BAB 2
LANDASAN TEORI

2.1 Kecerdasan Buatan

Kecerdasan Buatan (*Artificial Intelligence* atau AI) didefinisikan sebagai kecerdasan yang ditunjukkan oleh suatu *entitas* buatan. Sistem seperti ini umumnya dianggap komputer. Kecerdasan diciptakan dan dimasukkan kedalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia [7].

2.1.1 Definisi Kecerdasan Buatan

Kecerdasan buatan merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan zaman, maka peranan komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu komputer diharapkan dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan manusia.

Manusia bisa menjadi pandai dalam menyelesaikan segala permasalahan di dunia ini karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki seseorang tentu saja diharapkan akan lebih mampu dalam menyelesaikan permasalahan. Namun bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang mereka miliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik. Demikian pula, dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik.

Agar komputer bisa bertindak seperti dan sebaik manusia, maka komputer juga harus diberi bekal pengetahuan dan mempunyai kemampuan untuk menalar. Untuk tu pada *Artificial Intelligence* (AI) akan mencoba untuk memberikan beberapa metoda untuk membekali komputer dengan kedua komponen tersebut agar komputer bisa menjadi mesin yang pintar.

Pengertian kecerdasan buatan dapat dipandang dari berbagai sudut pandang, antara lain [7]:

1. Sudut pandang kecerdasan

   Kecerdasan buatan akan membuat mesin menjadi „cerdas‟ (mampu berbuat seperti apa yang dilakukan oleh manusia).

2. Sudut pandang penelitian

   Kecerdasan buatan adalah suatu studi bagaimana membuat agar komputer dapat melakukan sesuatu sebaik yang dikerjakan oleh manusia.

3. Sudut pandang bisnis

   Kecerdasan buatan adalah kumpulan peralatan yang sangat powerful dan metodologis dalam menyelesaikan masalah-masalah bisnis.

4. Sudut pandang pemrograman

   Kecerdasan buatan meliputi studi tentang pemrograman simbolik, penyelesaian masalah (problem solving) dan pencarian (searching).

   Untuk membuat aplikasi kecerdasan buatan ada 2 bagian utama yang sangat dibutuhkan, yaitu :

a. Basis Pengetahuan (*Knowledge Base*), berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan lainnya.
b. Motor Inferensi (*Inference Engine*), yaitu kemampuan menarik kesimpulan berdasarkan pengalaman.

2.2 Pola

Pola adalah suatu entitas yang terdefinisi dan dapat diidentifikasi serta diberi nama melalui ciri-cirinya (*feature*). Ciri-ciri tersebut digunakan untuk membedakan suatu pola dengan pola yang lainnya. Ciri yang baik adalah ciri yang memiliki daya pembeda yang tinggi, sehingga pengelompokan pola berdasarkan ciri yang dimiliki dapat dilakukan dengan keakuratan yang tinggi.

Pola adalah komposit/gabungan dari ciri yang merupakan sifat dari sebuah objek [8].

Beberapa contoh pola :

1. Huruf, memiliki ciri-ciri seperti tinggi, tebal, titik sudut, dan lengkungan garis.
2. Suara, memiliki ciri-ciri seperti amplitudo, frekuensi, nada, dan intonasi.
3. Tanda tangan, memiliki ciri-ciri seperti panjang, kerumitan, dan tekanan.
4. Sidik jari, memiliki ciri-ciri seperti lengkungan, dan jumlah garis.

Ciri pada suatu pola diperoleh dari hasil pengukuran terhadap objek uji. Khusus pada pola yang terdapat di dalam citra, ciri-ciri yang dapat diperoleh berasal dari informasi [8]:

1. Spasial: intensitas pixel, histogram, …
2. Tepi: arah, kekuatan, …
3. Kontur: garis, elips, lingkaran, …
4. Wilayah/bentuk: keliling, luas, pusat massa, …
5. Hasil transformasi Fourier: frekuensi, …

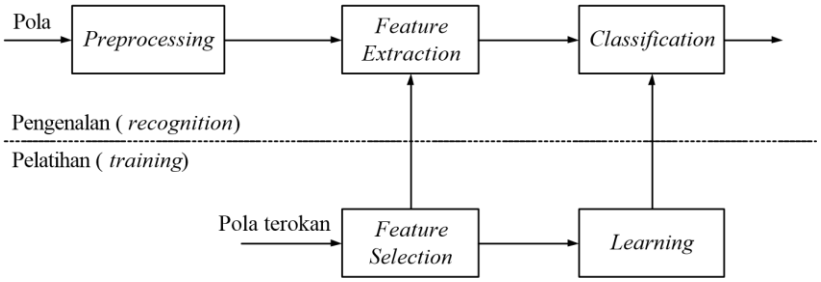
2.2.1 Pengenalan Pola

Pengenalan Pola adalah mengelompokkan data numerik dan simbolik (termasuk citra) secara otomatis oleh mesin (dalam hal ini komputer). Pengenalan pola bertujuan menentukan kelompok atau kategori pola berdasarkan ciri-ciri yang dimiliki oleh pola tersebut. Dengan kata lain, pengenalan pola membedakan suatu objek dengan objek lain. Terdapat dua pendekatan yang dilakukan dalam pengenalan pola: pendekatan secara statistik dan pendekatan secara sintaktik atau struktural [9].

a. Pengenalan Pola secara Statistik

Pendekatan ini menggunakan teori-teori ilmu peluang dan statistik. Ciri-ciri yang dimiliki oleh suatu pola ditentukan distribusi statistiknya. Pola yang berbeda memiliki distribusi yang berbeda pula. Dengan menggunakan teori keputusan di dalam statistik, kita menggunakan distribusi ciri untuk mengklasifikasikan pola.

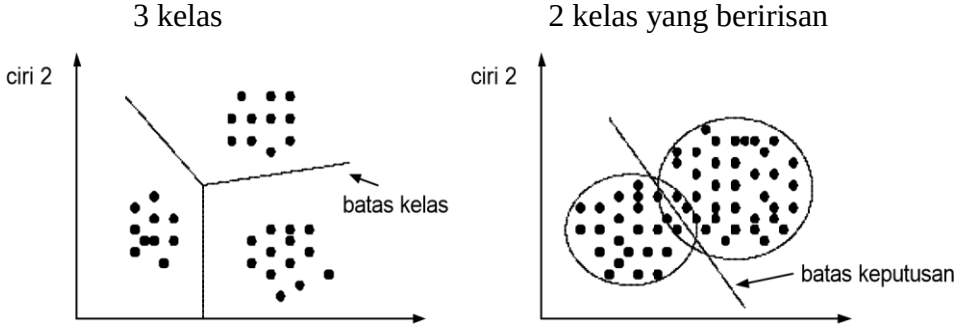
Sistem pengenalan pola dengan pendekatan statistik ditunjukkkan oleh diagram pada Gambar (2.1).



Gambar 1.1 Sistem Pengenalan pola dengan pendekatan statistik

Ada dua fase dalam sistem pengenalan pola: (i) fase pelatihan dan (ii) fase pengenalan. Pada fase pelatihan, beberapa contoh citra dipelajari untuk menentukan ciri yang akan digunakan dalam proses pengenalan serta prosedur klasifikasinya. Pada fase pengenalan, citra diambil cirinya kemudian ditentukan kelas kelompoknya .

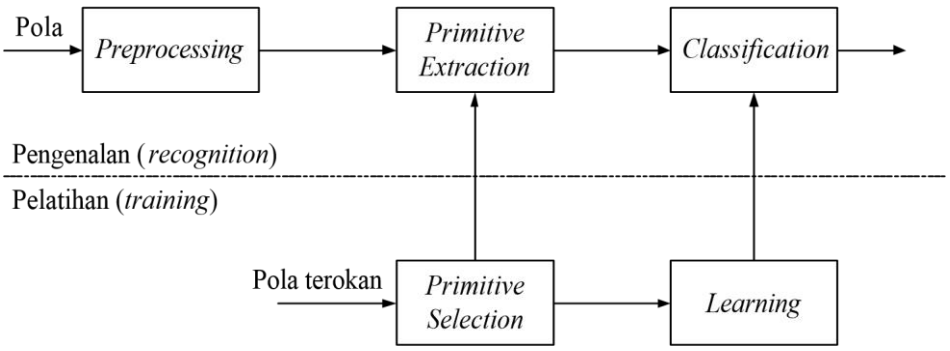
Kumpulan ciri dari suatu pola dinyatakan sebagai vektor ciri dalam ruang bahumatra (multi dimensi). Jadi, setiap pola dinyatakan sebagai sebuah titik dalam ruang bahumatra. Ruang bahumatra dibagi menjadi sejumlah uparuang (sub-ruang). Tiap uparuang dibentuk berdasarkan pola-pola yang sudah dikenali kategori dan ciri-cirinya (melalui fase pelatihan). Lihat Gambar (2.2).



Gambar 1.2 Contoh Pembagian Kelas Pola

b. Pengenalan Pola secara Sintaktik

Pendekatan ini menggunakan teori bahasa formal. Ciri-ciri yang terdapat pada suatu pola ditentukan primitif dan hubungan struktural antara primitif kemudian menyusun tata bahasanya. Dari aturan produksi pada tata bahasa tersebut kita dapat menentukan kelompok pola. Gambar (2.3) memperlihatkan sistem pengenalan pola dengan pendekatan sintaktik. Pengenalan pola secara sintaktik lebih dekat ke strategi pengenalan pola yang dilakukan manusia, namun secara praktek penerapannya relatif sulit dibandingkan pengenalan pola secara statistik.



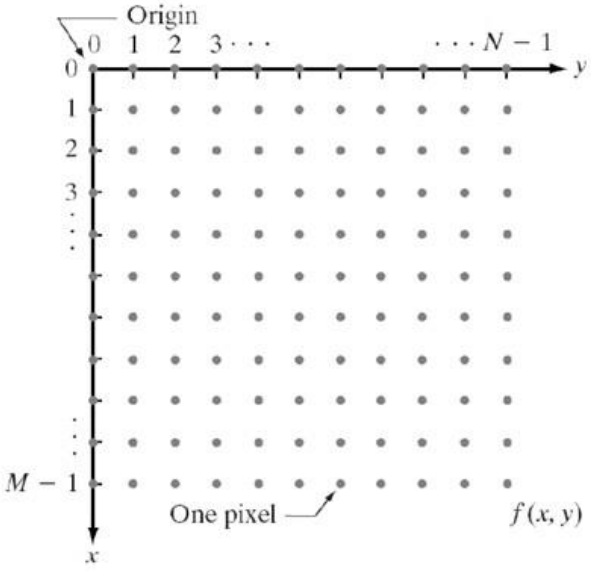
Gambar 1.3 Sistem pengenalan pola dengan pendekatan sintaktik.

Pendekatan yang digunakan dalam membentuk tata bahasa untuk mengenali pola adalah mengikuti kontur (tepi batas) objek dengan sejumlah segmen garis terhubung satu sama lain, lalu mengkodekan setiap garis tersebut (misalnya dengan kode rantai). Setiap segmen garis merepresentasikan primitif pembentuk objek.

2.3 Pengolahan Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar dua dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data dua dimensi. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai *real* maupun komplek yang direpresentasikan dengan deret *bit* tertentu.

Suatu citra dapat didefinisikan sebagai fungsi f(x,y) berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas suatu tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y dan nilai ampitudo f secara keseluruhan sehingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Gambar 2.4 menujukan posisi koordinat citra digital.



Gambar 1.4 Koordinat Citra Digital

Citra digital dapat ditulis dalam bentuk matrik sebagai berikut.

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

Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan *picture elements*, *image elements*, *pels*, atau *pixels*. Istialah terakhir (*pixel*) paling sering digunakan pada citra digital.

2.4 Operasi – Operasi Pengolahan Citra

Operasi-operasi yang dilakukan di dalam pengolahan citra banyak ragamnya. Namun, secara umum, operasi pengolahan citra dapat diklasifikasikan dalam beberapa jenis sebagai berikut [8]:

1. Perbaikan kualitas citra (*image enhancement*).

Jenis operasi ini bertujuan untuk memperbaiki kualitas citra dengan cara memanipulasi parameter-parameter citra. Dengan operasi ini, ciri-ciri khusus yang terdapat di dalam citra lebih ditonjolkan. Contoh-contoh operasi perbaikan citra:

a. perbaikan kontras gelap/terang
b. perbaikan tepian objek (edge enhancement)
c. penajaman (sharpening)
d. pembrian warna semu (pseudocoloring)
e. penapisan derau (noise filtering)

2. Pemugaran citra (image restoration).

Operasi ini bertujuan menghilangkan/meminimumkan cacat pada citra. Tujuan pemugaran citra hampir sama dengan operasi perbaikan citra. Bedanya, pada pemugaran citra penyebab degradasi gambar diketahui. Contoh-contoh operasi pemugaran citra:

a. penghilangan kesamaran (deblurring).
b. penghilangan derau (noise)

3. Pemampatan citra (image compression).

Jenis operasi ini dilakukan agar citra dapat direpresentasikan dalam bentuk yang lebih kompak sehingga memerlukan memori yang lebih sedikit. Hal penting yang harus diperhatikan dalam pemampatan adalah citra yang telah dimampatkan harus tetap mempunyai kualitas gambar yang bagus.

4. Segmentasi citra (image segmentation).

Jenis operasi ini bertujuan untuk memecah suatu citra ke dalam beberapa segmen dengan suatu kriteria tertentu. Jenis operasi ini berkaitan erat dengan pengenalan pola.

5. Pengorakan citra (image analysis)

Jenis operasi ini bertujuan menghitung besaran kuantitif dari citra untuk menghasilkan deskripsinya. Teknik pengorakan citra mengekstraksi ciri-ciri tertentu yang membantu dalam identifikasi objek. Proses segmentasi kadangkala

diperlukan untuk melokalisasi objek yang diinginkan dari sekelilingnya. Contoh-contoh operasi pengorakan citra:

b. Pendeteksian tepi objek (edge detection)
c. Ekstraksi batas (boundary)
d. Representasi daerah (region)

6. Rekonstruksi citra (image reconstruction)

Jenis operasi ini bertujuan untuk membentuk ulang objek dari beberapa citra hasil proyeksi. Operasi rekonstruksi citra banyak digunakan dalam bidang medis. Misalnya beberapa foto rontgen dengan sinar X digunakan untuk membentuk ulang gambar organ tubuh.

7. Perubahan Model Warna

Warna adalah persepsi yang dirasakan oleh sistem visual manusia terhadap panjang gelombang cahaya yang dipantulkan oleh objek [10]. Setiap warna mempunyai panjang gelombang yang berbeda. Warna merah mempunyai panjang gelombang paling tinggi, sedangkan warna ungu mempunyai panjang gelombang paling rendah. Warna-warna yang diterima oleh mata merupakan hasil kombinasi cahaya dengan panjang gelombang berbeda. Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang warna yang paling lebar adalah *red* (R), *green* (G), *blue* (B).

Dalam pembentukan citra digital model warna yang umum digunakan antara lain sebagai berikut :

a. Citra RGB

Citra RGB yang biasa disebut juga citra true color, disimpan dalam citra berukuran (m x n) x 3 yang mendefinisikan warna merah (*red*), hijau (*green*) dan biru (*blue*) untuk setiap pikselnya. Warna pada setiap piksel ditentukan berdasarkan kombinasi dari warna red, green dan blue (RGB). RGB merupakan citra 24 bit dengan komponen merah, hijau, dan biru yang masing-masing umumnya bernilai 8 bit sehingga intensitas kecerahan warna sampai 256 level dan kombinasi warnanya kurang dari sekitar 16 juta warna.

b. Citra Keabuan

Citra dengan derajat keabuan berbeda dengan citra RGB, citra ini didefinisikan oleh satu nilai derajat warna. Umumnya bernilai 8 bit sehingga intensitas kecerahan warna sampai 256 level dan kombinasi warnanya 256 varian. Tingkat kecerahan paling rendah yaitu 0 untuk warna hitam dan putih bernilai 255. Untuk mengkonversikan citra yang memiliki warna RGB ke derajat keabuan bisa menggunakan rumus = $\frac{R+G+B}{3}$ (2.1)

2.5 Histogram Citra

Informasi penting mengenai isi citra digital dapat diketahui dengan membuat histogram citra. Histogram citra adalah grafik yang menggambarkan penyebaran nilai-nilai intensitas pixel dari suatu citra atau bagian tertentu di dalam citra. Dari sebuah histogram dapat diketahui frekuensi kemunculan nisbi (*relative*) dari intensitas pada citra tersebut. Histogram juga dapat menunjukkan banyak hal tentang kecerahan (*brightness*) dan kontas (*contrast*) dari sebuah gambar. Karena itu, histogram adalah alat bantu yang berharga dalam pekerjaan pengolahan citra baik secara kualitatif maupun kuantitatif.

Dengan persamaan rumus : P(fn) = $\frac{nk}{n}$ (2.2)

Dimana :

nk : total nilai fn yang muncul

n : jumlah seluruh pixel

2.6 Tekstur

Tekstur adalah sifat-sifat atau karakteristik yang dimiliki oleh suatu daerah (di dalam citra) yang cukup besar sehingga secara alami sifat-sifat tadi dapat berulang dalam daerah tersebut. Pengertian dari tekstur dalam hal ini adalah keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital [10]. Untuk membentuk suatu tekstur setidaknya ada dua persyaratan yang harus dipenuhi antara lain :

1. Terdiri dari satu atau lebih piksel yang membentuk pola-pola primitif (bagian-bagian terkecil). Bentuk-bentuk pola primitif ini dapat berupa titk, garis

lurus, garis lengkung, luasan dan lain-lain yang merupakan elemen dasar dari sebuah bentuk.

2. Munculnya pola-pola primitif yang berulang-ulang dengan interval jarak dan arah tertentu sehingga dapat diprediksi atau ditemukan karakteristik perulangannya.



Gambar 1.5 Contoh Citra Tekstur Dari Vistex Database

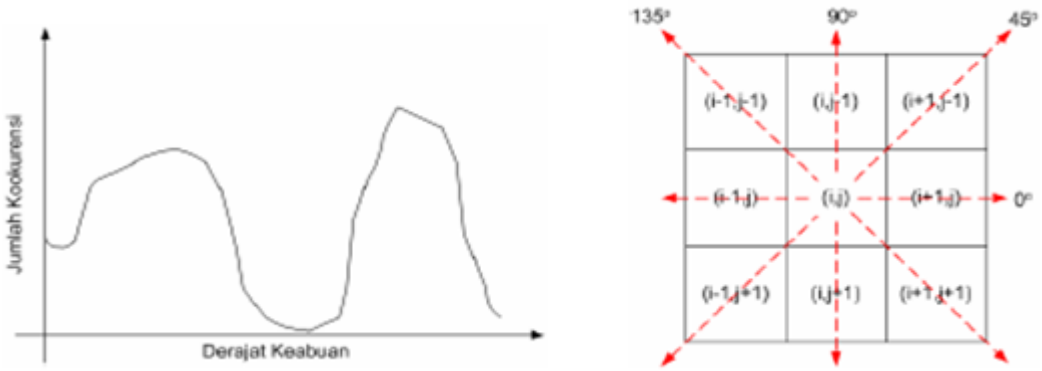
Suatu citra memberikan interpretasi tekstur yang berbeda apabila dilihat dengan jarak dan sudut pandang yang berbeda, manusia memandang tekstur berdasarkan deskripsi yang bersifat acak, seperti halus, kasar, teratur, tidak teratur, dan lain sebgainya. Hal ini merupakan deskripsi yang tidak tepat dan non-kuantitatif, sehingga diperlukan adanya suatu deskrpsi yang kuantitatif (matematis) untuk memudahkan analisis [10].

2.6.1 Analisis Tekstur

Analisis tekstur merukapan dasar dari berbagai macam aplikasi, aplikasi dari analisis tekstur antara lain adalah penginderaan jarak jauh, pencitraan medis, identifikasi kualitas suatu bahan (kayu, kulit, tekstil dan lain-lain). Pada analisis citra, dikategorikan menjadi lima kategori utama yaitu statistis, struktural, geometri, model dasar dan pengolahan sinyal. Pendekatan statistis mempertimbangakan bahwa bahwa internsitas dibangkitkan oleh medan acak dua dimensi, metode ini berdasar pada prekuensi-frekuensi ruang. Contoh metode statistis adalah fungsi autokorelasi, matriks kookurensi, tranformasi fourier, frekuensi tepi, run lenght. Fteknik struktural berkaitan dengan penyusupan bagian-bagian terkecil suatu citra. contoh metode struktural adalah model fraktal. Metode geometri berdasar atas perangkat geometri yang ada pada elemen tekstur. Contoh model dasar adalah medan acak. Sedangkan metode pengolahan sinyal adalah metode yang berdasarkan analisis frekuensi seperti transformasi gabor dan transformasi wavelet [10].

2.7 Ekstrasi Ciri Statistik

Ekstraksi ciri dilakukan untuk mendapatkan pola dari suatu citra yang akan dilatih maupun citra yang akan diuji. Metode ekstraksi ciri yang digunakan adalah ekstraksi ciri statistik. Metode ini menggunakan perhitungan statistik distribusi derajat keabuan (histogram) dengan mengukur tingkat kekontrasan, granularitas, dan kekasaran suatu daerah dari hubungan ketetanggaan antar piksel di dalam citra. Ekstraksi ciri statistik terbagi menjadi dua yaitu ekstraksi ciri statistik orde pertama dan orde kedua.



Gambar 1.6 Ilustrasi ekstraksi ciri statistik

Kiri : Histogram citra sebagai fungsi probabilitas kemunculan nilai intensitas pada citra
Kanan : Hubungan ketetanggaan antar piksel sebagai fungsi orientasi dan jarak spasial

2.7.1 Ekstraksi ciri orde pertama

Ekstraksi ciri orde pertama merupakan metode pengambilan ciri yang didasarkan pada karakteristik histogram citra. Histogram menunjukkan probabilitas kemunculan nilai derajat keabuan piksel pada suatu citra. Dari nilai-nilai pada histogram yang dihasilkan, dapat dihitung beberapa parameter ciri orde pertama, antara lain adalah *mean*, *variance*, *skewness*, *kurtosis*, dan *entropy* .

a. *Mean* (μ)

Menunjukkan ukuran dispersi dari suatu citra.

$$\mu = \sum_{n} fn\, P(fn) \qquad (2.3)$$

Dimana *fn* merupakan suatu nilai intensitas keabuan, sementara *p(fn)* menunjukkan nilai histogramnya (probabilitas kemunculan intensitas tersebut pada citra).

b. *Variance* (σ^2)

Menunjukkan variasi elemen pada histogram dari suatu citra.

$$\sigma^2 = \sum_n (fn - \mu)^2 P(fn) \quad (2.4)$$

c. *Skewness* (α_3)

Menunjukkan tingkat kemencengan relatif kurva histogram dari suatu citra.

$$\alpha_3 = \frac{1}{\sigma^3} \sum_n (fn - \mu)^3 P(fn) \quad (2.5)$$

d. *Kurtosis* (α_4)

Menunjukkan tingkat keruncingan relatif kurva histogram dari suatu citra.

$$\alpha_4 = \frac{1}{\sigma^4} \sum_n (fn - \mu)^4 P(fn) - 3 \quad (2.6)$$

e. *Entropy* (H)

Menunjukkan ukuran ketidakaturan bentuk dari suatu citra.

$$H = -\sum_n P(fn).^2 \log P(fn) \quad (2.7)$$

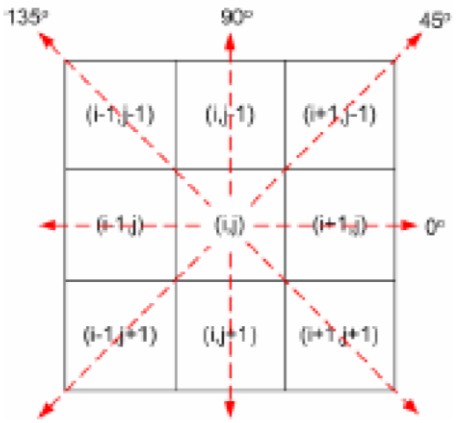
2.7.2 Ekstraksi ciri orde kedua

Pada beberapa kasus, ciri orde pertama tidak lagi dapat digunakan untuk mengenali perbedaan antar citra. Pada kasus seperti ini, kita membutuhkan pengambilan ciri statistik orde dua. Salah satu teknik untuk memperoleh ciri statistik orde dua adalah dengan menghitung probabilitas hubungan ketetanggaan antara dua piksel pada jarak dan orientasi sudut tertentu. Pendekatan ini bekerja dengan membentuk sebuah matriks kookurensi dari data citra, dilanjutkan dengan menentukan ciri sebagai fungsi dari matriks antara tersebut [11].

2.7.2.1 Metode Cooccurence Matrix (Gray Level Cooccurence)

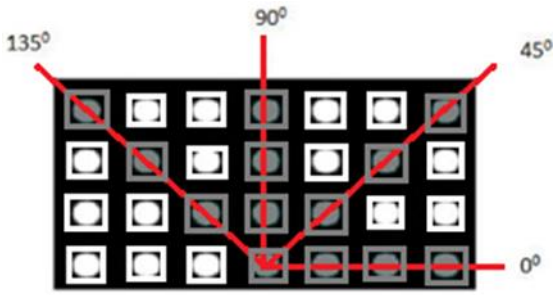
Metode *Cooccurence* Matrix adalah suatu matriks yang elemen-elemennya merupakan jumlah pasangan piksel yang memiliki tingkat kecerahan tertentu, di mana pasangan piksel itu terpisah dengan jarak d, dan dengan suatu sudut inklinasi θ. Dengan kata lain, matriks kookurensi adalah probabilitas munculnya graylevel i dan j dari dua piksel yang terpisah pada jarak d dan sudut θ [12].

Suatu piksel yang bertetangga yang memiliki jarak d diantara keduanya, dapat terletak di delapan arah yang berlainan, hal ini ditunjukkan pada Gambar(2.7).

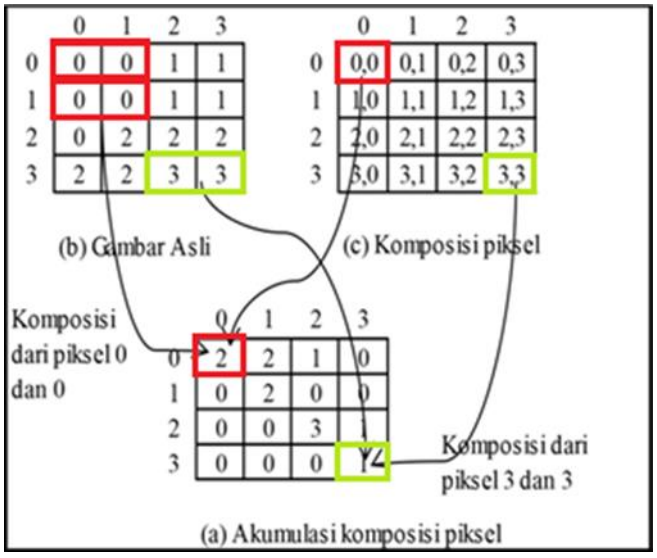


Gambar 1.7 Hubungan ketetanggaan antar piksel sebagai fungsi orientasi dan jarak spasial

Arah piksel tetangga untuk mewakili jarak dapat dipilih, misalnya 0^0, 45^0 , 90^0 , 135^0, atau seperti yang diilustrasikan pada Gambar(2.8). Sedangkan Gambar(2.9). menggambarkan bagaimana untuk menghasilkan matriks menggunakan arah 0o dan dengan jarak 1 piksel [12].



Gambar 1.8 Arah dalam menghitung GLCM



Gambar 1.9 Langkah pertama mengubah GLCM

Dengan menambahkan transposnya, matriks simetrik akan diperoleh, seperti ditunjukkan pada Gambar (2.10), tapi hasilnya masih belum ternormalisasi. Oleh karena itu, proses normalisasi harus dilakukan untuk menghapus ketergantungan pada ukuran citra dengan mengatur semua elemen dalam matriks sehingga total dari semua nilai elemen sama dengan 1. Gambar (2.11). merupakan hasil dari matriks yang telah ternormalisasi.

$$\begin{bmatrix} 2 & 2 & 1 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

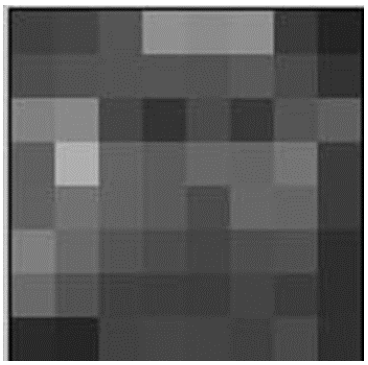
Gambar 1.10 Prosedure membuat matriks simetrik

Gambar (2.11) menjelaskan perubahan urutan matriks dari baris ke kolom lalu dijumlahkan dan akan menghasilkan matriks GLCM sebelum normalisasi.

$$\begin{bmatrix} \frac{4}{24} & \frac{2}{24} & \frac{1}{24} & \frac{0}{24} \\ \frac{2}{24} & \frac{4}{24} & \frac{0}{24} & \frac{0}{24} \\ \frac{1}{24} & \frac{0}{24} & \frac{6}{24} & \frac{1}{24} \\ \frac{0}{24} & \frac{0}{24} & \frac{1}{24} & \frac{2}{24} \end{bmatrix}$$

Gambar 1.11 Matriks ternormalisasi

Sebagai contoh Gambar(2.12) citra keabuan dengan ukuran 8 piksel yang memiliki derajat keabuan dengan rentang dari 0 sampai 7 atau (0, 7), dalam hal ini 0 menyatakan hitam dan 7 menyatakan putih, nilai antara 0 sampai 7 menyatakan warna keabuan yang terletak antara hitan dan putih.



Gambar 1.12 Citra keabuan dengan ukuran 8 piksel

Tabel 1.1 Array ukuran 8x8

1	1	2	5	5	6	2	0
3	3	3	3	3	4	2	1
4	5	3	2	4	3	5	4
3	7	4	3	4	3	6	2
3	4	4	3	2	5	5	2
5	4	3	3	2	4	5	0
4	3	2	2	2	2	3	1
1	1	3	3	3	2	3	1

Dari Tabel(2.1) tersebut diatas dapat dihitung probabilitas hubungan ketetanggaan antara dua piksel pada jarak dan orientasi sudut tertentu. Kemudian dihitung dengan jarak spasial 1 dan sudut 90^0 dan akan diperoleh matriks kookurensi yang dapat dihitung ciri statistik yang merepresentasikan citra yang diamati.

Karena matriks dari table (2.1) tersebut memiliki delapan aras keabuan, maka jumlah nilai piksel ketetanggaan dan nilai piksel referensi pada area kerja matriks berjumlah delapan, seperti tampak pada Tabel(2.2) berikut.

Tabel 1.2 Area kerja matriks

	0	1	2	3	4	5	6	7
0	0,0	0,1	0,2	0,3	0,4	0,5	0,5	0,7
1	1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7
2	2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7
3	3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7
4	4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7
5	5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7
6	6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7
7	7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7

Langkah selanjutnya yaitu menghitung nilai matriks dengan mengisikan jumlah hubungan spasial sehingga akan menghasilkan nilai matriks seperti pada Tabel (2.3).

Tabel 1.3 Pembentukan matriks kookurensi

GL	0	1	2	3	4	5	6	7
0	0	0	1	0	0	0	0	0
1	2	1	0	1	1	0	0	0
2	0	0	5	2	3	0	0	0
3	0	2	5	6	3	3	0	0
4	0	1	0	3	3	2	1	1
5	0	0	1	3	0	1	1	0
6	0	0	0	0	0	1	0	0
7	0	0	0	0	0	1	0	0

Proses yang ditunjukkan mulai Tabel 2.1, Tabel 2.2 dan Tabel 2.3 merupakan langkah pertama mengubah GLCM. Langkah selanjutnya nilai dari hasil pertama GLCM dicari nilai transposnya. Hasil dari nilai transpose dijumlahkan dengan nilai hasil pertama GLCM dan akan menghasilkan nilai matriks yang belum ternormalisasi seperti berikut.

Tabel 1.4 Matrix Transpose + Matriks pertama GLCM = Matrik simetris Yang belum ternormalisasi

GL	0	1	2	3	4	5	6	7

GL	0	1	2	3	4	5	6	7

0	0	2	0	0	0	0	0	0
1	0	1	0	2	1	0	0	0
2	1	0	5	5	0	1	0	0
3	0	1	2	6	3	3	0	0
4	0	1	3	3	3	0	0	0
5	0	0	0	3	2	1	1	1
6	0	0	0	0	1	1	0	0
7	0	0	0	0	1	0	0	0

+

0	0	0	1	0	0	0	0	0
1	2	1	0	1	1	0	0	0
2	0	0	5	2	3	0	0	0
3	0	2	5	6	3	3	0	0
4	0	1	0	3	3	2	1	1
5	0	0	1	3	0	1	1	0
6	0	0	0	0	0	1	0	0
7	0	0	0	0	0	1	0	0

=

GL	0	1	2	3	4	5	6	7
0	0	2	1	0	0	0	0	0
1	2	2	0	3	2	0	0	0
2	1	0	10	7	3	1	0	0
3	0	3	7	12	6	6	0	0
4	0	2	3	6	6	2	1	1
5	0	0	1	6	2	2	2	1
6	0	0	0	0	1	2	0	0
7	0	0	0	0	1	1	0	0

Matriks yang telah simetris di atas selanjutnya harus dinormalisasi elemen-elemennya yang dinyatakan dengan probabilitas. Nilai elemen untuk masing-masing sel dibagi dengan jumlah seluruh elemen spasial seperti gambar(2.13) berikut.

$$\begin{bmatrix} \frac{0}{108} & \frac{2}{108} & \frac{1}{108} & \frac{0}{108} & \frac{0}{108} & \frac{0}{108} & \frac{0}{108} & \frac{0}{108} \\ \frac{2}{108} & \frac{2}{108} & \frac{0}{108} & \frac{3}{108} & \frac{2}{108} & \frac{0}{108} & \frac{0}{108} & \frac{0}{108} \\ \frac{1}{108} & \frac{0}{108} & \frac{10}{108} & \frac{7}{108} & \frac{3}{108} & \frac{1}{108} & \frac{0}{108} & \frac{0}{108} \\ \frac{0}{108} & \frac{3}{108} & \frac{7}{108} & \frac{12}{108} & \frac{6}{108} & \frac{6}{108} & \frac{0}{108} & \frac{0}{108} \\ \frac{0}{108} & \frac{2}{108} & \frac{3}{108} & \frac{6}{108} & \frac{6}{108} & \frac{2}{108} & \frac{1}{108} & \frac{1}{108} \\ \frac{0}{108} & \frac{0}{108} & \frac{1}{108} & \frac{6}{108} & \frac{2}{108} & \frac{2}{108} & \frac{2}{108} & \frac{1}{108} \\ \frac{0}{108} & \frac{0}{108} & \frac{0}{108} & \frac{0}{108} & \frac{1}{108} & \frac{2}{108} & \frac{0}{108} & \frac{0}{108} \\ \frac{0}{108} & \frac{0}{108} & \frac{0}{108} & \frac{0}{108} & \frac{1}{108} & \frac{1}{108} & \frac{0}{108} & \frac{0}{108} \end{bmatrix}$$

Gambar 1.13 Nilai didalam Matriks Simetris dibagi jumlah nilai matriks simetris

Setelah memperoleh matriks kookurensi tersebut, maka langkah selanjutnya adalah menghitung ciri statistik orde kedua . Beberapa parameter statistik orde kedua antara lain :

a. *Angular Second Moment*

Menunjukkan ukuran sifat homogenitas citra.

$$ASM = \sum_i \sum_j \{P(i,j)\}^2 \quad (2.8)$$

Dimana *p*(*i*,*j*) merupakan menyatakan nilai pada baris i dan kolom j pada matriks kookurensi.

b. *Contrast*

Menunjukkan ukuran penyebaran (momen inersia) elemen-elemen matriks citra. Jika letaknya jauh dari diagonal utama, nilai kekontrasan besar. Secara visual, nilai kekontrasan adalah ukuran variasi antar derajat keabuan suatu daerah citra.

$$CON = \sum_i \sum_j (i-j)^2 \; P(i,j) \quad (2.9)$$

c. Correlation

Menunjukkan ukuran ketergantungan linear derajat keabuan citra sehingga dapat memberikan petunjuk adanya struktur linear dalam citra.

$$COR = \sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma i \; \sigma j} \quad (2.10)$$

Dimana :

μ_i = nilai rata – rata elemen baris pada matriks P(i,j) dinyatakan dengan rumus $\mu_i = \sum_i \sum_j i \, p(i,j)$

μ_j = nilai rata – rata elemen kolom pada matriks P(i,j) dinyatakan dengan rumus $\mu_j = \sum_i \sum_j j \, p(i,j)$

σ_i = nilai standar deviasi elemen baris pada matriks P(i,j) dinyatakan dengan rumus $\sigma_i = \sqrt{\sum_{i,j}(i-\mu i)^2 p(i,j)}$

σ_j = nilai standar deviasi elemen kolom pada matriks P(i,j) dinyatakan dengan rumus $\sigma_j = \sqrt{\sum_{i,j}(i-\mu j)^2 p(i,j)}$

d. *Variance*

Menunjukkan variasi elemen-elemen matriks kookurensi. Citra dengan transisi derajat keabuan kecil akan memiliki variansi yang kecil pula.

$$VAR = \sum_i \sum_j (i - \mu_i)(j - \mu_j)P(i,j) \quad (2.11)$$

e. *Inverse Different Moment*

Menunjukkan kehomogenan citra yang berderajat keabuan sejenis. Citra homogen akan memiliki harga IDM yang besar.

$$IDM = \sum_i \sum_j \frac{1}{1 + (i-j)^2} P(i,j) \quad (2.12)$$

f. *Entropy*

Menunjukkan ukuran ketidakteraturan bentuk. Harga ENT besar untuk citra dengan transisi derajat keabuan merata dan bernilai kecil jika struktur citra tidak teratur (bervariasi).

$$ENT_2 = -\sum_i \sum_j P(i,j) \,.\, log \;\; P(i,j) \quad (2.13)$$

2.7.3 Klasifikasi K-Nearest Neighbor

Algoritma K-NN merupakan algoritma yang melakukan klasifikasi berdasarkan kedekatan lokasi (jarak) suatu data dengan data yang lain.

Pada algoritma K-NN, data yang berdimensi q, jarak dari data tersebut ke data yang lain dapat dihitung. Nilai jarak ini lah yang digunakan sebagai nilai kedekatan/kemiripan antara data uji dengan data latih. Nilai K pada K-NN berarti K-data terdekat dari data uji [13].

Jarak *Euclidian* paling sering digunakan menghitung jarak. Jarak *euclidean* berfungsi menguji ukuran yang bisa dgunakan sebagai interpretasi kedekatan jarak antara dua obyek, yang direpresentasikan sebagai berikut:

$$d(x,y) \;\; = \sqrt{\sum_{k=1}^{n} (x_k - \; y_k)^2}$$

Keterangan:

d = jarak data uji ke data pembelajaran.

x_j = data uji ke-j, dengan j = 1, 2, . . . n.

y_j = data pembelejaran ke-j dengan j = 1, 2, . . . n.

2.8 Jenis – jenis Batu Akik

Berikut ini 10 jenis batu akik yang ada di Indonesia :

1. Batu Badar Besi

Batu badar besi terkenal karena kemampuan dia untuk terikat magnet dan ataupun terkena magnet akibat kandungan besi dan oksigen yang ada di dalam batu tersebut.

Ciri – ciri :

1. Bentuk : Bulat setengah telur.
2. Warna : hitam kecoklatan tidak tembus cahaya.
3. Kandungan : besi dan oksigen.
4. Indeks Bias : 2,94-3,22.
5. Berat Jenis : 5,5-65.
6. Skala Kekerasan (Mohs) : 5 hingga 6.

2. Batu Giok Hijau

Ciri – ciri :

1. Bentuk : seperti batuan umumnya .
2. Warna : hijau
3. Kandungan : jadeite dan nephrite dengan komposisi kimia NaAISi2O6 dan CA2(Mg,Fe)5Si8O22(OH)2
4. Indeks Bias : 1,666-1,680 (+/-0,008) dan 1,606-1,632 (0,009, 0,006)
5. Pembiasan Ganda : -
6. Berat Jenis : 3.34 dan 2.95
7. Skala Kekerasan (Mohs) : 6,5-7 dan 6,0-6,5

3. Batu Giok Hitam

Ciri – ciri :

1. Bentuk : seperti batuan umumnya .
2. Warna : hitam
3. Kandungan : jadeite dan nephrite dengan komposisi kimia NaAISi2O6 dan CA2(Mg,Fe)5Si8O22(OH)2

4. Indeks Bias : 1,666-1,680 (+/-0,008) dan 1,606-1,632 (0,009, 0,006)
5. Pembiasan Ganda : -
6. Berat Jenis : 3.34
7. Skala Kekerasan (Mohs) : 7

4. Batu Giok Merah

Ciri – ciri :

1. Bentuk : seperti batuan umumnya .
2. Warna : merah
3. Kandungan : jadeite dan nephrite dengan komposisi kimia NaAISi2O6 dan CA2(Mg,Fe)5Si8O22(OH)2
4. Indeks Bias : 1,666-1,680 (+/-0,008) dan 1,606-1,632 (0,009, 0,006)
5. Pembiasan Ganda : -
6. Berat Jenis : 2,45
7. Skala Kekerasan (Mohs) : 6,7

5. Batu Giok Putih

Ciri – ciri :

1. Bentuk : seperti batuan umumnya .
2. Warna : putih
3. Kandungan : jadeite dan nephrite dengan komposisi kimia NaAISi2O6 dan CA2(Mg,Fe)5Si8O22(OH)2
4. Indeks Bias : 1,666-1,680 (+/-0,008) dan 1,606-1,632 (0,009, 0,006)
5. Pembiasan Ganda : -
6. Berat Jenis : 3,67
7. Skala Kekerasan (Mohs) : 6,7

6. Batu Merah Siam

Batu merah siam identik dengan batu akik berdaya magis tinggi yang berasal dari alam ghaib.

Ciri – ciri :

1. Bentuk : bulat seperti telur ayam .
2. Warna : merah tembus cahaya
3. Kandungan : terdiri dari mineral korodum

4. Indeks Bias : 1,766 -1,774
5. Pembiasan Ganda : -
6. Berat Jenis : 3,97 sampai 4
7. Skala Kekerasan (Mohs) : 9

7. Batu Pancawarna

Ciri – ciri :

1. Bentuk : seperti batuan umumnya .
2. Warna : merah, abu, kuning, biru dan coklat.
3. Kandungan : terdiri dari mangan , besi , nikel
4. Indeks Bias : 1,54
5. Pembiasan Ganda : -
6. Berat Jenis : 2,88-2,89
7. Skala Kekerasan (Mohs) : 7 sampai 7,5.

8. Batu Pecah Seribu

Batu pecah seribu adalah batu mulia yg berasal dari birma dengan warna merah muda yg melambangkan cinta kasih sayang, selain itu juga terdapat warna-warna lain. Dengan motif retak2 didalam nya menambah keindahan batu mulia tersebut. Batu pecah seribu ini sangat langka walaupun ada mungkin harganya sangat fantastis.

Ciri – ciri :

1. Bentuk : bening seperti kristal.
2. Warna : merah serta hijau.
3. Kandungan : -
4. Indeks Bias : 1.54-1.55.
5. Berat Jenis : 2.72.
6. Skala Kekerasan (Mohs) : 6 sampai 7.

9. Batu Pirus

Ciri – ciri :

1. Bentuk : seperti batu akik yang permukaannya halus.
2. Warna : warna dari biru sampai hijau.
3. Kandungan : CuAl(PO4)2(OH)8.4H20

4. Indeks Bias : 1.610 sampai 1.650.
5. Berat Jenis : berkisar 2,76
6. Skala Kekerasan (Mohs) : 5 sampai 6.

10. Batu Sulaiman

Ciri – ciri :

1. Bentuk : seperti batuan umumnya.
2. Warna : di dominasi dengan warna kuning.
3. Kandungan : CuAl(PO4)2(OH)8.4H20
4. Indeks Bias : 1.544 sampai 1.553.
5. Berat Jenis : berkisar 2,64-2,66
6. Skala Kekerasan (Mohs) : 7.

2.9 Metode Perancangan Sistem

Metode perancangan yang digunakan adalah metode perancangan terstruktur. Metode ini diperkenalkan pada tahun 1970, yang merupakan hasil turunan dari pemrograman terstruktur. Pemrograman Terstruktur adalah suatu proses untuk mengimplementasikan urutan langkah untuk menyelesaikan suatu masalah dalam bentuk program [14] . Metode pengembangan dengan metode terstruktur ini terus diperbaiki sampai akhirnya dapat digunakan dalam dunia nyata.

Perancangan ini bertujuan untuk membuat model solusi terhadap problem yang sudah dimodelkan secara lengkap pada tahap analisis terstruktur. Prinsip dari pemrograman terstruktur adalah Jika suatu proses telah sampai pada suatu titik/langkah tertentu, maka proses selanjutnya tidak boleh mengeksekusi langkah sebelumnya/kembali lagi ke baris sebelumnya, kecuali pada langkah-langkah untuk proses berulang (*Loop*).

2.9.1 DFD (*Data Flow Diagram*)

DFD atau Diagram Arus Data adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, dimana data dihapus, proses apa yang menghasilkan data tersebut dan interaksi ‘antara data yang tersimpan dan proses yang dikarenakan pada data tersebut.

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau dimana data tersebut disimpan.

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur .

Kelebihan utama pendekatan aliran data, yaitu :

1. Kebebasan dari menjalankan implementasi teknis sistem.
2. Pemahaman lebih jauh mengenal keterkaitan satu sama lain dalam sistem dan subsistem.
3. Mengkomunikasikan pengetahuan sistem yang ada dengan pengguna melalui diagram aliran data.
4. Menganalisis sistem yang diajukan untuk menentukan apakah data-data dan proses yang diperlukan sudah ditetapkan.

Dibawah ini terdapat istilah-istilah untuk DFD yaitu:

1. *Entity*: *Terminator* atau *Source* atau *Destination* atau dikenal juga dengan *External Entity*, berupa orang, organisasi atau sistem lain yang berada diluar batas sistem yang berinteraksi dengan sistem yang sedang dikembangkan.
2. *Proses*: Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, prosedur atau alat yang digunakan untuk mentrasformasikan data.
3. *Data Flow*: (Arus Data), data yang mengalir dengan arah tertentu dari asal ke tujuan. Data yang mengalir dapat berupa dokumen, surat atau bentuk lainnya.
4. Data *Store*: (Penyimpanan Data), digunakan untuk menyimpan dan mengambil data oleh proses. Data yang disimpan dapat berupa data yang terkomputerisasi maupun tidak terkomputerisasi.

2.9.2 Komponen Terminator/Entitas Luar

Terminator mewakili entitas eksternal yang berkomunikasi dengan sistem yang sedang dikembangkan. Biasanya terminator dikenal dengan nama entitas luar (*external entity*). Sesuatu yang berada diluar system, tapi memberikan data

kedalam sistem atau memberikan data dari sistem. Disimbolkan dengan notasi kotak [15].

2.9.3 Komponen Proses

Komponen proses merupakan apa yang dikerjakan oleh system, proses dapat mengolah data atau aliran data masuk menjadi aliran data keluar. Proses berfungsi mentransformasikan satu atau beberapa data masukan menjadi satu atau beberapa data keluaran sesuai dengan spesifikasi data yang dinginkan [15].

2.9.4 Komponen Data Store

Komponen ini digunakan untuk membuat model sekumpulan paket data dan diberi nama dengan kata benda jamak, misalnya Mahasiswa. Data store ini biasanya berkaitan dengan penyimpanan-penyimpanan, seperti file atau database yang berkaitan dengan penyimpanan secara komputerisasi, misalnya file disket, file harddisk, file pita magnetik. Data store juga berkaitan dengan penyimpanan secara manual seperti buku alamat, file, folder, dan agenda.

Suatu Data Store dihubungkan dengan alur data hanya pada komponen proses, tidak dengan komponen DFD lainnya. Alur data yang menghubungkan Data Store dengan suatu proses mempunyai pengertian sebagai berikut :

1. Alur data dari Data Store yang berarti sebagai pembacaan atau pengaksesan satu paket tunggal data, lebih dari satu paket data, sebagian dari satu paket tunggal data, atau sebagian dari lebih dari satu paket data untuk suatu proses
2. Alur data ke Data Store yang berarti sebagai peng-update-an data, seperti menambah satu paket data baru atau lebih, menghapus satu paket atau lebih, atau mengubah/memodifikasi satu paket data atau lebih.

2.9.5 Komponen Data Flow/Alur Data

Arus data merupakan tempat mengalirnya informasi dan digambarkan dengan garis yang menghubungkan komponen dari sistem [15]. Arus data ditunjukan dengan arus panah dan garis diberi nama atas arus data yang mengalir. Arus data ini mengalir diantara proses, data store dan menunjukan arus data dari data yang berupa masukan untuk sistem atau hasil proses sistem.

2.10 Perangkat Lunak Pendukung

Adapun perangkat lunak pendukung yang digunakan dalam implementasi metode di aplikasi pengklasifikasian jenis batu akik dapat dilihat sebagai berikut :

2.10.1 Matlab

Matlab dikembangkan sebagai bahasa pemograman sekaligus alat visualisasi, yang menawarkan banyak kemampuan untuk menyelesaikan berbagai kasus yang berhubungan langsung dengan disiplin ilmu matematika. Matlab dibangun dari bahasa induknya yaitu bahasa C, namun tidak dapat dikatakan sebagai varian dari bahasa C, karena dalam sintak maupun cara kerjanya sama sekali berbeda dari bahasa C. Namun dengan hubungan langsungnya dengan bahasa C, Matlab memiliki kelebihan-kelebihan dari bahasa C bahkan mampu berjalan pada semua *platform* Sistem Operasi tanpa mengalami perubahan sintak sama sekali. Bahasa ini mengintegrasikan kemampuan komputasi, visualisasi dan pemograman dalam sebuah lingkungan yang tunggal dan mudah digunakan. Matlab memberikan sistem interaktif yang menggunakan konsep array/matrik sebagai standar variabel elemennya tanpa membutuhkan pendeklarasian array seperti pada bahasa lainnya.

Matlab dikembangkan oleh MathWorks, yang pada awalnya dibuat untuk memberikan kemudahan mengakses data matrik pada proyek LINPACK dan EISPACK. Selanjutnya menjadi sebuah aplikasi untuk komputasi matrik. Dari sejak awal dipergunakan, matlab memperoleh masukan ribuan pemakai. Dalam lingkungan pendidikan ilmiah menjadi alat pemrograman standar bidang matematika, rekayasa dan keilmuan terkait. Dan dalam lingkungan industri dapat menjadi pilihan paling produktif untuk riset, pengembangan dan analisa [16].

2.11 Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan suatu investigasi yang dilakukan untuk mendapatkan informasi mengenai kualitas dari produk atau layanan yang sedang diuji. Pengujian perangkat lunak juga memberikan pandangan mengenai perangkat lunak secara obyektif dan independen, yang bermanfaat dalam operasional bisnis untuk memahami tingkat risiko pada implementasinya. Teknik-teknik pengujian mencakup, namun tidak terbatas pada, proses mengeksekusi

suatu bagian program atau keseluruhan aplikasi dengan tujuan untuk menemukan "bug" perangkat lunak. Bug merupakan suatu kesalahan desain pada suatu perangkat keras komputer atau perangkat lunak komputer yang menyebabkan peralatan atau program itu tidak berfungsi semestinya. Bug umumnya lebih umum dalam dunia perangkat lunak dibandingkan dengan perangkat keras [17].

Pengujian perangkat lunak merupakan suatu tahapan penting dalam pembangunan perangkat lunak. Pengujian dilakukan dengan cara mengevaluasi konfigurasi perangkat lunak yang terdiri dari spesifikasi kebutuhan, deskripsi perancangan, dan program yang dihasilkan. Hasil evaluasi kemudian dibandingkan dengan hasil uji yang diharapkan. Jika ditemukan kesalahan, maka perbaikan perangkat lunak harus dilakukan untuk kemudian diuji kembali.

Pengujian perangkat lunak adalah proses menjalankan dan mengevaluasi sebuah perangkat lunak secara manual maupun otomatis untuk menguji apakah perangkat lunak sudah memenuhi persyaratan atau belum, atau untuk menentukan perbedaan antara hasil yang diharapkan dengan hasil sebenarnya. Peksanaan pengujian perangkat lunak biasanya disesuaikan dengan metodologi pembangunan perangkat lunak yang digunakan.

Pengujian dikatakan berhasil apabila pengujian yang dilakukan memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.

2.11.1 Teknik Pengujian

Teknik pengujian yang dapat digunakan dalam menguji perangkat lunak sebagai berikut :

a. Pengujian *White Box* : Pengujian *white box* adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Secara sekilas dapat diambil kesimpulan *white box testing* merupakan petunjuk untuk mendapatkan program yang benar secara 100%.

   Penggunaan metode pengujian *white box* dilakukan untuk :

1) Memberikan jaminan bahwa semua jalur independen suatu modul digunakan minimal satu kali
2) Menggunakan semua keputusan logis untuk semua kondisi true atau false
3) Mengeksekusi semua perulangan pada batasan nilai dan operasional pada setiap kondisi.
4) Menggunakan struktur data internal untuk menjamin validitas jalur keputusan.

Persyaratan dalam menjalankan strategi *White Box Testing* sebagai berikut:

1) Mendefinisikan semua alur logika
2) Membangun kasus untuk digunakan dalam pengujian
3) Mengevaluasi semua hasil pengujian
4) Melakukan pengujian secara menyeluruh

b. Pengujian *Black Box* : Pengujian *black box* digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Pada teknik ini, kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diiketahui kesalahan-kesalahannya.

   Beberapa jenis kesalahan yang dapat diidentifikasi :

   1) Fungsi tidak benar atau hilang
   2) Kesalahan antar muka
   3) Kesalahan pada struktur data (pengaksesan basis data)
   4) Kesalahan inisialisasi dan akhir program
   5) Kesalahan performasi

Walaupun sulit untuk menelusuri kesalahan yang mungkin didapat, teknik pengujian *black box* lebih sering dipilih untuk menguji perangkat lunak karena kemudahan dalam pelaksanaannya.

2.11.2 Pengujian Akurasi

Pengujian akurasi dilakukan dengan confusion matriks yaitu adalah sebuah tabel yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan [18]. Contoh confusion matrix untuk klasifikasi biner ditunjukkan pada tabel dibawah ini .

Tabel 1.5 Confusion Matriks

		Kelas Prediksi	
		1	0
Kelas Sebenarnya	1	TP	FN
	0	FP	TN

Keterangan untuk tabel 2.1dinyatakan sebagai berikut:

True Posstive (TP), yaitu jumlah dokumen dari kelas 1 yang benar dan diklasifikan sebagai kelas 1.

True Negative (TN), yaitu jumlah dokumen dari kelas 0 yang benar diklasifikasikan sebagai kelas 0.

False Positive (FP), yaitu jumlah dokumen dari kelas 0 yang salah diklasifikasikan sebagai kelas 1.

False Negative (FN) yaitu jumlah dokumen dari kelas 1yang salah diklasifikasikan sebagai kelas 0.

Perhitungan akurasi dinyatakan dengan rumus

$$\text{Akurasi} = \frac{TP+TN}{TP+FN+FP+TN} * 100\% \qquad (2.15)$$