

## Penerapan *Haar Cascade Classifier* dalam Pengenalan Pola Bentuk Wajah Menggunakan *OpenCV*

Citha Ahmad Pauzi<sup>1</sup>, Abdul Basit Yahya<sup>2</sup>, Ferly Taku Wildan<sup>3</sup>, Anwar Hidayatulloh<sup>4</sup>, Perani Rosyani.<sup>5</sup>

Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

Email : [1cithaahmadpauzi123@gmail.com](mailto:1cithaahmadpauzi123@gmail.com) , [2abdulbasityh@gmail.com](mailto:2abdulbasityh@gmail.com) , [3ferlywildan018@gmail.com](mailto:3ferlywildan018@gmail.com) ,  
[4anwarsmam@gmail.com](mailto:4anwarsmam@gmail.com) , [5dosen00837@unpam.ac.id](mailto:5dosen00837@unpam.ac.id)

**Abstrak** - Dalam penelitian kami ini menggunakan modul *OpenCV* bahasa pemrograman *Python* untuk mengenali wajah manusia. *Hair cascade* merupakan salah satu cara mengenali bentuk mata dan wajah. Tahap pertama akan digunakan data wajah dan mata open source *Intel*. Data ini kemudian digabungkan dengan modul *Cascade Classics OpenCV* untuk mengubah data ini menjadi pengenalan bentuk wajah berdasarkan titik-titik pada wajah yang cocok dengan data yang disediakan. Banyak pengenalan sistem wajah menggunakan teknik visi komputer sebagai metode pendeteksian objek. Metode *computer vision* dianggap cepat dan akurat karena digabungkannya beberapa konsep (*fitur Haar*, gambar *integral*, *AdaBoost*, *Cascade Classifier*) ke dalam satu metode pendeteksian objek utama. Banyak dari sistem pengenalan ini menggunakan *C* atau *C++* sebagai bahasa pemrogramannya dan *OpenCV* sebagai perpustakaan pengenalan objek. Hal ini dikarenakan perpustakaan *OpenCV* di terapkannya teknik *computer vision* pada sistem pendeteksian sehingga memudahkan untuk membangun sistem. Penelitian ini kami bertujuan untuk meng-implementasikan *computer vision* pada sistem pengenalan wajah sederhana dengan memanfaatkan *library OpenCV* dan menggunakan bahasa pemrograman *Python* sebagai dasar sistemnya.

**Kata Kunci:** *Python*, Pengenalan Pola Wajah, *OpenCV*, *Cascade classifier*, *AdaBoost*

*Abstract* - In this research we use the *OpenCV* module of the *Python* programming language to recognize human faces. *Hair cascade* is one way to recognize the shape of the eyes and face. The first stage will use *Intel's* open source facial and eye data. This data is then combined with *OpenCV's* *Cascade Classics* module to convert this data into facial shape recognition based on points on the face that match the provided data. Many facial recognition systems use computer vision techniques as an object detection method. The computer vision method is considered fast and accurate because it combines several concepts (*Haar* features, *integral image*, *AdaBoost*, *Cascade Classifier*) into one main object detection method. Many of these recognition systems use *C* or *C++* as their programming language and *OpenCV* as an object recognition library. This is because the *OpenCV* library applies computer vision techniques to its detection system, making it easier to build the system. This research aims to implement computer vision in a simple facial recognition system by utilizing the *OpenCV* library and using the *Python* programming language as the basis of the system.

*Keywords:* *Python*, *Facial Pattern Recognition*, *OpenCV*, *Cascade classifier*, *AdaBoost*

### 1. PENDAHULUAN

*Computer vision* (CV) banyak digunakan di berbagai bidang seperti stasiun kereta api, bandara, terminal, dan sistem keamanan untuk mengenali bentuk wajah manusia dan mendeteksi potensi kejahatan. Teknologi visi komputer, yang dapat mengikuti kemampuan otak manusia dan mata dalam melihat dan mengidentifikasi bentuk benda, sangat penting bagi perkembangan teknologi informasi.

*Computer vision* (CV) yang dipadukan dengan pembelajaran mesin memungkinkan kita memahami dan meniru sifat manusia dengan lebih baik. Pengembangan visi komputer memanfaatkan secara ekstensif modul sumber terbuka yang tersedia, seperti *OpenCV Intel*, yang tersedia dalam bahasa pemrograman *Python* dan *C++*.

Pengenalan wajah dan mata dalam *computer vision* menggunakan data *Haar Cascades* yang

menyediakan *koordinat* (x,y) untuk mengidentifikasi wajah dan mata berdasarkan *piksel* pada kamera. Perhitungan numerik meng-gunakan modul *NumPyt Python* digunakan untuk menghitung *koordinatt* wajah dan mata yang tepat.

**2. METODE PENELITIAN**

Perani Rosyani (2017) menyatakan bahwa pengenalan wajah adalah metode pengenalan yang ber-orientasi pada wajah. Wajah merupakan salah satu karakteristik *biometrik* yang digunakan untuk mengenali seorang selain karakteristik yang lain seperti ucapan, sidik jari, mata/retina, dll. Wajah adalah struktur multi-dimesi yang sangat kompleks dan membutuhkan teknik komputasi yang baik untuk pengenalan. Karakteristik organ kemudian direpresentasikan dalam bentuk *vektor* dan ditentukan menggunakan analisis komponen yang paling tepat, yang berupaya menghitung model terbaik yang menentukan informasi paling *relevan* untuk menggambarkan bentuk wajah.

Oleh karena itu, penelitian mengenai pengenalan wajah berkembang pesat. Meskipun banyak aplikasi *komersial* telah diterapkan, teknologi ini pada dasarnya belum sempurna. Penelitian lebih lanjut diperlukan untuk mencapai hasil yang diinginkan. Salah satu hal yang bisa ditambahkan atau diterapkan seiring berkembangnya teknolgi pengenalan wajah adalah meningkatkan kecepatan dan keakuratan pengenalan wajah. Banyak dari sistem pengenalan ini menggunakan teknik *visi* komputer sebagai metode pendeteksian objek. *Computer vision* dikenal sangat cepat dan akurat karena menggabungkan beberapa konsep (fitur *Haar*, *Integral Image*, *Cascade Classifier*, *AdaBoost*) ke dalam satu metode pendeteksian objek utama.

**2.1 Metode Penelitian**

Aura Amalia Warzuqni, Divia Putri Sabilla, Zara Agustin, Perani Rosyani (2022/9/30:p.16) menyatakan bahwa *Haar cascade classifier* atau yang di-kenal dengan nama lain *haar-like features* merupakan *rectangular features* (fungsi persegi), yang mem-berikan *indikasi* secara *spesifik* pada sebuah gambar atau gambar/*image*. Prinsip *haar-like features* adalah mengenali objek berdasarkan nilai sederhana fitur, tetapi tidak pada nilai *piksell* citra objek. Metode ini memiliki kelebihan yaitu komputasi nya sangat cepat, karena tidak memerlukan perhiitungan setiap nilai *piksell* dalam suatu citra. Untuk menerapkan algoritma *Haar Cascade Classifier* yaitu menggunakan *OpenCV* untuk memungkinkan sistem mengenali wajah dalam gambar *digital*. Data masukannya berupa data gambar berwarna yang jumlah objek wajahnya berubah. Gambar berwarna pertama- tama diubah menjadi gambar skala abu-abu, dan kemudian objek dengan wajah dikenali. Gambar 1 menunjukkan metode penelitian ini.



**Gambar 1. Metodologi Penelitian**

Algoritma *Haar Cascade Classifier* digunakan karna memiliki tingkat akurat yang sangat tinggi dan dapat *mengklasifikasikan* gambar *digital* dengan cepat. Tahap-tahap yang dilaksanakan pada riset ini antara lain melaksanakan beberapa kajian *literaturr* terkait algoritma *Haar Cascade Classifier* dan pendukung teori pengolahan *citra digital*. Teknik pengolahan citra yang digunakan pada penelitiin ini

adalah konversi citra menjadi *citra greyscale*.

Data citra kemudian dikumpulkan untuk dijadikan citra masukan berupa citra berwarna yang bervariasi ukuran citra dan posisi wajah di dalam citra. Data gambar ini diambil dari ukuran wajah pada gambar, posisi wajah, jarak ke objek wajah, ukurannya, dan lain-lain.

Langkah selanjutnya adalah *mengimpor* perpustakaan *OpenCV* menggunakan *OpenCV versi 3.4* dan bahasa pemrograman *Java* untuk mengembangkan sistem pengenalan wajah. Implementasi fungsi *Haar* diambil dari perpustakaan *OpenCV* yaitu *haarcascade\_frontalface\_alt2.xml* dan di-gunakan untuk mengklasifikasikan wajah pada gambar masukan. Pustaka yang di-gunakan untuk meng-ubah gambar berwarna menjadigambar abu-abu disebut *ImgCodecs.CV\_LOAD\_IMAGE\_GRAYSCALE*.

Langkah terakhir dalam penelitian ini adalah menganalisis dan menguji gambar masukan. Setelah sistem berhasil dibangun, data gambar masukan diolah untuk mendapatkan hasil pengenalan wajah yang akurat.

## 2.2 Landasan Teori

### 2.2.1 OpenCV dan Python

*Open Computer Vision (OpenCV)* adalah perpustakaan pemrosesan gambar sumber terbuka. Tujuannya agar komputer memiliki kemampuan pemrosesan *visual* yang mirip dengan manusia. *OpenCV* telah menyimpan banyak algoritma *visi* komputer dasar, termasuk modul deteksi objek yang menggunakan teknik *visi* komputer. Seperti yang dikutip dari artikel Rosyani (2019:41), "*OpenCV* menyediakan perpustakaan algoritma *Haar Cascade* terlatih yang dikategorikan menjadi wajah, mata, dll. Algoritma ini dapat mendeteksi wajah pada gambar *digital* menggunakan fungsi *cascade* yang dilatih pada data gambar *positif* dan *negatif*."

*Python* adalah bahasa pemrograman tingkat tinggi yang dapat ditafsirkan, *interaktif*, *berorientasi* objek, dan berjalan di hampir semua *platform* (*Mac, Linux, dan Windows*). *Python* adalah bahasa pemrograman yang mudah di pelajari karena sintaksisnya yang jelas, di-kombinasikan dengan penggunaan modul bawaan dan *struktur* data tingkat tinggi yang *efisien*.

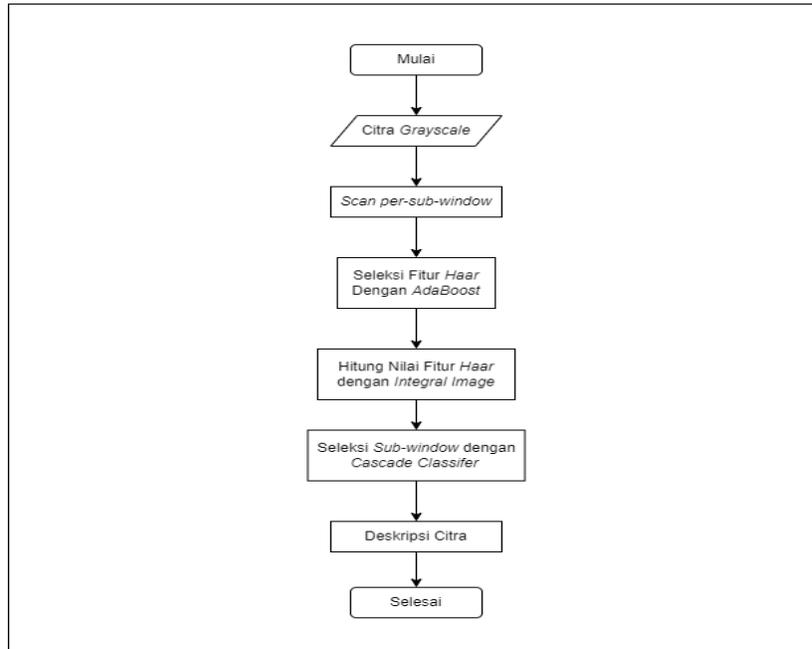
Untuk membangun penelitian ini, di-gunakan *wxPython*, *toolkit GUI* untuk bahasa pemrograman *Python*. *wxPython* memungkinkan pemrogram *Python* membangun aplikasi dengan fondasi yang kuat dan antar-muka pengguna *grafis* yang kaya, sederhana, dan mudah dipahami. Kami juga menggunakan *Boa Constructor*, lingkungan pengembangan *terintegrasi (IDE)* untuk *Python* dan *wxPython GUI Builder* lintas *platform*. *Boa Constructor*

memungkinkan Anda membuat dan mengubah bingkai secara visual/tanpa skrip dan menyediakan banyak pemeriksa objek, termasuk *browser* objek, *hierarki* warisan, *debugger* tingkat lanjut, dan bantuan bawaan.

### 2.2.2 Pengolahan Citra

Proses pengolahan citra yang pertama adalah mengubah citra *non digital* menjadi citra *digital*. "Dalam pengolahan citra wajah ada beberapa tahapan seperti pre-processing gambar, pengolahan warna gambar, sehingga deteksi objek wajah yang akan dikenali oleh mesin." (P Rosyani, Retnawati, 2023). *Digitalisasi* gambar melibatkan dua proses, pengambilan sampel dan kuantisasi. Sampling mengacu pada jumlah *piksel/blok* yang mendefinisikan gambar, sedangkan kuantisasi mengacu pada urutan nilai setiap *piksel* (mengacu pada jumlah *bit* dalam gambar *digital*, misalnya *2-bit* hitam/putih, *8-bit skala* abu-abu). Warna Asli *24-bit*. Pengolahan gambar Hal ini di-lakukan untuk meningkatkan kualitas gambar agar mudah *diinterpretasikan* oleh manusia dan komputer. Masukan dalam pengolahan *citra* adalah berupa gambar, dan keluarannya juga berupa gambar, namun dengankualitas yang lebih tinggi dari pada gambar masukan. Operasi pemrosesan citra yang terkait dengan pengenalan wajah meliputi skala abu-abu, operasi lingkungan, ambang batas, pemerataan *histogram*, dan perubahan ukuran.

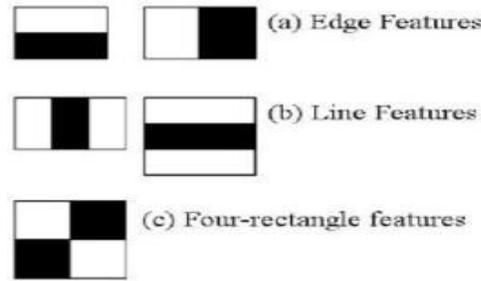
Proses analisis gambar diawali dengan pencitraan dan diakhiri dengan proses akhir penentuan tujuan atau tujuan tertentu, misalnya pengendalian *robot*. Penelitian ini fokus pada *identifikasi* wajah manusia pada latar belakang menggunakan metode *Haar Cascade Classify*.



**Gambar 2. Diagram Pengelohan Citra**

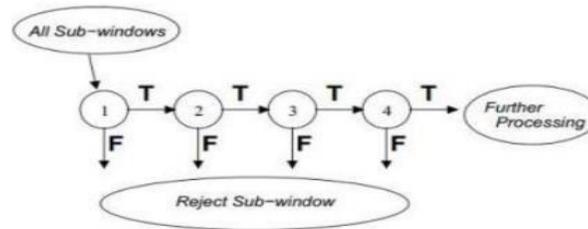
### 2.2.3 Haar Cascade Clasifier

*Haar Cascade* adalah algoritma pelatihan mesin untuk pengenalan objek yang diusulkan oleh *Paul Viola dan Michael Jones* pada tahun 2001. Algoritmenya merupakan pendekatan berbasis pembelajaran mesin yang menggunakan fungsi *cascade*, yang dilatih dari berbagai gambar *positif* dan *negatif*. Citra *positif* adalah citra yang memiliki target pengenalan, dan citra *negatif* adalah citra tanpa target pengenalan. Oleh karena itu, Anda dapat menggunakan fitur ini untuk mendeteksi objek pada gambar lain. Saat ini, *OpenCV* menyediakan perpustakaan algoritma *Haar Cascade* terlatih yang diklasifikasikan ke dalam beberapa katagori seperti wajah, retina/mata, dll. bergantung pada gambar yang di-latih. *Haar Cascade* menggunakan "*filter*" yang mirip dengan konsep *kernel konvolusi* untuk *mengekstrak* fitur dari gambar. Seperti yang dikutip dari artikel Rosyani (2021:35), "*Haar Cascade Classifier* merupakan algoritma untuk mengenali objek pada citra *digital*, khususnya untuk pengenalan wajah. Metode ini dapat mendeteksi pola bentuk wajah dengan akurasi yang relatif tinggi." Lalu *filter* ini disebut fitur *Haar* dan terlihat seperti Gambar 3.



**Gambar 3. Fitur Haar**

Filter ini mengamati satu bagian dalam satu waktu. Kemudian, untuk setiap bagian, semua intensitas piksel di bagian hitam putih dijumlahkan, dan selisih antara jumlah masing-masing dihitung. Nilai ini menjadi kuantitas fitur yang diekstraksi. Keunikan algoritma ini yaitu memiliki klasifikasi bertingkat. Klasifikasi ini terdiri dari beberapa lapisan yang masing-masing menghasilkan sub-gambar yang diasumsikan bukan wajah. Gambar 4 menunjukkan alur kerja klasifikasi bertingkat

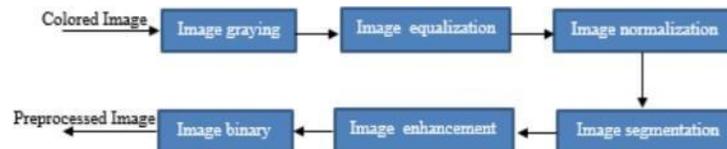


**Gambar 4. Klasifikasi Bertingkat Fitur Haar**

Pada tingkat klasifikasi pertama, setiap sub citra diklasifikasikan berdasarkan ciri- cirinya. Hasil klasifikasi awal ini adalah *T (True)* jika citra memenuhi sifat Haar tertentu, dan *F (False)* jika tidak. Klasifikasi ini membutuhkan sekitar 50% sub-gambar untuk diklasifikasikan pada tahap kedua. Hasil dari klasifikasi kedua adalah *T (True)* untuk citra yang memenuhi pengolahan citra integral dan *F (False)* untuk citra yang tidak. Seiring dengan meningkatnya tingkat klasifikasi, dibutuhkan persyaratan yang lebih spesifik, oleh karena itu lebih banyak fitur yang digunakan dalam Menerapkan Haar Cascade Classifier pada Aplikasi Pola Bentuk Wajah Menggunakan OpenCV. Jumlah sub-gambar yang lolos klasifikasi dikurangi menjadi sekitar 2%.

**2.2.4 Preprocessing Citra pada Pengenalan Pola Wajah**

Pemrosesan citra (*preprocessing*) sangat penting dalam pengenalan wajah. *Preprocessing* citra pada pengenalan wajah memiliki beberapa tahapan. Salah satunya adalah mengubah gambar berwarna menjadi gambar abu-abu (*image greying*) untuk mengurangi informasi yang tidak perlu.



**Gambar 5. Tahapan Preprocessing Citra Pengenalan Wajah**

Pada Gambar 5, tahap *preprocessing* citra dimulai dengan beberapa gabungan data citra berwarna yang diganti menjadi citra abu-abu (*image greying*). Kemudian dilanjutkan dengan beberapa

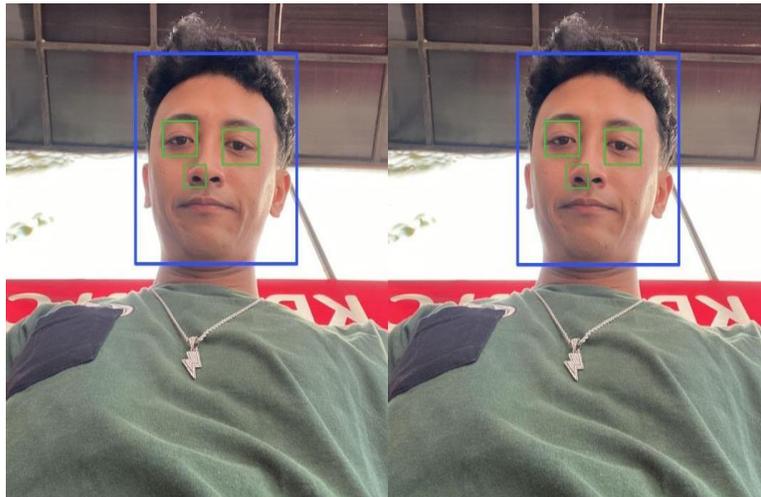
jalanan pemrosesan seperti pemerataan gambar , normalisasi gambar, *segmentasi* gambar, dan *binarisasi* gambar untuk meningkatkan kualitas gambar.

### 3. HASIL DAN PEMBAHASAN

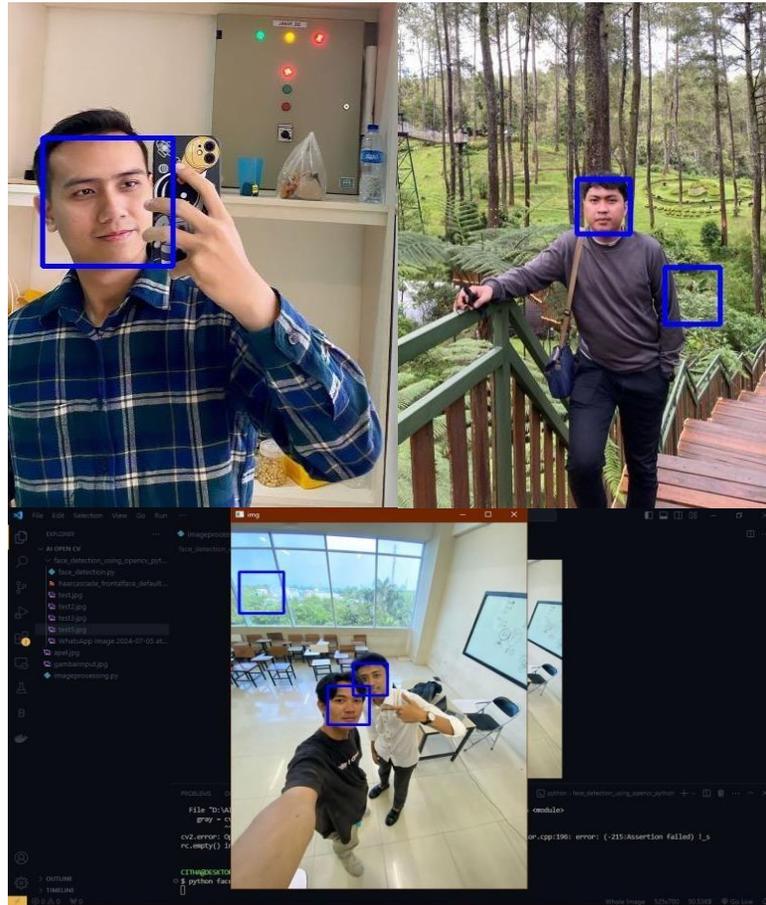
Pada penelitian ini kami mengembangkan perangkat lunak pengenalan gambar wajah yang menggunakan modul *OpenCV* dengan menggunakan bahasa pemrograman *Python*. Pengenalan wajah menggunakan data wajah sumber terbuka *Intel*. Data yang tersedia memfasilitasi pengenalan pola wajah dan dikombinasikan dengan *OpenCV* untuk proses pengenalan pada pola dan *klasifikasi*. Oleh karena itu, pada penelitian ini kami mengambil sampel data satu atau dua wajah secara bersamaan menggunakan *webcam* dalam kondisi cahaya terang, berkedip, dan menggunakan kacamata. Aplikasi dapat mengenali wajah berdasarkan wajah yang terdeteksi dan mata persegi berwarna-warni.

Selama tahap desain, kami merancang sistem pengenalan gambar wajah menggunakan teknik *klasifikasi OpenCV*. Algoritma *OpenCV* mengidentifikasi data wajah yang disediakan dan melakukan pengenalan wajah. *Klasifikasi* citra didasarkan pada nilai fitur sederhana dan menggunakan tiga, fitur persegi, fitur 3 persegi, dan fitur 4 persegi. Nilai dari fitur ini adalah perbedaan antara area hitam dan putih di dalam *subjendela* gambar.

Jumlah total fitur *Haar* di setiap gambar subjendela sangat besar, jauh lebih besar daripada jumlah *pixels*nya. Untuk dapat melakukan klasifikasi dengan cepat, proses pembelajaran harus menghilangkan sebagian besar fitur yang tersedia dan fokus pada sekelompok kecil fitur yang diperlukan. Buat templat wajah yang *efektif* menggunakan algoritma *AdaBoost*



Gambar 6. Pengenalan Pola Wajah dengan *OpenCV*

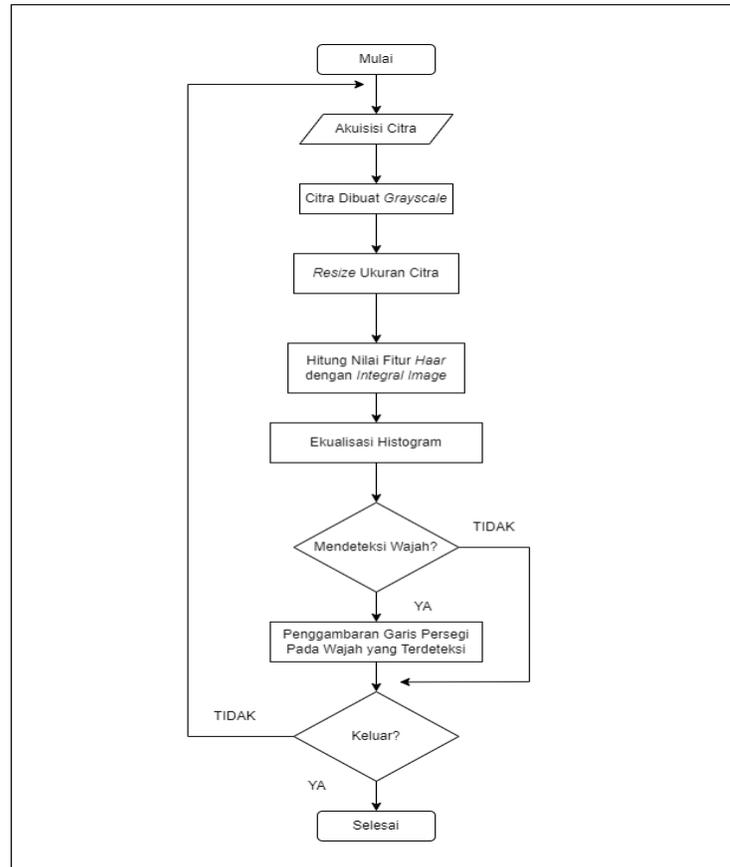


**Gambar 7. Deteksi Pola Wajah Pada Citra**

Metode *klasifikasi* yang digunakan pada penelitian ini menerapkan tahapan seleksiganda, dan pada setiap tahapan dilakukan proses seleksi menggunakan algoritma *AdaBoost* yang dilatih menggunakan fungsi *Haar*. Proses seleksi ini membantu memisahkan *subjendela* yang berisi objek *positif* (objek yang diinginkan ada pada gambar yang ter- deteksi) dari objek *negatif* (objek yang diinginkan tidakada pada gambarr yang ter-deteksi).

Alur kerja penelitian yang terstruktur diperlukan agar pekerjaan dapat dilakukan secara berkesinambungan dan berkesinambungan tanpa meng-ganggu jenis pekerjaan lainnya. Persiapan mencakup segala sesuatu yang berkaitan dengan proses desain, seperti mem-pelajari, dan memahami cara kerja algoritma, merancang algoritma dan diagram alur, membangun sistem, dan melakukan *analisis* sistem.

Sistem yang kami kembangkan mengacu pada *flowchart* pada Gambar 8, dan setelah *dieksekusi*, sistem mengaktifkan *camera web* untuk mengambil gambar. Pemrosesan gambar seperti skala abu-abu, pengubahan ukuran, dan pemerataan kemudian dilakukan. Selanjutnya, mendeteksi wajah dan setelah terdeteksi, menggambar garis persegi pada wajah.



Gambar 8. Diagram Alir Sistem Deteksi Wajah

Sistem yang dikembangkan melakukan pengambilan wajah dengan memindai seluruh area gambar masukan menggunakan *sub-jendela* minimal berukuran  $20 \times 20$  piksel. Proses ini diulang berulang kali pada faktor *skala* 1,1 hingga tercapai bagian kanan bawah gambar. *Ekstraksi* fitur *Haar* diterapkan pada setiap *subjendela* yang dipindai, dan pemilihan fitur dilakukan menggunakan algoritma *AdaBoost* untuk mengidentifikasi fitur yang paling relevan.

Karena terdapat banyak *subjendela* pada gambar, pemilihan *subjendela* dilakukan menggunakan *Cascade Classifier* (templat). Hanya *subjendela* yang melewati semua tingkat pemilihan pengklasifikasi yang digambarkan sebagai permukaan.

Pengujian sistem dilakukan menggunakan *notebook* dengan *webcam 1,3 MP* di ruangan dengan pencahayaan alami. *Skenario* pengujian yang dilakukan meliputi perubahan pencahayaan, jarak, kemiringan wajah, wajah yang disembunyikan oleh objek lain, deteksi wajah ganda, dan objek mirip wajah.

Hasil pengujian menunjukkan bahwa sistem dapat mengenali wajah pada gambar wajah bagian depan dengan akurasi 100%. Wajah yang terdeteksi ditampilkan dengan penanda persegi panjang di sekelilingnya. Berbagai fitur wajah yang mungkin dikenali atau tidak oleh sistem juga diuji.

## 4. KESIMPULAN DAN SARAN

### Kesimpulan:

Sistem pengenalan pola wajah berbasis *Haar Cascade Classifier* menggunakan *OpenCV* menunjukkan kinerja yang sangat baik. Sistem mengenali wajah manusia bagian depan dengan akurasi 100% dan kecepatan pengenalan kurang dari 0,5 detik. Sistem juga dapat mendeteksi wajah pada posisi *non-frontal* hingga  $\pm 71^\circ$  dan mendeteksi keberadaan beberapa wajah dalam sebuah gambar. Selain itu, sistem dapat mengenali objek yang menyerupai wajah manusia selama objek tersebut memiliki *kontur* yang mirip dengan *template* wajah.

### Saran:

Meskipun sistem ini menunjukkan kinerja yang baik, namun diperlukan penelitian lebih lanjut untuk meningkatkan akurasi pengenalan wajah, terutama ketika menangani kasus pengenalan palsu pada kondisi tertentu. Selain itu, meningkatkan sistem agar berhasil mengenali dan mengenali wajah bahkan ketika terdapat perbedaan besar antara gambar masukan dan gambar referensi akan meningkatkan keandalan dan fleksibilitas sistem dalam berbagai skenario aplikasi.

## DAFTAR PUSTAKA

- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 1, I-I. <https://ieeexplore.ieee.org/document/990517>
- Rosyani, P. (2021). Penerapan Algoritma Haar Cascade Classifier untuk Pengenalan Wajah Menggunakan OpenCV. Jurnal Informatika, 8(2), 139-146. <https://journal.unpak.ac.id/index.php/jinformatika/article/view/2440>
- Rosyani, P. (2019). Implementasi Algoritma Haar Cascade Classifier untuk Deteksi Wajah pada Citra Digital. Prosiding Semnastek, 1(1), 142-147. <http://prosiding.upnvj.ac.id/index.php/semnastek/article/view/205>
- N. R. Syambas and U. H. Purwanto, "Image processing and face detection analysis on face verification based on the age stages," in 2012 7th International Conference on Telecommunication Systems, Services, and Applications, TSSA 2012, 2012, doi: 10.1109/TSSA.2012.6366070.
- Perani Rosyani, (2017). "Pengenalan Wajah Menggunakan Metode Principal Component Analysis (PCA) dan Canberra Distance" P Rosyani - Jurnal Informatika Universitas Pamulang, [pdf-libre.pdf \(d1wqtxts1xzle7.cloudfront.net\)](pdf-libre.pdf(d1wqtxts1xzle7.cloudfront.net))
- Kadir, Abdul, (2005), "Dasar Pemrograman Python", Yogyakarta, Andi Offset.
- Aura Amalia Warzuqni, Divia Putri Sabilla, Zara Agustin, Perani Rosyani. (2022/9/30). "Analisa Sistem Presensi Kelas Menggunakan Pengenal Wajah Dengan Metode Haar CascadeClassifier". <https://journal.mediapublikasi.id/index.php/manekin/article/download/1315/619>
- Perani Rosyani & Retnawati Retnawati, (2023). "Ekstraksi Fitur Wajah Menggunakan Metode Viola Jones dengan Tools Cascade Detector." [Ekstraksi Fitur Wajah Menggunakan Metode Viola Jones dengan Tools Cascade Detector | Rosyani | JURIKOM \(Jurnal Riset Komputer\) \(stmik-budidarma.ac.id\)](Ekstraksi_Fitur_Wajah_Menggunakan_Metode_Viola_Jones_dengan_Tools_Cascade_Detector_|_Rosyani_|_JURIKOM_(Jurnal_Riset_Komputer)_(_stmik-budidarma.ac.id))
- Nugraha, Raditya, (2011), "GameTicTacToe dengan Gerakan Jari Menggunakan Metode Viola And Jones", Politeknik Elektronika Negeri Surabaya, ITS. <https://publikasiilmiah.ums.ac.id/handle/11617/3936?show=full>