

## BAB II TINJAUAN PUSTAKA

### 2.1 Tinjauan Pustaka

Beberapa penelitian terkait menjadi bahan referensi dalam proses penyusunan tugas akhir yaitu terkait klasifikasi malaria dengan menggunakan metode *hu moment* dalam proses ekstraksi fitur dan metode *support vector machine* untuk klasifikasi penyakit malaria.

Penelitian pertama terkait dengan penelitian ini ialah klasifikasi *plasmodium falcifarum* dan *plasmodium vivax* pada 3 fase yaitu sporozit, skizon dan gametosit. Penelitian menggunakan 180 data citra dengan ukuran 50x50 piksel. Metode yang digunakan yaitu *support vector machine one against one*. Hasil akurasi dari proses klasifikasi dengan SVM yaitu rata-rata 90% dari tiap parasit (Permata, Purnama, & Purnomo, 2013).

Penelitian terkait selanjutnya yaitu identifikasi parasit malaria dengan menggunakan metode *moment invariant* untuk memberi masukan fitur *translation, rotation and scale (TRS)*. Klasifikasi parasit menggunakan metode *K-Nearest Neighbors* (K-NN) dan *gaussian naif bayes*. Penelitian menggunakan beberapa sampel citra yaitu 9 *P. falciparum*, 2 *P. knowlesi*, 6 *P. malariae*, 6 *P. Ovale* dan 7 *P. vivax* dengan ukuran 300x300 piksel. Beberapa tahapan yang digunakan yaitu *data collection, photomicrograph pre-processing, feature extraction, dan classification*. Hasil penelitian yaitu memiliki *true positive rates* 77.78% dan 88.89% sedangkan *true negative rates* 95.24% dan 80.95% pada masing-masing metode *k-nearest neighbors* dan *gaussian naif bayes* (Ravendran, T. de Silva, & Senanayake, 2015).

Peneitian terkait berikutnya menyatakan bahwa parasit malaria dapat dideteksi dengan langkah-langkah seperti akuisisi citra, segmentasi, ekstraksi fitur dan klasifikasi. Fitur yang digunakan berupa warna, bentuk, ukuran dan intensitas, tekstur akan diekstraksi dan digunakan untuk tahap segmentasi dan klasifikasi. Tingkat pertama dari klasifikasi yaitu memeriksa sel darah yang terpapar parasit dan selanjutnya penentuan jenis parasit menggunakan metode *support vector*

*machine*. Hasil penelitian didapatkan akurasi lebih dari 90% untuk mendeteksi parasit malaria (Charpe & Bairagi, 2015)

Penelitian berikutnya yang menjadi rujukan ialah klasifikasi parasit malaria dengan membandingkan berbagai jenis metode *support vector machine* dan *k-nearest neighbor*. Penelitian dilakukan dengan 130 300x300 sampel citra dalam beberapa tahapan proses yaitu *preprocessing*, *image segmentation*, *feature extraction* menggunakan metode *Speed Up Robust Features* (SURF) dan *classification* dengan hasil nilai *true positive* paling besar pada jenis metode *fine gaussian SVM* dengan nilai 99.8% (Olugboja & Wang, 2017).

Penelitian terkait yang menggunakan metode *support vector machine* ialah klasifikasi malaria dengan kombinasi warna menggunakan metode *StatMom* dan fitur tekstur dengan metode *Histogram of Oriented Gradien (HOG)*, *Local Binarry Pattern* (LBP) dan *Grey-Level Co-occurrence Matrix* (GLCM). Sampel yang digunakan sebesar 46.973 gambar yang tersusun dalam beberapa kelas yaitu 20.001 *uninfected*, 5.742 *ring*, 17.424 *trophozoite*, 2.090 *schizont*, dan 1.716 *gametocytes*. Metode pengabungan fitur warna dan tekstur, yaitu YCrCb-HOG dengan klasifikasi SVM menunjukkan kinerja tertinggi dengan akurasi 96.9% dan rata-rata akurasi 85.4% (Bashar, 2019).

Penelitian terakhir yang menjadi rujukan penelitian ini ialah klasifikasi 4 jenis parasit malaria *plasmodium falciparum*, *malariae*, *ovale* dan *vivax* dengan metode *SVM non linier* serta fungsi kernel *Radial Base Function* (RBF). Hasil akurasi tertinggi yaitu 72.5% (Biantong, Furqon, & Soebroto, 2019).

Tabel 2.1 Rangkuman Tinjauan Pustaka

No	Penulis (tahun)	Judul	Objek	Metode	Hasil
1	Endi Permata., et all (2013)	Klasifikasi Jenis dan Fase Parasit Malaria <i>Plasmodium</i>	Malaria	<i>Support Vector Machine One Against One</i>	Akurasi= 85%- 95,55%

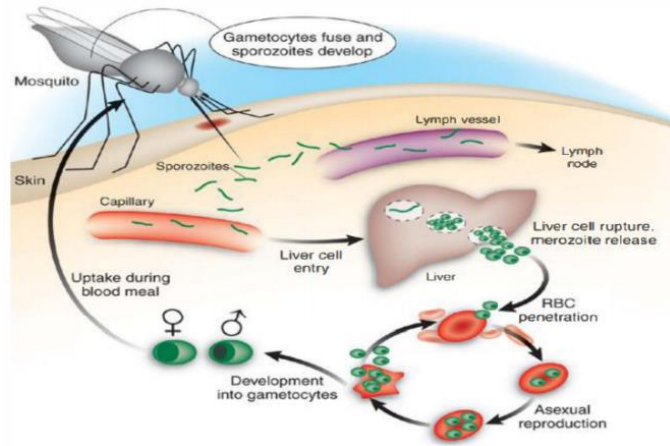
		<i>Falciparum</i> dan <i>Plasmodium Vivax</i> dalam Sel Darah Merah Menggunakan <i>Support Vector Machine One Against One</i>			
2	Ahalya Ravendran., et all (2015)	<i>Moment Invariant Features for Automatic Identification of Critical Malaria Parasits</i>	Malaria	Ekstraksi= <i>Hu moment</i> Klasifikasi= KNN dan <i>gaussian naive bayes</i>	KNN=77.78 % TP dan 95.24% TN <i>gaussian naive bayes</i> =88.8 9% TP dan 80,95% TN
3	Kshipra C. Charpe and V. K. Bairagi (2015)	<i>Automated Malaria Parasit and there Stage Detection in Microscopic Blood Images</i>	Malaria	Klasifikasi= SVM	Akurasi= Lebih dari 90%
4	Adedeji Olugboja and Zenghui Wang (2017)	<i>Malaria Parasit Detection Using Different Machine Learning Classifier</i>	Malaria	Ekstraksi= SURF Klasifikasi= Jenis-jenis metode SVM dan KNN	Akurasi= 82.6%- 86.3%

5	Md. Khayrul Bashar (2019)	<i>Improved Classification of Malaria Parasit Stages with Support Vector Machine Using Combined Color and Texture Features</i>	Malaria	Ekstraksi= <i>StatMom</i> , HOG, LBP dan GLCM Klasifikasi= SVM	Akurasi= 96.9%
6	Tryse Rezza Biantong., et all (2019)	Implementasi <i>Metode Support Vector Machine</i> untuk Klasifikasi Jenis Penyakit Malaria	Malaria	<i>Support Vector Machine</i>	Akurasi= 72.5 %

## 2.2 Dasar Teori

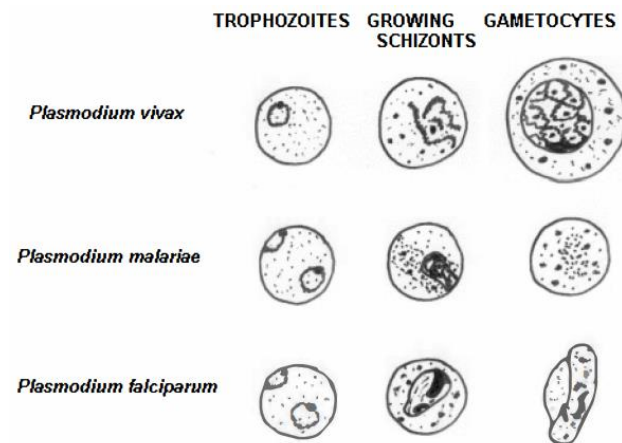
### 2.2.1 Malaria

Malaria merupakan salah satu penyakit berbahaya yang berasal dari suntikan nyamuk *anopheles* betina yang terinfeksi parasit *plasmodium* ke dalam darah manusia. Parasit menetap di organ hati untuk berkembang biak dan akhirnya merusak sel darah merah. Terdapat 4 jenis parasit malaria yaitu *plasmodium falciparum* penyebab penyakit malaria tropika, *plasmodium vivax* penyebab penyakit malaria tertiana, *plasmodium ovale* penyebab penyakit malaria ovale dan *plasmodium malariae* penyebab penyakit malaria kuartana. Parasit *falciparum* menjadi jenis parasit yang paling berbahaya dengan resiko kematian tertinggi dibandingkan dengan jenis yang lain yang memiliki efek sedang (Ravendran, T. de Silva, & Senanayake, 2015). Siklus penyakit malaria ditunjukkan pada Gambar 2.1.



Gambar 2.1 Siklus hidup parasit malaria (Charpe & Bairagi, 2015)

Nyamuk menyuntikkan parasit berupa *sporozites* ke dalam darah kemudian masuk kedalam hati dan menghancurkan sel-sel hati. Sel hati yang terinfeksi kemudian mengandung *sporozites* matang dan *mezosites* tersebar di seluruh darah dan menginfeksi sel-sel darah merah. *Mezosites* ini terdiri dari *trophozites*, *schizontes*, *gametosytes* yang merupakan fase dari parasit (Charpe & Bairagi, 2015). Fase penyakit malaria ditunjukkan seperti pada Gambar 2.2.

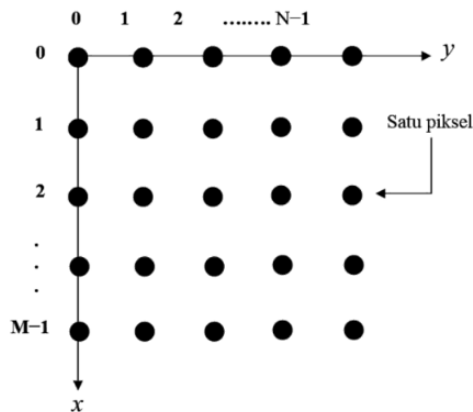


Gambar 2.2 Perbedaan bentuk parasit malaria pada tiga fase berbeda (Charpe & Bairagi, 2015)

### 2.2.2 Citra Digital

Citra digital adalah representasi gambar 2 dimensi yang terdiri dari M baris dan N kolom dimana perpotongan baris dan kolom disebut piksel. Piksel citra terdiri dari 2 bagian yaitu koordinat (x,y) dan nilai dari koordinat itu sendiri

yang disebut fungsi intensitas  $f(x,y)$ . Pengolahan citra adalah ilmu yang mempelajari tentang algoritma transformasi citra yang bertujuan untuk proses pengambilan informasi atau pengenalan objek yang terkandung pada citra (RM Sinaga, 2017). Sistem koordinat citra dalam teori pengolahan citra ditunjukkan pada Gambar 2.3.



Gambar 2.3 Sistem koordinat yang mewakili citra (Andono, Sutojo, & Muljono, 2017)

Secara matematis rumus persamaan representasi citra dari  $M$  baris dan  $N$  kolom dapat dilihat pada Gambar 2.4. Fungsi intensitas  $f(x,y)$  merupakan bentuk matrik yang terdiri dari kumpulan nilai-nilai dari tiap posisi koordinat sumbu  $x$  dan  $y$ . Semakin besar nilai  $M$  dan  $N$  maka semakin banyak jumlah koordinat dan bebanding lurus dengan nilai intensitas sehingga semakin bagus citra digital yang dihasilkan. Rumus persamaan representasi citra digital  $M \times N$  dapat dilihat pada Gambar 2.4.

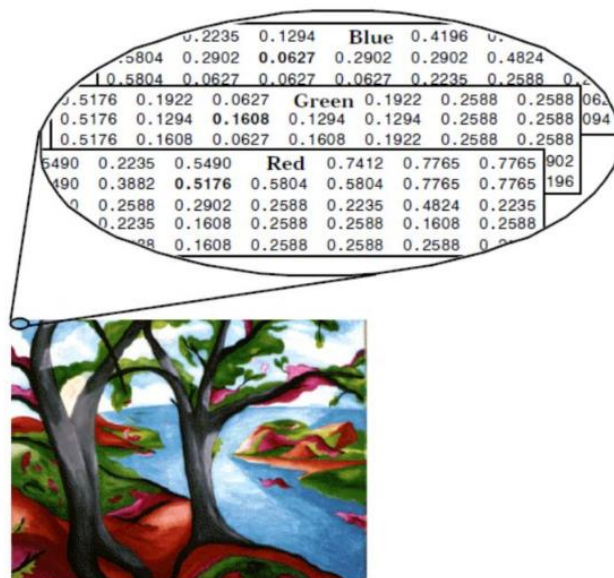
$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & \dots & \dots & f(1,M-1) \\ \dots & \dots & \dots & \dots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2.4 Rumus persamaan representasi citra digital  $M \times N$  (Andono, Sutojo, & Muljono, 2017)

Terdapat 3 jenis citra berdasarkan nilai piksel yaitu citra warna, citra biner dan citra abu (*grayscale*).

a. Citra Warna

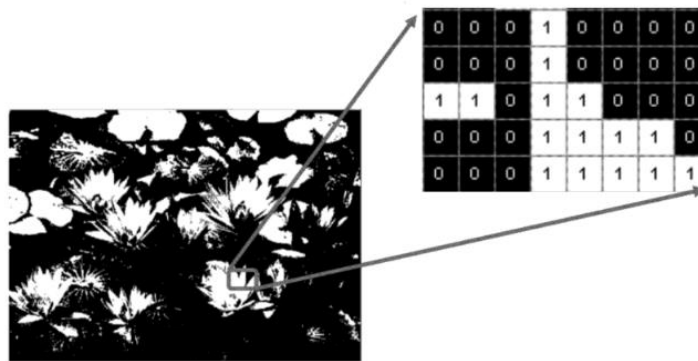
Citra warna adalah citra yang tiap-tiap piksel terdiri dari kumpulan warna merah (*red*), hijau (*green*) dan biru (*blue*) atau yang lebih dikenal dengan citra warna RGB. Warna RGB akan membentuk warna baru dengan kombinasi dan proporsi dari ketiga jenis warna tersebut yang akan disimpan pada tiap-tiap lokasi piksel. Masing-masing warna terdiri dari 8 bit sehingga total bit dari citra RGB yaitu 24 bit sehingga citra warna memiliki 24 juta kemungkinan (Andono, Sutojo, & Muljono, 2017). Citra warna ditunjukkan pada Gambar 2.5.



Gambar 2.5 Citra warna (Asmara, 2018)

b. Citra biner

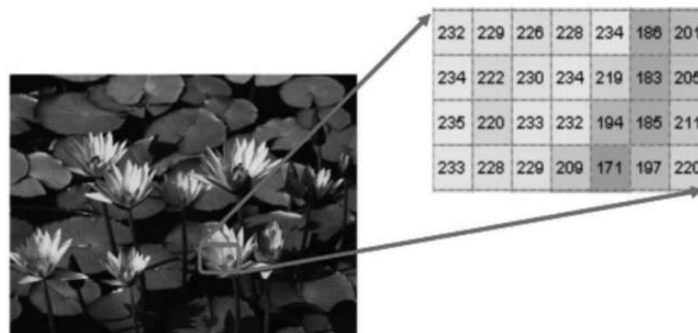
Citra biner adalah citra yang memiliki 2 warna saja yaitu hitam dan putih. Jika direpresentasikan dengan nilai biner yaitu 8 bit adalah warna hitam bernilai 0000 0000 dan putih bernilai 1111 1111. Biasa ditampilkan dengan nilai normalisasi 0 dan 1, atau desimal 0 dan 255, atau heksadesimal 00x dan FFx (Asmara, 2018). Citra biner ditunjukkan pada Gambar 2.6.



Gambar 2.6 Citra biner (Andono, Sutojo, & Muljono, 2017)

c. Citra *grayscale*

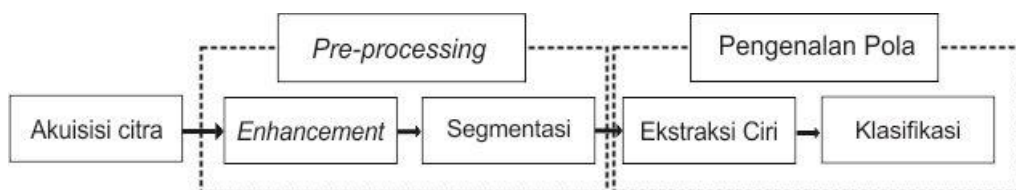
Citra *grayscale* adalah matrik data yang nilai-nilainya mewakili intensitas setiap piksel berkisar antara 0 sampai 255. Setiap piksel membutuhkan 8 bit memori” (Andono, Sutojo, & Muljono, 2017). Citra abu dapat dilihat seperti pada Gambar 2.7.



Gambar 2.7 Citra abu (Andono, Sutojo, & Muljono, 2017)

2.2.3 Algoritma Pengenalan Pola

Menurut (Andono, Sutojo, & Muljono, 2017) secara umum semua algoritma pengenalan pola terdiri dari tiga komponen utama yaitu persepsi data, ekstraksi ciri dan klasifikasi. Algoritma pengenalan pola secara umum dapat ditampilkan pada Gambar 2.8.



Gambar 2.8 Algoritma pengenalan pola



a. Akuisisi Citra

Tahap pertama merupakan akuisisi citra yaitu tahapan perubahan citra analog menjadi citra digital menggunakan alat digital seperti kamera digital, *smartphone*, *scanner* dan lain lain. Ada beberapa hal yang mempengaruhi kualitas citra yaitu resolusi alat, pencahayaan, jarak atau sudut pandang pengambilan citra, perbesaran atau pengecilan serta kondisi pengambilan objek dan lain-lain.

b. *Preprocessing*

*Preprocessing* merupakan tahap pemrosesan data agar menjadi layak untuk digunakan pada tahap selanjutnya. Tujuan *preprocessing* adalah mengurangi masalah atau gangguan (*noise*) ketika proses akuisisi. Tahapan *preprocessing* memiliki 2 tahapan antara lain:

1. *Enhancement*

*Enhancement* merupakan bagian *preprocessing* untuk meningkatkan kualitas citra. Berikut merupakan beberapa metode yang digunakan dalam proses *enhancement* yaitu *intensity adjustment*, *contrast*, *stretching*, *filtering* (*median filter*, *low pass filter*, *high pass filter*) dan lain-lain.

2. Segmentasi

Segmentasi merupakan proses pemisahan objek tertentu yang diinginkan (*foreground*) dengan objek lain yang tidak diinginkan (*background*). Hasil dari proses segmentasi yaitu citra biner dengan *foreground* diberi simbol 1 dan *background* diberi simbol 0. Metode segmentasi yang biasa digunakan ialah deteksi tepi, *thresholding*, *watershed*, *transformasi hough*, *k-means clustering* dan lain-lain.

c. Pengenalan Pola

Pengenalan pola merupakan tahap terakhir untuk mengetahui berbagai macam pola-pola yang didapatkan dari proses segmentasi. Pengenalan pola merupakan salah satu bidang dalam komputer sains, yang memetakan suatu data ke dalam konsep tertentu yang telah didefinisikan sebelumnya. Konsep tertentu ini disebut *class* atau

*category* (Nugroho, Witarto, & Handoko, 2003). Salah satu penerapan *pattern recognition* adalah klasifikasi jenis penyakit secara otomatis. Pengenaan pola terdiri dari 2 tahapan penting yaitu:

1. Ekstraksi Ciri

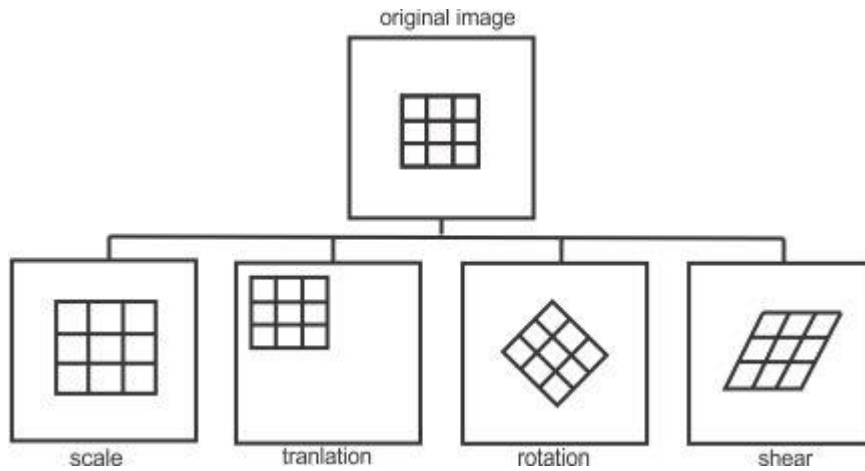
Ekstraksi ciri merupakan proses pengambilan ciri objek yang unik untuk menjadi pembeda dengan objek-objek lainnya. Pada penelitian ini difokuskan pada penggunaan metode *hu moment* yang akan dijelaskan lebih rinci pada sub bab 2.2.4.

2. Klasifikasi

Klasifikasi citra adalah proses pengelompokan seluruh piksel pada suatu citra kedalam kelas-kelas sehingga dapat diinterpretasikan sebagai suatu *property* yang spesifik. *Output* dari proses klasifikasi ialah pengelompokan kelas-kelas dari objek yang memiliki kesamaan ciri. Metode klasifikasi yang digunakan penulis dalam penelitian ini ialah *support vector machine* yang lebih rinci dijelaskan pada sub bab 2.2.5.

#### 2.2.4 *Hu Moment*

Salah satu masalah dalam pengenalan gambar ialah pengenalan otomatis dari pola gambar terlepas dari ukuran, posisi dan orientasinya yang dapat dicapai dengan cara membuat ekstraksi ciri yang invarian terhadap transformasi citra termasuk didalamnya skala, translasi dan rotasi (Chen, Petriu, & Yang, 2004). Transformasi citra dapat dilakukan dengan cara transformasi affin yang ditunjukkan pada Gambar 2.9.



Gambar 2.9 Transformasi Affin

Metode *Hu moment* atau momen invarian memiliki *property* objek yang invarian (tetap) terhadap transformasi affin sehingga dapat diartikan momen invarian ialah kumpulan momen yang merepresentasikan karakteristik citra serta memberikan informasi geometri citra berupa 7 nilai fitur yang invarian terhadap skala, translasi dan rotasi. Fungsi skala yaitu memperkecil atau memperbesar ukuran gambar dengan cara subsampling atau interpolasi untuk gambar input. Fungsi translasi yaitu memetakan posisi setiap piksel dalam gambar input ke posisi baru dalam gambar output. Fungsi rotasi yaitu memetakan posisi masing-masing piksel ke posisi baru dengan putaran pada sudut tertentu yang menghasilkan koordinat *non interger* (Huang & Leng, 2010). Momen invarian 2 dimensi p dan q dapat didefinisikan oleh Persamaan 2.1 sebagai berikut:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy, \quad p, q = 0, 1, 2, 3 \quad (2.1)$$

Dimana  $m_{pq}$  merupakan order moment (p+q) dari fungsi f(x,y).

*central moment* dari fungsi f(x,y) dapat didefinisikan dengan Persamaan 2.2 sebagai berikut:

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy, \quad p, q = 0, 1, 2, 3 \quad (2.2)$$

Dimana  $\bar{x} = m_{10}/m_{00}$  , dan  $\bar{y} = m_{01}/m_{00}$ . Titik piksel  $(\bar{x}, \bar{y})$  adalah pusat citra fungsi f(x,y). *Central moment* dari beberapa *order moment* didefinisikan oleh Persamaan 2.3 sebagai berikut:

$$\begin{aligned}
\mu_{00} &= m_{00} = \mu \\
\mu_{10} &= \mu_{01} = 0 \\
\mu_{20} &= m_{20} - \mu\bar{x}^2 \\
\mu_{11} &= m_{11} - \mu\bar{x}\bar{y} \\
\mu_{02} &= m_{02} - \mu\bar{y}^2 \\
\mu_{30} &= m_{30} - 3m_{20}\bar{x} + 2\mu\bar{x}^3 \\
\mu_{21} &= m_{21} - m_{20}\bar{y} - 2m_{11}\bar{x} + 2\mu\bar{x}^2\bar{y} \\
\mu_{12} &= m_{12} - m_{02}\bar{x} - 2m_{11}\bar{y} + 2\mu\bar{x}\bar{y}^2 \\
\mu_{03} &= m_{03} - 3m_{02}\bar{y} + 2\mu\bar{y}^3
\end{aligned} \tag{2.3}$$

*Central moment* normalisasi didefinisikan dengan Persamaan 2.4 sebagai berikut:

$$\eta^{pq} = \frac{\mu_{pq}}{(\mu_{00})^\gamma} \text{ dengan } \gamma = \frac{(p+q+2)}{2}, p+q = 2,3.. \tag{2.4}$$

*Central moment* yang telah ternormalisasi memiliki 7 fungsi *non* linier yang invarian terhadap translasi, skala dan rotasi yang didefinisikan dengan Persamaan 2.5 sebagai berikut:

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2 \\
\phi_3 &= (\eta_{30} + 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \\
\phi_5 &= (\eta_{30} + \eta_{12})^2(\eta_{30} + \eta_{12}) + [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{21} - \eta_{03})[3(\eta_{30} + \eta_{12})^2 \\
&\quad - (\eta_{21} - \eta_{03})^2] \\
\phi_6 &= (\eta_{20} + \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} - \eta_{03})^2] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} - \eta_{03}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] \\
&\quad - (\eta_{30} + 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 \\
&\quad - (\eta_{21} - \eta_{03})^2]
\end{aligned} \tag{2.5}$$

Penelitian-penelitian terkait yang menggunakan ekstraksi fitur dengan metode *hu moment* dapat dilihat pada Tabel 2.2.

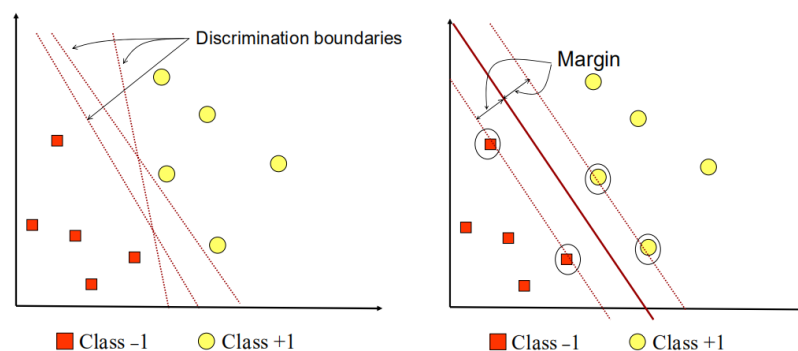
Tabel 2.2 Penelitian terkait dengan metode *hu moment*

No	Penulis (Tahun)	Judul
1	Marko Lukic (2017)	<i>Leaf Recognition Algorithm using Support Vector Machine with Hu Moments and Local Binary Patterns</i>
2	Rahma Juwita Sany., et all (2017)	Penggunaan Ciri <i>Geometric Invariant Moment</i> pada Pengenalan Tanda Tangan
3	Oky Dwi Nurhayati dan Isti Pudjihastuti (2018)	Metode <i>Moment Invariant Geometrik</i> untuk Menganalisis Jenis Daging Babi dan Daging Sapi
4	Aysegul Guven (2019)	<i>Alcoholism Detection Based On Hu Moment Invariants Of Event-Related Potentials</i>
5	Riska Yulianti (2019)	Pengenalan Pola Tulisan Tangan Suku Kata Aksara Sasak Menggunakan Metode <i>Moment Invariant dan Support Vector Machine</i>

Tabel 2.1 merangkum beberapa contoh penelitian yang pernah dilakukan 3 tahun terakhir dengan metode *hu moment* atau *moment invariant* seperti deteksi jenis daun, deteksi alcohol, deteksi jenis daging, tulisan dan tanda tangan serta masih banyak lagi.

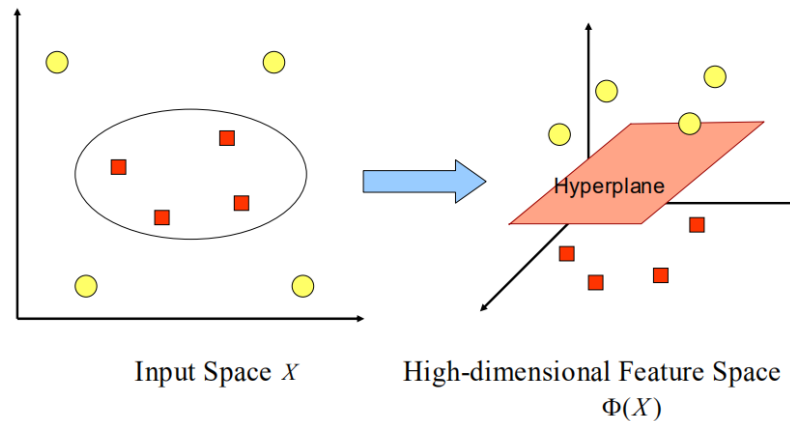
### 2.2.5 Support Vector Machine

*Support vector machine* adalah salah satu metode klasifikasi yang bekerja dengan cara memisahkan kelas-kelas dengan menggunakan bidang pemisah berupa garis lurus yang disebut dengan *hyperplane*. Bidang pemisah yang baik ialah yang memiliki jarak batasan (*margin*) terbesar antara garis lurus dengan pola terdekat dari masing-masing kelas yang disebut dengan *support vector*. *Hyperplane* terbaik dapat ditemukan dengan mengukur margin *hyperplane* dan mencari titik maksimalnya (Biantong, Furqon, & Soebroto, 2019). Pemisahan kelas-kelas oleh SVM dapat dilihat seperti pada Gambar 2.8.



Gambar 2.10 Pemisahan dua kelas -1 dan +1 dengan SVM (Nugroho, Witarto, & Handoko, 2003)

Gambar 2 menunjukkan ada beberapa *pattern* yang terkelompok dalam 2 kelas -1 (merah) dan +1 (kuning). Terdapat 2 kasus dalam memisahkan kelas-kelas oleh *hyperplane* yaitu kelas yang dapat dipisahkan secara sempurna disebut dengan SVM linier dan kelas yang yang tidak dapat dipisahkan dengan sempurna disebut SVM *non* linier. Ilustrasi konsep SVM *non* linier dapat dilihat seperti pada Gambar 2.11.



Gambar 2.11 Ilustrasi konsep SVM non linier (Nugroho, Witarto, & Handoko, 2003)

Pada dasarnya SVM *non* linier merupakan solusi dari masalah SVM linier dengan melakukan fungsi kernel pada ruang fitur dimensi tinggi. Definisi persamaan fungsi kernel linier dan non linier dapat dilihat pada Tabel 2.3.

Tabel 2.3 Definisi persamaan SVM linier dan *non* linier

Sifat SVM	Jenis kernel	Definisi Rumus
SVM Linier	Linier	$K(x, y) = x \cdot y$
SVM non linier	Polynomial	$K(x, y) = (x \cdot y + 1)^p$
	Gaussian RBF	$K(x, y) = e^{- x-y ^2/2\sigma^2}$
	Sigmoid	$K(x, y) = \tanh(Kx \cdot y - \delta)$

Keterangan:

p : pangkat (*degree of*)

$\sigma$  : nilai *sigma*

$\delta$  : nilai *delta*

Langkah-langkah penyusunan algoritma *support vector machine* dapat disajikan sebagai berikut (Huang, Davis, & Townshed, 2002):

1. Menentukan jumlah kelas yang akan diuji dan menyusun *training sample* yang terdiri dari piksel-piksel tiap kelas yang telah ditentukan.

2. Memetakan fungsi seluruh vektor  $x$  ke ruang fitur dimensi tinggi dan membangun *hyperplane* yang optimal melalui fungsi pemetaan  $\Phi$  melalui Persamaan 2.6 sebagai berikut:

$$\Phi: R^n \rightarrow H \quad (2.6)$$

Vektor  $x$  dalam ruang fitur dapat diwakili sebagai  $\Phi(x)$  pada ruang fitur dimensi tinggi.

3. Melakukan fungsi kernel untuk mencari *hyperplane* terbaik dari 2 data vektor. Algoritma pelatihan dapat dilakukan oleh fungsi kernel  $K$  tanpa mengetahui bentuk eksplisit dari  $\Phi$ . Berikut merupakan persamaan fungsi kernel  $K$  yang didefinisikan lewat Persamaan 2.7 sebagai berikut:

$$K(x_i, x_j) = \Phi(x_i)\Phi(x_j) \quad (2.7)$$

4. Fungsi kernel lalu dimasukkan kedalam fungsi *lagrangian* untuk menemukan *hyperplane* optimum melalui Persamaan 2.8 sebagai berikut:

$$L(a) = \sum_i^k \alpha_i - \frac{1}{2} \sum_{i=1}^k \sum_{j=1}^k \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (2.8)$$

5. Piksel  $x$  kemudian dapat diklasifikasikan oleh fungsi putusan SVM dengan Persamaan 2.9 sebagai berikut:

$$f(x) = \text{sign} \left( \sum_{\text{support vector}} y_i \alpha_i^0 K(x_i, x) - b^0 \right) \quad (2.9)$$

Keterangan:

$x_i$  = vektor dari data (piksel)

$x_j$  = vektor pada data training sampel

Penelitian terkait yang menggunakan klasifikasi dengan metode *support vector machine* telah banyak dilakukan peneliti seperti yang telah dirangkum pada Tabel 2.4.



Tabel 2.4 Penelitian terkait menggunakan metode SVM

No	Penulis (Tahun)	Judul
1	Andi Ade Yustika., et all (2019)	Deteksi Anemia Melalui Citra Sel Darah Menggunakan Metode <i>Discrete Wavelet Transform</i> (DWT) dan Klasifikasi <i>Support Vector Machine</i> (SVM)
2	Romi Wiryadinata., et all (2019)	Klasifikasi 12 Motif Batik Banten Menggunakan <i>Support Vector Machine</i>
3	Anisa Salama., et all (2019)	Klasifikasi Topik Ayat Al-Qur'an Terjemahan Berbahasa Inggris Menggunakan <i>Metode Support Vector Machine</i> Berbasis <i>Vector Space</i> Model dan <i>Word2Vec</i>
4	Ramadhani., et all (2020)	Klasifikasi Otomatis Motif Tekstil Menggunakan <i>Support Vector Machine</i> Multi Kelas
5	Mana Jain., et all (2020)	<i>Speech Emotion Recognition using Support Vector Machine</i>

Pada Tabel 2.4 adalah sedikit contoh penelitian yang pernah dilakukan dengan metode *support vector machine* 2 tahun terakhir. Berdasarkan Tabel 2.4 dapat diasumsikan bahwa metode SVM dapat digunakan untuk berbagai klasifikasi seperti penyakit, emosi, motif tekstil bahkan klasifikasi topik pada suatu bacaan terjemahan dengan akurasi yang beragam.

### 2.2.6 Confusion Matrix

*Confusion matrix* adalah matrik yang berisikan informasi data prediksi dan data aktual yang terdiri dari 4 jenis tipe yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* yang dapat dilihat pada Tabel 2.5.

Tabel 2.5 Tabel *confusion matrix*

		Nilai Prediksi	
		Positif	Negatif
Nilai Aktual	Positif	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>
	Negatif	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

Keterangan :

TP : Data positif yang diprediksi benar.

TN: Data negatif yang diprediksi benar.

FP : Data negatif diprediksi sebagai data positif.

FN : Data positif diprediksi sebagai data negatif.

*Confusion matrix* dapat menghasilkan banyak pengukuran seperti *True Positive Rate* (TPR)/*recall*/sensitivitas yang didefinisikan dengan Persamaan 2.10. *True Negative Rate* (TNR)/spesifisitas yang didefinisikan dengan persamaan 2.11. *False Positive Rate* (FPR) yang didefinisikan dengan Persamaan 2.12. *False Negative Rate* (FNR) yang didefinisikan dengan Persamaan 2.13. Presisi yang didefinisikan dengan Persamaan 2.14. Akurasi yang didefinisikan dengan Persamaan 2.15.

$$TPR = \frac{TP}{TP + FN} \quad (2.10)$$

$$TNR = \frac{TN}{TN + FP} \quad (2.11)$$

$$FPR = \frac{FP}{FP + TN} = 1 - TNR \quad (2.12)$$

$$FNR = \frac{FN}{FN + TP} = 1 - TPR \quad (2.13)$$

$$Presisi = \frac{TP}{TP + FP} \quad (2.14)$$

$$Akurasi = \frac{TP + TN}{TP + FP + FN + TN} \quad (2.15)$$

Keterangan:

TPR : Rasio prediksi benar positif dibandingkan dengan keseluruhan data benar positif (perolehan).

TNR : Rasio prediksi benar negatif dibandingkan dengan keseluruhan data benar negatif.

FPR : Rasio data negatif diprediksi sebagai data positif.

FNR : Rasio data positif diprediksi sebagai data negatif.

Presisi : Tingkat ketepatan dari informasi yang diminta dengan informasi yang diberikan oleh sistem (ketepatan).

Akurasi : Rasio benar positif dan negatif dari keseluruhan data.