

Lampiran: Instalasi Microsoft DirectX SDK, Intel OpenCV dan Intel IPL

1. Dahului dengan instalasi Microsoft Visual C++ 6.0 (MSVC 6.0).
2. Install DirectX SDK 8.1. Jawab “Yes” ketika ditanya apakah ingin menambahkan path directory pada MSVC. DirectX SDK ini dibutuhkan untuk mengakses informasi multimedia (video,audio, dll).
3. Install Intel IPL 2.5 (**ipl25.exe**) untuk dapat menjalankan fungsi-fungsi yang disertakan pada tools image processing ini. Hasil instalasi tools ini menghasilkan beberapa library yang dapat digunakan pada beberapa tools program developer (Borland Delphi, Borland C++, Visual Basic, Visual C++). Library ini sangat dibutuhkan pada banyak aplikasi Intel OpenCV.
4. Install Open versi 2.0 terlebih dahulu dengan menjalankan file **opencv_core_b2_0.exe** (merupakan file library dasar), lalu lanjutkan dengan menjalankan file **opencv_apps_b2_0.exe** (untuk instalasi beberapa aplikasi contoh OpenCV)
5. Upgrade OpenCV versi 2.0 menjadi versi 2.1 (terbaru):
 - a. Unzip file **opencv_core_b2.1.zip** dan ekstrak pada folder “c:\Program files\intel\” (tergantung dari instalasi awal).
 - b. Unzip file **opencv_apps_b2.1.zip** dan ekstrak pada folder “c:\Program files\intel\” (tergantung dari instalasi awal).
 - c. Unzip file **opencv_patch_b2.1.zip** dan ekstrak pada folder “c:\Program files\intel\” (tergantung dari instalasi awal).
 - d. Unzip file **opencv_dll_addon_b2.1.zip** dan ekstrak pada folder “c:\Program files\intel\” (tergantung dari instalasi awal).

Overwrite semua file Opencv yang lama (2.0) dengan hasil ekstrak yang baru (2.1).
6. Masuk ke dalam mode “Command Prompt”, pindah ke folder “C:\Program Files\Intel\OpenCV\Bin”. Lalu jalankan file “**RegisterAll.bat**” untuk menginstall filter-filter tambahan ke dalam register windows.
7. Tambahkan path directory pada MSVC 6.0 (klik Tools|Options|Directories) dari beberapa tool yang kita install sebelumnya (DirectX, IPL, OpenCV).
Pada include files:
 - C:\DXSDK\INCLUDE
 - C:\DXSDK\SAMPLES\MULTIMEDIA\COMMON\INCLUDE
 - C:\DXSDK\SAMPLES\MULTIMEDIA\DIRECTSHOW\BASECLASSES

- C:\PROGRAM FILES\INTEL\OPENCV\CV\INCLUDE
- C:\PROGRAM FILES\INTEL\OPENCV\APP'S\COMMON
- C:\PROGRAM FILES\INTEL\OPENCV\CVAUX\INCLUDE
- C:\PROGRAM FILES\INTEL\PLSUITE\INCLUDE

Pada library files:

- C:\DXSDK\LIB
- C:\DXSDK\SAMPLES\MULTIMEDIA\DIRECTSHOW\BASECLASSES\DEBUG
- C:\PROGRAM FILES\INTEL\OPENCV\LIB
- C:\PROGRAM FILES\INTEL\OPENCV\OTHERLIBS\CVLGRFMTS
- C:\PROGRAM FILES\INTEL\OPENCV\OTHERLIBS\HIGHGUI
- C:\PROGRAM FILES\INTEL\PLSUITE\LIB\MSVC

8. Build library tambahan pada:

- C:\PROGRAM FILES\INTEL\OPENCV\OTHERLIBS\HIGHGUI\HIGHGUI.DSW
- C:\PROGRAM FILES\INTEL\OPENCV\OTHERLIBS\CVLGRFMTS\CVLGRFMTS.DSW
- C:\DXSDK\SAMPLES\MULTIMEDIA\DIRECTSHOW\BASECLASSES\BASECLASSES.DSW

Build satu persatu library tambahan diatas dengan **batch build** (klik build|batch build). Tandai option debug & release.

9. Proses instalasi selesai, coba jalankan beberapa aplikasi-aplikasi OpenCV

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS KRISTEN PETRA**

USULAN TUGAS AKHIR

Nama : Lukman Vendy Wirawan
NRP : 26498011
Bidang Studi : Teknologi Perangkat Lunak
Judul Tugas Akhir : Sistem Pengenalan Plat Nomor Kendaraan Roda 4 dengan
Metode Principal Component Analysis (PCA)
Pembimbing I : Ir. Resmana Lim, M.Eng
Pembimbing II : Ir. Kartika Gunadi
Dilaksanakan : Semester genap 2001/2002

Surabaya, 12 Januari 2002

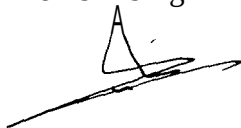
Yang mengusulkan,



Lukman Vendy Wirawan

Mengetahui:

Pembimbing I



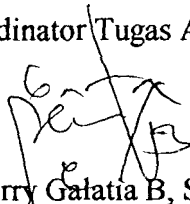
Ir. Resmana Lim, M.Eng

Pembimbing II



Ir. Kartika Gunadi

Koordinator Tugas Akhir



Cherry Galatia B, S.Si

USULAN TUGAS AKHIR

1. Judul Tugas Akhir

Sistem Pengenalan Plat Nomor Kendaraan Roda **4**

2. Latar Belakang Masalah

Perkembangan teknologi *image processing* hingga saat ini terus diperluas dengan tujuan untuk membantu manusia dalam melakukan pekerjaannya. Aplikasinya antara lain dalam dunia ilmu pengetahuan itu sendiri, sekuriti, robotika, dsb.

Pendeteksian dan pengenalan plat nomor mobil secara otomatis telah menjadi suatu aplikasi yang sangat penting dalam bidang *computer vision*. Sistem pendeteksian dan pengenalan ini bekerja apabila terdapat mobil yang melewati titik tertentu akan di-foto secara digital, dan mengidentifikasi secara otomatis dengan cara mencari lokasi dari plat nomor mobil tersebut, memisahkan setiap karakter yang ada dari plat nomor tersebut dan kemudian mengenali setiap karakter tersebut.

Beberapa aplikasi yang dapat menggunakan Sistem Pengenalan Plat Nomor Kendaraan adalah:

- Pelacakan kendaraan yang hilang
- Sistem kontrol dan sekuriti pada area parkir
- Kontrol lalu lintas (mendeteksi pelanggar rambu lalu lintas, parkir ilegal secara otomatis)
- dsb.

3. Perumusan Masalah & Ruang Lingkup

Masalah yang muncul adalah:

- Bagaimana cara mengolah gambar dari kamera untuk dapat diproses secara *real-time*?

- Bagaimana mendeteksi letak plat nomor mobil secara akurat? untuk kemudian dilakukan *image cropping* pada posisi plat nomor kendaraan.
- Bagaimana memisahkan (segmentasi) gambar plat nomor yang telah terdeteksi menjadi beberapa bagian karakter?
- Bagaimana melakukan pengenalan karakter dari hasil segmentasi?

Ruang lingkup dibatasi pada:

- Pengenalan ini hanya untuk plat nomor wilayah Indonesia saja.
- Karakter pada setiap plat nomor sesuai dengan standart yang ada.
- Menggunakan sistem operasi Windows 9.x
- Desain aplikasi dengan Microsoft Visual C++6.

4. Tujuan Tugas Akhir

Tujuan akhir dari penelitian ini adalah untuk membuat *software* yang dapat mendeteksi dan mengenali plat nomor kendaraan roda 4 dimana nantinya dapat diaplikasikan ke suatu sistem yang lebih lanjut.

5. Tinjauan Pustaka

Sistem Pengenalan Plat Nomor Kendaraan ini memiliki 2 level bagian:

1. *Plate Detection dan Segmentation (PlateFinder)*

Gambar 5.1. menunjukkan alur blok diagram *platefinder*. Dimana akan dijelaskan untuk masing-masing bagiannya dibawah ini:

A. *Image Pre-processing*

Langkah pertama adalah melakukan *grayscale* pada image untuk mengurangi informasi warna *image* yang tidak diperlukan. Hasilnya berupa *file image biner* yang memudahkan dalam pengaksesan digitnya. Untuk melakukan *grayscale* ini digunakan metode *The Niblack Binarization algorithm* yang terkenal kecepatannya dalam melakukan *thresholding*.

B. *Digit location*

Entity yang kemungkinan berupa angka atau huruf ditempatkan pada bagian ini. Program akan melakukan iterasi pada semua *pixel image*

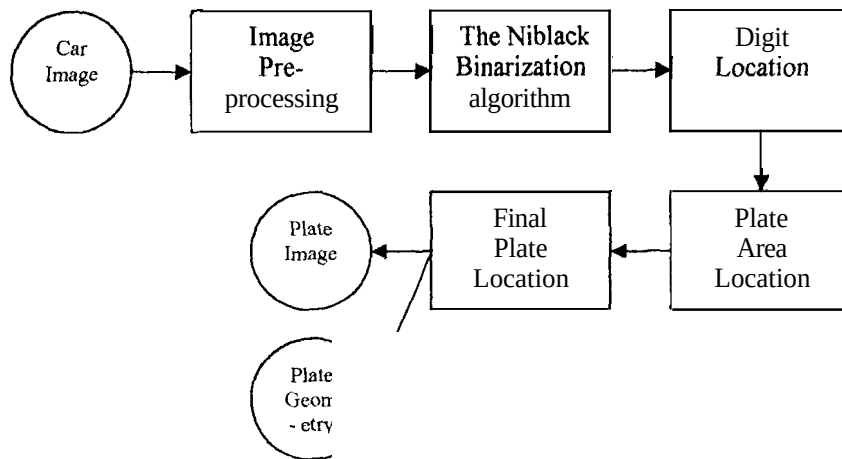
dan mengecek pada setiap iterasi, menggunakan sekumpulan aturan, apakah terdapat kandidat digit pada posisi yang bersangkutan.

C. *Plate area location*

Setelah proses sebelumnya selesai maka dilakukan pencatatan lokasi plat nomor dengan melakukan pembuatan kotak yang bentuknya kurang lebih sama dengan bentuk plat nomor.

D. *Final plate location*

Proses terakhir adalah melakukan *cropping* pada posisi kotak yang bersesuaian untuk memotong image sehingga hasilnya adalah berupa *image* plat nomor saja.



Gambar 5.1
Alur Blok Diagram *PlateFinder*

II. *Optical Character Recognition (OCR)*

Setelah proses deteksi plat selesai, selanjutnya plat tersebut akan diolah menjadi beberapa proses, meliputi:

A. *Thresholding*

Merubah plat nomor yang terdeteksi menjadi *grayscale*. Sebenarnya langkah ini telah dilakukan pada proses sebelumnya, sehingga kita tidak perlu melakukannya lagi dan dapat langsung menggunakannya.

B. *Character Extraction (Segmentation)*

Untuk memisah masing-masing karakter dari plat nomor, dilakukan dengan cara melakukan iterasi pada setiap pixel secara vertikal dan

horizontal. Tujuannya adalah untuk memotong secara tepat setiap karakter, sehingga pengenalannya akan menjadi lebih mudah.

C. Classifier

Pengenalan OCR ini berdasarkan sistem *Nearest Neighbour*. Hasil dari segmentasi (proses sebelumnya) akan dicocokkan dengan database yang telah di-*training*-kan.

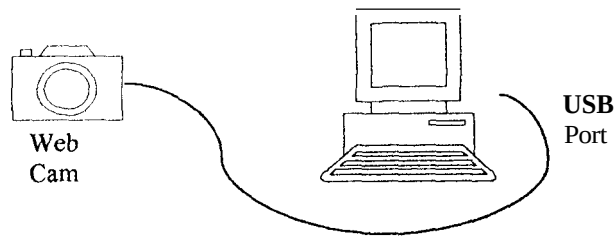
6. Metodologi Penelitian

Rencana penelitian dilakukan sesuai dengan langkah-langkah berikut:

1. Studi Literatur tentang:
 - 1.1 Microsoft Visual C++6.
 - 1.2 Intel Performance Library
 - 1.3 Intel Open Source Computer Vision.
 - 1.4 Microsoft DirectX SDK 8
 - 1.5 Teori dan algoritma untuk pengenalan karakter
 - 1.6 Teori dan algoritma untuk memproses suatu gambar (*image filtering*)
2. Pengumpulan Data
 - 2.1 Database gambar untuk masing masing karakter (*training image*)
 - 2.2 Penormalan image (*resizing, gray scaling, dsb.*)
3. Perencanaan dan Pembuatan Perangkat Lunak
 - 3.1 Modul untuk pengambilan *image* (gambar 6.1).
 - 3.2 Modul untuk mendeteksi letak plat nomor.
 - 3.3 Modul untuk melakukan *image filtering*.
 - 3.4 Modul untuk melakukan segmentasi.
 - 3.5 Modul untuk mendeteksi karakter yang telah disegmentasi.
 - 3.6 Penggabungan modul-modul yang ada menjadi satu sistem pengenalan plat nomor mobil.
4. Pengujian dan Analisa Data
 - 4.1 Pengujian perangkat lunak yang telah dibuat.
 - 4.2 Pengujian hasil dari program pengenalan tersebut, meliputi:
 - Variasi jenis kendaraan
 - Variasi iluminasi (tingkat cahaya yang ada)

- Variasi noise

4.3 Analisa hasil dari program.



Gambar 6.1
Proses Pengambilan *Image*

5. Pengambilan Kesimpulan

Pengambilan kesimpulan dilakukan dengan mengevaluasi hasil dari pengujian hingga didapat hasil yang terbaik.

7. Relevansi

Manfaat dari *software* Sistem Pengenalan Plat Nomor Kendaraan Roda 4 ini adalah untuk dapat mengenali plat nomor dari mobil yang diidentifikasi secara otomatis sehingga dapat mengefisiensikan dan membantu manusia dalam melakukan pekerjaannya.

8. Jadwal Kegiatan

Kegiatan	Bulan	I	II	III	IV	V	VI
Studi literatur		X					
Pengumpulan data		X					
Perencanaan sistem			X	X			
Pembuatan program			X	X	X	X	
Pengujian, implementasi, dan perbaikan				X	X	X	X
Penyerahan aplikasi							X

9. Daftar Pustaka

Charl Coetzee, Charl Botha, dan David Weber, -- PC Based Number Plate Recognition System". 1998.PDF

D, Ballard dan C, Brown, "*Computer Vision*". Prentice Hall, 1982.

Forsyth, David dan Ponce, Jean, "*Computer Vision -- A Modern Approach*", PDF

S. Draghici, "*A neural network based artificial vision system for lisencc plate recognition*", PDF