BAB II

LANDASAN TEORI

* 1. Penelitian Terdahulu

Sebelum melakukan perancangan aplikasi diperlukan beberapa studi literatur yang berkaitan dengan tema yang penulis angkat untuk mengetahui efektivitas penggunaan aplikasi identifikasi karies gigi tersebut.

Radiah, dkk (2013) mengatakan bahwa faktor yang paling penting dalam kehidupan manusia adalah kesehatan. Salah satu faktor kesehatan yang perlu diperhatikan oleh masyarakat yaitu kesehatan gigi dan mulut. Diantara masalah kesehatan gigi dan mulut yang ada, banyak yang terjadi kasus penyakit karies. Karies yaitu suatu penyakit yang terjadi di jaringan keras gigi berupa email, dentin dan sementum. Penyebabnya yaitu aktivitas jasad renik yang ada dalam suatu karbohodrat yang diragikan. Tahapan dari karies dimulai dengan adanya plak di permukaan gigi. Aspek-aspek penunjang untuk kesehatan gigi dari lingkungan, pengetahuan, pendidikan, kesadaran dan penanganan kesehatan gigi termasuk pencegahan dan perawatan perlu ditinjau lebih lanjut. Berdasarkakan penelitian tersebut, didapatkan bahwa mungkin banyaknya gigi yang mengalami karies terjadi karena adanya penuaan, jadi kemungkinkan terjadinya karies semakin meningkat. Disebutkan pula dalam penelitian dari Sihombing mengenai karakteristik penderita karies gigi di RSUD Dr. Pirngadi, Medan, seseorang usia lebih dari 14 tahun mengakibatkan waktu paparan makanan yang mengandung gula dan bersifat kariogenik terhadap gigi semakin lama, menjadikan mudah terjadi demineralisasi dan dapat terjadi karies jika tidak didukung oleh kebersihan gigi dan mulut. Penelitian tersebut didapati proporsi terbesar penyakit karies gigi pada pelajar atau mahasiswa yaitu 42,1%. Penggunaan gigi molar sebagai pengunyah makanan menjadikan gigi molar sebagai gigi yang paling sering mengalami karies karena terdapat penumpukan sisa-sisa makanan penyebab karies. Letaknya dibagian belakang yang paling sering mengalami banyak pit dan fisur sehingga plak dan bakteri mudah menumpuk pada gigi ini dan dapat menyebabkan karies [14].

Bangun (2010) meneliti tentang penggunakan pengolaha citra berupa gigi untuk mengenali adanya karies pada gigi dengan menggunakan metode deteksi tepi Canny, dimana metode ini cukup handal terhadap gangguan dibandingkan dengan metode lainnya. Saat dilakukan analisis parameter, kualitas struktur garis tepi yang diperoleh dari hasil pengujian bahwa citra karies gigi setelah dilakukan pengolahan dengan metode deteksi tepi Canny bentuk garis tepi yang dihasilkan menjadi putus-putus, berupa adanya bagian garis yang hilang, sedangkan struktur garis tepi untuk gigi sehat akan tampak garis tepi tersebut menyatu dan garis tepi tidak banyak yang hilang[1].

Karyati (2013) memaparkan keuntungan dari penggunaan citra medis adalah pengetahuan yang berkaitan dengan anatomi atau kondisi tubuh manusia, yaitu berkaitan dengan sesuatu yang biasanya terjadi dan apa yang tidak biasanya terjadi dengan memberikan pengetahuan kepada kita dan mempermudah kita untuk membedakan citra normal dengan citra abnormal. Zat atau sebuah material berupa *contrast agent* apabila dimasukkan kedalam organ tertentu pada manusia dapat digunakan untuk meningkatkan kualitas dari kontras gambar yang biasa dipelajari dalam bidang pencitraan medis biologi (*biological medical imaging*). Kebenaran dalam pengolahan citra medis dan penggunaan tekniknya sebagai bagian utama dari aplikasi medis. Metode *post-processing* haruslah kuat, dapat diandalkan, dan apabila memungkinkan dengan mensimulasikan pengolahan secara manual. Metode *post-processing* digunakan untuk mendapatkan proses pengolahan yang benar, tidak hanya mensimulasikan perilaku manusia. Ketika melakukan proses validasi dapat dengan menggunakan acuan standar yang ada yang berupa gambar atau data serta dengan proses manual yang dilakukan oleh ahli. Pengolahan citra medis sebagai sarana untuk melakukan spesifikasi terhadap masalah peningkatan sumber data medis untuk keperluan visualisasi acuan analisis selanjutnya secara khusus. Dengan adanya analisis citra medis dapat bermanfaat sebagai pengembangan teknik untuk melengkapi penilaian dari citra medis yang diperoleh dari para ahli dengan berdasarkan subjektif ahli dan dapat untuk menghasilkan informasi baru yang sesuai dengan hasil sebenarnya dan dapat diaplikasikan sebagai penelitian berikutnya[8].

* 1. Karies Gigi

Karies gigi adalah penyakit yang merusak bagian jaringan keras gigi yang ditandai dengan adanya demineralisasi (proses menghilangnya ion-ion mineral pada email gigi) komponen anorganik gigi, diikuti oleh hancurnya matriks organik gigi (Kidd dan Bechal, 1992). Demineralisasi bisa saja terjadi jika terdapat mikroorganisme yang melakukan metabolisme karbohidrat dan menghasilkan asam pada rongga mulut.

Penyebab terjadinya karies bukan hanya berasal dari satu jenis kejadian seperti penyakit pada umumnya sebagai pemicu terbentuknya karies, akan tetapi terdapat beberapa faktor yang mempengaruhi hal tersebut sehingga dapat terbentuk penyakit karies. Untuk dapat terjadi karies, terdapat tiga faktor penunjang utama yang saling berinteraksi di dalam rongga mulut yaitu faktor *host* berupa gigi, *agent* berupa mikroorganisme, lingkungan berupa substrat, dan juga waktu.

Gigi desidui dapat dikatakan sebagai gigi susu atau gigi sulung, yang merupakan gigi yang pertama kali tumbuh dalam masa pertumbuhan seseorang, dimana apabila gigi desidui ini tanggal akan digantikan dengan gigi permanen. Urutan serangan karies pada gigi desidui yang mudah terjadi karies mempunyai karakteristik urutan yang spesifik yaitu menyerang gigi molar desidui mandibula yang merupakan gigi pada rahang bawah, gigi molar desidui maksila yang merupakan gigi bagian belakang rahang atas sebagai tempat mengunyah makanan dan gigi anterior desidui maksila yang merupakan gigi bagian depan yang ada pada rahang atas. Tingkat terjadinya karies sangat jarang ditemui pada gigi anterior desidui mandibula (gigi susu depan pada rahang bawah) pada permukaan fasialnya (muka) dan lingualnya (bagian lidah) gigi desidui, tetapi bisa saja terjadi ketika terdapat karies rampan (karies yang terjadi secara tiba-tiba) atau nursing karies (karies botol).

Permukaan oklusal pada gigi molar pertama desidui baik maksila maupun mandibula mempunyai ketahanan yang lebih baik terhadap serangan karies dibandingkan dengan gigi molar kedua desidui. Morfologi permukaan oklusal akan mempengaruhi keadaan ini. Karies dapat terjadi pada antara gigi geligi secara penuh apabila pada permukaan interproksimal (sela gigi) pada gigi desidui baik anterior (gigi depan) maupun pada posterior (gigi belakang) terjadi kontak secara proksimal [21].

Tahap awal pada saat akan terjadinya karies gigi dapat dilihat dengan adanya *white spot* sebagai tanda sebelum terjadinya tahapan lesi gigi (keadaan abnormal gigi) pada kavitas (rusaknya atau berlubangnya gigi). Hal ini bisa terbentuk apabila memungkinkan untuk terjadi pengembangan dari *white-spot* tersebut. Pada kebanyakan kasus yang terjadi, biasanya diperlukan waktu 2 hingga 4 tahun untuk karies tersebut mengenai email menuju dentin pada bagian approximal (permukaan mesial gigi). Karies dapat digambarkan seperti bentuk gunung es dengan bagian nampak yang secara klinis dimulai dengan lesi *white-spot* berada di atas garis air. Bawah permukaan air tidak nampak secara kasat mata akan adanya lesi yang memerlukan beberapa alat untuk dapat dilakukan identifiaksi. Tanda pertama terliihatnya karies adalah lesi *white-spot*, yang mana akan terlihat lebih jelas ketika permukaan dikeringkan. Hal ini terjadi karena enamel ketika demineralisasi enamel akan berpori, pori-pori ini mengandung air yang apabila kering air di pori-pori diganti dengan udara dan lesi menjadi lebih jelas. Beberapa alat sebagai pendiagnosa karies seperti halnya bitewings, orthopantomogram (OPT), bimolar, periapicals. Bitwings sendiri merupakan pendiagnosa karies yang disarankan untuk digunakan karena dapat memberikan informasi mengenai karies baik berupa dentin oklusal gigi maupun aproksimal enamel dan karies gigi. OPT mempunyai penilaian yang lebih rendah dalam mendeteksi lesi aproksimal, sedangkan bimolar tidak begitu dapat diandalkan untuk memberikan diagnosa yang dikarenakan sering terjadi tumpang tindih pada strukturnya. Peripicals dapat melakukan diagnosa seakurat bitewings untuk mendeteksi karies, tetapi informasi yang didapat lebih sedikit pada bagian filmnya[23].

Klasifikasi karies menurut G.J Mount, dkk (2016) :

1. Berdasarkan *Site* (lokasi)
2. *Site* 1 : karies terletak pada pit dan fissure.
3. *Site* 2 : karies terletak di area kontak gigi (proksimal), baik anterior maupun posterior.
4. *Site* 3 : karies terletak di daerah servikal, termassuk enamel/permukaan akar yang terbuka.
5. Berdasarkan *Size* (size) ; jika kavitas berkembang dari lesi bercak putih menjadi kavitas berlanjut sehingga menghancurkan mahkota gigi. Mahkota tersebut diklasifikasikan menjadi:
6. *Size* 0 : lesi dini, awal demineralisasi.
7. *Size* 1 : kavitas minimal, melibatkan dentin namun belum terjadi. Kavitas yang masih minim dapat dilakukan perawatan remineralisasi.
8. *Size* 2 : size kavitas sedang, dimana masih terdapat struktur gigi yang cukup untuk dapat menyangga restorasi yang akan ditempatkan.
9. *Size* 3 : kavitas yang ber-*size* lebih besar, sehingga preparasi kavitas di perluas agar restorasi dapat digunakan untuk melindungi struktur gigi yang tersisa dari retak/patah.
10. *Size* 4 : sudah terjadi kehilangan sebagian besar struktur gigi seperti cups/sudut insisal [9].
    1. Citra

Citra merupakan suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra dihasilkan dari suatu sistem perekaman atau dengan dibuat suatu citra yang mana dapat dibagi menjadi beberapa jenis berdasarkan media keluarannya. Keluaran dapat bersifat optik berupa foto serta berupa penglihatan manusia, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi yang dihasilkan dari perangkat yang bersifat analog, dan bersifat digital dimana pada sifat ini citra dapat langsung disimpan pada suatu media penyimpanan ataupun dapat dilakukan suatu pengolahan pada citra tersebut.

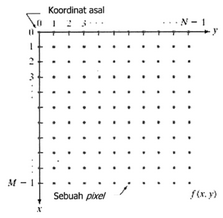
Citra analog mempunyai sifat kontinue yaitu berupa sinyal yang tidak terputus karena tidak adanya *sampling* (pengambilan sebagian citra pada titik tertentu). Sebagai contoh citra analog yaitu foto yang dicetak pada kertas, gambar pada monitor televisi, hasil foto menggunakan sinar-x, objek nyata di dunia, gambar lukisan, dan hasil cetak CT *scan*. Citra analog dapat dihasilkan dari peralatan analog, seperti halnya kamera yang masih menggunakan pita, pencetak CT *scan*, sistem radar, USG, dan sebagainya. Peralatan citra analog ini tidak menghasilkan citra dalam bentuk sinyal digital sehingga komputer tidak dapat melakukan pemrosesan citra dalam bentuk citra analog secara langsung sebelum dilakukan pengubahan kedalam bentuk digital.

Sedangkan citra digital sendiri merupakan citra yang dapat diolah oleh komputer karena citra ini berbentuk sinyal digital. Gambar 2.1. menunjukkan bentuk koordinat citra skala keabuan, dengan ukuran 150x150 piksel adalah elemen terkecil dari sebuah citra yang digunakan, kemudian diambil sebagian dengan menggunakan komputer dengan ukuran piksel yang diambil yaitu 9x9 piksel. Sehingga yang akan tertampil pada layar monitor adalah bentuk citra berukuran sangat kecil dengan bentuk persegi. Dalam citra digital seperti ini, komputer menyimpan informasi berbentuk data biner, sehingga hasil yang tertampil pada layar monitor dengan direpresentasikannya besar intensitas piksel yang merupakan hasil pengolahan dari komputer itu sendiri[20].

Menurut Putra (2010), pengolahan citra digital merupakan pemrosesan gambar 2 dimensi yang dilakukan menggunakan komputer. Apabila diartikan lebih luas pengolahan citra digital mengacu pada citra digital berbentuk data 2 dimensi dimana citra tersebut berupa sebuah larik (*array*) yang terdiri dari nilai bit-bit dari intensitas suatu piksel yang dilakukan pemrosesan. Pembentukan citra digital (diskrit) sendiri melalui beberapa tahapan yaitu akuisisi citra, *sampling*,dan kuantisasi[13].

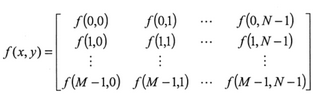
Urutan dari suatu citra berupa piksel-piksel tersebut dapat dituliskan ukurannya dengan simbol M untuk baris dan N sebagai ukuran kolom. Fungsi yang digunakan yaitu f(x,y) dengan x dan y merupakan koordinat yang berupa larik sesuai dengan jarak tertentu. Tingkat intensitas dari suatu citra pada titik dalam koordinat (x,y) merupakan sebuah amplitudo *f*. Citra dapat dikatakan berbentuk digital apabila citra tersebut menunjukkan nilai yang besarnya x,y dan amplitudo f berupa nilai yang mempunyai batasan tertentu atau berhingga.

Apabila nilai x,y, dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Gambar 2.1 menunjukkan posisi koordinat citra digital.



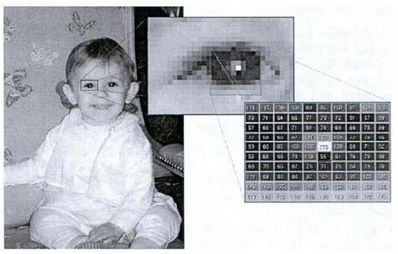
Gambar 2.1 Koordinat citra digital.

Penulisan fungsi citra digital dalam bentuk matriknya adalah sebagai berikut.



Piksel (*pixel / picture element*) merupakan istilah yang sering dijumpai dalam dunia digital, piksel dapat juga disebut dengan *image elements* atau *pels* yang merupakan komponen gambar atau representasi sebuah titik terkecil dalam sebuah gambar grafis, juga merupakan nilai intensitas yang saling bertemu antara baris dan kolom (pada posisi x,y).

Gambar 2.2 menampilkan contoh dari suatu citra digital dalam bentuk citra skala keabuan serta ditampilkan pula nilai intensitas dari citra tersebut pada area yang dibatasi dengan ukuran piksel 10x10 sebagai contoh untuk mengetahui besarnya nilai intensitas pada ukuran bidang tertentu, tidak secara keseluruhan.



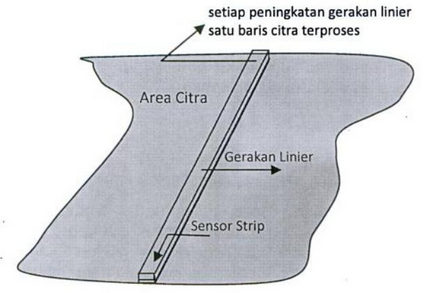
Gambar 2.2 Contoh citra dengan skala keabuan dan nilai intensitas pada cuplikan di bidang tertentu[10].

* + 1. Akuisisi citra

Proses akuisisi citra merupakan tahapan awal untuk mendapatkan citra digital. Akuisisi citra dapat didefinisikan sebagai proses pengambilan ataupun pengumpulan data suatu pandangan yang masih bersifat citra analog menjadi citra kontinu dengan menggunakan alat sensor, diperlukan pula respresentasi secara numerik dengan nilai diskrit agar dapat diolah dengan komputer secara digital. Alat untuk melakukan digitalisasi biasa disebut digitalizer yang berupa *scanner* maupun berupa kamera. Proses akuisisi citra dapat menggunakan beberapa jenis sensor, yaitu berupa sensor tunggal (*single sensor*), sensor garis (*sensor strip*), dan sensor larik (*sensor array*).

1. Photodiodode merupakan sensor tunggal yang paling familiar yang dibentuk dari silikon dengan tegangan keluaran yang sebanding dengan cahaya. Sensor Photodiode dapat digunakan untuk menciptakan citra 2 dimensi melalui proses pemindahan relatif di setiap sumbu x dan y antara sensor dan objeknya. Negatif film merupakan penyimpanan hasil dari proses akuisisi data citra yang digulung pada dinding tabung. Tabung tersebut akan berotasi. Gerakan rotasi tabung ini merupakan perpindahan dalam sumbu x. Metode ini tidak mahal tetapi memakan waktu yang cukup lama dalam melakukan akuisisi.
2. Sensor garis

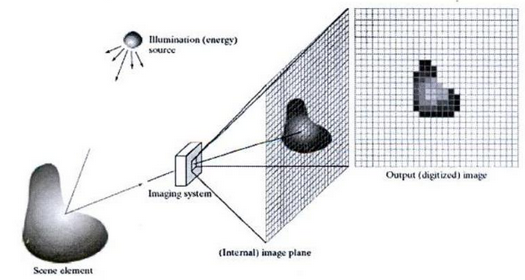
Sensor yang dapat melakukan akuisis sumbu x secara bersamaan karena disatukan dalam satu baris. Pencitraan yang dilakukan sensor garis merupakan sensor satu arah. Proses akuisisi citra dilakukan dengan menggerakan sensor searah dengan sumby y. mesin scanner merupakan salah satu alat yang memanfaatkan sensor garis. Gambar 2.3 menunjukkan sensor yang menggunakan pencitraan satu arah.



Gambar 2.3 Pemanfaatan sensor garis.

1. Sensor Larik

Sensor larik merupakan sensor yang dibentuk dari larik 2 dimensi. sensor larik sering ditemui pada kamera digital, pada kamera digital sensor ini disebut sensor ccd. Ukuran sensor rata-rata dapat mencapai 4000 x 4000 elemen.. Penggunaan dari sensor ini diperlihatkan pada Gambar 2.4.

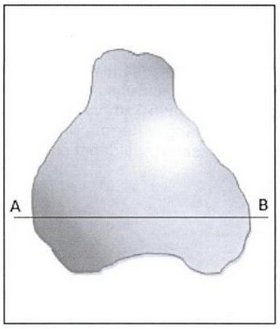


Gambar 2.4 Pemanfaatan sensor larik.

Sensor menangkap setiap iluminasi yang dipantulkan oleh objek kemudian diproyeksikan ke dalam bidang citra. Secara bersamaan sensor larik akan menghasilkan keluaran yang setara dengan integral dari cahaya yang diterima setiap sensor.

* + 1. Sampling

Tahap berikutnya setelah citra kontinu terbentuk adalah proses sampling untuk memmbentuk citra digital. Proses sampling merupakan proses digitasi pada koordinat x, y, hasil dari sensor masih berupa citra kontinu, sinyal kontinu pada nilai x, y dan juga amplitudonya (intensitas). Nilai x dan y yang kontinu akan diubah menjadi bentuk diskrit, yang ditunjukkan pada Gambar 2.5.



x

y

Gambar 2.5 scan pada baris AB.

Gambar 2.6 merupakan sinyal kontinu sepanjang titik a dan b yang diambil untuk selanjutnya dilakukan sampling.

x

y

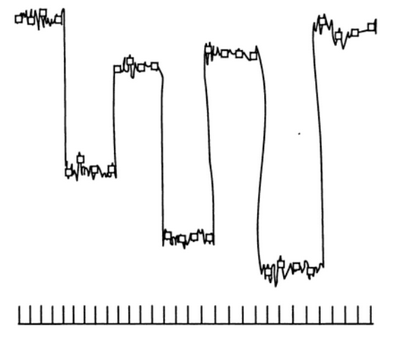


Gambar 2.6 Sinyal kontinu AB.

Proses sampling akan memberikan nilai koordinat diskrit pada setiap posisi di koordinat kontinu seperti Gambar 2.7. Koordinat diskrit ditandai dengan titik putih sepanjang sinyal analog.

x

y



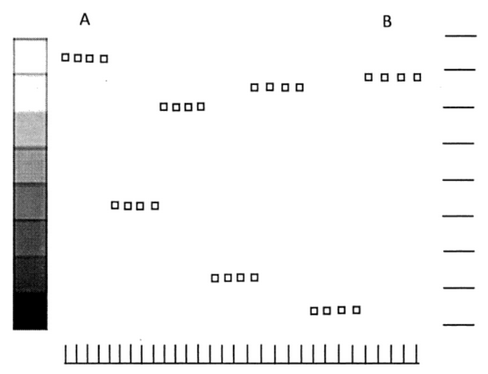
Gambar 2.7 Proses sampling.

* + 1. Kuantisasi

Proses kuantisasi merupakan perubahan nilai amplitudo kontinu menjadi nilai diskrit. Nilai amplitudo yang dikuantisasi adalah nilai-nilai hasil proses sampling pada koordinat diskrit. Gambar 2.8 menampilkan contoh proses kuantisasi ke dalam warna *grayscale* 8 level. Sinyal kontinu yang telah dikuantisasiakan memiliki sesuai dengan nilai level digitalnya yaitu dari 1 – 8.

x

y



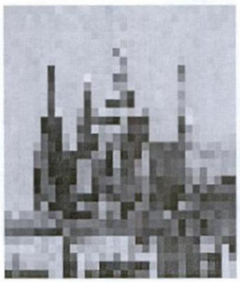
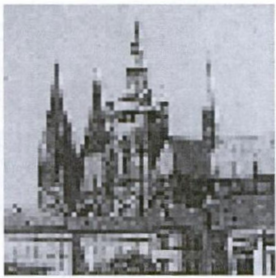
Gambar 2.8 hasil sampling dan kuantisasi.

Ilustrasi perbedaan proses sampling dan kuantisasi ditunjukkan pada Gambar 2.9 dan Gambar 2.10.



(a)

(b)

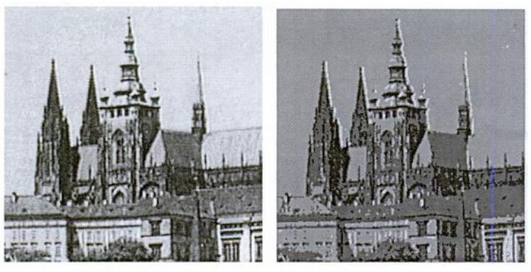


(c)

(d)

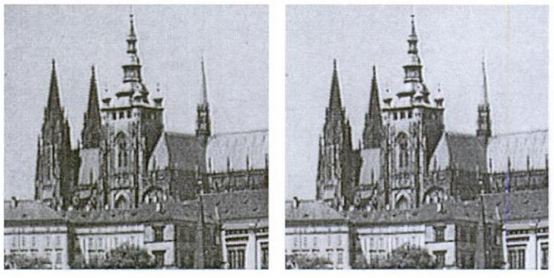
G

Gambar 2.9 Proses sampling (a) Citra asli 256 x 256 pixel (b) Samping 128 x 128 (c) samping 64 x 64 (d) samping 32 x 32



(a)

(b)



(c)

(d)

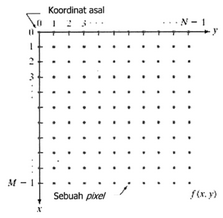


(e)

Gambar 2.10 Proses kuantisasi (a) Citra asli 256 level warna (b) Kuantisasi 64 level (c) Kuantisasi 16 level (d) Kuantisasi 4 level (e) Kuantisasi 2 level

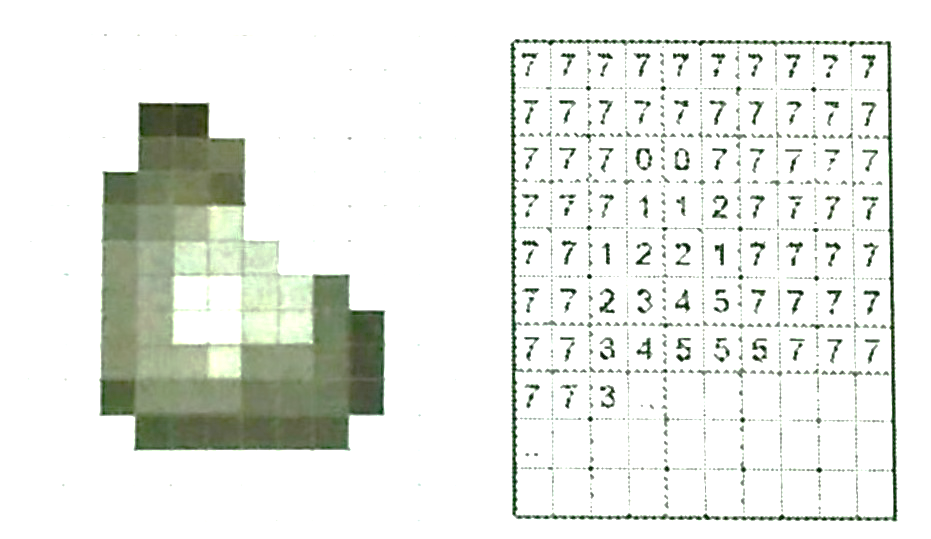
* + 1. Representasi Citra Digital

Hasil dari *sampling* dan digitalisasi adalah sebuah matriks dari bilangan-bilangan riil. Dimisalkan sebuah citra *f(x,y)* dilakukan proses sampling sehingga menghasilkan gambar digital dengan M baris dan N kolom. Nilai dari kordinat (x,y) sekarang berubah menjadi kuantitas diskrit, yang pada umumnya dinyatakan dengan bilangan bulat. Nilai koordinat di titik asal (*origin*) adalah (*x,y*) = (0,0). Nilai koordinat berikutnya, sepanjang baris pertama dari citra dinyatakan sebagai (x,y) = (0,1). Gambar 2.11 menunjukkan konvensi koordinat yang digunakan.



Gambar 2.11 Aturan koordinat yang digunakan.

Sebuah gambar digital lengkap ukuran *M* x *N* dapat ditulis dalam bentuk matriks berikut ini :



Gambar 2.12 Citra Digital dan representasinya.

Gambar 2.12 menunjukkan bentuk representasi citra digital dalam bentuk matriks denggan tingkat kecerahan 8.

Elemen-elemen dari matriks di atas disebut *image element, picture element, pixel* atau *pel*. Sebuah piksel (*pixel*) mempunyai dua properti yaitu koordinat atau posisi dari piksel tersebut dan nilai dari piksel tersebut. Sehingga sebuah piksel biasa dinyatakan sebagai fungsi dua dimensi f(x,y). Contoh sebuah piksel f(0,2)-7, berarti piksel tersebut berada pada baris 0 dan kolom 2, dengan nilai intensitas kecerahan = 7.

Dalam proses digitalisasi perlu ditetapkan nilai *M., N* dan *L* (level keabuan yang diperbolehkan untuk tiap *pixel*). Dalam menentukan Nilai *M* dan *N* tidak terdapat ketentuan atau bebas namun harus berupa bilangan bulat positif. Dengan mempertimbangkan penggunaan perangkat keras untuk pengolahan, penyimpanan dan sampling. Jumlah *gray level* atau level keabuan merupakan 2 pangkat bilangan bulat (L=2k). Ini berarti nilai level dari elemen citra (piksel) mempunyai range antara |0,L-1|. Sebagai contoh, sebuah citra yang mempunyai level keabuan L= 8, berarti nilai intensitas keabuan dari setiap pikselnya mempunyai *range* antara 0-7.

Jumlah bit yang dibutuhkan untuk menyimpan gambar digital adalah b=MxNxk. Dimana M merupakan jumlah baris citra, N meruapakan jumlah kolam citra dan k merupakan jumlah bit yang dibutuhkan untuk menyatakan sebuah nilai keabuan.

* + 1. Resolusi Spasial dan *Grey Level (Brightness*)

Terdapat beberapa faktor yang mempengaruhi besarnya informasi citra yang hilang, salah satunya yang memperngarui yaitu resolusi spasial dan resolusi kecemerlangan.

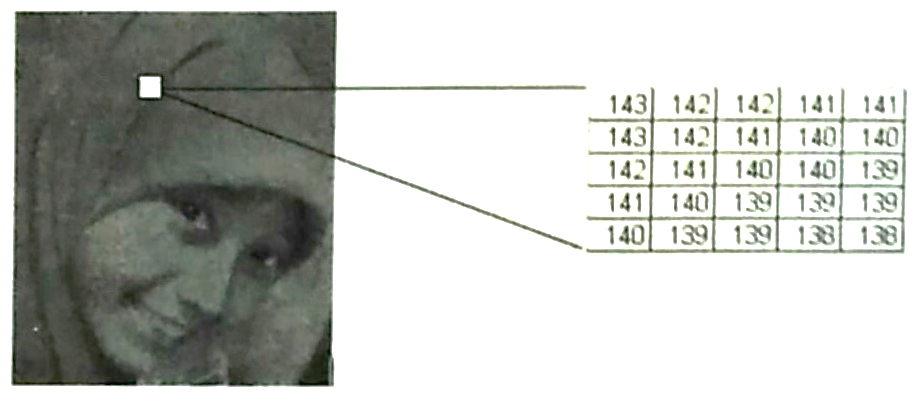
Resolusi spasial merupakan halus atau kasarnya pembagian kisi-kisi baris dan kolom. Transformasi citra kontinyu ke citra digitasi juga disebut sebagai digitasi (*sampling*). Sebuah gambar digital *L*-level dengan ukuran *M*x*N* mempunyai resolusi spasial (*spacial resolution*) = *M*x*N* pixel, serta resolusi level keabuan (*grey-level resolution*) = *L*-level. Hasil digitasi dengan jumlah baris 256 dan jumlah kolom 256 mempunyai resolusi spasial 256 x 256. Variasi jumlah sample dalam bidang digital image processing mengakibatkan perlunya pengolahan digital image dari suatu komputer digital. Gambar dihasilkan dari seluruh spektrum elektromagnetik mulai dari gamma sampai gelombang radio.

Level pengolahan gambar dapat dibagi menjadi tiga level, yaitu *low-level process* yang meliputi operasi dasar seperti *image preprocessing* berupa *reduce nouse, contrast enhancement,* dan *image sharpening.* Pada level ini input atau sumber masukan dan output atau hasil keluaran tetap berupa gambar. *Mid-level process* meliputi segmentasi (membagi sebuah gambar dalam region atau *object*), mendeskripsikan objek tersebut untuk direduksi dalam bentuk yang diinginkan dan klasifikasi (*recognition*) dari objek tersebut. *Input* atau masukkan dari proses ini adalah berupa gambar, dan *output*-nya berupa atribut yang diambil dari gambar tersebut contohnya *edge, counturs* dan identitas dari objek tertentu. *High-level process* meliputi pemberian arti dari suatu rangkaian objek-objek yang dikenali dan akhirnya menampilkan fungsi-fungsi kognitif secara normal sehubungan dengan penglihatan

* + 1. Konsep Citra *Grey* dan Warna

Nilai yang disediakan pada citra *grey* berjumlah 256, karena jumlah bit pada citra *grey* adalah 8, sehingga jumlah warna yang dihasilkan adalah 28 = 256, nilainya berada pada jangkauan 0 – 255. Model penyimpanan pada citra *grey* adalah *f(x,y)*=nilai intensitas, dengan x dan y merupakan posisi nilai intensitas.

Sebagai ilustrasi untuk citra *grey* yang diambil pada titik tertentu dengan lebar 7 dan tinggi 7 dapat dilihat pada Gambar 2.13.

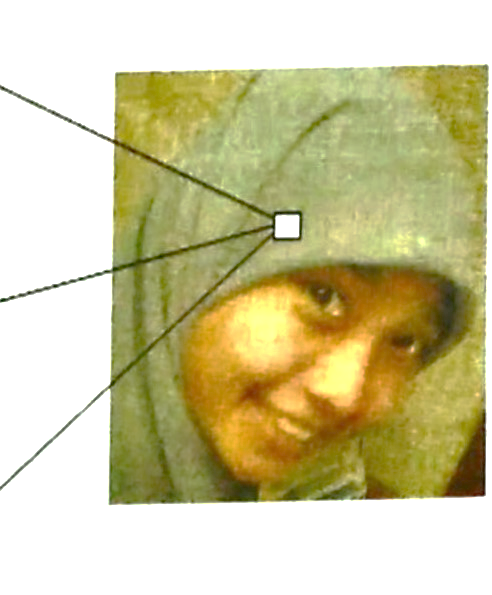


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 143 | 142 | 142 | 141 | 141 |
| 143 | 142 | 141 | 140 | 140 |
| 142 | 141 | 140 | 140 | 139 |
| 141 | 140 | 139 | 139 | 139 |
| 140 | 139 | 139 | 138 | 138 |

Gambar 2.13. Representasi citra aras keabuan yang diambil pada titik tertentu.

Sedangkan jumlah warna untuk citra RGB dihasilkan dengan mengalikan jumlah pada masing-masing komponennya, jumlah dari tiap komponennya, R=255 (8 bit), G=255 (8 bit) dan B=255 (8 bit). Sehingga seringkali disebut citra dengan intensitas 24 bit, jumlah warna RGB adalah sejumlah 28 \* 28 \* 28 = 16.777.216.

Gambar 2.14 memperjelas model representasi citra RGB, terlihat bahwa satu piksel mempunyai intensitas masing-masing komponen R, G dan B.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 135 | 134 | 137 | 139 | 141 |
| 138 | 138 | 138 | 140 | 141 |
| 137 | 136 | 139 | 141 | 142 |
| 133 | 132 | 138 | 140 | 142 |
| 134 | 133 | 136 | 138 | 140 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 137 | 136 | 139 | 141 | 143 |
| 140 | 140 | 140 | 142 | 143 |
| 139 | 138 | 141 | 143 | 144 |
| 135 | 134 | 140 | 142 | 144 |
| 136 | 135 | 138 | 140 | 142 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 136 | 135 | 138 | 140 | 142 |
| 139 | 139 | 139 | 141 | 142 |
| 138 | 137 | 140 | 142 | 143 |
| 134 | 133 | 139 | 141 | 143 |
| 135 | 134 | 137 | 139 | 141 |

Gambar 2.14 Representasi Citra RGB yang diambil pada titik tertentu. [19]

* + 1. Pengambangan

Pengambangan mempunyai pemikiran dasar untuk secara otomatis memilih satu atau beberapa pengambangan aras keabuan yang optimal untuk membagi objek kedalam basis latar pada persebaran aras keabuan. Metode pengambangan dapat dikategorikan kedaman dua kategori yaitu *global thresholding* dan *local thresholding*. Salah satu metode pengambangan yang termasuk kedalam *global thresholding* yaitu metode Otsu. Berdasarkan histogram citra aras keabuan, pengambangan Otsu membedakan antara klaster sebagai ukuran untuk memilih pengambangan yang optimal. [24]

* + 1. Morfologi

Operasi morfologi merupakan cara dalam pengolahan citra untuk mengolah struktur objek dalam citra sehingga didapatkan struktur objek yang diinginkan. Terdapat beberapa konsep dasar mengenai operasi morfologi yaitu dilasi, erosi, *opening*, dan *closing*. Penerapan fungsi dari morfologi sendiri berguna untuk mengekstrak komponen-komponen citra yang berguna diantaranya dalam representasi dan desktripsi bentuk, seperti ekstrak batas, dan komponen terhubung.

1. Dilasi

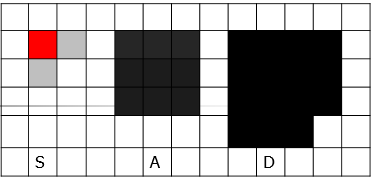
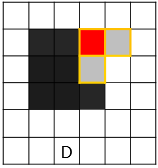
Dilasi yaitu proses penggabungan titik-titik latar (piksel bernilai 0) menjadi bagian dari objek (piksel bernilai 1), berdasarkan pada elemen pembangun (*structuring element*) *S* yang digunakan. A,B merupakan himpunan dari Z2 dengan komponen a = (a1, a2) dan b = (b1,b2). Secara metematis dilasi set A oleh set B didefinisikan pada persamaan 2.1.

(2.1)

Dari persamaan tersebut diketahui A merupakan citra biner yang dikenakan operasi sehingga terjadi perubahan bentuk objek, sedangkan B merupakan citra yang melakukan operasi tanpa mengalami perubahan bentuk. Hasil persamaan operasi dilasi A terhadap B disimbolkan dengan huruf C, persamaan ini dapat dilihat pada persamaan 2.2.

(2.2)

Proses dilasi terhadap setiap titik pada A bekerja dengan langkah yang pertama yaitu meletakkan titik poros S pada titik A, kemudian semua titik (x,y) yang terkena atau tertimpa oleh struktur S pada posisi tersebut diubah menjadi bernilai 1. Contoh dilasi dapat dilihat pada Gambar 2.15, dimana terdapat Gambar 2.15 (a) menunjukkan elemen pembangun dengan titik poros S, citra asli A, dan citra hasil dilasi terhadap citra A oleh citra D. Gambar 2.15 (b) menunjukkan proses dilasi yaitu penambahan titik-titik dengan cara titik poros S diletakkan pada masing-masing titik A.

(a) (b)

Gambar 2.15 Contoh Dilasi (a) Langkah dilasi (b) Proses pada saat posisi poros S ada di (2,0)

S memiliki titik (0,0) sebagai poros, (0,1) dari (+0,+1), dan (1,0) dari (+1,+0). Citra A memiliki titik (0,0),(1,0),(2,0),(0,1),(1,1),(2,1), (0,2),(1,2), dan (2,2). Titik poros S diletakkan pada titik A, dan titik A diberi angka 1 untuk semua titik (x,y) yang tertimpa oleh struktur S. Citra D adalah hasil dilasi pada citra A. Hasil proses dilasi ditunjukkan oleh Tabel 2.1.

Tabel 2.1 Hasil Proses Dilasi

|  |  |
| --- | --- |
| **Posisi poros** | **Sxy** |
| (0,0) | {(0,0),(1,0),(0,1)} |
| (1,0) | {(1,0),(2,0),(1,1)} |
| (2,0) | {(2,0),(3,0),(2,1)} |
| (0,1) | {(0,1),(1,1),(0,2)} |
| (1,1) | {(1,1),(2,1),(1,2)} |
| (2,1) | {(2,1),(3,1),(2,2)} |
| (0,2) | {(0,2),(1,2),(0,3)} |
| (1,2) | {(1,2),(2,2),(1,3)} |
| (2,2) | {(2,2),(3,2),(2,3)} |

1. Erosi

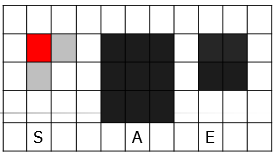
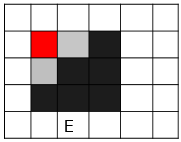
Erosi merupakan proses penghapusan titik-titik objek (piksel bernilai 1) menjadi bagian dari latar (piksel bernilai 0), berdasarkan pada *structuring element S* yang digunakan. Erosi berguna sebagai proses memperkecil ukuran objek. Erosi citra A oleh elemen penstruktur B secara matematis dituliskan pada persamaan 2.3.

(2.3)

Citra C sebagai hasil operasi erosi mengandung piksel-piksel dimana elemen penstruktur yang ditranslasikan bersesuaian dengan piksel-piksel pada citra A, dan hanya posisi translasi yang menempatkan titik asli dari elemen penstruktur pada piksel hitam dari citra A saja yang harus diperiksa.

Langkah proses erosi untuk setiap titik pada A yang dilakukan yaitu meletakkan titik poros S pada titik A tersebut, kemudian jika ada bagian dari S yang berada di luar A, maka titik poros dihapus atau dijadikan latar.

Gambar 2.16 menunjukkan contoh erosi. Pada Gambar 2.16 (a) terdapat elemen pembangun dengan titik poros S, citra asli A, dan citra D merupakan hasil erosi pada citra A. Pada Gambar 2.16 (b) terdapat proses erosi yaitu pengurangan titik-titik dengan cara titik poros S diletakkan pada masing-masing titik A.

(a) (b)

Gambar 2.16 Contoh Erosi (a) Proses erosi (b) Proses pada saat posisi poros S berada di (0,0)

S memiliki titik (0,0) sebagai poros, (0,1) dari (+0,+1), dan (1,0) dari (+1,+0). Citra A memiliki titik (0,0),(0,1),(0,2),(1,0),(1,1),(1,2), (2,0),(2,1), dan (2,2). Titik poros S diletakkan pada titik A, dan jika ada bagian dari S yang berada di luar A, maka titik poros dihapus atau dijadikan latar. Citra E adalah hasil erosi pada citra A. Hasil proses erosi ditunjukkan oleh Tabel 2.2. [**19**]

Tabel 2.2 Hasil Proses Erosi

|  |  |  |
| --- | --- | --- |
| **Posisi poros** | **Sxy** | **Kode** |
| (0,0) | {(0,0),(1,0),(0,1)} | 1 |
| (1,0) | {(1,0),(2,0),(1,1)} | 1 |
| (2,0) | {(2,0),(3,0),(2,1)} | 0 |
| (0,1) | {(0,1),(1,1),(0,2)} | 1 |
| (1,1) | {(1,1),(2,1),(1,2)} | 1 |
| (2,1) | {(2,1),(3,1),(2,2)} | 0 |
| (0,2) | {(0,2),(1,2),(0,3)} | 0 |
| (1,2) | {(1,2),(2,2),(1,3)} | 0 |
| (2,2) | {(2,2),(3,2),(2,3)} | 0 |

1. Opening

Operasi opening merupakan operasi erosi yang dilanjutkan dengan operasi dilasi secara berturut-turut dengan komponen struktur elemen yang sama. Operasi ini berfungsi untuk menghapus derau dengan bentuk yang sesuai dengan elemen penstruktur yang digunakan[25]. Contoh operasi opening ditunjukkan oleh Gambar 2.17. Efek yang dihasilkan adalah menghilangnya objek-objek kecil dan kurus, memecah objek pada titik-titik yang kurus, dan secara umum menghaluskan batas dari objek besar tanpa mengubah area objek secara signifikan. Rumus proses opening ditunjukkan pada persamaan 2.4.

(2.4)



Gambar 2.17 Contoh operasi O*pening* citra A oleh B (a) Citra A (b) Proses erosi citra A oleh B (c) Hasil erosi citra A (d) Proses dilasi citra A oleh B (e) Hasil operasi opening citra A oleh B

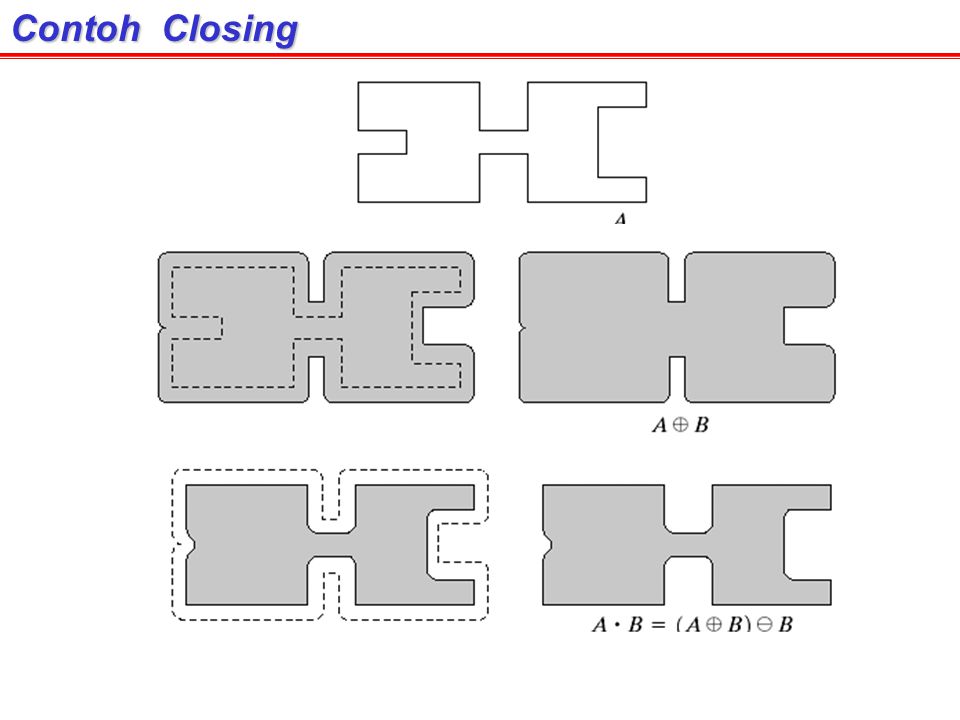
1. Closing

Closing pada umumnya menghubungkan bagian spasi, menghilangkan lubang kecil, dan mengisi celah dalam kontur. Closing A oleh B merupakan dilasi A oleh B, kemudian hasilnya dilanjutkan dengan erosi oleh B.

Contoh proses morfologi closing dapat dilihat pada Gambar 2.18, dimana dapat dilihat proses dimulai dari dilasi citra A oleh B, kemudian dilanjutkan dengan hasil dari proses dilasi dilakukan proses erosi oleh B.

(2.5)

Gambar 2.18 Proses morfologi *Closing* (a) Citra A (b) Proses dilasi citra A oleh B (c) Hasil dilasi citra A (d) Proses erosi citra A oleh B (e) Hasil operasi closing citra A oleh B [23]



(a)

(b)

(c)

(d)

(e)

* + 1. Pengenalan Objek

Pengolahan citra erat hubungannya dengan pengenalan objek yang dilakukan dengan tujuan untuk mengklasifikasikan objek yang tampak pada citra berdasarkan pengetahuan tertentu. Penelitian dilakukan dengan menggabungan pengenalan pola, konsep citra digital, matematika dan statistika. Tahapan-tahapan yang perlu dilakukan untuk melakukan pengenalan objek diperlukan *pre-precessing*, ekstraksi fitur, pengujuran kemiripan dan penentuan hasil pengenalan.

Tahap awal yang dilakukan adalah *pre-processing* citra, pada tahapan ini citra akan diambil untuk kemudian diolah untuk dilakukan pengubahan ukuran citra dalam ukuran piksel, kemudian diubah kedalam bentuk citra aras keabuanbaik data pelatihan maupun data uji coba. Tidak semua kasus selalu menggunakan tahapan *pre-processing* berupa pengubahan skala atau ukuran dan menjadikannya menjadi citra aras keabuan. *Pre-processing* tambahan berupa penapisan objek (*thinning*) akan diperlukan dalam kasus seperti pengenalan huruf, pengenalan tanda tangan, serta pengenalan tulisan tangan.

Pemunculan ciri diperlukan untuk melanjutkan tahap *pre processing*, tahap ini disebut sebagai ekstraksi fitur. Proses pengenalan objek akan bermasalah apabila dalam pengenalan objek menggunakan dimensi yang besar, apabila diabaikan akan menghambat waktu komputasi serta dapat menghilangkan ciri yang unik pada obyek yang dilatih maupun yang diuji[9].

* + 1. Ekstraksi Ciri Citra

Ciri citra adalah karakteristik yang membedakan atau atribut pada suatu citra. Ciri citra bersifat natural dalam artian bahwa fitur tersebut didefinisikan oleh visual tampilan gambar, sedangkan fitur buatan dihasilkan dari manipulasi gambar tertentu. Fitur alami meliputi pencahayaan area piksel dan piksel skala keabuan daerah tekstur. Histogram amplitudo gambar dan spektrum frekuensi spasial adalah contoh fitur buatan. Ciri citra sangat penting dalam isolasi daerah yang umum dimiliki pada suatu gambar (segmentasi citra) dan identifikasi selanjutnya atau pelabelan dari daerah tersebut (klasifikasi citra) [23].

* 1. OpenCV

OpenCV adalah pustaka sumber terbuka penglihatan komputer yang paling populer di dunia, terdapat lebih dari 500 algoritma yang dapat dioptimalisasikan untuk menganalisis gambar dan video. Di era gambar dan video digital, kebutuhan akan penglihatan komputer sangat sering, penggunakan algoritma gambar dan video yang disediakan oleh pustaka OpenCV dalam skala luas untuk bagian aplikasi penglihatan komputer. OpenCV dibawah lisensi BSD, yang berarti anda dapat membuat aplikasi bebas tanpa menyebutkan sumber kode anda. Tetapi terdapat beberapa code sumber didalam OpenCV yang dipatenkan [4] .

*Library* OpenCV ditulis dalam bahasa C dan C++ serta dapat dijalankan pada hampir semua sistem operasi seperi Linux, Windows, dan Mac OS X, dimana terdapat pengembangan aktif dalam Python, Ruby, Matlab, dan bahasa lain. OpenCV digunakan untuk perhitungan secara efisien dan sangat fokus dalam aplikasi waktu sebenarnya. Tujuan utama OpenCV adalah untuk menyediakan kebutuhan penglihatan komputer untuk membantu pengembangan aplikasi penglihatan yang cukup canggih secara cepat. Pustaka OpenCV juga menyertakan pemeriksaan produk pabrik, gambar medis, keamanan, tampilan antarmuka, kalibraasi kamera, penglihatan stereo, dan robotika. Karena penglihatan komputer dan mesin pembelajarannya saling berkesinambungan, OpenCV juga mendukung tujuan umum dari *Machine Learning Library* (MLL) [2] .

* 1. Bahasa Pemrograman Java

Bahasa Pemrograman Java merupakan salah satu dari beberapa bahasa pemrogramann yang dapat berjalan di berbagai sistem operasi, termasuk didalamnya sistem operasi pada telepon genggam, maka dari itu java sering disebut sebagai bahasa pemrograman yang multiplatform. Java merupakan bahasa Pemrograman Berorientasi Objek (PBO), serta memiliki pustaka yang lengkap. Bahasa pemrograman ini pertama kali dibuat oleh James Gosling saat masih bergabung dengan Sun Microsistems. Bahasa pemrograman ini merupakan pengembangan dari bahasa pemrograman C++ karena jika dilihat bahasa java banyak mengadopsi sintak bahasa C dan C++. Saat ini Java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi atau pun aplikasi berbasis web [17] .

Disebutkan pada situs web resmi dari Java menyebutkan hingga saat ini telah mengembangkan Java hingga versi 8.0. Platform atau teknologi Java dibagi menjadi tiga kelompok, yaitu: J2SE (Java 2 Standard Edition), J2EE (Java 2 Enterprise Edition), dan J2ME (Java 2 Micro Edition).

Tiga Komponen utama bahasa pemrograman Java, yakni:

* Bahasa pemrograman Class, Interface, dan JVM (*Java Virtual Machine*)
* Java Kombinasi antar Compiler (seperti bahasa Pascal, Cobol, Fortran, dan lain-lainnya)
* Interpreter (seperti bahasa LISP dan Basic).

Hasil kompilasi *Java berupa Virtual* (JVM) disebut *Java bytecode*. File *Java bytecode* yang sama dapat dijalankan pada semua jenis komputer yang memiliki *Java bytecode interpreter* [18] .

* 1. Android Studio

Android Studio merupakan IDE (*Integrated Development Environment*) yang sangat disarankan untuk mengembangkan perangkat lunak berbasis Android dan dapat digunakan secara gratis. Android Studio didasarkan pada perangkat lunak JetBrains IntelliJ IDEA. Versi stabil dari Android Studio dirilis pertama kali pada Desember 2004 dan telah mengganti Eclipse sebagai IDE utama untuk pengembangan Android. Di dalam Android Studio sendiri terdapat ADT(*Android Development Tool*) yang menyertainya.

Android Studio menyediakan fitur untuk membuat aplikasi yang diinginkan, seperti aplikasi untuk ponsel, phablet, tablet, jam dan perangkat pakai lainnya, Google Glass, aplikasi TV, dan juga Android Auto. Penggunaan Android Studio dapat dipasangkan pada sistem operasi Mac, Windows, atau Linux. Keperluan dalam menyimulasikan sebuah perangkat dan menampilkannya pada komputer, maka Android Studio menyediakan Android Emulator yang memungkinkan pengembang dapat melakukan pembuatan prototipe, mengembangkan, dan menguji aplikasi Android tanpa menggunakan perangkat keras ponsel pintar secara langsung. Emulator yang dapat digunakan berupa ponsel Android, tablet, Android Wear, dan perangkat Android TV. Tipe perangkat yang dikehendaki juga dapat ditentukan serta dapat menentukan tampilan emulator tersebut[5].

* 1. Android

Android adalah sebuah sistem operasi yang dikembangkan dari sistem operasi lain berbasus Linux untuk digunakan secara khusus pada perangkat bergerak seperti ponsel cerdas maupun tablet, untuk saat ini sudah dikembangkan pula perangkat pintar berbasis android seperti perangkat *wearable*. Sistem operasi android yang bersifat *opern source* menjadikan android sebagai salah satu sistem operasi yang digemari oleh banyak programer untuk beralih secara signifikan untuk mengembangkan aplikasi dan memodifikasi pada sistem operasi ini. Situs resmi dari android memberikan informasi bahwa terdapat lebih dari satu juta perangkat android diaktifkan dan diperkirakan dapat terus meningkat setiap harinya [7] .

Perusahaan pengembangan Android yaitu Android, Inc., serta sebagai pendirinya yaitu Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Android didukung secara finansial oleh perusahaan Google yang kemudian diambil alih pada tahun 2005. Pengembangan awal dari Android mempunyai tujuan untuk digunakan pada perangkat kamera digital dengan sebuah sistem operasi canggih didalamnya. Kebutuhan akan permintaan pasar akan perangkat kamera digital yang tidak cukup besar menjadikan pengembagan dari Android ini beralih ke pasar *smartphone* [15] .

Pada saat sekarang ini masyarakat paling banyak menggunakan perangkat dengan *platform* Android ini, dianatara penyebabnya yaitu karena Android sendiri memiliki kelebihan yang diantaranya:

1. Sistem Android dapat melakukan pekerjaan pada beberapa aplikasi sekaligus yang tidak terbatas yang disebut *Multi-Tasking*.
2. Terdapat banyak perangkat dengan Sistem Operasi Android yang disebabkan karena banyaknya vendor pendukung Sistem Operasi Android. Serta, ponsel pintar Android memiliki beragam harga untuk memenuhi kebutuhan pengguna.
3. Android memiliki dukungan Sistem Operasi yang cepat berkembang. Dibanding Sistem Operasi *mobile* lainnya, Android memiliki beberapa kelebihan, seperti dukungan format audio yang kaya, dukungan *multitouch,* hingga banyaknya pilihan aplikasi gratis dan *Open Source.*
4. Adanya internet membuat pengguna ponsel berbasis Android dapat menggunakan aplikasi yang memerlukan koneksi internet berbasis web untuk keperluan kerja dan keperluan notifikasi aplikasi sosial media[3] .
   1. Model Proses Pengembangan Perangkat Lunak

Pengembangan software pada mulanya dimulai dari titik awal. Analisa diperlukan untuk memenuhi kebutuhan spesifikasi untuk pembuatan aplikasi yang diinginkan dan dengan standar yang dibutuhkan. Masalah yang dihadapi banyak dari masalah pengembangan dari titik awal dan dalam proses pembuatannya hingga proses yang hampir selesai. Tetapi kebutuhan aplikasi dapat berubah-ubah sewaktu-waktu yang menjadikan perubahan tujuan pengembangan pembuatan *software*. Dalam pengembangan software jauh berbeda dengan praktiknya dikarenakan pemrogram juga merupakan manusia yang sering terdapat kesalahan dalam prosesnya.

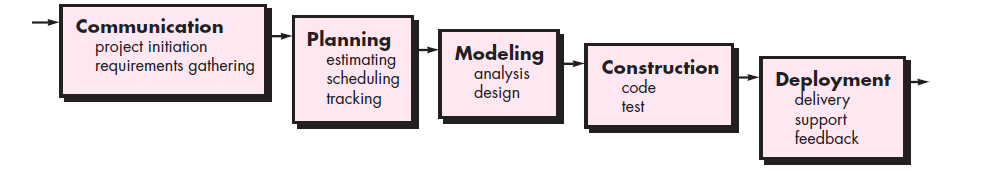
Model perangkat lunak sebenarnya menyerupai model *evolution-tree* dan juga *waterfall.* Pertimbangan dari perkembangan versi-versi ini pada dasarnya berulang. Artinya dari hasil revisi versi pertama menghasilkan versi kedua dan seterusnya. Iterasi merupakan aspek intrinsik rekayasa perangkat lunak, dan model siklus hidup berulang telah digunakan selama lebih dari 30 tahun (Larman dan Basili, 2003).

Dalam menyelesaikan masalah pengembangan perangkat lunak tim harus menyiapkan proses, metode, alat bantu. Model proses rekayasa perangkat lunak dipilih berdasarkan sifat aplikasi dan proyeknya. Terdapat beberapa metode pengembangan perangkat lunak yaitu : *Model siklus code-and-fix, Model siklus waterfall, model rapid-prototyping, Model open-source, Agile Processes, Synchronize-and-stabilize life - cycle model, Spiral life-cycle model*.

Model *waterfall* pertama dikemukakan oleh Royce (1970). Poin penting pada model *waterfall* adalah tidak ada fase yang selesai samapai dokumentasi pada fase itu selesai dan produk pada fase tersebut sudah diterima oleh tim penjamin kualitas software. Pengetesan harus dilakukan terus menerus selama pengembangan *software*. kadang, dalam proses perawatan perlu dipastikan bahwa versi yang sudah dimodifikasi masih memiliki fitur dari versi sebelumnya, namun juga memuaskan kebutuhan baru yang diminta oleh klien. Pengembangan perangkat lunak dengan model *iterative-and-incremental* sama-sama dikembangkan secara bertambah, dengan mengulang pada fase kebutuhan, analisa, desain dan implementasi hingga dirasa perulangan tidak diperlukan lagi.

Model *waterfall* memiliki banyak kelebihan termasuk pendekatan yang harus terartur, tetapi model ini terpaku pada dokumentasi juga bisa menjadi kelemahan yang disebabkan perbedaan pemahaman antara pengembang dan klien. maka perlu diperjelas bahwa hanya arsitek yang dapat membantu klien memahami apa yang akan dibuat. Masalahnya terkadang penjelasan arsitek tidak menjelaskan bagaiman produk dapat bekerja. Solusinya adalah model rapid-prototyping[16] .

Model waterfall disebut juga sebagai *classic life cycle,* yang merupakan paradigma model proses yang paling lama digunakan. Karakteristik yang dimiliki *Waterfall* antara lainmerupakan model proses sistematis dan sekuansial, tahapan yang diselesaikan harus dari awal terlebih dahulu sebelum proses yang lain dijalankan. Tahap dalam model *Waterfall* dapat dilihat pada Gambar 2.19[20].



Gambar 2.19 Model Waterfall menurut Roger S. Pressman.

Aktivitas yang terdapat pada setiap tahap model *waterfall* adalah :

1. *Communication*

Tahap ini merupakan tahap awal untuk pengenalan proyek. Kebutuhan secara lengkap dikumpulkan lalu menganalisis dan mendefinisikan kebutuhan yang harus terpenuhi di dalam program yang akan dibangun. Pengerjaan fase harus lengkap untuk bisa menghasilkan desain yang memenuhi.

1. *Planning*

Melakukan perancangan perangkat lunak. proses ini terbagi menjadi 3 (tiga) sub proses yaitu : *estimating*, *scheduling*, dan *tracking*.

1. *Modelling*

Memodelkan kebutuhan yang telah diperoleh sebelumnya ke dalam pemodelan analisis dan pemodelan desain yang nantinya akan digunakan sebagai acuan pengimplementasian ke dalam kode-kode bahasa pemrogaraman.

1. *Construction*

Tahapan untuk proses *coding* dan *testing,* maka akan dilakukan implementasi dari desain yang telah dibuat ke dalam kode program lalu akan dilakukan *testing*. Proses ini akan dilakukan berulang-ulang hingga *bu*g masalah atau kecacatan teridentifikasi dan dapat diperbaiki.

1. *Deployment*

Program dioperasikan dilingkungan yang sesuai dengan program tersebut dan melakukan pemeliharaan, seperti melakukan penyesuaian atau perubahan dikarena adanya adaptasi dengan situasi sebenarnya.