

PRAKTIKUM 1

OpenCV

Materi

- Instalasi OpenCV
- Kode program sederhana

Tujuan Praktikum

- Mahasiswa dapat melakukan instalasi OpenCV dan menjalankan program sederhana pada OpenCV

A. Penyajian

Computer Vision (CV) adalah salah satu cabang dari ilmu pengolahan citra digital (*digital image processing*) yang memungkinkan komputer memahami citra dunia nyata yang diperoleh. Berdasarkan pemahaman tersebut, program diharapkan dapat mengambil keputusan yang tepat. Beberapa kasus umum yang dapat diselesaikan dengan teknik CV adalah deteksi dan pengenalan wajah, pelacakan objek (*tracking*), navigasi, desain interaksi, dan proses manufaktur. Di Institut Pertanian Bogor, teknik CV juga diterapkan di bidang pertanian, antara lain untuk mengidentifikasi tanaman obat berdasarkan ciri morfologi dan tekstur, identifikasi penyakit pada tanaman pangan, penilaian keparahan penyakit, serta inspeksi kualitas buah dan sayuran. Sebagian besar teknik tersebut diterapkan dengan menggunakan pustaka OpenCV.

OpenCV (*Open Source Computer Vision*) adalah pustaka sumber terbuka yang dikembangkan untuk keperluan *computer vision*. OpenCV dikembangkan oleh Intel Corp dan bersifat *cross-platform*. OpenCV telah mendukung bahasa C, C++, Python, Java, dan Matlab.

OpenCV saat ini terdiri atas beberapa modul utama:

- **core**. Berisi definisi struktur-struktur data.
- **imgproc**. Berisi teknik pemroses citra seperti filter, transformasi, ruang warna, dan histogram.
- **video**. Berisi teknik analisis video seperti estimasi gerak, substraksi latar belakang, dan pelacakan objek.
- **calib3d**. Berisi teknik dasar untuk pengaturan *viewing window*, kalibrasi kamera, dan rekonstruksi objek 3D.
- **features2d**. Berisi teknik untuk ekstraksi fitur dan deskriptor.
- **objdetect**. Berisi teknik untuk deteksi objek dari kelas yang telah didefinisikan sebelumnya (seperti wajah dan mata).
- **highgui**. Berisi *interface* untuk akuisisi video dan gambar serta berisi fitur antarmuka sederhana.

- **gpu.** Berisi algoritme-algoritme yang lebih diefisiensikan dengan menggunakan *graphical processing unit* (GPU).
- dan modul pendukung lainnya.

Selain itu OpenCV juga dilengkapi dengan pustaka *Machine Learning* yang berisi algoritme seperti *naïve bayes*, *k-nearest neighbor*, *support vector machine*, *decision tree*, dan *neural network*.

Instalasi OpenCV

Versi stabil OpenCV pada saat tulisan ini dibuat ialah OpenCV 2.4.9. Pada versi ini, perbaikan terhadap bug yang ditemukan pada versi sebelumnya telah diperbaiki dan dukungan terhadap perangkat *mobile*, terutama Android, ditingkatkan. Versi ini dapat diunduh pada alamat opencv.org/downloads.html.

OpenCV 2.4.9 dapat dipasang dengan menggunakan bantuan CMake dan Codeblocks. Langkah-langkah yang perlu dilakukan ialah:

- Instalasi CMake, CodeBlocks (termasuk MinGW), dan OpenCV. Tulisan ini, mengasumsikan OpenCV diinstal pada direktori C:\opencv.
- Tambahkan [lokasi instalasi CMake]\bin pada System Variable Path komputer. Setelah itu, *restart* komputer Anda.
- Cari dan jalankan cmake-gui.exe yang terdapat pada direktori bin CMake.
- Aturlah parameter 'Where is the source code' ke C:\opencv\sources.
- Aturlah parameter 'Where to build the binaries' ke direktori manapun yang Anda mau, tapi disarankan di C:\opencv\mingw.
- Klik Configure. Jika ini pertama kalinya Anda menjalankan CMake, Anda akan diminta *generator* dari proyeknya, set "CodeBlock - MinGW Makefiles" pada *dropdown*, kemudian klik Finish. Jika Anda salah pilih, klik File - Delete Cache.
- Tunggulah hingga proses konfigurasi selesai.
- Setelah konfigurasi selesai, klik *Generate*. (Catatan: kombinasi centang yang dihasilkan proses konfigurasi sebelumnya jangan diubah).
- Tunggulah hingga proses selesai.
- Setelah selesai, di dalam direktori C:\opencv\mingw akan muncul proyek OpenCV.cbp, klik dua kali untuk membukanya di CodeBlock.
- Pastikan terlebih dahulu kita menggunakan kompiler MinGw di CodeBlock. Pilih Setting - Compiler - Toolchain executable - pada isian compiler's installation directory, masukkan alamat direktori MinGw yang ada dalam CodeBlock komputer kalian. (Kalau saya ada di C:\Program Files (x86)\CodeBlocks\MinGW).
- Ubah settingan Build CodeBlock Anda agar menginstall proyeknya. Pilih menu Build - Select Target - Install.
- Lakukan proses Build proyek. Pilih menu Build - Build, atau Ctrl+F9.
- Tunggu hingga prosesnya selesai, sekitar 30 menit.
- Setelah build selesai, masukkan C:\opencv\mingw\bin ke system variable path. Sekali lagi, restart komputer Anda.

Dengan binary-binary opencv sudah dibuat, kita bisa sekarang bisa menggunakan OpenCV 2.4.9. Kita hanya tinggal menghubungkannya dengan CodeBlock.

- Pilih menu Settings - Compiler

- Pilih tab Linker Settings - klik Add - pergi ke direktori C:\opencv\mingw\lib dan masukkan semua library - klik OK.
- Pilih tab Search directories - pilih tab Compiler - tambahkan direktori C:\opencv\build\include, C:\opencv\build\include\opencv, dan C:\opencv\build\include\opencv2.
- Masih pada tab Search directories, pilih tab Linker, tambahkan direktori C:\opencv\mingw\lib.
- Restart CodeBlock.

Kode Program Sederhana

Untuk menguji apakah sudah bisa atau tidak CodeBlock kalian menggunakan OpenCV. Buatlah *project console* baru dalam bahasa C++, kemudian ketikkanlah Program 1.

```

/****
* Program 1
* Program OpenCV sederhana untuk membaca dan menampilkan citra daun.
****/

//Header file standar OpenCV
#include <iostream>
#include <cv.h>
#include <highgui.h>
#include <cxcv.h>

using namespace std;
using namespace cv;

int main()
{
    Mat image; // membuat citra kosong
    image = imread("daun.jpg"); // membaca file bernama daun.jpg

    // membuat jendela di layar
    namedWindow( "Citra", CV_WINDOW_AUTOSIZE );

    // menampilkan gambar di dalam jendela tersebut
    imshow( "Citra", image );

    // menghentikan gambar hingga pengguna menekan tombol
    waitKey(0);

    return 0;
}

```

Sebelum menjalankan program, pastikan berkas gambar daun.jpg telah Anda salin pada direktori *project*. Setelah itu, *compile* dan jalankan *project* tersebut. Hasil keluaran program dapat dilihat pada Gambar 1.



Gambar 1 Keluaran Program 1 yang berupa citra daun

Penjelasan Kode Program

Bagian pertama program merupakan *header file* yang dibutuhkan oleh program OpenCV standar.

```
#include <iostream>
#include <cv.h>
#include <highgui.h>
#include <cxcv.h>
```

Bagian berikutnya merupakan definisi *namespace* yang akan sering kita gunakan untuk mempersingkat pemanggilan fungsi. Sebagai contoh, `IplImage` harus ditulis sebagai `cv::IplImage` apabila namespace `cv` tidak didefinisikan.

```
using namespace std;
using namespace cv;
```

Pada fungsi main, hal pertama yang dilakukan adalah mendeklarasikan variabel untuk menampung citra. Salah satu kelas yang dapat digunakan adalah `Mat`.

```
Mat image; // membuat citra kosong
```

Setelah itu, citra dimuat dengan menggunakan fungsi `imread()` ke dalam variabel `image` yang telah dideklarasikan sebelumnya. Fungsi ini dapat membaca citra dalam berbagai format seperti BMP, DIB, JPEG, JPE, PNG, PBM, PGM, PPM, SR, RAS, and TIFF.

```
image = imread( "daun.jpg" );
```

Highgui berisi fungsi untuk membuat jendela, yaitu `cvNamedWindow`. `CV_WINDOW_AUTOSIZE`: Properti dari window yang akan mengatur ukuran window sama dengan ukuran citra yang akan ditampilkan.

```
// membuat jendela di layar
namedWindow( "Citra", CV_WINDOW_AUTOSIZE );
```

Hal yang ingin ditampilkan dalam jendela dapat ditampilkan dengan menggunakan fungsi `cvShowImage()`. Parameter pada fungsi ini adalah nama jendela yang akan diisi beserta variabel tempat citra yang akan ditampilkan disimpan.

```
// menampilkan gambar di dalam jendela tersebut  
imshow ( "Citra", image );
```

Fungsi ini akan meminta program untuk berhenti dan menunggu *keystroke*. Jika argumen yang diberikan positif, program akan menunggu selama milidetik argumen. Jika argumen diatur ke 0 atau ke angka negatif, program akan menunggu sampai suatu tombol ditekan.

```
// menghentikan gambar hingga pengguna menekan tombol  
cvWaitKey(0);
```

Nama :
NRP :
Nama Dosen :
Nama Asisten :

B. Lembar Kerja Praktikum

1. Buatlah program sederhana untuk membuat matrix gambar
2. Buatlah program pencari nilai pixel (<150) dan mengganti dengan nilai 255.
3. Buatlah program untuk membalik matrix (transpose)