

RANCANG BANGUN SISTEM DETEKSI WAJAH DENGAN METODE VIOLA-JONES UNTUK MENGIDENTIFIKASI IDENTITAS SESEORANG

Yulita Mahardini Poysancin¹, Aryo Nur Utomo²

Program Studi Teknik Informatika, Fakultas Sains dan Teknologi Informasi
Institut Sains dan Teknologi Nasional
Jl. Moh. Kahfi II, Bhumi Srengseng Indah, Jagakarsa, Jakarta Selatan 12640
Telp. (021) 7874647, Fax. (021) 7866955

¹ yulita.mahardini@gmail.com, ² aryo.nurutomo@gmail.com

ABSTRAK

Sistem pengenalan wajah ini menggunakan OpenCV sebagai library deteksi objek. Hal ini dikarenakan library OpenCV terdapat metode Viola Jones kedalam sistem deteksinya, sehingga memudahkan dalam pembuatan sistem. Penelitian ini bertujuan untuk mengimplementasikan Viola Jones ke dalam sistem deteksi wajah yang dapat mengidentifikasi sebuah identitas dengan memanfaatkan library yang ada pada OpenCV dan akses ke Database SQLite. Bahasa pemrograman Python digunakan untuk membangun sistem ini. Metodologi penelitian ini dimulai dari studi literatur, pengumpulan data, perancangan sistem, kemudian menganalisis data. Setelah sistem selesai dibuat, dilakukan pengujian sistem terhadap karakteristik wajah yang dapat mengidentifikasi identitas seseorang.

Kata Kunci: Deteksi Wajah, Viola Jones, OpenCV, Python, Identifikasi Identitas.

ABSTRACT

This face recognition system uses OpenCV as an object detection library. This is because the OpenCV library contains the Viola Jones method into its detection system, making it easier to make the system. This study aims to implement Viola Jones into a face detection system that can identify an identity by utilizing existing libraries in OpenCV and access to the SQLite Database. The Python programming language is used to build this system. The research methodology starts from the study of literature, collecting data, designing the system, then analyzing the data. After the system is finished, the system tests the facial characteristics that can identify a person's identity.

Keyword: Face Detection, Viola Jones, OpenCV, Python, Identification Identity.

1. PENDAHULUAN

Perkembangan teknologi sekarang ini memang sangat menakjubkan, khususnya pada bidang informasi dan komunikasi. Teknologi yang tadinya bersifat analog kini berangsur-angsur mulai berpindah ke teknologi digital. Salah satu contohnya yaitu pemanfaatan kartu pintar atau *smart card* seperti *e-KTP* dengan chip yang memuat informasi data biodata, foto, citra tanda tangan dan dua sidik jari telunjuk kanan dan kiri dan metode pengamanan yang tinggi. Sesuai dengan yang dijelaskan oleh Badan Pengkajian dan Penerapan Teknologi pada *website* resminya bahwa kegunaan dari teknologi tersebut merupakan langkah signifikan bagi optimalisasi layanan administrasi pemerintahan dan layanan publik secara elektronik. Pemanfaatan teknologi tersebut akan memudahkan dalam hal pengidentifikasian identitas dari hasil perekaman data, sehingga dapat menghasilkan ketunggalan identitas dengan sebuah kode unik dan tunggal sebagai basis pembuatan *database* (Permata Shulur, 2015a).

Penggunaan teknologi seperti yang dijelaskan di atas merupakan salah satu contoh pemanfaatan teknologi biometrik. Pada sebuah paper dijelaskan bahwa teknologi biometrik merupakan suatu metode untuk mengidentifikasi seseorang berdasarkan ciri khusus pada diri atau karakteristik perilaku seseorang. Pada diri manusia terdapat beberapa bagian tubuh yang bisa membedakan dengan orang lain meskipun dengan saudara kembarnya. Ciri khusus tersebut bisa berupa wajah, retina mata, sidik jari, suara maupun bagian tubuh lain yang menjadi identitas pribadi yang bisa diinterpretasikan menjadi suatu parameter atau data yang mempunyai nilai kuantitatif. Identitas tersebut bisa dideteksi dan dikenali dengan memanfaatkan Teknologi yang berkembang sekarang ini salah satunya adalah deteksi wajah (Permata Shulur, 2015a).

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 serta dapat mengidentifikasi sebuah nama dari wajah yang akan di deteksi. Banyak dari sistem pendeteksian tersebut menggunakan metode *Viola Jones*.

Metode *Viola Jones* dikenal memiliki kecepatan dan keakuratan yang cukup tinggi karena menggabungkan beberapa konsep yaitu *Fitur Haar*, *Citra Integral*, *AdaBoost*, *Cascade Classifier* menjadi sebuah metode utama untuk mendeteksi objek (Prasetya & Nurviyanto, 2012).

Dalam bagian ini dipaparkan berbagai informasi yang berhubungan dengan penelitian dan perancangan program.

Wajah

Wajah merupakan bagian dari tubuh manusia yang menjadi fokus perhatian di dalam interaksi sosial. Wajah memainkan peranan vital dengan menunjukkan identitas seseorang.

Visi komputer (*Computer Vision*)

Visi komputer juga merupakan dan kumpulan dari metode-metode untuk mendapatkan, memproses, menganalisis suatu gambar atau dalam arti lain visi komputer, merupakan kumpulan metode-metode yang digunakan untuk menghasilkan angka-angka atau simbol-simbol yang didapat dari gambar yang diambil dari dunia nyata agar komputer dapat mengerti apa makna dari gambar tersebut.

Pengolahan Citra

Bertujuan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin(dalam hal ini komputer). Teknik-teknik pengolahan citra mentransformasikan citra menjadi citra lain. Jadi, masukannya adalah citra dan keluarannya juga citra, namun citra keluaran mempunyai kualitas lebih baik daripada citra masukan. Termasuk ke dalam bidang ini juga adalah pemampatan citra (*image compression*).

Biometrik

Biometrik (berasal dari bahasa Yunani *bios* yang artinya hidup dan *metron* yang artinya mengukur) secara umum adalah studi tentang karakteristik biologi yang terukur

Deteksi Wajah

Deteksi wajah dapat dipandang sebagai masalah klasifikasi pola dimana inputnya adalah citra masukan dan akan ditentukan output yang berupa label kelas dari citra tersebut.

Algoritma Viola Jones

Viola-Jones telah memperkenalkan sebuah *framework* deteksi wajah yang mampu memproses gambar dengan sangat cepat dengan tingkat deteksi yang tinggi. *Viola-Jones* menerapkan algoritma *Adaboost* (*adaptive boosting*) algoritma yang dapat meningkatkan kinerja pendeteksian.

Pada sebuah jurnal *ECCIS* kedelapan disebutkan bahwa metode *Viola-Jones* merupakan metode pendeteksian obyek yang memiliki tingkat keakuratan yang cukup tinggi yaitu sekitar 93,7 % dengan kecepatan 15 kali lebih cepat dari pada detektor *Rowley Baluja-Kanade* dan kurang lebih 600 kali lebih cepat dari pada detektor *Schneiderman-Kanade*. Metode *Viola-Jones* menggabungkan empat kunci utama yaitu *Haar-Like Feature*, *Integral Image*, *Adaboost* dan *Cascade classifier* yang penjelasannya bisa dilihat di bawah ini.(SG Almaga, 2015).

Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python juga didukung oleh komunitas yang besar.

OpenCV Library

OpenCV adalah suatu library gratis yang dikembangkan oleh developer-developer Intel Corporation. Library ini terdiri dari fungsi-fungsi computer vision dan API (*Application Programming Interface*) untuk *image processing* high level maupun low level dan sebagai optimasi aplikasi realtime. OpenCV sangat disarankan untuk programmer yang akan berkecukupan pada bidang computer vision, karena library ini mampu menciptakan aplikasi yang handal, kuat dibidang digital vision, dan mempunyai kemampuan yang mirip dengan cara pengolahan visual pada manusia, Karena library ini bersifat cuma-cuma dan sifatnya yang open source, maka dari itu OpenCV tidak dipesan khusus untuk pengguna arsitektur Intel, tetapi dapat dibangun pada hamper semua arsitektur.

Anaconda

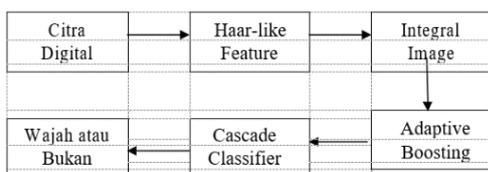
Anaconda adalah distribusi bebas dan *open source* bahasa pemrograman Python dan R untuk komputasi ilmiah (ilmu data, aplikasi pembelajaran mesin, pemrosesan data skala besar, analisis prediktif, dll). Yang bertujuan untuk menyederhanakan manajemen paket dan penyebaran.

Pencahayaan

Pencahayaan adalah ukuran dari berapa banyak flux cahaya yang tersebar di daerah tertentu. Seseorang dapat berpikir tentang fluks cahaya (diukur dalam lumen) sebagai ukuran "jumlah" total cahaya yang terlihat, dan pencahayaan sebagai ukuran intensitas pencahayaan pada suatu permukaan. Jumlah cahaya yang menerangi permukaan akan lebih samar-samar jika tersebar di area yang lebih besar, sehingga pencahayaan berbanding terbalik dengan area dimana pancaran cahaya adalah konstan.

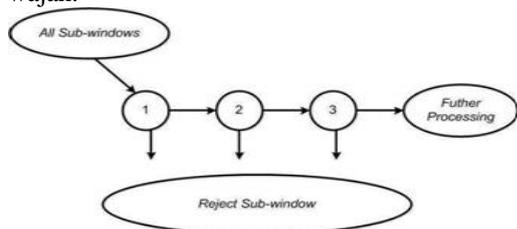
Cara Kerja Deteksi Wajah Viola Jones

Pada gambar berikut ini merupakan skema cara kerja deteksi wajah dengan menggunakan algoritma *Viola-Jones* mulai dari awal proses pendeteksian sampai dengan hasil dari proses pendeteksian.



Cascade Classifier

Karakteristik dari algoritma *Viola-Jones* adalah adanya klasifikasi bertingkat. Klasifikasi pada algoritma ini terdiri dari tiga tingkatan dimana tiap tingkatan mengeluarkan subcitra yang diyakini bukan wajah. Hal ini dilakukan karena lebih mudah untuk menilai subcitra tersebut bukan wajah ketimbang menilai apakah subcitra tersebut berisi wajah.



Pada klasifikasi tingkat pertama, tiap subcitra akan diklasifikasi menggunakan satu fitur. Hasil dari klasifikasi pertama ini berupa T (*True*). Untuk gambar yang memenuhi fitur *Haar* tertentu dan F (*False*) bila tidak. Klasifikasi ini kira-kira akan menyisakan 50% subcitra untuk diklasifikasi di tahap kedua. Hasil dari klasifikasi kedua berupa T (*True*) untuk gambar yang memenuhi proses integral image dan F (*False*) bila tidak. Seiring dengan bertambahnya tingkatan klasifikasi, maka diperlukan syarat yang lebih spesifik sehingga fitur yang digunakan menjadi lebih banyak. Jumlah subcitra yang lolos klasifikasi pun akan berkurang hingga mencapai jumlah sekitar 2%. Hasil dari klasifikasi terakhir berupa T (*True*) untuk gambar yang memenuhi proses *AdaBoost* dan F (*False*) bila tidak. (SG Almag, 2015).

SQLite

SQLite adalah suatu library yang menerapkan mesin database self-contained, serverless, zero-configuration, dan transactional. Self-contained berarti SQLite membutuhkan sedikit sekali dukungan dari library eksternal atau dari sistem operasi. Serverless berarti SQLite dalam mengakses database baik itu read atau write dapat secara langsung dari file database tanpa melalui proses server dan tidak mendukung pengaksesan secara remote (artinya database SQLite bisa dikendalikan dari jarak jauh dengan adanya jaringan computer ("Computer Network"), baik melalui jaringan lokal (intranet) atau internet), dimana kebanyakan mesin SQL database diterapkan sebagai proses server yang terpisah. Zeroconfiguration menunjukkan SQLite tidak membutuhkan instalasi sebelum penggunaannya. Transactional SQLite merupakan suatu transaksional database, dimana dalam melakukan perubahan proses query menerapkan Atomic Consistent, Isoalated, and Durable (ACID). Aplikasi database yang menggunakan SQL server memiliki beberapa kekurangan selain berbayar, yaitu membutuhkan instalasi sebelum menggunakan, membutuhkan server untuk memproses file database dan ukuran memori yang dibutuhkan ketika instalasi lebih besar dibandingkan menggunakan SQL Lite.

2. METODOLOGI PENELITIAN

Pada metodologi penelitian ini dilakukan langkah-langkah dalam cara pengumpulan data, analisis kebutuhan dalam pembuatan program dan implementasi program.

Studi Pustaka

Studi literatur terkait dalam penelitian ini adalah berdasarkan sebuah jurnal dari (Prasetya & Nurviyanto, 2012) mengenai deteksi wajah dengan metode *viola-jones* serta buku bacaan, skripsi dan tesis. Selain itu untuk mendukung data dan informasi dalam perancangan aplikasi maka wawancara juga akan dilakukan. Wawancara dilakukan dengan seorang dosen pada sebuah universitas yang memiliki keahlian sesuai dengan penelitian yang sedang dilakukan

Observasi

Informasi tersebut diperoleh dari berbagai sumber seperti artikel yang berasal dari internet serta pengamatan dari rancangan aplikasi yang sebelumnya telah dilakukan oleh (Prasetya & Nurviyanto, 2012) yang nantinya akan menjadi bukti yang akurat terhadap hasil penelitian.

Perangkat Keras dan Perangkat Lunak

Perangkat Keras

Perangkat keras computer yang digunakan pada penelitian ini adalah :

Prosesor Intel Core i3-6006U Gen 6th
RAM 4GB DDR4
Storage HDD 500GB
Layar 14 inchi (1366x768 px)
Intel HD Graphics 520
VGA Web Camera 0,3 MP

Perangkat Lunak

Python Interpreter ver. 3.6
Library OpenCV
Anaconda
SQLite ver.6.x

Implementasi Sistem

Perancangan sistem deteksi wajah menggunakan metode *viola-jones*, *viola-jones* memiliki algoritma yang dapat digunakan untuk mendeteksi wajah algoritma tersebut adalah *haar cascade classifier* pada algoritma *haarcascade* memiliki fitur yaitu *Adaboost* dan *cascade classifier*. Fitur *adaboost* merupakan fitur yang bertujuan untuk membentuk template wajah, sedangkan *cascade classifier* digunakan untuk menyeleksi citra agar citra yang dihasilkan tidak terlalu banyak, citra yang telah berhasil diseleksi digabungkan menjadi satu-kesatuan file berformat *.yml* hingga pada akhirnya sistem deteksi wajah dapat mendeteksi wajah serta dapat mengidentifikasi identitas dari pemilik wajah. Berikut ini merupakan penggalan *source code* dari penerapan fitur *haarcascade* dalam mendeteksi wajah.

```
import numpy as np
import cv2

detector =
cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

cap = cv2.VideoCapture(0)

while (True):

ret, img = cap.read()

gray = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY) faces =
detector.detectMultiScale(gray, 1.3, 5) for
(x, y, w, h) in faces:

cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0,
0), 2)

cv2.imshow('Face', img)
if cv2.waitKey(1) & 0xFF == ord('q'):
break

cap.release()
cv2.destroyAllWindows()
```

Source code tersebut akan menampilkan suatu wajah yang sudah terdeteksi.

Penjelasan lebih lanjut mengenai cara kerja sistem deteksi wajah hingga dapat mengidentifikasi identitas akan dijelaskan pada bagian 4.

Pengujian Sistem

Dalam perancangan sistem deteksi wajah menggunakan 8 sampel wajah masing-masing wajah tersebut membutuhkan 30 foto. Foto-foto tersebut nantinya akan diekstraksi menggunakan fitur *haarcascade* hingga diperoleh satu file yang memiliki format *.yml*. File tersebut merupakan gabungan-gabungan foto yang telah diekstraksi dan diklasifikasi. Program ini menggunakan metode *black box* dimana pengguna dapat dengan mudah memilih menu dan mendapatkan *output* karena program ini dirancang untuk mudah digunakan. Untuk mengetahui lebih lanjut mengenai penerapan sistem deteksi wajah pada program akan dipaparkan pada bagian 4.

Diagram Perancangan Sistem Deteksi Wajah

Pada gambar perancangan sistem aplikasi deteksi wajah di bawah ini hal pertama dilakukan yaitu memahami mengenai konsep dan bagaimana cara kerja deteksi wajah konsep dan cara tersebut dapat diketahui melalui penelitian yang sebelumnya telah dilakukan. Kemudian setelah memahami konsep mengenai deteksi wajah tahap selanjutnya menganalisis kebutuhan apa saja yang nantinya akan dibutuhkan dalam merancang aplikasi deteksi wajah seperti kebutuhan *hardware* dan *software*.

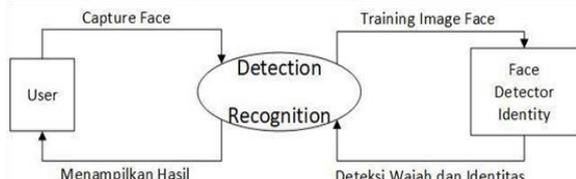
Selanjutnya menentukan algoritma apa yang akan digunakan dalam perancangan aplikasi pada penelitian ini menggunakan algoritma dan metode *viola-jones*. Kemudian mencari tools atau media yang akan digunakan dalam perancangan aplikasi untuk penelitian ini menggunakan *python* dan *openCV* sebagai media perancangan aplikasi deteksi wajah. Kemudian melakukan analisis anatara algoritma *viola-jones* dengan *tools python* dan *openCV* analisis ini dilakukan untuk perancangan dan implementasi pembangunan aplikasi.

Tahap selanjutnya menentukan model pendeteksian yang sesuai dengan *tools* tersebut. Kemudian setelah menemukan model pendeteksian yang sesuai selanjutnya menentukan pengguna yang fotonya akan digunakan untuk percobaan pendeteksian. Kemudian Setelah mendapat pengguna yang fotonya akan di deteksi tahap selanjutnya menganalisa algoritma *viola-jones* apakah dapat digunakan serta sesuai dengan tools dan model pendeteksian. Tahap terakhir setelah semua kebutuhan perancangan aplikasi terpenuhi maka selanjutnya dapat memulai membuat aplikasi deteksi wajah.



Data Flow Diagram (DFD) Aplikasi Deteksi Wajah

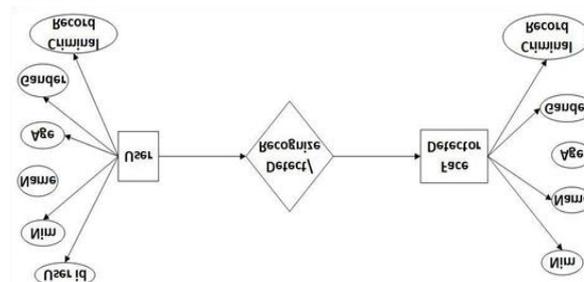
Data Flow Diagram dibawah ini menjelaskan bahwa tahap awal sebelum memulai proses deteksi user melakukan *capture* wajah terlebih dahulu kemudian data tersebut akan disimpan ke dalam folder dataset yang kemudian data tersebut diambil untuk dilakukan proses selanjutnya yaitu *training* wajah, hasil *capture* wajah diekstrasi menjadi *greyscale* yang kemudian disatukan kedalam satu file berformat *.yml* file tersebut disimpan kedalam folder *trainer*. Kemudian file *yml* tersebut akan di ekstrasi dan di olah, selain itu *detector* akan menginput file database ke dalam sistem hingga menghasilkan *output* berupa wajah yang berhasil di kanali beserta identitasnya, kemudian menampilkan output tersebut kepada user.



Entity Relationship Diagram (ERD) Aplikasi Deteksi Wajah

Entity Relationship Diagram di bawah ini menunjukan data yang terdapat pada database SQLite. Tabel pada user memiliki hubungan dengan *face detector* karena database pada tabel user akan diinput pada sistem *detect* atau *recognize* dan

selanjutnya data tersebut akan ditampilkan pada *face detector*.



Struktur Database

Berikut ini adalah struktur dari *database* sistem deteksi wajah untuk mengetahui identitas dari pemilik wajah. Tabel wajah memiliki 5 *field*, tabel ini berfungsi untuk menampilkan identitas dari pemilik wajah yang data identitasnya terdapat pada database.

Field	Type (Length)
Id	Varchar (50)
NIM	Varchar (50)
Name	Varchar (100)
Age	Varchar (4)
Gander	Varchar (20)
Criminal Record	Varchar (100)

Rancangan Tampilan Aplikasi Deteksi Wajah

Tampilan berupa gambar interface atau antar muka pada sistem aplikasi deteksi wajah yang berisi tampilan yang menampilkan beberapa output. Yang nantinya akan didesain sesuai dengan kebutuhan aplikasi sehingga dapat mempermudah pengguna dalam menjalankan aplikasi ini, berikut rancangan tampilan aplikasi sistem deteksi wajah.



Pada tampilan tersebut terdapat beberapa menu widget, pembuatan rancangan tampilan antar muka ini menggunakan library tkinter. Dalam tkinter telah menyediakan sekitar 14 widget dasar.

Dalam pembuatan tampilan antar muka ini menggunakan beberapa widget dasar diantaranya

Widget Perancangan Tampilan Antar Muka	
Widget	Deskripsi
Text	Memformat tampilan teks, menyarankan anda untuk menampilkan dan mengedit teks dengan gaya dan atribut, juga mendukung pemasangan image dan window
Label	Menampilkan teks atau gambar
Entry	Field untuk memasukan teks
Button	Tombol sederhana, digunakan untuk mengeksekusi suatu perintah atau operasi lainnya

3. HASIL DAN PEMBAHASAN

Pembahasan berikut adalah uraian setiap hasil tampilan aplikasi dan pengujian. Langkah demi langkah menjalankan aplikasi hingga deteksi wajah dapat memperoleh informasi mengenai identitas dari pemilik wajah.

Membuat Dataset Wajah

Tahap pertama yang harus dilakukan adalah membuat dataset pada tahap ini *user* diminta untuk memasukan id dengan cara memasukan angka pada *text filed* pada aplikasi kemudian tekan tombol *capture face*, wajah *user* akan dicapture dan secara otomatis diberi nama sesuai dengan id yang sebelumnya telah diinput dan file hasil *capture* tersebut akan tersimpan di dalam folder dataset sistem, hasil dari *capture* tersebut nantinya akan di ekstrasi menjadi citra *greyscale* hingga dapat digunakan pada proses selanjutnya.

Training Wajah

Tahap selanjutnya setiap citra wajah yang telah dicapture akan diekstrasi fitur menggunakan haar-like filter kemudian citra discan per-sub window dimulai dari kiri atas dengan ukuran 24x24 akan di sliding dari kiri ke kanan hingga dari atas ke bawah. Citra yang sebelumnya *greyscale* memiliki rentang nilai dari 0 sampai 255 akan dikonversi menggunakan LBPH (*Lokal Binary Pattern Histogram*) agar dapat membedakan area *black* dan area *white*, area *black* dimulai dari nilai 0-115 dan area *white* dimulai dari 155-255.

Kemudian jika citra sudah dapat dibedakan letak area *black* and *white* nya tahap selanjutnya menghitung fitur tersebut dengan cara menjumlahkan semua nilai *intensity pixel* pada area *white* dan area *black* hingga hasilnya dimasukan ke dalam rumus fitur yaitu Jumlah nilai *pixel* di semua area *white* – Jumlah nilai *pixel* di semua area *black*.

Pada basis ukuran 24 x 24 akan ada sekitar 160.000 kemungkinan bentuk *haar-like feature*, akan tetapi untuk mendeteksi wajah tidak diperlukan terlalu banyak fitur. *Haar-like feature* akan menggunakan 12 ukuran basis wajah dimulai ukuran dari ukuran 24 x 24 kemudian ukuran selanjutnya akan lebih besar dengan skala perbesaran 1,25. 12 ukuran yang berbeda tersebut akan di sliding dari

atas kebawah dan dari kiri ke kanan hingga pada area *white* dan area *black* akan dihitung menggunakan rumus fitur Jumlah nilai *pixel* di semua area *white* – Jumlah nilai *pixel* di semua area *black*. Hasil dari semua perhitungan tersebut dirata-ratakan yang kemudian hasil dari nilai rata-rata akan dijadikan nilai ambang batas atau *threshold*. Nilai nilai tersebut akan dikonversi kedalam sebuah file yang berformat *.yml*. Kemudian file tersebut disimpan ke dalam folder *training*, jika aplikasi selesai melakukan training akan muncul notifikasi "Image Trained. Dibawah ini merupakan tampilan dari training wajah.

Deteksi Wajah Dengan Identitas

Kemudian tahap selanjutnya melakukan pendeteksian, pada tahap ini user menekan tombol *detect face* sehingga webcam akan muncul dan melakukan proses penyesuaian wajah pada webcam dengan file yang berformat *yml*. File tersebut berisikan nilai yang digunakan untuk menyesuaikan wajah dengan identitas dari pemilik wajah, proses penyesuaian identitas tersebut dilakukan berdasarkan nomer id yang sebelumnya *user* telah masukan ke dalam database sistem, database tersebut berisikan identitas seseorang seperti id, nim, *name*, *age*, *gander*, dan *criminal records* data-data tersebut miliki tipe data *varchar* tipe data tersebut harus disamakan pada setiap data karena *source code* yang diterapkan pada program tidak dapat menggunakan tipe data *integer*.

Pada gambar di bawah ini merupakan tampilan interface aplikasi beserta output yang dikeluarkan.





Dibawah berikut adalah file-file dataset wajah yang ditempatkan pada satu folder.



Sedangkan struktur database pada DBMS SQLite seperti terlihat pada gambar dibawah ini.



Hasil Uji Coba

Pada uji coba tersebut terdapat 8 sampel wajah yang telah dilakukan uji coba dari tabel uji coba tersebut dapat disimpulkan semua sampel wajah dapat dideteksi dengan tingkat ke akurasian 100%. Wajah dapat diketahui identitas dari pemilik wajah sesuai data pada database.

Uji coba berikutnya adalah mengenali satu wajah yang tidak dapat diketahui identitas dari pemilik wajah hal tersebut disebabkan karena identitas dari pemilik wajah tersebut tidak terdapat di dalam database sistem sehingga sistem tidak menemukan catatan data pada database.

4. SIMPULAN

Berdasarkan hasil implementasi dan pengujian yang dilakukan terhadap sistem yang dibuat maka dapat disimpulkan bahwa :

1. Sistem dapat mendeteksi wajah manusia dengan batasan kecerahan cahaya antara ±50 lux untuk nilai maksimal gelap dan ±200 lux nilai maksimal terang.
2. Sistem dapat mendeteksi dalam keadaan langsung dengan akurasi 100%.
3. Sistem dapat mendeteksi identitas pemilik wajah.
4. Sistem dapat mendeteksi wajah yang teridentifikasi sesuai data pada database.
5. Sistem dapat mendeteksi wajah yang terdapat dalam *capture*.
6. Tidak dapat mendeteksi lebih dari 50 cm.

Saran

1. Menyempurnakan tampilan agar lebih mudah digunakan.
2. Mengembangkan program agar dapat terkoneksi langsung dengan catatan kriminal dari pihak berwajib.
3. Membuat program yang dapat diterapkan kedalam kamera pemantau.
4. Membuat rentang jarak yang lebih jauh dan pencahayaan yang lebih baik.
5. Meningkatkan tingkat akurasi yang disesuaikan dengan jauhnya jarak dan pencahayaan.

DAFTAR PUSTAKA

- [1] Bahasa, I. (2019). Pengertian Nama Dalam Ilmu Bahasa. Retrieved from <https://www.ilmubahasa.net/2015/02/pengertian-nama-dalam-ilmu-bahasa.html>.
- [2] Code, N. E., Society, I. E., Commission, I. E., & Standard, A. (2015). Tata cara perancangan sistem pencahayaan buatan pada bangunan gedung ., 1–32.
- [3] Dini, T. (2015). Konsep dasar python. *Konsep Dasar Python*, 1–6,
- [4] FrescoBot. (2019). Anaconda (distribusi Python). Retrieved from [https://en.wikipedia.org/wiki/Anaconda_\(Python_distribution\)](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)).
- [5] ITS, B. A. B., & Penunjang, T. (2015). kamera digital, dan. *BAB II Penunjang Teori*, 7–23.
- [6] Library binus. (2011). Computer vision, 7–8.
- [7] Muthi'ah, M. (2015). Rancang Bangun Database dan Migrasi Data Melalui teknik ORM untuk Aplikasi Dashboard Pemantauan Mytra.
- [8] Putro, M. D., Adji, T. B., & Winduratna, B. (2012). Sistem Deteksi Wajah dengan Menggunakan Metode Viola-Jones. *Tugas Akhir*, 1– 5.
- [9] rizafennisya. (2017). Pengolahan Citra Digital. Retrieved from <https://rizafennisya.wordpress.com/2017/01/19/definisi-pengolahan-citra-digital/>.
- [10] Setiyadi, A., & Harihayati, T. (2015). Penerapan SQLite Pada Aplikasi Pengaturan Waktu Ujian Dan Presentasi, *12(2)*, 221-226.
- [11] SG Almaga. (2015). perancangan aplikasi deteksi wajah menggunakan algoritma viola-jones, 302. Retrieved from repository.unpas.ac.id.
- [12] Unpas, R. (2017). Eksplorasi Antarmuka Grafis Pemakaian Tkinter Pada Lingkungan Bahasa Python. *Tkinter*, 1-26.