

# Pengenalan Pola Bentuk Wajah dengan OpenCV

Tengku Cut Al-Saidina Zulkhaidi \*<sup>1</sup>, Eny Maria<sup>2</sup>, Yulianto<sup>3</sup>

Program Studi Teknologi Rekayasa Perangkat Lunak, Politeknik Pertanian Negeri Samarinda  
e-mail: \*rezer0punch@gmail.com, <sup>2</sup>mariaeny.siringo2@gmail.com, <sup>3</sup>yulianto.tile@yahoo.com

## Abstrak

Pada penelitian ini akan menggunakan module OpenCV pada bahasa pemrograman python untuk mengenali wajah seseorang yang menggunakan Haar Cascades untuk mengenali bentuk wajah dan mata. Tahapan awal menggunakan open source dari intel untuk data wajah dan mata, dipadukan dengan module cascade classifier pada openCV untuk merubah data menjadi pengenalan bentuk wajah dari titik pada wajah yang dianggap sesuai dengan data yang telah disediakan. Banyak dari beberapa sistem pendeteksian wajah menggunakan metode computer vision sebagai metode pendeteksi objek. Metode computer vision dikenal memiliki kecepatan dan keakuratan yang tinggi karena menggabungkan beberapa konsep (Haar Features, Integral Image, AdaBoost, dan Cascade Classifier) menjadi sebuah metode utama untuk mendeteksi objek. Banyak dari sistem deteksi tersebut menggunakan C atau C++ sebagai bahasa pemrograman, dan OpenCV sebagai library deteksi objek. Hal ini dikarenakan library OpenCV menerapkan metode computer vision kedalam sistem deteksinya, sehingga memudahkan dalam pembuatan sistem. Penelitian ini bertujuan untuk mengimplementasikan computer vision ke dalam sistem deteksi wajah sederhana dengan memanfaatkan library yang ada pada OpenCV dan memanfaatkan bahasa pemrograman Python sebagai pondasi sistem.

**Kata kunci**—Pengenalan Pola, OpenCV, Python

## 1. PENDAHULUAN

Pada bandara dan juga pada security keamanan telah menggunakan *computer vision* untuk mengenali bentuk wajah seseorang dengan mata dan untuk mengetahui apakah orang tersebut pelaku kejahatan atau akan melakukan tindakan kejahatan. Begitu penting teknologi *computer vision* maka pengembangan teknologi informasi akan membantu di segala bidang. *Computer Vision* adalah teknologi yang membuat komputer dapat melihat dan mengenali bentuk yang meniru mata dan otak manusia. Penggunaan *computer vision* dapat dipadukan dengan *machine learning* untuk memahami dan menirukan sifat manusia dengan baik.

Pengembangan *computer vision* banyak menggunakan module yang telah disediakan para pengembang seperti Intel yang menyediakan module OpenCV secara *open source* yang dapat digunakan untuk bahasa pemrograman Python dan C++. Pendeteksian mata dan wajah mengambil data yang telah disediakan oleh Intel yaitu *Haar Cascades* (koordinat (x,y)) yang mengidentifikasi wajah dan mata dengan pixel pada kamera. Perhitungan numeric menggunakan *numpy* modul pada python agar dapat menghitung koordinat dengan tepat pada wajah dan mata [1].

## 2. METODE PENELITIAN

Pengenalan wajah adalah suatu metoda pengenalan berorientasi pada wajah. Pengenalan ini dapat dibagi menjadi dua, yaitu dikenali atau tidak dikenali. Pada pola wajah akan di deteksi wajah berbentuk kotak dan mengenali wajah. Metoda pengenalan wajah memakai dua prosedur, yaitu pengenalan kontur wajah dengan mengenali mata dan postur wajah. Karakteristik organ

tersebut kemudian dinyatakan dalam bentuk vektor dan analisis komponen yang paling sesuai, mencari perhitungan model terbaik yang menjelaskan bentuk wajah dengan mengutip informasi yang paling relevan.

Saat ini, penelitian mengenai pengenalan wajah dengan cepat berkembang. Aplikasi komersial tentang ini telah banyak diimplementasikan namun pada dasarnya teknologi ini belum sempurna. Penelitian perlu terus dikembangkan untuk memperoleh hasil yang diinginkan. Satu hal yang dapat ditambah atau diimplementasikan dalam pengembangan teknologi pengenalan wajah yaitu dengan menambah tingkat kecepatan dan akurasi dalam pendeteksian wajah. Banyak dari sistem pendeteksian tersebut menggunakan metode *computer vision* sebagai metode pendeteksi objek. *Computer vision* dikenal memiliki kecepatan dan keakuratan yang cukup tinggi karena menggabungkan beberapa konsep (Fitur Haar, Citra Integral, AdaBoost, Cascade Classifier) menjadi sebuah metode utama untuk mendeteksi objek.

### 2.1 OpenCV dan Python

*Open Computer Vision* (OpenCV) merupakan *library open source* yang tujuannya dikhususkan untuk melakukan pengolahan citra. Maksudnya adalah agar komputer mempunyai kemampuan yang mirip dengan cara pengolahan visual pada manusia. OpenCV telah menyediakan banyak algoritma visi komputer dasar. OpenCV juga menyediakan modul pendeteksian objek yang menggunakan metode *computer vision*.

Python adalah salah satu bahasa pemrograman tingkat tinggi yang bersifat *interpreter*, *interactive*, *object-oriented*, dan dapat beroperasi hampir di semua platform: Mac, Linux, dan Windows. Python termasuk bahasa pemrograman yang mudah dipelajari karena sintaks yang jelas, dapat dikombinasikan dengan penggunaan modul siap pakai, dan struktur data tingkat tinggi yang efisien. Distribusi Python dilengkapi dengan suatu fasilitas seperti *shell* di Linux. Lokasi penginstalan Python biasa terletak di `"/usr/bin/Python"`, dan bisa berbeda. Menjalankan Python, cukup dengan mengetikkan "Python", tunggu sebentar lalu muncul tampilan `">>>"`, berarti Python telah siap menerima perintah. Ada juga tanda `"..."` yang berarti baris berikutnya dalam suatu blok prompt `'>>>'`. Text editor digunakan untuk modus skrip.

Untuk membangun penelitian ini digunakan wxPython yang merupakan *toolkit* GUI untuk bahasa pemrograman Python. wxPython memungkinkan programmer Python untuk membuat aplikasi dengan pondasi kuat, grafis antarmuka dengan pengguna yang sangat fungsional, sederhana, dan mudah. wxPython diimplementasikan sebagai modul ekstensi oleh Python (kode asli). wxPython membungkus wxWidget sebagai salah satu GUI library populer yang ditulis dalam bahasa C++. Selain itu, digunakan pula *Boa Constructor* yang merupakan *Integrated Development Environment* (IDE) untuk Python dan wxPython GUI *Builder* yang *cross-platform*. *Boa Constructor* mampu membuat, memanipulasi frame secara visual (tanpa skrip), dan ada banyak object *inspector* seperti: browser objek, hirarki warisan, *debugger* yang canggih, dan bantuan yang sudah terintegrasi.

### 2.2 Pengolahan Citra Awal

Proses ketika citra non-digital diubah ke citra digital. Citra digital diperoleh dari hasil digitisasi citra analog. Digitisasi citra melibatkan dua proses, yaitu sampling dan kuantisasi. Sampling menunjukkan banyaknya pixel/blok untuk mendefinisikan suatu gambar. Kuantisasi menunjukkan banyaknya derajat nilai pada setiap pixel (menunjukkan jumlah bit pada gambar digital, black/white dengan 2 bit, grayscale dengan 8 bit, true color dengan 24 bit). Pengolahan Citra dilakukan untuk memperbaiki kualitas citra agar mudah untuk diinterpretasi oleh manusia/komputer. Masukannya adalah citra dan keluarannya juga citra, tetapi dengan kualitas lebih baik dari pada citra masukan. Operasi pengolahan citra yang berhubungan dalam deteksi wajah: *grayscale*, *neighborhood operation*, *thresholding*, *histogram equalization*, *resizing*. Proses analisis citra, dimulai dari pencitraan, sampai proses terakhir, sehingga didapat sebuah keputusan untuk maksud atau tujuan tertentu, misalnya, memandu robot, dan lain-lain. Pada penelitian ini lebih focus bagaimana menentukan wajah manusia di antara background.

---



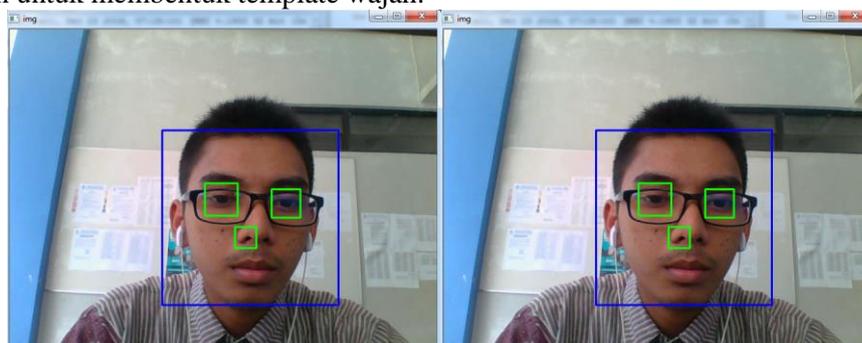
Gambar 1 Diagram Pengolahan Citra

### 3. HASIL DAN PEMBAHASAN

Dalam penelitian ini pengembangan perangkat lunak pengenalan citra wajah menggunakan modul *OpenCV* dengan menggunakan bahasa pemrograman python untuk menulis *source code*. Pada identifikasi wajah juga menggunakan *open source* dari intel untuk data pada wajah. Data yang telah disediakan akan memudahkan dalam pengenalan pola wajah dan akan dipadukan dengan *OpenCV* untuk pengenalan pola dan klasifikasinya. Pada penelitian ini, diambil sampel data dengan *webcam* pada 1 wajah dan 2 wajah sekaligus. Saat mata berkedip dan menggunakan kacamata, pada tempat yang terang. Aplikasi mengenali wajah dengan bentuk kotak dengan warna-warna tertentu sesuai wajah dan mata.

Pada tahap desain dilakukan perancangan sistem pengenalan citra wajah, yaitu Metode klasifikasi pada *OpenCV* mengidentifikasi data yang telah disediakan, algoritma *OpenCV* akan mengenali data wajah menjadi pengenalan wajah.

Klasifikasi citra dari nilai fitur-fitur sederhana dan menggunakan tiga jenis fitur, yaitu fitur persegi, fitur tiga persegi, dan fitur empat persegi. Nilai dari fitur-fitur tersebut adalah selisih antara daerah hitam dan putih. Di dalam tiap *sub-window image*, jumlah total dari Fitur *Haar* sangat besar, jauh lebih besar jika dibandingkan dengan jumlah *pixel*. Untuk memastikan pengklasifikasian dapat dilakukan secara cepat, proses pembelajaran harus menghilangkan fitur-fitur mayoritas yang tersedia, dan memusatkan pada sekumpulan kecil fitur yang perlu. Ada *Boost* bertujuan untuk membentuk template wajah.



Gambar 2 Pengenalan Pola Wajah dengan OpenCV

Suatu metode klasifikasi yang menggunakan beberapa tingkatan dalam penyeleksian. Pada tiap tingkatan dilakukan penyeleksian menggunakan algoritma *AdBoost* yang telah di-*training* dengan menggunakan Fitur *Haar*. Penyeleksian berguna untuk memisahkan antara *sub-window* yang mengandung positif objek (gambar yang terdeteksi memiliki objek yang diinginkan) dengan negatif objek (gambar yang terdeteksi tidak memiliki objek yang diinginkan).

Suatu prosedur penelitian dibutuhkan agar pekerjaan dapat dilaksanakan secara berurutan dan berkelanjutan tanpa harus mengganggu jenis pekerjaan lainnya. Persiapan meliputi segala sesuatu yang berhubungan

dengan proses perancangan, yakni: mempelajari dan memahami cara algoritma bekerja, perancangan untuk algoritma dan flowchart, pembuatan sistem, kemudian menganalisis sistem. Sistem yang akan dibuat mengacu diagram alir pada Gambar 2 dimana sistem setelah dijalankan akan mengaktifkan *webcam* untuk mengakuisisi citra, lalu dilakukan pengolahan citra: *grayscale*, *resizing*, *equalization*. Wajah kemudian dicari, jika wajah terdeteksi, akan dilakukan penggambaran garis persegi pada wajah tersebut.



Gambar 3 Diagram Alir Sistem Deteksi Wajah

Sistem akan mencari wajah ke berbagai lokasi citra. Awalnya, citra masukan di-scan per-sub window, dimulai dari kiri atas dengan ukuran minimal 20x20, diulangi secara iterasi dengan skala perbesaran 1.1. Proses ini diulangi dengan pergeseran  $\Delta x$  dan  $\Delta y$  sampai kanan bawah. Setiap sub-window yang di-scan, diterapkan Fitur Haar. Karena banyaknya fitur haar pada tiap sub-window, dilakukan penyeleksian fitur dengan AdaBoost. Penyeleksian fitur akan melibatkan nilai fitur, nilai fitur tersebut dihitung dengan Integral Image. Jumlah sub-window pada suatu citra terlalu banyak, maka dilakukan penyeleksian sub-window oleh Cascade Classifier (template). Sub-window yang lolos seluruh tahapan seleksi Classifier akan dideskripsikan sebagai wajah.

Pengujian dilakukan dengan notebook yang ber-webcam dengan resolusi 1.3 MP, dilakukan di kamar dengan bersumberkan cahaya matahari yang masuk. Ada beberapa macam pengujian yang dilakukan: berdasarkan pencahayaan, jarak, kemiringan, wajah terhalang objek

lain, beberapa wajah, dan berdasarkan karakter objek yang menyerupai wajah. Gambar 2 juga merupakan contoh pengujian berdasarkan kemiringan wajah  $0^\circ$  (depan). Terlihat sistem dapat mendeteksi wajah dengan akurat ketika posisi wajah dari sisi depan. Akurat artinya, pendeteksian benar 100% pada citra wajah tanpa ada false positive dan false negative. Telah dilakukan beberapa pengujian terhadap karakteristik wajah yang dapat dideteksi maupun tidak. Wajah yang terdeteksi akan muncul tanda segi empat di sekeliling wajah

#### 4. KESIMPULAN

- Sistem dapat mendeteksi wajah manusia dalam keadaan frontal dengan akurasi 100% dan waktu deteksi kurang dari 0.5 detik.
- Sistem juga dapat mendeteksi wajah manusia dalam keadaan non-frontal (tercatat mampu mendeteksi wajah dengan kemiringan  $\pm 71^\circ$ ) dan dapat mendeteksi adanya beberapa wajah dalam suatu citra.
- Sistem dapat juga mendeteksi objek yang menyerupai wajah ketika objek tersebut memiliki kontur yang sama dengan kontur wajah manusia (kontur wajah pada template), misalnya, wajah boneka atau topeng.

#### 5. SARAN

Penelitian lanjutan perlu dilakukan untuk meningkatkan akurasi pengenalan wajah. Karena akan banyak pengguna dengan wajah yang mirip apabila sistem diberlakukan secara masal. Tingkat error yang tinggi pada kondisi-kondisi tertentu perlu dilakukan perbaikan agar sistem dapat lebih akurat dalam mengenali wajah.

Perlu dilakukan perbaikan agar sistem dapat mendeteksi dan mengenali wajah dengan baik meskipun terdapat perbedaan yang mencolok antara citra masukan dengan citra acuan.

#### DAFTAR PUSTAKA

- [1] E. Maria, Yulianto, Y. P. Arinda, Jumiathy, and P. Nobel, "Segmentasi Citra Digital Bentuk Daun Pada Tanaman Di Politani Samarinda Menggunakan Metode Thresholding," *Jurti*, vol. 2, no. 1, pp. 37–46, 2018.
- [2] Arihutomo, Mukhlas, (2010), "Rancang Bangun Sistem Penjejakan Objek Menggunakan Metode Viola Jones Untuk Aplikasi EyeBot", ITS.
- [3] Chandra, Devy. Prajnagaja, Nagarjuna. Nugroho, Lintang Agung, (2011), "Studi Pendeteksian Wajah dengan Metode Viola Jones", BINUS.
- [4] Kadir, Abdul, (2005), "Dasar Pemrograman Python", Yogyakarta, Andi Offset.
- [5] Nugraha, Raditya, (2011), "GameTicTacToe dengan Gerakan Jari Menggunakan Metode Viola And Jones", Politeknik Elektronika Negeri Surabaya, ITS.
- [6] Gonzalez dan Woods. 2001. Digital Image Processing. California: Addison-Wesley Publishing Company

- [7] Viola, Paul. & Jones, Michaels, (2001), "Rapid Object Detection using boosted Cascade of Simple Features", IEEE CVPR.
  - [8] Imagine Publishing Ltd. . (2014). Raspberry Pi Complete manual. Bournemouth: Imagine Publishing Ltd Richmond house.
  - [9] E. S. R. N. Derian Indra Bramantio, "Perancangan dan Implementasi Keamanan Pengenalan Wajah dengan Metode Eigenface ," vol. 3, p. 80, 2016. Jurnal Fakultas Teknik Elektro Universitas Telkom.
  - [10] Bradburry, A., & Everard, B. (2014). Learning python with Raspberry Pi.
  - [11] Munir,Rinaldi. 2004.Pengolahan Citra digital dengan Pendekatan Algoritmik. Bandung: Informatika Bandung
-