

BAB II

TINJAUAN PUSTAKA

2.1 Image Processing

Image processing atau pengolahan citra merupakan suatu metode atau teknik yang dapat digunakan untuk memproses citra atau gambar dengan jalan memanipulasinya menjadi suatu data gambar yang diisikan untuk mendapatkan suatu informasi tertentu mengenai obyek yang sedang diamati .dan hal –hal yang perlu diperhatikan sebagai referensi pembuatannya disini diantaranya adalah:

2.1.1 Pengolahan Citra

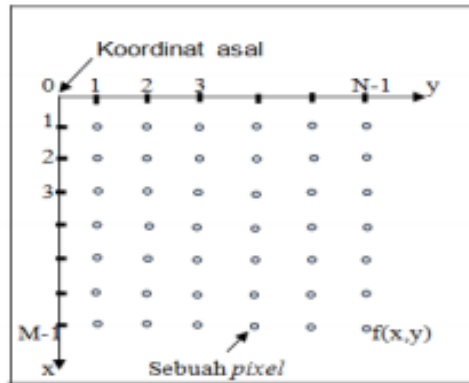
Representasi dua dimentasi untuk bentuk-bentuk fisik nyata tiga dimensi. Citra dalam perwujudan dapat bermacam-macam,mulai dari gambar perwujudan nya dapat bermacam –macam,mulai dari gambar putih pada sebuah foto (yang tidak bergerak)sampai pada gambar warna yang bergerak pada televisi .proses transformasi dari bentuk tiga dimensi ke bentuk dua dimensi untuk menghasilkan citra akan dipengaruhi oleh bermacam-macam factor yang mengakibatkan citra penampilan citra suatu benda tidak sama persis dengan bentuk fisik nyatanya. Teknik dan proses untuk mengurangi atau menghilangkan efek degradasi pada citra meliputi teknik perbaikan atau peningkatan citra (*image enchancement*),restorasi citra (image restoration) dan tranformasi special (*special transformation*),subyek lain dari pengolahan citra digital diantaranya adalah pengkodean citra ,segmentasi citra(image segmentation),representasi edan diskripsi citra (image representation and diskription)^[11]

2.1.2 Model Citra

Merupakan matrik dua dimensi dari fungsi intensitas cahaya, karena itu referensi citra menggunakan dua variabel yang menunjuk posisi pada bidang dengan sebuah fungsi intensitas cahaya yang dapat dituliskan sebagai berikut $f(x,y)$.karena cahaya merupakan salah satu bentuk energi yang dalam persamaan 2.1:

$$0 < f(x) < \sim \quad (2.1)$$

Konversi system koordinat citra diskrit ditunjukkan pada gambar 1:



Gambar 2.1 Koordinat Citra Digital

Citra yang dilihat sehari-hari merupakan cahaya yang direfleksikan sebuah obyek. fungsi $f(x,y)$ dapat dilihat sebagai fungsi dengan dua unsur, pertama merupakan besarnya sumber cahaya yang melengkapi pandangan kita terhadap obyek (illumination), kedua merupakan besaran cahaya yang difleksikan oleh obyek dalam pandangan kita (reflectance component). keduanya dituliskan fungsi yang berturut-turut $i(x,y)$ dan $r(x,y)$. merupakan kombinasi perkalian untuk membentuk fungsi $f(x,y)$ yang dapat ditulis dengan persamaan 2.2 sebagai berikut: [\[11\]](#)

$$f(x,y) = i(x,y)r(x,y) \quad (2.2)$$

$$0 < i(x,y) < \sim$$

$$0 < r(x,y) < 1$$

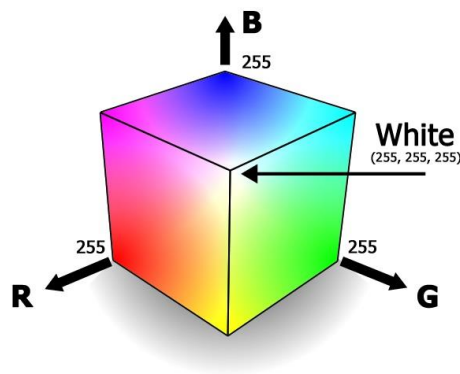
Persamaan diatas menandakan bahwa nilai kerefleksian dibatasi oleh nilai 0 (total absorbtion) dan nilai satu (total reflectance) fungsi $i(x,y)$ yang sudah didiskritkan baik koordinat special maupun tingkat kecerahannya. kata continue disini dijelaskan bahwa indek x dan y bernilai bulat .kita dapat menganggap citra digital (berikutnya akan disingkat dengan citra) sebagai matrik dengan ukuran $M \times N$ yang baris dan kolomnya menunjukkan titik-titiknya yang diperlihatkan pada persamaan berikut:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,N-1) \\ f(M-1,0) & f(M-1,1) & f(M-1,N-1) \end{bmatrix}$$

Citra yang tidak berwarna atau hitam putih dikenal juga sebagai citra dengan derajat keabuan (citra grey level /greyscale). Derajat keabuan yang dimiliki ini bias beragam mulai dari dua derajat keabuan yaitu (0 dan 1) yang dikenal juga sebagai citra monokrome. 16 derajat keabuan dan 256 derajat keabuan. Dalam sebuah citra monochrome, sebuah pixel diwakili oleh 1 bit data yang berisikan data tentang derajat keabuan yang dimiliki oleh pixel tersebut. data akan berisi 1 apabila pixel berwarna putih dan akan berisi 0 bila pixel berwarna hitam.

2.1.2.1 Citra RGB

Pada pengolahan warna gambar, ada bermacam-macam model warna. Model *RGB* (*red green blue*) merupakan model yang banyak digunakan, salah satunya adalah monitor. Pada Gambar 2.2 menunjukkan model warna *RGB*.^[3]



Gambar 2.2 Warna RGB

Model warna *RGB* merupakan model warna yang paling umum digunakan pada pengolahan citra digital. Citra *RGB* disusun dari tiga buah kanal warna primer yaitu Merah (*Red*), Hijau (*Green*) dan Biru (*Blue*). Setiap kanal penyusun warna *RGB* berisikan intensitas warnanya dalam skala 8 bit atau range nilai antara 0 hingga 255.

Pada tiap piksel elemen sebuah citra, berisikan perpaduan ketiga warna tersebut. Seperti pada **Error! Reference source not found.** warna putih, perpaduan ketiganya adalah pada nilai maksimal (255,255,255) sedangkan untuk warna hitam, perpaduan warna ketiganya berada pada nilai minimal (0,0,0). Dari ketiga perpaduan warna ini didapatkan 16 juta warna variasi ^[4]

2.1.2.2 Citra *grayscale*

Model warna keabuan (*grayscale*) merupakan warna dengan skala 8 bit atau range nilai antara 0 hingga 255. Warna keabuan dapat dikatakan sebagai warna dengan satu kanal warna, yaitu kanal warna 8 bit. Untuk mendapatkan citra dengan skala keabuan ini, dapat dilakukan pengubahan dari citra *RGB* ke *grayscale*. Hal yang paling umum dilakukan adalah dengan membagi tiga komposisi warna *RGB* tersebut. Masing-masing warna memiliki bobot 0.33, atau beberapa metode mengusulkan komposisi bobot tiap warnanya. Bergantung pada kebutuhan dan ketetapan rumusan standarnya. ^[4]

2.1.2.3 Citra Biner

Citra biner adalah citra yang pikselnya memiliki kedalaman bit sebesar 1 bit sehingga hanya memiliki dua nilai intensitas warna yaitu 0 (hitam) dan 1 (putih). Citra *grayscale* dapat dikonversi menjadi citra biner melalui proses *thresholding*. Dalam proses *thresholding*, dibutuhkan suatu nilai *threshold* sebagai nilai pembatas konversi. Nilai intensitas piksel yang lebih besar atau sama dengan nilai *threshold* akan dikonversi menjadi 1. Sedangkan nilai intensitas piksel yang kurang dari nilai *threshold* akan dikonversi menjadi 0. Misalnya nilai *threshold* yang digunakan adalah 128, maka piksel yang mempunyai intensitas kurang dari 128 akan diubah menjadi 0 (hitam) dan yang lebih dari atau sama dengan 128 akan diubah menjadi 1 (putih).

2.2 Haar Cascade Classifier

Haar-Like Feature

Haar like Feature merupakan metode yang lazim digunakan dalam pendeteksian obyek. Nama Haar sendiri mengacu pada Haar Wavelet, sebuah fungsi matematika yang berbentuk kotak dan memiliki prinsip seperti pada fungsi Fourier. Haar-like features merupakan rectangular features (fungsi persegi), yang memberikan indikasi secara spesifik pada sebuah gambar atau image.

Prinsip Haar-like features adalah mengenali obyek berdasarkan nilai sederhana dari fitur tetapi bukan merupakan nilai piksel dari image obyek tersebut. Metode ini memiliki kelebihan yaitu komputasinya sangat cepat, karena hanya bergantung pada jumlah piksel dalam persegi bukan setiap nilai piksel dari sebuah image. Deteksi obyek kendaraan pada penelitian ini merupakan pemodifikasian sistem Haar-like features dari deteksi wajah yang pertama kali dilakukan oleh Viola dan Jones kemudian dikembangkan oleh Lienhart^[5] Metode yang diusulkan Viola dan Jones menggabungkan empat kunci utama untuk mendeteksi sebuah obyek :

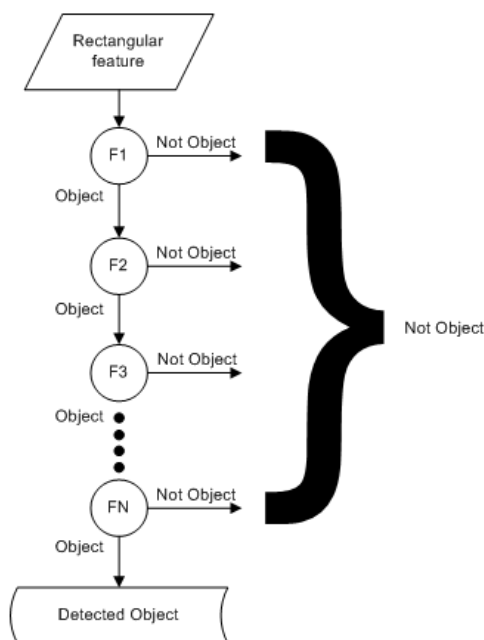
1. Fitur persegi sederhana, disebut fitur Haar
2. Integral image untuk pendeteksian fitur dengan cepat
3. Metoda AdaBoost machine-learning
4. Cascade classifier untuk mengkombinasikan banyak fitur

Haar like feature memproses citra dalam sebuah kotak persegi dengan ukuran tertentu misalnya 24 x 24 pixel seperti ditunjukkan pada gambar 1 sebagai berikut:



Gambar 2.3 *Haar like feature*

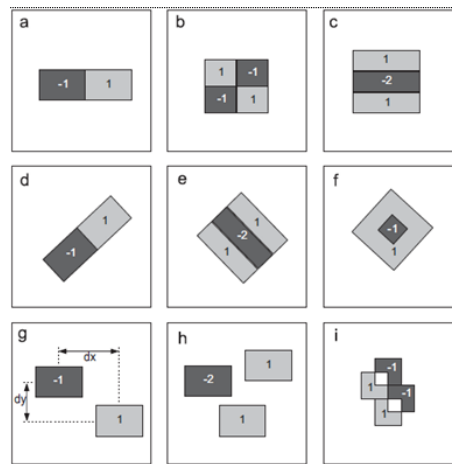
Di dalam kotak inilah proses filtering obyek dilakukan untuk diketahui apakah ada atau tidak obyek yang akan dideteksi. Proses filterisasi ini dilakukan secara bertingkat yang menyebabkan metode ini nantinya disebut sebagai Haar Cascade Classifier seperti ditunjukkan pada skema filter di gambar :



Gambar 2.4 proses filtering Haar Cascade Classifier

Hasil deteksi *Haar-like Feature* kurang akurat jika hanya menggunakan satu fungsi saja. Semakin tinggi tingkatan filter pendeteksian maka semakin tepat pula sebuah obyek dideteksi akan tetapi akan semakin lama proses pendeteksiannya. Pemrosesan *Haar-like feature* yang banyak tersebut diatur dalam *classifier cascade*.

Haar-Wavelet (Wave = Gelombang) merupakan gelombang persegi (interval gelap dan interval terang) yang kemudian dibandingkan nilai rata-rata *pixel* keduanya. Apabila perbandingan nilai rata-rata intensitas tersebut berada di atas *threshold* (ambang batas), maka dikatakan memenuhi syarat fitur Haar. Untuk gambar bergerak seperti video, proses ini dilakukan secara diskrit dengan mencuplik video pada frame rate tertentu. Macam-macam variasi Haar-like *feature* ditunjukkan pada gambar sebagai berikut



Gambar 2.5 Berbagai variasi persegi Haar dengan Bobot Standar Persegi Panjang

Keterangan gambar ^[6]:

a, b : Fitur Haar yang diusulkan Papageogiou dkk.

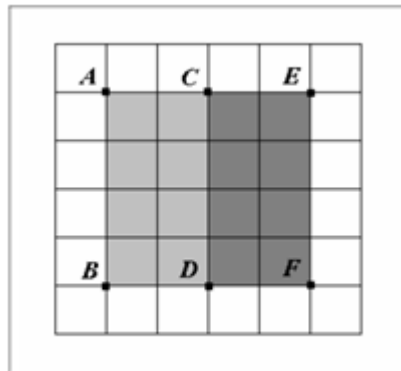
c : Fitur Haar yang diusulkan Viola dan James

d, e, f : Variasi fitur Haar yang diusulkan Leinhardt

g, h : Penguraian Liental terhadap Fitur Haar-like

i : Fitur Haar-like Viola dan James untuk menangkap struktur diagonal dalam penampilan obyek

Perhitungan nilai fitur Gambar 3 di atas ditunjukkan pada Gambar 4 dan formula (1) sebagai berikut:



Gambar 2.6. Fitur Persegi Haar-like

$$\text{NilaiFitur(ABFE)} = \text{JumlahNilaiPixel(ABDC)} - \text{JumlahNilaiPixel(CDFE)} \dots\dots (1)$$

Apabila NilaiFitur(ABFE) memiliki nilai di atas *threshold* maka dikatakan memenuhi syarat. Seperti dijelaskan pada gambar 2, apabila sebuah fitur dikatakan tidak memenuhi syarat, maka area ABFE tidak terdapat obyek yang dideteksi dan area persegi berpindah lokasi akan tetapi jika persegi ABFE memenuhi fitur, maka aturan fitur berikutnya dilakukan. Jika semua syarat fitur dipenuhi dikatakan pada persegi ABFE dikatakan terdapat obyek.

Pertama, ditentukan terlebih dahulu area yang akan dideteksi apakah terdapat obyek atau tidak. Proses berikutnya adalah melakukan pendeteksian obyek menggunakan Haar Cascade Clasifier dengan langkah-langkah yang akan dijelaskan sebagai berikut.

Citra integral adalah sebuah citra yang nilai tiap pixel-nya merupakan penjumlahan nilai pixel atas dan kirinya. Sebagai contoh pada gambar 5 (c) sebuah daerah persegi yang akan di-scan menggunakan persegi gelap terang memiliki nilai sebagai berikut:

2	3	1	3	6	5
3	1	2	5	4	4
1	2	3	4	5	4
4	4	5	6	7	3
5	4	3	6	7	4
1	2	3	4	5	6

Gambar 2.7 Citra Masukan

Pada citra masukan yang diberi persegi pada gambar 5 (c) terlihat pada gambar 9 sebagai berikut:

2	3	1	3	6	5
3	1	2	5	4	4
1	2	3	4	5	4
4	4	5	6	7	3
5	4	3	6	7	4
1	2	3	4	5	6

Gambar 2.8 Persegi Haar Like Pada Citra masukan

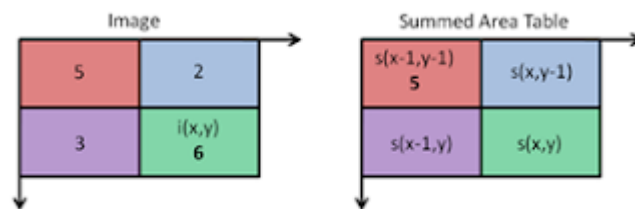
Proses perhitungan nilai gelap dan nilai terang di formula 1 menghasilkan nilai Fitur Haar sama dengan 22. Untuk menghitung Fitur nilai Haar menggunakan Summed Area Table atau yang dikenal sebagai Integral image, pertama dibentuk matriks nilai integral image.

Proses integral image matriks citra ditentukan menggunakan formula (3) sebagai berikut:

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

..... (3)

Dimana $x-1$ dan $y-1$ adalah pixel tetangga dari pixel yang akan dihitung seperti terlihat pada gambar 10 sebagai berikut:



Gambar 2.9 Pixel Tetangga pada Proses Integral image

Menggunakan formula 3 di atas, *integral image* dari citra masukan di Gambar 6 terlihat pada gambar 11 sebagai berikut:

2	5	6	9	15	20
5	9	12	20	30	39
6	12	18	30	45	58
10	20	31	49	71	87
15	29	43	67	96	116
16	32	49	77	111	137

Gambar 2.10 Matriks Integral Image dari Citra Masukan

Nilai fitur Haar dari area matriks di gambar 11 dihitung menggunakan formula (4) sebagai berikut:

$$i(x',y')=s(A)+s(D)+s(B)-s(C)$$

..... (4)

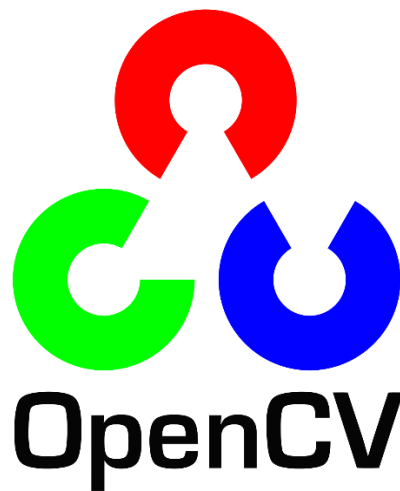
Sehingga:

$$\text{fitur nilai Haar} = (32+2-16-5)-(49+5-32-6)+(77+6-49-9) = 22$$

Hasil ini sesuai dengan perhitungan secara manual menggunakan formula 1. Nilai 22 tersebut kemudian dibandingkan dengan *threshold* yang sudah ditentukan sebagai pendeteksi obyek. Apabila nilai fitur Haar lebih tinggi daripada *threshold*, maka dapat dikatakan pada area tersebut memenuhi filter Haar. Sesuai *flowchart* pada gambar 2, proses ini akan dilanjutkan untuk menguji kembali area tersebut dengan filter Haar yang lain dan apabila seluruh filter Haar terpenuhi maka dikatakan pada area tersebut terdapat obyek yang diamati.

2.3 Open Source Computer Vision (OpenCV)

OpenCV merupakan sebuah *library* untuk fungsi-fungsi *computer vision* dan OpenCV yang mendukung pembuatan program ini.^[7] Program OpenCV merupakan program bebas dan berada dalam lingkungan sumber terbuka dari lisensi *BSD*. Pustaka yang digunakan merupakan pustaka lintas platform. Program tersebut dikontribusikan sebagian besar sebagai pengolahan citra secara *real-time*. Jika pustaka tersebut terdapat kesamaan pustaka *integrated performance primitives* dari intel dalam sistem komputer, maka program tersebut akan digunakan rutin untuk mempercepat proses kinerja program secara otomatis. Pada **Error! Reference source not found.**⁵ menunjukkan logo dari library OpenCV.^[12]



Gambar 2.11 Logo OpenCV

2.4 Python

Python adalah bahasa pemrograman model skrip (*scripting language*) yang berorientasi obyek. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai *platform* sistem operasi. Python merupakan bahasa pemrograman yang *freeware* atau perangkat bebas dalam arti sebenarnya, tidak ada batasan dalam penyalinannya atau mendistribusikannya. Lengkap dengan *source codenya*, *debugger* dan *profiler*, antarmuka yang terkandung di dalamnya untuk pelayanan antarmuka, fungsi sistem, *GUI* (antarmuka pengguna grafis), dan basis datanya^[8]. Berikut adalah lambang atau logo Python pada Gambar 2.6.



Gambar 2.12 Logo Python

Python adalah salah satu bahasa pemrograman tingkat tinggi yang bersifat interpreter, interaktif, *object-oriented* dan dapat beroperasi di hampir semua *platform*, seperti keluarga *UNIX*, *mac*, *windows*, dan lainnya. Sebagai bahasa

tingkat tinggi, Python termasuk salah satu bahasa pemrograman yang mudah untuk dipelajari karena sintaks yang jelas dan elegan, dikombinasikan dengan penggunaan modul-modul siap pakai dan struktur data tingkat tinggi yang efisien.^[9]

2.5 Webcam

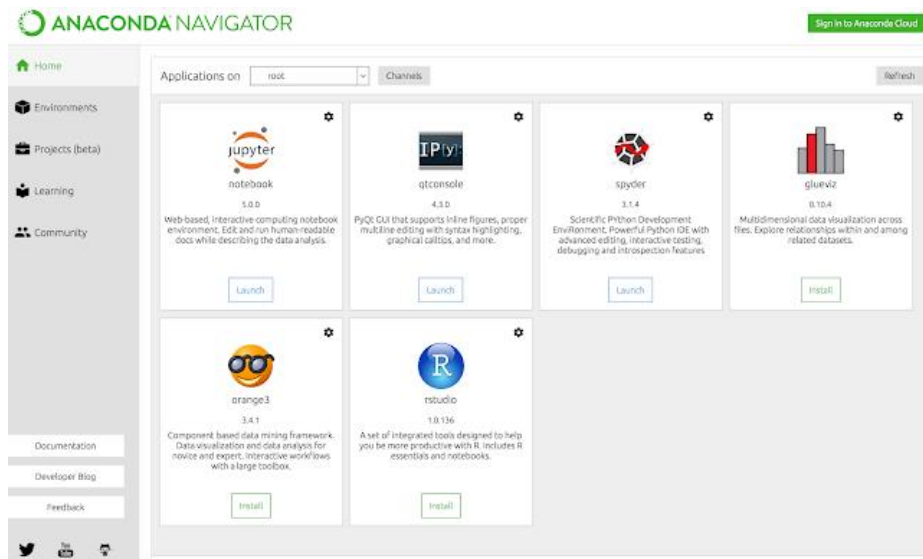
Webcam adalah sebuah *PC* kamera yang dikoneksikan ke internet. *Webcam* secara luas mulai digunakan sekitar satu dekade yang lalu oleh banyak aplikasi dan dunia web.^[13] *Webcam* atau kamera web, pada dasarnya adalah sebuah kamera digital yang terhubung ke komputer, yang berfungsi untuk mengambil citra yang akan diolah oleh komputer. Pada awalnya *webcam* digunakan sebagai alat komunikasi yang menampilkan rentetan citra dan dapat diakses melalui *world wide web*. Namun, seiring perkembangannya *webcam* digunakan juga untuk keperluan lainnya.^[8] Pada Gambar 2.7 merupakan salah satu contoh *webcam*^[10].



Gambar 2.13 Webcam Logitech C270

2.6 Anaconda

Anaconda merupakan sebuah distribusi bahasa pemrograman Python dan R untuk keperluan komputasi ilmiah (scientific computing) seperti data science, machine learning, data processing skala-luas, analisis prediksi, dan lain sebagainya.



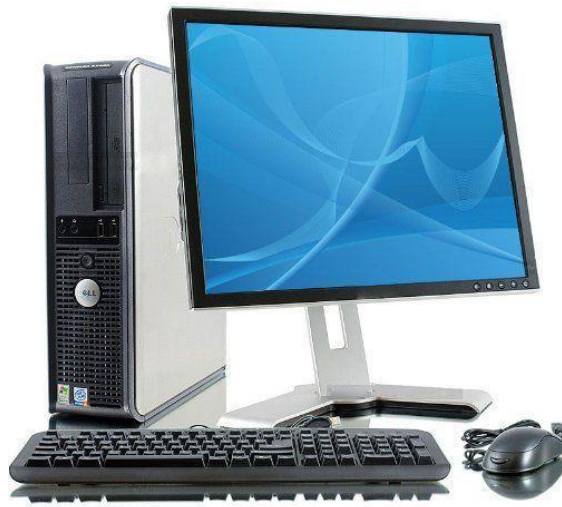
Gambar 2.14 Tampilan anaconda navigator

Anaconda diciptakan agar mempermudah pengguna manajemen paket python. Dengan menggunakan Anaconda, maka versi dari paket yang ada, di manajemen oleh package management system conda.

Anaconda telah digunakan oleh lebih dari 12jt pengguna, dan merupakan yang terpopuler sebagai distribusi Python dan R, dan memiliki lebih dari 1400 paket data science yang cocok untuk Windows, Linux, dan MacOS.

2.7 Personal Computer (PC)

Hamacher, Vranesic, & Zaky^[14] mengatakan bahwa komputer didefinisikan sebagai sebuah mesin penghitung elektronik yang cepat dapat menerima informasi input digital, memprosesnya sesuai dengan suatu program yang tersimpan di memorinya (stored program) dan menghasilkan output informasi.

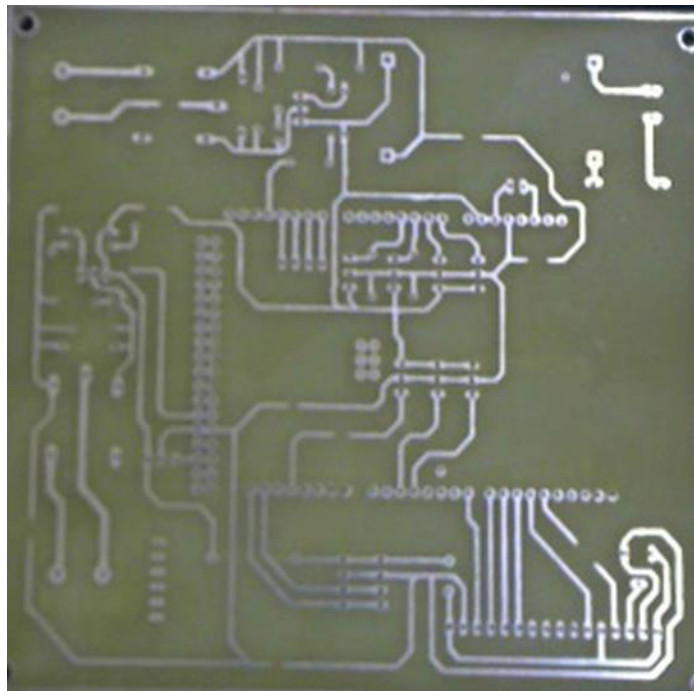


Gambar 2.15 Komputer desktop

Personal Computer berfungsi sebagai main processing platform maka dari itu untuk menunjang kehandalan system, spesifikasi yg digunakan adalah Intel i58250U dengan RAM 12GB

2.8 PCB

PCB yang dapat dideteksi adalah *Single Layered PCB* dengan ukuran 13,5x12cm



Gambar 2.16 Gambar PC