

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Kurma**

Buah Kurma merupakan buah yang banyak ditemui di Timur Tengah yang memiliki kandungan nutrisi yang baik untuk tubuh manusia. Kurma mengandung makronutrien seperti karbohidrat, lemak, protein, dan serat. Karbohidrat pada kurma terdiri dari gula sederhana seperti fruktosa, glukosa, dan sukrosa. Gula sederhana ini merupakan sumber energi yang tinggi dan mudah diserap tubuh (Fandi et al., 2020).

#### **2.2 Jenis Kurma**

Pada penelitian ini menggunakan 3 jenis kurma yang akan diteliti untuk di klasifikasi berdasarkan tekstur menggunakan algoritma *Local Binary Pattern* dan *Support Vector Machine*, diantaranya:

##### **2.2.1 Kurma Ajwa**

Kurma Ajwa memiliki warna yang hitam pekat dan ada beberapa yang mendekati warna merah. Kurma Ajwa juga memiliki tekstur yang unik seperti guratan di seluruh permukaan kurma. Ukuran dari kurma Ajwa cenderung lebih kecil dibandingkan dengan jenis kurma yang lain (Fandi et al., 2020).

##### **2.2.2 Kurma Sukkari**

Kurma Sukkari memiliki warna coklat terang dan pada bagian ujungnya berwarna kuning ranum. Kurma Sukkari memiliki tekstur renyah pada permukaan kulit daging kurma. Ukuran dari kurma Sukkari relatif lebih besar dibandingkan dengan kurma Ajwa (Fandi et al., 2020).

##### **2.2.3 Kurma Deglet Noor (Tunisia)**

Kurma Deglet Nour memiliki warna coklat pekat. Tesktur dari kurma ini cenderung lembek dibandingkan dengan kurma Ajwa dan Sukkari. Kurma ini

berbentuk lonjong dan ukurannya lebih besar dibandingkan kurma Ajwa dan Sukari (Fandi et al., 2020).

### 2.3 Citra

Citra yaitu salah satu komponen yang berperan penting sebagai suatu bentuk dari informasi visual (Furqan, Sriani, et al., 2020). Citra (image) adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (continue) dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat optik, misalnya mata pada manusia, kamera dan pemindai (scanner), sehingga bayangan objek yang disebut citra tersebut terekam (Izzati Saila Hafsa, 2015).

Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal – sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpan. Suatu citra analog tidak dapat secara langsung diproses oleh komputer. Untuk memproses citra analog menggunakan komputer maka sebelumnya citra analog harus diubah menjadi citra digital (Izzati Saila Hafsa, 2015).

### 2.4 Jenis Citra

Pada umumnya citra dibagi menjadi 2 diantaranya adalah:

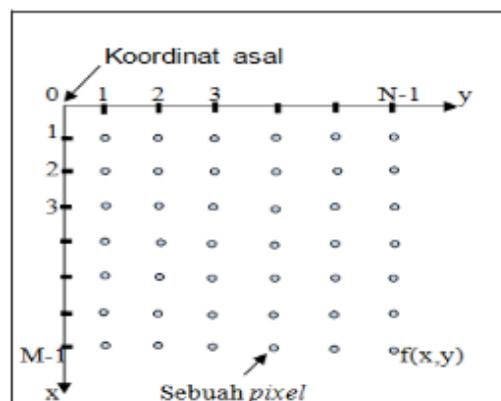
#### 2.4.1 Citra Analog

Citra analog adalah citra yang bersifat kontinu. Seperti gambar pada monitor televisi, foto sinar X, foto yang tercetak di kertas foto, lukisan, pemandangan alam, hasil CT scan, gambar-gambar yang terekam pada pita kaset, dan lain sebagainya. Citra analog tidak dapat direpresentasikan dalam komputer sehingga tidak dapat diproses di komputer secara langsung. Oleh sebab itu, agar citra dapat diproses di komputer, proses konversi analog ke digital harus dilakukan terlebih dahulu. Citra analog dihasilkan dari alat-alat analog, seperti video kamera analog, kamera foto

analog, CT scan, sensor rontgen untuk foto thorax, sensor gelombang pendek pada sistem radar, sensor ultrasound pada sistem USG dan lain-lain. Reduksi noise adalah suatu proses menghilangkan atau mengurangi noise dari suatu signal, tetapi untuk implementasinya, reduksi noise tergantung dari jenis signal yang akan diproses. Sinar-x adalah salah satu bentuk dari radiasi elektromagnetik dengan panjang gelombang berkisar antara 10 nanometer ke 100 pikometer (sama dengan frekuensi dalam rentang 30 petahertz – 30 exahertz) dan memiliki energi dalam rentang 100 eV – 100 KeV. Sinar-X umumnya digunakan dalam diagnosis gambar medis dan kristalografi sinar-X (Pratiwi & Waruwu, 2018).

#### 2.4.2 Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar 2 dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data 2 dimensi. Citra digital adalah sebuah larik (array) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan 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 atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y, dan nilai amplitudo f secara keseluruhan berhingga (finite) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut citra digital. Gambar 2.1 menunjukkan posisi koordinat citra digital (Nafi'iyah, 2015).



Gambar 2.1 Koordinat Citra Digital (Nafi'iyah, 2015)

Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu pita magnetik. Menurut presisi yang digunakan untuk menyatakan titik-titik koordinat pada domain spasial atau bidang dan untuk menyatakan nilai keabuan atau warna suatu citra, maka secara teoritis citra dapat dikelompokkan menjadi empat kelas citra, yaitu citra kontinu-kontinu, kontinu-diskrit, diskrit-kontinu, dan diskrit-diskrit. Di mana label pertama menyatakan presisi dari titik-titik koordinat pada bidang citra sedangkan label kedua menyatakan presisi nilai keabuan atau warna. Kontinu dinyatakan dengan presisi angka tidak berhingga, sedangkan diskrit dinyatakan dengan presisi angka berhingga (Nafi'iyah, 2015).

Komputer digital bekerja dengan angka-angka presisi terhingga, dengan demikian hanya citra dari kelas diskrit-diskrit yang dapat diolah dengan komputer; citra dari kelas tersebut lebih dikenal sebagai citra digital. Citra digital merupakan suatu array dua dimensi atau suatu matriks yang elemen-elemennya menyatakan tingkat keabuan dari elemen gambar, jadi informasi yang terkandung bersifat diskrit. Citra digital tidak selalu merupakan hasil langsung data rekaman suatu sistem. Kadang-kadang hasil rekaman data bersifat kontinu seperti gambar pada monitor televisi, foto sinar-X, dan lain sebagainya. Dengan demikian untuk mendapatkan suatu citra digital diperlukan suatu proses konversi, sehingga citra tersebut selanjutnya dapat diproses dengan komputer (Nafi'iyah, 2015).

Citra digital dapat ditulis dalam bentuk matriks sebagai berikut:

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

Nilai pada suatu irisan antara baris dan kolom (pada posisi  $x, y$ ) disebut dengan picture elements, image elements, pels, piksels. Istilah terakhir (piksels) paling sering digunakan pada citra digital (Nafi'iyah, 2015).

Citra digital dibentuk oleh kumpulan titik yang dinamakan piksel. Setiap piksel digambarkan sebagai satu kotak kecil. Setiap piksel mempunyai koordinat posisi. Sistem koordinat yang dipakai untuk menyatakan citra digital ditunjukkan di Gambar 2.2 (Nafi'iyah, 2015).



Gambar 2.2 Sistem Koordinat Citra Berukuran  $M \times N$  ( $M$  Baris,  $N$  Kolom)  
(Nafi'iyah, 2015)

Dengan sistem koordinat yang mengikuti asas pemindaian pada layar TV standar itu, sebuah piksel mempunyai koordinat berupa  $(x, y)$ . Dalam hal ini,  $x$  menyatakan posisi kolom, dan  $y$  menyatakan posisi baris. Piksel pojok kiri-atas mempunyai koordinat  $(0,0)$  dan piksel pada pojok kanan-bawah mempunyai koordinat  $(M-1,N-1)$  (Nafi'iyah, 2015).

## 2.5 Jenis Citra Digital

Pada umumnya citra digital dibagi menjadi 3 diantaranya adalah:

### 2.5.1 Citra Berwarna

Citra berwarna, atau biasa dinamakan citra RGB, merupakan jenis citra yang menyajikan warna dalam bentuk komponen R (merah), G (hijau), dan B (biru). Setiap komponen warna menggunakan delapan bit (nilainya berkisar antara 0 sampai dengan 255). Dengan demikian, kemungkinan warna yang dapat disajikan mencapai  $255 \times 255 \times 255$  atau 16.581.375 warna (Nafi'iyah, 2015).

### 2.5.2 Citra Berskala Keabuan

Sesuai dengan nama yang melekat, citra jenis ini menangani gradasi warna hitam dan putih, yang tentu saja menghasilkan efek warna abu-abu. Pada jenis gambar ini, warna dinyatakan dengan intensitas. Dalam hal ini, intensitas berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih. Istilah lain citra keabuan adalah citra grayscale yaitu citra yang nilai pikselnya 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 piksel 8 bit (256 derajat keabuan), tetapi ada juga citra grayscale yang kedalaman pikselnya bukan 8 bit, misalnya 16 bit untuk penggunaan yang memerlukan ketelitian tinggi (Nafi'iyah, 2015).

### 2.5.3 Citra Biner

Citra biner adalah citra dengan setiap piksel hanya dinyatakan dengan sebuah nilai dari dua kemungkinan (yaitu nilai 0 dan 1). Nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Citra jenis ini banyak dipakai dalam pemrosesan citra, misalnya untuk kepentingan memperoleh tepi bentuk suatu objek (Nafi'iyah, 2015).

## 2.6 Elemen Dasar Citra Digital

Citra digital memiliki beberapa elemen dasar antara lain sebagai berikut di bawah ini:

### 2.6.1 Kecerahan (*Brightness*)

Kecerahan (*brightness*), yaitu intensitas cahaya rata-rata dari suatu area yang melingkupinya (Ndruru & Tarigan, 2017).

### 2.6.2 Kontras (*Contrast*)

Kontras (*contrast*), yaitu sebaran terang dan gelap pada sebuah citra. Sebuah citra disebut kontras rendah jika komposisi citra tersebut sebagian besar terang atau

sebagian besar gelap. Citra dengan kontras yang baik jika sebaran bagian terang dan gelap merata (Ndruru & Tarigan, 2017).

### 2.6.3 Kontur (*Contur*)

Kontur (contour), yaitu keadaan yang ditimbulkan oleh perubahan intensitas piksel-piksel yang berdekatan, sehingga kita dapat mendeteksi tepi objek pada suatu citra (Ndruru & Tarigan, 2017).

### 2.6.4 Warna

Warna (colour), yaitu persepsi yang dirasakan oleh sistem visual manusia terhadap panjang gelombang cahaya yang dipantulkan oleh objek (Ndruru & Tarigan, 2017).

### 2.6.5 Bentuk (*Shape*)

Bentuk (shape), yaitu informasi yang ditangkap oleh sistem visual manusia mengenai citra suatu objek (Ndruru & Tarigan, 2017).

### 2.6.6 Tekstur (*Texture*)

Tekstur (texture), yaitu distribusi spasial dari derajat keabuan di dalam piksel-piksel yang bertetangga (Ndruru & Tarigan, 2017).

## 2.7 Pengolahan Citra

Pengolahan citra merupakan proses memanipulasi dan menganalisis citra dengan bantuan komputer. dengan menggunakan deteksi tepi yang menentukan titik-titik tepi dari obyek, data yang digunakan dalam deteksi tepi berupa citra digital, citra dari sudut pandang matematis merupakan fungsi menerus (continue) dari intensitas cahaya pada bidang dua dimensi  $f(x,y)$  dengan  $x$  dan  $y$  adalah koordinat spasial dan amplitudo  $f$  pada pasangan koordinat  $(x,y)$  yang disebut intensitas atau derajat keabuan citra pada titik tersebut. parameter yang digunakan adalah secara visual dan dari jumlah piksel warna putih pada citra keluaran (Putri, 2016).

## 2.8 Ekstraksi Ciri

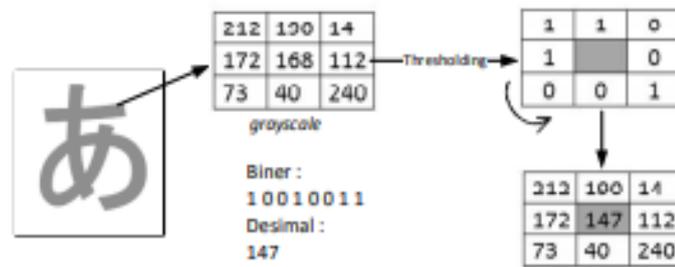
Ekstraksi ciri merupakan langkah awal dalam menentukan parameter sebagai interpretasi analisis tekstur citra. Proses ini berkaitan dengan kuantisasi karakteristik citra ke dalam sekelompok nilai ciri yang sesuai (Agmalaro et al., 2013). Ekstraksi ciri memiliki tiga tingkatan yaitu Low-level, middle-level dan high-level. Ekstraksi ciri berdasarkan isi visual seperti warna dan tekstur merupakan Low-level feature. Ekstraksi tiap objek dalam citra dan mencari hubungannya merupakan Middle-level feature. Sedangkan Ekstraksi ciri berdasarkan informasi semantik yang terkandung dalam citra merupakan high-level feature (No et al., 2016).

## 2.9 Local Binary Pattern (LBP)

*Local Binary Pattern* pertama kali diperkenalkan oleh Ojala et al, didefinisikan sebagai ukuran tekstur grayscale yang invarian, disebut invarian Karena hampir tidak dipengaruhi oleh pencahayaan yang berbeda. LBP ampuh untuk mendeskripsikan suatu tekstur, mempunyai daya pembeda yang akurat (Amynarto et al., 2018). LBP adalah metode analisis tekstur yang menggunakan model statistika dan struktur. Operator LBP menggunakan perbandingan nilai keabuan dari piksel-piksel ketetanggaan. Operator dasar LBP berukuran 3 x 3 menggunakan 8 piksel ketetanggaan in dari sebuah piksel tengah ic. Piksel ketetanggaan ke-n tersebut di-threshold menggunakan nilai keabuan dari piksel tengah seperti yang ditunjukkan pada persamaan (1) dan fungsi thresholding  $s(x)$  seperti yang ditunjukkan pada persamaan (2). Kode binary hasil operator LBP piksel ketetanggaan akan digunakan untuk merepresentasikan fitur dari piksel tengah ic (Amat et al., 2017).

$$LBP(x_c y_c) = \sum_{n=0}^7 s(i_n - i_c) 2^n \quad (1)$$

$$s(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (2)$$



Gambar 2.3 Ilustrasi LBP

Gambar 2.3 merupakan ilustrasi dari proses LBP. Proses pertama adalah melakukan pengurangan piksel ketetangaan dengan piksel tengah (1). Selanjutnya hasil pengurangan di-threshold menggunakan persamaan (2), jika hasilnya  $\geq 0$  maka diberi nilai 1 dan jika hasilnya  $< 0$  maka diberi nilai 0. Setelah itu, nilai biner piksel ketetangaan akan disusun berlawanan arah jarum jam dan 8 bit biner tersebut dikonversi ke dalam nilai desimal untuk menggantikan nilai piksel tengah ic. Menurut kelebihan dari LBP adalah mudah diimplementasikan dan tingkat komputasinya lebih rendah sehingga tidak membutuhkan waktu yang lama dalam ekstraksi fitur (Amat et al., 2017).

## 2.10 Histogram

Histogram merupakan grafik yang merepresentasikan penyebaran nilai-nilai intensitas piksel dari suatu citra. Dari sebuah histogram dapat diketahui frekuensi kemunculan relatif dari intensitas pada citra tersebut. Histogram dapat menunjukkan kecerahan (brightness) dan kontras (contrast) dari sebuah gambar. Histogram adalah alat bantu yang berharga dalam pekerjaan pengolahan citra baik secara kualitatif maupun kuantitatif. Misalkan citra digital memiliki  $L$  derajat keabuan, yaitu dari nilai 0 sampai  $L-1$  (misal pada citra dengan kuantisasi derajat keabuan 8-bit, nilai derajat keabuan dari 0 sampai 255) (Hidayatullah, 2017). Secara matematis histogram citra dihitung dengan rumus:

$$h_i = \frac{n_i}{n}, i = 0, 1, \dots, L - 1 \quad (3)$$

Ket:

$n_i$  = jumlah piksel yang memiliki derajat keabuan

$n$  = jumlah seluruh piksel didalam citra

## 2.11 Ekstraksi Ciri Statistik

Ekstraksi ciri 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, antara lain adalah mean, variance, skewness, kurtosis, dan entropy (Yuda Permadi & Murinto, 2015).

### 1. Mean ( $\mu$ )

Mean ( $\mu$ ) menunjukkan ukuran dispersi dari suatu citra, dihitung menggunakan persamaan dibawah ini:

$$\mu = \sum_{n=0}^N f_n p(f_n) \quad (4)$$

Ket:

$f_n$  = nilai intensitas keabuan

$p(f_n)$  = nilai histogram

### 2. Entropy (H)

Entropy (H) menunjukkan ukuran ketidakaturan bentuk dari suatu citra, dihitung menggunakan persamaan dibawah ini:

$$H = - \sum_{n=0}^N p(f_n) \cdot \log p(f_n) \quad (5)$$

Ket:

$p(f_n)$  = nilai histogram

### 3. Variance ( $\sigma^2$ )

Variance ( $\sigma^2$ ) menunjukkan variasi elemen pada histogram dari suatu citra, dihitung menggunakan persamaan dibawah ini:

$$\sigma^2 = \sum_{n=0}^N (f_n - \mu)^2 p(f_n) \quad (6)$$

Ket:

$f_n$  = nilai intensitas keabuan

$p(f_n)$  = nilai histogram

$\mu$  = nilai mean

#### 4. Skewness ( $\alpha_3$ )

Skewness ( $\alpha_3$ ) menunjukkan tingkat kemencengan relatif kurva histogram dari suatu citra, dihitung menggunakan persamaan dibawah ini:

$$\alpha_3 = \frac{1}{\sigma^3} \sum_{n=0}^N (f_n - \mu)^3 p(f_n) \quad (7)$$

Ket:

$f_n$  = nilai intensitas keabuan

$p(f_n)$  = nilai histogram

$\mu$  = nilai mean

$\sigma^3$  = standar deviasi dari nilai intensitas keabuan

#### 5. Kurtosis ( $\alpha_4$ )

Kurtosis ( $\alpha_4$ ) menunjukkan tingkat keruncingan relatif kurva histogram dari suatu citra, dihitung menggunakan persamaan dibawah ini:

$$\alpha_4 = \frac{1}{\sigma^4} \sum_{n=0}^N (f_n - \mu)^4 p(f_n) - 3 \quad (8)$$

Ket:

$f_n$  = nilai intensitas keabuan

$p(f_n)$  = nilai histogram

$\mu$  = nilai mean

$\sigma^4$  = standar deviasi dari nilai intensitas keabuan

## 2.12 Klasifikasi Citra

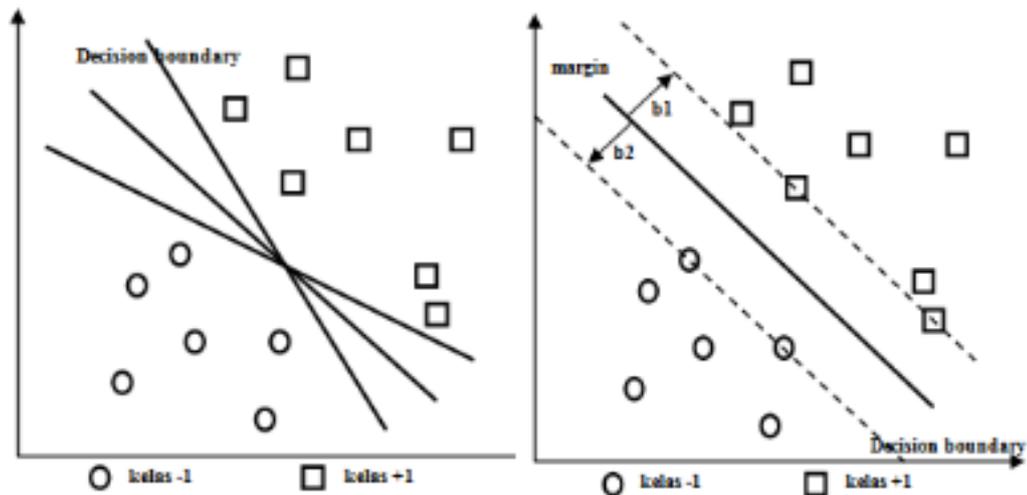
Klasifikasi merupakan salah satu task analisis data dimana dibentuk sebuah model (classifier) untuk menyatakan suatu objek ke salah satu kategori atau kelas yang sudah didefinisikan sebelumnya. Klasifikasi data merupakan proses yang terdiri atas dua langkah-langkah pertama adalah tahap pelatihan (learning), dimana sebuah algoritma klasifikasi akan membentuk classifier dengan menganalisis atau belajar dari sekumpulan data latih. Pada langkah kedua, model yang terbentuk digunakan untuk mengklasifikasikan data uji. Akurasi dari sebuah classifier untuk sekumpulan data uji yang diberikan merupakan persentase dari data-data uji yang diklasifikasikan dengan benar oleh classifier (Asmara et al., 2018).

## 2.13 Support Vector Machine (SVM)

SVM merupakan sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang ciri (feature space) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan bias pembelajaran yang berasal dari teori pembelajaran statistic. SVM merupakan metode klasifikasi yang kini banyak dikembangkan dan diterapkan. Metode ini berasal dari teori pembelajaran statistik yang menjanjikan dan memberikan hasil yang lebih baik dibandingkan dengan metode yang lainnya. SVM bekerja sangat baik pada himpunan data berdimensi tinggi. Ide dasar dari SVM adalah memaksimalkan batas hyperplane yang dijelaskan pada Gambar 2.4 (Neneng et al., 2016).

SVM memiliki keuntungan menentukan jarak antara dua set data dari dua kelas yang berbeda. SVM dapat menentukan hyperplane yang memisahkan secara optimal dan maksimum sebagai solusi dalam permasalahan dari pembelajaran mesin. Formulasi dari SVM yang termudah yaitu linier, di mana hyperplane terletak di tengah ruang data input. Masalah klasifikasi dapat diselesaikan dengan menentukan garis hyperplane yang memisahkan antara dua kelompok dengan menghitung margin antara kedua kelas, kemudian menghasilkan titik hyperplane yang paling optimal. Margin adalah jarak antara hyperplane dan anggota terdekat

dari setiap kelas. Hasil penentuan hyperplane adalah inti dari proses pembelajaran di SVM (Furqan, Kurniawan, et al., 2020).



Gambar 2.4 Batas keputusan yang mungkin untuk set data

Konsep klasifikasi dengan SVM adalah sebagai usaha mencari hyperplane terbaik yang berfungsi sebagai pemisah dua buah kelas data pada input space. Pada Gambar 2.4 memperlihatkan beberapa data yang merupakan anggota dari dua buah kelas data yakni +1 dan -1. Bentuk lingkaran merupakan simbol data yang tergabung dalam kelas -1, sedangkan bentuk bujur sangkar merupakan simbol data yang tergabung dalam kelas +1. Hyperplane pemisah terbaik antara kedua kelas tersebut dapat ditemukan dengan mengukur margin hyperplane dan mencari titik maksimalnya. Margin merupakan jarak antara hyperplane tersebut dengan data terdekat dari masing-masing kelas. Data yang paling dekat inilah yang disebut dengan support vector. Pada Gambar 2.4 di atas menunjukkan hyperplane terbaik karena terletak di tengah-tengah kedua kelas. Data lingkaran dan bujur sangkar yang dilewati garis putus-putus (garis batas margin) adalah support vector. Inti dari proses pelatihan pada SVM ini adalah usaha untuk mencari lokasi hyperplane (Neneng et al., 2016)

Secara matematika, formulasi problem optimasi SVM untuk kasus klasifikasi linear di dalam primal space adalah sebagai berikut:

$$\min \frac{1}{2} \|w\|^2$$

Syarat

$$y_i (wx_i + b) \geq 1, l = 1, \dots, l \quad (9)$$

Optimalisasi ini dapat diselesaikan dengan pengali Langrange:

$$L_p = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i y_i (w \cdot x_i + b) \quad (10)$$

$\alpha_i$  adalah Langrange Multiplier yang berkorespondensi dengan  $x_i$ . Nilai  $\alpha_i$  adalah nol atau positif. Untuk meminimalkan pengali Langrange, persamaan 10 harus diturunkan pada  $w$  dan  $b$ , dan diset dengan nilai nol untuk syarat optimalisasi di atas.

Syarat 1:

$$\frac{\partial L_p}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^n \alpha_i y_i x_i \quad (11)$$

Syarat 2:

$$\frac{\partial L_p}{\partial b} = 0 \Rightarrow \sum_{i=1}^n x_i y_i = 0 \quad (12)$$

$N$  adalah jumlah data yang menjadi support vector. Karena Langrange Multiplier ( $\alpha$ ) tidak diketahui nilainya, persamaan di atas tidak dapat diselesaikan secara langsung untuk mendapatkan  $w$  dan  $b$ . Untuk menyelesaikan masalah tersebut, persamaan 10 dimodifikasi menjadi kasus pemaksimalan, syarat optimal untuk dualitasnya menggunakan kendala (constraint) Karush Kuhn Tucker (KKT) berikut:

Syarat 1:

$$\alpha_i [y_i (w \cdot x_i + b) - 1] = 0 \quad (13)$$

Syarat 2:

$$\alpha_i \geq 0, i = 1, 2, \dots, N \quad (14)$$

Dengan menerapkan kendala pada persamaan 13 dan 14, dipastikan bahwa nilai Lagrange Multiplier sama banyaknya dengan data latih, tetapi sebenarnya banyak dari data latih yang Lagrange Multiplier nya sama dengan nol (karena hanya beberapa saja yang akan menjadi support vector) ketika menerapkan syarat pertama. Kendala di atas menyatakan bahwa Lagrange  $\alpha_i$  harus nol, kecuali untuk data latih  $x_i$  yang memenuhi persamaan

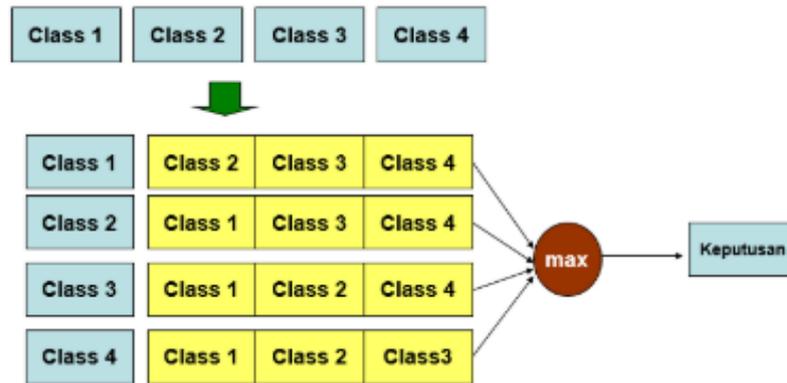
$$y_i(w \cdot x_i + b) = 1 \quad (15)$$

#### 2.14 Klasifikasi Multikelas *One-Versus-Rest*

SVM pertama kali diperkenalkan oleh Vapnik, hanya dapat mengklasifikasikan data ke dalam dua kelas. Namun penelitian lebih lanjut terus dilakukan untuk mengembangkan SVM sehingga bisa mengklasifikasi data yang memiliki jumlah kelas lebih dari dua atau klasifikasi multikelas. Pada metode klasifikasi multikelas *One-Versus-Rest*, terdapat  $k$  model klasifikasi SVM, dengan  $k$  adalah banyak kelas. SVM melakukan klasifikasi dua kelas antara satu kelas dengan kelas-kelas lain yang dianggap sebagai satu kelas. Contohnya, terdapat permasalahan klasifikasi dengan 4 buah kelas, untuk pelatihan digunakan 4 buah SVM 2 kelas seperti pada Tabel 2.1 dan penggunaannya dalam mengklasifikasi data baru dapat dilihat pada Gambar 2.5.

Tabel 2.1 Contoh 4 buah SVM 2 kelas dengan metode *One-Versus-Rest*.

$y_i = +1$	$y_i = -1$	Fungsi Keputusan
Kelas 1	Bukan kelas 1	$f^1(x_d) = w^1 \cdot x_d + b^1$
Kelas 2	Bukan kelas 2	$f^2(x_d) = w^2 \cdot x_d + b^2$
Kelas 3	Bukan kelas 3	$f^3(x_d) = w^3 \cdot x_d + b^3$
Kelas 4	Bukan kelas 4	$f^4(x_d) = w^4 \cdot x_d + b^4$



Gambar 2.5 Contoh klasifikasi dengan metode One-Versus- Rest

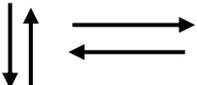
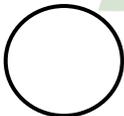
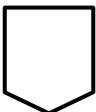
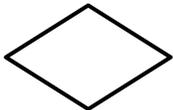
Berdasarkan Gambar 2.5, kelas dari suatu data  $x_d$  ialah kelas yang mempunyai nilai maksimal terhadap fungsi keputusan yang dinyatakan oleh persamaan (6).

$$class\ x = \operatorname{argmax} w^k \cdot x_d + b^k \quad (16)$$

### 2.15 Flowchart

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian (Murdock & Murdock, 2018). Dalam referensi lain flowchart adalah sebuah diagram yang dapat menjelaskan dengan detail proses sebuah program (Sari et al., 2020)

Tabel 2.2 Simbol-simbol *flowchart* (Murdock & Murdock, 2018)

Simbol	Arti	Keterangan
	<i>Input/Output</i>	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i> .
	Proses	Simbol proses digunakan untuk mewakili suatu proses.
	Garis Alir	Simbol garis alir ( <i>flow lines</i> ) digunakan untuk menunjukkan arus dari proses.
	Penghubung	Simbol penghubung ( <i>connector symbol</i> ) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama.
	Penghubung	Simbol penghubung ( <i>connector symbol</i> ) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman lainnya.
	Keputusan	Simbol keputusan ( <i>decision symbol</i> ) digunakan untuk suatu penyeleksian kondisi di dalam program.
	Proses Terdefinisi	Simbol proses terdefinisi ( <i>predifined process symbol</i> ) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.

## 2.16 Matlab

Matlab merupakan singkatan dari matrix laboratory. Matlab pada awalnya ditulis untuk memudahkan akses perangkat lunak matrik yang telah dibentuk oleh *linpack* dan *eispack*. Saat ini perangkat matlab telah menggabung dengan *lapack* dan *blas library*, yang merupakan satu kesatuan dari sebuah seni tersendiri dalam perangkat lunak untuk komputasi matrix. Dalam lingkungan perguruan tinggi teknik, Matlab merupakan perangkat standar untuk memperkenalkan dan mengembangkan penyajian materi matematika, rekayasa dan keilmuan. Di industri, matlab merupakan perangkat pilihan untuk penelitian dengan produktifitas yang tinggi, pengembangan dan analisisnya (Ilmiah & Exacta, 2011).

Matlab adalah sebuah bahasa dengan high performance untuk komputasi masalah teknik. Matlab mengintegrasikan komputasi, visualisasi, dan pemrograman dalam suatu model yang sangat mudah untuk dipakai dimana masalah-masalah dan penyelesaiannya diekspresikan dalam notasi matematika yang familiar. Penggunaan Matlab meliputi bidang matematika dan komputasi, pembentukan algoritma, akuisisi data, pemodelan, simulasi dan pembuatan prototipe, analisa data, eksplorasi dan visualisasi, grafik keilmuan dan bidang rekayasa (Alfiyan, 2017).

Matlab merupakan suatu sistem interaktif yang memiliki elemen data dalam suatu array sehingga kita tidak lagi dipusingkan dengan masalah dimensi. Hal ini memungkinkan kita untuk memecahkan banyak masalah teknis yang terkait dengan komputasi, khususnya yang berhubungan dengan matrik dan formulasi vektor, yang mana masalah tersebut merupakan momok apabila harus diselesaikan dengan menggunakan bahasa level rendah seperti Pascall, C dan Basic (Alfiyan, 2017). Sebagai sebuah sistem, Matlab tersusun dari 5 bagian utama:

### 1. *Development Environment*

*Development Environment* merupakan sekumpulan perangkat dan fasilitas untuk penggunaan fungsi-fungsi dan file-file Matlab. Beberapa perangkat ini merupakan sebuah *Graphical User Interfaces* (GUI). Termasuk didalamnya adalah Matlab desktop dan command window, command history, sebuah editor dan

debugger, browsers untuk melihat help, workspace, files, dan search path (Alfiyan, 2017).

## 2. *Matlab Mathematical Function Library*

*Matlab Mathematical Function Library* merupakan sekumpulan algoritma komputasi mulai dari fungsi-fungsi dasar seperti: sum, sin, cos, complex arithmetic, sampai dengan fungsi-fungsi yang lebih kompleks seperti matrix inverse, matrix eigenvalues, bessel functions, dan fast fourier transforms (Alfiyan, 2017).

## 3. *Matlab Language*

*Matlab Language* merupakan suatu *highlevel matrix/array language* dengan *control flow statements, functions, data structures, input/output*, dan fitur-fitur object-oriented programming. Ini memungkinkan bagi penggunanya untuk melakukan pemrograman dalam lingkup sederhana untuk mendapatkan hasil yang cepat, dan melakukan pemrograman dalam lingkup yang lebih besar untuk memperoleh hasil dan aplikasi yang kompleks (Alfiyan, 2017).

## 4. *Graphics*

Matlab memiliki fasilitas untuk menampilkan vector dan matrices sebagai suatu grafik. Didalamnya melibatkan high level functions untuk visualisasi data dua dimensi dan data tiga dimensi, *image processing, animation*, dan *presentation graphics* (Alfiyan, 2017).

## 5. *Matlab Application Program Interface (API)*

*Matlab Application Program Interface (API)* merupakan suatu *library* yang memungkinkan program yang telah ditulis dalam bahasa C dan Fortran mampu berinteraksi dengan Matlab. Ini melibatkan fasilitas untuk pemanggilan routines dari Matlab (*dynamic linking*), pemanggilan Matlab sebagai sebuah *computational engine*, dan untuk membaca dan menuliskan MATfiles (Alfiyan, 2017).