>> 100行で書く画像処理最先端

>> 環境の構築と簡単なサンプルの作成 |

>> 正会員 山崎俊彦†

キーワード：画像処理，映像処理

1. ま え が き

映像やメディア処理の技術は日進月歩で，毎年「これはすごい！」とつい叫んでしまうような研究成果がさまざまな学会等で発表され続けています。もちろん，論文などで発表された最新のアルゴリズムを完全に再現するのは大変ですが，その核となる技術を調べてみると，意外と基本に忠実で，数学的な知識とちょっとしたアイデアから成っていることがわかります。そこで本シリーズでは，いくつかの代表的なアルゴリズムを題材にして，画像・映像・メディア処理の基礎をご紹介しますとともに，インターネットなどで入手できるサンプルコードをただ利用するのではなく，自分の手で100行程度のプログラミングで実装することによって「なるほど！」を体験していただければと思っています。素材となる論文もご紹介するのでご興味があれば是非読んでいただきたいと思います。

2. 環境の構築

実際のプログラミングを行う前に，プログラミングを行うための環境を整備しないとはいけません。次回以降ご紹介するサンプルプログラムは基本的にMicrosoft Windows XP SP3上でMATLAB 2008bか，もしくはMicrosoft Visual Studio 2008+OpenCV¹⁾を利用したC言語で作成していく予定です（開発時間の関係上，ほとんどがMATLABになると思われます）。OpenCVについては，文献2)をご参照ください。OS依存があるような特殊なライブラリは利用しない予定です。また，原理さえわかっればその他のプログラミング言語でも簡単に構築できるはずです。

2.1 MATLABの入手とインストール

MATLABの製品版のライセンスを有している場合，それをお使いください。ライセンスをお持ちでない学生以外

† 東京大学 大学院 情報理工学系研究科 電子情報学専攻
"Environmental Setup and Samples" by Toshihiko Yamasaki
(Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo, Tokyo)

の方はMathWorks日本のWEBサイト³⁾より評価版を入手することができます(図1)。利用期間などに制限事項については充分ご注意ください。学生の方は評価版を利用することができません。MATLAB & Simulink Student Version (英語版)が米国Mathworks社のホームページから\$99で購入できるようです⁴⁾(図2)。そちらをご利用ください。

図1 MATLABの評価版の入手方法

図2 MATLAB & Simulink Student Version (英語版)の入手方法

図3 cygwinを利用したoctaveのインストール

図4 Microsoft Visual Studio 2008 Express Editionのインストール

MATLABに似たプログラミング環境を目指したOctaveというソフトも公開されています。こちらは、著者はほとんど利用したことがありませんが、選択肢の一つとしてご紹介しておきます。Windowsを利用する場合は後述のcygwinから(図3)、それ以外の場合は公式サイト⁵⁾からパッケージやソースファイルを入手の上セットアップしてください。

2.2 C言語コンパイラとOpenCVの入手とインストール

Windowsを利用する場合、フリーソフトですとMicrosoft Visual Studio 2008 Express Edition⁶⁾やCygwin⁷⁾に含まれるGCCなどで、OpenCVの動作報告がなされているようです。C言語の開発環境をお持ちでない場合、上記のうちのいずれかの使用をおすすめします。また、Mac OSやLinux、FreeBSDなどではGCCが標準で利用できるといいます。

Microsoft Visual Studio 2008 Express Editionに関しては、文献6)よりVisual C++をオンラインでインストールするか(図4)、もしくはisoファイルをダウンロード後、マウントプログラムやDVD作成ソフトを利用してインストールしてください。

Cygwinは、文献7)よりsetup.exeをダウンロードし、任意のサイトを選択してインストールを行ってください。途中、利用するパッケージを選択する画面(Select Packages)が出ますが、Develカテゴリーのgccを選択してください。また、本稿では補助的にperlを利用することもあるので、それもInterpretersカテゴリーから選択してインストールすることをおすすめします。OctaveもMathカテゴリーからインストールできます。まとめると、標準のインストールパッケージに加え、下記の参考のインストールをおすすめします。

- Devel→gcc (クリックすると必要な他の項目も自動選択されます)
- Interpreters→perl
- Math→dctabe

細かい設定方法や使用方法など、そのほかの詳細については紙面の都合上説明できませんので、関連の書籍やインターネットなどの情報をご参照ください。特にOpenCVは使用前にヘッダファイルフォルダの設定やライブラリの設定などが必要です。

3. サンプルプログラムの作成

～遠隔カメラ制御プログラム～

今回は最新の画像処理アルゴリズムではなく、著者が遊びで作成した遠隔カメラ操作プログラムを使いながら、OpenCVの動作確認を行います。USBカメラを使用します。

まずは、プログラム1にあるC言語ソースファイルをテキストエディタ等で入力し、コンパイルしてください。ここではmobile.shutter.exeというプログラム名にします。このプログラムは、USBカメラで静止画を撮影し、指定されたファイル名で保存するだけの簡単なものです。例えば、hoge.jpgという名前で静止画を撮影するにはコマンドプロンプトやcygwin bash shellからフォルダを移動し、

> mobile.shutter.exe hoge.jpg

とするだけです。なお、任意のフォルダで作成したexeファイルを実行させるにはC:\Program Files\OpenCV\binにある必要なdllをexeファイルと同じフォルダにおいてください。無事動作したでしょうか？

これだけでは面白味に欠けるので、携帯電話やパソコンからあるアドレスにメールを送ると、自動的に写真を撮影して返信してくれるというアプリにしたいと思います。概念図を図5に示します。

まずは平文でのPOP3/SMTPが利用可能なメールサービスのアカウントを取得しましょう。ただし、平文での通信はパスワードが盗聴・悪用される可能性があるので充分ご注意ください。ここでは上記の条件を満たすYahoo!メールを利用します。ヘルプの「メールソフトで送受信するにはヘルプ」の内容にしたがって、POPアクセス機能を有効にしてください。ヘルプには記載されていませんが、上記の設定を

図5 遠隔カメラ制御プログラムの動作概念図

した後、ブラウザからどこかへメールを送信し、画面に表示される数字を入力して認証を受けないとPOPによるアクセスが有効にならないようです。コマンドラインでメールを送受信するプログラムは今回の主目的ではないので、自前では作成せず、POP3 Client⁹⁾とSMAIL¹⁰⁾を利用します。それぞれマニュアルをみてサーバの設定を行ってください。一定時間間隔でPOP3 Clientを起動し、メールがきていたらmobile.shutter.exeを起動して写真を撮影した後SMAILで写真を添付して返信するという一連の動作を行うperlプログラムをプログラム2に示します。Cygwin bash shellからすべてのプログラムをおいたディレクトリに移動し、

```
> perl mobile.shutter.pl
```

としてご自分で作成したYahoo!メールのアカウントにメールを送ってみてください。写真が添付された返信メールが送られてきたでしょうか。

このプログラムのメリットはメールを介したシステムなため、WEBサーバの整備など複雑な設定が不要なこと、

さらにはグローバルIPが不要なことです。また、ちょっとしたプログラムの改変で静止画ではなく動画を送信する機能、利用者がメールを送信したときではなく一定間隔でファイルを送る機能、メールの本文に別途コマンドを書いてプログラムに細かな指示を与える機能なども実現できます。通信の安全性を考えると平文ではなくSSLを用いた通信の方が理想的です。SSL通信対応のメール送受信ができるとYahoo!メールだけでなくGMAILなども利用できるので、インターネットで調べるなどしてご自分で拡張してみてください。

なお、本稿で作成したソースコードは、文献11)からもダウンロードできます。各人の責任においてご活用下さい。

(2009年10月5日受付)

〔文 献〕

- 1) <http://opencv.jp/>
- 2) 永田雅人：“私の研究開発ツール-第5回-動画像処理ライブラリーOpenCV”，映情学誌，61，11，pp.1602-1605（2007）
- 3) <http://www.mathworks.co.jp/>
- 4) http://www.mathworks.com/store/productIndexLink.do?s_cid=buynow_sv1
- 5) <http://www.gnu.org/software/octave/>
- 6) <http://www.microsoft.com/japan/msdn/vstudio/Express/>
- 7) <http://www.cygwin.com>
- 8) <http://sourceforge.net/projects/opencvlibrary/>
- 9) <http://www.vector.co.jp/soft/unix/net/se231360.html>
- 10) <http://dip.picolix.jp/disp2.html>
- 11) <http://www.hal.t.u-tokyo.ac.jp/~yamasaki/ite/>

やまざき としひこ
山崎 俊彦 1999年，東京大学工学部電子工学科卒業。2004年，東京大学大学院博士課程修了。現在，東京大学大学院情報理工学系研究科電子情報学専攻准教授。主として，実写3次元映像や3次元CG，デジタル・セキュリティなどを中心とした画像・映像処理に関する研究およびVLSIの研究に従事。博士（工学），正会員。

```
1  /*
2  mobile.shutter.cpp
3  (C) Toshihiko YAMASAKI
4  本プログラムを使用したことによるいかなる障害、損害も責任はとりません
5  各自の責任において使用してください。
6  */
7
8  #include <stdio.h>
9  #include <cv.h>
10 #include <highgui.h>
11
12 int main(int argc, char** argv)
13 {
14     CvCapture* capture = NULL;
15     IplImage* captureImage = NULL;
16
17     if(argc != 2){
18         printf("usage: mobile.shutter.exe [filename]¥n");
19         return -1;
20     }
21
22     if((capture = cvCaptureFromCAM(0))!=NULL)
23     {
24         printf("Cannot find cameras.");
25         return -1;
26     }
27
28     captureImage = cvQueryFrame(capture);
29     cvSaveImage(argv[1] , captureImage);
30     cvReleaseCapture (&capture);
31
32     return 0;
33 }
```

プログラム1 mobile.shutter.cpp

```

1 # Mobile.shutter.pl
2 #(C) Toshihiko YAMASAKI
3 #本プログラムを使用したことによるいかなる障害、損害も責任はとりません
4 #各自の責任において使用してください。
5
6 # 使用時は37行目の[your mail address]をご自分のメールアドレスに置き換えてください。
7
8 sub fromlist(){
9     my @list;
10    open(IN,"message.txt") || die "no messages";
11    while(<IN>){
12        chomp;
13        if($_ =~ /^From:    /){
14            $pat = $_;
15            $pat =~ s/^From:    //;
16            push(@list,$pat);
17        }
18    }
19    close(IN);
20    return @list;
21 }
22
23
24
25 while(1){
26     print "mail check¥n";
27     system("perl ./pop3.pl -f message.txt -d");
28     @list=&fromlist;
29     while(@list){
30         $maddr=pop(@list);
31         ($sec, $min, $hour, $mday, $mon, $year, $wday, $yday, $isdst) = localtime(time);
32         $year=$year+1900;
33         $jpg=sprintf("%04d%02d%02d_%02d%02d%02d%02d.jpg", $year, $mon, $mday, $hour, $min, $sec);
34         print "image sent to $maddr¥n";
35         system("./mobile.shutter.exe $jpg");
36         sleep(1);
37         system("./smail -t -hsmtp.mail.yahoo.co.jp -f¥"[your mail address]¥@yahoo.co.jp¥" -s¥"pic from mobile_shutter¥" -
a$jpg ¥"$maddr¥");
38     }
39     print "sleeping...¥n";
40     sleep(20);
41 }

```