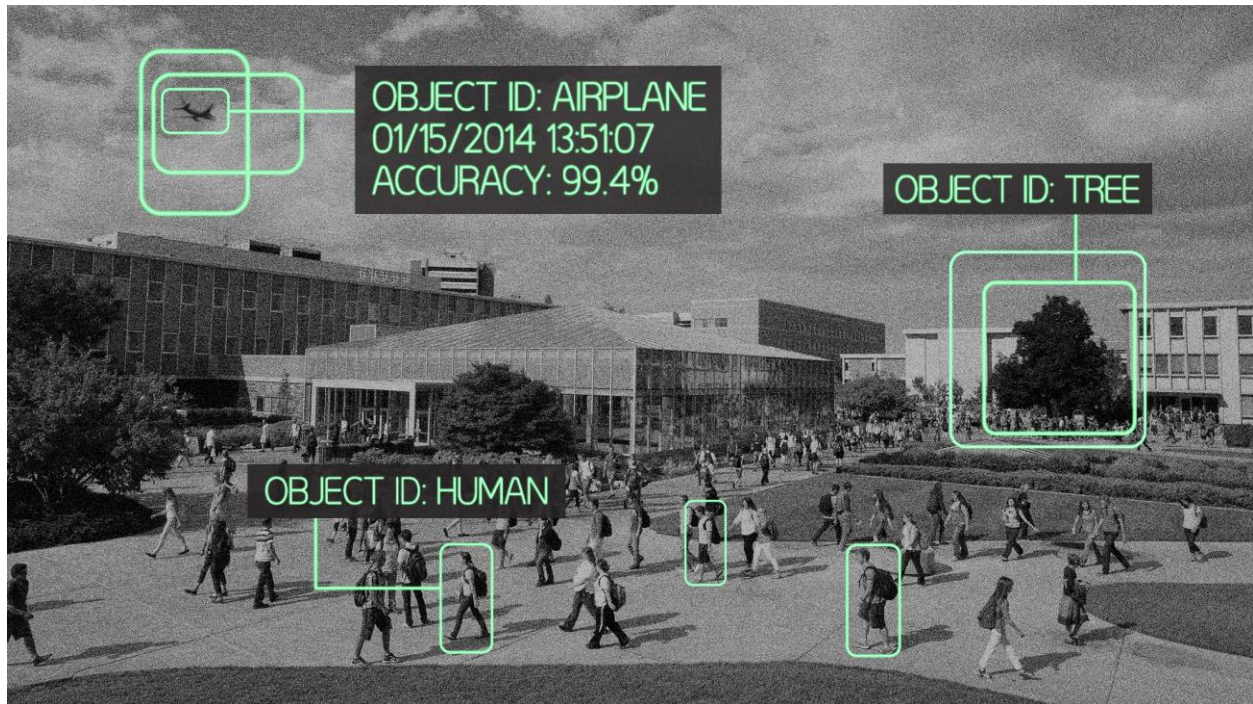


HAAR LIKE FEATURE PADA VIOLA JONES

Oleh : mulkan.ms@gmail.com



Tulisan ini dibuat untuk menjelaskan dasar-dasar haar like feature pada teknik viola jones.

<http://www.softscients.web.id/2017/01/number-plate-car-detection.html>

<http://www.softscients.web.id/2017/01/cars-detection-in-opencv.html>

Tapi penulis hanya menjelaskan haar like featurnya saja, bukan keseluruhan, karena penulis hanya menggunakan OpenCV dan Python dalam implemementasi object detection.

Metode Viola-Jones merupakan salah satu metode pendeteksian objek yang cukup populer, yang dapat memberikan hasil dengan tingkat keakuratan yang cukup tinggi dan dengan kecepatan yang sangat tinggi. Metode ini, diusulkan oleh Paul Viola dan Michael Jones pada tahun 2001, dengan menggabungkan empat kunci utama untuk mendeteksi suatu objek:

1. Fitur persegi sederhana, disebut fitur *Haar*
2. *Integral image* untuk pendeteksian fitur dengan cepat
3. Metoda AdaBoost *machine-learning*
4. *Cascade classifier* untuk mengkombinasikan banyak fitur

Saya akan membahas pada point pertama yaitu **fitur haar**. Ada banyak jenis fitur haar yang digunakan



Dapat dilihat bahwa terdapat 5 fitur. Cara menghitung nilai dari fitur ini adalah mengurangkan nilai *pixel* pada area hitam dengan *pixel* pada area putih. Dengan Rumus untuk mendapatkan nilai fitur sesuai dengan jumlah kotak jika B , W : *Black* (Hitam), *White* (Putih), maka cara hitungannya yaitu

Dua kotak : $W - B$

Tiga kotak : $W_1 + W_2 - B$

Empat kotak : $(W_1 + W_2) - (B_1 + B_2)$

Misalkan pada contoh berikut

2	4	7	5	8
1	5	1	7	7
5	6	9	5	6
8	9	10	6	7
10	12	8	3	6

hitam 60
 putih 47
 fitur haar = hitam-putih 13

Menghasilkan ciri fitur 13, dengan begitu akan dapat ditentukan tingkat *luminance* dari citra yang akan dideteksi dan dapat dibedakan mana bagian citra yang mencirikan wajah manusia. Untuk mempermudah proses penghitungan nilai fitur, metode Viola Jones menggunakan sebuah media berupa citra integral.

Ilustrasi perhitungan integral image

1	1	1		1	2	3
1	1	1		2	4	6
1	1	1		3	6	9

Maka

2	4	7	5	8
1	5	1	7	7
5	6	9	5	6
8	9	10	6	7
10	12	8	3	6

INTEGRAL IMAGE

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

Cara menghitung selisih kotak hitam dengan putih sebagai berikut

KONDISI SATU

A		C	
B		D	

KONDISI DUA

	A		B
	C		D

Maka ciri fitur haar like adalah

CIRI FITUR = D+A-B-C

Kondisi pertama (hitam)

INTEGRAL IMAGE

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

$$\begin{aligned}
 & \text{hitam} = \frac{D+A-(B+C)}{2} \\
 & \text{putih} = \frac{E19+C15-E15-C19}{2} \\
 & \text{fitur haar} = \text{hitam} - \text{putih} = 13
 \end{aligned}$$

Kondisi kedua (putih)

INTEGRAL IMAGE

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

$$\begin{array}{rcl}
 & D+A-(B+C) & \\
 \text{hitam} & \underline{60} & \\
 \text{puti} & = \text{G19} + \text{E15} - \text{G15} - \text{E19} & \\
 \text{fitur haar} = \text{hitam} - \text{putih} & \underline{13} &
 \end{array}$$

Sehingga menjadi berikut

INTEGRAL IMAGE

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

$$\begin{array}{rcl}
 & D+A-(B+C) & \\
 \text{hitam} & 60 & \\
 \text{putih} & 47 & \\
 \text{fitur haar} = \text{hitam} - \text{putih} & 13 &
 \end{array}$$

Sama bukan?? Yaitu akan menghasilkan angka 13, sehingga teknik **integral image** menjadi lebih efektif untuk menghitung **luminance/haar like feature**.

Berikut contoh dilokasi lainnya

2	4	7	5	8
1	5	1	7	7
5	6	9	5	6
8	9	10	6	7
10	12	8	3	6

hitam	34
putih	24
hitam-putih	10

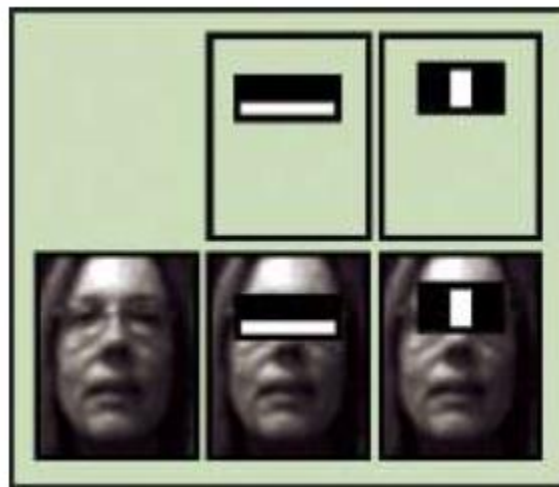
INTEGRAL IMAGE

2	6	13	18	26
3	12	20	32	47
8	23	40	57	78
16	40	67	90	118
26	62	97	123	157

$D+A-(B+C)$

hitam	34
putih	24
fitur haar = hitam	10

Sama-sama menghasilkan nilai 10. Tentu dalam teknik viola jones tidak hanya menggunakan 1 ciri fitur diatas, tapi akan sangat banyak sekali

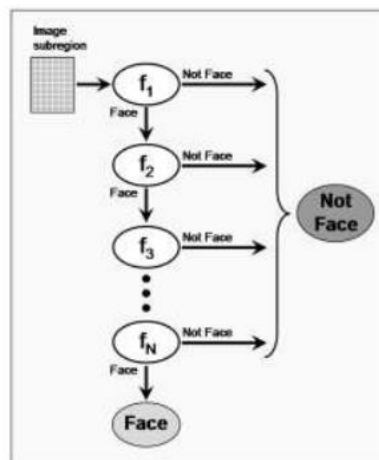




Gambar 6. Contoh pencarian dengan haar-like feature

Selanjutnya, setelah nilai fitur didapatkan Viola-Jones menggunakan metode AdaBoost *machine-learning* untuk mengetahui suatu fitur apakah merepresentasikan ada tidaknya wajah dalam suatu citra masukan. Metode AdaBoost menggabungkan banyak *classifier* lemah menjadi satu *classifier* kuat. *Classifier* adalah suatu ciri yang menandakan adanya objek wajah dalam suatu citra. Sedangkan *classifier* lemah adalah suatu jawaban benar namun memiliki tingkat kebenaran yang kurang akurat, jika digabung maka *classifier* lemah tersebut akan menghasilkan suatu *classifier* kuat.

Gambar 2.13, menjelaskan bagaimana metode AdaBoost ini akan menyeleksi nilai fitur dari citra masukan, jika pada fitur tersebut tidak terdeteksi adanya wajah maka akan disimpulkan tidak ada wajah, jika ada akan diteruskan ke tingkat selanjutnya sampai ke tingkat terakhir, lalu akan disimpulkan pada citra masukan terdeteksi adanya wajah. Proses penyeleksian fitur tersebut dengan cara membandingkan nilai fitur dengan nilai ambang yang telah ditentukan pada tiap tingkat, jika nilai fitur sama atau diatas nilai ambang maka terdeteksi adanya wajah dalam fitur tersebut.



Gambar 2.13. Metode AdaBoost Machine-Learning

Tingkatan proses penyeleksian fitur diurutkan berdasarkan bobot terberat sampai teringan, sehingga dapat secepat mungkin untuk mengklasifikasikan bahwa tidak terdeteksi wajah dalam citra masukan, pengurutan proses penyeleksian fitur ini disebut *cascade classifier*.