3.1), sovrapposti alle immagini originali. Da notare l'esattezza del risultato.

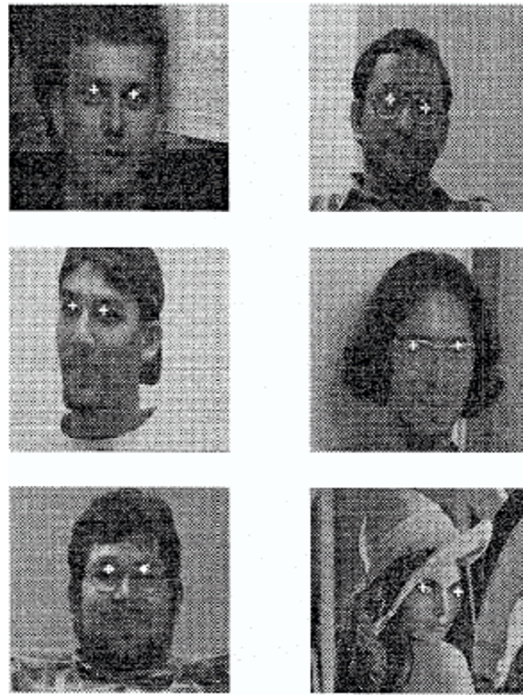


Figura 3.5: Locazione degli occhi attraverso il primo metodo

3.4 Estensione del metodo

Nel capitolo successivo verrà implementato la prima parte del metodo (analisi spaziale) per la localizzazione degli occhi all'interno di immagini. Kothari & Mitchell stanno sperimentando delle estensioni del metodo per determinare la direzione dello sguardo e tracciare il punto di focalizzazione. Il loro approccio si basa sulla variazione dell'eccentricità dell'iride rispettivamente con l'angolo dello sguardo. Lo strumento utilizzato sono le *Reti Neurali* introdotte nei capitoli precedenti.

Capitolo 4

Implementazione del Primo Metodo di *Kothari-Mitchell*

In questo capitolo verrà descritta l'implementazione del Primo Metodo di *Kothari-Mitchell* (Analisi spaziale dei *bins*) descritto nel capitolo precedente. L'algoritmo implementato elabora le immagini e cerca all'interno di esse i punti di maggiore accumulazione dei *bins* che dovrebbero corrispondere agli occhi.

4.1 Acquisizione dati

L'ambiente di lavoro scelto per l'implementazione dell'algoritmo è il Matlab (MATrix LABoratory). Il programma è fornito di librerie grafiche, *Image Processing Toolbox*, nelle quali sono implementate molte funzioni basilari per l'elaborazione delle immagini di qualsiasi formato.

La lettura delle immagini è resa possibile dal comando **imread** che restituisce un oggetto di dimensioni $m \times n \times 3$, dove $m \times n$ è la dimensione dell'immagine e la terza coordinata serve per distinguere i tre livelli RGB (Red-Green-Blue) del formato originale. Il nostro obiettivo è quello di lavorare su un'unica matrice che descriva i livelli monocromatici di grigio dell'immagine; a questo

scopo modifichiamo l'oggetto RGB con il comando **rgb2gray**. Quanto detto si traduce in:

```
RGB=imread(nomefile);
I=rgb2gray(RGB);
```

4.2 Costruzione della matrice dei *Bins*

Sia $I \in M^{m \times n}$ la matrice associata alla nostra immagine. Per quanto detto sopra ogni elemento della matrice rappresenta il livello di grigio del pixel associato, quindi $I_{ij} \in \{0, 255\}$. Sia $f : D \rightarrow \mathbb{R}$, dove $D \subseteq \mathbb{R}^2$, tale che $\forall i = 1, \dots, m$ e $j = 1, \dots, n$ si ha che $f(i, j) = I_{ij}$. Il primo passo nella costruzione della matrice dei *bins*, che chiameremo B , è la determinazione $\forall(i, j)$ del vettore gradiente $-\nabla f(i, j)$.

Sia $A \in M^{m \times n}$ e $D \in M^{u \times v}$, con $u \leq m$ e $v \leq n$, allora definiamo la convoluzione discreta di $A \otimes D$ in (i, j) come:

$$(A \otimes D)(i, j) = \sum_{h=-\lfloor u/2 \rfloor}^{\lfloor u/2 \rfloor} \sum_{k=-\lfloor v/2 \rfloor}^{\lfloor v/2 \rfloor} D[h, k] A[i - h, j - k] \quad (4.1)$$

L'approssimazione discreta delle componenti di $\nabla f(i, j)$ avviene attraverso la convoluzione della I con delle maschere che chiameremo dx e dy , che in genere sono della seguente forma:

$$\begin{pmatrix} -1 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$$

Durante lo sviluppo dell'algoritmo si è notato che l'uso di queste maschere però non portava a buoni risultati e per questo si è preferito l'utilizzo degli operatori di Sobel, cioè:

$$\begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

Quindi otteniamo che:

$$\frac{\partial f}{\partial x}(i, j) = (I \otimes dx)(i, j) \quad (4.2)$$

$$\frac{\partial f}{\partial y}(i, j) = (I \otimes dy)(i, j) \quad (4.3)$$

Il Matlab permette di effettuare la convoluzione tra matrici attraverso la funzione **imfilter**.

Il passo successivo è stato costruire la matrice delle norme e degli argomenti di $-\nabla f(i, j)$; per facilitare i calcoli gli argomenti sono stati approssimati con dei multipli di $\frac{\pi}{4}$. Le funzioni che sono state utilizzate sono **abs** per il calcolo delle norme e **angle** per il calcolo degli argomenti; l'approssimazione degli angoli è stata realizzata attraverso l'implementazione di una sottofunzione chiamata **angolo**.

A differenza di quanto specificato nel capitolo precedente si è preferito utilizzare *bins* della dimensione di un singolo pixel; di conseguenza la matrice B ha le stesse dimensioni di I . Dopo aver inizializzato B alla matrice nulla, attraverso l'utilizzo di diversi cicli si incrementa di 1 l'elemento B_{kh} se questo viene attraversato da una segmento di lunghezza l , origine (i, j) e direzione θ (l'argomento di $-\nabla f(i, j)$). Durante questa fase è emersa l'importanza di due parametri:

1. Per evitare possibili agenti di rumore e facilitare i calcoli non sono stati presi in esame i gradienti generati da pixels situati in zone con minima variazione del livello di grigio. La sogliatura dei pixel è stata effettuata usando il valor medio delle norme dei gradienti;
2. La lunghezza del segmento (*passo*) generato dall'opposto del gradiente è il secondo parametro. Valori troppo bassi o troppo alti determinano

una locazione inefficace dell'occhio; a differenza del valore di sogliatura del gradiente non si è trovata una relazione in grado di automatizzare la ricerca del valore più idoneo all'immagine presa in esame.

La seguente procedura è stata sviluppata nella funzione **BinII**.

L'analisi del valore dei *bins* verrà trattata nel paragrafo successivo.



Figura 4.1: Immagine campione

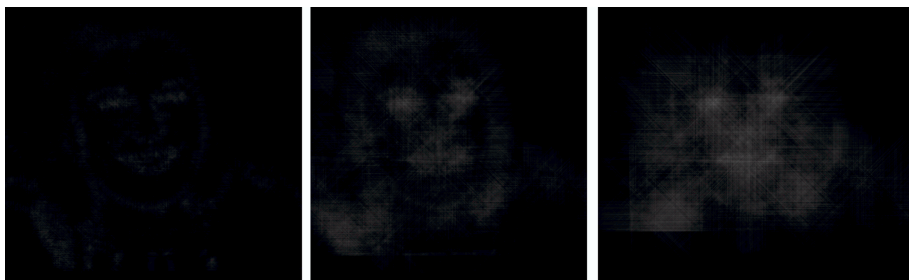


Figura 4.2: Le tre immagini rappresentano la matrice B associate alla fig. 4.1, con *passo* rispettivamente di 15, 40 e 80

4.3 Selezione dei *bins*

Una volta costruita la matrice B è necessario sottoporla ad un'analisi per la ricerca dei 2 *bins* che dovrebbero localizzare gli occhi. Si ricorda che la funzione che mette in relazione il valore di una coppia di *bins* e le loro relative ordinate è:

$$J : D \times D \rightarrow \mathbb{R}$$

$$J(i, j, i', j') = (B_{ij} - B_{i'j'})^2 + \lambda (j - j')^2 \quad (4.4)$$

L'obiettivo è quello di cercare la coppia di coordinate per le quali J sia minimo. Il primo passo è stato quello di selezionare gli elementi della matrice B che superassero una certa soglia (anche in questo caso è stato utilizzato il valor medio).

In seguito è stato sviluppato un algoritmo che confrontasse tutte le coppie di coordinate residue alla ricerca del minimo nella matrice. Questo metodo si è ovviamente dimostrato disastroso, in quanto la mole di calcoli associata a questa operazione è dell'ordine di $m^2 \times n^2$; è immediato intuire che la procedura funziona bene per matrici di dimensioni molto ridotte, mentre è totalmente inefficace su matrici di grandi dimensioni come quelle associate alle immagini.

Il tentativo successivo è stato quello di studiare gli zeri di ∇J , ma anche questo ha portato scarsi successi, in quanto l'approssimazione discreta delle coordinate del gradiente porta a errori nel nostro caso di notevole importanza. Questo metodo potrebbe essere migliorato attraverso l'interpolazione della J e il successivo studio del gradiente, ma è una strada che non è stata seguita.

L'ultimo tentativo, che si è dimostrato vincente, è stato quello di cercare il *bin* con maggiore accumulazione e poi cercare il minimo vincolato a questo punto della J . Ovviamente si ottengono ottimi risultati quando il primo *bin* localizzato corrisponde ad uno dei due occhi; questa è una situazione molto comune quando nell'immagine presa in esame sono pochi gli elementi di disturbo.

Come già evidenziato nel capitolo precedente il parametro λ assume una rilevante importanza. In condizioni di luminosità omogenea e nel caso in cui il volto sia in posizione verticale il valore ottimale, riscontrato empiricamente, è di circa 1; nel caso in cui la luminosità differisce nelle due sezioni facciali è opportuno utilizzare valori più alti.

Il costo computazionale di questa procedura è dell'ordine $m \times n$, ed è implementata nella funzione **MaxII**, che restituisce l'immagine originale con evidenziati i *bins* trovati.



Figura 4.3: Le immagini rappresentano il risultato della procedura. Le prime tre si riferiscono ad esperimenti fatti usando come *passo* rispettivamente 15, 40, 80 e $\lambda = 1$, l'ultima *passo*=80 e $\lambda = 30$

4.4 Analisi del lavoro

Le sperimentazioni effettuate sulle immagini dimostrano come i fattori fondamentali per l'esatta locazione degli occhi siano il *passo* e il parametro λ .

Quando l'illuminazione risulta carente risulta necessario l'impiego di un *passo* elevato, anche se potrebbe favorire lo sviluppo di rumori esterni. Al contrario, come è dimostrato nella fig. 4.3, in casi di buona luminosità un valore eccessivamente grande di questa quantità può provocare la formazione di *bins* spuri che impediscono la giusta locazione del secondo occhio.

Il parametro λ , sotto buone condizioni di luminosità e con un *passo* adeguato, non condiziona eccessivamente la scelta del secondo *bin* se rimane vincolato ad un intorno di 1. La terza e la quarta immagine della fig. 4.3 mostrano come in caso di cattive condizioni (in questo caso la scelta inopportuna del *passo*) la variazione di λ non riesce ad ottimizzare il risultato della procedura. Questa è solo una prima fase di ricerca nell'implementazione del Metodo di *Kothari-Mitchell*. Attraverso l'utilizzo del Matlab si è dimostrata empiricamente l'esattezza della procedura; un primo passo per il miglioramento dovrebbe essere quello di utilizzare un'ambiente di lavoro più efficiente che consenta di lavorare non solo su singoli fotogrammi ma su video. Il passo successivo sarebbe quello di automatizzare la scelta dei parametri in funzione delle condizioni di luminosità dell'ambiente e il filtraggio di eventuali elementi di disturbo.

Di seguito vengono mostrati i codici relativi al lavoro effettuato.

4.5 Codice

Acquisizione dati

```
RGB=imread(nomefile);  
I=rgb2gray(RGB);
```

Costruzione della matrice dei *Bins*

```
function B=BinII(I,passo)  
  
[n,m]=size(I);  
II=double(I);  
dx=[-1,0,1;-2,0,2;-1,0,1];  
dy=[-1,-2,-1;0,0,0;1,2,1];  
Dx=imfilter(II,dx);  
Dy=imfilter(II,dy);  
H=abs(Dx+i*Dy);  
M=mean(mean(H));  
Theta=angle(-Dy-i*Dx);  
  
B=zeros([n,m]);  
for h=1+passo:n-passo  
    for k=1+passo:m-passo  
        if H(h,k)>M  
            alfa=angolo(Theta(h,k));  
            switch alfa  
                case 0  
                    for w=0:passo
```



```
        B(h-w,k)=B(h-w,k)+1;
    end
case 1
    for w=0:passo
        B(h-w,k-w)=B(h-w,k-w)+1;
    end
case 2
    for w=0:passo
        B(h,k-w)=B(h,k-w)+1;
    end
case 3
    for w=0:passo
        B(h+w,k-w)=B(h+w,k-w)+1;
    end
case 4
    for w=0:passo
        B(h+w,k)=B(h+w,k)+1;
    end
case 5
    for w=0:passo
        B(h+w,k+w)=B(h+w,k+w)+1;
    end
case 6
    for w=0:passo
        B(h,k+w)=B(h,k+w)+1;
    end
case 7
    for w=0:passo
        B(h-w,k+w)=B(h-w,k+w)+1;
    end
end
```

```
        end
    end
end
```

Approssimazione dell'argomento di $-\nabla f$

```
function alfa=angolo(beta)
if (beta>=-pi & beta<-7/8*pi) | (beta>=7/8*pi & beta<=pi)
    alfa=0;
elseif (beta>=-7/8*pi & beta<-5/8*pi)
    alfa=1;
elseif (beta>=-5/8*pi & beta<-3/8*pi)
    alfa=2;
elseif (beta>=-3/8*pi & beta<-1/8*pi)
    alfa=3;
elseif (beta>=-1/8*pi & beta<1/8*pi)
    alfa=4;
elseif (beta>=1/8*pi & beta<3/8*pi)
    alfa=5;
elseif (beta>=3/8*pi & beta<5/8*pi)
    alfa=6;
elseif (beta>=5/8*pi & beta<7/8*pi)
    alfa=7;
end
```

Selezione dei *Bins*

```
function R=MaxII(B,I,lambda)

[n,m]=size(B);
dx=[-1,0,1;-2,0,2;-1,0,1];
dy=[-1,-2,-1;0,0,0;1,2,1];
Bx=imfilter(B,dx);
By=imfilter(B,dy);
BB=abs(Bx+i*By);

x1=[0,0];
M=max(max(B));
for h=1:n
    for k=1:m
        if B(h,k)==M
            x1=[h,k];
        end
    end
end

x2=[0,0];
Jmin=realmax;
for h=1:n
    for k=1:m
        if BB(h,k)<=1
            J=(B(x1(1),x1(2))-B(h,k))^2+lambda*(x1(1)-h)^2;
            if J<Jmin
                Jmin=J;
                x2=[h,k];
            end
        end
    end
end
```

```
        end
    end
end

R=I;
for h=x1(1)-5:x1(1)+5
    for k=x1(2)-5:x1(2)+5
        R(h,k)=256;
    end
end
for h=x2(1)-5:x2(1)+5
    for k=x2(2)-5:x2(2)+5
        R(h,k)=256;
    end
end
```

Visualizzazione risultato

```
imshow(uint8(R))
```

Conclusioni

In questa tesi ho descritto alcuni sistemi che permettono a persone affette da sindrome *locked-in* di comunicare e interagire con l'ambiente. In particolare mi sono soffermato sui sistemi di *eye-tracking*, descrivendo le tecnologie e le tecniche sui quali si basano. Durante la trattazione è emerso l'importanza del livello di non invasività di tali sistemi e l'accuratezza dei risultati ottenuti; a questo scopo ho descritto il Metodo di *Zhu-Yang* che soddisfa entrambe le caratteristiche evidenziate.

Nella seconda parte di questa tesi ho descritto e implementato il Metodo di *Kothari-Mitchell*, che permette di localizzare la posizione dell'occhio all'interno di immagini o intere sequenze video. L'ambiente di lavoro che ho utilizzato è il Matlab. Durante lo sviluppo sono emersi alcuni fattori di rilievo che offrono spunti per il miglioramento della procedura, come ad esempio tecniche di configurazione automatica dei parametri *passo* e *lambda*.

Questa è solo una prima fase di ricerca, che potrebbe portare allo sviluppo di un sistema completo di comunicazione per persone in particolari condizioni di disabilità. Un primo passo sarebbe quello di implementare le tecniche di *eye-tracking* e *eye-detection* considerate in ambienti di lavoro più efficienti, mentre il secondo quello di correlare il sistema di un software che interpreti i segnali visivi.

Bibliografia

- [1] Bauby J.D., *Lo scafandro e la farfalla*, Ponte delle grazie, Milano, 1997
- [2] D. Scott & J. M. Findlay, *Visual Search, Eye Movements and Display Units, Human Factors Report*, University of Durham, UK, 1993
- [3] Jacob R.J.K., *The use of eye movements in human-computer interaction techniques: What you look at is what you get*, ACM Transactions on Information Systems 9(3), 152-169, 1991
- [4] Baluja S. & Pomerleau D., *Non-intrusive gaze tracking using artificial neural networks*, Research Paper CMU-CS-94-102, School of Computer Science, Carnegie Mellon University, Pittsburgh , USA, 1994
- [5] Müller P.U., Cavegn D., d'Ydewalle G. & Groner R. *A comparison of a new limbus tracker, corneal reflection technique, purkinje eye tracking and electro-oculography*, in G. d'Ydewalle & J. V. Rensbergen, eds, 'Perception and Cognition', Elsevier Science Publishers, B.V., pp. 393-401, 1993
- [6] Cleveland D. & Cleveland N., *Eyegaze eyetracking system*, in 'Imagina: Images Beyond Imagination. Eleventh Monte-Carlo International Forum on New Images', LC Technologies, Virginia, 1992
- [7] Wooding D. *Eye-Movement Monitoring Equipment*, Internet Document, University of Wales College of Medicine, Cardiff, UK, of <ftp://mailbase.ac.uk/pub/lists/eye-movement/files/equipment.txt> or <http://www.cf.ac.uk/uwcm/dr/groups/vision/em.txt>, 1995

-
- [8] Zhu J., Yang J. *Subpixel Eye Gaze Tracking*, Fifth IEEE International Conference on Automatic Face and Gesture Recognition , Carnegie Mellon University, Pittsburgh, 2002
 - [9] Trucco E., Verri A., *Introductory techniques for 3-D Computer Vision*, Prentice-Hall, USA, 1998
 - [10] Althof R.J., Wind M.G.J. and Dobbins III J.T., *A rapid and automatic image registration algorithm with subpixel accuracy*, IEEE Trans. Medical Imaging, 16(3): 308-316, 1997
 - [11] Thevenaz P., Ruttimann U.E. and Unser M., *A pyramid approach to subpixel registration based on intensity*, IEEE Trans. Image Processing, 7(1): 27-41, 1998
 - [12] Umeda K. and Takahashi T., *Subpixel stereo method: a new methodology of stereo vision*, International Conference on Robotics and Automation, pages 3215-3220, 2000
 - [13] Achard C., Bigorgne E. and Devars J., *A sub-pixel and multispectral corner detector*, International Conference on Pattern Recognition, pages 959-962, 2000
 - [14] Astrom K. and Heyden A., *Stochastic modeling and analysis of sub-pixel edge detection*, International Conference on Pattern Recognition, pages 86-90, 1996
 - [15] Hung D.C.D. and Mitchell O.R., *Subpixel edge estimation using geometrical edge models with noise miniaturization*, IEEE Southwest Symposium on Image Analysis and Interpretation, pages 112-117, 1994
 - [16] Jensen K. and Anastassiou D., *Subpixel edge localization and the interpolation of still images*, IEEE Trans. Image Processing, 4(3): 285-295, 1995

-
- [17] Kisworo M., Venkatesh S. and West G.A.W., *Detection of curved edges at subpixel accuracy using deformable models*, IEE Proceedings on Vision, Image, and Signal Processing, 142(5): 304-312, 1995
 - [18] Gorman L. O., *Subpixel precision of straight-edged shapes for registration and measurement*, IEEE Trans. Pattern Analysis and Machine Intelligence, 18(7): 746-751, 1996
 - [19] Kothari R., Mitchell J.L., *Detection of eye locations in unconstrained visual images*, IEE International Conference on Image Processing, vol III, Lausanne, 1996
 - [20] <http://www.diku.dk/~panic/eyegaze/article.html#contents>
 - [21] <http://www.bc.edu/eagleeyes>
 - [22] <http://www.eyegaze.com>
 - [23] http://www.handimatica.it/Handi2002/Atti_2002/28_11%20Convegno%20disabilita%20plurime/Malavasi.htm#Malavasi
 - [24] <http://www.axixa.com/au/id2.html>
 - [25] <http://www.tempomedico.it/news99/635lock.htm>
 - [26] http://www.ninds.nih.gov/health_and_medical/disorders/lockedinsyndrome_doc.htm

Ringraziamenti

Ringrazio la mia famiglia per l'opportunità che mi ha dato e la pazienza dimostrata in questi anni

Estrema gratitudine a Massimo Ferri che mi ha supportato e incoraggiato durante questi mesi di lavoro

Ringrazio Massimiliano Malavasi e l'intero gruppo dell'Ausilioteca per l'importante contributo e il continuo sostentamento

Ringrazio i miei amici di una vita, Fabio, Maurizio e Gianluca per non avermi mai abbandonato...gli amici di Bologna per i bei momenti trascorsi insieme

Un affettuoso ringraziamento a Tatiana per essermi stata vicina e avermi confortato nei momenti più difficili