

DASAR DASAR OPENCV

DASAR DASAR OPENCV

Cerdas Mengoprasikan Gambar

Faisal Najib Abdullah
Informatics Research Center



Kreatif Industri Nusantara

*‘Jika Kamu tidak dapat
menahan lelahnya
belajar, Maka kamu harus
sanggup menahan
perihnya Kebodohan.’
Imam Syafi’i*

CONTRIBUTORS

FAISAL NAJIB ABDULLAH, Informatics Research Center., Politeknik Pos Indonesia,
Bandung, Indonesia

CONTENTS IN BRIEF

1	Pengenalan OpenCV
----------	--------------------------

1

DAFTAR ISI

Daftar Gambar	xi
Daftar Tabel	xiii
Foreword	xvii
Kata Pengantar	xix
Acknowledgments	xxi
Acronyms	xxiii
List of Symbols	xxv
Introduction	xxvii
<i>Faisal Najib Abdullah</i>	
1 Pengenalan OpenCV	1
1.1 OpenCV	1
1.1.1 Definisi OpenCV	1
1.1.2 Sejarah Kecerdasan Opencv	3
1.2 Instalasi OpenCV	4
1.2.1 Instalasi pada Windows	4
1.2.2 Instalasi pada OS X	10
	ix

1.2.3	Instalasi pada Ubuntu	18
1.2.4	Instalasi modul Contrib	21

DAFTAR GAMBAR

DAFTAR TABEL

Listings

FOREWORD

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa

KATA PENGANTAR

Buku ini diciptakan bagi pengembang yang ingin mengembangkan kemampuan memprogram di bahasa python menggunakan library opencv.

FAISAL NAJIB ABDULLAH

*Bandung, Jawa Barat
Januari, 2020*

ACKNOWLEDGMENTS

Terima kasih atas masukan dari pembimbing agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Belajar.

R. M. A.

ACRONYMS

AI	Artificial Intelligence
ETL	Extract Transform Load
NLP	Natural Language Processing

SYMBOLS

- A Amplitude
- $\&$ Propositional logic symbol
- a Filter Coefficient

- \mathcal{B} Number of Beats

INTRODUCTION

FAISAL NAJIB ABDULLAH

Informatics Research Center
Bandung, Jawa Barat, Indonesia

Pada era disruptif saat ini. Pemrograman merupakan sebuah kebutuhan dalam sebuah organisasi pengembangan perangkat lunak. Buku ini diharapkan bisa menjadi penghantar para programmer, analis, IT Operation dan Project Manajer. Dalam melakukan implementasi program pada diri dan organisasinya.

Rumusnya cuman sebagai contoh aja biar keren.

$$ABCDEF\alpha\beta\Gamma\Delta\sum_{def}^{abc} \tag{I.1}$$

BAB 1

PENGENALAN OPENCV

1.1 OpenCV

1.1.1 Definisi OpenCV

Computer Vision adalah ilmu pemrograman komputer untuk memproses dan pada akhirnya memahami gambar dan video, atau hanya mengatakan membuat komputer melihatnya. Memecahkan sebagian kecil dari tantangan Visi Komputer tertentu, menciptakan kemungkinan baru yang menarik dalam teknologi, teknik, dan bahkan hiburan. Untuk memajukan penelitian visi dan menyebarkan pengetahuan visi, sangat penting untuk memiliki perpustakaan fungsi pemrograman dengan kode yang dioptimalkan dan portabel, dan mudah-mudahan tersedia secara gratis. Ini adalah tujuan asli tim Intel pada tahun 1999 ketika OpenCV (Open Source Computer Vision Library) secara resmi diluncurkan. Sejak itu, sejumlah programmer telah berkontribusi pada perkembangan perpustakaan terbaru. Perubahan besar terbaru terjadi pada tahun 2009 (OpenCV 2) yang mencakup perubahan utama pada antarmuka C++. Rilis perpustakaan terbaru dapat ditemukan di situs web resmi OpenCV. Saat ini perpustakaan memiliki 2500 algoritma yang dioptimalkan. Ini digunakan secara luas di seluruh dunia, memiliki 2,5 juta unduhan dan 40 ribu orang di grup peng-

guna. OpenCV dapat digunakan dalam aplikasi akademik dan komersial juga, di bawah lisensi BSD. Untuk menguasai setiap elemen pustaka OpenCV perlu berkonsultasi dengan banyak buku yang tersedia tentang topik OpenCV. Namun demikian, membaca materi yang lebih komprehensif seperti itu harus lebih mudah setelah memahami ide dasar tentang OpenCV dari makalah ini. Bahkan untuk membuatnya lebih nyaman, teks yang disajikan di sini dengan cermat mengikuti salah satu sumber OpenCV terbaru.

OpenCV adalah pustaka Pengolah Gambar yang dibuat oleh Intel, dapat diunduh secara bebas dan tersedia untuk C, C++, Java dan Python, Windows, Linux, Mac OS, iOS dan Android Terbaru Versi-versinya adalah opencv2.4.13 dan opencv-3.1. Ini adalah Sumber terbuka dan juga Mudah digunakan dan diinstal. Ini dirancang untuk efisiensi komputasi fokus yang kuat pada aplikasi waktu nyata Implementasi pertama adalah dalam bahasa pemrograman C; Namun, popularitasnya tumbuh dengan implementasi C++ pada Versi 2.0. Fungsi baru diprogram dengan C++. Namun, saat ini, perpustakaan memiliki antarmuka penuh untuk bahasa pemrograman lain, seperti Java, Python, dan MATLAB / Octave. OpenCV tersedia secara bebas untuk diunduh di <http://opencv.org>. Situs ini menyediakan versi terakhir untuk distribusi (saat ini, 3.0 beta) dan versi yang lebih lama. OpenCV mengharuskan gambar dalam BGR atau Grayscale untuk ditampilkan atau disimpan. Kalau tidak, efek yang tidak diinginkan dapat terjadi.

Pustaka OpenCV (sejak versi 2.2) dibagi menjadi beberapa modul, di mana setiap modul dapat dipahami, secara umum, sebagai didedikasikan untuk satu kelompok masalah penglihatan komputer. Semua kelas dan fungsi didefinisikan dalam ruang nama cv. Oleh karena itu untuk mengaksesnya kita dapat mendahului definisi fungsi utama dengan deklarasi menggunakan namespace cv; atau awali nama kelas dan fungsi OpenCV berdasarkan spesifikasi namespace cv. Objek utama adalah Mat kelas. Seperti dilibatkan oleh nama kelas itu pada dasarnya adalah sebuah matriks yang memegang nilai-nilai piksel dari beberapa gambar dan, di samping itu, sejumlah atribut tentang suatu gambar. Dalam kasus yang paling sederhana, sebuah gambar dapat dibuat sebagai `cv::Mat image`; , membuat gambar dengan ukuran 0 x 0. Mungkin variabel anggota yang paling penting dari objek gambar adalah data di mana anggota `image.data` sebenarnya adalah penunjuk ke memori yang dialokasikan blok yang berisi data gambar (dalam kasus sepele ini adalah `image.data = 0`). Atau selama pembuatan objek Mat kita bisa secara eksplisit menentukan ukuran awal dan jenis setiap elemen matriks. Jenis ini menentukan, misalnya, menandatangani nilai gambar piksel 1-byte, atau tiga saluran untuk gambar berwarna, atau bahkan angka titik apung 32-bit atau 64bit.

Setelah objek matematika kelas didefinisikan, fitur yang bagus tentangnya (tidak ada dalam versi awal OpenCV) adalah bahwa alokasi / dealokasi memori dilakukan secara otomatis. Misalnya, memori yang secara otomatis dialokasikan selama gambar dibacakan ke beberapa objek, juga akan secara otomatis dirilis setelah objek yang sesuai keluar dari ruang lingkup. Hal penting lainnya adalah bahwa kelas Mat mengimplementasikan penghitungan referensi dan salinan dangkal. Oleh karena itu, ketika gambar ditugaskan ke yang lain, data gambar itu sendiri tidak disalin dan kedua gambar menunjuk ke blok memori yang sama (ini juga berlaku untuk gambar

yang dilewati / dikembalikan oleh nilai). Namun, karena jumlah referensi didukung, memori yang dialokasikan untuk data gambar (piksel) itu sendiri akan dirilis hanya ketika semua referensi ke gambar dihancurkan.

Dalam versi sebelum OpenCV 2, fungsi dan struktur seperti C digunakan (masih bisa jadi) dan struktur utama adalah IplImage. Meskipun ada cara mudah untuk mengubah struktur IplImage menjadi objek cv :: Mat, sangat disarankan untuk menghindari struktur data yang sudah usang ini.

1.1.2 Sejarah Kecerdasan Opencv

Resmi diluncurkan pada tahun 1999, proyek OpenCV pada awalnya merupakan inisiatif Intel Research untuk memajukan aplikasi intensif CPU, bagian dari serangkaian proyek termasuk penelusuran sinar waktu nyata dan dinding layar 3D. Kontributor utama untuk proyek ini termasuk sejumlah pakar optimisasi di Intel Rusia, serta Tim Perpustakaan Kinerja Intel. Pada hari-hari awal OpenCV, tujuan proyek digambarkan sebagai:

1. Memajukan penelitian visi dengan menyediakan tidak hanya kode terbuka tetapi juga dioptimalkan untuk infrastruktur visi dasar. Tidak ada lagi menciptakan kembali roda.
2. Menyebarkan pengetahuan visi dengan menyediakan infrastruktur umum yang dapat dibangun oleh pengembang, sehingga kode akan lebih mudah dibaca dan dapat ditransfer.
3. Aplikasi komersial berbasis visi mutakhir dengan membuat kode portabel yang dioptimalkan kinerja tersedia secara gratis - dengan lisensi yang tidak memerlukan kode untuk terbuka atau bebas sendiri.

Versi alpha pertama dari OpenCV dirilis ke publik di Konferensi IEEE pada Computer Vision dan Pattern Recognition pada tahun 2000, dan lima beta dirilis antara tahun 2001 dan 2005. Versi 1.0 pertama dirilis pada tahun 2006. Versi 1.1 "pra-rilis" dirilis pada Oktober 2008.

Rilis utama kedua dari OpenCV adalah pada Oktober 2009. OpenCV 2 mencakup perubahan besar pada antarmuka C ++, yang bertujuan untuk lebih mudah, pola yang lebih aman, fungsi baru, dan implementasi yang lebih baik untuk yang sudah ada dalam hal kinerja (terutama pada multi- sistem inti). Rilis resmi sekarang terjadi setiap enam bulan dan pengembangan sekarang dilakukan oleh tim Rusia independen yang didukung oleh perusahaan komersial.

Pada Agustus 2012, dukungan untuk OpenCV diambil alih oleh yayasan nirlaba OpenCV.org, yang mengelola pengembang dan situs pengguna.

Pada Mei 2016, Intel menandatangani perjanjian untuk mengakuisisi Itseez, pengembang OpenCV terkemuka.

1.2 Instalasi OpenCV

1.2.1 Instalasi pada Windows

Windows tidak datang dengan Python yang sudah diinstal. Namun, wizard instalasi tersedia untuk Python, NumPy, SciPy, dan OpenCV yang telah dikompilasi. Atau, kita dapat membangun dari sumber. Sistem build OpenCV menggunakan CMake untuk konfigurasi dan Visual Studio atau MinGW untuk kompilasi.

Jika kami ingin dukungan untuk kamera kedalamannya, termasuk Kinect, pertamanya kita harus menginstal OpenNI dan SensorKinect, yang tersedia sebagai binari yang dikompilasi dengan wizard penginstalan. Kemudian, kita harus membangun OpenCV dari sumber.

Catatan

Versi OpenCV yang dikompilasi sebelumnya tidak menawarkan dukungan untuk kamera yang dalam. Pada Windows, OpenCV 2 menawarkan dukungan yang lebih baik untuk Python 32-bit daripada Python 64-bit. Namun, dengan mayoritas komputer yang dijual hari ini menggunakan sistem 64-bit, instruksi kami akan merujuk pada 64-bit. Semua installer memiliki versi 32-bit yang tersedia dari situs yang sama dengan 64-bit.

1. Beberapa langkah berikut merujuk pada pengeditan variabel PATH sistem. Tugas ini dapat dilakukan di jendela Variabel Lingkungan di Panel Kontrol.
2. Pada Windows Vista / Windows 7 / Windows 8, klik menu Start dan luncurkan Panel Kontrol. Sekarang, navigasikan ke System and Security, Sistem, Pengaturan sistem lanjutan. Klik pada tombol Variabel Lingkungan....
3. Pada Windows XP, klik pada menu Mulai dan navigasikan ke Control Panel — Sistem. Pilih tab Advanced. Klik pada tombol Variabel Lingkungan....
4. Sekarang, di bawah System variable, pilih Path dan klik tombol Edit....
5. Buat perubahan sesuai petunjuk.
6. Untuk menerapkan perubahan, klik pada semua tombol OK (sampai kita kembali ke jendela utama Control Panel).
7. Kemudian, keluar dan masuk kembali (atau reboot).

Menggunakan penginstal biner (tidak ada dukungan untuk kamera kedalaman)

Anda dapat memilih untuk menginstal Python dan pustaka terkait secara terpisah jika Anda mau; Namun, ada distribusi Python yang datang dengan installer yang akan mengatur seluruh tumpukan SciPy (yang mencakup Python dan NumPy), yang membuatnya sangat sepele untuk mengatur lingkungan pengembangan.

Salah satu distribusi tersebut adalah Anaconda Python (dapat diunduh di). Setelah installer diunduh, jalankan dan ingat untuk menambahkan path ke instalasi Anaconda ke variabel PATH Anda mengikuti prosedur sebelumnya.

Berikut adalah langkah-langkah untuk mengatur Python7, NumPy, SciPy, dan OpenCV:

1. Unduh dan instal 64 bit Python 2.7.9 dari

<https://www.python.org/ftp/python/3.7.6/python-3.7.6-amd64>

2. Unduh dan instal NumPy 1.6.2.

3. Unduh dan pasang SciPy 11.0 dari

<http://www.lfd.uci.edu/~gohlke/pythonlibs/#scipy><http://sourceforge.net/projects/scipy/files/scipy/1.10.0/Python%202.7%2064-bit/Python2.7.9-64bit-scipy-1.10.0-64bit.exe/download>

(ini sama dengan NumPy dan ini adalah pemasang komunitas).

4. Unduh ZIP self-extracting dari OpenCV 3.0.0 dari

<https://github.com/Itseez/opencv>

. Jalankan ZIP ini, dan ketika diminta, masukkan folder tujuan.

5. Salin

```
<unzip_destination>\opencv\build\python\2.7\cv2.pyd
```

ke

```
C:\Python2.7\Lib\site-packages
```

(dengan asumsi bahwa kami telah menginstal Python 2.7 ke lokasi default). Jika Anda menginstal Python 2.7 dengan Anaconda, gunakan folder instalasi Anaconda daripada instalasi Python default. Sekarang, instalasi Python baru dapat menemukan OpenCV.

6. Langkah terakhir diperlukan jika kita ingin skrip Python dijalankan menggunakan instalasi Python baru secara default. Edit variabel PATH sistem dan tambahkan C: /Python2.7 (dengan asumsi kami telah menginstal Python 2.7 ke lokasi default) atau folder instalasi Anaconda Anda. Hapus jalur Python sebelumnya, seperti C:/Python2.6. Logout dan login kembali (sebagai alternatif, reboot).

Menggunakan CMake dan kompiler

Windows tidak dilengkapi dengan kompiler atau CMake. Kita perlu menginstalnya. Jika kami ingin dukungan untuk kamera kedalaman, termasuk Kinect, kami juga perlu menginstal OpenNI dan SensorKinect.

Mari kita asumsikan bahwa kita telah menginstal Python 2.7, NumPy, dan SciPy 32-bit baik dari binari (seperti dijelaskan sebelumnya) atau dari sumber. Sekarang, kita dapat melanjutkan dengan menginstal kompiler dan CMake, menginstal opsional OpenNI dan SensorKinect, dan kemudian membangun OpenCV dari sumber:

1. Unduh dan instal CMake 3.1.2 dari

<http://www.cmake.org/files/v3.1/cmake3.1.2-win32-x86.exe>

Saat menjalankan penginstal, pilih Tambahkan CMake ke PATH sistem untuk semua pengguna atau Tambahkan CMake ke PATH sistem untuk pengguna saat ini. Jangan khawatir tentang fakta bahwa CMake versi 64-bit tidak tersedia, CMake hanyalah alat konfigurasi dan tidak melakukan kompilasi apa pun. Sebaliknya, pada Windows, itu menciptakan file proyek yang dapat dibuka dengan Visual Studio.

2. Unduh dan instal Microsoft Visual Studio 2013 (edisi Desktop jika Anda bekerja pada Windows 7) dari

<https://www.visualstudio.com/products/free-developeroffers-vs>

Perhatikan bahwa Anda harus masuk dengan akun Microsoft Anda dan jika Anda tidak memilikinya, Anda dapat membuatnya di tempat. Instal perangkat lunak dan reboot setelah instalasi selesai. **Untuk MinGW, dapatkan penginstalnya dari**

<http://sourceforge.net/projects/mingw/files/Installer/mingw->

dan

<http://sourceforge.net/projects/mingw/files/OldFiles/mingw->

Saat menjalankan penginstal, pastikan jalur tujuan tidak mengandung spasi dan bahwa kompiler C ++ opsional disertakan. Edit variabel PATH sistem dan tambahkan;

C:\MinGW\bin

(dengan asumsi MinGW diinstal ke lokasi default). Mulai ulang sistem.

3. Secara opsional, unduh dan instal OpenNI 1.5.4.0 dari tautan yang disediakan di beranda GitHub di OpenNI di

`https://github.com/OpenNI/OpenNI`

4. Anda dapat mengunduh dan menginstal SensorKinect 0.93 dari

`https://github.com/avin2/SensorKinect/blob/unstable/Bin/Sens`

Atau, untuk Python 64-bit, unduh di

`https://github.com/avin2/SensorKinect/blob/unstable/Bin/Sen`

(64-bit). Perhatikan bahwa repositori ini tidak aktif selama lebih dari tiga tahun.

5. Unduh ZIP self-extracting dari OpenCV 3.0.0 dari

`https://github.com/Itseez/opencv`

Jalankan ZIP yang mengekstraksi sendiri, dan ketika diminta, masukkan folder tujuan apa pun, yang akan kita sebut sebagai

`<unzip_destination>. Subfolder, <unzip_destination>\opencv`

kemudian dibuat.

6. Buka Command Prompt dan buat folder lain tempat build kita akan menggunakan ini perintah:

`mkdir <build_folder> Ubah direktori folder build: cd <build`

7. Sekarang, kami siap mengonfigurasi bangunan kami. Untuk memahami semua opsi, kita dapat membaca kode di

`<unzip_destination>\opencv\CMakeLists.txt`

Namun, untuk tujuan buku ini, kita hanya perlu menggunakan opsi yang akan memberi kita rilis dengan binding Python, dan secara opsional, kedalaman dukungan kamera melalui OpenNI dan SensorKinect.

8. Buka CMake (cmake-gui) dan tentukan lokasi kode sumber OpenCV dan folder tempat Anda ingin membangun pustaka. Klik pada Konfigurasi. Pilih proyek yang akan dihasilkan. Dalam hal ini, pilih Visual Studio 12 (yang sesuai dengan Visual Studio 2013). Setelah CMake selesai mengkonfigurasi proyek, itu akan menampilkan daftar opsi build. Jika Anda melihat latar belakang merah, itu berarti bahwa proyek Anda mungkin perlu dikonfigurasi ulang: CMake mungkin melaporkan bahwa ia gagal menemukan beberapa dependensi. Banyak dependensi OpenCV adalah opsional, jadi jangan terlalu khawatir. **Catatan** Jika build gagal diselesaikan atau Anda mengalami masalah di kemudian hari, coba instal dependensi yang hilang (sering kali tersedia sebagai binari prebuilt), dan

kemudian bangun kembali OpenCV dari langkah ini. Anda memiliki opsi untuk memilih / membatalkan pilihan opsi bangunan (sesuai dengan perpustakaan yang telah Anda instal pada mesin Anda) dan klik Konfigurasi lagi, sampai Anda mendapatkan latar belakang yang jelas (putih).

9. Di akhir proses ini, Anda dapat mengeklik Hasilkan, yang akan membuat file OpenCV.sln di folder yang Anda pilih untuk membangun. Anda kemudian dapat menavigasi ke

```
<build_folder>/OpenCV.sln
```

dan membuka file dengan Visual Studio 2013, dan melanjutkan dengan membangun proyek,

```
ALL_BUILD
```

. Anda perlu membangun versi Debug dan Rilis dari OpenCV, jadi lanjutkan dan bangun perpustakaan dalam mode Debug, lalu pilih Lepaskan dan bangun kembali (F7 adalah kunci untuk meluncurkan bangunan).

10. Pada tahap ini, Anda akan memiliki folder bin di direktori build OpenCV, yang akan berisi semua file .dll yang dihasilkan yang memungkinkan Anda untuk memasukkan OpenCV dalam proyek Anda. Atau, untuk MinGW, jalankan perintah berikut:

```
cmake -D: CMAKE_BUILD_TYPE = RELEASE -D: WITH_OPENNI = ON -O
"MinGWMakefiles" <unzip_destination>\opencv
```

```
Jika OpenNI tidak diinstal, abaikan -D: WITH_OPENNI = ON. (D
<openni_install_destination>\Lib -D: OPENNI_INCLUDE_DIR =
<openni_install_destination>\Sertakan -
D: OPENNI_PRIME_SENSOR_MODULE_BIN_DIR =
<sensorkinect_install_destination>\Sensor\Bin.
```

Atau, untuk MinGW, jalankan perintah ini:

```
mingw32-make
```

11. Salin

```
<build_folder> \ lib \ Release \ cv2.pyd (dari Visual Studio
<build_folder> \ lib \ cv2.pyd (dari build MinGW) hingga
<python_installation_folder> \ paket-situs.
```

12. Terakhir, edit variabel PATH sistem dan tambahkan

```
<build_folder>/bin/Release
```

(untuk build Visual Studio) atau

```
<build_folder>/bin
```

(untuk build MinGW). Mulai ulang sistem anda.

1.2.2 Instalasi pada OS X

Beberapa versi Mac digunakan dengan versi Python 2.7 yang sudah diinstal sebelumnya yang disesuaikan oleh Apple untuk kebutuhan internal sistem. Namun, ini telah berubah dan versi standar OS X dikirimkan dengan instalasi standar Python. Di python.org, Anda juga dapat menemukan biner universal yang kompatibel dengan sistem Intel baru dan PowerPC lama.

Catatan

Anda dapat memperoleh penginstal ini di

<https://www.python.org/downloads/release/python-279/>

(lihat PPC Mac OS X 32-bit atau tautan Intel® Mac OS X 64-bit). Menginstal Python dari file .dmg yang diunduh hanya akan menimpa instalasi sistem Python Anda saat ini.

Untuk Mac, ada beberapa pendekatan yang mungkin untuk mendapatkan standar Python 2.7, NumPy, SciPy, dan OpenCV. Semua pendekatan pada akhirnya membutuhkan OpenCV untuk dikompilasi dari sumber menggunakan Alat Pengembang Xcode. Namun, tergantung pada pendekatannya, tugas ini otomatis bagi kami dalam berbagai cara oleh alat pihak ketiga. Kami akan melihat pendekatan semacam ini menggunakan MacPorts atau Homebrew. Alat-alat ini berpotensi melakukan segala yang dapat dilakukan CMake, ditambah lagi membantu kami mengatasi dependensi dan memisahkan pustaka pengembangan kami dari pustaka sistem.

Tip

Saya merekomendasikan MacPorts, terutama jika Anda ingin mengkompilasi OpenCV dengan dukungan kamera mendalam melalui OpenNI dan SensorKinect. Tambalan dan skrip yang relevan, termasuk beberapa yang saya kelola, siap pakai untuk MacPorts. Sebaliknya, Homebrew saat ini tidak memberikan solusi yang sudah jadi untuk mengkompilasi OpenCV dengan dukungan kamera yang dalam. Sebelum melanjutkan, pastikan bahwa Alat Pengembang Xcode disiapkan dengan benar:

Unduh dan instal Xcode dari Mac App Store atau

<https://developer.apple.com/xcode/downloads/>

Selama instalasi, jika ada opsi untuk menginstal Command Line Tools, pilihlah.

Buka Xcode dan terima perjanjian lisensi.

Langkah terakhir diperlukan jika penginstal tidak memberi kami opsi untuk menginstal Alat Baris Perintah. Arahkan ke Xcode, Preferensi, Unduh, dan klik tombol Instal di sebelah Command Line Tools. Tunggu instalasi untuk menyelesaikan dan keluar dari Xcode.

Atau, Anda dapat menginstal alat baris perintah Xcode dengan menjalankan perintah berikut (di terminal):

```
$ xcode-select {install}
```

Sekarang, kami memiliki kompiler yang diperlukan untuk pendekatan apa pun.

Menggunakan MacPorts dengan paket yang sudah jadi

Kita bisa menggunakan manajer paket MacPorts untuk membantu kita mengatur Python 2.7, NumPy, dan OpenCV. MacPorts menyediakan perintah terminal yang mengotomatiskan proses mengunduh, mengkompilasi, dan menginstal berbagai perangkat lunak sumber terbuka (OSS). MacPorts juga menginstal dependensi sesuai kebutuhan. Untuk setiap perangkat lunak, dependensi dan resep bangunan didefinisikan dalam file konfigurasi yang disebut Portfile. Repositori MacPorts adalah kumpulan dari Portfiles.

Mulai dari sistem di mana Xcode dan alat-alat command-line-nya sudah diatur, langkah-langkah berikut akan memberi kita instalasi OpenCV melalui MacPorts:

1. Unduh dan instal MacPorts dari

<http://www.macports.org/install.php>.

2. Jika Anda ingin dukungan untuk kamera kedalaman Kinect, Anda perlu memberi tahu MacPorts tempat untuk mengunduh Portfile khusus yang telah saya tulis. Untuk melakukannya, edit

`/opt/local/etc/macports/sources.conf` (dengan asumsi bahwa MacPorts diinstal ke lokasi default). Tepat di atas garis,

`rsync: //rsync.macports.org/release/ports/ [default]`, tambahkan baris berikut:

`http://nummist.com/opencv/ports.tar.gz`

Simpan file. Sekarang, MacPorts tahu bahwa ia harus mencari Portfile di repositori online saya terlebih dahulu, dan kemudian repositori online default.

3. Buka terminal dan jalankan perintah berikut untuk memperbarui MacPorts:

```
$ sudo port selfupdate
```

Saat diminta, masukkan kata sandi Anda.

4. Sekarang (jika kita menggunakan repositori saya), jalankan perintah berikut untuk menginstal OpenCV dengan binding Python 2.7 dan dukungan untuk kamera kedalaman, termasuk Kinect:

```
$ sudo port instal opencv + python27 + openni_sensorkinect
```

Atau (dengan atau tanpa repositori saya), jalankan perintah berikut untuk menginstal OpenCV dengan binding Python 2.7 dan dukungan untuk kamera kedalaman, tidak termasuk Kinect:

```
$ sudo port instal opencv + python27 + openni
```

Catatan

Ketergantungan, termasuk Python 2.7, NumPy, OpenNI, dan (dalam contoh pertama) SensorKinect, diinstal secara otomatis juga. Dengan menambahkan `python27` ke perintah, kami menentukan bahwa kami ingin varian OpenCV (membangun konfigurasi) dengan Python 2.7 binding. Demikian pula, menambahkan `openni_sensorkinect` menentukan varian dengan dukungan selalu mungkin untuk kamera kedalaman melalui OpenNI dan SensorKinect. Anda dapat menghilangkan `+ openni_sensorkinect` jika Anda tidak bermaksud menggunakan kamera kedalaman, atau Anda dapat menggantinya dengan `+ openni` jika Anda bermaksud menggunakan kamera kedalaman yang kompatibel dengan OpenNI tetapi tidak dengan Kinect. Untuk melihat daftar lengkap varian yang tersedia sebelum menginstal, kita dapat memasukkan perintah berikut:

```
$ port varian opencv
```

Bergantung pada kebutuhan penyesuaian kami, kami dapat menambahkan varian lain ke perintah pemasangan. Untuk fleksibilitas yang lebih besar, kita dapat menulis varian kita sendiri (seperti yang dijelaskan di bagian selanjutnya).

5. Juga, jalankan perintah berikut untuk menginstal SciPy:

```
$ sudo port install py27-scipy
```

6. Eksekusi instalasi Python bernama `python2.7`. Jika kita ingin menautkan python default yang dapat dieksekusi ke `python2.7`, mari kita juga jalankan perintah ini:

```
$ sudo port install python_select
```

```
$ sudo port pilih python python27
```

Menggunakan MacPorts dengan paket kustom Anda sendiri

Dengan beberapa langkah tambahan, kita dapat mengubah cara MacPorts mengkompilasi OpenCV atau perangkat lunak lainnya. Seperti disebutkan sebelumnya, resep build MacPorts didefinisikan dalam file konfigurasi yang disebut Portfiles. Dengan membuat atau mengedit Portfile, kita dapat mengakses alat bantu yang sangat dapat dikonfigurasi, seperti CMake, sementara juga memanfaatkan fitur MacPorts, seperti resolusi ketergantungan.

Mari kita asumsikan bahwa kita sudah menginstal MacPorts. Sekarang, kita dapat mengkonfigurasi MacPorts ke gunakan Portfile khusus yang kita tulis:

1. Buat folder di suatu tempat untuk menampung Portfiles khusus kami. Kami akan merujuk ke folder ini sebagai

```
<local_repository>
```

2. Edit file `/opt/local/etc/macports/sources.conf` (dengan anggapan bahwa MacPorts diinstal ke lokasi default). Tepat di atas

```
rsync: //rsync.macports.org/release/ports/ default baris, t
file: // <local_repository>
Misalnya, jika <local_repository> adalah / Users / Joe / Por
file: /// Pengguna / Joe / Portfiles
```

Catat tiga tebasan dan simpan file. Sekarang, MacPorts tahu bahwa ia harus mencari Portfiles di

```
<local_repository>
```

terlebih dahulu, dan kemudian, repositori online default-nya.

3. Buka terminal dan perbarui MacPorts untuk memastikan bahwa kami memiliki Portfile terbaru dari repositori default:

```
$ sudo port selfupdate
```

4. Mari kita salin opencv Portfile repositori default sebagai contoh. Kami juga harus menyalin struktur direktori, yang menentukan bagaimana paket dikategorikan oleh MacPorts:

```
$ mkdir <local_repository> / graphics /
$ cp
/opt/local/var/macports/sources/rsync.macports.org/release/p
cs / opencv <local_repository> / grafis
```

Sebagai alternatif, untuk contoh yang menyertakan dukungan Kinect, kita dapat mengunduh repositori online saya dari <http://nummist.com/opencv/ports.tar.gz>, unzip, dan salin seluruh folder grafiknya ke

```
<local_repository>
```

```
:
```

```
$ cp <unzip_destination> / graphics <local_repository>
```

5. Edit

```
<local_repository>/graphics/opencv/Portfile.
```

Perhatikan bahwa file ini menentukan flag konfigurasi, dependensi, dan varian CMake. Untuk detail tentang pengeditan Portfile, buka

```
http://guide.macports.org/#development.
```

Untuk melihat flag konfigurasi CMake mana yang relevan dengan OpenCV, kita perlu melihat kode sumbernya. Unduh arsip kode sumber dari

```
https://github.com/Itseez/opencv/archive/3.0.0.zip, unzip ke
```

Setelah melakukan pengeditan ke Portfile, simpanlah.

6. Sekarang, kita perlu membuat file indeks di repositori lokal kita sehingga MacPorts dapat menemukan Portfile baru:

```
$ cd <local_repository>
$ portindex
```

7. Mulai sekarang, kita dapat memperlakukan file pembuka kustom kita sama seperti paket MacPorts lainnya. Sebagai contoh, kita dapat menginstalnya sebagai berikut:

```
$ sudo port instal opencv + python27 + openni_sensorkinect
```

Perhatikan bahwa Portfile repositori lokal kami lebih diutamakan daripada Portfile repositori default karena urutan urutannya. `/opt/local/etc/macports/sources.conf`.

Menggunakan Homebrew dengan paket yang sudah jadi (tidak ada dukungan untuk kamera kedalaman)

Homebrew adalah manajer paket lain yang dapat membantu kami. Biasanya, MacPorts dan Homebrew tidak boleh diinstal pada mesin yang sama.

Mulai dari sistem di mana Xcode dan alat-alat command-line-nya sudah diatur, langkah-langkah berikut akan memberi kita instalasi OpenCV melalui Homebrew:

1. Buka terminal dan jalankan perintah berikut untuk menginstal Homebrew:

```
$ ruby -e "$ (curl -fsSkLraw.github.com/mxcl/homebrew/go) "
```

2. Tidak seperti MacPorts, Homebrew tidak secara otomatis menempatkan executable-nya di PATH. Untuk melakukannya, buat atau edit file

```
~/.profile
```

dan tambahkan baris ini di bagian atas kode:

```
eksport PATH = / usr / local / bin: / usr / local / sbin: $
```

Simpan file dan jalankan perintah ini untuk menyegarkan PATH:

```
$ source ~/.profile
```

Perhatikan bahwa executable yang diinstal oleh Homebrew sekarang diutamakan daripada executable yang diinstal oleh sistem.

3. Untuk laporan diagnostik mandiri Homebrew, jalankan perintah berikut:

```
$ brew doctor
```

Ikuti saran pemecahan masalah yang diberikannya.

4. Sekarang, perbarui Homebrew:

```
$ brew update
```

5. Jalankan perintah berikut untuk menginstal Python 2.7:

```
$ brew install python
```

6. Sekarang, kita dapat menginstal NumPy. Pilihan paket pustaka Python Homebrew terbatas, jadi kami menggunakan alat manajemen paket terpisah yang disebut pip, yang dilengkapi dengan Homebrew Python:

```
$ pip install numpy
```

7. SciPy berisi beberapa kode Fortran, jadi kita memerlukan kompiler yang sesuai. Kita dapat menggunakan Homebrew untuk menginstal kompiler gfortran:

```
$ brew install gfortran
```

Sekarang, kita dapat menginstal SciPy:

```
$ pip install scipy
```

8. Untuk menginstal OpenCV pada sistem 64-bit (semua perangkat keras Mac baru sejak akhir 2006), jalankan perintah berikut:

Tip Mengunduh kode contoh Anda dapat mengunduh file kode contoh untuk semua buku Penerbitan Packt yang telah Anda beli dari akun Anda di <http://www.packtpub.com>.

Jika Anda membeli buku ini di tempat lain, Anda dapat mengunjungi

<http://www.packtpub.com/support>

dan mendaftar agar file-file tersebut diemail langsung kepada Anda.

Menggunakan Homebrew dengan paket kustom Anda sendiri

Homebrew memudahkan untuk mengedit definisi paket yang ada:

```
$ brew edit opencv
```

Definisi paket sebenarnya adalah skrip dalam bahasa pemrograman Ruby.

Kiat untuk mengeditnya dapat ditemukan di halaman Wiki Homebrew di

<https://github.com/mxcl/homebrew/wiki/Formula-Cookbook>.

Sebuah skrip dapat menentukan flag konfigurasi Make atau CMake, antara lain.

Untuk melihat flag konfigurasi CMake mana yang relevan dengan OpenCV, kita perlu melihat kode sumbernya. Unduh arsip kode sumber dari

<https://github.com/Itseez/opencv/archive/3.0.0.zip>

, unzip ke lokasi mana pun, dan baca

```
<unzip_destination>/OpenCV-2.4.3/CMakeLists.txt.
```

Setelah mengedit skrip Ruby, simpan. Paket khusus dapat diperlakukan seperti biasa. Misalnya, dapat diinstal sebagai berikut:

```
$ brew install opencv
```

1.2.3 Instalasi pada Ubuntu

Pertama dan terpenting, berikut adalah catatan singkat tentang versi Ubuntu dari sistem operasi: Ubuntu memiliki siklus rilis 6 bulan di mana setiap rilis adalah versi minor .04 atau .10 dari versi utama (14 pada saat penulisan). Namun, setiap dua tahun, Ubuntu merilis versi yang diklasifikasikan sebagai dukungan jangka panjang (LTS) yang akan memberi Anda dukungan lima tahun oleh Canonical (perusahaan di belakang Ubuntu). Jika Anda bekerja di lingkungan perusahaan, disarankan untuk menginstal salah satu versi LTS. Yang terbaru yang tersedia adalah 14.04.

Ubuntu hadir dengan Python 2.7 yang sudah diinstal. Repositori Ubuntu standar berisi paket OpenCV 2.4.9 tanpa dukungan untuk kamera yang dalam. Pada saat penulisan ini, OpenCV 3 belum tersedia melalui repositori Ubuntu, jadi kita harus membuatnya dari sumber. Untungnya, sebagian besar sistem Unix-like dan Linux datang dengan semua perangkat lunak yang diperlukan untuk membangun proyek dari awal yang sudah diinstal. Ketika dibangun dari sumber, OpenCV dapat mendukung kamera kedalaman melalui OpenNI dan SensorKinect, yang tersedia sebagai binari yang dikompilasi dengan skrip instalasi.

Menggunakan repositori Ubuntu (tidak ada dukungan untuk kamera kedalaman)

Kita dapat menginstal Python dan semua dependensi yang diperlukan menggunakan manajer paket apt, dengan menjalankan perintah berikut:

```
> sudo apt-get install build-essential
> sudo apt-get install cmake git libgtk2.0-dev pkg-config
libavformat-dev libswscale-dev
> sudo apt-get install python-dev python-numpy libtbb2 li
```

Secara setara, kita bisa menggunakan Ubuntu Software Center, yang merupakan tampilan grafis apt package manager.

Membangun OpenCV dari sumber Sekarang kita telah menginstal seluruh tumpukan Python dan cmake, kita dapat membangun OpenCV. Pertama, kita perlu mengunduh kode sumbernya <https://github.com/Itseez/opencv> strak arsip dan pindahkan ke folder yang tidak di-zip di terminal. Kemudian, jalankan perintah berikut:

```
> mkdir build
> cd build
> cmake -D CMAKE_BUILD_TYPE =Release -D CMAKE_INSTALL_PREFIX=/usr/local
> make
> make install
```

Setelah instalasi berakhir, Anda mungkin ingin melihat contoh Python OpenCV di `<opencv_folder>/opencv/samples/python` dan `<script_folder>/python`.

Instalasi pada sistem mirip Unix lainnya Pendekatan untuk Ubuntu (seperti yang dijelaskan sebelumnya) kemungkinan akan bekerja pada distribusi Linux apa pun yang berasal dari Ubuntu 14.04 LTS atau Ubuntu 14.10 sebagai berikut: Kubuntu 14.04 LTS atau Kubuntu 14.10 Xubuntu 14.04 LTS atau Xubuntu 14.10 Linux Mint 17

Pada Debian Linux dan turunannya, manajer paket apt berfungsi sama seperti di Ubuntu, meskipun paket yang tersedia mungkin berbeda.

Di Gentoo Linux dan turunannya, manajer paket Portage mirip dengan MacPorts (seperti dijelaskan sebelumnya), meskipun paket yang tersedia mungkin berbeda.

Pada turunan FreeBSD, proses instalasi sekali lagi mirip dengan MacPorts; sebenarnya, MacPorts berasal dari sistem instalasi port yang diadopsi pada FreeBSD. Bacalah Buku Pegangan FreeBSD yang luar biasa di <https://www.freebsd.org/doc/en/books/porters-handbook/> untuk ikhtisar proses instalasi perangkat lunak.

Pada sistem mirip Unix lainnya, manajer paket dan paket yang tersedia mungkin berbeda. Konsultasikan dokumentasi manajer paket Anda dan cari paket dengan opencv di namanya. Ingatlah bahwa OpenCV dan binding Python-nya dapat dipecah menjadi beberapa paket.

Juga, cari semua catatan instalasi yang diterbitkan oleh penyedia sistem, pengelola repositori, atau komunitas. Karena OpenCV menggunakan driver kamera dan codec media, membuat semua fungsinya berfungsi dapat menjadi rumit pada sistem dengan dukungan multimedia yang buruk. Dalam beberapa keadaan, paket sistem mungkin perlu dikonfigurasi ulang atau diinstal ulang untuk kompatibilitas.

Jika paket tersedia untuk OpenCV, periksa nomor versinya. OpenCV 3 atau lebih tinggi direkomendasikan untuk tujuan buku ini. Juga, periksa apakah paket menawarkan binding Python dan dukungan kamera mendalam melalui OpenNI dan SensorKinect. Terakhir, periksa apakah ada orang di komunitas pengembang yang melaporkan keberhasilan atau kegagalan dalam menggunakan paket.

Jika, sebaliknya, kami ingin melakukan pembuatan kustom OpenCV dari sumber, mungkin bermanfaat untuk merujuk ke skrip instalasi untuk Ubuntu (seperti yang dibahas sebelumnya) dan menyesuaikannya dengan manajer paket dan paket yang ada di sistem lain.

1.2.4 Instalasi modul Contrib

Berbeda dengan OpenCV 2.4, beberapa modul terdapat dalam repositori yang disebut `opencv_contrib`, yang tersedia di https://github.com/Itseez/opencv_contrib. Saya sangat merekomendasikan menginstal modul ini karena mengandung fungsionalitas tambahan yang tidak termasuk dalam OpenCV, seperti modul pengenalan wajah. Setelah diunduh (baik melalui zip atau git, saya sarankan git agar Anda dapat tetap up to date dengan perintah `git pull` sederhana), Anda dapat menjalankan kembali perintah `cmake` Anda untuk memasukkan pembangunan OpenCV dengan modul `opencv_contrib` sebagai berikut:

```
cmake -DOPENCV_EXTRA_MODULES_PATH = <opencv_contrib> / mo  
<opencv_source_directory>
```

Jadi, jika Anda telah mengikuti prosedur standar dan membuat direktori build di folder unduhan OpenCV Anda, Anda harus menjalankan perintah berikut:

```
mkdir build && cd build  
cmake -D CMAKE_BUILD_TYPE = Lepaskan -DOPENCV_EXTRA_MODULES_PATH =  
<opencv_contrib> / modules -D CMAKE_INSTALL_PREFIX = / us  
make
```

