

**CONSIGLIO NAZIONALE DELLE RICERCHE**

**ISTITUTO DI SCIENZA E TECNOLOGIE DELL'INFORMAZIONE**

**Contratto di collaborazione NES s.r.l. – ISTI/CNR**

**Sviluppo software per acquisizione di immagini  
mediante dispositivo portatile**

Sebastiano Bazzano, Graziano Bertini, Lucio Di Giovannantonio

B4-09      aprile 2003

## **Sviluppo software per acquisizione di immagini mediante dispositivo portatile**

Sebastiano Bazzano\*, Graziano Bertini\*\*, Lucio Di Giovannantonio\*\*\*

- \* Laureando in Ingegneria Elettronica presso l'ISTI
- \*\* ISTI -CNR
- \*\* Collaboratore associato alla ricerca

Al gruppo di lavoro per la definizione delle specifiche di alcune parti del sistema hanno partecipato, da parte della NES, Giampiero Di Simone e per l'ISTI Vincenzo Di Salvo e Luca Tarrini.

## INDICE

1. Introduzione .....	4
2. Specifiche di progetto del sistema software.....	5
3. Funzionamento del sistema.....	7
3.1 Descrizione dei menù.....	7
3.1.1 Menù Live .....	8
3.1.2 Menù File .....	9
3.1.3 Menù Download.....	9
4. Passaggio delle immagini su desktop .....	11
5. Struttura del software .....	12
5.1 SDK per FlyJacket .....	12
5.2 Elenco API fornite .....	12
6. Compressione Jpeg .....	17
7. Guida in linea .....	18
8. Setup iPAQ .....	19
9. Conclusioni .....	21
10. Bibliografia. ....	21

# 1. Introduzione

I notevoli sviluppi nel campo dell'elettronica per pc hanno consentito la nascita e lo sviluppo di computer di dimensioni sempre più ridotte, così oltre ai computer portatili (notebook) si stanno diffondendo con grande rapidità i computer palmari, pc di dimensioni ridottissime tanto da poter entrare in una tasca o da tenere nel palmo di una mano, da cui il nome. Fino a qualche anno fa questi dispositivi erano relegati a ruolo di semplici agende elettroniche per tenere appuntamenti, ma negli ultimi anni l'introduzione di componenti sempre più ridotti hanno portato le prestazioni a livelli più che soddisfacenti per capacità elaborative, anche di grosso carico, così questi dispositivi stanno entrando sempre di più nell'uso comune.

Tra le prestazioni offerte da questi dispositivi c'è anche la possibilità di gestire il segnale videocomposito, in modo da poter rilevare sequenze video mediante collegamento a telecamere, così nell'ambito di un progetto conto/terzi dell'ISTI-CNR con la società NES S.r.l. di Pisa, è stato deciso di utilizzare un computer palmare per acquisizione immagini, mediante la cattura di fotogrammi di una sequenza video fornita da un dispositivo da usare in abbinamento al palmare, e successiva archiviazione su personal computer.

Il materiale è stato fornito dalla NES S.r.l. e si tratta di un palmare della Compaq, iPAQ 3870, di un dispositivo per acquisizione video Fly-Jacket con relativo SDK per sviluppatori; per lo sviluppo del software è stato utilizzato l'ambiente di sviluppo per sistemi embedded della Microsoft, l'Embedded Visual Tools 3.

## 2. Specifiche di progetto del sistema software

L'architettura del sistema che dovrà utilizzare l'iPAQ e gli obiettivi di massima sono stati indicati direttamente dalla NES S.r.l., mentre le specifiche software sono state formulate in collaborazione (dopo diversi incontri e approfondimenti) e riguardano il funzionamento generale del dispositivo, la struttura dell'interfaccia grafica e le funzionalità.

Il dispositivo deve eseguire le funzioni mediante l'uso di pulsanti esterni azionabili a mano mediante un kit assemblato appositamente dalla ditta NES, un po' come avviene nei bancomat; questa richiesta rende inutilizzabili le funzionalità del dispositivo (iPAQ-FlyJacket), si è reso necessario riscrivere le procedure fornite dal dispositivo in modo da poter essere utilizzate secondo tale specifica.

Dovremo realizzare delle modalità di input diverse da quelle standard, cioè non si potrà usare i click del mouse su pulsanti graficati sul display, né i tasti della tastiera software. Quindi nel nostro caso i pulsanti che permetteranno di accedere alle funzionalità principali sono esterni all'interfaccia nativa e il loro funzionamento andrà simulato mediante l'invio di opportuni messaggi dall'hardware che realizza i pulsanti nell'applicativo.

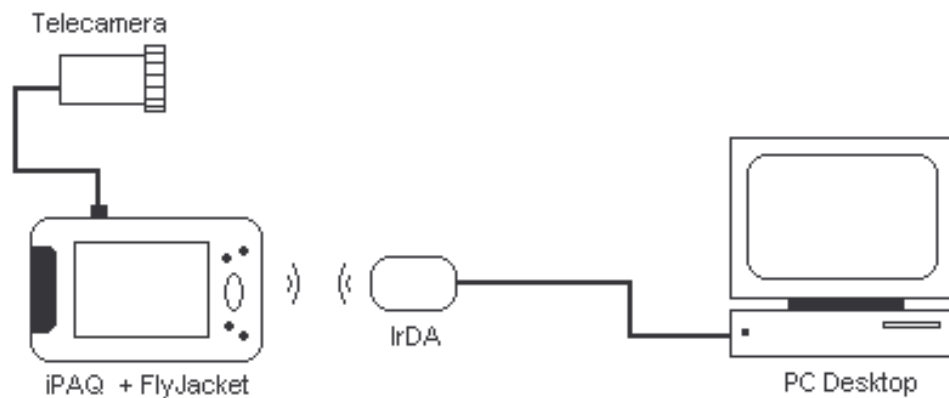
In fase di messa a punto è stata utilizzata una tastierina per palmari dalla quale sono stati isolati i quattro tasti cursore, in modo da poter intercettare l'evento "*pressione del tasto x*" e poter inviare il messaggio corretto all'applicazione per fare eseguire l'azione X. Un altro aspetto dell'interfaccia grafica richiesta è quello di mascherare completamente il sistema operativo, mediante la visualizzazione a schermo intero, in modo da nascondere le barre degli strumenti e altri menù presenti nella modalità nativa, e impedire l'accesso alle funzionalità di base offerte dal dispositivo. E' stato richiesto inoltre che il palmare possa funzionare in modo da presentare il display con la dimensione maggiore in posizione orizzontale, questo per avere una maggiore area di visualizzazione per preview delle immagini; la risoluzione standard degli schermi di questi dispositivi è di 240x320, e tenendo conto che bisogna inserire delle icone (a fianco delle immagini stesse) che descrivono le funzionalità dei pulsanti esterni, lo spazio a disposizione per la visualizzazione del flusso video è di dimensioni piuttosto ridotte, invertendo la presentazione si può sfruttare una risoluzione di 320x240 che permette di avere una maggiore area visibile per la visualizzazione del video.

Le funzionalità richieste devono essere tali da consentire l'avvio della telecamera per la visualizzazione del flusso video; quando l'operatore lo desidera preme il tasto che permette di acquisire il frame video correntemente visualizzato, si apre così la schermata che permette di vedere la qualità della foto così scattata; a questo punto si può o scattare la foto e tornare alla ripresa con la

telecamera oppure, se si decide di accettare la foto, si possono inserire un paio di righe di commento, salvare la foto e tornare alla visualizzazione del video per acquisire altre foto. Deve essere possibile scorrere le foto acquisite per decidere quali eliminare e/o stampare (tramite una stampantina locale). In un secondo tempo si deve effettuare lo scarico delle foto scattate su un pc desktop in un ufficio, posizionando il palmare in un contenitore realizzato appositamente dalla NES e utilizzando per il trasferimento la porta a raggi infrarossi con un opportuno protocollo .

A questo scopo è stata fornita la sequenza dei menù da visualizzare per rispettare le specifiche, la loro descrizione verrà dettagliata quando verrà illustrata la realizzazione del software.

A causa della limitata capacità di memoria del dispositivo, (64 MB di RAM), è stata ipotizzato di salvare le foto in un formato grafico compresso come il jpeg al posto del formato bitmap, anche a scapito della qualità delle immagini; da prove effettuate si è notato che tale compressione non influisce in modo apprezzabile sulla qualità dell'immagine.



**Configurazione generale del sistema di acquisizione e gestione di immagini**

### 3. Funzionamento del sistema.

L'applicazione si presenta mediante una sequenza di schermate che rappresentano i menù delle funzionalità offerte, secondo le specifiche fornite.

I tasti utilizzati per attivare le funzionalità sull'IPAQ sono i tasti direzione rispettivamente:

- . Freccia Alto per la prima selezione in alto
- . Freccia Basso per la seconda selezione
- . Freccia Destra Per la terza
- . Freccia Sinistra per la quarta.

La schermata principale (fig.1) visualizza il logo della NES consente di avviare il programma mediante la pressione del tasto "Avanti", il tasto "Help" consente di accedere una piccola guida in linea che illustra l'uso dei vari menù, mentre il tasto "About" apre una finestra che mostra le informazioni sul programma.



Figura 1 Schermata principale dell'applicazione

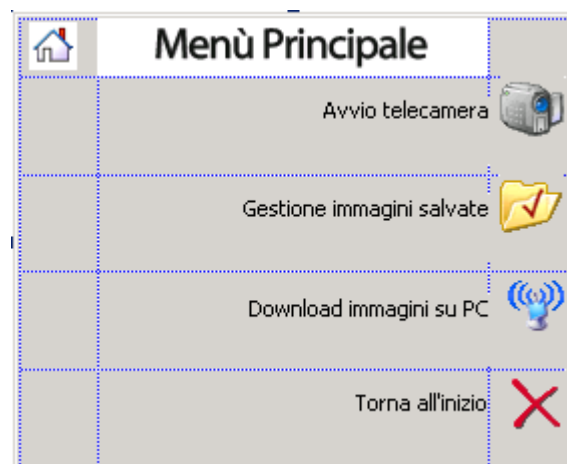


Figura 2 Menù principale dell'applicazione

#### 3.1 Descrizione dei menù

Il menù principale, la cui schermata è mostrata in fig. 2 racchiude tutte le funzionalità del sistema, attivabili premendo il tasto relativo:

- "Avvio telecamera": apre il menù "Live" e consente di effettuare riprese e salvare immagini;
- "Gestione immagini salvate": consente la visualizzazione delle immagini finora salvate, ed eventualmente cancellarle e/o stamparle mediante una stampante a raggi infrarossi;

- “Download *immagini su PC*”: consente lo scarico delle immagini della giornata su un PC mediante il supporto a raggi infrarossi.

Di seguito vengono riportate le sequenze dei vari menù che compongono l'applicazione con le relative descrizioni delle funzionalità.

### 3.1.1 Menù Live

Il menù “*Live*” (fig.3) si apre premendo il tasto relativo alla descrizione “*Avvio telecamera*” nel menù principale e permette di visualizzare il flusso video e di catturare i frames desiderati; si compone di un sottomenù che permette di salvare le immagini catturate. La telecamera viene avviata mediante la pressione del tasto “*Avvia*”, inizia così la visualizzazione del flusso video, quando l'operatore pensa di avere un'immagine buona preme il tasto “*Cattura*” e il frame corrente viene catturato e salvato in un file temporaneo in formato bitmap; nel successivo menù “*Salva*” (fig.4) è possibile decidere se acquisire definitivamente l'immagine salvata o scartarla e tornare alla visualizzazione video premendo direttamente il pulsante “*Live*”. Prima di salvare l'immagine è possibile inserire due righe di testo come commento all'immagine, semplicemente inserendo il testo nelle due caselle di testo sotto l'immagine e premendo il tasto “*Accetta*”, una volta inserite le righe di commento mediante il tasto “*Salva*” l'immagine viene salvata in una cartella del dispositivo che porta il nome della data del giorno. Dopo il salvataggio si torna alla visualizzazione del video, ed è possibile continuare a scattare altre foto.



Figura 3 Menù Live

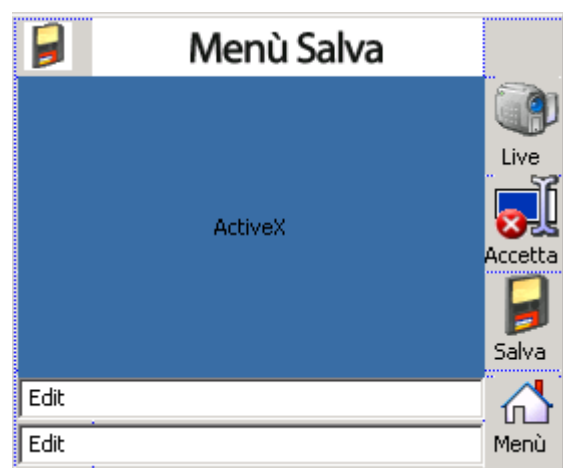


Figura 4 Menù Salva

Dopo la conversione jpeg dell'immagine salvata si apre il menù "Salva", che consente salvare definitivamente l'immagine sul dispositivo; all'apertura del menù viene mostrata l'immagine così come è stata salvata, se non è accettabile può essere cancellata semplicemente premendo il tasto "Live" che torna alla visualizzazione del flusso video senza salvare l'immagine sul dispositivo.

La gestione della bitmap e della scrittura del testo è realizzata mediante l'utilizzo di un componente ActiveX (S309 PictureBox) che facilita la visualizzazione delle immagini e permette di scrivere del testo in sovrimpressione sulla bitmap.

### 3.1.2 Menù File

Il menù "File" si apre premendo il tasto "Gestione immagini salvate" e permette di cancellare e/o stampare le immagini salvate e non prevede ulteriori sottomenù. Mediante il tasto "Indietro" si scorrono le immagini salvate dalla più recente, in maniera circolare, quando è selezionata l'immagine corretta può essere cancellata o stampata mediante gli opportuni tasti. Premendo il tasto "Cancella" viene chiesta conferma mediante apposite MessageBox opportunamente ridefinite per adattarle allo stile dell'interfaccia. Le immagini vengono caricate con l'ausilio del componente ActiveX S309 PictureBox che permette di caricare sia immagini in formato bitmap che jpeg.

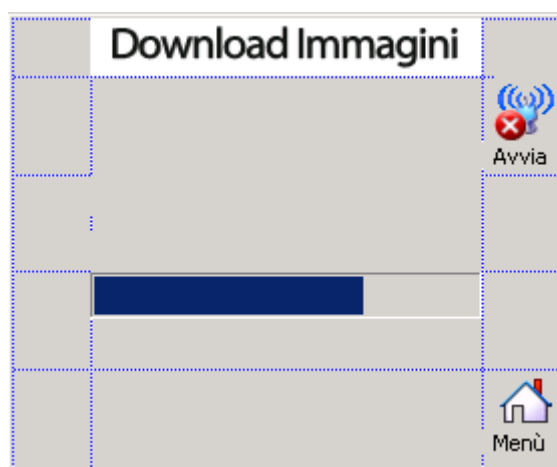


Figura 5 Menù File

### 3.1.3 Menù Download

Anche il menù "Download" (fig.6) non prevede ulteriori sottomenù e viene richiamato mediante la pressione del tasto "Download immagini su PC" dal menù principale; premendo il tasto

“Avvia” si inizia la procedura di scarico delle immagini salvata dal dispositivo al PC, mediante il supporto a raggi infrarossi.



**Figura 6 Menù Download**

## 4. Passaggio delle immagini su desktop

Il trasferimento delle immagini acquisite dal palmare verso il Pc desktop avviene creando una connessione di tipo Client/Server attraverso la porta infrarossi (IrDA). Accedendo al menù “Download”(fig. 6), dal menù principale dell’applicazione, è possibile trasferire le immagini catturate verso il PC di destinazione. Sul visore si potranno notare i seguenti tasti:

- “Avvia”.: Questa funzione avvierà il trasferimento delle immagini salvate, da notare che, nel caso in cui non vi sia alcuna immagine da trasferire, la funzione verrà disabilitata e sul display verrà raffigurata l’icona con una X rossa;
- “Menù”: Premendo questo tasto si abbandonerà il menù “Download” tornando al menù principale.

A centro schermo avremo delle informazioni sul numero di immagini da trasferire ed una stima del tempo necessario per completare l’operazione (in secondi). Per poter effettuare il download è necessario, prima di tutto, allineare le due porte ad infrarossi, attendere che il sistema operativo del Pc rilevi la disponibilità della connessione, eseguire il programma di ricezione sul Pc e cliccare su “Ricevi”, solo a questo punto sarà possibile premere il tasto “Avvia” sul palmare.

La procedura di invio si appoggia su socket, ovvero un’astrazione per la comunicazione basata sul protocollo TCP/IP, solo che a differenza dell’utilizzo standard su rete internet, viene adoperato sulla connessione IrDA tramite la dichiarazione:

```
SOCKADDR_IRDA address = { AF_IRDA, 0, 0, 0, 0, "IRServer" };
```

La trasmissione delle immagini avviene seguendo le specifiche dell’IrDA cioè a piccoli pacchetti (che nel nostro caso sono di 1KB), quindi è stato necessario un opportuno protocollo che ci garantisca la corretta sequenza di invio dei pacchetti stessi.

Per quanto riguarda la correttezza delle informazioni il protocollo IrDA inserisce un controllo su ogni pacchetto trasmesso, il Pc fa le opportune verifiche e successivamente rende disponibile il pacchetto sul buffer di ricezione per il completamento dei dati relativi all’immagine.

La struttura del protocollo prevede l’invio del numero di immagini da trasferire e successivamente in maniera ciclica vengono spediti i pacchetti relativi alla dimensione dell’immagine, il nome (completo di estensione), e il contenuto (sempre a pacchetti da 1KB) dell’immagine stessa.

Sul lato Server si riceveranno i pacchetti bufferizzati rispettando la sequenza originale di invio. Le immagini ricevute vengono archiviate all’interno della cartella di default (C:\Cattura\), e ordinate per data (nel formato gg-mm-aaaa) in varie sotto cartelle per facilitarne la consultazione.

## 5. Struttura del software

Come accennato il software è stato sviluppato utilizzando l'ambiente di sviluppo Embedded Visual C++ 3.0 che permette di sviluppare su personal computer il software e di scaricarlo poi sul dispositivo mediante il proprio software di gestione.

### 5.1 SDK per FlyJacket

L'acquisizione video è stata gestita mediante le funzionalità offerte dal produttore del FlyJacket mediante opportune librerie che permettono la gestione del dispositivo, oltre al driver per il funzionamento insieme al FlyJacket è stato fornito una serie di librerie e DLL che permettono agli sviluppatori di accedere alle funzionalità del dispositivo. Le funzionalità del FlyJacket sono racchiuse in un serie di funzioni richiamabili mediante il link di una DLL o libreria statica, e permettono di inizializzare il dispositivo, iniziare le riprese e catturare un frame.

### 5.2 Elenco API fornite

Di seguito viene riportato l'elenco delle API fornite dal prodotto del FLYJacket, per una descrizione piu' dettagliata si rimanda al manuale elettronico i formato pdf fornito con i driver del dispositivo, *FJ\_API\_PartI.pdf - FJ\_API\_PartII.pdf - FJ\_API\_PartIII.pdf - FJ\_API\_PartIV.pdf*.

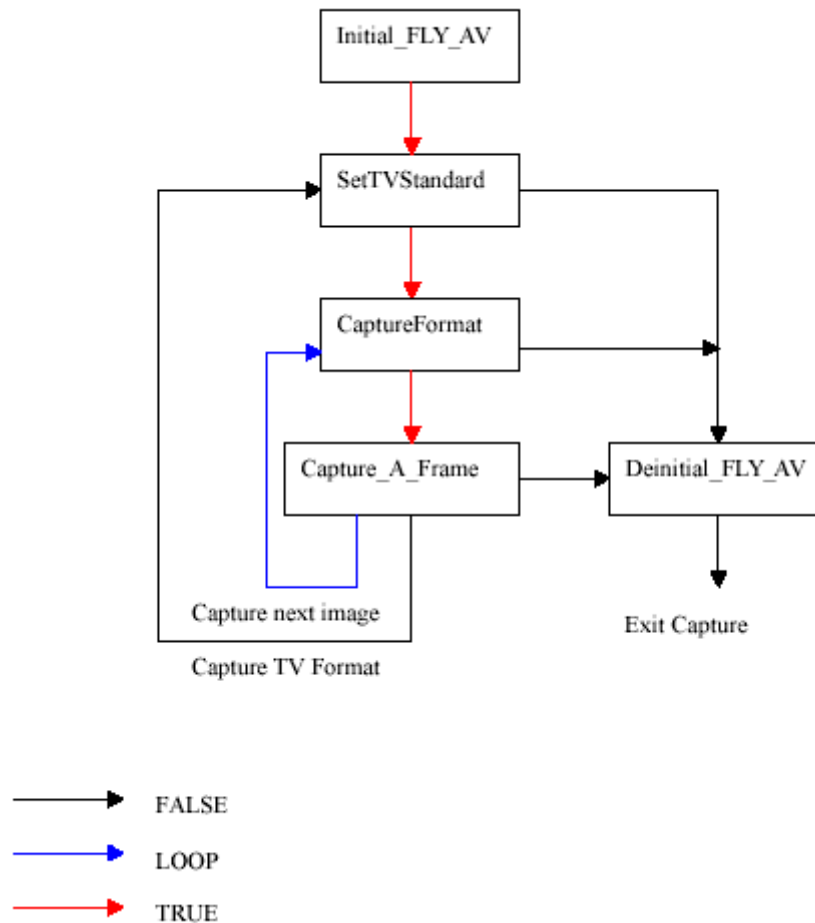
- **BOOL Initial\_FLY\_AV(void)**, permette di inizializzare il dispositivo;
- **BOOL SetTVStandard(DWORD \*pdTVStandard)**, imposta il video input standard (NTSC o PAL);
- **BOOL CaptureFormat(INT iWidth,INT iHeight,INT iType)**, permette di impostare il formato di cattura, dimensione e tipo (YUV o RGB);
- **BOOL Capture\_A\_Frame (LPBYTE pretBuff, DWORD \*pdSize)**, cattura un immagine dal video input;
- **void Deinitial\_FLY\_AV(void)**, deinizializza il dispositivo;
- **BOOL StartPreview(INT iWidth,INT iHeight,INT iType,INT iInputType)**, inizia la visualizzazione del flusso video;
- **BOOL StopPreview(void)**, ferma la visualizzazione del flusso video;
- **BOOL PreviewFrame(LPBYTE pretBuff)**, visualizza un frame;
- **BOOL InputSelect(UCHAR pdInputSelect)**, permette di selezionare l'input fra AV-IN o S-Video;
- **BOOL CheckFlyJacketPack(void)**, controlla la presenza del dispositivo;

Le seguenti funzioni permettono di regolare la qualita' dell'immagine (luminosita', saturazione etc.)

- **void Brightness(UCHAR \*uValue,UCHAR iMode);**

- **void Contrast(INT8 \*iValue,UCHAR iMode);**
- **void Saturation(INT8 \*iValue,UCHAR iMode); // Saturation:-128 ~ 127**
- **void Hue(INT8 \*iValue,UCHAR iMode); // Hue :-128**

L'utilizzo delle API descritto puo' essere rappresentato dal seguente diagramma a blocchi:



**Figura 7 Capture API control flow**

### Sample Code

```

...
...
iRetVal = Initial_FLY_AV();
if(iRetVal == FALSE)
    MessageBox(hWnd,L"initial fail!!!",L"FlyJacket",MB_OK);
else
    SetTVStandard((unsigned long *)&TVMODE_NTSC);
...
...
// set capture format = 640*480 & return data is RGB
CaptureFormat(640,480,2);
Sleep(100);
Buffer = (unsigned char *)malloc(640 * 480 * 3);
//get the capture data
Capture_A_Frame((unsigned char *) Buffer ,&dRetBytes);
...
//Show the capture image
  
```

```

//Save to file
...
...
...
free(Buffer);
Deinitial_FLY_AV();
...
...

```

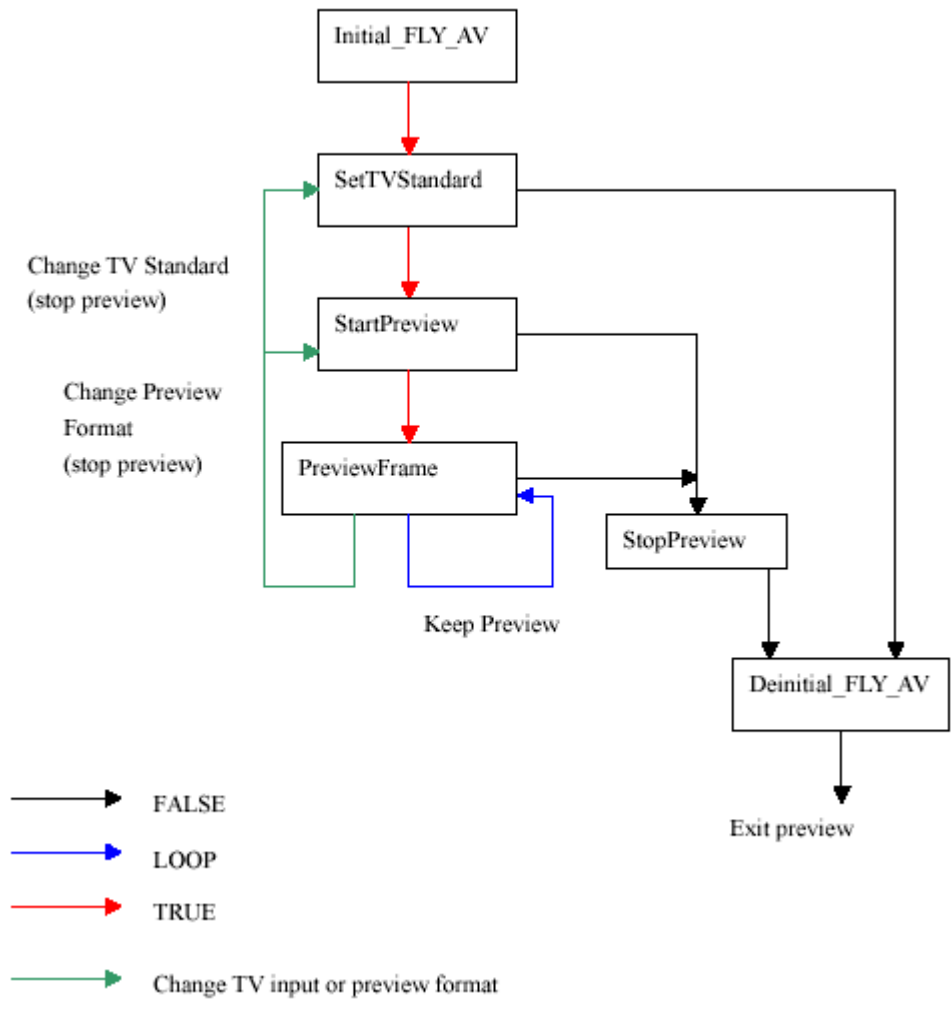


Figura 8 Preview API control flow

**Sample Code**

```

...
<the same as capture code>
...
// set preivev format = 320 * 240
StartPreview(320,240,2,2); // Normal mode, return RGB data
Sleep(1000); // must add delay time
While(1) //
{
...
...
PreviewFrame((unsigned char *) Buffer);
...

```

```

...
}
StopPreview();
...
<the same as capture code>

```

Per visualizzare il flusso video, viene attivato un timer, con un suo quanto temporale e allo scadere viene fatta una PreviewFrame, il codice risultante ha la seguente struttura:

```

//Inizializzazione FlyJacket
iRetVal = Initial_FLY_AV();
if(iRetVal == FALSE)
{
    //!!!!!!bisogna far apparire la nostra messagebox
    CErrore Err;
    CDialog::OnCancel();
    Err.DoModal();
    return TRUE;
}
//Imposto lo standard TV
iRetVal = SetTVStandard((unsigned long *)&g_TVSignal);
if (iRetVal == FALSE)
{
    //!!!!!!bisogna far apparire la nostra messagebox
    CErrore Err;
    CDialog::OnCancel();
    Err.DoModal();
    return TRUE;
}
g_InitFlag=TRUE;
//alloco il buffer e inizio lo start preview
ReAllocateBuffer();
iRetVal = tartPreview(g_prexsize,g_preysize,FormatoPre,InputDataType);
if (iRetVal == FALSE)
{
    //bisogna far apparire la nosta messagebox
    KillTimer(ID_TIMER1);
    StopPreview();
    g_bTimeFlag = FALSE;
    CErrore Err;
    CDialog::OnCancel();
    Err.DoModal();
    return TRUE;
}
g_TVMode=FALSE;
Sleep(100);
//imposto un timer, ad ogni intervallo mostro un'immagine
SetTimer(ID_TIMER1, TIMER_TICK, TimerProc);
g_bTimeFlag=TRUE;

void CALLBACK TimerProc(HWND hwnd,UINT uMsg,UINT idEvent,DWORD dwTime)
{
    ReAllocateBuffer();
    iRetVal = PreviewFrame((unsigned char *)g_pRGBOut->rgb);
    if (iRetVal == FALSE)
    {
        //bisogna far apparire la nosta messagebox

```

```

        return;
    }
    ShowRGB(hWnd, g_pRGBOut);
}

```

Per la cattura si ha il seguente codice:

```

BOOL iRetVal;
DWORD dRetBytes;
s_RGB *pRGBOut;
Sleep(100);
//Catturo l'immagine
SetFileHeader(g_xsize, g_ysize);
iRetVal = CaptureFormat(g_xsize, g_ysize, FormatoImmagine);
if (iRetVal == FALSE)
{
    //bisogna far apparire la nostra messagebox
    CErrore Err;
    CDialog::OnCancel();
    Err.DoModal();
    return TRUE;
}
Sleep(100);
pRGBOut = (s_RGB *) AllocFileRGB(g_xsize, g_ysize);
iRetVal = Capture_A_Frame((unsigned char *)pRGBOut->rgb, &dRetBytes);
if (iRetVal == FALSE)
{
    //bisogna far apparire la nostra messagebox
    CErrore Err;
    CDialog::OnCancel();
    Err.DoModal();
    return TRUE;
}
//salvo in un file temporaneo
swprintf(g_sFileName, L"\\Cattura\\Temp.bmp");
SaveaBMP(pRGBOut->rgb, g_sFileName, g_xsize, g_ysize);
//Faccio lo stoppreview
iRetVal = StopPreview();
if (iRetVal == FALSE)
{
    //bisogna far apparire la nostra messagebox
    CErrore Err;
    CDialog::OnCancel();
    Err.DoModal();
    return TRUE;
}
//Faccio la Deinitial_FLY_AV
Deinitial_FLY_AV();
InvalidateRect(NULL, TRUE);
FreeFileRGB(pRGBOut);

```

## 6 Compressione Jpeg

Al fine di ottenere un'immagine di dimensioni piuttosto contenute, viste le limitate capacità di memoria di tali dispositivi, è stato applicato un algoritmo di compressione all'immagine catturata dalla telecamera. L'immagine viene catturata in formato bitmap, di elevata qualità (o comunque pari alla qualità della sorgente video) ma di anche elevate dimensioni; si è pensato allora all'algoritmo di compressione jpeg, in grado di ridurre sensibilmente la dimensione dell'immagine con una perdita di informazione più che accettabile per i nostri scopi.

## 7 Guida in linea

E' stata inoltre realizzata una piccola guida in linea che illustra a grandi linee il funzionamento del dispositivo; essa è attivabile mediante la pressione del tasto "*Help*" dalla schermata principale (fig. 1), ed è composta da una schermata principale che illustra il funzionamento generale del dispositivo e dei vari menù, e da una schermata per ognuno dei tre menù principali che ne descrivono il funzionamento in maniera più dettagliata

## 8 Setup iPAQ

In questa sezione vengono descritti i passi preliminari per il corretto funzionamento del software di cattura immagini per iPAQ ovvero l'installazione del driver per FlyJacket, Tastiera Hardware, JSlandscape, la registrazione del controllo OCX, creazione della cartella di default.

**Prima di procedere alla installazione dei pacchetti e' necessario che sia presente il servizio di sincronizzazione ActiveSync tra iPAQ e PC tramite porta USB descritto sul manuale d'uso ed installazione dell'iPAQ.**

1. **Installazione driver FlyJacket:** per l'installazione e' necessario inserire l'iPAQ nell'apposito modulo di espansione, eseguire il programma di setup fornito a corredo e seguire le indicazioni via via visualizzate.
2. **Tastiera Hardware:** per l'installazione e' necessario eseguire il programma di setup fornito a corredo della tastiera.
3. **JSlandscape:** questo software ci permette di visualizzare il visore dell'iPAQ in landscape, ovvero ruotato di 90 gradi in senso antiorario, al fine di poter avere una maggiore area disponibile per la preview delle immagini. L'installazione si esegue facendo doppio click sul pacchetto JSLS311Setup-PPC2002.exe (fornito insieme al cd). Ad installazione avvenuta, avviare il software dal palmare ed inserire il codice seriale fornito per poter accedere a tutte le sue funzionalita'. A questo punto e' possibile personalizzare la modalita' di visualizzazione cliccando sulla quarta possibile soluzione, (ovvero con visore in orizzontale e tasti sulla destra) lasciando inalterata la risoluzione (320x240). Con "OK" verranno salvate le modifiche ed il palmare si riavviera' con la nuova visualizzazione. A questo punto si notera' in primo piano un controllo, il JSControl Box, che permette di ruotare a "caldo" il visore toccando col pennino le frecce e che (non rivestendo particolare importanza) e' possibile nascondere associando ad esso un tasto hardware. La procedura e' la seguente: cliccando su Programmi (lo si trova dal menu' start dell'iPAQ) cliccare sull'icona iPAQ MicroKeyboard, cliccare sul tab "Tasti di accesso rapido", evidenziare il tasto F1 e scegliere, sulla finestra in basso, il controllo JS ControlBox, cliccare OK in alto a destra. A questo punto sara' possibile nascondere il controllo premendo il primo tasto grigio a sinistra della tastierina.

4. **Registrazione OCX S309PictureBox:** per l'installazione fare doppio click su S309PictureBox\_PocketPC.exe e seguire le indicazioni. Ad installazione completata e' necessario registrare il controllo copiando il file regsvrce.exe sull'iPAQ (incluso nel cd) tramite l'esplora risorse di ActiveSync, a copia avvenuta lanciare il programma (lo si trova su My Documents) ed inserire il nome del controllo: /windows/S309PictureBox.ocx .
  
5. **Creazione cartella di default:** creare una cartella sulla root directory dell'iPAQ di nome "Cattura", questa sara' la cartella dove verranno salvati i files temporanei e le sottocartelle contenenti le immagini acquisite dal programma.

## 9 Conclusioni

Lo sviluppo di tutto il sistema e' stato caratterizzato da numerose difficolta' per la gestione delle periferiche esterne al palmare, dal mascheramento dei messaggi del sistema operativo, dalla gestione della porta infrarossi.

Nonostante la documentazione presente online sul sito Microsoft, non e' stato possibile accedere alle funzionalita' di base offerte dal palmare, ad esempio, l'invio delle immagini acquisite tramite il tool di sincronizzazione ActiveSync nativo sull'iPAQ, questo proprio perche' l'accesso alla connessione ad infrarossi richiedeva una operazione manuale da effettuare sul display (non essendo possibile configurarla di default), inoltre non avrebbe garantito un invio in successione di tutte le immagini catturate causando cosi' un trasferimento lento e macchinoso.

Altra difficolta' e' stata la gestione dei tasti cursore sulla tastiera esterna in quanto, essendo dei tasti di sistema, vengono gestiti direttamente dal sistema operativo e non dal processo in esecuzione, quindi si e' dovuto "mascherare" la funzionalita' originale del tasto, attraverso degli scambi di messaggi col sistema operativo, per associare ad essa una funzione gestita dal programma di cattura. Essendo noto come lo sviluppo su sistemi portatili sia in forte espansione, si presume che il pieno accesso alle funzionalita' native venga reso disponibile direttamente dall'ambiente di sviluppo, in modo tale da essere piu' accessibile in caso di eventuali modifiche successive da parte di sviluppatori esterni al progetto senza la necessita' di dover conoscere in maniera profonda e precisa il sistema.

## 10 Bibliografia.

Pacchetto di supporto all'iPAQ JSLandscape: disponibile all'indirizzo

<http://www.jimmysoftware.com/Software/Landscape3/default.asp>

Sito Microsoft per i tools embedded per pocket PC

<http://msdn.microsoft.com/vstudio/device/embedded/datasheet.asp>