



UNIVERSITÀ DI BRESCIA
FACOLTÀ DI INGEGNERIA
Dipartimento di Elettronica per l'Automazione

Laboratorio di Robotica Avanzata
Advanced Robotics Laboratory

Corso di Robotica
(Prof. Riccardo Cassinis)

Web-camera per
LDRA

Elaborato di esame di: **Alessandro Saetti, Sergio Spinoni**

Consegnato il: **30 agosto 2001**

Sommario

L'elaborato consiste nella produzione di una pagina HTML da inserire nel sito internet di "Robotica", che mostri continuamente cosa sta avvenendo nel laboratorio stesso mediante l'utilizzo di una web-camera.

1. Introduzione

Il laboratorio di Robotica Avanzata (LRA) è costituito di un paio di stanze dove gli studenti possono liberamente recarsi. E' ora possibile grazie a questo elaborato avere una visione di quanto succede in ogni minuto nel suddetto laboratorio. Dal sito di Robotica (www.ing.unibs.it/~cassinis) è possibile trovare un link a una pagina (www.frank.ing.unibs.it/webcam/arl_online.html) contenente le immagini di quanto accade nel laboratorio.

Il lavoro è durato circa 1 mese e mezzo. Un terzo di questo tempo è stato impiegato navigando in Internet per cercare il software necessario; la restante parte è stata impiegata per la realizzazione, in base al materiale raccolto e alle web-camera a disposizione, di quanto richiesto.

2. Il problema affrontato

Il problema affrontato consiste, come precedentemente accennato, nella produzione di una pagina HTML contenente l'immagine del laboratorio di robotica. Inoltre da tale pagina dovrà essere possibile accedere a una piccola collezione storica delle ultime immagini catturate del suddetto laboratorio.

La web-camera deve essere collegata a "NoFive", uno dei calcolatori all'interno della rete del laboratorio e collocato nel laboratorio stesso, mentre il sito di robotica si trova invece su "Frank". Perciò ci sono diverse soluzioni per rendere reperibile detta pagina.

E' interessante notare che se "NoFive" è un PC sul quale è installato Linux, "Frank" è invece un calcolatore Mac con il suo corrispondente sistema operativo. Un'altra macchina che abbiamo avuto a disposizione prende il nome di "Brad" che, come "Nofive", è un PC con un sistema operativo Linux.

Per potere giungere a questo risultato il nostro lavoro si è basato sui seguenti punti:

- Scelta di una web-camera opportuna
- Reperimento in Internet del software
- Installazione della web-camera
- Configurazione di Linux
- Creazione di script e pagine HTML

2.1. Scelta di una web-camera opportuna

Questa scelta è stata molto più ardua di quanto si potesse pensare e, se non fosse stato per gli stessi utenti di Linux disseminati in tutto il mondo, sarebbe potuta anche concludersi malamente. Infatti nessuna tra le maggiori case produttrici di web-camera si preoccupa di sviluppare dei driver per Linux e, cosa ancora più grave, non intendono divulgare le specifiche necessarie per la realizzazione “fai da te” dei primi. Immaginiamo che il motivo per cui non vengano prodotti driver per Linux sia la velocità con cui nascono nuovi kernel di tale S.O.. Produrre driver per Linux richiederebbe probabilmente un continuo lavoro da parte degli sviluppatori delle case produttrici in questione. D'altra parte un investimento in tal senso porterebbe un aumento dei loro potenziali clienti visto che gli utenti di Linux sparsi per il mondo non sono certo pochi. Il motivo per cui non intendono neppure divulgare le specifiche tecniche riguardanti le proprie web-camera potrebbe essere la necessità di proteggere i propri prodotti dalla concorrenza delle altre case produttrici. Questo nostro parere è confermato dal fatto che in Internet c'è una grande quantità di driver per Linux, ma tutti questi sono inerenti a web-camere oramai obsolete; tanto obsolete da essere fuori produzione. Ci siamo concessi queste considerazioni per poter dissentire dalla politica di queste case produttrici.

Tornando al problema propositoci è stata richiesta una web-camera a colori con una risoluzione non eccessiva.

La maggior parte delle web-camera su mercato soddisfano a questi requisiti. Possiamo ad esempio citare le quickcam prodotte dalla Logitech (www.logitech.com/cf/products/cameras.cfm), le Creative VideoBlaster prodotte dalla Creative (www.europe.creative.com/products/category.asp?lang=6&cate=6) oppure ancora le PcCameras prodotte dalla Philips (www.philips.com/productoverview/pcperipherals.html) oppure www.pc-cameras.philips.com); solo per citare alcune tra le case produttrici più importanti.

La nostra scelta a questo punto si è dovuta concentrare non tanto sui requisiti hardware ma sul software a disposizione in Internet.

2.2. Reperimento in Internet del software

Indirizzi utili per informazioni, software da scaricare e link inerenti sono stati trovati combinando sui più importanti motori di ricerca (www.altavista.com, www.google.com, ecc...) le parole chiave “web-camera”, “quickcam”, “Linux”, “driver”, “Grabber”. Per potere ripercorrere il cammino da noi compiuto più agevolmente in futuro citiamo alcuni siti interessanti.

Nel sito “Web-camera mit Linux” (www.multimedia4linux.de/webcam.html) si trovano una serie di driver per web-camera: molte di esse sono fuori produzione mentre molte altre sono di marche sconosciute.

“Video for Linux resources” (www.exploits.org/v4l) è un sito contenente link a driver, tool e ultime notizie riguardanti il problema del funzionamento di una web-camera sotto Linux. Da tale sito si riesce ad accedere a una pagina molto interessante che prende il nome di “CPiA webcam driver” (<http://webcam.souceforge.net>) contenente i driver per web-camera che adottano il chip CPiA. Tra tali web-camera rientrano la Creative Blaster Webcam II della Creative che purtroppo sono fuori produzione. Quest'ultima è stata soppiantata dalla Creative Blaster Webcam III non ancora supportata dal software rintracciabile su tale sito.

Nonostante ciò navigando in questo sito (<http://sourceforge.net>) abbiamo trovato una grande varietà di progetti inerenti a web-camere.

Un elenco di molti progetti si può trovare digitando: http://sourceforge.net/softwaremap/trove_list.php?form_cat=126.

Ad esempio troviamo all'indirizzo <http://sourceforge.net/projects/webcam> la possibilità di scaricare driver per CPiA webcam, all'indirizzo <http://sourceforge.net/projects/homeconnectusb> un progetto concernente driver per la 3COM's Home Connect Web-camera e all'indirizzo <http://sourceforge.net/softwaremap/spca50x> dei driver per la Intel Pc Camera Pro, tuttora in produzione.

Incoraggiati da tale scoperta abbiamo continuato a navigare nel sito e abbiamo scoperto dei driver per la Logitech Quickcam Express (<http://sourceforge.net/projects/qce-ga>) e per la Logitech Quickcam VC (<http://sourceforge.net/projects/usb-quickcam-VC>).

Un altro driver relativo a tale web-camera è rintracciabile alla pagine (<http://www.bode.cs.tum.edu/~acher/quickcam/quickcam.html>) "Linux Driver for Logitech Quickcam Express".

Una descrizione dei tool supportati dal primo di questi due driver relativi alla Quickcam Express si trova alla pagina <http://qce-ga.sourceforge.net/en-index.html>.

Un altro sito da tenere in considerazione è www.smcc.demon.nl/webcam/index.html. In esso si trovano i supporti per una grande quantità di telecamere della Philips ancora in produzione; tra esse citiamo: Vesta, Vesta Pro, Vesta Scanner, ToUCam Fun e ToUCam Pro. Vi è la possibilità di scaricare il software che le supporta relativo a varie versioni del kernel di Linux e di apprendere notizie e informazioni utili all'installazione.

Inoltre da quest'ultimo sito è possibile arrivare alla pagina: <http://members.brabant.chello.nl/~j.vreeken/se401> dove si trova un driver in grado di supportare quattro web-camera della Kengsinton tuttora in produzione e la Philips Vesta Fun.

Altro sito nel quale si trovano interessanti link è <http://home.texoma.net/~abark/webcam/index.html>. Da esso si riesce ad accedere ai siti "Quickcam Third Party Driver" (www.crynwr.com/qcpc) dove si trovano driver per la Logitech e Connectix Quickcam (PCParallel) e informazioni utili su come condurre il Reverse Engineering per tale web-camera. Si riesce inoltre ad accedere al sito "Third Party Software" (www.cs.duke.edu/~reynolds/quickcam). Questo ultimo sito in particolare è da tenere in alta considerazione perché in esso c'è una vasta collezione di software relativo al funzionamento della Quickcam in dotazione al laboratorio: frame-grabber (tra i quali citiamo Cqcam e Gqcam), tool per la videoconferenza e per la gestione dello streaming, ed esempi di script scritti in Perl.

Il software principale prende il nome di "Cqcam" ed è reperibile presso l'indirizzo Internet: www.cs.duke.edu/~reynolds/cqcam. In tale sito si trovano HOW-TO, FAQ, link relativi al software necessario per il funzionamento del pacchetto e la possibilità di scaricare il programma in questione, che naturalmente si trova anche nel CD relativo a questo elaborato.

Cqcam è un pacchetto software contenente sia i driver che i tool in grado di acquisire un'immagine dalla web-camera e metterla a disposizione su file.

Da tale sito abbiamo inoltre appreso l'esistenza di un tool di nome "Stamp" in grado di marciare un'immagine con stringhe ed eventualmente spedirla via ftp. Per una trattazione più approfondita di questi due tool rimandiamo il lettore al capitolo 4.

Purtroppo da tale sito non si riesce ad accedere alla pagina dalla quale si riesce a scaricare Stamp per una questione di aggiornamento mancato. Abbiamo segnalato tale mancanza e speriamo che, se necessario in futuro, si possa accedere a tale pagina anche da tale sito. Ad ogni modo l'utility in questione si riesce a scaricare all'indirizzo <http://stamp.sourceforge.net/index.html>. In tale sito si trovano spiegazioni, FAQ e la possibilità di scaricare il software in questione, che ad ogni modo si troverà anche nel CD allegato.

Citiamo inoltre il tool "Gqcam", utility del tutto analoga a Cqcam, ma in grado di lavorare in ambiente X scaricabile dal sito: <http://cse.unl.edu/~cluening/gqcam>.

Segnaliamo inoltre il sito: "mini HOW-TO" (<http://mosquito.dhs.org/mini-HOWTO-cqcam.html>) contenente informazioni molto utili riguardanti i settaggi hardware e software necessari, l'utilizzo di Cqcam, link a tool per la videoconferenza e il movie-grabber, esempi di script elementari e altro.

2.3. Installazione della web-camera

Dopo aver scelto la web-camera opportuna abbiamo dovuto collegare la web-camera al calcolatore e installare l'apposito software per il suo funzionamento. Ci proponiamo di parlare di questo punto più approfonditamente nei prossimi capitoli.

2.4. Configurazione di Linux

Dopo aver installato la web-camera ci siamo dovuti porre il problema di come rendere reperibile la pagina HTML a Frank, il Web-Server contenente il sito di Robotica. Questo problema è stato più difficile a causa della natura di "Frank": esso è un Mac.

Una prima soluzione è stata quella di lasciare sia la pagina HTML che le varie immagini grabbate su "Nofive" rendendolo quindi a sua volta un Web-Server.

Su "Nofive", dopo aver configurato opportunamente Apache, è stato lanciato il demone Httpd. In tal modo da Internet è stato reso accessibile "Nofive" sotto l'indirizzo <http://nofive.ing.unibs.it>. A tale indirizzo corrisponde su "Nofive" la directory `/home/NoFive/http/html/`. Per cui depositando la nostra pagina HTML in tale directory e mettendo nel sito di robotica un opportuno link alla pagina (http://nofive.ing.unibs.it/LRA_on-line.html) sia essa che le immagini sarebbero disponibili.

Questa è senz'altro la soluzione più semplice; solitamente "Nofive" non ha grandi compiti da assolvere quindi non gli viene aggiunto un carico di lavoro troppo esasperante. Non ha nessuna rilevanza ai fini del problema affrontato pensare al fatto che "Nofive" una volta spento non potrebbe più svolgere questo suo nuovo compito. Infatti, anche se la pagina HTML si trovasse su un altro calcolatore, essendo "Nofive" spento anche la web-camera lo sarebbe e quindi non sarebbero fornite nuove immagini. L'unico svantaggio a questa soluzione è che "Nofive", per sua natura orientata alla ricerca, non ha nessuna garanzia di continuare a funzionare in futuro come http-server.

Questa prima soluzione è stata subito implementata da noi anche solo semplicemente per condurre delle prove riguardo alla pagina HTML stessa e alla web-camera.

Un'altra possibilità è tenere la pagina HTML su "Frank" e spedire le immagini residenti su "Nofive" via FTP a "Frank".

Questo naturalmente deve essere fatto in maniera trasparente all'utente di "Nofive" (il compito deve essere svolto da uno script e ogni utente non deve accorgersi del lavoro che nel frattempo il calcolatore svolge) quindi è necessario aggirare la fase di login richiesta dal protocollo FTP.

La fase di login può essere aggirata attraverso l'auto-login. Per fare ciò è necessario creare un file nella propria home-directory di nome *.netrc*. Questo file deve contenere delle stringhe nel seguente formato:

machine nome_host login nome_user password password_user

A questo punto digitando: *ftp frank* sarà direttamente aperta una connessione su "Frank" senza richiedere né login né password.

Attenzione ad ogni modo ai privilegi del file ".netrc"; essendoci depositata la password dell'utente esso deve essere leggibile solo da parte di quest'ultimo. Questo per mantenere un certo livello di sicurezza dell'intero sistema. Se così non fosse fatto Ftp si rifiuterebbe di aprire la connessione esordendo con un messaggio del tipo:

Error - .netrc not correct permission.

Per impostare correttamente i privilegi è necessario digitare:

chmod 400 .netrc

A questo punto digitando:

ls .netrc -l

dovrebbe apparire:

-r----- 1 user user data .netrc

In tal modo solo il proprietario del file potrà leggerlo e le informazioni riguardanti le sue password saranno al sicuro.

Una volta aperta una connessione è possibile spedire e ricevere file utilizzando le pipe.

Ad esempio si possono scrivere i comandi che si vogliono eseguire su "Frank" nel file "prova" e digitare:

ftp frank < prova

Il file "prova" nel nostro caso dovrebbe contenere i seguenti comandi:

```
binary
send mycam.jpg
close
```

ciò provocherebbe la spedizione del file "mycam.jpg" a "Frank".

Non volendo utilizzare le pipe è comunque possibile ricorrere ad altri metodi. Abbiamo scaricato da Internet un programma, a cui abbiamo già in precedenza fatto riferimento, di nome "Stamp" che tra le altre sue finalità ha la capacità di spedire via FTP i file. A causa però del sistema operativo di "Frank" abbiamo trovato delle difficoltà in quanto esso non permette a STAMP di sovrascrivere dei file.

Ad esempio di fronte al tentativo di sovrascrivere un file di nome "prova.jpg" crea un nuovo file chiamato "prova.jpg.tmp". A questo punto ulteriori tentativi di sovrascrivere il file "prova.jpg" creano una sovrascrittura del file "prova.jpg.tmp". Provando STAMP con un FTP-server di natura Linux questo problema non si è presentato. Per ulteriori informazioni su Stamp e sulle sue opzioni rimandiamo il lettore al capitolo 4.

La soluzione di spedire file via FTP come qui descritto è stata da noi ampiamente provata.

Una terza soluzione è quella di creare un Network File System (NFS) tra “Nofive” e “Brad” (essendo “Frank” un Mac e quindi incapace di fare ciò).

A questo punto sarebbe comunque necessario configurare “Brad” con Apache in modo da renderlo visibile a “Frank” e al resto di Internet tramite il protocollo HTTP.

La pagina HTML in questa terza possibilità potrebbe risiedere su “Brad” e le immagini su “Nofive” in una directory resa comune grazie al NFS.

Per configurare il sistema è necessario, dopo aver installato gli opportuni pacchetti su “Nofive”, creare una directory comune (ad esempio /pub) ed editare il file /etc/exports introducendo la seguente riga:

```
/pub brad(ro)
```

Per lanciare i demoni necessari a questo punto sarà sufficiente digitare:

```
/etc/rc.d/init.d/portmap start
```

```
/etc/rc.d/init.d/nfs start
```

Per quanto riguarda “Brad” sarà sufficiente, dopo aver creato la directory in cui leggere i dati condivisi (ad esempio /mnt/webcam), introdurre nel file “fstab” la seguente riga:

```
NoFive:/pub /mnt/webcam nfs bg,ro,soft,timeo=3 1 1
```

Così facendo la directory condivisa da “Nofive” verrà montata automaticamente all’accensione di “Brad” e sarà sempre disponibile qualora naturalmente “Nofive” sia acceso.

In tal modo l’unico in grado di leggere tale directory sarebbe “Brad” e non si avrebbero problemi di sicurezza. Inoltre “Brad” a differenza di “Nofive” potrebbe in futuro continuare a funzionare come Http-Server essendo i suoi compiti attuali molto limitati.

Facciamo notare che non è possibile comunicare con “Brad” via FTP per ragioni di sicurezza. Infatti l’unico FTP-Server presente all’interno della rete di robotica è “Frank”, che per sua natura, essendo un Mac, è meno attaccabile dall’esterno.

Questa soluzione è stata da noi provata ma non in laboratorio; ad ogni modo non ci sarebbe alcuna difficoltà ad implementarla anche lì.

Una quarta soluzione potrebbe essere quella di utilizzare dei socket per mettere in comunicazione “Nofive” e “Brad”. I socket costituiscono dei nuovi “punti di accesso” per la comunicazione tra processi diversi di una stessa macchina o di macchine diverse.

Ad esempio su “Nofive” ci potrebbe essere un socket che ogni minuto viene attivato in grado di spedire i dati a un altro socket residente su “Brad” in continua attesa di dati a lui destinati.

Questa soluzione è stata solo teoricamente avanzata ma non implementata in quanto le prime tre ci sembravano ugualmente valide e meno complicate di quest’ultima.

2.5. Creazione di script e pagine HTML

Una volta installata la telecamera e il suo software e configurato opportunamente “Nofive” resta il problema di catturare ogni minuto l’immagine ed elaborarla. Questo viene fatto tramite degli script di cui parleremo più apertamente nel capitolo 3.

In generale i loro compiti sono quelli di catturare l’immagine, marciarla con la data e l’ora corrente e mandarla nella opportuna directory dove potrà poi essere caricata dalla

pagina HTML. Lo script si dovrà necessariamente preoccupare di acquisire correttamente l'immagine a livello di luminosità e contrasto. Bisogna inoltre tenere conto che la luminosità del laboratorio varia a seconda delle ore della giornata. Di giorno la stanza è illuminata dalla luce solare mentre di sera da quella artificiale. Queste osservazioni riguardanti la natura del problema dovranno essere prese in considerazione.

Inoltre lo script dovrà provvedere a mettere da parte alcune immagini per tenerne una collezione storica. Ad esempio si potrebbe pensare di tenere un'immagine ogni 5 minuti e di metterne a disposizione una dozzina in modo da avere una visione di ciò che è successo in laboratorio nell'ultima ora.

Dal canto suo la pagina HTML non dovrà poi fare molto quindi non sarà necessaria una pagina HTML dinamica. Unico suo compito, come precedentemente accennato, sarà quello di mostrare l'ultima immagine messaggi a disposizione dallo script. Per fare ciò potrebbe essere dotata di un apposito pulsante oppure ricaricare automaticamente l'immagine ad ogni intervallo di tempo prefissato. Inoltre dovrà avere un link a un'altra pagina in grado di mostrare l'archivio storico delle immagini raccolte dagli script stessi.

3. La soluzione adottata

E' stato relativamente facile, come precedentemente accennato, trovare software relativo a web-camere ormai fuori produzione. Siccome in laboratorio sono a disposizione delle web-camere vecchie ma ben supportate abbiamo deciso di utilizzarle per svolgere quanto richiestoci.

La web-camera a cui stiamo facendo riferimento è la **Connectix Color Quickcam**, web-camera collegabile alle porte parallele del computer.



Fig. 1: Connectix Color Quickcam

Il software che abbiamo scelto di adottare è "Cqcam" per quanto riguarda la cattura delle immagini e "Stamp" per la loro marchiatura. Per quanto concerne questi tools rimandiamo il lettore al capitolo 4 dove potrà trovare passo per passo tutte le istruzioni per installare il software in questione.

Tra le varie soluzioni inerenti alla possibile configurazione di Linux che sono state precedentemente descritte abbiamo adottato la seconda.

Abbiamo deciso di spedire su "Frank" l'ultima immagine da noi salvata. Su "Frank" ci sarà anche la pagina HTML in grado di visualizzare l'immagine in questione e con un collegamento all'archivio storico.

Per quanto riguarda questo archivio storico è stato deciso per uniformità di collocare anch'esso su Frank e in particolare nella directory: */Frank HD/pub/webcam/*. Infatti in tale directory sono presenti anche le immagini catturate da una seconda web-camera della Connectix presente nel laboratorio e collegata a un computer Mac. Così facendo

entrambe le visuali possono essere reperite sia dalla pagina HTML che da un'eventuale Robot che, in base a queste immagini, potrebbe orientarsi nella navigazione del laboratorio stesso. Così anche la pagina HTML destinata alla gestione dell'archivio storico è stata posizionata su Frank.

Per accedere a questa directory però non è sufficiente loggarsi come "anonymous". In tal modo infatti l'utente avrebbe solo diritti di lettura per quanto concerne il contenuto della directory */Frank HD/pub/*. Per potere scrivere nella directory sopra citata abbiamo dovuto richiedere un'opportuna login. In particolare lo user-name è: **webcams** mentre la password è **quickcam**.

Per quanto riguarda la connessione FTP con "Frank" abbiamo deciso di utilizzare le pipe. L'utilizzo di Stamp, come precedentemente spiegato, ci è risultato difficoltoso per i vincoli imposti dal sistema operativo di "Frank" nella riscrittura dei file; ciò nonostante non abbiamo potuto fare a meno di questo prezioso programma per marchiare l'immagine con la data e l'ora correnti.

Per quanto riguarda gli script necessari per la cattura delle immagini abbiamo inizialmente implementato un semplice script di questo tipo:

```
#!/bin/bash
while (echo "")
do
    cqcaml -32- -s 1 -j > thisPicture.jpg
    sleep 2
    stamp
    sleep 58
done
```

Tale script deve essere richiamato ogni volta che la macchina viene riavviata quindi è necessario introdurre in fondo al file *"/etc/rc.d/rc.local"* una riga che ne permetta la ricarica.

Una soluzione di questo genere non ci è però sembrata l'ideale; infatti in tal modo non avremmo potuto gestire in modo attivo i vari parametri con cui catturare l'immagine. In ogni momento della giornata sarebbero stati gli stessi: sia durante il giorno che durante la notte dove le condizioni di luminosità sono ben diverse.

Ci è sembrato opportuno sfruttare il demone crond modificando opportunamente il file */etc/crontab* in modo che nelle ore diurne si potesse catturare l'immagini con determinati parametri mentre in quelle notturne con altri più adatti alle diverse condizioni di luminosità. Per realizzare ciò è stato necessario caricare da crond due differenti script: il primo per le ore diurne prende il nome di "qcaml" mentre il secondo, funzionante durante le ore notturne, si chiama "qcamlnight". Inoltre delegando il compito di sincronizzazione al demone crond siamo sicuri che venga effettivamente catturata un'immagine al minuto.

Sempre compito del demone crond è la gestione dell'archivio; dato che abbiamo deciso di salvare dodici immagini, per un totale di un'ora di storia passata, è necessario che ogni 5 minuti venga fatto partire da crond un altro apposito script per la gestione dell'archivio in questione. Quest'ultimo script prende il nome di "storage".

Di questi script parleremo più approfonditamente nel capitolo 4 visto che essi fanno parte del software necessario per il funzionamento della web-camera.

Per quanto riguarda le situazioni di errore che abbiamo deciso di gestire citiamo la possibilità che la web-camera sia disinserita e quella che "NoFive" sia spento. Il primo problema è affrontato dallo script "Qcaml" e "Qcamlnight" mentre il secondo da "NoFiveoff". I primi due script sono gestiti dal demone crond mentre il terzo dal processo "init". Init legge un particolare file di nome "inittab". Questo file di sistema

viene letto ogni qualvolta si desidera cambiare “initlevel” e quindi anche durante lo spegnimento della macchina.

Per l'impostazione del file “inittab” rimandiamo il lettore al capitolo 4.

4. Modalità operative

In questo capitolo ci proponiamo di spiegare come connettere la web-camera al calcolatore e come installare tutti il software necessario per il suo funzionamento.

4.1. Componenti necessari

4.1.1. Componenti hardware

E' necessario disporre di una qualsiasi tra queste web-camere:

- Connectix Color Quickcam (PC\Parallel)
- Connectix Color Quickcam 2 (PC\Parallel)
- Logitech Color Quickcam (PC\Parallel)
- Logitech Color Quickcam 2 (PC\Parallel)

Nel nostro caso, come precedentemente accennato, abbiamo reperito la Connectix Color Quickcam, già in dotazione presso il laboratorio di Robotica.

Questa web-camera riesce a lavorare ad una risoluzione massima di 640x480, anche se in tale condizione la qualità delle immagini non ci è sembrata eccezionale; buono è invece il suo rendimento qualitativo a risoluzioni inferiori come 320x240. Navigando in Internet abbiamo comunque notato che tutte le pagine HTML che mostrano una immagine di una web-camera utilizzano quest'ultima risoluzione.

Per quanto riguarda la velocità non è certo il massimo in quanto limitata dalle capacità della porta parallela per PC. Nell'acquisizione di un'immagine possono essere necessari anche un paio di secondi; tuttavia dovendo acquisire un'immagine ogni minuto questo punto è risultato per noi del tutto irrilevante.

4.1.2. Componenti software

I componenti software necessari per la realizzazione del progetto sono i seguenti:

4.1.2.1 CQCAM

E' un pacchetto sviluppato da Patrick Reynolds (reynolds@cs.duke.edu) in grado di supportare le Color Quickcam e le Color Quickcam 2 prodotte dalla Logitech e dalla Connectix.

Tutte queste web-camere sono parallele e non più in produzione. Non sono state sviluppate versioni più recenti in grado di supportare le moderne web-camere per USB per la reticenza della Logitech a fornire le loro specifiche.

Cqcam contiene sia il driver della web-camera che il software necessario per catturare l'immagine e salvarla su disco.

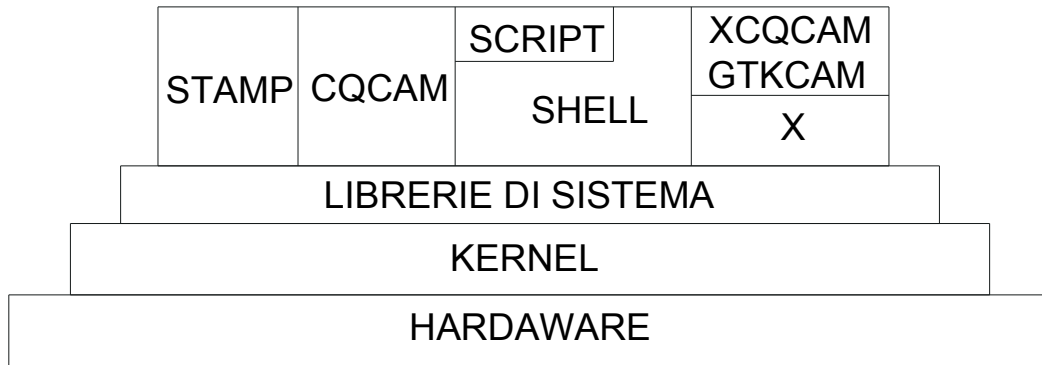


Fig. 2: Architettura di sistema alla luce dei nuovi componenti software installati

Per catturare un'immagine è sufficiente digitare da linea di comando:

```
cqcam > "file_di_destinazione.jpg"
```

Nel suo pacchetto ci sono anche altri interessanti programmi che permettono di vedere in ambiente X il video catturato dalla quickcam in ogni momento e di regolarne i parametri. Essi prendono il nome di "Xqcam" e di "Gtkcam" e si possono trovare nelle omonime sottodirectory. Tra i due segnaliamo Gtkcam che ci è parso decisamente il più evoluto.

Per l'installazione di tale software rimandiamo il lettore al capitolo 4.2 e per la taratura dei parametri con i quali acquisire le immagini al capitolo 4.3.

4.1.2.2 STAMP

Stamp è un programma in grado di applicare messaggi testo, come ad esempio la data e l'ora, sopra un'immagine in formato Jpeg. E' stato realizzato da Patrick Keane (keane@danet.com) e Eric Werner (ebw@city-net.com).

E' stato appositamente concepito per essere usato all'interno di script che gestiscono web-camera sotto Linux. Infatti esso stesso offre la possibilità di trasmettere via FTP l'immagine marchiata al Web-Server in cui è contenuta la pagina HTML da tenere aggiornata.

Per un aiuto sui possibili modi di lanciare il programma è sufficiente digitare:

```
stamp -h.
```

Una schermata di questo tipo apparirà all'utente:

```
Usage:
  stamp <options>
  -f,          --ftp          Use FTP information from stamprc
  -r [file],   --rcfile       Use [file] for stamprc information
  -v,          --version     Show version information
  -h,          --help        This help message
```

Report bugs to <patrick@cs.pitt.edu> or <pk@style64.org>

Quindi una volta opportunamente configurato sarà sufficiente digitare:

```
stamp -f
```

se si desidera spedire via ftp i file marchiati; oppure:

```
stamp
```

se si desidera semplicemente marciare il file.

Per l'installazione di tale software rimandiamoli lettore al capitolo 4.2 e per la taratura dei parametri con i quali acquisire le immagini al capitolo 4.3.

4.1.2.3 SCRIPT & PAGINA HTML

Naturalmente per riuscire a far funzionare il tutto abbiamo dovuto, oltre a produrre la pagina HTML vera e propria che è quanto è visibile all'utente finale, realizzare vari script. In particolare sotto la gestione di crond vengono richiamati gli script "qcamday", "qcamnight" e "storage" mentre sotto la gestione di "init" viene richiamato lo script "nofiveoff".

Qcamday è lo script che viene avviato ogni minuto dal demone crond durante le ore diurne (in particolare dalle 8 alle 17). Questo script ha il compito di acquisire l'immagine dalla web-camera mediante "Cqcam", apporci data e titolo tramite "Stamp" e spedirla a "Frank" via FTP tramite l'uso opportuno delle pipe (comqcam.txt). Nel caso la web-camera fosse assente provvederà a fornire a "Frank" un'apposita immagine che comunicherà all'utente questo problema. In tal modo è stata risolta la prima delle situazioni di errore precedentemente accennate.

Qcamnight è lo script che viene avviato ogni minuto dal demone crond durante le ore notturne (in particolare dalle 17 alle 8). Questo script è del tutto analogo al precedente in quanto a compiti. L'unica differenza sta nei parametri con cui acquisisce l'immagine: dovendo funzionare durante la notte l'ambiente risulterà diversamente illuminato e quindi catturerà immagini con un parametro di luminosità e di contrasto opportuni. Sia Qcamday che Qcamnight sono avviati ogni minuto dal demone crond.

Storage viene avviato ogni 5 minuti dal demone crond durante tutte le ore del giorno. Ha il compito di gestire l'archivio storico delle immagini. Ci siamo prefissati di gestire 12 immagini in modo da avere una visione di quello che è successo nel laboratorio nell'ultima ora. In particolare Storage provvede a fare slittare i file indirizzati dalla pagina HTML eliminando l'ultimo file e rinominando gli altri in modo da avere una corrispondenza corretta tra ciò che si trova indicato nella pagina HTML e i nomi dei file. Essendo questi file residenti su "Frank" queste operazioni appena descritte vengono eseguite via FTP per mezzo delle pipe (comstorage.txt).

"Nofiveoff" è uno script gestito da "init" in base alle modifiche apportate al file "inittab". In particolare questo script viene eseguito in situazione di spegnimento o di reset del calcolatore. Suo compito è, prima che Linux inizi le consuete operazioni di spegnimento, spedire a "Frank" un'opportuna immagine contenente un messaggio indicante che il calcolatore in questione è spento. Essendo spento l'immagine non verrà più aggiornata fino al riavvio di "Nocive" dove gli script precedentemente menzionati provvederanno a sovrascrivere l'immagine spedita precedentemente con la consueta immagine del laboratorio e a porre fine così alla situazione anomala.

Qui di seguito riportiamo un diagramma a blocchi delle operazioni fatte da qcamday (o qcamnight) e storage.

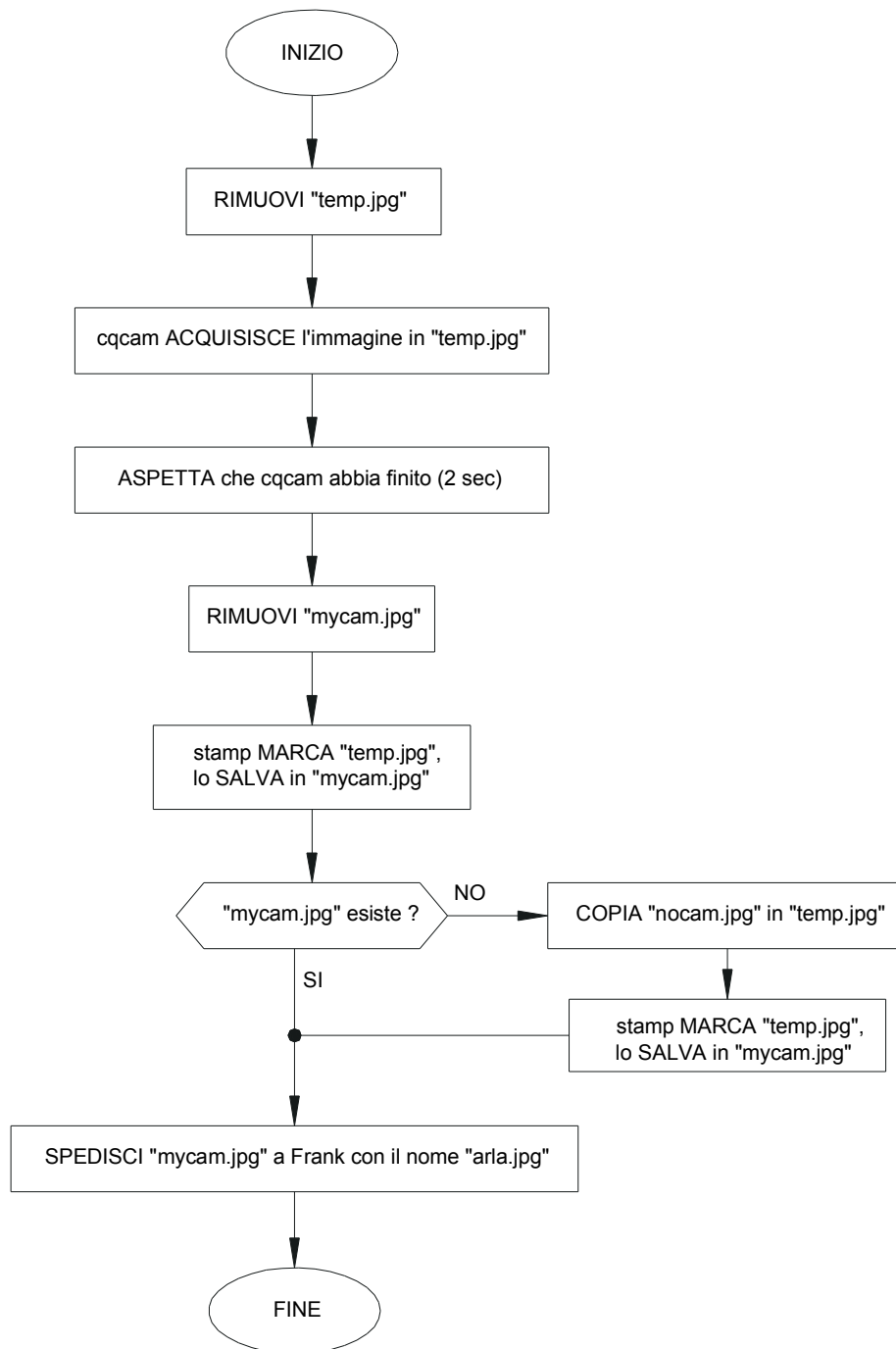


Fig. 3: Diagramma a blocchi di “Qcamday” e “Qcamnight”

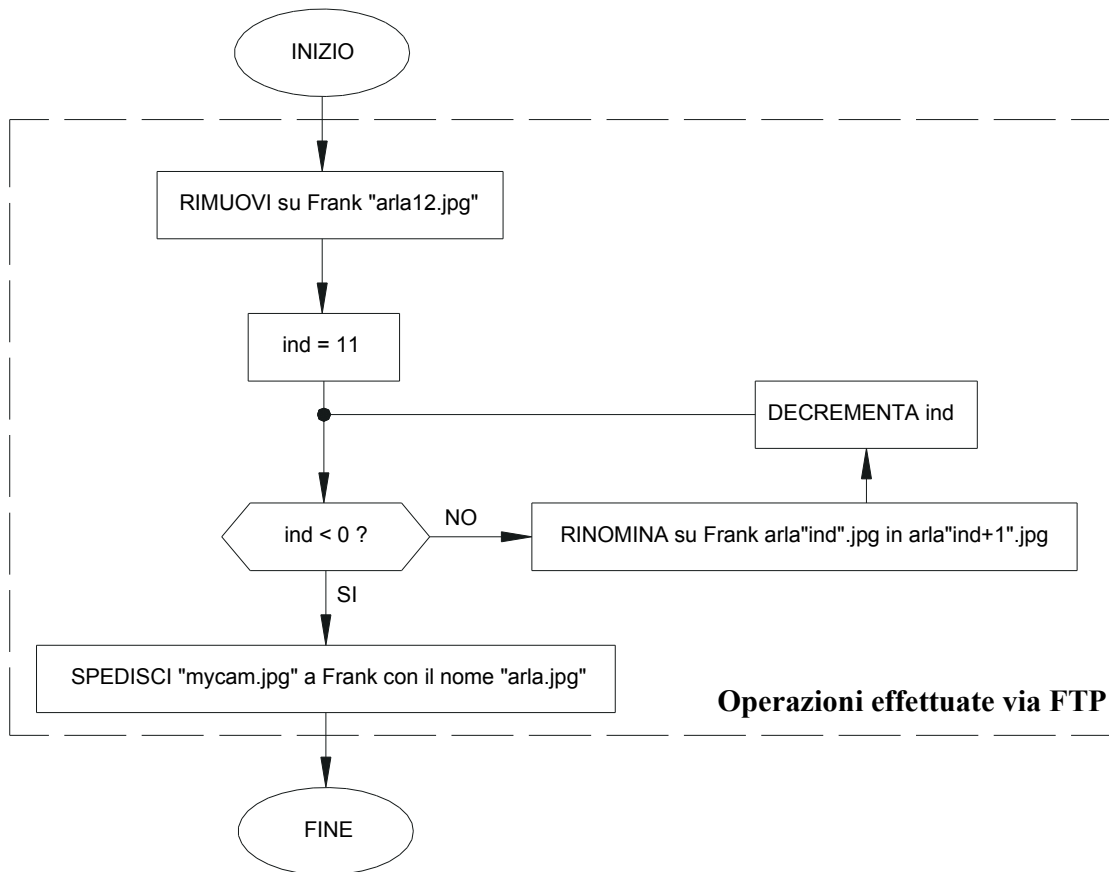


Fig. 4: Diagramma a blocchi di "Storage"

Per quanto concerne le pagine HTML ne abbiamo prodotte due versioni. La prima si aggiorna automaticamente ogni 60 secondi ricaricando in tal modo l'ultima immagine salvata dagli script. La seconda invece presenta un opportuno bottone che, quando cliccato, aggiorna la pagina.

La prima soluzione potrebbe sembrare la più comoda, ma è anche la meno standard nel senso che alcuni browser potrebbero non ricaricare da soli l'immagine ogni minuto. La seconda è sicuramente più robusta e di conseguenza è stata poi adottata.

Per l'installazione di tutto ciò rimandiamoli lettore al capitolo 4.2.

4.2. Modalità di installazione

In questo capitolo ci proponiamo di spiegare dettagliatamente come installare l'hardware, cioè la web-camera a nostra disposizione, e il software necessario per il suo funzionamento.

4.2.1. Installazione dell'hardware

Innanzitutto è necessario collegare le web-camera. Questo punto potrebbe parere scontato ma, a causa dell'arretratezza della web-camera, l'utente potrebbe trovarsi in difficoltà, così come è successo a noi la prima volta che ne siamo entrati in contatto.

La Connectix Color Quickcam è dotata di ben 3 connettori: uno di essi è parallelo mentre gli altri due sembrano inerenti alla tastiera; ed effettivamente lo sono!

Per connettere la web-camera agganciare il connettore parallelo della camera alla porta parallela del computer; estrarre il connettore della tastiera e collegarlo alla presa femmina del cavo della web-camera; infine collegare il connettore maschio del cavo della web-camera alla presa della tastiera del PC. Tutto ciò è necessario in quanto i +5V necessari per l'alimentazione della web-camera sono prelevati dal connettore della tastiera.

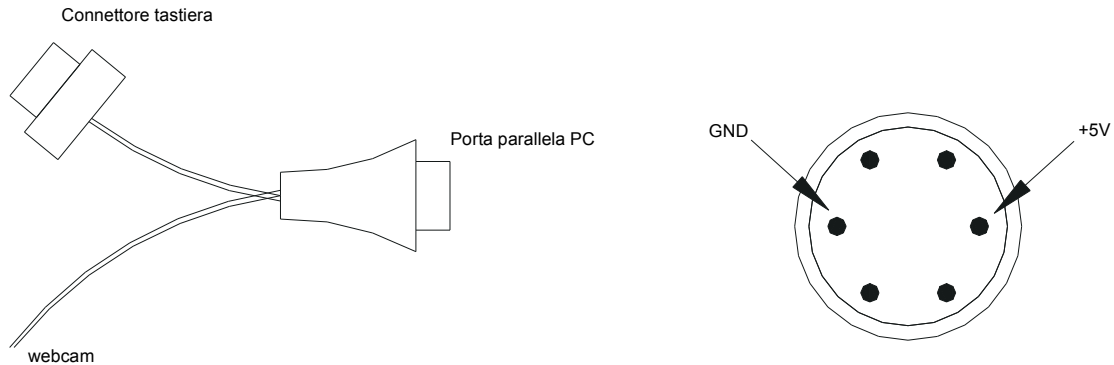


Fig. 5: A sinistra i connettori della web-camera. A destra i piedini del connettore di una tastiera.

4.2.2. Installazione del software

Per installare il software richiesto è necessario innanzitutto loggarsi come Root sul computer sul quale si desidera effettuare l'installazione (nel nostro caso "Nofive"). Infatti durante l'installazione sono necessari i privilegi di Root per potere copiare file in directory dove un normale utente non avrebbe accesso.

In secondo luogo inserire il cd-rom allegato alla relazione nel lettore cd-rom e montare quest'ultimo (ciò potrebbe non essere necessario in caso nel file fstab fosse già presente l'auto-mount del cd-rom). Ad ogni modo per montare il cd-rom è necessario digitare:

```
mount /dev/cdrom /mnt/cdrom -t iso9660 -r
```

Portarsi poi in una directory opportuna (suggeriamo */usr/local*) e digitare:

```
tar xvfz /mnt/cdrom/quickcam.tar.gz
```

A questo punto dovrebbero essersi create tre directory:

```
../cqcam-0.91
```

```
../stamp-2-0.8
```

```
../install
```

La prima di esse è relativa a Cqcam, la seconda a Stamp e la terza alle pagine HTML e agli script che dovranno essere installati. Quest'ultima directory potrà essere rimossa dopo l'installazione.

Ci proponiamo a questo punto di parlare apertamente di come installare tutti tre questi elementi.

4.2.2.1 CQCAM (./cqcaml-0.91)

Per l'installazione portarsi nella directory creata dalla scompattazione:

```
cd cqcaml-0.91
```

e digitare nell'ordine i seguenti comandi:

```
./configure
```

per avviare il "configure" script;

```
make
```

per compilare Cqcaml;

```
make install
```

per installare Cqcaml.

Alla fine dell'installazione i seguenti file:

./cqcaml-0.91/cli/cqcaml: Programma per grappare le immagini da linea di comando

./cqcaml-0.91/xfe/xcqcaml: Interfaccia grafica di Cqcaml

./cqcaml-0.9/gtkfe/gtkcaml: Nuova interfaccia grafica per GTK+

./cqcaml-0.91/webcam/webcam: Utility per lo streaming

si dovrebbero trovare nella directory */usr/local/bin*.

In teoria l'installazione dovrebbe copiare nella propria home-directory (in questo caso quella di Root) un file di configurazione di nome *.cqcrc*; ma qualora non dovesse essere fatto l'utente prima di usare il programma dovrebbe provvedere egli stesso a farlo. Se ciò non dovesse essere fatto Cqcaml non funzionerebbe!!!

Per una corretta installazione è necessario che sul calcolatore ci siano già installate le librerie per la gestione delle jpeg versione 6.0 o superiore (*libjpeg*). Per rintracciare queste ultime si può fare riferimento o alla propria distribuzione o all'indirizzo: <ftp://ftp.uu.net/graphics/jpeg>.

Opzionalmente se si desidera compilare *xcqcaml* sono necessarie *libX11* e *libXext*, mentre per *gtkcam* deve essere installato *GTK+*.

4.2.2.2 STAMP (./stamp-2.0.8)

Per l'installazione portarsi nella directory creata dalla scompattazione:

```
cd stamp-2.0.8
```

e digitare nell'ordine i seguenti comandi:

```
./configure
```

per avviare il "configure" script;

```
make
```

per compilare Stamp;

```
make install
```

per installare Stamp.

A questo punto se l'installazione non lo ha già fatto è necessario copiare nella propria home-directory il file di configurazione di Stamp che prende il nome di *.stamprc*.

```
cp ../stamp-2.0.8/doc/stamprc ~/.stamprc
```

In mancanza di tale file Stamp non funzionerebbe comunicando un errore del tipo: "Segmentation fault".

Per potere installare correttamente Stamp sono necessarie le librerie per la manipolazione delle Jpeg e opzionalmente è richiesto GTK. Quest'ultimo non è essenziale, ma avendolo è possibile utilizzare Gstamp per la configurazione di Stamp. Gstamp è un interfaccia grafica che facilita la modifica del file ".stamprc". Per impostare correttamente la configurazione di tale file rimandiamo il lettore al paragrafo 4.3.

4.2.2.3 SCRIPT di INSTALL

"Install" è uno script realizzato da noi che esegue le operazioni necessarie per configurare "Nofive" in modo da potere catturare e gestire le immagini della web-camera. Innanzitutto esso verifica l'esistenza dei programmi necessari: Cqcam e Stamp. A questo punto questo script copia gli script "qcamday", "qcamnight" e "storage" e "Nofiveoff" nelle opportune directory. Compito di questo script è inoltre quello di modificare opportunamente crontab e fare ripartire il demone crond in modo da avviare le operazioni di cattura e gestione delle immagini da parte degli altri script.

Per far eseguire a Install i compiti qui spiegati portarsi nella directory creata dalla scompattazione:

```
cd install
```

e digitare nell'ordine i seguenti comandi:

```
./install
```

Attenzione prima di installare gli script con "install" è opportuno installare Cqcam e Stamp e configurare quest'ultimo a mano o mediante Gstamp. Infatti Install rilancia il demone crond e quindi tutto deve essere già appunto per l'esecuzione dei vari script.

Se si desiderasse anche copiare le pagine HTML su "Frank" sarebbe possibile digitare:

```
ftp frank <install_frank.txt
```

Install_frank.txt è appunto un file contenete tutti i comandi per spedire i file necessari.

A questo punto se tutto è andato per il meglio la directory "install" può anche venire cancellata. Sia la pagina HTML che gli script contenuti in tale directory sono infatti già stati depositate nelle directory corrette dalle quali verranno utilizzati al momento opportuno.

Per quanto riguarda il demone crond dopo l'installazione il file */etc/crontab* dovrebbe avere un aspetto del tipo:

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/bin
MAILTO=root
HOME=/root/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
```

```

42 4 1 * * root run-parts /etc/cron.monthly
# ----- Script per la gestione della webcam -----
* 8-17 * * * root /etc/cron.d/qcamday &>/dev/null
* 18-23,0-7 * * * root /etc/cron.d/qcamnight &>/dev/null
*/5 * * * * root /etc/cron.d/storage &>/dev/null
# -----

```

dove l'ultima parte riguarda appunto la gestione della web-camera.

Facciamo notare che se non si presentasse in tale forma risulterebbe molto importante settare opportunamente la directory "home". Infatti Ftp per il suo funzionamento, come precedentemente accennato, fa riferimento a un file che prende il nome di .netrc e si trova in /root. Quando viene avviato Ftp fa riferimento alla directory di home dell'utente in questione e dato che il proprietario di crond è appunto root se non si settasse in tal modo la home si avrebbe un errore. In alternativa a ciò si potrebbe anche copiare il file .netrc in "/.

Un problema analogo ha Stamp che fa riferimento al file .stamprc nella directory di Root, ma a differenza di FTP presenta tra le sue opzioni (-r) la possibilità di esprimere il percorso al quale trovare detto file.

Dopo aver modificato il file /etc/crontab in modo da farlo risultare come sopra riportato è necessario fare ripartire il demone crond. Per fare ciò digitare:

```
/etc/rc.d/init.d/crond restart
```

Inoltre se si volesse segnalare all'utente Internet il fatto che Nofive è spento sarebbe necessario modificare il file "Inittab" alla voce "System Initialization" aggiungendo le due righe in grassetto qui sotto riportate:

```

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

18:0:wait:/etc/cron.d/nofiveoff &>/dev/null
10:0:wait:/etc/rc.d/rc 0
11:1:wait:/etc/rc.d/rc 1
12:2:wait:/etc/rc.d/rc 2
13:3:wait:/etc/rc.d/rc 3
14:4:wait:/etc/rc.d/rc 4
15:5:wait:/etc/rc.d/rc 5
17:6:wait:/etc/cron.d/nofiveoff &>/dev/null
16:6:wait:/etc/rc.d/rc 6

```

La riga corrispondente all'initlevel 0 indica lo script da eseguire quando si ferma la macchina mentre quella corrispondente a initlevel 6 si riferisce a quando la macchina viene resettata.

Per spedire a Frank l'immagine viene fatto uso di FTP e anche se il proprietario del processo init che si occupa appunto di leggere tale file è Root la Home-directory a cui viene fatto riferimento è "/. Quindi per fare reperire a FTP il file .netrc è necessario copiare detto file in "/. In alternativa è possibile usare le opzioni FTP di Stamp cioè:

```
stamp -r /root/.stamprc
```

4.3. Modalità di taratura

In questo capitolo ci proponiamo di spiegare dettagliatamente come settare il software precedentemente installato.

4.3.1. CQCAM

Per il settaggio di parametri di Cqcam digitare:

```
cqcam -h
```

apparirà una schermata del tipo:

```
Usage: cqcam-0.91/cli/cqcam [options]

-32[+|-]          Turn 32-bpp mode on or off (off = 24bpp)
-a[+|-]          Use/suppress brightness and color balance auto-
                 adjustments
-b val           Set brightness
-B val           Set black level
-c val           Set contrast
-d val           Specify (or skip) camera-detection
-E val           Set software blue level
-G val           Set software green level
-h              View this brief help screen
-H val           Set hue (blue level)
-j              Write output in JPEG format
-l val           Set left column
-m delay,name[,count] Movie mode:capture count frames, one every val
                 seconds to name-X.ppm
-P val           Set port to attempt (val must be in hex format)
-q val           Set JPEG quality, for use with -j (default=50)
-r             Remember (store) the brightness in .cqcr
-R val           Set software red level
-s val           Set scale factor (decimation)
-S val           Set saturation
-t val           Set top row
-u             Force a unidirectional port mode
-V             Print version information and exit
-w val           Set white level
-x val           Set width
-y val           Set height
```

Le opzioni con cui abbiamo usato Cqcam all'interno dei nostri script sono:

```
cqcam -32- -j -s 1 >file_destinazione
```

Brevemente ciò significa che Cqcam funziona in modalità 24 bit per pixel (acquisendo le immagini in formato 320x240 ma con una maggiore qualità dell'immagine), salva l'immagine in formato jpeg e tiene il fattore di scala pari a 1.

Un'altra opzione interessante è `-q` che permette di regolare la qualità dell'immagine. Se ad esempio si porta tale valore a 100 si ottiene un'immagine di ottimo livello, ma che occupa decisamente troppi byte. Per default la qualità della Jpeg vale 50 e in tale condizione la grandezza del file ammonta a circa 8 o 9 Kbyte e la qualità dell'immagine è tutto sommato discreta. A fronte di un piccolo aumento della qualità la grandezza del file cresce di un fattore 10 arrivando ad occupare 65 Kbyte; quindi abbiamo ritenuto che il valore di default corrispondesse alla giusta via di mezzo tra qualità ed efficienza e non abbiamo ritoccato tale parametro.

Tuttavia durante le ore notturne abbiamo preferito catturare le immagini con delle diverse tarature:

```
cqcam -32- -j -b 235 -B 128 -S 128 -w 160 -H 15 -s 1 >file_destinazione
```

in aggiunta ai parametri usati per le ore diurne vi è una regolazione sulla luminosità, sul livello di nero, di bianco, di blu e di saturazione dell'immagine.

In alternativa a ciò si potrebbe richiamare Cqcam con le stesse opzioni sia di giorno che di notte affidandosi all'auto-adjustment (-a+); ma a fronte di alcune prove abbiamo notato che la qualità delle immagini risultava migliore con le opzioni così tarate.

4.3.2. STAMP

Per configurare Stamp è possibile editare direttamente il file “.stamprc” nella propria home-directory. Esso dovrebbe assumere un aspetto di questo tipo:

```
upperstring date +%a' '%B' '%d' '%r
lowerstring This is LRA of Brescia University
upperfont /usr/local/share/stamp/fonts/8x8.fnt
lowerfont /usr/local/share/stamp/fonts/stretch.fnt
infile /usr/local/nofive/httpd/temp.jpg
outfile /usr/local/nofive/httpd/mycam.jpg
ftphost frank.ing.unibs.it
ftppath ./webcam/.
ftpupload mycam.jpg
ftplogin webcams
ftppasswd quickcam
ustringexec 1
lstringexec 0
shaderate 8
quality 50
rotate 0
use3d 1
usecolors 0
redfore 255
greenfore 255
bluefore 255
redback 62
greenback 25
blueback 205
previewfile /usr/local/nofive/httpd/webcam.jpg
```

Dato che Stamp si prefigge di stampare due righe di testo (uno in alto e una in basso alla figura) sono necessarie le fonti di scrittura (upperfont e lowerfont) e il testo delle corrispondenti stringhe (upperstring e lowerstring).

Il programma prendendo l'immagine da marchiare (infile) dà come output, se necessario, due file: uno sul computer locale (outfile) e uno su un calcolatore remoto (ftpupload e ftppath). Per spedire il file a quest'ultimo è necessario che si conosca l'host, la login e la password (rispettivamente ftphost, ftplogin e ftppasswd). La qualità dell'immagine Jpeg di uscita può essere settata direttamente in questo file (quality) ed è buona norma che non ecceda la qualità dell'immagine in ingresso (vedi valore di default della qualità impostata a sua volta da Cqcam). Tutti gli altri settaggi indicano i colori del testo e dello sfondo sotto il testo stesso.

In alternativa a un settaggio manuale è possibile utilizzare Gstamp: un programma dotato di interfaccia grafica per la configurazione di “.stamprc”. E' sufficiente digitare:

gstamp

in un terminale in ambiente X affinché appaia una finestra di dialogo del tipo:

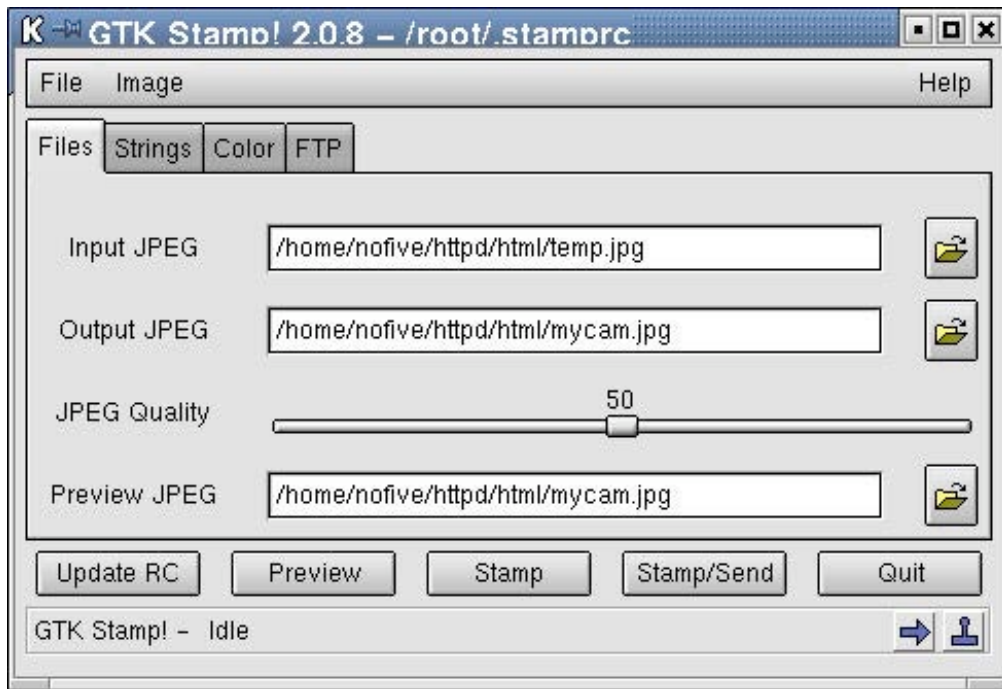


Fig. 6: Finestra di dialogo di GSTAMP

avente le seguenti voci:

- files: contenente i file di input, output, preview e la qualità dell'immagine
- strings: contenente le stringhe sopra e sotto l'immagine con le relative fonti
- color: contenente la possibilità di settare i colori del testo e il suo sfondo
- ftp: contenente le opzioni necessarie per spedire l'immagine via FTP; cioè host, login, password, path e nome dell'immagine

Il tutto è molto più intuitivo rispetto a quanto potrebbe essere compilare direttamente il file .stamprc con un editor di testo.

Notiamo che se si volessero configurare diversamente i percorsi riguardanti il file di input (infile) e il file di output (outfile) sarebbe necessario modificare gli script che assumono appunto di trovare in tale directory l'immagine catturata. In particolare andrebbero modificati i file Qcamday, Qcamnight e Nofiveoff alla voce FIGDIR e i file comqcam.txt e comstorage.txt contenenti le operazioni da effettuare via FTP. Tutti questi file per una loro eventuale modifica possono trovarsi nella directory /etc/cron.d.

4.4. Avvertenze

Per quanto riguarda la Connectix Quickcam è possibile, all'accensione del calcolatore, che a volte la telecamera venga rilevata e a volte no. Questo è dovuto all'autoprobe di Linux; quest'ultimo è fatto per facilitare le cose all'utente. Il suo compito è quello di sondare gli indirizzi delle periferiche, nel nostro caso della porta parallela, per trovare l'hardware connesso, per quanto ci riguarda la web-camera. Una volta che l'ha trovata inserisce l'indirizzo nel file di configurazione di Cqcam; in tal modo il driver salterà in futuro l'autoprobe e userà direttamente tale indirizzo. Questo problema è dovuto al modo di funzionamento della web-camera. Dal momento in cui la telecamera è accesa continua a passare tra gli stati picture-mode e command-mode. In picture-mode la quickcam misura la luce che colpisce l'obiettivo; copia l'immagine nella sua memoria e

passa in command-mode dove attende comandi dal PC (ad esempio la copia dell'immagine sul PC). Il command-mode termina quando la quickcam si accorge che l'ultima immagine salvata comincia ad affievolirsi; a tal punto torna in picture-mode. Quando è in picture-mode la quickcam periodicamente cambia i suoi bit di stato leggibili dal PC. Questo permette al PC di sapere se la quickcam è ancora attiva; attraverso questi bit di stato l'autoprobe di Linux stabilisce se una web-camera è attaccata o meno alla porta parallela. Il problema nel rilevamento della telecamera è dovuto al fatto che la durata del picture-mode è variabile in funzione della luminosità settata; se la web-camera resta troppo poco in picture-mode l'autoprobe non riesce a vedere nessun cambiamento di bit e quindi non rileva la camera.

Facciamo presente che la Connectix Color quickcam è una web-camera ideata per gli interni; se si volesse utilizzare all'aperto per ottenere dei buoni risultati sarebbero necessari dei filtri.

Avendo a disposizione più porte parallele c'è la possibilità di collegare più quickcam. A questo punto le web-camera si troveranno su porte parallele diverse e per fare riferimento all'una o all'altra bisognerà intervenire direttamente sul file di configurazione a cui Cqcam fa riferimento. Esso contiene i valori di default di Cqcam tra cui da quale porta parallela dovere ogni qual volta leggere i dati.

Per quanto riguarda Cqcam un comune problema che si può presentare è che la qualità dell'immagine risulti piuttosto scadente come se l'immagine fosse sbiadita. Ragione di ciò potrebbe essere un eccessivo carico di lavoro del calcolatore in questione o un processore di potenza troppo bassa. Per ulteriori problemi riguardanti Cqcam segnaliamo l'utilizzo di una mailing-list all'indirizzo www.crynwr.com/qcpc e un archivio di quesiti all'indirizzo www.crynwr/qcpc/archive.

Qualora l'immagine elaborata da Stamp fosse alquanto strana il motivo potrebbe essere il fatto che la qualità dell'immagine in ingresso è già di per sé troppo bassa. Per ovviare a questo inconveniente è possibile catturare l'immagine con Cqcam con un più alto livello di ottimizzazione (vedi opzione -q di Cqcam).

5. Conclusioni e sviluppi futuri

Attualmente sul mercato non sono più presenti telecamere per porte parallele; quindi quanto precedentemente spiegato riguardante il software da installare non sarebbe più valido con una web-camera dell'ultima generazione.

Se in futuro si volesse comprare una telecamera USB la scelta dovrebbe essere molto oculata in quanto non tutte le telecamere hanno sviluppati dei driver che le renderebbero capaci di funzionare in ambiente Linux. Naturalmente questo discorso non vale per ambienti Windows visto il monopolio che di fatto ha la Microsoft nel settore dei S.O.. Ad ogni modo volendo utilizzare un S.O. Linux noi abbiamo faticato molto per rintracciare su Internet i driver e i tool per riuscire a realizzare quanto richiesto (vedi capitolo 2).

Dopo una attenta ricerca abbiamo deciso di concentrare la nostra attenzione sulle seguenti web-camere:

- Logitech Quickcam Express (per porta USB)
- Philips ToUCam Pro (USB)

La prima in particolare perché è una telecamera molto diffusa e su di essa siamo riusciti a trovare più di un driver e una buona documentazione mentre la seconda perché già in dotazione all'Ingegnere Meriggi.

Naturalmente non sono le sole telecamere supportate che siamo riusciti a trovare; tra quelle ancora in produzione possiamo citare le telecamere della Kensington, la Intel PC Camera Pro e la 3COM's Home Connect Webcam... ma non ci sono sembrate inserite in progetti abbastanza seri da essere presi in considerazione (vedi capitolo 2 per reperire gli indirizzi internet).

Nei prossimi paragrafi ci proponiamo di parlare più apertamente di queste 2 famiglie di telecamere: delle loro caratteristiche, dei problemi che si possono trovare nella loro installazione nonché della installazione del software stesso per riuscire a metterle in funzione.

5.1. Logitech Quickcam Express (USB)

Questa web-camera della Logitech è una delle più diffuse al mondo: ha un prezzo molto contenuto, raggiunge una risoluzione massima di 352x288 e una minima di 176x144, si basa sul formato di palette RGB24 e di conseguenza ha milioni di colori.

Essendo una web-camera USB la sua velocità non è limitata come per le sua antenata a porta parallela e riesce a raggiungere i 30 frame per secondo.



Fig. 7: Logitech Quickcam Express

Siamo riusciti a trovare dei driver sviluppati da persone diverse in grado di supportare questa web-camera.

Grazie alla sua vasta diffusione siamo riusciti a procurarcene un modello e a condurre qualche esperimento in casa. Abbiamo testato i driver e così possiamo esporre dei risultati.

5.1.1. LOGITECH QUICKCAM EXPRESS (USB)

Questo tra i due driver da noi testati è quello che ha dato i risultati migliori a livello di qualità dell'immagine. Per correttezza citiamo i suoi sviluppatori: Jean-Frederic Clere (jfclere@sinix.net) e Nikolas Zimmermann (wildfox@kde.org).

Il driver si basa su un modulo che dopo essere compilato può essere opportunamente inserito nel kernel in esecuzione.

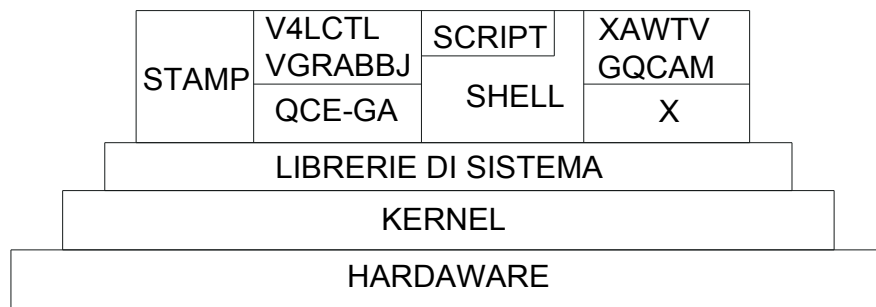


Fig. 8: Architettura di sistema alla luce dei componenti software necessari per la Logitech Quickcam Express

Per l'installazione seguire i seguenti passi:

scompattare il pacchetto digitando:

```
tar xvfz qce-ga-0.40b.tar.gz
```

entrare nella directory creata dalla scompattazione:

```
cd qce-ga.0.40
```

e digitare :

```
make
```

per compilare il driver e:

```
./quickcam.sh
```

che eseguirà i seguenti passi:

```
insmod videodev
insmod usbcore
insmod usb-uhci
insmod usb-ohci
insmod uhci
```

Se questi moduli non sono rintracciabili o caricabili (controllare /var/log/messages) probabilmente è necessario ricompilare il kernel. Per la ricompilazione del kernel rimandiamo il lettore al capitolo 5.4.

Se al contrario si ha una distribuzione di Linux abbastanza recente questi moduli dovrebbero già essere a disposizione e si potrà procedere al caricamento del modulo riguardante la specifica web-camera digitando:

```
insmod mod_quickcam.o
```

Insmod è un apposito comando in grado di installare un modulo caricabile nel kernel.

A questo punto suggeriamo di digitare:

```
modprobe -a videodev
modprobe -a mod_quickcam.o
```

Il comando modprobe è in grado di gestire i moduli caricabili nel kernel ad alto livello. Questa utility, come insmod, è fatta per rendere il kernel di Linux maneggiabile da tutti gli utenti. Con l'opzione “-a” si caricano automaticamente tutti i moduli necessari per il modulo in questione.

I parametri accettabili da mod_quickcam durante il suo caricamento sono elencabili con il comando:

```
modinfo -p mod_quickcam
```

in tal caso apparirà una schermata del tipo:

```
debug int, description "Sets the debug output (1,2,4,8,16,32)"
mode int, description "Sets the speed (0-1)"
```

Se invece si digita *lsmod* dovrebbe apparire una schermata del tipo:

```
Module                Size  Used by
mod_quickcam          26896  0  (unused)
videodev              4512   1  [mod_quickcam]
usb-uhci              20672  0  (unused)
usbcore               47248  1  [mod_quickcam usb-uhci]
```

che illustra all'utente quali sono i moduli attualmente caricati nel kernel.

A titolo informativo facciamo presente che per rimuovere un modulo inserito è presente il comando *rmmmod*.

Notiamo che il modulo videodev è utilizzato dal nostro driver della quickcam. In tal modo quando si fa riferimento al dispositivo /dev/video (che non è altro che un link al dispositivo /dev/video0) in realtà ci si riferisce alla web-camera.

Così facendo l'immagine acquisita dalla web-camera diventa manipolabile da un qualsiasi gestore basato su Video4Linux. Video4Linux si basa sui seguenti dispositivi:

Device Name	Minor Range	Function
/dev/video	0-63	Video Capture Interface
/dev/radio	64-127	AM/FM Radio Devices
/dev/vtx	192-223	Teletext Interface Chips
/dev/vbi	224-239	Raw VBI Data (Intercast/teletext)

Un programma basato su Video4Linux apre ed esamina questi dispositivi.

Prima di utilizzare la telecamera sarebbe opportuno assicurarsi che esistano appunto questi file di device. Per una loro eventuale installazione rimandiamo il lettore al capitolo 5.5.

Quando un grabber basato su Video4Linux legge da /dev/video i moduli necessari (videodev e mod_quickcam) dovrebbero già essere caricati in memoria grazie alla procedura precedentemente seguita; se così non fosse il kernel affiderà la situazione al demone kerneld affinché esso possa risolverla. In particolare il kernel notificherà che è richiesto un compito da lui non affrontabile e manderà un messaggio a kerneld con una descrizione simbolica di ciò che è stato richiesto. Tale descrizione conterrà l'ID del dispositivo a cui si vuole fare riferimento (in caso di /dev/video0 gli ID in questione sono pari a 81,0 che rispettivamente indicano il "major" e "minor device number"). Il demone chiederà a questo punto a modprobe di caricare il modulo in grado di svolgere il compito richiesto al kernel. Modprobe dal canto suo controllerà la sua tabella di traduzione interna per controllare se esiste un modulo caricabile in grado di risolvere il problema. Questa tabella può essere riconfigurata per rendere i moduli di gestione della quickcam caricabili Per fare ciò è necessario editare il file /etc/modules.conf aggiungendo le seguenti righe:

```
alias char-major-81 videodev
alias char-major-81-0 mod_quickcam
```

Così si genererà una corrispondenza tra il dispositivo `/dev/video` e i moduli necessari al suo funzionamento.

A questo punto insmod inserirà il modulo che modprobe ha deciso di caricare e kerneld comunicherà al kernel l'esito delle operazioni fatte. Se il modulo caricato non venisse utilizzato per più di un minuto kerneld provvederebbe mediante `rmmod` a rimuoverlo.

La modifica del file `/etc/modules.conf` con le conseguenze qui spiegate può essere effettuata in alternativa alla procedura di inserimento dei moduli vista in precedenza.

Per quanto riguarda i grabber che si possono utilizzare se si digita:

`xawtv`

in ambiente X è possibile vedere un filmato in real-time di ciò che acquisisce la telecamera.

Xawtv è un semplice programma che si finalizza di mostrare la TV. Esso fa riferimento al dispositivo `/dev/video` e quindi nel nostro caso anziché mostrare i vari canali della TV mostra semplicemente l'output della web-camera.

Una volta caricato questo programma è possibile da tastiera catturare un'immagine: digitando CTRL-G viene creata un'immagine in formato ppm mentre digitando CTRL-J ne viene creata una in formato jpeg. Questi file avranno un nome del tipo `snap-???*.jpeg`.

Se si vuole grappare un'immagine da linea di comando (come servirebbe a noi dato che l'immagine dovrebbe essere catturata da uno script in modo che l'operazione risulti trasparente) è possibile ricorrere a un altro comando:

`v4lctl snap jpeg full <nomefile.jpeg> setnorm PAL capture grabdisplay fullscreen`

Sia `xawtv-remote` che `v4lctl` sono programmi per controllare da linea di comando dispositivi basati su Video4linux.

Digitando `v4lctl -h` si ottiene la seguente schermata:

```
usage: v4lctl [ options ] command
options:
  -v, --debug=n      debug level n, n = [0..2]
  -c --device=file   use <file> as video4linux device
  -h --help          print this text
```

Sia `v4lctl` che `xawtv-remote` accettano gli stessi comandi; tra essi citiamo quelli che abbiamo utilizzato per grappare l'immagine:

`snap [jpeg | ppm] [full | win | widthxheight] <file-name>` per catturare l'immagine

`fullscreen` per passare in full-screen mode

`capture [on | off | overlay | grabdisplay]` per settare il modo di cattura dell'immagine

`setnorm <norm>` per settare lo standard TV (NTSC/PAL/SECAM)

Entrambi questi programmi sono presenti nelle ultime più comuni distribuzioni di Linux. Se ad ogni modo si volessero utilizzare grabber appositi segnaliamo `v4grabbj` di cui parleremo in seguito.

Con questo driver si possono utilizzare un massimo di 2 web-camere efficientemente: una per ogni porta USB; questo a causa del grosso carico di lavoro che comporta la gestione della web-camera che occupa circa il 90% della larghezza di banda della connessione.

Per avere una idea sulle qualità delle immagini che si può ottenere con questa web-camera rimandiamo il lettore al capitolo 8.

5.1.2. LINUX DRIVER PER LOGITECH QUICKCAM EXPRESS WITH HDCS1000/PB0100 IMAGE SENSOR

Questo driver è stato sviluppato da Georg Acher (acher@in.tum.de) che ha condotto il reverse-engineer anche del driver precedentemente citato.

Anche questo driver una volta compilato crea un modulo caricabile nel kernel.

Per installare il driver seguire i seguenti passi:

```
make
```

per compilare il driver; dopo essersi loggiati come Root digitare:

```
mknod /dev/quick c 180 80
```

per creare un file speciale;

```
chmod 666 /dev/quick
```

per cambiare i privilegi di tale file; e

```
insmod quickcamex.o
```

per inserire il modulo nel kernel.

A questo punto come normale utente si può vedere sotto X quello che la telecamera sta mostrando digitando:

```
./quickcam
```

Questo driver non utilizza Video4Linux così abbiamo trovato difficoltà a servircene per i nostri scopi cioè a grappare un'immagine da linea di comando.

5.2. Philips Vesta ToUCam Pro (USB)

La Philips Vesta ToUCam Pro è una web-camera prodotta dalla Philips da connettere alla porta USB del PC; raggiunge la risoluzione massima di 640x480 e a tali risoluzioni arriva a trasferire fino a 30 frame per secondo, è basata su un sensore CCD e su un formato di palette RGB24.



Fig. 9: Philips Vesta ToUCam Pro

Per detta telecamera abbiamo trovato solo un driver!

5.2.1. Supporto Linux per Philips USB Web-Cameras

All'indirizzo <http://www.smcc.demon.nl/webcam/index.html> abbiamo trovato un driver in grado di supportare le seguenti telecamere della Philips:

- PCA645VC
- PCA646VC
- PCVC675K (Vesta)
- PCVC680K (Vesta Pro)
- PCVC690K (Vesta Scanner)
- PCVC730K (ToUCam Fun)
- PCVC740K (ToUCam Pro)

Il driver si basa su un modulo che, dopo essere stato compilato, può essere opportunamente inserito nel kernel in esecuzione.

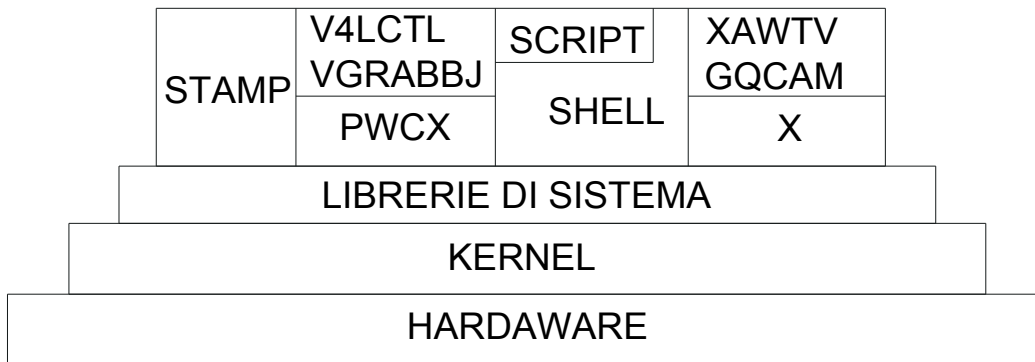


Fig. 10: Architettura di sistema alla luce dei componenti software necessari per la Philips Vesta ToUcam Pro

Per l'installazione seguire i seguenti passi:

scompattare il pacchetto digitando:

```
tar xvfz usb-pwc-???.tar.gz
```

ed entrare nella directory corrispondente:

```
cd usb-pwc-???
```

Saranno state create delle sottodirectory diverse a seconda delle varie versioni del kernel di Linux. Le directory per i kernel della famiglia 2.2 dovranno contenere i seguenti file:

```
-rw----- 1 root    root    45480 Apr  6 02:06 usb-pwc-6.3-smp.o
-rw----- 1 root    root    45224 Apr  6 02:06 usb-pwc-6.3-up.o
```

mentre quelli della serie 2.4 i file:

```
-rw----- 1 root    root    46416 Apr  6 02:06 usb-pwc-6.3-p5-smp.o
-rw----- 1 root    root    46192 Apr  6 02:06 usb-pwc-6.3-p5-up.o
-rw----- 1 root    root    46290 Apr  6 02:06 usb-pwc-6.3-p6-smp.o
-rw----- 1 root    root    46098 Apr  6 02:06 usb-pwc-6.3-p6-up.o
```

Entrare nella directory corrispondente al proprio kernel; per sapere quale kernel è installato sul proprio calcolatore è sufficiente digitare:

```
uname -r
```

I moduli con l'appellativo "p5" sono compilati per normali processori Pentium mentre quelli con "p6" per Pentium II o superiori. Per quanto riguarda i kernel 2.2 non vi è

nessuna distinzione perché non necessaria. I moduli con l'appellativo "up" si riferiscono a calcolatori con architettura a singolo processore mentre quelli con "smp" per architetture con più processori presenti.

Dopo aver individuato il file opportuno copiarlo nella directory:

/lib/modules/kernel-version/kernel/drivers/usb per quanto riguarda i kernel della famiglia 2.4; mentre:

/lib/modules/kernel-version/usb per i kernel della famiglia 2.2..

Digitando:

```
modprobe usb-pwc-???.o
```

modprobe dovrebbe reperire il modulo nella directory "usb". Se così non fosse è comunque possibile installarlo utilizzando insmod:

```
insmod usb-pwc-???.o
```

Dato che il driver è già stato compilato per uno specifico kernel si potrebbero avere dei problemi con le distribuzioni della RedHat, della Mandrake, ecc... Ad esempio per la Mandrake avente un kernel del tipo 2.2.19 digitando *uname -r* apparirà invece un codice di versione del tipo 2.2.19-15mdk. Ciò causa problemi di compatibilità con i moduli già precompilati. Per ovviare a ciò è possibile forzare l'inserimento del modulo digitando:

```
insmod -f usb-pwc-???.o
```

A questo punto è possibile, se non sono ancora presenti (vedi comando *lsmod*), caricare i seguenti moduli:

```
insmod videodev
insmod usbcore
insmod usb-uhci; oppure:
insmod usb-ohci
```

Se questi moduli non sono rintracciabili o caricabili (controllare */var/log/messages*) probabilmente è necessario ricompilare il kernel. Per la ricompilazione del kernel rimandiamo il lettore al capitolo 5.4.

Digitando *lsmod* dovrebbe apparire una schermata del tipo:

Module	Size	Used by
...
usb-pwc-???	???	0 (unused)
videodev	4512	1 [usb-pwc-???]
usb-uhci	20672	0 (unused)
usbcore	47248	1 [usb-pwc-??? usb-uhci]
...

Il driver presenta la compatibilità con i programmi basati su Video4Linux e quindi per vedere ciò che acquisisce la telecamera in formato video è possibile utilizzare *xawtv* nella stessa maniera precedentemente spiegata per la web-camera della Logitech. Analogamente per catturare le immagini è possibile utilizzare *v4lctl* con le stesse opzioni viste precedentemente:

```
v4lctl snap jpeg full <nomefile.jpeg> setnorm PAL capture grabdisplay fullscreen
```

oppure *vgrab* di cui parleremo più approfonditamente nel capitolo 5.3.

E' possibile durante il caricamento del modulo specificare opzionalmente i parametri con i quali dovrà lavorare:

```
modprobe usb-pwc-???.o <parametri>
```

Descriviamo qui in breve i parametri che il modulo può accettare:

- Size <val>: indica la risoluzione con la quale il driver acquisisce le immagini:

valori accettabili dal parametro	Risoluzione
Sqcif	128x96
Qsif	160x120
qcif	176x144
Sif	320x240
Cif	352x288
vga	640x480

- fps <val>: indicante i frame per secondo
- palette <val>: specifica la modalità di palette da usare; le palette supportate sono le seguenti: bgr24, rgb24, rgb32, bgr32, yuyv, yuv420, yuv420p
- fbufs <val>: Questo parametro specifica il numero di buffer interni che è possibile utilizzare per immagazzinare un frame. Il valore può andare da 2 a 5 (per default è pari a 3)
- mbufs <val>: Questo parametro specifica il numero di buffer interni da riservare a certe funzioni come mmap, vidiocgmbuf, vidiocmcapture, ecc...Questo parametro può variare da 1 a 4 e per default vale 2
- power_save: quando l'opzione è abilitata il modulo proverà a spegnere la telecamera quando viene fatta una chiamata di close e a riaccenderla di fronte a una chiamata di open.
- compression <val>: indica il fattore di compressione con il quale la web-camera spedisce i dati sulla porta USB
- trace: per attivare il debug-level

Il driver è in grado di supportare più telecamere Philips contemporaneamente. La larghezza di banda dell'USB limita il loro utilizzo: in teoria è possibile utilizzare 2 telecamera a media risoluzione, ma abbassando la risoluzione è possibile alzare il numero delle web-camere fino a un massimo di 5.

5.3. VGrabbj

Vgrabbj è un grabber basato su Video4Linux. Per le ultime versioni del programma e informazioni a riguardo rimandiamo ai siti <http://vgrabbj.gecius.de> oppure <http://sourceforge.net/projects/vgrabbj>.

Questo programma è stato sviluppato basandosi su web-camere della Philips, ma naturalmente, essendo basato su tecnologia Video4Linux, è compatibile anche con ogni driver di web-camera che si basa sullo stesso principio, quale ad esempio qce-ga nel caso della Logitech Express Quickcam.

Per installare il pacchetto digitare:

```
tar xvfz vgrabbj-???.tar.gz
```

entrare poi nella directory corrispondente e digitare nell'ordine:

./configure

per controllare che nel sistema siano presenti tutte le librerie necessarie

./make

per compilare il programma e creare i file binari

./make install

per installare il programma.

Per una corretta installazione di vgrabbj sono necessarie le seguenti librerie:

- libjpeg (<ftp://ftp.uu.net/graphics/jpeg/jpegsrc.v6b.tar.gz>)
- libpng (<ftp://ftp.cdrom.com/pub/png/src/libpng-1.0.6.tar.gz>)
- libz (<http://www.cdrom.com/pub/infozip/zlib/zlib.tar.gz>)
- Freetype (<http://freetype.sourceforge.net>) opzionalmente per avere funzioni di timestamp

Digitando vgrabbj -h apparirà una schermata del tipo:

```
vgrabbj, Version 0.7.4
sage: vgrabbj [options]
-h                This screen
-l <seconds>      Daemonize & sleep <seconds> (min. 1!) between
                  images
-L <microseconds> Daemonize & sleep <microseconds> between images
-b                Switch vgrabbj's brightness adjustment (default:
                  off)
-q <quality>      Quality setting (1-100, default: 75), JPEG only
-i <sqcif|qsif|qcif|sif|cif|vga>
                  Sets the imagesize of input device to qcif=128x96,
                  qsif=160x120, qcif=176x144, sif=320x240,
                  cif=352x288, or vga=640x480 (default: 352x288)
-o <jpeg|png>      Output format (default: jpeg)
-f <filename>     Write to <filename> (default: /dev/stdout)
-d <device>       Read from <device> as input (default: /dev/video)
-w                Disable setting of image-size, necessary for
                  certain cams e.g. IBM USB-Cam, QuickCam)
-s <device>       See capabilities of <device>
-S                Switch BGR colormap to RGB colormap (try if colors
                  are odd)
-D <0|2|3|4|6|7>  Set log/debug-level (0=silent, 7=debug, default:4)
-V                Display version information and exit
-F <value>        Force usage of specified palette (see videodev.h
                  for values)
Fallback to supported palette, if this one is not supported
```

Example: vgrabbj -l 5 -f /usr/local/image.jpg

Would write a single jpeg-image to image.jpg approx. every five seconds
The video stream has to one of RGB24, RGB32, YUV420, YUV420P or YUYV.

E' quindi sufficiente in prima approssimazione digitare:

vgrabbj -f <filename.jpeg> -o jpeg -q 50

per catturare l'immagine dalla web-camera salvandola su file in formato jpeg con una qualità ottimale.

5.4. Settaggio del kernel per USB

Se non sono caricabili i moduli videodev, audio e usb-uhci è necessario ricompilare il kernel di Linux. Per fare ciò portarsi nell'opportuna directory:

```
cd/usr/src/linux
```

e a questo punto, per configurare opportunamente i settaggi necessari, digitare:

```
make menuconfig; oppure make xconfig
```

Apparirà così una schermata con le varie opzioni da settare.

Eeguire le seguenti istruzioni:

- Nella sessione "Loadable Module Supporte": settare a **OFF** la voce "Set Version Information"
- Per quanto riguarda la famiglia dei kernel 2.4 nella sessione "Input core support": settare a **ON** "Input Core Support" e a **ON** "Event interface support" per abilitare il bottone a disposizione sulla web-camera
- Nella sessione "Sound" se si ha una sound-blaster e la web-camera ha un microfono settare a **ON** la voce "Sound Card Support"
- Per quanto riguarda la famiglia 2.2 dei kernel nella sessione "Character device" settare a **ON** la voce Video4Linux. E' anche possibile trovare una omonima sessione per Video4Linux. Per la famiglia dei kernel 2.4 nella sessione "Multimedia device" settare a **ON** la voce "Video for Linux" e la voce "information in proc filesystem". Questa naturalmente per permettere l'acquisizione delle immagini della web-camera
- Nella sessione "USB Support" settare a **ON** la voce "Support for USB" e abilitare o UHCI oppure OHCI e settare a **ON** la voce "USB Audio Support"

Dopo avere fatto le suddette modifiche salvare le impostazioni e uscire.

A questo punto è necessario digitare quanto segue:

```
make dep
```

```
make zImage
```

```
make modules
```

```
make modules_install
```

E' possibile raggruppare i comandi per automatizzare l'intera procedura:

```
make dep zImage modules modules_install
```

In ultima istanza digitare:

```
make install
```

per installare i kernel e i moduli compilati nella posizione adeguata.

Possiamo ritrovare tutto nelle seguenti directory:

```
/boot/System.map -> System.map-2.x.x
```

```
/boot/System.map-2.x.x
```

```
/boot/vmlinuz
```

```
/boot/vmlinuz-2.x.x
```

e i moduli in:

```
/lib/modules/2.x.x
```

A questo punto modificare il file “/etc/lilo.conf” per includere il nuovo kernel e riavviare il calcolatore.

5.5. Settaggio dei device-file necessari

Può essere necessario assicurarsi che i file di device siano effettivamente presenti nel sistema. Per fare ciò digitare:

```
ls -lad /dev/video*
```

ciò comporterà un elenco dei video-device:

```
lrwxrwxrwx  1 root  root  11 Aug 15 21:07 /dev/video -> dev/video0
crw-rw----  1 root  video  81,  0 Nov 25 1998 /dev/video0
crw-rw----  1 root  video  81,  1 Nov 25 1998 /dev/video1
crw-rw----  1 root  video  81,  2 Nov 25 1998 /dev/video2
crw-rw----  1 root  video  81,  3 Nov 25 1998 /dev/video3
```

Normalmente questi file sono già presenti se è installato Video4Linux. Se non dovessero essere presenti e non si volesse ricompilare il kernel si potrebbero creare manualmente.

Dopo essersi loggati come Root eseguire i seguenti comandi:

```
cd /dev
for ii in 0 1 2 3 4; do
  mknod video$ii c 81 $ii
  chown root.root video$ii
  chmod 600 video$ii
done
ln -s /dev/video /dev/video0
```

per creare i device-file necessari con i corretti “major” e “minor device number”. Notiamo che il “major-number” deve essere obbligatoriamente 81 mentre il “minor-number” può variare da 0 a 63 e quindi complessivamente potranno esserci 64 video-device.

6. Codice prodotto

6.1. Script

Qui di seguito riportiamo gli script da noi scritti: “Qcamday” per la cattura delle immagini durante le ore diurne, “Qcamnight” per la cattura delle immagini nelle ore notturne, “Storage” per la gestione dell’archivio delle immagini e infine “Install” per l’installazione degli script e della pagina HTML.

6.1.1. QCAMDAY

```
#!/bin/bash
```

```

#
# Questo script aggiorna l'immagine del laboratorio di robotica
# presa dalla Connectix Quickcam collegata a NOFIVE nelle ore diurne

CQCAMDIR="/usr/local/bin"
FIGDIR="/home/nofive/httpd/html"
STAMPDIR="/usr/local/bin"
COMDIR="/etc/cron.d"

rm -f $FIGDIR/temp.jpg
$CQCAMDIR/cqcam -32- -j -l 5 -s 1 >$FIGDIR/temp.jpg 2>/dev/null
sleep 2

rm $FIGDIR/mycam.jpg
$STAMPDIR/stamp -r /root/.stamprc

if [ -f $FIGDIR/mycam.jpg ]; then ftp frank<$COMDIR/comqcam.txt
  #Picture taken, stamped and send
else
  cp $FIGDIR/nocam.jpg $FIGDIR/temp.jpg
  $STAMPDIR/stamp -r /root/.stamprc

  ftp frank <$COMDIR/comqcam.txt

  # No camera attached
fi

# Fine script

```

Comqcam.txt è un file che funge da pipe contenente i comandi che devono essere eseguiti via FTP. Il suo unico compito è quello di spedire correttamente l'ultima immagine scattata a Frank. Il suo aspetto è il seguente:

```

binary
send /home/nofive/httpd/html/mycam.jpg webcam/arla.jpg
exit

```

6.1.2. QCAMNIGHT

```

#!/bin/bash
#
# Questo script aggiorna l'immagine del laboratorio di robotica presa
# dalla Connectix Quickcam collegata a NOFIVE nelle ore notturne

CQCAMDIR="/usr/local/bin"
FIGDIR="/home/nofive/httpd/html"
STAMPDIR="/usr/local/bin"
COMDIR="/etc/cron.d"

rm -f $FIGDIR/temp.jpg
$CQCAMDIR/cqcam -32- -j -l 5 -s 1 >$FIGDIR/temp.jpg 2>/dev/null
sleep 2

rm $FIGDIR/mycam.jpg
$STAMPDIR/stamp -r /root/.stamprc
if [ -f $FIGDIR/mycam.jpg ]; then ftp frank <$COMDIR/comqcam.txt

```

```

#Picture taken, stamped and send
else
  cp $FIGDIR/nocam.jpg $FIGDIR/temp.jpg
  $STAMPDIR/stamp -r /root/.stamprc

  ftp frank <${COMDIR}/comqcam.txt
  # No camera attached
fi

# Fine script

```

Comqcam.txt è lo stesso file già citato precedentemente per quanto riguarda Qcamday avente il compito di contenere le istruzioni da eseguire via FTP.

6.1.3. STORAGE

```

#!/bin/bash
#
# Questo script gestisce l'archivio delle immagini

COMDIR="/etc/cron.d"

sleep 20
ftp frank <${COMDIR}/comstorage.txt
exit 0

# Fine script

```

Comstorage.txt è un file facente funzioni di pipe per quanto riguarda il comando FTP. In esso sono contenute le istruzioni da eseguire via FTP per la gestione dell'archivio. Il suo aspetto è il seguente:

```

binary
delete webcam/arla12.jpg
rename webcam/arla11.jpg webcam/arla12.jpg
rename webcam/arla10.jpg webcam/arla11.jpg
rename webcam/arla09.jpg webcam/arla10.jpg
rename webcam/arla08.jpg webcam/arla09.jpg
rename webcam/arla07.jpg webcam/arla08.jpg
rename webcam/arla06.jpg webcam/arla07.jpg
rename webcam/arla05.jpg webcam/arla06.jpg
rename webcam/arla04.jpg webcam/arla05.jpg
rename webcam/arla03.jpg webcam/arla04.jpg
rename webcam/arla02.jpg webcam/arla03.jpg
rename webcam/arla01.jpg webcam/arla02.jpg
rename webcam/arla.jpg webcam/arla01.jpg
send /home/nofive/httpd/html/mycam.jpg webcam/arla.jpg
exit

```

6.1.4. NOFIVEOFF

```

#!/bin/bash
#
# Questo script aggiorna l'immagine del laboratorio di robotica
# presa dalla Connectix Quickcam collegata a NOFIVE

```

```

FIGDIR="/home/nofive/httpd/html"
STAMPDIR="/usr/local/bin"
COMDIR="/etc/cron.d"

rm $FIGDIR/mycam.jpg
cp $FIGDIR/nocomp.jpg $FIGDIR/temp.jpg

$STAMPDIR/stamp -r /root/.stamprc

ftp frank <$COMDIR/comqcam.txt

# Fine script

```

6.1.5. INSTALL

```

#!/bin/bash
#
# This script install files about ARL-Online

DESTDIR="/etc/cron.d"
CRONDIR="/etc/rc.d/init.d"
CHKDIR="/usr/local/bin"
HTMLDIR="/home/nofive/httpd/html"

echo Installazione script e pagina HTML
echo
echo Controllo programmi necessari...

if [ -f $CHKDIR/cqcam ]; then echo CQCAM...OK
else
    echo Si prega di installare CQCAM
    exit 0
fi

if [ -f $CHKDIR/stamp ]; then echo STAMP...OK
else
    echo Si prega di installare STAMP
    exit 0
fi

cp qcamday qcamnight storage nofiveoff comqcam.txt comstorage.txt
$DESTDIR

cp nocomp.jpg nocam.jpg $HTMLDIR

if [ -f /root/.netrc ]; then chmod 600 /root/.netrc fi
cat netrc >>/root/.netrc
chmod 400 /root/.netrc

cat cron >>/etc/crontab

echo Riavvio del demone crond
$CRONDIR/crond restart

echo
echo Installazione Completata!!!

# Fine script

```

Citiamo anche *Install_frank.txt* che è un file contenente i comandi necessari per spedire via FTP la pagina HTML contenente l'ultima immagine catturata e la pagina HTML contenente l'archivio storico delle ultime immagini. Esso presenta le seguenti istruzioni:

```
binary
send arl_online.html webcam/arl_online.html
send sfarlch.gif webcam/sfarlch.gif
send scrilab.gif webcam/scrilab.gif
send storage.html webcam/storage.html
send image.gif webcam/image.gif
exit
```

6.2. Pagina HTML

Come precedentemente accennato abbiamo realizzato due versioni di pagina HTML. La prima si aggiorna automaticamente ogni 60 secondi mentre la seconda presenta un bottone di "Reload". Inoltre abbiamo prodotto una terza pagina per la gestione dell'archivio storico delle immagini.

Riportiamo qui di seguito i codici HTML di entrambe:

6.2.1. Pagina con Reload automatico

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <META content="text/html; charset=windows-1252" http-
equiv=Content-Type>
    <META content=60 http-equiv=refresh>
    <META content="Alessandro Saetti - Sergio Spinoni"
name="Author">
  </head>

  <body alink="#c0ffc0" background="sfarlch.gif" link="yellow"
text="white" vlink="silver" bottommargin="6" leftmargin="0"
marginheight="0" marginwidth="0" topmargin="6" rightmargin="0">
    <H1> <center> <font size="7" color="black">Advanced Robotics
Laboratory on-line </H1>
    <HR>
      <center>
        <table align="CENTER" width="900" height="117"
cellspacing="0" border="0">
          <tr>
            <TD ALIGN="right"></font><BR>
            </td>
          </tr>
        </table>

        <center>
          <font size="5" color="black">Here is the latest Web Cam
Picture...
```

```

        <p><font size="7" color="black"></td>
        <br>
        <i>
        <font size="5" color="black">Updates every 60 seconds, or until
your browser gives up </i></font></center>
        <BR>
        <font size="3" color="black">This is a picture of what's
appening in ARL !!!
        <BR>
        <BR>
        <font size="3" color="black"> Do you want to see last 100
photos? click <A href="http://nofive.ing.unibs.it/storage.html"
TITLE="ARL history"> <font size="3" color="blue">here </A> <font
size="3" color="brown">
        <BR>
        <BR>
        <HR>

<div align="right">
<SCRIPT language=JavaScript>
<!-- Hide this script from incompatible Web browsers! -->
document.writeln("<i>Last updated:  ",document.lastModified)
document.writeln("</font></i> ")
// Hide this script from incompatible Web browsers! -->
</SCRIPT>
</CENTER></P></div></BODY></HTML>

```

6.2.2. Pagina con tasto di Reload

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>

    <head>
        <META content="text/html; charset=windows-1252" http-
equiv=Content-Type>
        <META content="Alessandro Saetti - Sergio Spinoni"
name="Author">

        <SCRIPT LANGUAGE="JavaScript">

var timerID = null;
var timerRunning = false;

function stopclock ()
{
    if(timerRunning)
        clearTimeout(timerID);
    timerRunning = false;
}

function showtime ()
{
    var now = new Date();
    var hours = now.getHours();
    var minutes = now.getMinutes();

```

```

var seconds = now.getSeconds()
var timeValue = "" + ((hours >12) ? hours -12 :hours)

timeValue += ((minutes < 10) ? ":0" : ":") + minutes
timeValue += ((seconds < 10) ? ":0" : ":") + seconds
timeValue += (hours >= 12) ? " p.m." : " a.m."

document.clock.face.value = timeValue;

timerID = setTimeout("showtime()",1000);
timerRunning = true;
}

function startclock ()
{
    // Make sure the clock is stopped

    stopclock();
    showtime();
}
</SCRIPT>

</head>

<body onLoad="startclock(); timerONE=window.setTimeout"
alink="#c0ffc0" background="sfarlch.gif" link="yellow" text="white"
vlink="silver" bottommargin="6" leftmargin="0" marginheight="0"
marginwidth="0" topmargin="6" rightmargin="0" >

    <H1> <center> <font size="7" color="black">Advanced Robotics
Laboratory on-line </H1>
    <HR>

        <center>
            <table align="CENTER" width="900" height="117"
cellspacing="0" border="0">
                <tr>
                    <td align="right"></font><BR>
                    </td>
                </tr>
            </table>

            <center>
                <font size="5" color="black">Here is the latest Web Cam
Picture...
                <p><font size="7" color="black"></td><BR>

                <FORM><INPUT TYPE="button"onClick="history.go(0)" value="Reload
image">
                </FORM></CENTER>

                <font size="5" color="black"><i>Click button to refresh the
image!!! </i></font></center>

```



```

    <font size="3" color="black"><BR>This is a picture of what's
appening in ARL !!!
    <BR>
    <BR>
    <table align="left" width="400" cellspacing="0" border="0">
    <td width="250">
    <font size="3" color="black"> What time is it at Brescia
University?
    </td>
    <td width="150">
    <form name="clock" onSubmit="0">
    <input type="text" name="face" size=13 value=""></font>
    </td>
    </table>
    <BR>
    <BR>
    <div align="left">
    <font size="3" color="black"> Do you want to see last 100
photos? click <A href="http://nofive.ing.unibs.it/storage.html"
TITLE="ARL history"> <font size="3" color="blue">here </A>
    </div>
    <BR>
    <BR>
    <HR>
    <font size="3" color="brown"> <div align="right">
    <SCRIPT language=JavaScript>
    <!-- Hide this script from incompatible Web browsers! -->
    document.writeln("<i>Last updated: ",document.lastModified)
    document.writeln("</font></i> ")
    // Hide this script from incompatible Web browsers! -->
    </SCRIPT>
    </CENTER></P></div></BODY></HTML>

```

6.2.3. Pagina per la “history”

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>

    <head>
    <META content="text/html; charset=windows-1252" http-
equiv=Content-Type>
    <META content="Alessandro Saetti - Sergio Spinoni"
name="Author">
    <META NAME="PRAGMA" CONTENT="no-cache">
    </head>

    <body alink="#c0ffc0" background="sfarlch.gif" link="yellow"
text="white" vlink="silver" bottommargin="6" leftmargin="0"
marginheight="0" marginwidth="0" topmargin="6" rightmargin="0">
    <H1> <center> <font size="7" color="black">Advanced Robotics
Laboratory history </H1> </center>
    <HR>
    <align left>
    <font size="3" color="black">These are the latest 12 photos
about ARL...
    <BR>

```

```
<BR>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla01.jpg" TARGET="new"
TITLE="ARL's image 5 minute ago"> <font size="3" color="blue"> 5
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla02.jpg" TARGET="new"
TITLE="ARL's image 10 minute ago"> <font size="3" color="blue"> 10
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla03.jpg" TARGET="new"
TITLE="ARL's image 15 minute ago"> <font size="3" color="blue"> 15
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla04.jpg" TARGET="new"
TITLE="ARL's image 20 minute ago"> <font size="3" color="blue"> 20
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla05.jpg" TARGET="new"
TITLE="ARL's image 25 minute ago"> <font size="3" color="blue"> 25
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla06.jpg" TARGET="new"
TITLE="ARL's image 30 minute ago"> <font size="3" color="blue"> 30
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla07.jpg" TARGET="new"
TITLE="ARL's image 35 minute ago"> <font size="3" color="blue"> 35
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla08.jpg" TARGET="new"
TITLE="ARL's image 40 minute ago"> <font size="3" color="blue"> 40
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla09.jpg" TARGET="new"
TITLE="ARL's image 45 minute ago"> <font size="3" color="blue"> 45
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla10.jpg" TARGET="new"
TITLE="ARL's image 50 minute ago"> <font size="3" color="blue"> 50
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla11.jpg" TARGET="new"
TITLE="ARL's image 55 minute ago"> <font size="3" color="blue"> 55
minutes ago </A>
<BR>
<IMG alt=[DIR] src="image.gif"> <A
href="http://frank.ing.unibs.it/webcam/arla12.jpg" TARGET="new"
```

```

TITLE="ARL's image 1 hour ago" <font size="3" color="blue"> 1 hour
ago </A> <font size="3" color="brown">
  <BR>

  <BR>
  <HR>

  <div align="right">
  <SCRIPT language=JavaScript>
  <!-- Hide this script from incompatible Web browsers! -->
  document.writeln("<i>Last updated:  ",document.lastModified)
  document.writeln("</font></i> ")
  // Hide this script from incompatible Web browsers! -->
  </SCRIPT>
  </CENTER></P></div></BODY></HTML>

```

7. Preview della pagina HTML

Riportiamo l'anteprima delle pagine HTML prodotte. La prima riguarda la visualizzazione dell'ultima immagine scattata mentre la seconda è una piccola raccolta delle immagini più recenti. Come precedentemente accennato ci siamo prefissi di tenerne salvate una dozzina; ciascuna di queste immagini viene conservata ogni 5 minuti e quindi si può con tale pagina avere un quadro approssimativo di ciò che è successo in laboratorio nell'ultima ora.

7.1. Advanced Robotics Laboratory on-line

Advanced Robotics Laboratory on-line



Here is the latest Web Cam Picture...



Updates every 60 seconds, or until your browser gives up

This is a picture of what's appening in ARL !!!

What time is it at Brescia University?

3:16:50 p.m.

photos about last hour? click [here](#)

Do you want to see

Last updated: 09/01/2001 13:10:42

7.2. Advanced Robotics Laboratory history

Advanced Robotics Laboratory history

These are the latest 12 photos about ARL...



[5 minutes ago](#)



[10 minutes ago](#)



[15 minutes ago](#)

-  [20 minutes ago](#)
-  [25 minutes ago](#)
-  [30 minutes ago](#)
-  [35 minutes ago](#)
-  [40 minutes ago](#)
-  [45 minutes ago](#)
-  [50 minutes ago](#)
-  [55 minutes ago](#)
-  [1 hour ago](#)

Last updated: 09/01/2001 11:27:50

8. Feature

Presentiamo qui di seguito alcune immagini catturate con le web-camera private.

Le prime sono state scattate in laboratorio e sono frutto del lavoro di Cqcam e delle possibilità offerte dalla Connectix Color Quickcam.

Le ultime sono foto scattate con la Logitech Quickcam Express utilizzando il software "v4lctl" frutto del lavoro da noi condotto a casa. Da queste ultime è possibile notare la risoluzione e la qualità delle immagini che si possono ottenere mediante detta telecamera; in modo da deciderne un eventuale impiego.



Fig. 11 e 12: Immagini scattate in laboratorio con la Connectix Color Quickcam



Fig. 13 e 14: Immagini scattate a casa con la Logitech Color Quickcam

Infine riportiamo le due immagini da noi create che segnalano a un ignaro utente Internet le possibili situazioni di errore che ci siamo prefissati di fronteggiare:

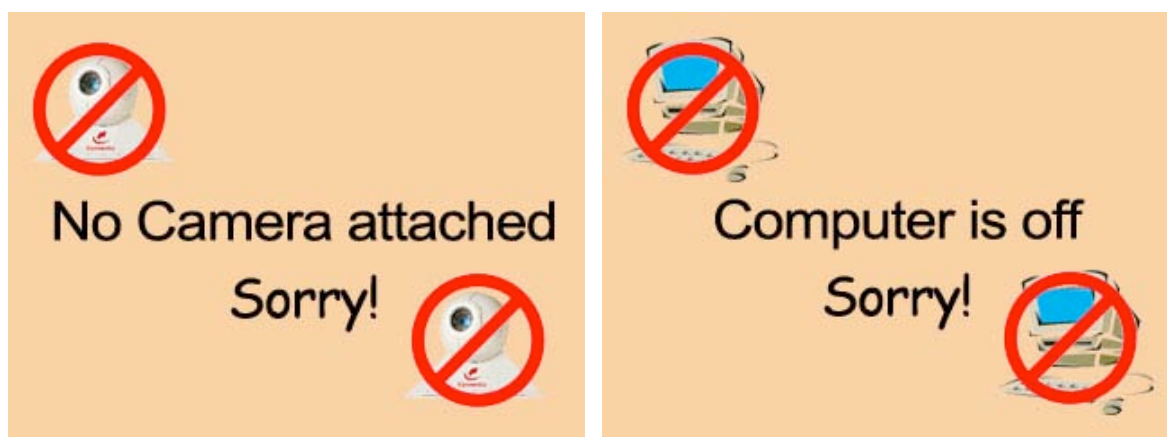


Fig. 15 e 16: A sinistra l'immagine che segnala l'assenza della web-camera. A destra un'immagine che segnala il fatto che il computer al quale è attaccata la telecamera è spento.

Bibliografia

Gran parte del materiale raccolto su cui ci siamo basati per condurre il nostro elaborato deriva direttamente da Internet. Per quanto concerne testi possiamo citare:

[1] Medri, D.: *Linux Facile*, 2001.

Indice

SOMMARIO	1
1. INTRODUZIONE.....	1
2. IL PROBLEMA AFFRONTATO.....	1
2.1. Scelta di una web-camera opportuna	2
2.2. Reperimento in Internet del software	2
2.3. Installazione della web-camera	4
2.4. Configurazione di Linux	4
2.5. Creazione di script e pagine HTML	7
3. LA SOLUZIONE ADOTTATA.....	7
4. MODALITÀ OPERATIVE.....	10
4.1. Componenti necessari	10
4.1.1. Componenti hardware	10
4.1.2. Componenti software	10
4.1.2.1. Cqcam	10
4.1.2.2. Stamp	11
4.1.2.3. Script & Pagina HTML	12
4.2. Modalità di installazione	14
4.2.1. Installazione dell'hardware	15
4.2.2. Installazione del software	15
4.2.2.1. Cqcam (./cqcam-0.91)	16
4.2.2.2. Stamp (./stamp-2.0.8)	17
4.2.2.3. Script di Install	17
4.3. Modalità di taratura	15
4.3.1. Cqcam	19
4.3.2. Stamp	20
4.4. Avvertenze	22
5. CONCLUSIONI E SVILUPPI FUTURI.....	23
5.1. Logitech Quickcam Express (USB)	24

5.1.1.	Logitech Quickcam Express (USB)	24
5.1.2.	Linux Driver per Logitech Quickcam Express with HDCS1000/PB0100 Sensor	28
5.2.	Philips Vesta ToUcam Pro (USB)	28
5.2.1.	Supporto Linux per Philips USB Web-cameras	29
5.3.	VGrabbj	32
5.4.	Settaggio del Kernel per USB	33
5.5.	Settaggio dei device-file necessari	34
6.	CODICE PRODOTTO.....	35
6.1.	Script	35
6.1.1.	Qcamday	35
6.1.2.	Qcamnight	36
6.1.3.	Storage	36
6.1.4.	NoFiveoff	37
6.1.5.	Install	37
6.2.	Pagina HTML	38
6.2.1.	Pagina con Reload automatico	39
6.2.2.	Pagina con tasto di Reload	40
6.2.3.	Pagina per la "history"	42
7.	PREVIEW DELLA PAGINA HTML.....	43
7.1.	Advanced Robitcs Laboratory on-line	44
7.2.	Advanced Robitcs Laboratory history	45
8.	FEATURE.....	46
BIBLIOGRAFIA.....		47
INDICE		47