

New search x untitled report x +

← → ↻ 🔒 quetext.com/reports/0952111d0cb2bddfb89a ☆ 🔍 🌐 🗑️ ⋮

Apps New Tab

sudut. Lihat gambar 2.3 untuk contohcontoh tekstur primitif ini.

Gambar 2.3

2.5.3. Face Recognition Menggunakan Local Binary Patterns

Metode LBP dapat diterapkan pada citra wajah untuk mengekstrak fitur yang dapat digunakan untuk mendapatkan persamaan ukuran. Gagasan utama dari metode ini yaitu penghitungan kode LBP dari setiap piksel suatu gambar dengan tetap mempertahankan setiap pola yang ada. Representasi dari tekstur gambar yang diperoleh akan tampil dalam bentuk histogram. Histogram tersebut dapat digunakan untuk mengukur kesamaan antar masing-masing gambar dengan menghitung jarak dari masing-masing histogram yang dihasilkan.

No plagiarism found!

Delete

Help

untitled report x +

← → ↻ 🔒 https://www.quetext.com/reports/3e95df5a5a43491eeb1c ☆ 🔍 🌐 🗑️ ⋮

untitled

2.6. Viola-Jones Face Detector

Deteksi wajah Viola-Jones menggunakan prosedur (Viola,P,2001) pengklasifikasikan gambar yang didasarkan oleh nilai fitur sederhana. Berdasarkan jumlah persegi panjangnya, maka fitur dibagi menjadi 3 jenis yaitu seperti yang dapat dilihat pada gambar 2.4 berikut:

Gambar 2.4

Fitur A dan B terdiri dari dua persegi panjang, fitur C terdiri dari tiga persegi panjang dan fitur D empat persegi panjang. Contoh integral image dapat dilihat pada Gambar 2.5 berikut:

Gambar 2.5

Untuk menentukan nilai pada daerah yang diarsir, dapat dilakukan dengan

No plagiarism found!

Delete

Help

kedua metode tersebut.

3.1.1 Algoritma

Haar-Like Feature, Integral image, Adaboost (Adaptive Boosting), dan Cascade Classifier merupakan alur yang dilakukan agar mendapatkan sebuah output wajah yang terdeteksi atau tidak pada sebuah citra pada. Algoritma yang digunakan pada deteksi wajah ini yaitu Viola-Jones dan LBPH.

1. Pemilihan fitur

a. Haar-like feature


Tahap pertama yaitu merubah image menjadi citra grayscale.

b. Integral Image

Hasil penjumlahan nilai pixel pada daerah yang dideteksi oleh fitur haar akan dihitung dengan menggunakan integral image. Dengan menggunakan integral image perhitungan dilakukan dengan cepat dan hasilnya akurat dikarenakan hanya butuh satu kali scan.

Gambar 3.2 Citra Masukan

$$s(x,y) = s(x,y-1) + s(x-1,y) + i(x,y) + s(x,y) - s(x-1,y-1)$$



No plagiarism found!

Delete

Help

untitled


2. Klasifikasi bertingkat

a. Adaboost (Adaptive Boosting)

Adaptive boosting merupakan teknik yang menghasilkan classifier yang baik dari gabungan banyak classifier lemah atau classifier dasar. Satuan dari classifier dasar tersebut disebut dengan weak learner. Algoritma adaboost dilakukan untuk dilakukan deteksi pada image yang bertujuan untuk mengetahui ada atau tidaknya fitur wajah pada daerah klasifikasi fitur lemah. Selanjutnya dilakukan perbandingan dan perhitungan antara classifier lemah dengan classifier lainnya secara acak. Setelah itu akan terbentuk kombinasi yang linier.

b. Cascade classifier

Cascade classifier merupakan pengorganisasian fitur-fitur kedalam bentuk klasifikasi yang bertingkat. Untuk dapat mengetahui benar atau tidaknya ada fitur wajah dalam fitur wajah, dilakukan dengan cara bertingkat.



No plagiarism found!

Delete

Help

untitled report

https://www.quetext.com/reports/321a8c7b85735c648b49

← → ↻

☆ 🔍 🚫 📄 👤 ⋮

☰

🔍

📁

👤

🚀

berperan sebagai berikut:

- a.Membuat fungsi-fungsi yang akan digunakan.
- b.Menghubungkan antara library pendeteksian fitur dengan aplikasi.
- c.Membuat tampilan GUI (Graphical user interface).


2.Library

Library yang digunakan dalam pembuatan aplikasi deteksi wajah ini adalah openCV library. Algoritma pendeteksian yang akan digunakan sudah dibangun pada library tersebut.

3.Algoritma deteksi wajah.

Pembuatan aplikasi perbandingan 2 metode deteksi wajah ini menggunakan algoritma Viola-Jones dan LBPH. Adapun peranan dari algoritma dalam pembuatan aplikasi ini adalah sebagai berikut:

- a.Melakukan pendeteksian wajah pada citra.
- b.Menerapkan karakteristik dan metode pendeteksian pada aplikasi yang akan dibuat.



No plagiarism found!

Delete

🔍 Help