BAB II

DASAR TEORI

2.1 Definisi Tentang Batuan

2.1.1 Batuan

Batuan adalah suatu masa mineral yang dapat terdiri atas satu jenis mineral atau lebih. Mineral adalah suatu bahan atau unsur kimia, gabungan beberapa unsur kimia sebagai hasil proses alam, bersifat homogen dan mempunyai susunan atau rumus kimia tertentu. Batuan akan mengalami proses penghancuran dan kemudian akan terpindahkan atau tergerak oleh berbagai macam proses alam seperti aliran alir, hembusan angin, gelombang pantai, maupun *gletser*. Media pengangkut tersebut dikenal sebagai alat pengikis, yang dapat membawa *fragmen* atau bahan yang larut ke tempat-tempat tertentu berupa sedimen dan berupaya untuk meratakan permukaan bumi. Kemudian terjadi perubahan dari batuan lepas menjadi batuan yang keras melalui pembebanan dan perekatan oleh senyawa mineral dalam larutan menjadi batuan sedimen. Batuan-batuan tersebut akan menyesuaikan dengan lingkungan yang baru sehingga terbentuklah batuan malihan atau *metamorfis*.



Gambar 2.1 Macam-macam batuan mineral

Berdasarkan jenis batuan tersebut maka secara garis besar batuan di alam ini dibagi menjadi tiga bagian, yaitu :

1. Batuan beku
2. Batuan sedimen
3. Batuan metamorfis

Dari ketiga batuan diatas, penulis lebih tertarik membahas jenis batuan nomor dua yaitu batuan sedimen, karena paling banyak ditemui di daerah padalarang sebagai laboratorium dunia yang banyak diteliti oleh peneliti-peneliti dari luar negeri.

2.1.2 Batuan Sedimen

Batuan sedimen adalah batuan yang terbentuk dari hasil proses pelapukan, erosi, pengangkutan dan pengendapan dari batuan yang sudah ada, baik batuan beku, sedimen maupun batuan *metamorf*. Batuan sedimen yang terbentuk melalui proses-proses ini dinamakan batuan sedimen klastik. Selain batuan sedimen klastik ada batuan sedimen non klastik yaitu batuan sedimen yang terbentuk dari unsur-unsur *organisme* ataupun kimiawi, misalnya garam dapur terjadi karena proses penguapan, batu gamping koral terbentuk karena *organisme moluska*.

Batuan sedimen klastik yang sangat terkenal antara lain batu pasir, batu lempung. Batuan sedimen non klastk yang sangat terkenal antara lain batu gamping, garam dapur dan batu *gips.*

2.1.3 Proses Sedimen

Batuan yang berasal dari hasil rombakan berbagai jenis batuan adalah batuan sedimen. Batuan sedimen ini terbentuk dengan proses pertama tentunya adalah pecahnya atau terabrasinya batuan sumber yang kemudian hasil pecahannya tertransportasi dan mengendap di suatu area tertentu. Proses-proses tersebut telah lazim disebut sebagai proses-proses sedimentasi. Proses sedimentasi pada batuan sedimen klastik terdiri dari dua proses, yakni proses sedimentasi secara mekanik dan proses sedimentasi secara kimiawi.

Proses Sedimentasi Mekanik

Proses sedimentasi secara mekanik merupakan proses dimana butir-butir sedimen tertransportasi hingga diendapkan di suatu tempat. Proses ini dipengaruhi oleh banyak hal dari luar. Transportasi butir-butir sedimen dapat dipengaruhi oleh air, gravitasi, angin, dan es. Dalam cairan, terdapat dua macam aliran, yakni laminar (yang tidak menghasilkan transportasi butir-butir sedimen) dan *turbulent* (yang menghasilkan transportasi dan pengendapan butir-butir sedimen). Arus turbulen ini membuat partikel atau butiran-butiran sedimen mengendap secara suspensi, sehingga butiran-butiran yang diendapkan merupakan butiran sedimen berbutir halus (pasir hingga lempung). Proses sedimentasi yang dipengaruhi oleh gravitasi dibagi menjadi empat, yakni yang dipengaruhi oleh arus *turbidit*, *grain flows*, aliran sedimen cair, dan *debris flows*.

Proses Sedimentasi Kimiawi

Proses sedimentasi secara kimiawi terjadi saat pori-pori yang berisi *fluida* menembus atau mengisi pori-pori batuan. Hal ini juga berhubungan dengan reaksi mineral pada batuan tersebut terhadap cairan yang masuk tersebut. Berikut ini merupakan beberapa proses kimiawi dari *diagenesis* batuan sedimen klastik:

a) Pelarutan *(Dissolution)*, mineral melarut dan membentuk porositas sekunder.

b) Sementasi *(Cementation)*, pengendpan mineral yang merupakan semen dari batuan, semen tersebut diendapkan pada saat proses primer maupun sekunder.

c) *Authigenesis*, munculnya mineral baru yang tumbuh pada pori-pori batuan

d) *Recrystallization*, perubahan struktur kristal, namun kompsisi mineralnya tetap sama. Mineral yang biasa terkristalisasi adalah kalsit.

e) *Replacement*, melarutnya satu mineral yang kemudian terdapat mineral lain yang terbentuk dan menggantikan mineral tersebut

f) Kompaksi *(Compaction)*

g) Bioturbasi *(Bioturbation)*, proses sedimentasi oleh hewan (makhluk hidup).

2.1.4 Macam-Macam Batuan Sedimen

1. Batuan Sedimen Klastik

Batuan sedimen klastik merupakan batuan sedimen yang terbentuk dari pengendapan kembali *detritus* atau pecahan batuan asal. Batuan asal dapat berupa batuan beku, *metamorf* dan sedimen itu sendiri. Batuan sedimen diendapkan dengan proses mekanis, terbagi dalam dua golongan besar dan pembagian ini berdasarkan ukuran besar butirnya. Cara terbentuknya batuan tersebut berdasarkan proses pengendapan baik yang terbentuk dilingkungan darat maupun dilingkungan laut. Batuan yang ukurannya besar seperti breaksi dapat terjadi pengendapan langsung dari ledakan gunung api dan diendapkan disekitar gunung tersebut dan dapat juga diendapkan dilingkungan sungai dan batuan batu pasir bisa terjadi dilingkungan laut, sungai dan danau. Semua batuan diatas tersebut termasuk ke dalam golongan *detritus* kasar. Sementara itu, golongan *detritus* halus terdiri dari batuan lanau, serpih dan batua lempung dan napal. Batuan yang termasuk golongan ini pada umumnya diendapkan di lingkungan laut dari laut dangkal sampai laut dalam.

Proses diagenesa antara lain :

a) Kompaksi sedimen

Yaitu tepatnya butir sedimen satu terhadap yang lain akibat tekanan dari berat beban di atasnya. Disini *volume* sedimen berkurang dan hubungan antar butir yang satu dengan yang lain menjadi rapat.

b) Sementasi

Sementasi yaitu turunnya material-material di ruang antar butir sedimen dan secara kimiawi mengikat butir-butir sedimen dengan yang lain. Sementasi makin efektif bila derajat kelurusan larutan pada ruang butir makin besar.

c) Rekristalisasi

Rekristalisasi yaitu pengkristalan kembali suatu mineral dari suatu larutan kimia yang berasal dari pelarutan material sedimen selama diagenesa atu sebelumnya. Rekristalisasi sangat umum terjadi pada pembentukan batuan karbonat.

d) Autigenesis

Autigenesis yaitu terbentuknya mineral baru di lingkungan diagenesa, sehingga adanya mineral tersebut merupakan partikel baru dlam suatu sedimen. Mineral autigenik ini yang umum diketahui sebagai berikut : karbonat, *silica, klorita, gypsum* dan lain-lain.

e) Metasomatisme

Metasomatisme yaitu pergantian material sedimen oleh berbagai mineral *autigenik*, tanpa pengurangan *volume* asal.

2. Batuan Sedimen Non-Klastik

Batuan sedimen Non-Klastik merupakan batuan sedimen yang terbentuk sebagai hasil penguapan suatu larutan, atau pengendapan material di tempat itu juga. Proses pembentukan batuan sedimen kelompok ini dapat secara kimiawi, biologi atau organik, dan kombinasi di antara keduanya *(biokimia)*. Secara kimia, endapan terbentuk sebagai hasil reaksi kimia, misalnya CaO + CO2 ® CaCO3. Secara organik adalah pembentukan sedimen oleh aktivitas binatang atau tumbuh-tumbuhan, sebagai contoh pembentukan rumah binatang laut (karang), terkumpulnya cangkang binatang (fosil), atau terkuburnya kayu-kayuan sebagai akibat penurunan daratan menjadi laut. Contohnya; Batu gamping *(Limestone)*, batu bara *(Coal)*, dan lain-lain. Batuan sedimen yang terbentuk dari hasil reaksi kimia atau bisa juga dari kegiatan organisme. Reaksi kimia yang dimaksud adalah kristalisasi langsung atau reaksi organik (Pettjohn, 1975). Menurut R.P. Koesoemadinata, 1981 batuan sedimen dibedakan menjadi enam golongan yaitu :

a) Golongan detritus dasar

Batuan sedimen diendapkan dengan proses mekanis. Termasuk dalam golongan ini antara lain adalah breksi, konglomerat dan batu pasir. Lingkungan tempat pengendapan batuan ini di lingkungan sungai dan danau atau laut.

b) Golongan detritus halus

Batuan yang termasuk kedalam golongan ini diendapkan di lingkungan laut dangkal sampai laut dalam. Yang termasuk kedala golongan ini adalah batu lanau, serpih, batu lempung dan nepal.

c) Golongan karbonat

Batuan ini umum sekali terbentuk dari kumpulan cangkang *moluska, algae* dan *foraminifera,* atau oleh proses pengendapan yang merupakan rombakan dari batuan yang terbentuk lebih dahulu dan diendapkan disuatu tempat. Proses pertama biasa terjadi di lingkungan laut litoras sampai neritik, sedangkan proses kedua diendapkan pada lingkungan laut neritik sampai bahtial. Jenis batuan karbonat ini banyak sekali macamnya tergantung pada material penyusunnya.

d) Golongan silikat

Proses terbentuknya batuan ini adalah gabungan antara pross organik dan kimiawi untuk lebih menyempurnakannya. Termasuk golongan ini yaitu tanah diatom. Batuan golongan ini tersebarnya hanya sedikit dan terbatas sekali.

e) Golongan evaporit

Proses terjadinya batuan sedimen ini harus ada air yang memiliki larutan kimia yang cukup pekat. Pada umumnya batuan ini terbentuk di lingkungan danau atau laut yang tertutup, sehingga sangat memungkinkan terjadi pengayaan unsure-unsur tertentu. Dan faktor yang penting juga adalah tingginya penguapan maka akan terbentuk suatu endapan dari larutan tersebut. Batuan-batuan yang termasuk kedalam batuan ini adalah *gips*, *danhidrit*, batu garam.

f) Golongan batu bara

Batuan sedimen ini terbentuk dari unsur-unsur organik yaitu dari tumbuh-tumbuhan. Dimana sewaktu tumbuhan tersebut mati dengan cepat tertimbun oleh suatu lapisan yang tersebut di atasnya sehingga tidak akan memungkinkan terjadinya pelapukan. Lingkungan terbentuknya batubara adalah khusus sekali, ia harus memiliki banyak sekali tumbuhan sehingga kalau timbunan itu mati tertumpuk menjadi satu di tempat tersebut.

2.1.5 Kandungan Batuan Sedimen Karbonat

Karbonat adalah batuan sedimen yang terbentuk dari mineral karbonat yang utama yaitu kalsit dan dolomit serta kadang-kadang aragonite (tidak stabil). Batuan sedimen karbonat terdiri CaCO3 (kalsium karbonat) dan MgCO3. Batuan sedimen terdiri dari susunan mineral-mineral karbonat (senyawanya terdiri dari atom tunggal karbon dan tiga atom oksigen, dimana karbonat mempunyai struktur kimia CO3. Batuan utamanya yaitu batuan *gamping* atau *limestone* (CaCO3>50%) dan dolomit *(dolostone)* (CaMg(CO3)2). Batuan ini mengandung sekurang-kurangnya 22% dari keseluruhan batuan sedimen yang ada. Batuan sedimen karbonat mengandung mineral :

- Kalsit *(Calcite)* CaCO2 (yang utama) dengan struktur kristal *rhombohedral* (heksagonal) banyak ditemukan pada batuan sedimen tua, bentuknya stabil dan kalsit mendominasi bahan rangka selama *Paleozoic*, kalsit *low* magnesium (<4%) dan *high* magnesium (>4%)
- Dolomit *(dolomite)* CaMg(CO3)2 dengan struktur kristal *rhombohrdral (*heksagonal), berasosiasi dengan mineral kalsit dan *evaporate*, batuan gamping berubah oleh *fluida* yang kaya akan magnesium, *volume*-nya berkurang sebesar 12% yang direpresentasikan oleh *void* atau *vugs*.
- Aragonit *(aragonite)* CaCO3 dengan struktur kristal *orthorhombic*, saat ini banyak ditemukan dalam batuan sedimen karbonat (zaman Cenozoic), dan metastabil (dapat berubah menjadi kalsit).
- Beberapa karbonat magnesium *(magnesite)* MgCO3. Dengan struktur kristal heksagonal *(rhombohedral).*

2.2 Citra Digital

Citra digital adalah gambar dua dimensi yang dapat ditampilkan pada layar monitor komputer sebagai himpunan berhingga (*diskrit*) nilai digital yang disebut pixel (*picture elements).* Piksel adalah elemen citra yang memiliki nilai yang menunjukkan intensitas warna.

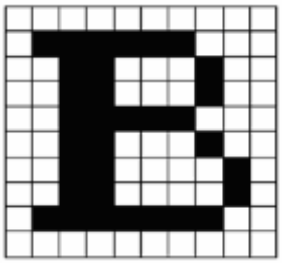
Berdasarkan cara penyimpanan atau pembentukannya, citra digital dapat dibagi menjadi dua jenis. Jenis pertama adalah citra digital yang dibentuk oleh

kumpulan *pixel* dalam *array* dua dimensi. Citra jenis ini disebut citra *bitmap* (*bitmap image*) atau citra raster *(raster image)*. Jenis citra yang kedua adalah citra yang dibentuk oleh fungsi-fungsi geometri dan matematika. Jenis citra ini disebut grafik vektor (*vector graphics*).

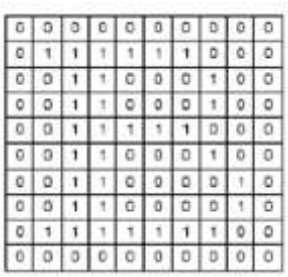
Dalam pembahasan tugas akhir ini, yang dimaksud citra digital adalah citra *bitmap*. Citra digital (*diskrit*) dihasilkan dari citra analog (*kontinu*) melalui digitalisasi. Digitalisasi citra analog terdiri atas *sampling* dan kuantisasi (*quantization*) Sampling adalah pembagian citra ke dalam elemen-elemen diskrit *(pixel*), sedangkan kuantisasi adalah pemberian nilai intensitas warna pada setiap *pixel* dengan nilai yang berupa bilangan bulat (G.W. Awcock, 1996).

2.3 Citra Biner

Citra biner adalah gambar pada bidang dua dimensi. Dalam tinjauan matematis, citra yang hanya terdiri atas dua warna, yaitu hitam dan putih. Oleh karena itu, setiap *pixel* pada citra biner cukup dipresentasikan dengan satu bit. Pada gambar 2.2 merupakan citra biner, sedangkan pada gambar 2.3 merupakan representasi dari citra biner, dimana citra yang berwarna putih memiliki nilai satu, sedangkan citra yang berwarna hitam memiliki nilai nol.



Gambar 2.2 Citra biner



Gambar 2.3 Representasi citra biner

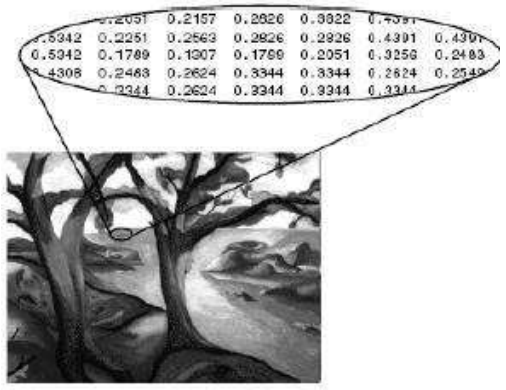
Meskipun saat ini citra berwarna lebih disukai karena memberi kesan yang lebih kaya daripada citra biner, namun tidak membuat citra biner tidak digunakan lagi. Pada beberapa aplikasi citra biner masih tetap dibutuhkan, misalkan citra logo instansi (yang hanya terdiri dari warna hitam dan putih), citra kode barang (*barcode*) yang tertera pada label barang, citra hasil pemindaian dokumen teks, dan sebagainya. Seperti yang sudah disebutkan diatas, citra biner hanya mempunyai dua nilai derajat keabuan : hitam dan putih. *Pixel-pixel* objek bernilai satu dan *pixel-pixel* latar belakang bernilai nol. Pada waktu menampilkan gambar, adalah putih dan satu adalah hitam. Jadi pada citra biner, latar belakang berwarna putih sedangkan objek berwarna hitam seperti tampak pada gambar 2.2 diatas. Meskipun komputer saat ini dapat memproses citra putih-abu abu (*grayscale*) maupun citra berwarna, namun citra biner masih tetap di pertahankan keberadaannya.

2.3.1 Citra Grayscale

Dalam komputasi, suatu citra digital *grayscale* adalah suatu citra dimana nilai dari setiap *pixel* merupakan sampel tunggal. Citra yang ditampilkan dari citra jenis ini terdiri atas warna abu-abu, bervariasi pada warna hitam pada bagian yang intensitas terlemah dan warna putih pada intensitas terkuat. Citra *grayscale* berbeda dengan citra hitam-putih *(black* dan *write)*, dimana pada konteks komputer, citra hitam putih hanya terdiri atas dua warna saja yaitu hitam dan putih saja. Pada citra *grayscale* warna bervariasi antara putih dan abu-abu, tetapi variasi warna diantaranya sangat banyak. Citra *grayscale* seringkali merupakan perhitungan dari intensitas cahaya pada setiap *pixel* pada *spektrum elektromagnetik.*

Citra *grayscale* disimpan dalam format delapan *bit* untuk setiap sampel piksel, yang memungkinkan sebanyak 256 intensitas. Format ini sangat membantu dalam pemrograman karena manipulasi bit yang tidak terlalu banyak. Pada aplikasi lain seperti pada aplikasi *medical imaging* dan *remote sensing* biasa juga digunakan format 10 bit, 12 bit dan 16 bit yaitu citra yang nilai *pixel*-nya merepresentasikan derajat keabuan atau intensitas warna putih. Nilai intensitas paling rendah merepresentasikan warna hitam dan nilai intensitas paling tinggi

merepresentasikan warna putih. Pada umumnya citra *grayscale* memiliki kedalaman *pixel* delapan bit (256 derajat keabuan), tetapi ada juga citra *grayscale* yang kedalaman *pixel*-nya bukan delapan bit, misalnya 16 bit untuk penggunaan yang memerlukan ketelitian tinggi. Pada gambar 2.4 merupakan contoh citra *grayscale.*



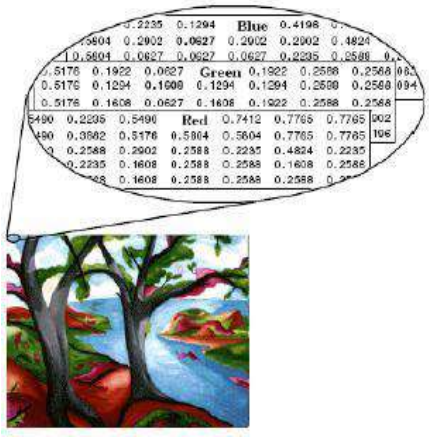
Gambar 2.4 Citra *grayscale*

Citra *grayscale* merupakan citra satu kanal, dimana citra f(x,y) merupakan fungsi tingkat keabuan dari hitam keputih, x menyatakan *variable* kolom atau posisi pixel di garis jelajah dan y menyatakan *variable* kolom atau posisi pixel di garis jelajah. Intensitas f dari gambar hitam putih pada titik (x,y) disebut derajat keabuan (*grey level*), yang dalam hal ini derajat keabuannya bergerak dari hitam keputih. Derajat keabuan memiliki rentang nilai dari ***I min*** sampai ***I max***, atau ***Imin*** < f < ***Imax***, selang (***Imin, Imax***) disebut skala keabuan. Biasanya selang (***Imin, Imax***) sering digeser untuk alasan-alasan praktis menjadi selang [0,L], yang dalam hal ini nilai intensitas nol meyatakan hitam, nilai intensitas L meyatakan putih, sedangkan nilai intensitas antara nol sampai L bergeser dari hitam ke putih. Sebagai contoh citra *grayscale* dengan 256 level artinya mempunyai skala abu dari nol sampai 255 atau [0,255], yang dalam hal ini intensitas nol menyatakan hitam, intensitas 255 menyataka putih, dan nilai antara nol sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih.

2.3.2 Citra Asli

Citra asli yaitu citra yang nilai pixel-nya merepresentasikan warna tertentu. Banyaknya warna yang mungkin digunakan bergantung kepada

kedalaman pixel citra yang bersangkutan. Citra berwarna direpresentasikan dalam beberapa kanal yang menyatakan komponen-komponen warna penyusunnya. Banyaknya kanal yang digunakan bergantung pada model warna yang digunakan pada citra tersebut. Pada gambar 2.5 merupakan contoh citra RGB.



Gambar 2.5 Citra RGB

Intensitas suatu pada titik pada citra berwarna merupakan kombinasi dari tiga intensitas : derajat keabuan merah (fmerah(x,y)), hijau (fhijau(x,y)), dan biru (fbiru(x,y)). Persepsi visual citra berwarna umumnya lebih kaya di bandingkan dengan citra hitam putih. Citra berwarna menampilkan objek seperti warna aslinya meskipun tidak selalu tepat demikian. Warna-warna yang diterima oleh mata manusia merupakan hasil kombinasi cahaya dengan panjang gelombang berbeda.

2.4 Tentang Jaringan Syaraf Tiruan (JST)

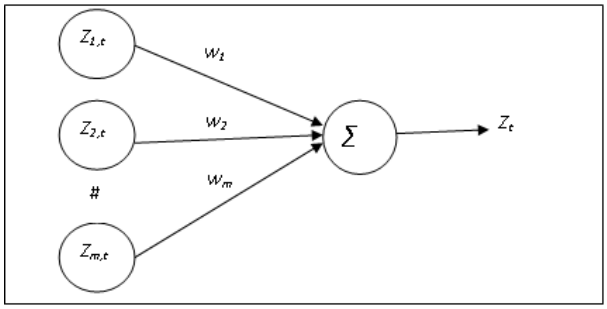
Jaringan Syaraf Tiruan (JST) atau yang biasa disebut *Artificial Neural Network* (ANN) atau *Neural Network* (NN) saja, merupakan sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf pada makhluk hidup. *Neural Network* berupa suatu model sederhana dari suatu syaraf nyata dalam otak manusia seperti suatu unit *threshold* yang *biner*. *Neural network* merupakan sebuah mesin pembelajaran yang dibangun dari sejumlah elemen pemrosesan sederhana yang disebut *neuron* atau *nod*e. Setiap *neuron* dihubungkan dengan *neuron* yang lain dengan hubungan komunikasi langsung melalui pola hubungan yang disebut arsitektur jaringan. Bobot-bobot pada koneksi mewakili besarnya informasi yang digunakan jaringan. Metode yang digunakan untuk menentukan bobot koneksi tersebut dinamakan dengan *algoritma* pembelajaran.

Setiap *neuron* mempunyai tingkat aktivasi yang merupakan fungsi dari input yang masuk padanya. Aktivasi yang dikirim suatu *neuron* ke *neuron* lain berupa sinyal dan hanya dapat mengirim sekali dalam satu waktu, meskipun sinyal tersebut disebarkan pada beberapa *neuron* yang lain.

Misalkan input Z1,t, Z2,t, …, Zm,t yang bersesuaian dengan sinyal dan masuk ke dalam saluran penghubung. Setiap sinyal yang masuk dikalikan dengan bobot koneksinya yaitu w1, w2, …, wm sebelum masuk ke blok penjumlahan yang berlabel ∑. Kemudian blok penjumlahan akan menjumlahkan semua input terbobot dan menghasilkan sebuah nilai yaitu Zt_in.

$$Z_{t_}in = .w_i = Z_{t,1}.w_1 + Z_{t,1}.w_2 + \ldots + Z_{m,1}.w_m$$

Aktivasi Zt ditentukan oleh fungsi input jaringannya, Zt=f(Zt_in) dengan f merupakan fungsi aktivasi yang digunakan.



Gambar 2.6 Ilustrasi struktur jaringan syaraf tiruan

2.4.1 Konsep Dasar Jaringan Syaraf Tiruan (JST)

Jaringan syaraf tiruan terdiri dari beberapa *neuron* dan ada hubungan antar *neuron- neuron* seperti pada otak manusia. *Neuron* atau *sel saraf* adalah sebuah unit pemroses informasi yang merupakan dasar operasi jaringan syaraf tiruan. Jaringan syaraf tiruan terdiri atas beberapa elemen penghitung tak linier yang masing-masing dihubungkan melalui suatu pembobot dan tersusun secara paralel. Pembobot inilah yang nantinya akan berubah beradaptasi selama proses pelatihan. Pelatihan perlu dilakukan pada suatu jaringan syaraf tiruan sebelum digunakan untuk menyelesaikan masalah. Hasil pelatihan jaringan syaraf tiruan dapat diperoleh tanggapa yang benar terhadap masukan yang diberikan terkena *derau* atau berubah oleh suatu keadaan.

A. Faktor Bobot

Faktor bobot merupakan suatu nilai yang mendefinisikan tingkat atau kepentingan hubungan antara suatu *node* dengan *node* yang lain. Semakin besar bobot suatu hubungan menandakan semakin pentingnya hubungan kedua *node* tersebut. Bobot merupakan suatu hubungan berupa bilangan *real* maupun *integer*, tergantung dari jenis permasalahan dan model yang digunakan. Bobot-bobot tersebut bisa ditentukan untuk berada didalam interval tertentu. selama proses pelatihan, bobot tersebut dapat menyesuaikan dengan pola-pola input. Jaringan dengan sendirinya akan memperbaiki diri terus-menerus karena adanya kemampuan untuk belajar. Setiap ada suatu masalah baru, jaringan dapat belajar dari masalah baru tersebut, yaitu dengan mengatur kembali nilai bobot untuk menyesuaikan karakter nilai.

B. Fungsi Aktivasi

Setiap *neuron* mempunyai keadaan internal yang disebut *level* aktivasi atau *level* aktivitas yang merupakan fungsi input yang diterima. Secara tipikal suatu *neuron* mengirimkan aktivitasnya kebeberapa *neuron* lain sebagai sinyal. Yang perlu diperhatikan adalah bahwa *neuron* hanya dapat mengirimkan satu sinyal sesaat, walaupun sinyal tersebut dapat dipancarkan ke beberapa *neuron* yang lain.

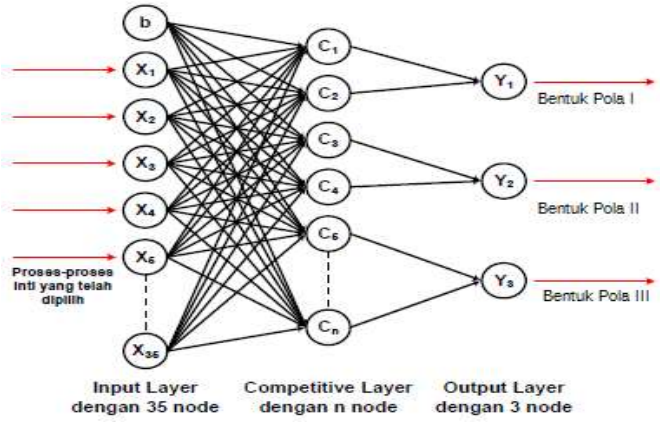
2.4.2 Karekteristik Dasar Jaringan Syaraf Tiruan (JST)

Penyelesaian masalah dengan jaringan syaraf tiruan tidak memerlukan pemrograman. Jaringan syaraf tiruan menyelesaikan masalah melalui proses belajar dari contoh-contoh pelatihan yang diberikan. Biasanya pada jaringan syaraf tiruan diberikan sebuah himpunan pola pelatihan yang terdiri dari sekumpulan contoh pola. Proses belajar jaringan syaraf tiruan berasal dari serangkaian contoh-contoh pola yang diberikan. metode pelatihan yang sering dipakai adalah metode belajar terbimbing. Selama proses belajar itu pola masukan disajikan bersama-sama dengan pola keluaran yang diinginkan. Jaringan akan menyesuaikan nilai bobotnya sebagai tanggapan atas pola masukan dan sasaran yang disajikan tersebut.

2.5 Metode *Learning Vector Quantization* (LVQ).

Metode *Learning Vektor Quantization* (LVQ) yaitu merupakan metoda untuk melakukan pelatihan terhadap lapisan-lapisan kompetitif *supervised*. Lapisan kompetitif akan belajar secara otomatis untuk melakukan klasifikasi terhadap vektor input yang diberikan. Apabila beberapa vektor input memiliki jarak yang sangat berdekatan, maka vektor-vektor input tersebut akan dikelompokkan dalam kelas yang sama.

Dari hal ini LVQ merupakan metode klasifikasi data adaptif berdasarkan pada data pelatihan dengan informasi kelas yang diinginkan yang bisa pula di ambil dari sebuah JST dari *levenberg Marquardt* yang di ambil yaitu nilai jumlah pelatihan *epoh*-nya dan nilai *error (MSE)*. Walaupun merupakan suatu metoda pelatihan *supervised* tetapi LVQ menggunakan teknik data *clustering unsupervised* untuk pra proses set data dan penentuan *cluster center*-nya. Arsitektur jaringan LVQ hampir menyerupai suatu jaringan pelatihan kompetitif kecuali pada masing-masing unit *output*-nya yang dihubungkan dengan suatu kelas tertentu. Proses pelatihan pada jaringan dilakukan untuk mengeset nilai bobot jaringan (w_i) yang akan mengalami perubahan pada setiap pelatihan yang dilakukan.



Gambar 2.7 Arsitektur *Learning Vector Quantization (LVQ)*

2.5.1 Cara Kerja *Learning Vector Quantization (*LVQ)

Pada Metoda pembelajaran yang dipilih pada penelitian ini adalah *Learning Vector Quantization* (LVQ). Algoritma pembelajaran LVQ terdiri dari dua tahap, yaitu :

1) Metoda pelajaran data *unsupervised* digunakan untuk menempatkan beberapa *cluster center* tanpa menggunakan informasi kelas.
2) Informasi kelas digunakan untuk *fine-tune cluster center* untuk memperkecil banyaknya kasus kesalahan klasifikasi.

2.5.2 Proses Pelatihan *Learning Vector Quantization* (LVQ)

Dari tujuan diadakannya proses pelatihan pada JST adalah agar JST dapat diandalkan dalam mengenali pola yang diinputkan sehingga dapat diperoleh hasil yang diinginkan. Karena itu sangatlah penting untuk melatih JST dengan *training set* yang sesuai dengan tugas JST itu sendiri dalam mengenali suatu pola bentuk pada permukaan batuan.

Kinerja yang diukur pada proses pelatihan jaringan syaraf tiruan pada metode *Learning Vector Quantization* adalah :

- Lamanya proses pelatihan jumlah *(epoch)*

  Semakin cepat proses pelatihan maka semakin baik kinerja proses pelatihan tersebut. Hal ini berarti jumlah *epoch* untuk mencapai nilai *error* yang diinginkan adalah semakin mendekati nol.
- Perhitungan *error* (MSE).

  Perhitungan *error* merupakan pengukuran bagaimana jaringan dapat belajar dengan baik sehingga jika dibandingkan dengan pola yang baru akan dengan mudah dikenali. Apabila nilai *error* ini masih cukup besar, hal ini mengidikasikan bahwa masih perlu dilakukan lebih banyak pembelajaran lagi hingga nilai *error* mendekati nol. Apabila *output* jaringan tepat sama dengan target akan bernilai nol.

Koefisien *regresi* (R) terhadap respon *output* jaringan dan target yang diinginkan. Apabila *output* jaringan tepat sama dengan target maka koefisien *regresi* akan bernilai satu.

2.6 Dasar Pengolahan Citra Digital

Citra digital (*digital image*) merupakan proses mengolah piksel- piksel dalam citra digital untuk suatu tujuan tertentu. Beberapa alasan dilakukan pengolahan citra digital antara lain:

1. Untuk mendapatkan citra asli dari suatu citra yang sudah buruk karena pengaruh *derau* (*noise*). Proses pengolahan bertujuan mendapatkan citra yang diperkirakan mendekati citra sesungguhnya.
2. Untuk memperoleh citra dengan karakteristik tertentu dan cocok secara visual yang dibutuhkan untuk tahap yang lebih lanjut dalam pemrosesan analisis citra.

Dalam proses akuisisi, citra yang akan diolah ditransformasikan dalam suatu representasi numerik. Pada proses selanjutnya representasi tersebut yang akan diolah secara digital oleh komputer. Pengolahan citra pada umumnya dapat dikelompokkan dalam dua jenis kegiatan yaitu:

1. Memperbaiki kualitas citra sesuai kebutuhan
2. Mengolah informasi yang terdapat pada citra

Bidang aplikasi yang kedua ini sangat erat kaitannya dengan *computer aided analysis* yang umumnya bertujuan untuk mengolah suatu objek citra dengan cara mengekstraksi informasi penting yang terdapat didalamnya. Dari informasi tersebut dapat dilakukan proses analisis dan klasifikasi secara cepat dengan memanfaatkan *algoritma* komputer. Dari pengolahan citra diharapkan terbentuk suatu sistem yang dapat memproses citra masukan hinggacitra tersebut dapat dikenali cirinya. Pengenalan ciri inilah yang sering diaplikasikan dalam kehidupan sehari- hari.

2.7 Thresholding

Thresholding adalah proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih sehingga dapat diketahui daerah mana yang termasuk obyek dan *background* dari citra secara jelas. Selama proses *thresholding*, setiap pixel dalam foto ditandai sebagai "objek" *pixel* jika nilai mereka adalah lebih besar dibandingkan nilai ambang (asumsi obyek menjadi lebih terang dari pada latar belakang) dan sebagai "latar belakang" *pixel* lain. Konvensi ini dikenal sebagai ambang di atas. Varian termasuk di bawah ambang batas yang berlawanan dari ambang di atas, di dalam batas, dimana piksel yang berlabel "obyek" jika ada di antara dua nilai *thresholds* dan di luar batas, yang merupakan kebalikan dari dalam ambang (Shapiro, dkk. 2001: 83). Biasanya, obyek piksel diberi nilai satu sedangkan piksel latar belakang diberi nilai nol akhirnya, biner gambar yang dibuat oleh setiap piksel warna putih atau hitam, tergantung pada *pixel* labelnya. Citra hasil *thresholding* biasanya digunakan lebih lanjut untuk proses pengenalan objek serta ekstraksi fitur. Metode *thresholding* secara umum dibagi menjadi dua, yaitu :

2.7.1 Thresholding global

Thresholding global adalah *thresholding* yang dilakukan dengan mempartisi histogram dengan menggunakan sebuah *threshold* (batas ambang) *global* T, yang berlaku untuk seluruh bagian pada citra.

2.7.2 Thresholding adaptif

Thresholding adaptif adalah *thesholding* yang dilakukan dengan membagi citra menggunakan beberapa sub citra. Lalu pada setiap sub citra, *segmentasi* dilakukan dengan *threshold* yang berbeda.

Yang menjadi fokus dalam tugas akhir ini adalah metode *thresholding global*, *thresholding* dikatakan *global* jika nilai *threshold* T hanya bergantung pada *f(x,y)*, yang melambangkan tingkat keabuan pada titik *(x,y)* dalam suatu citra. Citra hasil *thresholding* dapat didefinisikan sebagaimana Persamaan 2.1 Setelah

proses *treshold* selesai dilakukan, kemudian dilakukan denoising pada image sel darah.

$$g(x,y) = \begin{cases} 1 & if \quad f(x,y) > T \\ 0 & if \quad f(x,y) \le T \end{cases} \qquad (2.1)$$

2.8 Pengolahan Citra Menggunakan MATLAB.

Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra.

Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh *bit-bit* tertentu. Kebutuhan untuk pengolahan citra secara mudah dan cepat sangat diperlukan. Penelitian ataupun penerapan di lapangan yang melibatkan proses pengolahan citra, kadang-kadang menyulitkan dalam bidang pemrograman karena rutin program yang berhubungan dengan komputer grafik membutuhkan keahlian khusus dalam implementasinya. Matlab sebagai salah satu *tools* pemrograman untuk membantu bidang pendidikan dan penelitian telah menyediakan bermacam-macam '*toolbox*' yang disesuaikan dengan bidang keilmuan masing-masing, salah satunya adalah '*Image Processing Toolbox'.* Dengan memanfaatkan '*toolbox*' tersebut, pengguna dapat dengan mudah melakukan penelitiannya.

Matlab adalah sebuah bahasa (pemrograman) dengan unjuk kerja tinggi untuk komputasi teknis, yang mengintegrasikan komputasi, visualisasi, dan pemrograman di dalam lingkungan yang mudah penggunaannya dalam memmecahkan persoalan dengan solusinya yang dinyatakan dengan notasi matematik. Sistem matlab terdiri dari lima bagian utama, yaitu:

1. Bahasa pemrograman MATLAB

   Bagian ini adalah bahasa (pemrograman) tingkat tinggi yang menggunkan *matriks* atau *array* dengan pernyataan aliran kendali

program, struktur data, masukan dan keluaran, serta fitur-fitur pemrograman berorientasi objek.

2. Lingkungan kerja MATLAB

   Bagian ini adalah sekumpulan kakas dan fasilitas MATLAB yang digunakan oleh pengguna atau pemogram.

3. Penanganan grafik

   Bagian ini adalah sistem grafik MATLAB, termasuk perintah-perintah program tingkat tinggi untuk visualisasi data dua dimensi dan tiga dimensi, pengolahan citra, animasi, dan presentasi grafik. Selain itu bagian ini juga termasuk perintah-perintah program tingkat rendah untuk menetapkan sendiri tampilan grafik seperti halnya membuat antarmuka pengguna grafis untuk aplikasi-aplikasi MATLAB.

4. Pustaka *(library)* fungsi matematis MATLAB

   Bagian ini adalah koleksi *algoritma* komputasi mulai dari fungsi dasar seperti menjmlahkan *(sum),* menentukan nilai *sinus*, *kosinus*, dan aritmatika bilangan kompleks, fungsi-fungsi seperti *inverse matriks*, nilai *eigen matriks*, fungsi *Bessel*, dan FFT (*Fast Fourier Transform*).

5. API (*Application Program Interface*) MATLAB

   Bagian ini adalah pustaka *(library)* untuk menuliskan program bahasa C dan *fortran* berinteraksi dengan MATLAB, termasuk fasilitas untuk memanggil delapan rutin program dari MATLAB *(Dynamic Lingking),* memanggil MATLAB sebagai mesin komputasi dan untuk pembacaan serta penulisan MAT-*Files.*

2.8.1 Software Matlab

Matlab adalah sebuah bahasa (pemrograman) dengan kinerja yang tinggi (*high-performance*) untuk komputasi teknis, yang mengintegrasikan komputasi, visualisasi dan pemrograman di dalam lingkungan yang mudah penggunaannya dalam memecahkan persoalan dengan solusinya yang dinyataka dengan notasi matematik. Penggunaan matlab, yaitu:

- Matematik dan komputasi
- Pengembangan algoritma

- ❖ Pemodelan, simulasi dan pembuatan *prototipe*
- ❖ Grafik untuk sains dan teknik
- ❖ Pengembangan aplikasi, termasuk pembuatan antarmuka grafis untuk pengguna GUI (*Graphical User Interface*).

Matlab adalah sistem interaktif yang menggunakan elemen data dasarnya adalah *array* yang tidak membutuhkan dimensi. Hal ini mempermudah untuk menyelesaikan masalah komputasi terutama menyangkut *matriks* dan *vector*.

Nama matlab merupakan singkatan dari *matrix laboratory*. Pada awalnya matlab dibuat untuk mempermudah pengembangan perangkat lunak berbasis matriks oleh proyek *LINPACK* dan *EISPACK*.

Matlab memiliki beberapa keunggulan dan kemudahan dalam desain dan analisis sistem kendali dan sistem linear. Fungsi-fungsi pendukung terangkum dalam *Control System Toolbox* dan *Signal Proccesing Toolbox*. Kebanyakan operasi dalam sistem kendali dan sistem linear berhubungan dengan operasi matriks, aritmatika kompleks, menghitung akar, model konversi, dan penggambaran fungsi-fungsi yang kompleks. Seperti telah kita ketahui sebelumnya, matlab telah didesain untuk memudahkan pengerjaan operasi-operasi tersebut diatas, dan analisa fungsi. Pelatihan ini lebih ditekankan pada pemrograman dan menggunakan fungsi-fungsi dasar dalam bahasa matlab sehinga teori mengenai ilmu sistem itu sendiri hanya akan disinggung sedikit pada topik-topik yang *relevan.*

2.8.2 Spesifikasi GUI *(Graphical User Interface)* Matlab

Berikut ini akan dibahas teori dasar antarmuka sistem pengguna. teori antarmuka meliputi perancangan menu-menu pada aplikasi yang akan dibuat. Perancangan dilakukan dengan menggunakann *tools* yang sudah tersedia di matlab, *tools* tersebut dapat diakses langsung dengan cara mengetikkan *guide* pada *Command Window* untuk membuat dan memprogram antarmuka aplikasi muncul.

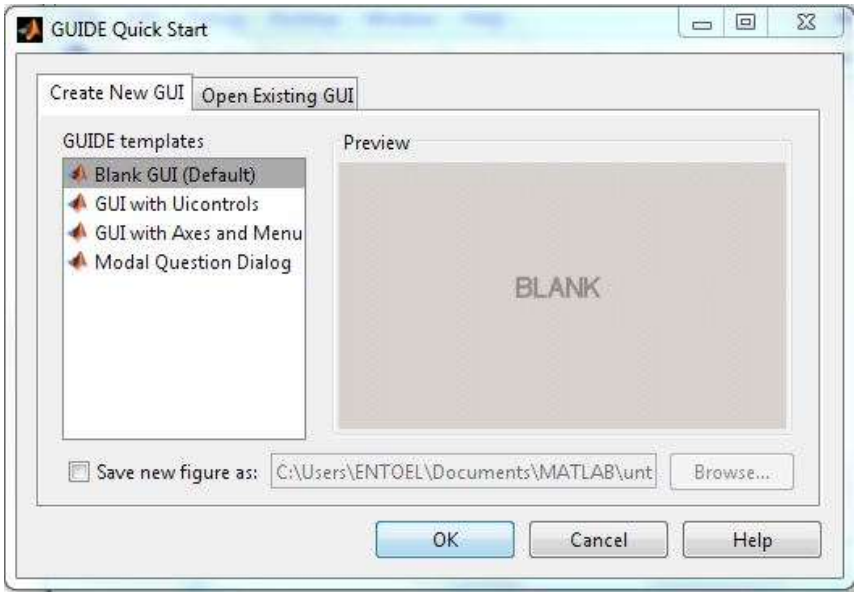
Memulai *GUIDE*:

1. Pertama ketikkan *guide* di *command window* yang sudah dibuka, seperti terlihat pada gambar 2.8.



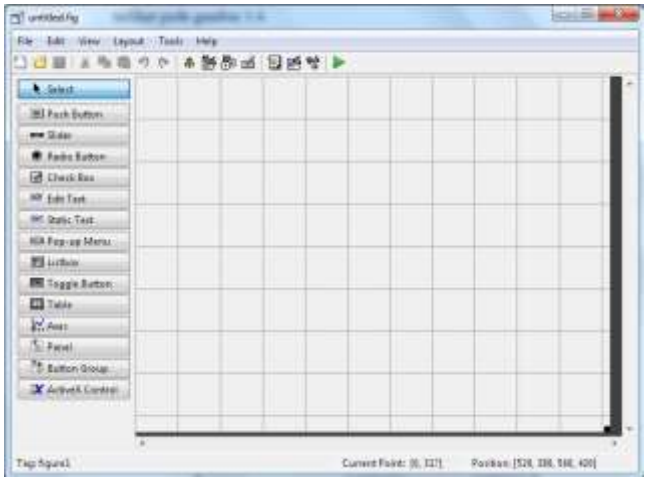
Gambar 2.8 *Command window*

2. Setelah selesai langkah diatas, maka akan muncul tampilan seperti gambar 2.9 dan pilih *Blank GUI* (*Default*).



Gambar 2.9 *Guide quick start*

3. Gambar 2.10 adalah layar untuk mendesain antarmuka aplikasi yang akan dibuat.



Gambar 2.10 *GUI editor*

4. Gambar 2.11 adalah pilihan *element tools* yang bisa dipakai untuk merancang antarmuka.



Gambar 2.11 *Element tools.*

2.9 Software Antarmuka Matlab

Matlab disini yaitu sebuah bahasa pemrograman dengan kinerja yang tinggi (*high-performance*) untuk komputasi teknis, yang mengintegrasikan komputasi, visualisasi dan pemrograman di dalam lingkungan yang mudah penggunaannya dalam memecahkan persoalan dengan solusinya yang dinyatakan dengan notasi matematik dan grafik. cocok untuk perancangan sistem, pengolahan citra dan jaringan syaraf tiruan.

Berikut merupakan tampilan dari s*oftware* antarmuka sistem dengan Matlab *Versi* 7.1 / 2010b.



Gambar 2.12 Antarmuka sistem

Berikut penjelasan dari tiap komponen pada antarmuka sistem (Gambar 2.12) :

1. **Panel *file***, didalamnya terdapat menu-menu proses pengoahan citra diantaranya :

   a .Pushbutton, untuk menampilkan *image*.

   b. Pushbutton ***refresh***, untuk mereset *file image* yang sudah disimpan dan dikeluarkan hasil pengolahan citra.

   c. Pushbutton ***exit***, untuk *mengelurkan program.*
2. **Panel *prepocessing***, didalamnya terdapat menu-menu proses pengolahan citra diantaranya :
   a. Pushbutton ***Convert to grayscale***
   b. Pushbutton ***Deteksitepi***
3. **Panel *processing***, didalamnya terdapat menu-menu proses pengolahan citra diantaranya :
   a. Pushbutton ***Watershed***
   b. Pushbutton ***Ekstraksi ciri***
   c. Pushbutton ***Labeling***

4. **Panel identifikasi**, didalamnya terdapat menu-menu proses pengolahan citra diantaranya :
   a. Pushbutton **pola wajik**
   b. Pushbutton **bukan wajik**
5. **Panel Informasi**, didalamnya terdapat menu-menu proses pengolahan citra diantaranya :
   a. Pushbutton ***trainning JST***
   b. *Pushbutton* **jumlah pola wajik**
   c. Pushbutton **jumlah bukan wajik**
   d. Pushbutton **jumlah keseluruhan**
   e. *Editteks,* **nilai penjumlahan**
6. ***Axes*1,2,3,4,5,6** untuk menampilkan hasil dari semua pengolahan citra.

2.10 Sistem Matlab

Sebagai sebuah sistem matlab tersusun dari lima bagian utama, yaitu :

- *Development environment*. Merupakan sekumpulan perangckat dan fasilitas yang membantu anda untuk menggunakan fungsi-fungsi dan *file-file* matlab. Beberapa perangkat ini merupakan sebuah *Graphical User Interfaces* (GUI). Termasuk didalamnya adalah matlab *desktop*, *ommand window*, *command history*, sebuah *editor*, *debugger*, dan *browsers* untuk melihat *help*, *workspace*, *files*, dan *search path*.
- Matlab *mathematical function library*. Merupakan sekumpulan *algoritma* komputasi mulai dari fungsi-fungsi dasar sepertri: *sum, sin, cos*, dan *complex arithmetic*, sampai dengan fungsi-fungsi yang lebih komplek seperti *matrix inverse, matrix eigenvalues, bessel functions*, dan *fast fourier transforms.*
- Matlab *language*. Merupakan suatu *high level matrix* atau *array language* dengan *control flow statements, functions*, data *structures,* input atau *output,* dan fitur-fitur *object oriented programming*. Ini memungkinkan bagi kita untuk melakukan kedua hal baik pemrograman dalam lingkup sederhana untuk mendapatkan hasil yang cepat dan pemrograman dalam

lingkup yang lebih besar untuk memperoleh hasil-hasil dan aplikasi yang komplek.

- *Graphics* matlab memiliki fasilitas untuk menampilkan vektor dan *matriks* sebagai suatu grafik. Di dalamnya melibatkan *high level functions* (fungsi-fungsi level tinggi) untuk visualisasi data dua dikensi dan data tiga dimensi, *image processing*, *animation*, dan *presentation graphics*. Ini juga melibatkan fungsi *level* rendah yang memungkinkan bagi anda untuk membiasakan diri untuk memunculkan grafik mulai dari bentuk yang sederhana sampai dengan tingkatan *graphical user interfaces* pada aplikasi matlab.

Matlab *Application Program Interface* (API). Merupakan suatu *library* yang memungkinkan program yang telah anda tulis dalam bahasa C dan *Fortran* mampu berinterakasi dengan matlab. Ini melibatkan fasilitas untuk pemanggilan *routines* dari matlab (*dynamic linking*), pemanggilan matlab sebagai sebuah *computational engine*, dan untuk membaca dan menuliskan mat-*files*.