



Zugriff auf Echtzeit-Digitaldaten mit IRBGRAB.DLL

Dokumentation

Stand: Oktober 2004

1 Grundlegendes

Die von InfraTec vertriebenen Kameras können mit Hilfe digitaler Framegrabber oder der 1394-Schnittstelle Bilddaten in Echtzeit an einen Hostrechner übermitteln. Um diese Funktionalität auch für Kundenprojekte zur Verfügung zu stellen, ist eine DLL-Schnittstelle geschaffen worden, die diese Funktionalität bereitstellt.

Diese DLL trägt den Namen IRBGRAB.DLL und ermöglicht das Initialisieren des Framegrabbers, das Starten der Echtzeitbilderfassung und die Übernahme von Bilddaten.

Für die VarioCAM wird außerdem eine Funktion zur Kommandoübertragung über Firewire angeboten.

2 Auflistung der DLL-Funktionen

Die DLL stellt den lesenden Zugriff auf folgende Eigenschaften eines Thermobildes innerhalb einer IRB-Datei zur Verfügung:

- Höhe und Breite des Thermografiebildes
- Emissivität, Absorption, Objektentfernung, Umgebungs- und Pfadtemperatur der Thermografieaufnahme
- Temperaturen an (x,y)-Koordinaten als Schwarzstrahler- und korrigierte Werte

Die DLL ist mit Borland Delphi erstellt, die Aufrufkonvention DLL ist stdcall. Die DLL kann gleichzeitig 10 IRB-Dateien verwalten.

2.1 function initgrabber(aHandle : THandle; fn, section : PChar): THandle

Zweck: initialisiert den Framegrabber mit Hilfe der Angaben in der Konfigurationsdatei. Diese ist kameraspezifisch und Bestandteil des Lieferumfanges

Parameter: aHandle (THandle) : der das Fenster-Handle des Hauptfensters der übergeordneten Anwendung (In Delphi: Application.Handle) muss übergeben werden.
fn (PChar) : Pfadname der Konfigurationsdatei
sect (PChar) : Bezeichnung der genutzten Sektion der Konfigurationsdatei

Rückgabe: THandle : 1 .. 10 bei erfolgreichem Initialisieren des Grabbers
0 bei Fehler

2.2 function closegrabber(aHandle : THandle): WORDBOOL

Zweck: schließt den Grabber und beendet die Verbindung zur Kamera

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz

Rückgabe: True bei Erfolg
False bei Misserfolg

2.3 procedure releasgrabber(aHandle : THandle)

Zweck: gibt zugehörige Verwaltungsstruktur in DLL wieder frei, Aufruf nach closegrabber (aHandle:THandle)

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz

Rückgabe: keine

2.4 function startgrabber(aHandle: Integer) : WORDBOOL

Zweck: startet die fortlaufende Datenakquisition in einem Ringpuffer in der Kamera
Parameter: aHandle (THandle) : mit initGrabber erzeugte Referenz
Rückgabe: True bei Erfolg
False bei Misserfolg

2.5 function stopGrabber(aHandle : THandle) : WORDBOOL

Zweck: stoppt die fortlaufende Datenakquisition
Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
Rückgabe: True bei Erfolg
False bei Misserfolg

2.6 function getimgwidth(aHandle : THandle) : Integer

Zweck: gibt die Breite des von der Kamera gelieferten Bildes zurück
Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
Rückgabe: 1..320 (abhängig von Sensortyp und eingestellten Parametern)

2.7 function getimgheight(aHandle : THandle) : Integer

Zweck: gibt die Höhe des von der Kamera gelieferten Bildes zurück
Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
Rückgabe: 1..240 (abhängig von Sensortyp und eingestellten Parametern)

2.8 function grabvalues(aHandle: THandle; pbuf : Pointer; rbufpos : Integer) : Integer

Zweck: überträgt in den durch pbuf bezeichneten Speicherbereich die Temperatur für jeden Bildpunkt des im Ringpuffer angewählten Bildes in Kelvin
Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
pbuf (Pointer) : Zeiger auf allokierten Speicherbereich der Größe
`getimgwidth(aHandle) * getimgheight(aHandle) * SizeOf(Single)`
(Single ist vergleichbar mit dem C-Datentyp float)
bei nil wird die Größe des benötigten Speichers in byte zurückgegeben, aber kein Datenfeld erzeugt.
rbufpos (Integer): Nummer der abzurufenden Ringpufferposition
0.. 7: Position 0 bis 7 des Ringpuffers
-2: letzte gefüllte Position
Rückgabe: Feldgröße in Byte bei Erfolg
0 bei Misserfolg
1. kein neues Bild im Ringpuffer
2. wenn der Handle nicht gültig ist
3. der Pointer auf den Puffer den Wert nil hat
4. die Kamera nicht antwortet und es zu einem Timeout kommt

2.9 function grabpicture(aHandle: THandle; pbuf : Pointer; rbufpos : Integer) : Integer

Zweck: überträgt ein Rohdatenbild aus dem Ringpuffer in den durch pbuf bezeichneten Speicherbereich

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
 pbuf (Pointer) : Zeiger auf allokierten Speicherbereich der Größe
`getimgwidth(aHandle) * getimgheight(aHandle) * SizeOf(Word)`
 bei nil wird die Größe des benötigten Speichers in byte zurückgegeben, aber kein Datenfeld erzeugt.
 rbufpos (Integer): Nummer der abzurufenden Ringpufferposition
 0.. 7: Position 0 bis 7 des Ringpuffers
 -2: letzte gefüllte Position

Rückgabe: Feldgröße in Byte bei Erfolg
 0 bei Misserfolg (kein neues Bild im Ringpuffer)

2.10 function grabframe(aHandle: THandle; pbuf : Pointer; rbufpos : Integer; var newhdr : WORDBOOL) : Integer

Zweck: überträgt ein Rohdatenbild inklusive Datenheader aus dem Ringpuffer in den durch pbuf bezeichneten Speicherbereich

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
 pbuf (Pointer) : Zeiger auf allokierten Speicherbereich der Größe
`getimgwidth(aHandle) * getimgheight(aHandle) * SizeOf(Word) + 3000`
 bei nil wird die Größe des benötigten Speichers in byte zurückgegeben, aber kein Datenfeld erzeugt.
 rbufpos (Integer): Nummer der abzurufenden Ringpufferposition
 0.. 7: Position 0 bis 7 des Ringpuffers
 -2: letzte gefüllte Position
 newhdr (WORDBOOL): zeigt an, ob sich Header geändert hat

Rückgabe: Feldgröße in Byte bei Erfolg
 0 bei Misserfolg (kein neues Bild im Ringpuffer)

2.11 function getcalib(aHandle: THandle; pbuf : Pointer; newhdr : WORDBOOL) : Integer

Zweck: überträgt einen Datenheader in den durch pbuf bezeichneten Speicherbereich

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
 pbuf (Pointer) : Zeiger auf allokierten Speicherbereich
 bei nil wird die Größe des benötigten Speichers in byte zurückgegeben, aber kein Datenfeld erzeugt.

Rückgabe: newhdr (WORDBOOL): zeigt an, ob sich Header geändert hat
 Feldgröße in Byte bei Erfolg
 0 bei Misserfolg

2.12 function getlut(aHandle: THandle; pbuf : Pointer) : Integer

Zweck: erzeugt eine Lookup-Tabelle mit 257 Einträgen, aus der durch lineare Interpolation von digitalisierten Pixelwerten in Kelvin-Temperaturen umgerechnet werden kann
lut: array[0..256] of Single

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
pbuf (Pointer) : Zeiger auf allokierten Speicherbereich
bei nil wird die Größe des benötigten Speichers in byte zurückgegeben, aber kein Datenfeld erzeugt.

Rückgabe: Feldgröße in Byte bei Erfolg
0 bei Misserfolg

2.13 function sendcommand(aHandle : THandle; cmd : PChar; answer : PChar; timeout: Integer): WORDBOOL

Zweck: sendet Kommando zur Kamera (nur VC) und gibt Antwort zurück

Parameter: aHandle (THandle) : mit initgrabber erzeugte Referenz
cmd (PChar) : an die Kamera abzusetzendes Kommando
answer (Pchar) : Platzhalter für Antwort der Kamera (max. 100 Byte)
timeout (Integer) : Timeout für Antwort in ms

Rückgabe: in answer