

BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Studi

Piyabute Fuangkhon [10] melakukan penelitian deteksi puting payudara pada citra pornografi dengan menggabungkan antara deteksi kulit dan deteksi puting. Deteksi kulit menggunakan pengolahan citra dengan melakukan konversi ruang warna RGB ke HSV. Setelah didapatkan area kulit pada citra yang akan dideteksi, langkah selanjutnya adalah dengan mengumpankan area kulit tersebut ke arsitektur Neural Network *window by window* untuk menentukan apakah jendela yang diuji adalah puting dengan algoritma *unsupervised* Neural Network yaitu Kohonen Self Organizing Maps. Dari hasil pengujian akurasi didapatkan untuk uji data negatif sebesar 99,78%, sedangkan untuk uji data positif sebesar 65,39%.

Nazrul Effendy *et al.* [13] dalam penelitiannya melakukan pengenalan pola dengan algoritma Learning Vector Quantization (LVQ). Tahap pertama yang dilakukan adalah melakukan deteksi tepi dengan operator sobel, pendeteksian ini menggunakan pendekatan dua objek setengah lingkaran dan lingkaran didalamnya memiliki jarak yang berdekatan yang kemudian diubah dalam bentuk vektor untuk diolah menggunakan jaringan syaraf tiruan dengan algoritma LVQ. Pada penelitian tidak melakukan pengujian nilai akurasi pendeteksian.

Xiaoyin Wang *et al.* [3] dalam penelitiannya mengusulkan sebuah algoritma dengan nama *breast detecting algorithm* yang menggunakan *nipple feature* (NF) untuk mengenali objek payudara, dengan pertama kali melakukan deteksi kulit. Kemudian, melakukan *Candidate Region Segmentation* dengan memilih area NF dari citra menggunakan algoritma *fuzzy cluster*. Setelah itu dilanjutkan dengan proses *Feature Extraction of Candidate Region* untuk mendapatkan area puting payudara dengan algoritma *Forward Propagation*. Dan langkah terakhir *Feature Training and Classification*. Metode yang mereka usulkan memperoleh nilai *recall rate* sebesar 90,01% dan *precision rate* 93,18%.

Shen *et al.* [14] dalam penelitiannya terdapat 6 sub model, yaitu: deteksi wajah, deteksi kulit, deteksi bentuk tubuh, deteksi organ intim (payudara dan vagina), ekstraksi fitur, dan pengklasifikasian dengan algoritma *decision tree*. Deteksi wajah memainkan peran penting dalam model ini untuk dilanjutkan ke deteksi tahap selanjutnya. Permasalahan akan muncul jika wajah tidak berhasil terdeteksi padahal citra tersebut adalah citra pornografi. Nilai *true positive rate* sebesar 89,50% dan *false positive rate* sebesar 10,50%. Akurasi deteksi sebesar 89,50% dengan *error rate* 18,64%.

Yue Wang *et al.* [2] dalam penelitiannya mengkombinasikan algoritma AdaBoost dengan informasi bentuk dan statistik warna kulit pada area sekitar puting payudara dalam dua tahap. Tahap pertama deteksi puting dengan algoritma AdaBoost untuk mendapatkan area yang dianggap sebagai puting (*region candidate*). Kemudian, mengkombinasikan hasil deteksi tahap pertama dengan informasi bentuk dan statistik warna kulit pada area sekitar puting payudara. Tahap kedua ini digunakan untuk mereduksi kesalahan area deteksi. Akurasi deteksi pada penelitian ini sebesar 75,6%, meningkat dari penelitian sebelumnya yang hanya 65,4% tapi dengan data *testing* yang berbeda.

X. Kejun *et al.* [15] mengembangkan penelitian yang dilakukan oleh Yue Wang dengan membuat *extended Haar-Like Feature* pada algoritma AdaBoostnya, juga dipadukan dengan fitur-fitur warna, tekstur dan bentuk. Dan metode ini dapat dengan efektif mereduksi *False Positive Rate*. Penelitian ini juga mengkomparasi tiga jenis *cascade AdaBoost classifiers* yaitu *basic Haar-like feature*, *Extended Haar-like feature* dan *final cascade AdaBoost classifier* yang dikombinasikan dengan fitur-fitur warna, tekstur dan bentuk (*join features*). Dan juga melakukan komparasi metode sebelumnya [2] dengan menguji pada data *testing* yang sama. Akurasi deteksi yang diperoleh sebesar 90,37%.

Pada Tabel 1 ditunjukkan rangkuman penelitian terkait deteksi payudara dan puting payudara.

Tabel 1 Rangkuman penelitian terkait deteksi payudara dan puting payudara

No.	Nama	Peneliti	dan	Masalah	Metode	Hasil
-----	------	----------	-----	---------	--------	-------

	Tahun			
1.	Piyabute Fuangkhon 2005 [10]	Deteksi Puting Payudara	Hibrid <i>skin detection</i> dan <i>nipple detection</i> menggunakan SOM	Tingkat akurasi 65,39%
2.	Nazrul Effendy <i>et al.</i> 2008 [13]	Deteksi Payudara	Ekstraksi fitur berbasis deteksi tepi dan LVQ	Tidak menampilkan hasil pengujian
3.	Xiaoyin Wang <i>et al.</i> 2009 [3]	Deteksi Payudara	Ekstraksi fitur berbasis <i>fuzzy cluster</i> dan <i>forward propagation</i> <i>neural network</i>	Recall rate 90.01% dan precisions 93.18%
4.	Shen <i>et al.</i> 2010 [14]	Deteksi <i>Erotic Parts</i> (Payudara dan Vulva)	Ekstraksi <i>facial features</i> dan <i>trunk contour</i> dilanjutkan dengan <i>erotic detector</i> berbasis <i>color and shape features</i> dikomparasikan dengan <i>traditional model</i> .	<i>true positive rate</i> sebesar 89.50% dan <i>false positive rate</i> sebesar 10.50%. Akurasi deteksi sebesar 89.50% dengan <i>error rate</i> 18.64%.
5.	Yue Wang <i>et al.</i> 2010 [2]	Deteksi Puting Payudara	<i>Hybrid</i> algoritma Adaboost dan ekstraksi fitur	Tingkat akurasi deteksi 75.6%

			payudara	
6.	X. Kejun <i>et al.</i> 2012 [15]	Deteksi Puting Payudara	<i>Extended Haar- Like Feature</i> digabung dengan fitur- fitur warna, tekstur dan bentuk	Tingkat akurasi deteksi 90.37%

2.2 Tinjauan Pustaka

2.2.1 Puting Payudara

Dalam kamus *Oxford Dictionary* disebutkan definisi puting payudara (*nipple*) adalah tonjolan kecil di saluran susu dari mamalia betina berakhir dan dari mana susu dapat disekresikan. Definisi lain menyebutkan puting payudara merupakan proyeksi berpigmen pada permukaan dada pria dan payudara pada wanita. Pada wanita dewasa, saluran yang mengalirkan susu dari kelenjar susu ke permukaan luar payudara. Daerah datar berpigmen berupa lingkaran disekitar puting payudara disebut areola [33]. Pada penelitian ini pendeteksian puting meliputi wilayah areola pada puting payudara wanita.

Puting payudara merupakan salah satu *erotic parts* yang merupakan ciri sebuah citra pornografi [34]. Oleh karena itu, salah satu cara untuk memblokir citra atau gambar yang mengandung unsur pornografi adalah dengan mendeteksi organ intimnya yang dalam hal ini berupa puting payudara yang meliputi wilayah areola.

2.2.2 Pornografi

Henry George Liddell dan Robert Scott didalam *A Greek-English Lexicon* memberikan definisi pornografi. “Porno” (π opvo) berasal dari bahasa Yunani yang melambangkan pelacur serta budak perempuan, karena prostitusi pada masa Yunani kuno biasa membeli budak untuk dijadikan pelacur. “Graphy” (γραφος) melambangkan menulis. Secara Etimologi pornografi adalah menggambarkan wanita

sebagai budak seks, dan secara seksual serta psikis kembali memperbudak perempuan dalam produksi dan proses konsumsi [16]. Namun demikian, beberapa wacana ilmu sosial mencoba untuk mendefinisikan pornografi sebagai Media seksual eksplisit yang memiliki tujuan utama untuk membangkitkan seksual penonton [17]. Dalam kamus Oxford Dictionaries pornografi memiliki definisi “*printed or visual material containing the explicit description or display of sexual organs or activity, intended to stimulate sexual excitement*” (bahan cetak atau visual yang berisi deskripsi eksplisit atau memamerkan organ atau aktivitas seksual, dimaksudkan untuk merangsang gairah seksual). Pornografi juga secara khusus merupakan konten yang berisi *nudity* (ketelanjangan) dan berbagai variasi adegan seks atau persetubuhan [18].

Terkadang konten pornografi dimasukkan ke dalam konten yang secara umum bukan konten seksual, seperti film, buku, majalah, komik, laman web dan lain sebagainya. Misalkan didalam sebuah film yang bukan film porno sering didapati satu atau banyak adegan yang mengandung unsur pornografi. Beberapa negara memiliki aturan kebijaksanaan yang berbeda dalam menyikapi permasalahan pornografi. Ada beberapa negara yang melarang dan menganggap ilegal semua bentuk konten pornografi. Namun, ada yang menganggapnya legal dengan batasan usia tertentu. Bahkan, beberapa negara ada yang membolehkan untuk melakukan aktivitas berkaitan dengan seks pada usia yang relatif lebih rendah. Di negara-negara barat umumnya permasalahan pornografi lebih terkonsentrasi pada kasus pornografi anak (*child pornography*) [19].

Pornografi tidak hanya memberikan dampak negatif pada sektor ekonomi saja [1], namun juga memberikan dampak pada berbagai hal. Menurut Patrick F. Fagan [30] dampak tersebut berpengaruh pada beberapa hal diantaranya yang pertama efek pada pikiran yaitu pornografi secara signifikan merusak sikap dan persepsi mengenai hakikat hubungan seksual. Pria yang biasa melihat pornografi memiliki kecenderungan yang lebih tinggi untuk berperilaku seksual abnormal, mengidap agresi seksual, pergaulan bebas, dan bahkan pemerkosaan. Selain itu, pria mulai melihat wanita dan bahkan anak-anak sebagai objek seks, komoditas atau instrumen untuk kesenangan mereka, bukan sebagai seseorang dengan martabat mereka sendiri.

Kedua, efek pada tubuh yaitu pornografi sangat membuat ketagihan. Aspek adiktif pornografi memiliki substrat biologis, dengan pelepasan hormon dopamin bertindak sebagai salah satu mekanisme untuk membentuk jalur transmisi ke pusat-pusat kesenangan otak. Juga, meningkatnya kebebasan seksual yang ditimbulkan oleh pornografi meningkatkan risiko tertular penyakit menular seksual atau menjadi orangtua yang tanpa disadari kehadirannya disebabkan kehamilan diluar nikah. Ketiga, efek pada hati yaitu pornografi mempengaruhi kehidupan emosional manusia. Pria yang telah menikah yang terlibat dalam pornografi merasa kurang puas dengan hubungan seksual dalam pernikahan mereka dan kurang memiliki perasaan emosional yang melekat pada istri-istri mereka. Wanita yang menikah dengan pria pecandu pornografi banyak memberikan laporan berkaitan dengan perasaan dikhianati pasangan, ketidakpercayaan, dan kemarahan. Penggunaan pornografi dapat menyebabkan perselingkuhan dan bahkan perceraian. Remaja yang melihat pornografi merasa malu, berkurang kepercayaan diri, dan mengalami ketidakpastian seksual.

Seiring perkembangan teknologi informasi khususnya internet, maka perkembangan konten pornografi semakin meluas dan aksesnya pun sangat mudah. Umumnya di internet konten pornografi dalam bentuk tulisan, citra (gambar dan foto digital) dan video pornografi. Para pengakses konten pornografi tersebut bukan hanya dari kalangan dewasa, tapi mulai remaja dan anak dibawah umur. Bahkan remaja adalah yang paling rentan terkena dampak negatif dari persebaran konten pornografi. Satu dari tujuh remaja mengalami ajakan yang tidak diinginkan akibat dari pengaruh pornografi seperti ajakan untuk bertemu langsung, meminta remaja untuk berbicara masalah seksual atau meminta untuk foto telanjang [20]. Para remaja yang terdiri dari pria dan wanita ini mengakses konten pornografi dengan berbagai kata kunci yang mereka kehendaki [21].

Tabel 2 Karakteristik pemaparan terhadap Internet Pornografi pada usia dibawah 18 tahun [21]

	Boys n = 192	Girls n = 371
Exposure to Internet pornography		
Yes, before 18**	93.2%	62.1%
Yes, after 18***	4.2	20.6
Never***	2.6	17.3
Exposure to Internet pornography before age 18	n = 179	n = 226
Age at first exposure		
8	0.6%	0.0%
9	0.6	0.0
10	0.6	0.5
11	1.7	1.0
12	10.9	7.7
13	16.0	15.3
14*	21.1	12.4
15	22.9	22.5
16**	20.0	33.0
17	5.7	7.7
Mean at first exposure**	14.3	14.8
Number of times spent viewing for more than 30 minutes		
Never***	30.6%	76.8%
Once	6.9	9.1
Up to 10 times**	27.8	11.4
More than 10 times***	34.8	2.7
Mean number of times spent viewing for more than 30 minutes***	9.8	1.0
Reasons for viewing Internet pornography		
Wanted the sexual excitement***	69.3%	16.8%
Curious about different things people do sexually***	53.1	26.1
Wanted information about sex***	39.7	19.5
With friends who wanted to do it**	34.1	20.8
Never looked for pornography on purposeb***	6.8	42.3
Types of images seen at least once		
Naked people but no sexual activity***	98.3%	88.9%
Mixed-sex intercourse***	93.8	73.5
Naked people showing genitals**	91.1	81.0
Sexual activity among more than 2 people***	83.2	57.1
Same-sex intercourse**	69.3	55.3
Sexual activity involving bondage***	38.6	22.6
Sexual activity between people and animals***	31.8	17.7
Sexual activity involving urine or feces	21.8	16.4
Rape or sexual violence*	17.9	10.2
Sexual pictures of children	15.1	8.9
Mean number of times seen images ^c		
Naked people but no sexual activity***	18.0	6.4
Naked people showing genitals***	16.5	5.3
Mixed-sex intercourse***	16.7	4.5
Sexual activity among more than 2 people***	12.5	3.2
Same-sex intercourse**	11.1	3.3
Sexual activity involving bondage***	3.2	0.9
Sexual activity between people and animals**	1.6	0.6
Sexual activity involving urine or feces**	1.5	0.5
Rape or sexual violence**	1.1	0.3
Sexual pictures of children*	0.9	0.3
TOTAL***	82.8	24.1

2.2.3 Algoritma Viola and Jones

Algoritma Viola and Jones pertama kali diusulkan oleh Paul Viola dan Michael Jones pada tahun 2011 [22] untuk deteksi wajah manusia. Dalam penelitiannya, Viola dan Jones menggunakan 4916 citra positif dan 10.000 citra yang tidak berisi wajah untuk proses pelatihan (*training*). Setiap citra input kemudian dihitung nilai

fiturnya menggunakan *integral image*. Hasil perhitungan *training* menggunakan suatu algoritma *boosting* yang merupakan variasi dari algoritma AdaBoost. Hasil *training* inilah yang digunakan untuk membentuk *cascaded classifier*, yang digunakan untuk mengklasifikasikan wajah.

Kemudian algoritma ini dikembangkan lagi oleh Rainer Lienhart dan Jochen Maydt [23]. Algoritma ini memberikan tiga kontribusi utama yaitu citra integral atau *summed area tables* [23], algoritma pembelajaran berbasis AdaBoost dan *cascade classifier*. Kontribusi pertama dari algoritma ini adalah sebuah representasi citra baru yang disebut citra integral yang memungkinkan untuk evaluasi fitur dengan sangat cepat. Sebagian penelitian ini dilandasi dari penelitian Papageorgiou *et al.* [24]. Viola and Jones menggunakan seperangkat fitur yang mirip dengan fungsi Haar Basis. Untuk dapat menghitung fitur-fitur tersebut dengan cepat pada berbagai skala, Viola Jones memperkenalkan suatu representasi citra integral. Citra integral dapat dihitung menggunakan sedikit operasi per piksel. Setelah citra integral dihitung, maka fitur Haar-like juga dapat dihitung pada berbagai skala dan lokasi dalam waktu yang konstan.

Kontribusi kedua dari algoritma ini adalah metode untuk membangun *classifier* dengan memilih sejumlah kecil fitur penting menggunakan AdaBoost. Dalam setiap *subwindow* citra jumlah total fitur Haar-like sangat besar, jauh lebih besar dari jumlah piksel pada citra tersebut. Untuk memastikan agar klasifikasi cepat, proses pembelajaran harus menghilangkan sebagian besar fitur yang tersedia, dan fokus pada satu set kecil fitur penting. Penyeleksian fitur diperoleh dengan modifikasi sederhana prosedur AdaBoost: *weak learner* dibatasi sehingga setiap *classifier* lemah yang dikembalikan dapat bergantung hanya pada fitur tunggal. Akibatnya setiap tahap dari proses *boosting*, yang memilih *classifier* lemah baru, dapat dilihat sebagai proses seleksi fitur. AdaBoost memberikan algoritma *learning* yang efektif dan memiliki batas kuat pada kinerja generalisasi.

Kontribusi ketiga dari algoritma ini adalah metode untuk menggabungkan *classifier-classifier* yang lebih kompleks secara berturut-turut dalam struktur *cascade* yang secara dramatis meningkatkan kecepatan detektor dengan memfokuskan perhatian pada daerah-daerah yang menjanjikan pada citra. Proses yang lebih kompleks hanya

diperuntukkan bagi daerah-daerah yang menjanjikan pada citra tersebut. Ukuran kunci pendekatan seperti itu adalah *false negative rate* pada proses atensi.

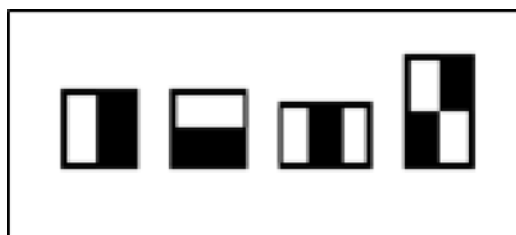
Beberapa *sub windows* yang tidak ditolak oleh *initial classifiers* diproses oleh sederetan *classifiers* yang lainnya, yang masing-masing lebih kompleks dari *classifier* yang terakhir. Jika ada *classifier* yang menolak sub window, maka tidak dilakukan proses selanjutnya. Struktur proses deteksi cascade merupakan degenerasi pohon keputusan.

2.2.3.1 Haar-Like Feature

Algoritma Viola and Jones menggunakan fitur yang dinamakan fitur Haar-like yang mirip dengan fungsi Haar-basis [24]. Fitur Haar-like merupakan sebuah persegi panjang dua dimensi yang terdiri dari area gelap dan terang. Sistem pendeteksian objek Viola Jones mengklasifikasikan citra berdasarkan nilai dari fitur-fitur sederhana. Ada beberapa alasan untuk menggunakan fitur dibanding penggunaan piksel. Salah satu alasan utamanya adalah bahwa fitur dapat meng-*encode* pengetahuan domain *ad-hoc* yang sulit dipelajari menggunakan data latih dengan jumlah yang terbatas. Alasan lainnya adalah bahwa sistem berbasis fitur beroperasi lebih cepat dari pada sistem berbasis piksel.

Ada tiga macam fitur [25] yaitu *two-rectangle feature*, *three-rectangle feature* dan *four-rectangle feature*.

Gambar 1 Haar-like Feature

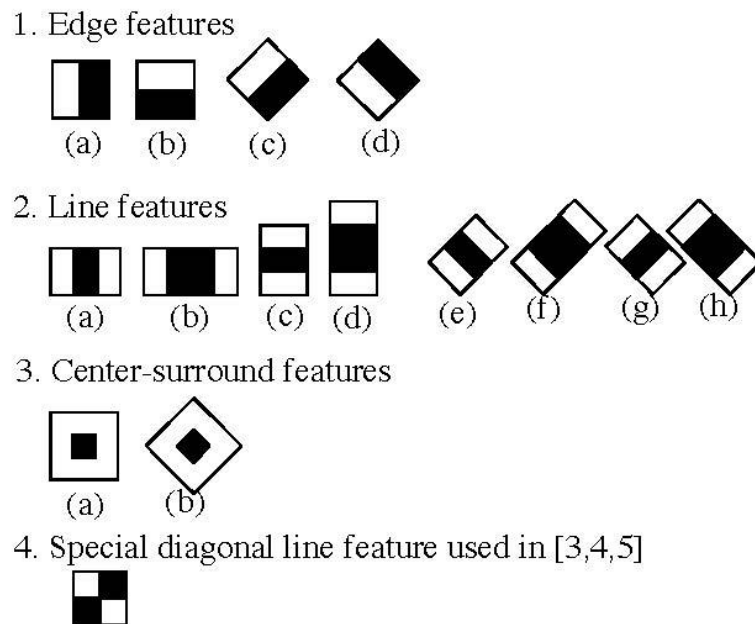


Nilai dari fitur dua persegi panjang (*two-rectangle feature*) adalah selisih antara jumlah nilai piksel *gray level* dalam dua wilayah persegi panjang. Setiap daerah memiliki ukuran dan bentuk yang sama dan berbatasan secara horizontal maupun vertikal (lihat Gambar. 1). Fitur tiga persegi panjang (*three-rectangle feature*) melakukan perhitungan fitur dengan menghitung jumlah piksel pada daerah dua

persegi panjang yang berada luar dikurangi jumlah persegi panjang yang ditengah. Fitur empat persegi panjang (*four-rectangle feature*) menghitung selisih piksel diantara pasangan diagonal persegi panjang.

Fitur Haar-like dikembangkan lagi oleh R Lienhart *et al.* [23] dengan membuat turunannya (*Extended Haar-like Features*) dengan mengembangkan dua hal utama. Pertama, seperangkat fitur Haar-like dasar dirotasi 45° dari kondisi awal, yang menambahkan tambahan domain pengetahuan untuk kerangka belajar dan juga yang sulit untuk belajar. Fitur-fitur baru dapat dihitung dengan cepat di semua skala dalam waktu yang konstan. Kedua, kita memperoleh prosedur optimasi posting baru untuk meningkatkan kinerja *classifier* secara signifikan. Total fitur yang dihasilkan sebanyak 15 fitur.

Gambar 2 *Extended Haar-like Features*



Nilai dari fitur Haar-like dapat ditentukan dengan mencari selisih nilai antara jumlah nilai-nilai piksel *gray level* dalam daerah kotak hitam dan daerah kotak putih. Proses perhitungan ini dilakukan dengan mengurangi nilai rata-rata piksel daerah gelap dengan nilai rata-rata piksel daerah terang. Jika nilai yang dihasilkan diatas nilai *threshold* (ambang batas), maka subwindow yang dideteksi dengan fitur tersebut dianggap sebagai positif objek dan berhenti tidak ke *stage* selanjutnya. Penentuan

nilai *threshold* dilakukan saat *training* menggunakan algoritma AdaBoost[26]. Rumus perhitungannya:

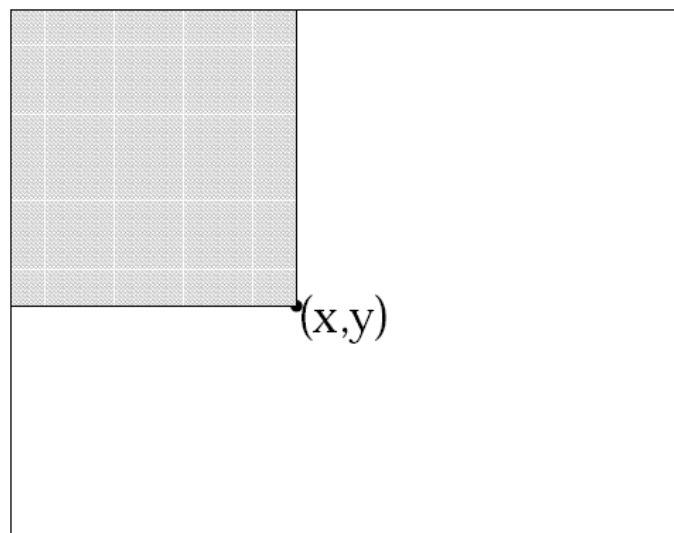
$$f(x) = \text{Sum}_{\text{black rectangle}} (\text{pixel gray level}) - \text{Sum}_{\text{white rectangle}} (\text{pixel gray level}) \quad (1)$$

Untuk menghitung nilai penjumlahan piksel daerah gelap dan daerah terang menggunakan representasi citra integral. Dengan citra integral perhitungan fitur sangat cepat.

2.2.3.2 Integral Image

Untuk mengetahui ada tidaknya ratusan fitur Haar-like di setiap lokasi gambar dan pada beberapa skala secara efisien, Viola dan Jones menggunakan teknik yang disebut *integral image*. Secara umum, "*integrating*" berarti menambahkan unit-unit kecil bersama-sama. Dalam hal ini, unit-unit kecil adalah nilai-nilai piksel. Nilai integral untuk setiap piksel adalah jumlah dari semua piksel di atasnya dan kiri. Mulai dari kiri atas dan melintasi ke kanan dan ke bawah, seluruh gambar dapat diintegrasikan dengan sedikit operasi integer saja per piksel. Citra integral pada lokasi x,y berisi jumlah piksel sebelah atas dan sebelah kiri x,y , termasuk:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2)$$



Gambar 3 Nilai citra integral pada titik (x,y) adalah jumlah seluruh piksel yang berada di atas dan di kiri titik tersebut

Di mana $ii(x,y)$ adalah citra integral dan $i(x,y)$ adalah gambar asli.

$$s(x,y) = s(x,y-1) + i(x,y) \quad (3)$$

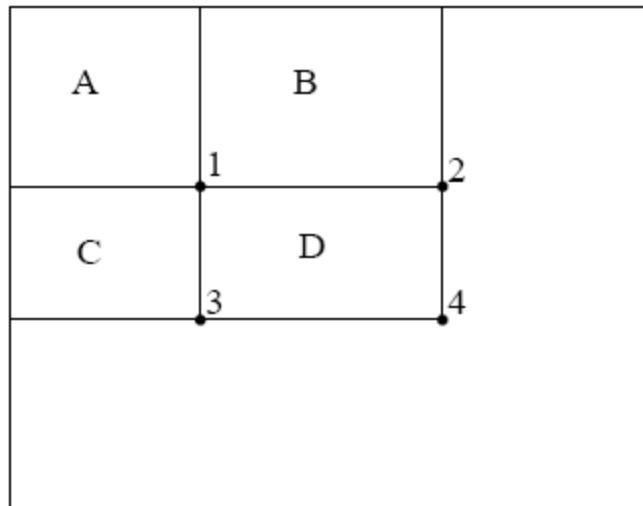
$$ii(x,y) = ii(x-1,y) + s(x,y) \quad (4)$$

Atau dapat dituliskan secara sederhana:

$$ii(x,y) = i(x,y) + ii(x-1,y) + ii(x,y-1) - ii(x-1,y-1) \quad (5)$$

Dimana $s(x,y)$ adalah jumlah kumulatif baris, $s(x,-1) = 0$, dan $ii(-1,y) = 0$ citra integral yang dapat dihitung sekali jalan (*single pass*) dari citra input.

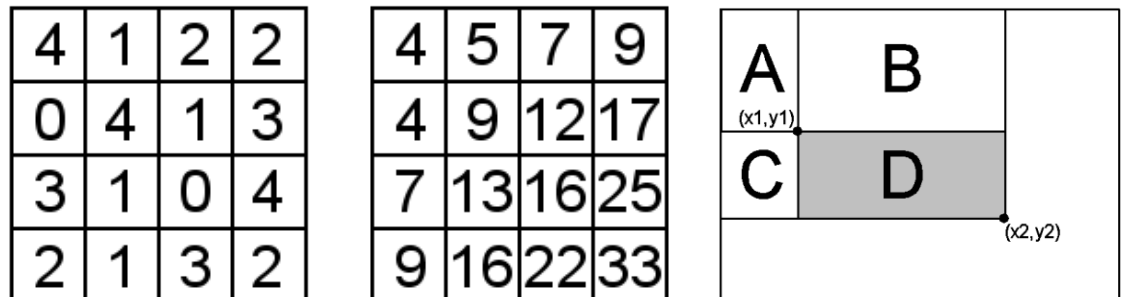
Dengan menggunakan citra integral, jumlah dari setiap persegi dapat dihitung dengan menggunakan empat referensi array. Perbedaan antara jumlah dua persegi dapat dihitung dengan delapan referensi. Fitur dua persegi dapat dihitung dengan enam referensi array, delapan untuk fitur tiga persegi, dan sembilan untuk fitur empat persegi.



Gambar 4 Perhitungan citra integral

Nilai piksel pada persegi panjang D dapat dihitung dengan empat referensi array. Nilai citra integral pada daerah 1 adalah jumlah piksel persegi panjang A. Nilai piksel pada daerah 2 adalah jumlah piksel persegi A+B. Nilai piksel pada lokasi 3

adalah jumlah piksel A+C dan lokasi 4 adalah jumlah dari A+B+C+D. Nilai piksel persegi panjang D dapat dihitung dengan $4+1 - (2+3)$.



Gambar 5 *Integral image*. Kiri: nilai piksel citra inputan. Center: perhitungan citra integral. Kanan: menggunakan citra integral untuk menghitung persegi panjang D.[27]

Misalnya, terdapat citra input dengan berukuran 5x5 piksel dengan masing-masing nilai sebagai berikut:

1	2	2	4	1
3	4	1	5	2
2	3	3	2	4
4	1	5	4	6
6	3	2	1	3

Citra integral dari citra input diatas adalah:

1	3	5	9	10
4	10	13	22	25
6	15	21	32	39
10	20	31	46	59
16	29	42	58	74

Berdasarkan rumus perhitungan nilai piksel pada sebuah citra integral pada Gambar 4, kita bisa menghitung nilai piksel dari citra integral tersebut.

Misalkan kita mengambil contoh untuk mendapatkan nilai piksel dari citra integral.

Contoh 1:

A	B
C	D

Citra integral:

1	3
4	10

Nilai piksel pada daerah hitam = $D + A - (B + C) = 10 + 1 - (3 + 4) = 4$

Dibuktikan dengan citra input:

1	2
3	4

Nilai piksel pada daerah hitam = 4

Contoh 2:

Citra input

A	D	G
B	E	H
C	F	I

Citra integral

1	3	5
4	10	13
6	15	21

Jumlah nilai piksel pada daerah hitam = $F + A - (D + C) = 15 + 1 - (3 + 6) = 7$

Dibuktikan dengan jumlah piksel citra input $4 + 3 = 7$

1	2	2
3	4	1
2	3	3

2.2.3.2.1 Perbandingan perhitungan dengan citra integral dan tidak

Seperti yang sudah dijelaskan sebelumnya bahwa nilai fitur bisa diperoleh dari selisih penjumlahan daerah gelap dengan daerah terang. Untuk itu akan dibandingkan dua metode perhitungan fitur Haar-like dengan citra integral dan tanpa citra integral.

Contoh 1:

Perhitungan nilai fitur tanpa citra integral:

1	2	2	4	1
3	4	1	5	2
2	3	3	2	4
4	1	5	4	6
6	3	2	1	3

Nilai fitur = $|Sum_{black\ rectangle\ (pixel\ gray\ level)} - Sum_{white\ rectangle\ (pixel\ gray\ level)}|$

$$= |(1+2+3+4) - (2+4+1+5)|$$

$$= |10 - 12|$$

$$= 2$$

Perhitungan nilai fitur dengan citra integral dengan menambahkan *buffer* pada baris pertama dan kolom pertama dengan mengisi nilainya 0:

0	0	0	0	0	0
0	1	3	5	9	10
0	4	10	13	22	25
0	6	15	21	32	39

0	10	20	31	46	59
0	16	29	42	58	74

$$\begin{aligned}
\text{Nilai fitur} &= |Sum_{black\ rectangle\ (pixel\ gray\ level)} - Sum_{white\ rectangle\ (pixel\ gray\ level)}| \\
&= |[10+0-(0+0)] - [22+0-(10+0)]| \\
&= |10 - 12| \\
&= 2
\end{aligned}$$

Contoh 2:

Perhitungan nilai fitur tanpa citra integral:

1	2	2	4	1
3	4	1	5	2
2	3	3	2	4
4	1	5	4	6
6	3	2	1	3

$$\begin{aligned}
\text{Nilai fitur} &= |Sum_{black\ rectangle\ (pixel\ gray\ level)} - Sum_{white\ rectangle\ (pixel\ gray\ level)}| \\
&= |(4+1+3+3+1+5+3+2) - (5+2+2+4+4+6+1+3)| \\
&= |22 - 27| \\
&= 5
\end{aligned}$$

Perhitungan dengan citra integral:

1	3	5	9	10
4	10	13	22	25
6	15	21	32	39
10	20	31	46	59
16	29	42	58	74

$$\begin{aligned}
\text{Nilai fitur} &= |Sum_{black\ rectangle\ (pixel\ gray\ level)} - Sum_{white\ rectangle\ (pixel\ gray\ level)}| \\
&= |[42+1-(16+5)] - [74+5-(42+10)]| \\
&= |22 - 27| \\
&= 5
\end{aligned}$$

Contoh 3:

Perhitungan nilai fitur tanpa citra integral:

1	2	2	4	1
3	4	1	5	2
2	3	3	2	4
4	1	5	4	6
6	3	2	1	3

$$\begin{aligned}
\text{Nilai fitur} &= |Sum_{black\ rectangle\ (pixel\ gray\ level)} - Sum_{white\ rectangle\ (pixel\ gray\ level)}| \\
&= |(4+1+5)+(1+5+4) - (3+3+2)| \\
&= |10+10-8| \\
&= |20-8| \\
&= 12
\end{aligned}$$

Perhitungan dengan citra integral:

1	3	5	9	10
4	10	13	22	25
6	15	21	32	39
10	20	31	46	59
16	29	42	58	74

$$\text{Nilai fitur} = |Sum_{black\ rectangle\ (pixel\ gray\ level)} - Sum_{white\ rectangle\ (pixel\ gray\ level)}|$$

$$\begin{aligned}
&= | \{ ((22+1) - (4+9)) + ((46+6) - (10+32)) - ((32+4) - (6+22)) \} | \\
&= | 10+10 - 8 | \\
&= | 20-8 | \\
&= 12
\end{aligned}$$

Dari beberapa contoh diatas dapat dilihat bahwa perhitungan nilai fitur dengan citra integral sangat efektif tanpa harus menghitung keseluruhan nilai piksel pada daerah gelap dan daerah terang, cukup hanya dengan empat referensi array. Perhitungan fitur dengan citra integral ini akan sangat bermanfaat sekali terlebih jika jumlah piksel yang akan dihitung sangat banyak.

2.2.3.3 Algoritma AdaBoost

Algoritma Adaboost merupakan singkatan dari *Adaptive Boosting*, pertama kali diperkenalkan oleh Freund dan Schapire pada tahun 1997 [28]. Algoritma Adaboost merupakan salah satu algoritma *machine learning* yang digunakan untuk *feature selection* dan melatih *classifiers*. Pada bentuk aslinya algoritma Adaboost digunakan untuk melakukan *boosting* kinerja klasifikasi dari sebuah algoritma pembelajaran yang sederhana, seperti digunakan untuk *memboosting* kinerja *simple perceptron*. Hal tersebut dilakukan dengan mengkombinasikan sekumpulan fungsi klasifikasi lemah untuk membentuk sebuah *classifier* yang lebih kuat yang kemudian diistilahkan dengan *weak learner*. Misalkan sebuah algoritma pembelajaran perceptron menelusuri sekumpulan perceptron yang mungkin dan mengembalikan perceptron dengan kesalahan klasifikasi terendah. Pembelajar disebut lemah karena fungsi klasifikasi yang terbaik sekalipun tidak dapat mengklasifikasikan data latih dengan baik. Agar pembelajar lemah dapat ditingkatkan, maka harus dipanggil untuk memecahkan sederetan masalah pembelajaran. Setelah tahap pertama pembelajaran, beberapa sampel diberi nilai bobot ulang dengan tujuan untuk menekankan sampel-sampel tersebut yang salah diklasifikasikan oleh *classifier* lemah sebelumnya. Terakhir adalah classifier kuat mendapatkan bentuk sebuah perceptron, sebuah kombinasi berbobot dari *classifiers* lemah yang diikuti dengan sebuah nilai ambang (*threshold*).

Prosedur AdaBoost konvensional secara sederhana dapat diartikan sebagai proses seleksi fitur yang rakus. Dengan mempertimbangkan masalah umum pada kasus *boosting*, di mana satu set besar fungsi klasifikasi digabungkan menggunakan suara mayoritas yang terbobot. Tantangannya adalah untuk mengasosiasikan besar bobot masing-masing fungsi klasifikasi yang bagus dan sebuah bobot yang lebih kecil dengan fungsi yang rendah. AdaBoost merupakan mekanisme agresif untuk memilih satu set kecil fungsi klasifikasi yang bagus dengan tetap memiliki varietas yang signifikan. Yang menggambarkan sebuah analogi antara classifier lemah dan fitur, AdaBoost merupakan prosedur yang efektif untuk mencari tahu sejumlah kecil fitur yang bagus, dimana tetap memiliki varietas yang signifikan.

Salah satu metode praktis untuk menyelesaikan masalah ini yaitu dengan membatasi classifier lemah ke kumpulan masing-masing fungsi klasifikasi yang bergantung pada sebuah fitur tunggal. Untuk mendukung tercapainya tujuan ini, algoritma pembelajaran lemah didesain untuk memilih fitur persegi panjang tunggal yang terbaik dalam memisahkan sampel positif dan negatif. Untuk masing-masing fitur, *classifier* lemah menentukan fungsi klasifikasi *threshold* yang optimal sedemikian rupa sehingga jumlah minimum kesalahan klasifikasi bisa didapatkan. Sebuah classifier lemah ($h(x, f, p, \theta)$) terdiri dari sebuah fitur (f), *threshold* (θ), dan sebuah polaritas yang mengindikasikan pertidaksamaan:

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } p f(x) < p|\theta \\ 0 & \text{sebaliknya} \end{cases} \quad (6)$$

Nilai x pada percobaan Viola and Jones 24x24 piksel *subwindow* pada sebuah citra.

Dalam prakteknya, tidak ada fitur tunggal yang dapat melaksanakan tugas klasifikasi dengan kesalahan yang rendah. Fitur yang diseleksi pada proses awal memiliki tingkat kesalahan antara 0.1 dan 0.3. Fitur yang diseleksi pada tahap selanjutnya, dengan tugas yang semakin sulit, memiliki tingkat kesalahan antara 0.4 dan 0.5.

Berikut langkah-langkah algoritma AdaBoost:

Tabel 3 Algoritma *boosting* untuk *learning query* secara online [22]

- Diberikan sampel citra $(x_1, y_1), \dots, (x_n, y_n)$ dimana $y_i=0,1$ untuk sampel negatif dan positif berturut-turut.
- Inisialisasikan bobot $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ untuk $y_i = 0,1$ berturut-turut, dimana m dan l adalah jumlah positif dan negatif berturut-turut.
- Untuk $t = 1, \dots, T$:

1. Normalisasikan bobot,

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$$

Sehingga w_t adalah distribusi probabilitas.

2. Pilih classifier lemah terbaik dengan urutan ke bobot kesalahan

$$\epsilon_t = \min_{f,p,\theta} \sum w_i |h(x_i, f, p, \theta) - y_i|.$$

3. Definisikan $h_t(x)=h(x,f_b,p_b,\theta_t)$ dimana f_b, p_b , dan θ_t adalah bentuk minimize dari ϵ_t .

4. Update bobot:

$$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$

dimana $e_i = 0$ jika sampel x_i diklasifikasikan dengan benar, $e_i = 1$ sebaliknya, dan $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.

1. Terakhir classifier kuat adalah:

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ \text{sebaliknya} & \end{cases}$$

$$\text{dimana } \alpha_t = \log \frac{1}{\beta_t}$$

Tabel diatas menjelaskan algoritma yang digunakan untuk menyeleksi *classifier* lemah kunci dari sekumpulan *classifier* lemah yang mungkin [25].

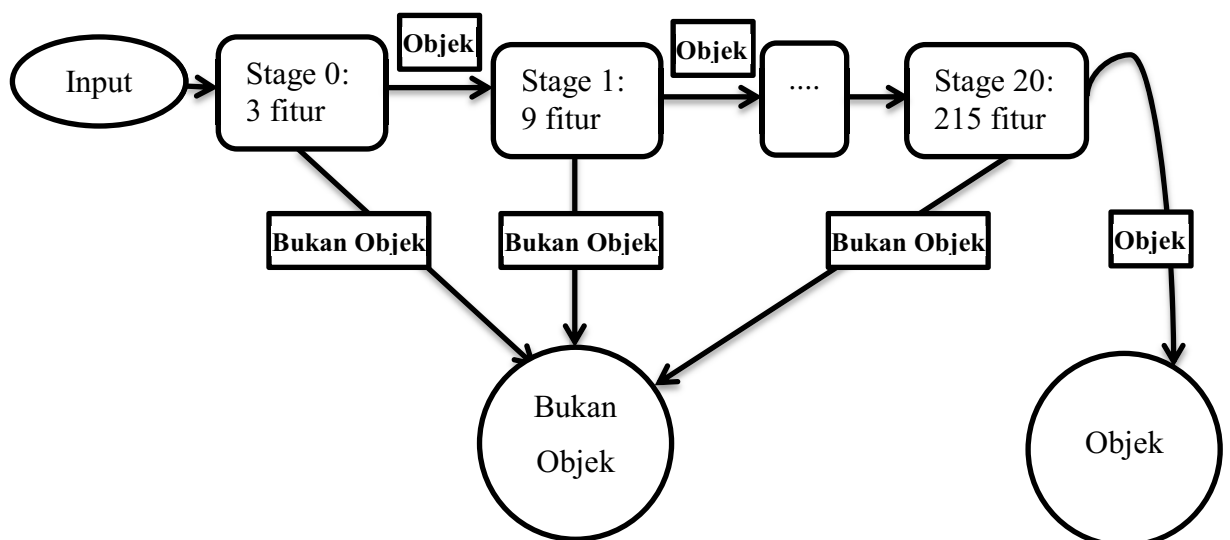
2.2.3.4 Cascade Classifier

Cascade classifier digunakan untuk mencapai peningkatan deteksi secara radikal dengan mengurangi waktu komputasi. *Boosted classifier* dapat dibangun dengan

menolak beberapa *subwindow* yang negatif ketika mendeteksi hampir semua sampel positif (nilai *threshold boosted classifier* dapat disesuaikan, sehingga tingkat negatif palsu (*false negative rate*) mendekati nol). *Classifier* sederhana digunakan untuk menolak sebagian besar *subwindows* sebelum *classifiers* yang lebih kompleks dipanggil untuk mencapai tingkat positif palsu (*false positive rate*) yang rendah.

Bentuk keseluruhan proses deteksi berupa degenerasi pohon keputusan, yang disebut *cascade*. Hasil positif dari *classifier* pertama memicu evaluasi *classifier* kedua yang juga telah disesuaikan untuk mencapai tingkat deteksi yang sangat tinggi. Hasil positif dari *classifier* kedua memicu *classifier* ketiga, dan seterusnya. Hasil negatif pada titik manapun mengarah ke penolakan langsung dari *subwindow*.

Tahapan dalam *cascade* dibangun oleh *training classifier* menggunakan AdaBoost dan kemudian menyesuaikan ambang batas untuk meminimalkan negatif palsu. Nilai *threshold* AdaBoost dirancang untuk menghasilkan tingkat kesalahan rendah pada *data training*. Secara umum nilai *threshold* yang lebih rendah memberikan hasil tingkat deteksi lebih tinggi dan tingkat positif palsu yang lebih tinggi.



Gambar 6 Struktur *cascade classifiers*

Serangkaian *classifiers* diterapkan pada setiap *subwindow*. *Classifier* awal menghilangkan sejumlah besar sampel negatif dengan sedikit pemrosesan. Lapisan berikutnya menghilangkan sampel negatif tambahan namun memerlukan perhitungan tambahan. Setelah beberapa tahap pemrosesan, jumlah *subwindow* telah dikurangi

secara radikal. Pemrosesan lebih lanjut dapat melakukan hal apapun seperti tahap penambahan dari *cascade*.

Classifier tahap pertama dapat dibangun dari dua fitur *classifier* yang kuat dengan mengurangi *threshold* untuk meminimalkan negatif palsu. Diukur terhadap *training set* validasi, *threshold* dapat disesuaikan untuk mendeteksi 100% dari objek (misal:wajah) dengan tingkat positif palsu sebesar 40%.

Proses pelatihan *cascade* melibatkan dua jenis pengorbanan. Dalam kebanyakan kasus *classifier* dengan lebih banyak fitur akan mencapai tingkat deteksi lebih tinggi dan tingkat positif palsu rendah. Pada saat yang sama *classifier* dengan lebih banyak fitur memerlukan lebih banyak waktu untuk menghitung. Pada prinsipnya kita dapat mendefinisikan kerangka optimasi dengan mengatur jumlah tahap *classifier*, jumlah fitur di setiap tahap, dan nilai ambang setiap tahap, ditukar untuk meminimalkan jumlah yang diharapkan dari dievaluasi fitur. Namun sayangnya untuk bisa menemukan nilai optimum ini merupakan masalah sangat sulit.

Dalam prakteknya kerangka sederhana yang digunakan untuk menghasilkan *classifier* yang efektif dan efisien. Setiap tahap dalam *cascade* harus mengurangi tingkat positif palsu dan mengurangi tingkat deteksi. Target dipilih untuk pengurangan nilai minimum positif palsu dan penurunan maksimum dalam deteksi. Setiap tahap dilatih dengan menambahkan fitur sampai deteksi target dan tingkat positif palsu terpenuhi. *Stages* ditambahkan sampai keseluruhan target untuk positif palsu dan tingkat deteksi terpenuhi.

Detektor akhir dipindai di seluruh gambar dalam berbagai skala dan lokasi. Penskalaan dicapai dengan skala detektor itu sendiri, bukan skala gambar. Proses ini masuk akal karena fitur dapat dievaluasi pada skala apapun dengan nilai yang sama. Hasil deteksi yang baik diperoleh dengan menggunakan seperangkat skala faktor 1,25 secara terpisah.

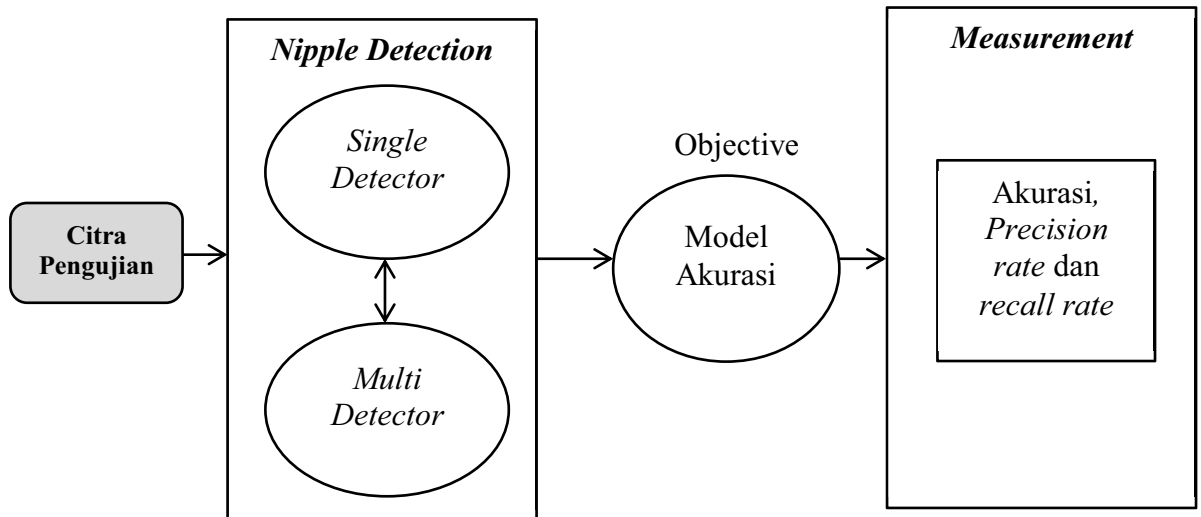
Detektor juga dipindai di seluruh lokasi. Lokasi berikutnya diperoleh dengan menggeser *window* sejumlah beberapa pixel Δ . Proses pergeseran ini dipengaruhi oleh skala detektor: jika skala saat ini adalah s *window* digeser oleh $[s\Delta]$, di mana $[]$ adalah operasi pembulatan.

Pemilihan $\triangle$ sangat berpengaruh baik pada kecepatan serta ketepatan detektor. Karena gambar pelatihan memiliki beberapa variabilitas translasi detektor belajar yang mencapai kinerja deteksi yang bagus terlepas dari pergeseran kecil dalam gambar. Akibatnya detektor *subwindow* dapat digeser lebih dari satu piksel setiap kali. Namun, ukuran langkah lebih dari satu pixel cenderung menurunkan tingkat deteksi lebih sedikit dan juga mengurangi jumlah positif palsu [25].

2.2.4 Theoretical Model

Skema *theoretical model* ditunjukkan oleh Gambar 7, penelitian bermula dari banyaknya metode untuk mendeteksi citra pornografi namun hanya sebatas pada perkiraan luasan persentase kulit pada objek yang akan dideteksi. Jika nilai persentase lebih besar dari nilai *threshold* yang ditentukan, maka akan dianggap sebagai citra pornografi. Pada kenyataannya terjadi beberapa kekeliruan antara lain citra yang memiliki luasan area kulit lebih dominan dibanding *background*-nya akan dianggap sebagai citra porno, begitu juga area citra yang memiliki luasan warna yang memiliki kesamaan dengan warna kulit akan dianggap sebagai citra pornografi juga dan sebaliknya citra yang mengandung unsur pornografi akan tetapi dengan area kulit yang lebih sedikit maka dianggap bukan citra pornografi, dan ini merupakan suatu kesalahan fatal. Salah satu kekeliruan yang mendasar pada metode tersebut adalah tidak melakukan deteksi pada organ intim citra pornografi. Padahal sebuah citra dianggap porno adalah apabila menampilkan organ intim dengan tujuan untuk membangkitkan gairah [17]. Berdasarkan studi literatur diketahui bahwa bahwa hal tersebut dapat dicarikan solusi dengan deteksi organ intim menggunakan algoritma Viola and Jones. Algoritma ini sangat cepat dan *robust* dalam mendeteksi objek. Akan tetapi, masih didapatkan banyak *region* palsu pada citra yang dideteksi. Oleh karena itu, perlu dilakukan dua kali proses deteksi untuk mengurangi jumlah *region* palsu dan meningkatkan akurasi deteksi. Deteksi pertama dengan mendapatkan area payudara sebagai lokasi yang menampung puting yang akan dijadikan *Region of Interest* (ROI). Setelah didapatkan area payudara, dilanjutkan dengan mendeteksi lokasi puting pada area payudara tersebut. Diharapkan dengan multi deteksi akan didapatkan hasil yang lebih akurat.

Tujuan dari penelitian ini adalah meningkatkan model akurasi deteksi algoritma Viola and Jones dalam mendeteksi puting payudara. Pengukuran (*measurement*) kinerja akurasi metode yang diusulkan menggunakan parameter pengukuran *precision rate* dan *recall rate*. Dari hasil pengujian kita akan mengetahui berapa *precision rate* dan *recall rate* serta akurasi dari kedua metode.



Gambar 7 *Theoretical Model*