

# Una piattaforma aperta per dispositivi palmari a basso costo a supporto dell'integrazione sociale e formativa degli studenti non vedenti

Filippo Battaglia<sup>1</sup>, Giancarlo Iannizzotto<sup>1</sup>, Lucia Lo Bello<sup>2</sup>

<sup>1</sup> Dipartimento di Scienze Cognitive, della formazione e degli studi culturali  
Università di Messina Via Concezione, 6-8, 98100 Messina  
[filbattaglia@libero.it](mailto:filbattaglia@libero.it), [ianni@unime.it](mailto:ianni@unime.it)

<sup>2</sup> Dipartimento di Ingegneria Elettrica, Elettronica e Informatica (DIEEI)  
Università di Catania V.le A. Doria, 6 95125 Catania  
[lucia.lobello@dieei.unict.it](mailto:lucia.lobello@dieei.unict.it)

*La disponibilità di moderni dispositivi handheld dotati di hardware ad elevate prestazioni, webcam e connettività wireless ne suggerisce l'impiego per lo sviluppo di applicazioni di ausilio per utenti non vedenti. Gran parte degli ausili attualmente disponibili per questo scopo è relativamente costosa, poco espandibile, basata su sistemi chiusi e poco integrabile. Appare quindi molto interessante la realizzazione di ausili basati su handheld 'consumer' che possano fornire un insieme di servizi integrati. Molte applicazioni di fatto esistono già, ma sono progettate e sviluppate per sistemi pc e pertanto non è possibile, o non è conveniente, operarne un porting direttamente sui dispositivi handheld. Con l'obiettivo di offrire una soluzione al problema, questo lavoro introduce Blind Assistant, un ambiente software per l'ausilio agli utenti non vedenti modulare, espandibile, open source e basato su una piattaforma software portabile, dotata di un toolchain standard che consente di semplificare le operazioni di scrittura e di porting del software verso piattaforme handheld non compatibili con gli ambienti per personal computer.*

## 1. Introduzione

In attesa dello sviluppo e della diffusione di dispositivi protesici in grado di sostituire completamente o almeno parzialmente l'occhio umano, la realizzazione di dispositivi portatili in grado di svolgere funzioni di riconoscimento visivo primario potrebbe contribuire in maniera significativa al miglioramento del livello di autonomia e di qualità della vita di molti soggetti non-vedenti o ipovedenti. Con particolare riferimento alla condizione degli studenti non vedenti, le opportunità offerte dall'adozione di strumenti informatici

sono molteplici, ed i potenziali benefici sono notevoli. Tali strumenti, infatti, non solo possono migliorare l'efficienza del lavoro didattico, ma possono dare un contributo fondamentale all'autonomia e all'autosufficienza dell'allievo non vedente ed alla sua inclusione nell'ambiente scolastico. Quest'ultima è una condizione abilitante per il successo dell'integrazione scolastica dello studente non vedente, che si declina in due dimensioni, ossia l'integrazione sociale, che fa sì che il non vedente si senta parte integrante del gruppo, e l'integrazione formativa, che fa sì che il non vedente possa sfruttare al massimo il suo potenziale di apprendimento per allargare lo spettro del proprio sapere ed acquisire nuove competenze [Sas09] [Age].

Funzioni di visione artificiale quali riconoscimento di fisionomie, lettura automatica di testi, rilevamento automatico della presenza di ostacoli, rilevamento e conteggio delle persone presenti possono già oggi essere realizzate con un'accuratezza adeguata ed in real-time con l'impiego di una telecamera ed un personal computer [BC01]. D'altra parte, sebbene sia plausibile dotare uno studente non vedente di un dispositivo delle dimensioni di un notebook pur di fornirgli funzioni percettive di alto livello ed utilità, fattori come peso, dimensioni ed autonomia restano un serio problema quando si vogliono realizzare degli ausili realmente mirati a fornire autonomia e mobilità.

Un secondo problema è quello del costo: se per gli ausili per persone diversamente abili la soglia media di spesa è tradizionalmente considerata più alta rispetto ad altre aree applicative, per motivi forse oggi non del tutto condivisibili [Nar06], è sicuramente vero che lo sviluppo di tecnologia a basso costo può sensibilmente abbassare la soglia per l'accesso alle tecnologie stesse, con evidenti ricadute sociali.

Nell'ambito dei dispositivi informatici, i costi della tecnologia si abbattano in maniera decisa quando si fa uso di dispositivi di ampia diffusione (dispositivi 'consumer'). D'altra parte, una piattaforma hardware capace di eseguire software di visione artificiale deve possedere una elevata potenza di calcolo: è richiesta la presenza di processori dotati di elevate prestazioni nei calcoli in virgola mobile, una memoria dinamica ed un'ampiezza di banda adeguate alla elaborazione di immagini di risoluzione sufficientemente elevata. La piattaforma hardware deve inoltre supportare una videocamera di buona qualità, sia in termini di compatibilità e velocità di comunicazione dell'interfaccia che di disponibilità dei driver.

La realizzazione di una piattaforma in grado di supportare uno studente non vedente nelle sue attività scolastiche quotidiane ha la necessità di acquisire ed elaborare, in tempo reale o con tempi molto brevi, immagini e video. Per questo tipo di elaborazioni è disponibile un'ampia varietà di software di elevato livello, anche open source. Ad esempio, la libreria OpenCV [BK08] include un'ampia gamma di funzioni di visione artificiale come il riconoscimento di persone ed il rilevamento di ostacoli ed offre le basi per lo sviluppo di altre funzioni avanzate come il riconoscimento di ambienti. Essa è ufficialmente supportata per Microsoft Windows, Linux, Apple desktop e, recentemente, anche Android. Questa disponibilità potrebbe suggerire la piattaforma Android come preferenziale per lo sviluppo delle applicazioni necessarie, ma nella maggior parte dei casi i dispositivi esistenti sul mercato e basati su questa piattaforma

presentano caratteristiche hardware insufficienti o, in caso contrario, costi molto elevati. Questa considerazione è generalizzabile: la maggior parte delle piattaforme handheld con caratteristiche prestazionali sufficientemente elevate da consentirne l'impiego per l'analisi delle immagini in tempo reale appartengono alla fascia più elevata degli smartphone (e pertanto sono molto costose) oppure alla fascia *gaming*, come ad esempio la Sony PSP.

Molto spesso i software disponibili per la realizzazione delle funzioni richieste per un valido ausilio a studenti non vedenti sono diversi ed eterogenei e ciascuno di questi software ha una propria sezione del codice specifica per la piattaforma di utilizzo. Pertanto, se si desidera utilizzare questi software su una piattaforma diversa da quella nativa (che è di solito un pc), è in genere necessario eseguire, per ciascuno di essi, una laboriosa operazione di porting che va ripetuta ad ogni aggiornamento dei software utilizzati. Ad esempio l'ambiente di sviluppo di Android, sebbene questo sistema operativo sia basato su kernel Linux e sia open source, non consente il porting diretto (quindi semplice) degli applicativi sviluppati in ambito desktop pc.

In conclusione, se da un lato si può senza dubbio decidere di sviluppare unicamente su piattaforme che supportino Linux o altri sistemi operativi su cui sia possibile il porting diretto delle applicazioni e delle librerie di supporto, è necessario confrontarsi con il fatto che molte, forse la maggior parte, delle piattaforme handheld di costo ragionevole e di prestazioni elevate appartengono al mercato gaming e non supportano Linux o sono molto costose e che, in ogni caso, il porting di applicativi scritti per pc verso tali piattaforme non è semplice.

Malgrado i problemi qui evidenziati, in passato sono stati realizzati diversi esempi di applicazioni in ausilio agli utenti non vedenti [MTFD+07][MSB98][DSM02][IT99][KS97]. Purtroppo, nella maggior parte dei casi, i sistemi che vengono proposti come ausili per i non vedenti non sono piattaforme di sviluppo, ma specifiche applicazioni (in genere closed source) che girano su dispositivi dedicati. Tali sistemi, quindi, sono chiusi allo sviluppo ulteriore da parte di terze parti ed alla integrazione fra loro, a meno di espliciti accordi di collaborazione con il produttore originario. E' chiaro allora che sarebbe particolarmente vantaggioso potere sviluppare un framework open source che consenta di svolgere le stesse elaborazioni, ma su dispositivi palmari di ultima generazione o su console per videogiochi portatili, in modo da poter sfruttare la potenza di calcolo di questo tipo di dispositivi, la presenza di webcam integrate, di connessioni di rete e di sensori GPS, insieme ai bassi costi che derivano dalla produzione delle unità in milioni di pezzi ed in un regime di forte concorrenza tra i produttori. Tale framework consentirebbe l'integrazione di più funzionalità, sviluppate ed ottimizzate da sviluppatori differenti, in un unico programma open source e totalmente gratuito.

In questo lavoro viene presentato Blind Assistant, una piattaforma open source per il supporto agli studenti non vedenti nel loro inserimento nell'ambito scolastico e per le loro attività quotidiane. Blind Assistant è costruito su Nanodesktop, un framework anch'esso open source che consente lo sviluppo ex-novo o il porting estremamente semplificato di applicazioni da ambiente desktop ad ambiente handheld. Il sistema complessivo che ne risulta è

modulare ed integrabile, è economico ed efficiente e facilmente adattabile alle esigenze specifiche degli studenti.

Nel seguito, la Sezione 2 richiama alcune caratteristiche fondamentali del framework Nanodesktop, in Sezione 3 viene descritta la piattaforma Blind Assistant e, infine, la Sezione 4 conclude la trattazione ed introduce alcuni sviluppi futuri del lavoro svolto.

## **2. Nanodesktop**

Le considerazioni precedenti hanno portato allo sviluppo di una piattaforma software chiamata Nanodesktop [BIR09], finalizzata allo sviluppo di applicazioni di visione artificiale su sistemi hardware non-standard per i quali non siano disponibili un kernel Linux pienamente funzionale ed una catena di compilazione (toolchain) sufficientemente standard, come ad esempio i dispositivi basati su sistemi operativi come il PSP-OS della Sony, Symbian-OS o Android, che ha un kernel Linux ma una toolchain non direttamente compatibile con le applicazioni scritte per sistemi desktop.

La funzione di Nanodesktop è quella di astrarre alle librerie poste agli strati superiori il comportamento dell'hardware. In questo modo, quando si desidera portare le proprie librerie da una piattaforma ad un'altra, è sufficiente una semplice ricompilazione del codice. E' la tecnologia Nanodesktop che si occupa di convertire le chiamate in un formato compatibile con il sistema operativo e con la macchina in cui in quel momento l'algoritmo è in esecuzione.

Il progetto Nanodesktop nacque, 6 anni fa, con il solo obiettivo di consentire un porting della libreria OpenCV su dispositivi diversi da quelli supportati ufficialmente; successivamente esso si è esteso molto, ed oggi esistono versioni Nanodesktop di OpenCV, del software di sintesi vocale Flite [TBC98], del software di riconoscimento vocale PocketSphinx [HDKC+06] [LHR90], di librerie VOIP (OSIP2 e eXOSIP), di SQLite, di DevIL, di FreeType, di CURL ed una versione dell'interprete Python.

## **3. Blind Assistant**

Per uno studente non vedente un sistema tascabile in grado di eseguire dei sofisticati algoritmi di visione artificiale può essere di grande aiuto. Un sistema di riconoscimento facciale può identificare i nomi e le posizioni delle persone note che sono presenti in quel momento nelle vicinanze. Un sistema di riconoscimento visivo del luogo, magari in sinergia con un sistema GPS, può consentire al non vedente di sapere in quale ambiente si trova in quel momento. Un OCR, usato in combinazione con un software di sintesi vocale, può leggere al non vedente il testo che viene in quel momento rilevato dalla videocamera, compreso il contenuto di un cartello o di una targa sul muro.

Blind Assistant è un'applicazione di assistenza ai non vedenti, il cui codice sorgente è rilasciato sotto licenza LGPL ed il cui primo prototipo utilizza come hardware la console tascabile Sony Playstation Portable. Esso svolge 6 funzioni, tutte volte a facilitare sia l'attività formativa che l'inclusione dello studente non vedente nel suo gruppo "sociale":

1. Riconoscimento facciale
2. Riconoscimento di ambienti
3. Riconoscimento di tag con codici a barre, DataMatrix e QR-Code
4. Riconoscimento caratteri (OCR)
5. Riconoscimento colori
6. Lettura posta elettronica.

Allo scopo di consentire il controllo allo studente non vedente, l'applicazione comprende un sistema di riconoscimento vocale basato sul software open source PocketSphinx ed un sistema di sintesi vocale basato sul software open source Flite (si veda Fig. 1). Il software presuppone che ci sia un educatore che provveda alla configurazione iniziale dei vari sistemi: tutte le impostazioni vengono salvate in memoria e vengono ripristinate automaticamente all'avvio.

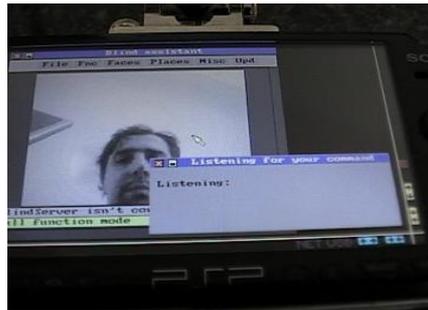


Fig. 1: Schermata di Blind Assistant in attesa di un comando vocale.

### 3.1 Riconoscimento facciale

Un sistema di riconoscimento facciale è un software in grado di riconoscere le persone che sono presenti in una foto o in video. Normalmente questi sistemi di riconoscimento sono eseguiti da dispositivi molto sofisticati e dotati di una elevata potenza di calcolo.



Fig. 2: Schermata di Blind Assistant in modalità riconoscimento facciale.

Nel caso del Blind Assistant, invece, è stato possibile eseguire un algoritmo di riconoscimento facciale su un piccolo processore embedded come il MIPS della PSP (si veda Fig. 2). Il sistema di riconoscimento facciale di Blind Assistant è in grado di operare su un set di massimo 10 persone. Per ogni persona viene preventivamente eseguita una fase di training, durante la quale viene acquisito un set di 8 immagini. Il sistema guida il personale di assistenza attraverso un wizard e pertanto anche la fase di training è molto semplice. I dettagli della tecnologia utilizzata sono fuori dallo scope di questo lavoro. In breve, l'immagine viene sottoposta prima ad una normalizzazione della luminosità, poi ad un processo di face detection per mezzo dell'algoritmo di Viola-Jones [VJ01], ed infine ad un processo di face recognition per mezzo dell'algoritmo PCA (Principal Component Analysis) [TP91].

Il software determina se uno dei volti noti è presente nell'immagine e la sua posizione. I dati vengono inseriti in un registro presenze: solo quando il soggetto risulta presente in 3 frame consecutivi, il programma provvede a riprodurre un messaggio che avverte l'utente della presenza del soggetto e della sua posizione media ("Mario è di fronte a te", "Mario è alla tua sinistra", "Mario è alla tua destra").

### 3.2 Riconoscimento di ambienti

Blind Assistant è in grado di riconoscere l'ambiente (ad esempio l'aula) che viene inquadrato dalla webcam, per consentire al non vedente di orientarsi anche in mancanza di un adeguato *tagging* degli ambienti. Solo per questa specifica applicazione, allo scopo di ottenere prestazioni di tipo online (quasi in tempo reale), si è preferito appoggiarsi ad un server esterno (un comune personal computer) con il quale l'handheld si collega via rete wireless (Fig. 3).

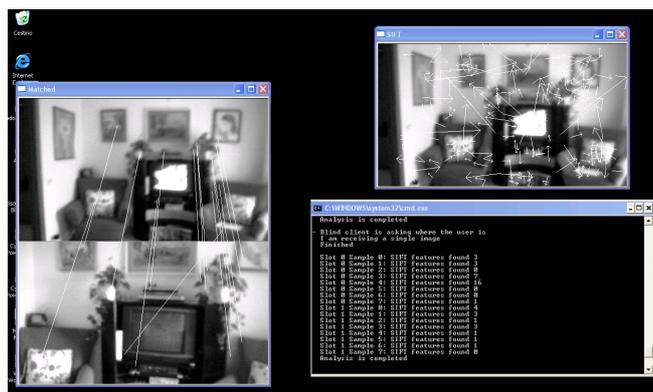


Fig. 3: Blind Assistant (lato server) in fase di riconoscimento di un ambiente.

Il problema del riconoscimento degli interni di un edificio articolato come una scuola è abbastanza complesso. In letteratura si è osservato che uno degli algoritmi più robusti che possono essere utilizzati nel processo di riconoscimento degli interni di una stanza è quello basato su [AB07].

### 3.3 Modalità riconoscimento caratteri (OCR)

Blind Assistant possiede anche la capacità di leggere un testo allo studente non vedente, per mezzo della videocamera associata (si veda Fig. 4).

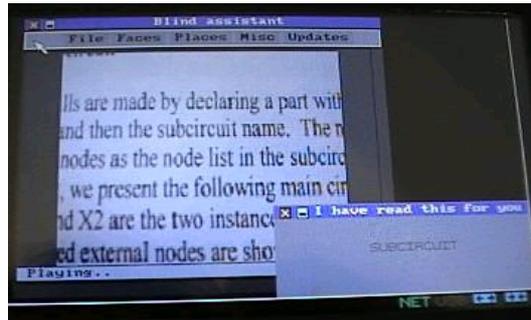


Fig. 4: Schermata di Blind Assistant in modalità OCR

Lo studente non vedente preme un tasto specifico sulla console e pronuncia l'ordine: "Abilita OCR". A questo punto, Blind Assistant commuta nella modalità OCR ed inizia a catturare le immagini ad alta risoluzione. Quando si vuole leggere un testo, si posiziona la console di fronte ad esso e si preme un secondo tasto sulla console: un'immagine ad elevata risoluzione viene catturata dalla videocamera. Questa viene poi sottoposta ad OCR direttamente sulla PSP. Il testo ottenuto in genere contiene degli errori grammaticali, quindi viene usato un algoritmo di correzione dei termini basato su un dizionario. Infine, i risultati vengono letti al non vedente per mezzo del sintetizzatore vocale integrato.

### 3.4 Riconoscimento colori

Ci sono dei casi in cui lo studente non vedente può volere sapere di che colore è un oggetto di fronte a sé, ad esempio una porta. Per risolvere questo problema, Blind Assistant include un software che verifica qual è il colore dominante dei pixel presenti nell'immagine catturata dalla webcam. Per abilitare il riconoscimento colori, l'utente deve premere un tasto e deve pronunciare il comando "Abilita riconoscimento colori". Premendo poi un secondo tasto, viene avviata un'analisi dei colori presenti nel frame inquadrato. Il sistema determina il colore dominante tramite un confronto a massima verosimiglianza dei valori RGB rilevati con quelli di una look-up table. Il risultato viene infine comunicato al non vedente per mezzo del sintetizzatore vocale.

### 3.5 Lettura della posta elettronica

Blind Assistant può essere usato anche per servizi che sfruttano il suo sintetizzatore vocale, il suo sistema di riconoscimento del parlato e la connessione wireless della PSP per fornire al non vedente funzioni informative. Questa possibilità è molto importante per un non vedente, perché gli consente di ricevere informazioni e messaggi senza la necessità di usare un personal

computer. Per attivare la lettura della posta elettronica, lo studente non vedente deve premere un tasto e pronunciare il comando "Abilita posta elettronica".

### 3.6 Lettura codici a barre

Un lettore di codici a barre (barcode reader) può aiutare un non vedente a riconoscere un oggetto od un ostacolo se questo è adeguatamente marcato con una etichetta barcode. Si può anche impiegare una versione più sofisticata di questo approccio: un codice a barre bidimensionale, basato su uno standard aperto come il Datamatrix o il QR-code, può offrire maggior robustezza al rumore, correzione automatica di errore e rilevamento automatico nel video. Blind Assistant integra un modulo lettore sia per Datamatrix che per QR-code, oltre che per i codici a barre più comuni (si veda Fig. 5).



Fig. 5: Una schermata di Blind Assistant in modalità lettura codici a barre.

Nei nostri test il sistema è stato in grado di rilevare e leggere un'etichetta di 5x5 cm ad una distanza massima di 80 cm (con la telecamera standard offerta dal produttore del dispositivo: un sensore migliore avrebbe prodotto risultati migliori). Il comportamento del sistema è invariante rispetto alle rotazioni della telecamera e ad ampie variazioni di posa. La frequenza di rilevazione e lettura è di 1Hz. L'etichetta viene rilevata automaticamente quando presente nel video acquisito: in questo caso essa viene decodificata ed i dati in essa codificati vengono passati al sintetizzatore vocale Flite, che li legge all'utente.

### 3.7 Valutazione delle prestazioni

Per verificare l'efficacia della soluzione proposta, è stato condotto un test su 15 studenti, ciascuno dei quali ha indossato degli occhiali speciali che, oscurando la visuale, simulano la condizione di cecità. Ogni partecipante ha testato ciascuna delle funzioni del sistema, quindi ha compilato un questionario contenente domande sulle sensazioni avute durante l'uso, sulle funzioni giudicate efficaci e non, e sulle funzioni mancanti che dovrebbero essere aggiunte. La valutazione dell'efficacia delle singole funzioni è stata espressa su una scala da 0 a 5.

Il sistema di riconoscimento del viso è stato considerato abbastanza efficace (punteggio medio  $M=3.82$ ), malgrado il tasso di riconoscimento non fosse allo stato dell'arte (circa l'80%). Il sistema OCR, il sistema di lettura delle email ed il

sistema di lettura dei codici a barre sono stati considerati molto efficaci (rispettivamente  $M=4.12$ ,  $M=3.89$ , e  $M=4.79$ ). Al contrario, il sistema di riconoscimento degli ambienti è stato considerato non particolarmente efficace ( $M=2.21$ ), a causa della sensibilità dell'algoritmo usato alle variazioni di luminosità.

Infine, il sistema di riconoscimento dei colori è stato considerato efficace, ma non particolarmente utile per gli obiettivi della piattaforma.

#### **4. Discussione e sviluppi futuri**

Questo articolo presenta Nanodesktop e Blind Assistant, rispettivamente una piattaforma per lo sviluppo ed il porting di software su dispositivi handheld ed un sistema modulare ed espandibile in ausilio a studenti non vedenti.

Nel lavoro sono descritte le principali problematiche ed i requisiti da tenere in considerazione nel progetto di sistemi portatili orientati all'ausilio di studenti non vedenti e la piattaforma costituita da Nanodesktop e Blind Assistant viene proposta come una possibile soluzione. Blind Assistant e Nanodesktop costituiscono una piattaforma altamente portatile e versatile, open source e dotata di strumenti di compilazione standard, che si propone l'obiettivo di colmare il gap fra i dispositivi esplicitamente pensati e progettati per l'ausilio ai non vedenti, spesso costosi e poco integrabili fra loro, ed i dispositivi consumer, meno costosi nonostante le elevate prestazioni e più diffusi sul mercato.

La leggerezza e l'ingombro limitato del dispositivo handheld, unite ai costi non proibitivi, suggeriscono l'adottabilità di Blind Assistant nelle scuole a supporto dell'integrazione sociale formativa degli studenti non vedenti. A tal fine, è necessaria una stretta integrazione con gli end-user e i vari stakeholder, educatori, docenti, operatori del sociale, per tarare opportunamente le applicazioni e le caratteristiche dell'interfaccia utente. Inoltre è in fase di definizione una campagna di test e valutazione sul campo delle prestazioni e dell'usabilità della piattaforma presentata.

Nanodesktop è un software in continua evoluzione. Per le prossime release si sta valutando la possibilità di un porting sulle piattaforme Android e iPhone-Os. Questo permetterebbe il porting del software Blind Assistant su dispositivi dotati di una potenza di calcolo molto superiore rispetto alla Sony PSP. Se, da un lato, allo stato attuale queste piattaforme sono molto più costose della PSP, la disponibilità di una simile potenza di calcolo permetterebbe l'esecuzione, all'interno del Blind Assistant, di più complessi algoritmi di visione artificiale, come il alcune applicazioni 3D. Diverrà inoltre possibile usufruire del sensore GPS integrato.

Per quanto riguarda le nuove funzioni, si stanno valutando gli algoritmi in letteratura per il riconoscimento in tempo reale delle facce più adatti in termini di accuratezza e di complessità computazionale e si sta lavorando sul porting di una serie di applicazioni orientate alle attività formative. Infine è in fase di studio

la possibilità di aggiungere un driver per sensori ad ultrasuoni da connettere attraverso la porta per periferiche, realizzando un sistema di collision avoidance che potrebbe essere usato dallo studente non vedente nelle situazioni di affollamento e per evitare gli ostacoli.

## Bibliografia

- [AB07] B. Ayers and M. Boutell, "Home interior classification using sift keypoint histograms," *Computer Vision and Pattern Recognition*, 2007, pp. 1–6, Jun. 2007.
- [Age] Agenzia Nazionale per lo Sviluppo dell'Autonomia Scolastica. Handitecno. [Online]. Available: <http://handitecno.indire.it/>
- [BC01] W. Barfield and T. Caudell (Eds.), "Fundamentals of wearable computers and augmented reality," Routledge Publishers, Jan 2001.
- [BIR09] F. Battaglia, G. Iannizzotto, and F. L. Rosa, "An open and portable software development kit for handheld devices with proprietary operating systems," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 4, pp. 2436–2444, 2009.
- [BK08] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*. Cambridge, MA: O'Reilly, 2008.
- [DSM02] L. Di Stefano and S. Mattocchia, "Real-time stereo within the videt project," *Real-Time Imaging*, vol. 8, no. 5, pp. 439–453, Oct. 2002.
- [HDKC+06] D. Huggins-Daines, M. Kumar, A. Chan, A. W. Black, M. Ravishankar, and A. I. Rudnicky, "Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices," *Proc. of the IEEE Conf. on Acoustics, Speech, and Signal Processing*, pp. 185–188, may 2006.
- [IT99] D. Ikeya and J. Takeno, "Research and development of a hand-held vision system for the visually impaired," 8th IEEE International Workshop on Robot and Human Interaction, pp. 13–17, Sep. 1999.
- [KS97] I. Kopecek and P. Smrz, "Navigation and information system for visually impaired people," Tech. Report, Masryk University, Faculty of Informatics, May 1997.
- [LHR90] K.-F. Lee, H.-W. Hon, and R. Reddy, "An overview of the sphinx speech recognition system," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, no. 1, pp. 35–45, jan 1990.
- [MSBN98] N. Molton, S. Se, and J. Brady, "A stereo vision-based aid for the visually impaired," *Image and Vision Computing*, vol. 16, no. 4, Apr. 1998.
- [MTFD+ 07] C. Mancas-Thillou, S. Ferreira, J. Demeyer, C. Minetti, and B. Gosselin, "A multifunctional reading assistant for the visually impaired," *J. Image Video Process.*, vol. 2007, no. 3, pp. 1–11, 2007.
- [Nar06] P. Narasimhan, "Assistive embedded technologies," *IEEE Computer*, vol. 39, no. 7, pp. 85–87, jan 2006.
- [Sas09] L. D. Sasanelli. (2009, Mar) *Tecnologie didattiche per l'integrazione: Un portale per la scuola*. <http://www.rivistadidattica.com/metodologia/metodologie/92.htm>
- [TBC98] P. Taylor, A. Black, and R. Caley, "The architecture of the festival speech synthesis system," *Proceedings of The Third ESCA Workshop in Speech Synthesis*, pp. 147–151, 1998.
- [TP91] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *IEEE Computer Vision and Pattern Recognition*, 1991. pp. 586-591.
- [VJ01] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 511, 2001.