

# DETEKSI KEDIPAN MATA PADA VIDEO MENGGUNAKAN *OPEN CV*

Haruno Sajati<sup>1</sup>, Anggraini Kusumaningrum<sup>2</sup>, Bagus Budi Utomo<sup>3</sup>

Program Studi Teknik Infomatika

Sekolah Tinggi Teknologi Adisutjipto

jati@stta.ac.id<sup>1</sup>, anggraini@stta.ac.id<sup>2</sup>, Bag0ezbudhitama@gmail.com<sup>3</sup>,

## ABSTRACT

*In the development of technology in the current era is one of the most rapid progress, it is marked by almost all managers of data and information has been done with the computer because of the increasingly diverse information issues handled, Detection of objects is one of the most important early stages before Done object recognition process. Eye blink detection can be used to open a new file or an application. In this research use case detection eye blinking on video using OpenCV to open notepad application. The blink detection uses the haar cascade classifier method for realtime detection and the OpenCV Library. The first step determines the pattern 1111100000. Next enter the video sample avi format to match with a predetermined pattern. In this program the system managed to detect the eye according to the pattern by circling the eye area when the eyes are open or literate and when the eyes closed or brake there is no circle in the eye area for the correct video sample.*

**Keywords:** *Blind Detection On Video , Haar Cascade Classifier, OpenCV*

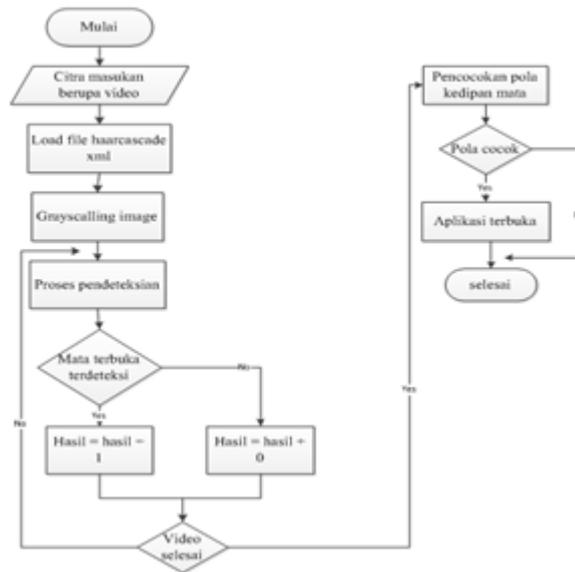
## 1. Pendahuluan

Perkembangan teknologi komputer pada era sekarang ini merupakan salah satu kemajuan yang amat pesat, hal ini ditandai dengan hampir semua pengelola data dan informasi telah dilakukan dengan komputer karena semakin beraneka ragam permasalahan informasi yang ditangani. Sehingga banyak pekerjaan manusia yang telah digantikan oleh komputer seperti pendeteksian obyek. Sistem pendeteksian obyek merupakan suatu teknologi yang sangat membantu karena dengan adanya pergantian posisi manusia oleh komputer, pekerjaan dapat diselesaikan dengan lebih optimal, namun tetap diandalkan. Pendeteksian obyek merupakan salah satu tahapan awal yang sangat penting sebelum dilakukan proses pengenalan obyek. Banyak dari sistem pendeteksian tersebut menggunakan metode Viola Jones sebagai metode pendeteksian obyek. Metode Viola Jones dikenal memiliki kecepatan dan keakuratan yang cukup tinggi karena menggabungkan beberapa konsep (*Haarlike Feature, Citra Integral, AdaBoost, Cascade Classifier*) menjadi sebuah metode utama untuk mendeteksi obyek. Deteksi kedipan mata dapat digunakan untuk membuka file baru atau suatu aplikasi. Dalam penelitian ini digunakan *Haar Cascade Classifier* sebagai pendeteksi bagian tubuh, khususnya wajah dan mata. Maka perlu dilakukan penelitian yang berjudul “Deteksi Kedipan Mata Pada Video Menggunakan *OpenCV*”. Dengan penelitian ini diharapkan dapat menjadikan deteksi kedipan mata sebagai pencarian pola untuk membuka suatu aplikasi. Sekaligus ikut dalam mengembangkan peningkatan interface antara manusia dan komputer yang lebih natural.

## 2. Metodologi Penelitian

### 2.1. Perancangan Sistem

Perancangan sistem adalah proses menyusun atau merancang dan mendesain suatu sistem informasi yang baru dengan baik, yang isinya yaitu langkah-langkah operasi dalam proses pengolahan data dan prosedur untuk mendukung operasi sistem. Hasil sistem yang dirancang harus sesuai dengan kebutuhan pemakai untuk mendapatkan informasi. Langkah-langkah Deteksi Kedipan Mata Pada Video Menggunakan *OpenCV* dengan metode *haarcascade classifier* pada *library openCV 2.4.11* dapat dilihat pada Gambar 1.

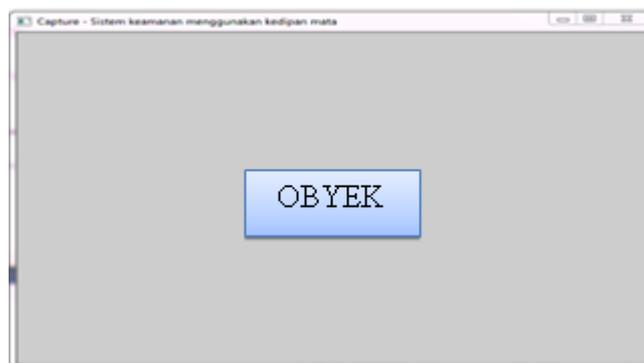


Gambar 1. Flowchart Deteksi Kedipan Mata Pada Video Menggunakan OpenCV

Gambar 1. merupakan flowchart perancangan alur sistem yang digunakan dalam proses deteksi mata dan kedipan mata menggunakan *haarcascade classifier* yang tersimpan dalam bentuk file XML. Langkah pertama, melakukan proses perekaman video, selanjutnya dilakukan proses *grayscale* untuk proses deteksi mata dan kedipan mata. Kemudian aplikasi mengakses file *EmguCV Haarcascade\_eye\_tree\_eyeglasses.xml* untuk melakukan *training file haarcascade.xml* untuk menentukan fitur *haar* pada video input dan mempelajari *haar*, sehingga dapat menentukan gambar mata dan bukan mata dan hasil keluarannya berupa pola, apabila pola benar aplikasi akan terbuka dan jika pola salah tidak ada aplikasi yang terbuka.

## 2.2. Perancangan Tampilan Program

Perancangan tampilan program merupakan mekanisme komunikasi antara pengguna (*user*) dengan sistem. Tampilan program dapat menerima informasi pengguna (*user*) dan memberikan informasi kepada pengguna (*user*) untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi. Tampilan program pada penelitian ini dapat dilihat pada Gambar 2.



Gambar 2. Tampilan Program

Gambar 2 ini merupakan tampilan dari perancangan program. Dimana obyek yang diambil berupa gambar/input berupa video, kemudian obyek terdeteksi dan dilakukan penjejakan/dilingkari pada area mata untuk mendeteksi kedipan mata. Tampilan berikutnya merupakan tampilan aplikasi notepad terbuka yang dapat dilihat pada Gambar 3.



Gambar 3 Tampilan Aplikasi Notepad Terbuka

Tampilan Gambar 3 adalah saat aplikasi notepad terbuka apabila pola kedipan mata terdeteksi sesuai dengan pola yang sudah ditentukan, proses pendeteksian pola dilakukan saat mata terbuka/melek dan saat mata tertutup/merem.

### 3. Hasil dan Pembahasan

Pengujian (*testing*) merupakan suatu tindakan yang dilakukan untuk mendapatkan informasi mengenai keefektifan kinerja deteksi kedipan mata pada video menggunakan *opencv*. Sehingga dapat dilihat tingkat keakurasian dalam pendeteksian obyek tersebut. Pencocokan pola merupakan proses akhir dalam implementasi deteksi kedipan mata pada video menggunakan *opencv*, sehingga didapatkan perbandingan antara hasil pencocokan pola yang benar dengan pola yang salah.

#### 3.1. Hasil Pencocokan Pola Benar

Hasil pencocokan pola dilakukan setelah mendapatkan pola yang benar. Hasil pencocokan pola pada deteksi kedipan mata pada video menggunakan *opencv* dapat dilihat pada Gambar 4.

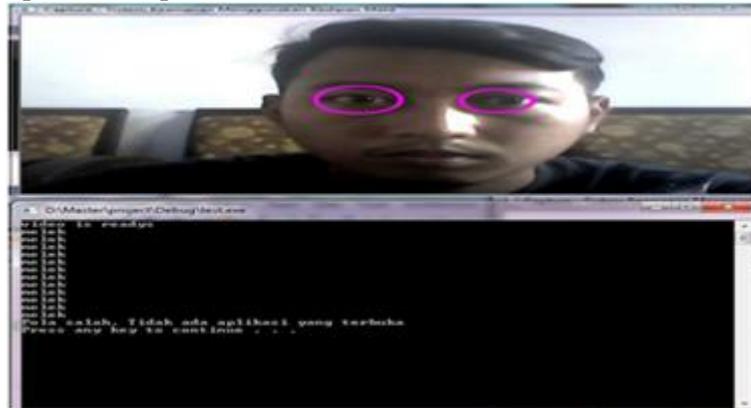


Gambar 4 Hasil Uji Program

Proses pertama kali untuk mendapatkan hasil seperti Gambar 4 yaitu memasukan *sample video* 1100. Langkah selanjutnya menjalankan program deteksi kedipan mata pada video menggunakan *opencv*, jika *sample video* 1100 cocok dengan pola yang sudah ditentukan ( 1111100000 ) saat dijalankan secara otomatis aplikasi notepad akan terbuka, jika angka 1 dan 0 pada pola yang sudah ditentukan lebih dari 5 dan dicocokkan dengan *sample video* 1100 maka aplikasi notepad tidak akan terbuka. Apabila aplikasi notepad terbuka, program deteksi kedipan mata pada video menggunakan *opencv* berhasil dan selesai.

### 3.2. Hasil Pencocokan

Pencocokan pola salah dilakukan untuk membandingkan hasil pola benar dan salah. Hasil pencocokan pola salah dapat dilihat pada Gambar 5.



Gambar 5. Hasil Pola Salah

Cara yang digunakan untuk mengetahui hasil dari pencocokan pola salah dengan memasukan dan menjalankan *sample video* pola 1111. Setelah dijalankan maka akan muncul pemberitahuan jika pola salah dan program langsung berhenti.

## 4. Penutup

### 4.1. Kesimpulan

Berdasarkan hasil dari implementasi sistem adapun yang dapat disimpulkan pada penelitian ini adalah sebagai berikut:

1. Deteksi kedipan mata pada video menggunakan *opencv* dibuat hanya mengenali atau mengidentifikasi pada area mata.
2. Metode *haar cascade classifier* ideal digunakan untuk deteksi kedipan mata secara *realtime*.
3. Hasil pengujian deteksi kedipan mata pada video menggunakan *opencv* secara *realtime*, sistem berhasil mendeteksi atau melingkari area mata dan notepad dapat terbuka.
4. *Sample data* dengan kualitas citra yang baik sangat diperlukan, selain kualitas citra yang baik, jumlah *sample data* sangat mempengaruhi hasil yang didapatkan.

Deteksi kedipan mata pada video menggunakan *opencv* yang dibangun sementara hanya dapat membuka aplikasi notepad.

### 4.2. Saran

Didapatkan beberapa saran untuk pengembangan selanjutnya, sebagai berikut:

1. Deteksi kedipan mata menggunakan *opencv* dapat dikembangkan untuk mampu menangkap obyek dalam wajah posisi apapun.
2. Tampilan program untuk kedepannya bisa diperbaiki agar lebih menarik dan mudah dipahami.
3. Untuk penelitian selanjutnya program ini dapat dikembangkan untuk membuka file yang tersimpan/tersembunyi, jadi tidak hanya file baru.
4. Kualitas pendeteksian dapat ditingkatkan dengan menambah jumlah *file training*.

**Daftar Pustaka**

- [1] Agustian Indra & Sulistyaningsih, 2012. Implementasi Pengolahan Citra Berbasis Metode *Contours Finding* Dan *Surf* Untuk Deteksi Kedip Mata, Jurusan Teknik Elektro Universitas Bengkulu, *Proceeding Semantik 2012*, Bengkulu.
- [2] Ardianto Andi, dkk, 2014. Aplikasi Pengembalian Buku Mandiri Menggunakan Segmentasi Citra Cover Buku, *Jurnal System Vol. 3 No 1*, STMIK STIKOM, Surabaya.
- [3] Jogyanto HM., MBA., Akt., Ph.D. 2005. *Analisis dan Desain*, Penerbit Andi. Yogyakarta.
- [4] Kadir Abdul & Susanto Adhi. 2013. *Teori dan Aplikasi Pengolahan Citra*, Penerbit Andi. Yogyakarta.
- [5] Kusmanto RD, dkk, 2012. Aplikasi Sensor Vision untuk Deteksi Multiface dan Menghitung Jumlah Orang, Jurusan Teknik Komputer. Politeknik Negeri Sriwijaya, *Seminar Teknologi Informasi dan Komunikasi Terapan 2012 (semantik 2012)*, Palembang.
- [6] Putra Darma, 2010. *Pengolahan Citra*, Penerbit Andi. Yogyakarta.
- [7] Santoso Hadi & Harjoko Agus, 2013. *Haar Cascade Classifier* dan Algoritma *Adaboost* untuk Deteksi Banyak Wajah Dalam Ruang Kelas, Universitas Gadjah Mada, *Jurnal System Vol 6 No 2*, Yogyakarta
- [8] Syarif Muhammad & Wijanarto, 2015. Deteksi Kedipan Mata Dengan *Haar Cascade Classifier* dan *Contour* untuk *Password Login* Sistem, Teknik Informatika Universitas Dian Nuswantoro, *Techno COM, Vol 14 No 4*, Semarang.