

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Sebagai bahan acuan dalam mengerjakan tugas akhir ini akan dipaparkan hasil penelitian pendahulu yang digunakan sebagai referensi oleh penulis, diantaranya :

Penelitian yang dilakukan oleh Refta Listia dan Agus Harjoko (2014) dengan judul “Klasifikasi Massa pada Citra *Mammogram* Berdasarkan *Gray Level Cooccurrence Matrix (GLCM)*” menjelaskan bahwa penelitian ini adalahn klasifikasi citra pada *mammogram* untuk penyakit kanker payudara. Kanker payudara adalah penyakit yang paling umum diderita oleh wanita pada banyak negara. Pemeriksaan kanker payudara dapat dilakukan dengan menggunakan *mamografi*. Pada penelitian ini, pendekatan yang diusulkan bertujuan untuk mengklasifikasi *mammogram* berdasarkan tiga kelas yaitu kelas normal, tumor jinak, dan tumor ganas. Sistem yang diusulkan terdiri dari empat langkah utama yaitu preprosesing, segmentasi, ekstraksi fitur dan klasifikasi. Pada tahap preprosesing akan dilakukan *grayscale*, *interpolasi*, *amoeba mean filter* dan segmentasi. Ekstraksi ciri menggunakan *Gray Level Cooccurrence Matrix (GLCM)* dan akan dihitung ciri-ciri statistik pada 4 arah ($d=1$ dan $d=2$) , *GLCM 8 arah ($d=1$)* dan *GLCM 16 arah ($d=2$)*. Fitur yang digunakan ada 5 yaitu kontras, energi, *entropi*, korelasi dan *homogenitas*. Langkah terakhir adalah klasifikasi menggunakan *Backpropagation*. Beberapa parameter penting divariasikan dalam proses ini seperti *learning rate* dan

jumlah *node* dalam lapisan tersembunyi. Hasil penelitian menunjukkan bahwa fitur ekstraksi GLCM 4 arah (dengan jarak $d=1$ memiliki akurasi terbaik dalam mengklasifikasi *mammogram* yaitu sebesar 81,1% dan khusus pada arah akurasi klasifikasi diperoleh sebesar 100%. Data yang digunakan dalam penelitian ini berupa citra *mammogram* dari *database* MIAS (*Mammographic Image Analysis Society*). Objek dari penelitian ini adalah kedua payudara wanita yang diambil dari posisi kanan dan kiri (RCC dan LCC) dengan ukuran 1024 piksel x 1024 piksel yang disimpan dalam bentuk *bitmap*. Citra yang digunakan dalam penelitian ini berjumlah 103 citra yang terdiri dari 62 data latih (20 citra normal, 20 citra tumor ganas, 22 citra tumor jinak) dan 41 citra data uji (15 citra normal, 7 citra tumor ganas, 19 citra tumor jinak). Setiap citra akan diekstraksi menggunakan *Gray Level Cooccurrence Matrix* (GLCM) sehingga untuk orientasi 4 arah (dan) jarak $d=1$ dan $d=2$ total citra untuk pelatihan sebanyak 248 dan total citra uji sebanyak 164 citra. Sedangkan untuk ekstraksi ciri citra dengan menggunakan GLCM 8 dan 16 arah citra yang digunakan sebanyak 103 citra. Berdasarkan analisis sistem maka untuk membangun sistem klasifikasi citra dibutuhkan rancangan proses. Alur sistem yang secara garis besar meliputi input citra, *preprocessing*, *segmentasi*, pembentukan GLCM, ekstraksi GLCM, pelatihan dan pengujian JST-BP untuk mendapatkan hasil klasifikasi citra. Kesimpulan yang didapat didalam penelitian ini adalah (1) Klasifikasi dengan menggunakan metode ekstraksi ciri GLCM 4 arah dan $d=1$ dimana menggunakan 2 layer tersembunyi dengan 19 *node* pada *layer* tersembunyi, *learning rate* (α) = 0,001 memiliki rata-rata akurasi sebesar 81,1%. (2) Klasifikasi dengan menggunakan metode ekstraksi

ciri GLCM dengan jarak $d=1$ dimana menggunakan 2 *layer* tersembunyi dengan 19 *node* pada *layer* tersembunyi, *learning rate* (α) = 0,001 memiliki sensitivitas, *spesifisitas* dan akurasi sebesar 100% yaitu pada orientasi arah. (3) Metode ekstraksi ciri GLCM 4 arah dengan $d=1$ mampu mengklasifikasi citra *mammogram* ke dalam kelas normal, tumor ganas dan tumor jinak.

Penelitian yang dilakukan oleh Paulus Simanjuntak, Irma, Nuning Kurniasih, Mesran, dan Janner Simarmata (2018) dengan judul “Penentuan Kayu Terbaik Untuk Bahan Gitar Dengan Metode *Weighted Aggregated Sum Product Assessment* (WASPAS)” menjelaskan bahwa dalam penelitian ini penulis membuat sistem pendukung keputusan penentuan kayu terbaik untuk bahan gitar dengan metode *Weighted Aggregated Sum Product Assessment* (WASPAS), metode ini dipilih karena mampu menyeleksi alternatif terbaik dari sejumlah alternatif yang ada, dalam hal ini alternatif yang dimaksudkan yaitu kayu yang terpilih berdasarkan kriteria-kriteria yang telah ditentukan. Sistem pendukung keputusan ini dapat memberikan keputusan alternatif kayu terpilih yang nantinya dapat dijadikan sebagai acuan untuk menentukan kayu terbaik dalam pembuatan gitar. Metode *Weighted Aggregated Sum Product Assessment* (WASPAS) dalam prosesnya memerlukan kriteria-kriteria yang akan dijadikan bahan perhitungan pada proses perankingan dan penilaian penentuan kayu terbaik untuk membuat gitar, kriteria-kriteria yang akan menjadi bahan pertimbangan pada proses perankingan tersebut yaitu jenis kayu, serat kayu, tekstur, dan berat. Penulis juga menentukan rangking kecocokan setiap kriteria yang dinilai dari sangat baik, baik, cukup, buruk, dan sangat buruk. Dari kriteria yang sudah ditentukan penulis membuat tingkat kepentingan

kriteria berdasarkan nilai bobot yaitu sangat baik bernilai (1), baik bernilai (2), cukup bernilai (3), buruk bernilai (4), dan sangat buruk bernilai (5). Hasil kesimpulan penelitian ini adalah dengan mengidentifikasi dan menentukan kriteria-kriteria dan bobot yang paling berpengaruh dalam penentuan kayu terbaik untuk membuat gitar merupakan langkah pertama pada metode *Weighted Aggregated Sum Product Assessment* (WASPAS) yang harus dilakukan sebelum melakukan *normalisasi* nilai R_{ij} dan penghitungan nilai Q_i , sehingga diperoleh hasil sebagai suatu solusi untuk mendukung pengambilan keputusan dalam penentuan kualitas kayu terbaik tersebut.

2.2 Perbandingan Metode

➤ Perbandingan Metode JST-RBF dengan JST Backpropagation

Observasi dan model prediksi yang menggunakan kedua metode ini pada data jumlah mahasiswa baru periode 2012-2016. Dengan menggunakan parameter yang sama yaitu, *target error*=0,0001 dan *learning rate*=0,01, kedua metode ini memberikan hasil *performance* yang berbeda, seperti yang ditampilkan pada Tabel 2.1 berikut ini.

Tabel 2. 1 Statistik *Performance* JST-RBF dan Backpropagation

| <i>Performance</i> | JST-RBF | JST Backpropagation |
|--------------------|----------|---------------------|
| MAE | 0,0074 | 0,2129 |
| RMSE | 0,0096 | 0,2752 |
| <i>Error</i> | 12,6532% | 13,3217% |

Hasil simulasi dan analisis data yang dilakukan, maka dapat diperoleh suatu kesimpulan berikut ini :

1. Model prediksi yang terbaik menggunakan metode JST RBF dengan parameter-parameter, Target *Error*=0,0001 dan *Learning Rate*= 0,01. Menggunakan metode JST Backpropagation dengan parameter-parameter, Target *Error* = 0,0001 dan *Learning Rate*= 0,01, dengan fungsi aktivasi "tansig" untuk semua layer, dan fungsi "traingd" untuk fungsi *training*.

2. Menggunakan parameter-parameter (Target *Error*=0,0001 dan *Learning Rate*= 0,01) yang sama pada kedua metode dalam memprediksi jumlah mahasiswa baru di Universitas Wijaya Putra Surabaya, diperoleh perbedaan pada indeks statistik MAE, RMSE, dan *Error*. Indeks statistik menggunakan metode JST RBF diperoleh MAE=0,0074, RMSE=0,0096, dan *error*=12,6532%. Indeks statistik menggunakan metode JST Backpropagation diperoleh MAE=0,2129, RMSE=0,2752, dan *error*=13,3217%. Dari perbandingan indeks statistik ini bisa disimpulkan bahwa metode JST RBF lebih baik dalam memprediksi jumlah mahasiswa baru.

3. Dengan menggunakan data jumlah mahasiswa baru Universitas Wijaya Putra periode tahun 2012-2016 sebagai data training dan data testing, diperoleh prediksi jumlah mahasiswa baru menggunakan metode JST RBF diperoleh, Surabaya=450, Gresik=188, Lamongan=27, Mojokerto=52, Pasuruan=128, dan Kabupaten/Kota Lain=188. Menggunakan metode JST Backpropagation hasil prediksi; Surabaya=575, Gresik=6, Lamongan=48, Mojokerto=385, Pasuruan=436, dan Kabupaten/Kota Lain=206.

(Sumber : Alven Safik Ritonga dan Suryo Atmojo, 2018)

➤ **Perbandingan Metode JST-RBF dengan *Multilayer Perceptron* (MLP)**

Untuk mengetahui algoritma mana yang memiliki akurasi yang lebih tinggi dalam analisa risiko kredit mobil, digunakan metode *Confusion Matrix* dan *Kurva ROC* dengan hasil komparasi yang terlihat pada tabel 2.2:

Tabel 2. 2 Hasil Perbandingan

| | MLP | RBF |
|-------------------------|------------|------------|
| <i>Confusion Matrix</i> | 0,961 | 0,892 |
| <i>Kurva ROC(AUC)</i> | 0,999 | 0,947 |

Tabel hasil perbandingan di atas memperlihatkan bahwa model algoritma *neural network* model *multilayer perceptron* memiliki tingkat keakuratan yang lebih tinggi dibandingkan *neural network* model *Radial basis function*, baik dengan menggunakan metode *confusion matrix* maupun *kurva ROC*.

Kesimpulan yang dapat diambil berdasarkan penelitian ini adalah bahwa :

1. *Performance* model *neural network radial basis function* untuk analisa risiko kredit mobil memberikan tingkat akurasi kebenaran sebesar 89,2% dengan nilai *area under the curva* (AUC) sebesar 0,947. Sedangkan *performance* model *neural network multilayer perceptron* memberikan tingkat akurasi kebenaran sebesar 96,1% dengan nilai *area under the curve* (AUC) sebesar 0,999. Hal ini menunjukkan bahwa kedua model tersebut termasuk katagori klasifikasi sangat baik karena memiliki nilai AUC antara 0.90-1.00.

2. Berdasarkan tingkat akurasi dan nilai *area under the curve* (AUC), maka *performance* metode *neural network model multilayer perceptron* lebih baik dibandingkan model *radial basis function*.

(Sumber : Amrin, 2018)

➤ **Perbandingan Metode Regresi Logistik dan JST-RBF**

Pemilihan metode terbaik antara *regresi logistik* dengan jaringan saraf fungsi *radial basis function* dilakukan dengan melihat ketepatan klasifikasi pada masing-masing kategori variabel respon. Berdasarkan percobaan yang sudah dilakukan diketahui bahwa dengan menggunakan metode *regresi logistik*, ketepatan klasifikasi data *data testing* adalah sebesar 38 persen sedangkan ketepatan klasifikasi data *testing* pada metode RBF adalah sebesar 67,08 persen. Berdasarkan perhitungan yang sudah dilakukan bisa kita simpulkan bahwa metode RBF memberikan tingkat akurasi yang lebih baik daripada metode *regresi logistik ordinal*. Untuk lebih meyakinkan metode mana yang lebih baik digunakan, maka dilakukan *10-fold cross validasi*.

Tabel 2. 3 Hasil *10-Fold Cross Validasi* Pada Data *Testing*

| Percobaan | <i>Regresi Logistik</i> | RBF |
|-----------|-------------------------|-------|
| 1 | 25,50 | 68,19 |
| 2 | 31,00 | 67,92 |
| 3 | 37,50 | 69,59 |
| 4 | 34,00 | 67,19 |
| 5 | 38,00 | 67,09 |

| Keterangan : Terusan dari tabel 2.3 | | |
|-------------------------------------|------------------|--------------|
| Percobaan | Regresi Logistik | RBF |
| 6 | 27,50 | 76,74 |
| 7 | 34,00 | 69,19 |
| 8 | 35,50 | 67,03 |
| 9 | 36,50 | 67,38 |
| 10 | 35,50 | 67,33 |
| Rata-rata | 33,50 | 68,77 |

Pada tabel 2.3 terlihat bahwa dengan adanya sepuluh kali percobaan, pada setiap percobaan menunjukkan bahwa tingkat akurasi metode *radial basis function* lebih baik daripada metode *regresi logistik ordinal*. Hasil dari 10 *fold-validasi* menunjukkan rata-rata tingkat akurasi metode *regresi logistik ordinal* sebesar 33,50 persen dan metode *radial basis function* sebesar 68,77 persen. Berdasarkan simulasi perhitungan di atas bisa disimpulkan bahwa metode *radial basis function* dengan pendekatan *k-mean cluster* lebih baik daripada metode *regresi logistik ordinal*.

Dari penelitian ini bisa ditarik beberapa kesimpulan sebagai berikut :

1. Dengan metode *regresi logistik ordinal* diketahui variabel yang berpengaruh terhadap respon yaitu variabel X1, X2, X3, X4_2, X5_1, X8_1, X8_4, X8_6, X8_7, X8_8, X8_9, X8_10, X9_2, X9_3, X10, X11, dan X12.
2. Permodelan menggunakan metode *radial basis function* dengan pendekatan *k-mean cluster* memberikan tingkat akurasi yang lebih baik dari pada metode *regresi logistik ordinal* dimana ketepatan klasifikasi pada data *testing* dengan

metode RBF sebesar 67.08 dan ketepatan klasifikasi metode *regresi logistik ordinal* sebesar 38 persen.

3. Dengan 10 – *fold cross validasi* diperoleh hasil rata-rata tingkat akurasi metode *regresi logistik ordinal* sebesar 33,50 persen sedangkan tingkat akurasi metode *radial basis function* dengan pendekatan k-mean *cluster* adalah sebesar 68,77 persen. Berdasarkan perhitungan ini bisa diambil kesimpulan bahwa metode *radial basis function* dengan pendekatan k-mean *cluster* merupakan metode yang lebih baik dibandingkan *regresi logistik ordinal*.

(Sumber : Yenita Mirawanti dan Brodjol Sutijo Supri Ulama, 2008)

2.3 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) atau *Decision Support System* (DSS) adalah sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi semi terstruktur dan tak terstruktur. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat.

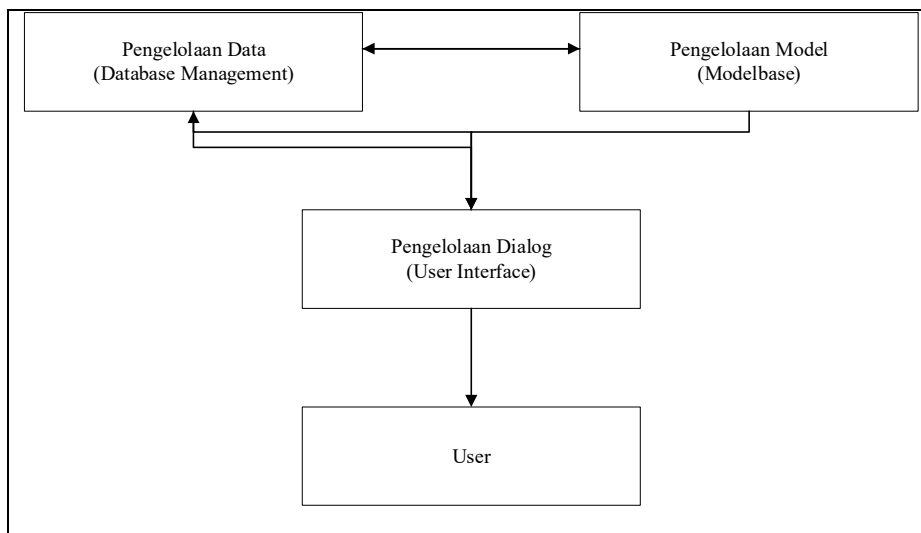
Sistem Pendukung Keputusan (SPK) bertujuan untuk menyediakan informasi, membimbing, memberikan prediksi serta mengarahkan kepada pengguna informasi agar dapat melakukan pengambilan keputusan dengan lebih baik.

Sistem Pendukung Keputusan (SPK) merupakan implementasi dari teori – teori pengambilan keputusan yang telah diperkenalkan oleh ilmu – ilmu seperti *operator research* dan *management science*, dengan perbedaan bahwa

jika dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual (biasanya untuk mencari nilai minimum, optimum, atau optimum), saat ini sistem yang terkomputerisasi telah menawarkan kemampuannya untuk menyelesaikan persoalan yang sama dalam waktu yang relatif singkat.

(Menurut Sprague dan Watson) mendefinisikan Sistem Pendukung Keputusan (SPK) sebagai sistem yang memiliki lima karakteristik, yaitu : sistem yang berbasis komputer, dipergunakan untuk membantu para pengambil keputusan, untuk memecahkan masalah – masalah rumit yang mustahil dilakukan dengan kalkulasi manual, melalui cara simulasi yang interaktif, serta data dan model analisis sebagai komponen utama.

Komponen Sistem Pendukung Keputusan (SPK) secara umum, dibangun oleh tiga komponen besar, yaitu : database management, model *base* dan *software system/user interface*. Komponen Sistem Pendukung Keputusan (SPK) tersebut dapat digambarkan seperti gambar 2.1 dibawah ini



Gambar 2. 1 Komponen Sistem Pendukung Keputusan (SPK)

Keterangan dari *flowchart* gambar 2.1 diatas :

- *Database Management* merupakan subsistem data yang terorganisasi dalam suatu basis data. Data yang merupakan suatu Sistem Pendukung Keputusan (SPK) dapat berasal dari luar maupun dalam lingkungan. Untuk keperluan Sistem Pendukung Keputusan (SPK), diperlukan data yang relevan dengan permasalahan yang hendak dipecahkan melalui simulasi.
- *Modelbase* merupakan suatu model yang mempresentasikan permasalahan kedalam format kuantitatif sebagai dasar simulasi atau pengambilan keputusan, termasuk didalamnya tujuan dari permasalahan, komponen – komponen terkait, batasan – batasan yang ada, dan hal – hal yang terkait lainnya. *Modelbase* memungkinkan pengambilan keputusan menganalisa secara utuh dengan mengembangkan dan membandingkan solusi alternatif.
- *User Interface/Pengelolaan Dialog* merupakan penggabungan antara dua komponen sebelumnya yaitu *Database Management* dan *Modelbase* yang disatukan dalam komponen ketiga (*User Interface*), setelah sebelumnya dipresentasikan dalam bentuk model yang dimengerti komputer. *User Interface* menampilkan keluaran sistem bagi pemakai dan menerima masukan dari pemakai kedalam Sistem Pendukung Keputusan (SPK).

Sistem Pendukung Keputusan (SPK) dapat memberikan berbagai manfaat dan keuntungan, diantaranya adalah untuk memperluas kemampuan pengambilan keputusan dalam memproses data, untuk membantu pengambil keputusan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur, dapat menghasilkan solusi dengan cepat serta hasilnya dapat diandalkan, dapat menjadi stimulan bagi pengambil keputusan dalam memahami persoalannya, karena mampu menyajikan berbagai macam alternatif.

2.4 Weighted Aggregated Sum Product Assessment (WASPAS)

Metode *Weighted Aggregated Sum Product Assessment* (WASPAS) adalah penentuan atau pencarian pilihan terbaik serta mengurangi kesalahan-kesalahan atau mengoptimalkan dalam penafsiran untuk pemelihan nilai tertinggi dan terendah. Penerapan metode WASPAS merupakan kombinasi unik dua unsur yang dikenal sebagai *MCDM approaches*, WMM, dan model produk berat (WPM) memerlukan normalisasi linier dari elemen hasil. Dengan metode WASPAS kriteria kombinasi optimum dicari berdasarkan dua kriteria optimum. Kriteria pertama, kriteria keberhasilan rata-rata tertimbang sama dengan metode WSM. Perhitungan ini adalah pendekatan yang populer dan diadopsi untuk MCDM untuk mengevaluasi beberapa alternatif dalam beberapa kriteria keputusan. Untuk membenarkan ketetapan penetapan dan ketetapan pendekatan MCDM yang hampir baru, yaitu metode penilaian jumlah agregat berbobot (WASPAS).

(Sumber : Ahmad Zulfandi , Dian Anggara , Liza Handayani, 2018)

Berikut merupakan langkah-langkah kerja dari metode WASPAS, yaitu:

1. Menentukan *Normalisasi* Matriks

$$R = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix}$$

2. *Normalisasi* nilai R_{ij} , Jika nilai maksimal dan minimal telah ditentukan maka persamaan sebagai berikut :

Untuk kriteria *Benefit/Tinggi* :

$$a) R_{ij} = \frac{x_{ij}}{\max x_{ij}} \dots \dots \dots (1)$$

Untuk kriteria *Cost/Rendah* :

$$b) R_{ij} = \frac{\min x_{ij}}{x_{ij}} \dots \dots \dots (2)$$

3. Menghitung nilai Alternatif (Q_i), nilai *normalisasi Matrix* dan Bobot WASPAS dengan menggunakan rumus sebagai berikut :

$$Q_i = 0,5 \sum_{j=1}^n R_{ij} W_j + 0,5 \prod_{j=1}^n (R_{ij})^{w_j} \dots \dots \dots (3)$$

Nilai Q_i yang terbaik merupakan nilai tertinggi.

2.5 RGB (*Red, Green, Blue*) dan *Grayscale*

Red (Merah), *Green* (Hijau) dan *Blue* (Biru) merupakan warna dasar yang dapat diterima oleh mata manusia. Setiap *piksel* pada citra warna mewakili warna yang merupakan kombinasi dari ketiga warna dasar RGB. Setiap titik pada citra warna membutuhkan data sebesar 3 *byte*. Setiap warna dasar memiliki intensitas tersendiri dengan nilai minimum nol (0) dan nilai

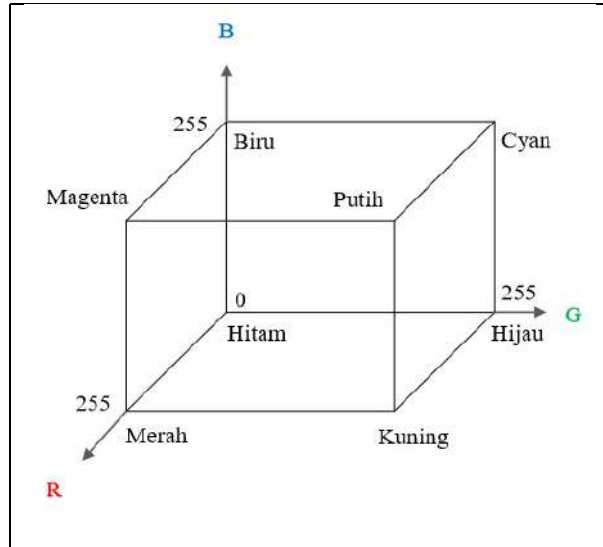
maksimum 255 (8 bit). RGB didasarkan pada teori bahwa mata manusia peka terhadap panjang gelombang 630nm (merah), 530 nm (hijau), dan 450 nm (biru).

Representasi warna di atas dapat diambil beberapa kesimpulan yaitu, RGB terdiri dari tiga warna utama, yaitu merah, hijau, dan biru. Campuran dua warna pada RGB menghasilkan warna baru, yaitu kuning = merah + hijau, cyan = hijau + biru, dan magenta = biru + merah. Bila seluruh warna merah, hijau, dan biru dicampur akan menghasilkan warna putih. Bila warna merah, hijau, dan biru tidak dicampur maka akan menghasilkan warna hitam. Jenis warna lain akan dihasilkan oleh variasi campuran warna dan intensitas campuran setiap warna. Tabel 2.1 dibawah ini menunjukkan contoh warna dan nilai R, G, dan B.

Tabel 2. 4 Warna Dan Nilai R, G, dan B

| Warna | R | G | B |
|--------|-----|-----|-----|
| Merah | 255 | 0 | 0 |
| Hijau | 0 | 255 | 0 |
| Biru | 0 | 0 | 255 |
| Hitam | 0 | 0 | 0 |
| Putih | 255 | 255 | 255 |
| Kuning | 0 | 255 | 255 |

Warna RGB juga bisa dilihat pada gambar 2.2 dibawah ini, pada warna RGB ini berdimensi 3.



Gambar 2. 2 Warna RGB Dalam Dimensi 3

Pada gambar 2.3 dibawah ini merupakan citra RGB, yaitu gambar citra warna dan representasi warnanya, dimana yang disetiap piksel – piksel dinyatakan dengan nilai masing – masing R, G, dan B.



Gambar 2. 3 Citra Dan Representasi RGB

Citra *grayscale* merupakan citra *digital* yang hanya memiliki satu nilai kanal pada setiap *pikselnya*, artinya nilai dari $Red = Green = Blue$. Nilai-nilai tersebut digunakan untuk menunjukkan intensitas warna.

Citra yang ditampilkan dari citra jenis ini terdiri atas warna abu-abu, bervariasi pada warna hitam pada bagian yang intensitas terlemah dan warna putih pada intensitas terkuat. Citra *grayscale* berbeda dengan citra "hitam-putih", dimana pada konteks komputer, citra hitam putih hanya terdiri atas 2 warna saja yaitu "hitam" dan "putih" saja. Pada citra *grayscale* warna bervariasi antara hitam dan putih, tetapi variasi warna diantaranya sangat banyak. Citra *grayscale* seringkali merupakan perhitungan dari intensitas cahaya pada setiap *piksel* pada *spektrum elektromagnetik single band*.

Citra *grayscale* disimpan dalam format 8 *bit* untuk setiap sampel piksel, yang memungkinkan sebanyak 256 intensitas. Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing R, G dan B menjadi citra *grayscale* dengan nilai X, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai R, G dan B sehingga dapat dituliskan menjadi:

$$X = (R+G+B)/3$$

$$\text{Warna} = \text{RGB}(X, X, X)$$

Citra *biner* (*binary image*) adalah citra *digital* yang hanya memiliki 2 kemungkinan warna, yaitu hitam dan putih. Citra *biner* disebut juga dengan citra W&B (*White & Black*) atau citra *monokrom*. Hanya dibutuhkan 1 *bit* untuk mewakili nilai setiap piksel dari citra *biner*.

Pembentukan citra *biner* memerlukan nilai batas keabuan yang akan digunakan sebagai nilai patokan. Piksel dengan derajat keabuan lebih besar dari nilai batas akan diberi nilai 1 dan sebaliknya piksel dengan derajat keabuan lebih kecil dari nilai batas akan diberi nilai 0.

Citra *biner* sering sekali muncul sebagai hasil dari proses pengolahan, seperti segmentasi, pengembangan, *morfologi* ataupun *dithering*. Fungsi dari *binerisasi* sendiri adalah untuk mempermudah proses pengenalan pola, karena pola akan lebih mudah terdeteksi pada citra yang mengandung lebih sedikit warna.

2.6 Gray Level Co-occurrence Matrix (GLCM)

Ekstraksi fitur merupakan bagian *fundamental* atau terpenting dalam analisis suatu citra. Fitur sendiri merupakan karakteristik unik dari suatu objek yang diamati. Salah satu metode ekstraksi fitur adalah *Gray Level Co-occurrence Matrix* (GLCM) termasuk dalam metode statistik yang menguji tekstur dengan mempertimbangkan hubungan spasial dari tiap piksel. *Matrix co-occurrence* adalah suatu matriks yang dibangun dengan menggunakan *histogram* tingkat kedua. *Matrix co-occurrence* adalah matriks yang berukuran $L \times L$, dimana L merupakan banyaknya tingkat keabuan pada suatu citra *grayscale* dengan elemen-elemen $P(x_1, x_2)$ yang merupakan distribusi *probabilitas* bersama dari pasangan *piksel* dengan tingkat keabuan x_1 yang berlokasi pada koordinat (j,k) dengan x_2 yang berlokasi pada koordinat (m,n) .

GLCM merupakan metode ekstraksi ciri yang menggunakan perhitungan tekstur pada orde kedua yaitu memperhitungkan pasangan dua piksel citra asli, sedangkan pada orde pertama menggunakan perhitungan

statistik berdasarkan nilai piksel citra asli dan tidak memperhatikan piksel ketetangaan.

Berdasarkan orde statistiknya, dapat dikelompokkan menjadi 3 (tiga), yaitu :

1. Statistik orde-kesatu

Pada orde-kesatu ini merupakan metode pengambilan ciri yang didasarkan pada karakteristik *histogram* suatu citra. *Histogram* menunjukkan *probabilitas* kemunculan nilai dari derajat keabuan piksel dalam suatu citra dengan mengabaikan hubungan antar piksel tetangga. Analisa tekstur dengan menggunakan orde pertama ini lebih baik dalam merepresentasikan tekstur citra dalam parameter seperti *mean*, *skewnes*, *variance*, *kurtosis* dan *entropy*.

2. Statistik orde-kedua

Pada orde-kedua ini mempertimbangkan hubungan antara dua piksel yang bertetangaan pada sebuah citra. Dalam kinerjanya, orde-kedua ini memerlukan bantuan *matrix co-occurrence* untuk citra keabuan, biasanya disebut dengan GLCM (*Gray Level Co-occurrence Matrix*). Dalam metode GLCM terdapat 14 fitur yang dapat di ekstraksi untuk mendapatkan ciri dari suatu citra yang diamati. Analisa tekstur dengan menggunakan orde-kedua ini lebih baik dalam merepresentasikan tekstur citra dalam parameter seperti *contrast*, *correlation*, *homogeneity* dan *energy*.

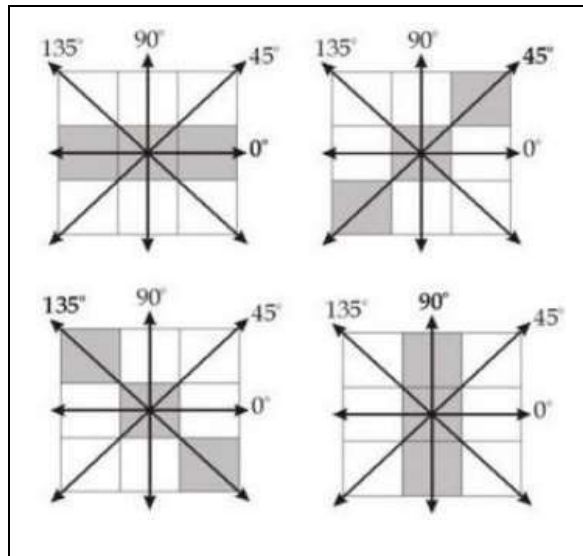
3. Statistik orde-ketiga

Dalam orde-ketiga ini mempertimbangkan hubungan antara tiga atau lebih piksel, secara teoritis memungkinkan akan tetapi tidak secara umum, karena akan sulit diterapkan.

Pada beberapa kasus, ciri orde pertama tidak lagi dapat digunakan untuk mengenali perbedaan antar citra. Pada kasus seperti ini, membutuhkan pengambilan ciri statistik orde dua. Salah satu teknik untuk memperoleh ciri statistik orde kedua 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.

Gray Level Co-occurrence Matrix (GLCM) adalah suatu matriks yang elemen-elemennya merupakan jumlah dari pasangan piksel yang memiliki tingkat kecerahan tertentu, dimana pasangan piksel tersebut terpisah dengan jarak (d) dan sudut (θ). GLCM merupakan metode yang digunakan untuk mendapatkan suatu ciri dari sebuah citra yang nantinya hasil dari pencarian ciri dari suatu citra tersebut dapat dijadikan sebuah inputan untuk mengklasifikasikan citra tersebut kedalam kelompok atau kelas-kelas tertentu yang sudah disepakati sebelumnya. Matriks *ko-okurensi* merupakan matriks yang akan digunakan untuk melakukan proses perhitungan pada metode GLCM. Ada empat sudut atau arah (lihat Gambar 2.4) yang digunakan dalam metode GLCM untuk menentukan hubungan antar *piksel* yang mempunyai pola ketetanggaan dalam suatu citra digital, yaitu 0° , 45° , 90° , 135° .

Sedangkan untuk jarak antar *piksel* biasanya ditetapkan sebesar 1 piksel, 2 piksel, 3 piksel, dan seterusnya.



Gambar 2. 4 Ilustrasi Sudut Dalam Metode GLCM

Langkah-langkah untuk mendapatkan fitur ekstraksi ciri dalam metode GLCM, ialah :

1. *Normalisasi* suatu citra dengan mengubah rentang nilai intensitas *pixel*
2. Membuat area kerja matriks.
3. Menentukan hubungan spasial antara *piksel* referensi dengan *piksel* tetangga dengan memberikan nilai untuk sudut (θ), dan jarak (d).
4. Menghitung jumlah *kookurensi* matriks dan mengisikannya pada area kerja matriks.
5. Menjumlahkan matriks *kookurensi* dengan matriks *transposenya* untuk menjadikan matriks yang simetris.
6. *Normalisasi* matriks untuk mengubahnya ke bentuk *probabilitas*.
7. Setelah mendapatkan matriks *kookurensi* tersebut, kemudian dihitung analisa tekstur dengan menggunakan orde-kedua dengan menggunakan parameter seperti Korelasi, Kontras, Homogenitas, dan Energi.

2.7 Jaringan Saraf Tiruan

Kecerdasan buatan dapat memungkinkan komputer untuk “berpikir”. Dengan cara menyederhanakan program, kecerdasan buatan dapat menirukan proses belajar manusia sehingga informasi baru dapat diserap dan digunakan sebagai acuan untuk menyelesaikan masalah-masalah yang akan terjadi. Salah satu teknik dalam kecerdasan buatan yang banyak digunakan adalah Jaringan Saraf Tiruan. Seperti namanya, Jaringan Saraf Tiruan (JST) adalah tiruan dari jaringan saraf manusia. Yang ditiru tentu saja adalah cara kerjanya, yaitu menyampaikan input melalui *neuron-neuron* yang membentuk jaringan untuk akhirnya disampaikan menjadi *output*.

Jaringan saraf tiruan (JST) (*artificial neural network* (ANN)), atau juga disebut *simulated neural network* (SNN), atau umumnya hanya disebut *neural network* (NN)), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan saraf manusia / jaringan saraf biologis (JSB). JST dapat digunakan untuk memodelkan hubungan yang kompleks antara *input* dan *output* untuk menemukan pola-pola pada data.

Jaringan Saraf Tiruan adalah paradigma pemrosesan suatu informasi yang terinspirasi oleh sistem sel saraf biologi, sama seperti otak yang memproses suatu informasi. Elemen mendasar dari paradigma tersebut adalah struktur yang baru dari sistem pemrosesan informasi. Jaringan Saraf Tiruan, seperti manusia, belajar dari suatu contoh. Jaringan Saraf Tiruan dibentuk untuk memecahkan suatu masalah tertentu seperti pengenalan pola atau klasifikasi karena proses pembelajaran.

Jaringan Saraf Tiruan berkembang secara pesat pada beberapa tahun terakhir. Jaringan Saraf Tiruan telah dikembangkan sebelum adanya suatu komputer *konvensional* yang canggih dan terus berkembang walaupun pernah mengalami masa vakum selama beberapa tahun. Salah satu contoh metode jaringan saraf tiruan yang dikenal adalah metode *radial basis function* (RBF). Jaringan saraf tiruan *radial basis function* (JSTRBF) dikenal sebagai salah satu jaringan saraf yang memiliki tiga lapis bersifat *feedforward* yang dapat memecahkan masalah klasifikasi atau pengenalan pola. Jaringan saraf tiruan merupakan salah satu *representasi* buatan dari otak manusia dengan menggunakan program komputer yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut.

Radial Basis Function adalah model *neural network* yang mentransformasi *input* secara *nonlinear* dengan menggunakan fungsi aktivasi *Gaussian* pada lapisan unit *hidden* sebelum diproses *linear* pada lapisan *output*. Berguna atau setidaknya suatu jaringan saraf tiruan ditentukan dari hasil pelatihannya yang berupa bobot *neuronnya*. *Radial Basis Function* memiliki algoritma pelatihan yang agak unik karena terdiri atas metode *supervised* dan *unsupervised* sekaligus, dimana metode *supervised* merupakan metode yang setiap pola diberikan ke dalam jaringannya telah diketahui *outputnya*. Selisih antara pola *output* aktual (*output* yang dihasilkan) dengan pola *output* yang dikehendaki (*output target*) yang disebut *error* digunakan untuk mengoreksi bobot jaringan. Sedangkan metode *unsupervised* adalah metode yang tidak membutuhkan

target *output*. Pada metode ini tidak dapat ditentukan hasil seperti apakah yang diharapkan selama proses pembelajaran. Kombinasi antara 2 jenis metode ini pada jaringan *Radial Basis Function* menghasilkan suatu sistem yang handal dalam mengatasi ketidak*linearan* sistem itu sendiri.

Setiap *input* dari jaringan ini akan mengaktifkan semua fungsi aktivasi pada *hidden layer*. Setiap unit dari *hidden layer* merupakan fungsi *basis*. Di dalam *hidden layer* terdapat sejumlah fungsi *basis* yang sejenis sesuai dengan perancangan. Setiap fungsi basis akan menghasilkan sebuah keluaran dengan bobot tertentu. Hal yang bersifat khusus pada Jaringan Saraf Tiruan RBF ialah sebagai berikut :

1. Pemrosesan sinyal dari *input layer* ke *hidden*, sifatnya *nonlinier*, sedangkan dari *hidden layer* ke *output layer* sifatnya *linier*.
2. Pada *hidden layer* digunakan sebuah fungsi aktivasi berbasis *radial*, misalnya fungsi *Gaussian*.
3. Pada *output* unit, sinyal dijumlahkan dengan metode *ordinary least square*.
4. Sifat jaringannya ialah satu arah, tidak seperti pada jaringan *feedforward neural network* (FFNN) yang bersifat bolak-balik.

Karakteristik dari Jaringan Saraf Tiruan adalah :

- Pola hubungan antar *neuron* (yang menjadi arsitekturnya).
- Metode penentuan bobot dalam koneksi (disebut sebagai proses latihan, pembelajaran, atau Algoritma).
- Fungsi aktivasi.

2.7.1 Cara Kerja Jaringan Saraf Tiruan

Cara kerja jaringan saraf tiruan adalah sebagai berikut :

- Informasi (*input*) akan dikirim *neuron* dengan bobot kedatangan tertentu.
- *Input* ini akan diproses suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang.
- Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap *neuron*.
- Apabila input tersebut melewati suatu nilai ambang tertentu, maka *neuron* tersebut akan diaktifkan, tetapi kalau tidak, *neuron* tersebut tidak akan diaktifkan.
- Apabila *neuron* tersebut diaktifkan, maka *neuron* tersebut akan mengirimkan output melalui bobot-bobot outputnya ke semua *neuron* yang berhubungan dengannya.

Neuron Layer

- Pada jaringan saraf, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan *neuron* (*neuron layers*).
- Biasanya *neuron-neuron* pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya (kecuali lapisan input dan lapisan *output*).
- Informasi yang diberikan pada jaringan saraf akan dirambatkan dari lapisan ke lapisan, mulai dari lapisan input sampai ke lapisan *output* melalui lapisan yang lainnya, yang sering dikenal dengan nama lapisan tersembunyi (*hidden layer*).

- Umumnya *neuron-neuron* yang terletak pada lapisan yang sama akan memiliki keadaan yang sama sehingga pada setiap lapisan yang sama *neuron-neuron* memiliki fungsi aktivasi yang sama.
- Bila *neuron-neuron* pada suatu lapisan (misal lapisan tersembunyi) akan dihubungkan dengan *neuron-neuron* pada lapisan lain (misal lapisan *output*) maka setiap *neuron* pada lapisan tersebut (lapisan tersembunyi) juga harus dihubungkan dengan setiap *neuron* pada lapisan lainnya (lapisan *output*).
- Faktor terpenting dalam menentukan kelakuan suatu *neuron* adalah fungsi aktivasi dan pola bobotnya.

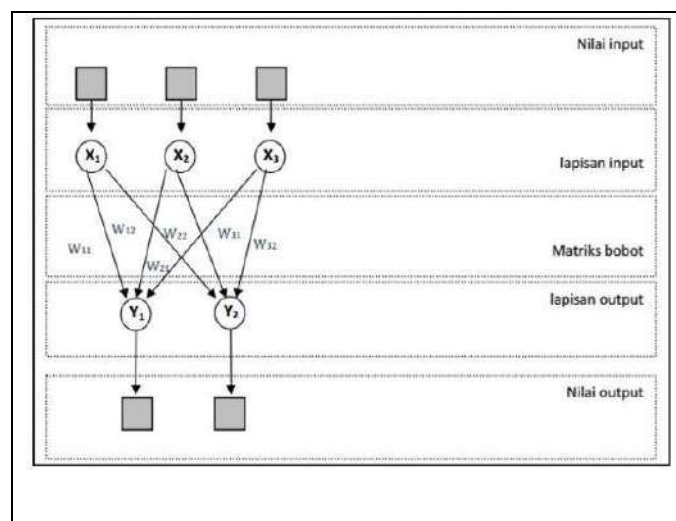
2.7.2 Arsitektur Jaringan Saraf Tiruan

Ada beberapa bentuk arsitektur jaringan saraf yaitu :

1. Jaringan dengan lapisan tunggal (*single layer net*)

Jaringan ini hanya memiliki satu lapisan dengan bobot–bobot terhubung.

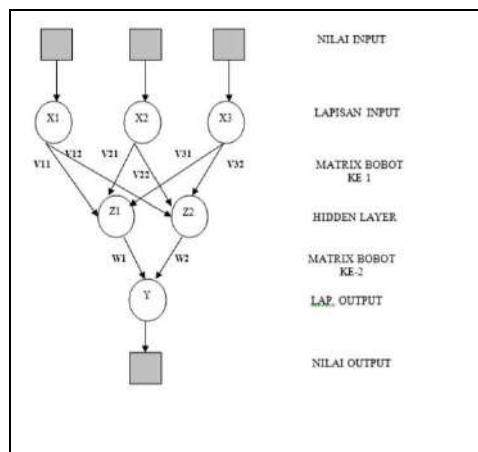
Jaringan ini menerima input yang kemudian secara langsung akan mengolahnya menjadi output tanpa harus melalui lapisan tersembunyi.



Gambar 2. 5 Jaringan dengan lapisan tunggal (*single layer net*)

2. Jaringan dengan banyak lapisan (*multilayer*)

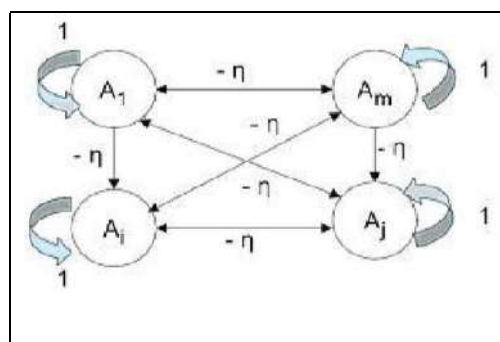
Jaringan ini memiliki satu atau lebih lapisan yang terletak diantara lapisan input dan lapisan *output*nya (memiliki satu atau lebih lapisan tersembunyi). Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit dari pada jaringan dengan lapisan tunggal, tetapi pembelajarannya lebih rumit. Jaringan ini pada banyak kasus lebih sukses dalam menyelesaikan masalah.



Gambar 2. 6 Jaringan dengan banyak lapisan (*multilayer*)

3. Jaringan dengan lapisan *kompetitif* (*competitive layer net*)

Pada jaringan ini sekumpulan *neuron* bersaing untuk mendapatkan hak menjadi aktif. Gambar berikut menunjukkan salah satu contoh arsitektur jaringan dengan lapisan *kompetitif* yang memiliki bobot $-\eta$



Gambar 2. 7 Jaringan dengan lapisan kompetitif (*competitive layer net*)

2.7.3 Proses Pembelajaran

Perubahan yang terjadi selama pembelajaran adalah perubahan nilai bobot. Nilai bobot akan bertambah, jika informasi yang diberikan oleh *neuron* yang bersangkutan tersampaikan, sebaliknya jika tidak disampaikan oleh suatu *neuron* ke *neuron* yang lain, maka nilai bobot yang menghubungkan keduanya akan dikurangi.

Pada saat pembelajaran dilakukan pada *input* yang berbeda, maka nilai bobot akan diubah secara dinamis hingga mencapai nilai yang cukup seimbang. Apabila nilai ini telah tercapai yang mengindikasikan bahwa

Tiap-tiap *input* telah berhubungan dengan *output* yang diharapkan. Ada beberapa metode untuk proses pembelajaran pada jaringan saraf tiruan ini, diantaranya

A. Pembelajaran Terawasi (*Supervised Learning*)

- *Output* telah diketahui sebelumnya.
- Satu pola *input* akan diberikan ke satu *neuron* pada lapisan *input*.
- Pola dirambatkan disepanjang jaringan saraf hingga sampai ke *neuron* pada lapisan *output*.
- Lapisan *output* ini akan membangkitkan pola *output* yang nantinya akan dicocokkan dengan pola *output* targetnya.
- Apabila terjadi perbedaan antara *output* hasil pembelajaran dengan pola target, maka disini akan muncul *error*.
- Apabila *error* ini masih cukup besar, mengindikasikan bahwa masih perlu dilakukan lebih banyak pembelajaran lagi.

B. Pembelajaran Tak Terawasi (*Unsupervised Learning*)

- Tidak memerlukan target *output*.
- Tidak dapat ditentukan hasil seperti apakah yang diharapkan selama proses pembelajaran.
- Selama proses pembelajaran, nilai bobot disusun dalam suatu range tertentu tergantung pada nilai *input* yang diberikan.
- Tujuan pembelajaran ini adalah mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu.
- Pembelajaran ini biasanya sangat cocok untuk pengelompokan (klasifikasi) pola.

C. Gabungan pembelajaran terawasi dan tak terawasi (*hybrid*)

- Merupakan kombinasi dari kedua pembelajaran tersebut.
- Sebagian dari bobot-bobotnya ditentukan melalui pembelajaran terawasi dan sebagian lainnya melalui pembelajaran tak terawasi.