



第9章 環境構築から画像処理テクニックまで

USB 接続のパソコン用カメラで 物体認識と動き検出

横山 大明/関口 正浩
Hiroaki Yokoyama/Masahiro Sekiguchi

画像認識を行いたい場合には、そのための環境を用意する必要があります。ここではUSB接続のパソコン用カメラ(以下USBカメラ)を使ってWindows上で画像処理をするための環境を整える方法と、実際の処理例を紹介します。

この方法の最大の利点は手軽なことです。USBカメラの価格は数千円程度と安価で、演算にはパソコンを使うため、さまざまな処理を比較的簡単に実現できます。最終的には専用ハードウェアを使って画像認識を行う場合でも、初期の段階でのアルゴリズムの検討や、どの程度の性能のカメラが要求されているのかを見極めるためには十分に有効な方法でしょう。

知っておきたいこと

● パソコンへの映像入力デバイス

パソコンに映像を取り込む方法はいくつかあります。例を挙げると、ビデオ出力をPCI接続のキャプチャボードを通して取り込む方法や、IEEE1394経由でDVカメラの映像を取り込む方法などです。

その中でも**USBカメラを選択した理由は、ほかの方法に比べて機器の入手性が良く、安価であること**にあります。USBカメラにはレンズなどの光学系、撮像素子(主にCMOS画像センサ)とその周辺回路、パソコンに接続するためのUSBインターフェースのすべてが備わっています。現在販売されているパソコンのほとんどにはUSB端子が用意されています。したがって、これ一つで手軽に周辺の映像を入力できます。

● 画像取り込みのライブラリ

Windows上で映像を取り込むにはDirectShowを使いました。これはマイクロソフト社⁽¹⁾が開発したDirectXのうち、ビデオやオーディオなどマルチメディア

関係の処理を行うためのライブラリです。

動作OSはDirectShowが動作するWindowsになりますが、開発中にエラーが発生した場合の安定性を考え、**Windows NT系のOSを使うことをお勧めします**。なお、本稿では最新のDirectX8.1bを使っています。今回は簡単に画像処理に挑戦するというのでUSBカメラを例にあげましたが、**DirectShowを使った画像取得の方法はWindows上で扱うことのできるビデオ・キャプチャ・デバイスの多くに適用できます**。

● パソコンを使った画像認識プログラム

映像を取り込んだ後、認識のための演算にはパソコンを使います。得意不得意はありますが、**パソコンのもつ十分な演算速度と潤沢なメモリを使い、幅広い処理を実現できます**。プログラムを制作する段階では試行錯誤の繰り返しになりますが、そのやりやすさではパソコンよりも優れた開発環境はないでしょう。



〈写真1〉Creative Webcam5の外観 [クリエイティブメディア(株)]

Keywords

パソコン用カメラ, USBカメラ, DirectShow, DirectX8.1b, Borland社, Borland C++ Compiler, BCC, Creative Webcam5, M.Fotsch氏, borland_dx8_libs.zip, CCapDS, RGB24ビット・フォーマット, DIB, Device Independent Bitmap, 物体認識, しきい値処理, 動体検出, ラベリング, YUV420.

コンパイラにはBorland社⁽²⁾から無償配布されているBorland C++ Compiler(以下BCC)を利用しました。ここにMicrosoft社から同じく無償配布されているDirectXの開発キットDirectX8.1b SDKを追加して、C/C++言語を使ってプログラムを開発していきます。

● 使用したUSBカメラ

カメラはクリエイティブメディア⁽³⁾から販売されている“Creative Webcam5”という製品(写真1)を使用しました。撮像素子として約30万画素のCMOSイメージ・センサを使っていて、最大640×480ピクセルの解像度(VGA)でパソコンへ映像を入力できます。

フレーム・レートは解像度が640×480ピクセルの場合には15フレーム/秒程度、320×240ピクセルの場合には30フレーム/秒程度です。2002年11月現在は5～6千円程度で入手できます。市場に出回っているUSBカメラは、このクラスのものが一番多いと思います。

そのほかには、撮像素子としてCCDを搭載したものや、最近はさらに安価で約10万画素のCMOSセンサを使ったものも販売されているようです。

開発環境の構築

● 必要なファイルの準備

開発に必要なコンパイラやライブラリを用意します。DirectX SDKはマイクロソフト社のホームページ⁽¹⁾から、BCCはBorland社のホームページ⁽²⁾からダウンロード

〈図1〉 Borland用Directx8インポート・ライブラリの配置場所



〈リスト1〉 BCC用設定ファイル

```
-I"c:\DXSDK\include;c:\Borland\Bcc55\include"
-L"c:\DXSDK\bilib;c:\Borland\Bcc55\lib"
```

(a) bcc32.cfg

```
-L"c:\DXSDK\bilib;c:\Borland\Bcc55\lib"
```

(b) ilink32.cfg

〈表1〉 CCapDSの公開メンバ関数

機能	関数
初期化処理	bool CCapDS::Initialize(HWND hwnd)
終了処理	void CCapDS::CleanUp()
カメラからの入力開始	bool CCapDS::Start()
カメラからの入力停止	bool CCapDS::Stop()
明度などカメラ・パラメータ調整	bool CCapDS::ShowDevProperty()
解像度、映像フォーマット調整	bool CCapDS::ShowPinProperty()
静止画読み込み	EBitmap2 *CCapDS::GetSample()
	bool CCapDS::GetSample(EBitmap2 *eb)
映像フォーマットを得る	const BITMAPINFOHEADER *CCapDS::GetBitmapInfoHeader()
エラー処理	const TCHAR *CCapDS::GetErrMsg()
	const char *CCapDS::GetErrPosMsg()
	const char *CCapDS::GetErrPosMsg()

ンロードできます。

DirectX SDKにはBorland社製のコンパイラで扱うことのできる形式のインポート・ライブラリが含まれていません。M. Fotsch氏がBorland用のインポート・ライブラリをダウンロードできる形(borland_dx8_libs.zip)で公開しているので、使わせていただくことにします⁽⁴⁾。

章末に示したURLから以上3点を用意してください。DirectX SDKとBCCはサイズが大きいため、ブロードバンド環境のない方はダウンロードが困難だと思います。さまざまな雑誌で付録に収録されていることもあるので、探してみるのも手でしょう。DirectX SDKは、Microsoft社のホームページからCD-ROMを申し込むことも可能です。

● DirectX SDKとBCCのインストール

まずDirectX SDKをインストールします。zipファイルを展開したフォルダにあるセットアップ・プログラムの指示にしたがってインストールしてください。途中retailバージョンかdebugバージョンかを選択できますが、本稿で紹介する内容を試す範囲ではretailバージョンで十分です。

次にBCCをインストールします。これもインストーラの指示にしたがってインストールしてください。最後にBorland用DirectX8インポート・ライブラリをインストールします。図1に示す位置にzipファイルを展開してください。デフォルト設定以外でDirectX SDKをインストールした方は適宜読み替えてください。