KLASIFIKASI PENYAKIT IKAN DI KARANTINA KELAUTAN DAN PERIKANAN BADAN KARANTINA IKAN, PENGENDALIAN MUTU DAN KEAMANAN HASIL PERIKANAN (BKIPM-KKP) KELAS II BANDUNG MENGGUNAKAN HIDDEN MARKOV MODEL

Salman Muhammad[1], Nelly Indriani W.[2]

[1,2] Program Studi Teknik Informatika
Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia
Jl. Dipati Ukur No. 112-116 Bandung
E-mail : miftah.faris07@yahoo.co.id[1], indiwidi@gmail.com[2]

ABSTRAK

Ikan merupakan salah satu hewan air yang selalu bersentuhan dengan lingkungan perairan sehingga mudah terinfeksi penyakit baik itu bertipe patogen maupun non patogen melalui air. Ada beberapa faktor yang sering mengakibatkan ikan mengalami kematian yakni faktor *internal* dan *external*. BKIPM-KKP sering melalukan tinjauan terhadap penangkaran-penangkaran yang ada di wilayah Jawa Barat, dan terkadang pada saat tinjauan terhadap penangkaran tersebut mengalami kesulitan pada saat mengklasifikasikan penyakit yang diderita ikan yang ada di penangkaran, karena untuk mengetahui penyakit tersebut harus melalui tahap laboratorium BKIPM-KKP terlebih dahulu dan hasil dari laboratorium tidak akan keluar pada saat itu juga untuk proses klasifikasi. Berdasarkan dari penelitian melalui observasi terhadap laboratorium BKIPM-KKP, wawacara kepada stasiun dan pihak laboratorium, dan pengumpulan data yang dibutuhkan untuk menyelesaikan masalah yang sedang terjadi, serta beberapa jurnal dan sumber lainnya yang berkaitan tentang klasifikasi menggunakan *Hidden Markov Model* (HMM).

Citra yang digunakan merupakan citra hasil dari mikroskopis baik bertipe .jpg maupun .png, dimana citra tersebut akan diproses menggunakan pengolahan citra mulai dari *grayscale, edge detection canny* dan ekstraksi, setelah citra melalui tahap ekstraksi maka tahap selanjutnya yaitu proses klasifikasi menggunakan *Hidden Markov Model (HMM)*

Setelah melakukan pengujian dengan citra yang didapat dari hasil mikroskopis menggunakan metode *Hidden Markov Model (HMM)* bahwa menggunakan proses pengolahan citra saja sudah cukup untuk menentukan ciri dari citra penyakit ikan, sehingga apabila hanya menggunakan prose pengolahan citra saja, keakuratan yang didapat hanya 69% dari 100 data citra yang diuji dalam proses klasifikasi

Kata kunci : Klasifikasi penyakit ikan, Penyakit, Ikan, *Hidden Markov models*, Image Processing, BKIPM-KKP, Laboratorium, Mikroskopis.

1. PENDAHULUAN

Pada bagian ini akan dipaparkan mengenai latar belakang masalah, maksud dan tujuan.

1.1 Latar Belakang Masalah

Ikan merupakan salah satu hewan air yang selalu bersentuhan dengan lingkungan perairan sehingga mudah terinfeksi penyakit baik itu bertipe *patogen* maupun *non patogen* melalui air. Pada umumnya penyakit ikan memiliki dua faktor penyebab terjangkitnya penyakit yaitu faktor *internal* dan *external*. Faktor *internal* atau dalam yakni gangguan terhadap air yang menyebabkan sistem kekebalan pada ikan menurun, ada juga gangguan terhadap genetic dan metabolisme. Faktor *eksternal* meupakan gangguan yang berasal dari luar luar yaitu yang bersifat *patogen* dan *non patogen*. Faktor *ekternal* yang bersifat *patogen* [1,2] antara lain virus, parasit, jamur bakteri, dan *protozoa*, faktor yang bersifat *non-pathogen* antara lain suhu, pH, nutrisi, dan kualitas air, biasanya ikan sering terkena penyakit karena faktor *eksternal*. [3]

BKIPM-KKP sering sekali melakukan tinjauan terhadap penangkaran-penangkaran yang ada di wilayah Jawa Barat, dan terkadang pada saat tinjauan terhadap penangkaran tersebut, BKIPM-KKP sering menemukan penangkaran yang mengalami masalah terhadap penyakit ikan, BKIPM-KKP mengalami kesulitan karena untuk mengetahui penyakit tersebut harus dimasukkan kedalam laboratorium BKIPM-KKP dan harus menunggu berhari-hari untuk melakukan proses klasifikasi.

Dalam proses klasifikasi penyakit ikan, ciri yang digunakan adalah gambar dari penyakit ikan yang didapat dari hasil mikroskopis laboratorium, sehingga didapat citra dengan kualitas yang baik

untuk digunakan sebagai data dalam proses klasifikasi dan dari hasil proses klasifikasi penyakit ikan, akan dicari keakuratannya untuk membuktikan proses klasifikasi tersebut. Sehingga hasil dari klasifikasi penyakit ikan dengan keakuratan yang baik, dalam waktu yang cukup cepat dapat menurunkan tingkat penularan penyakit terhadap ekosistem yang sedang diserang oleh penyakit.

Proses klasifikasi penyakit dapat dilakukan dengan berbagai metode. Kecerdasan Buatan (*Artificial Intelligent*). Dalam penelitian ini, metode klasifikasi penyakit dari sampel mikroskopis laboratorium yang digunakan adalah *Hidden Markov Model* (HMM). HMM merupakan metode yang sering digunakan dalam proses *recognition* DNA. Dalam riset terakhir penelitian tentang kesehatan menggunakan *Hidden Markov Model* (HMM) sudah pernah dilakukan yaitu merancang pengenalan penyakit darah [4]. Hasilnya bahwa penyakit darah dapat dikenali dengan menggunakan *Hidden Markov Model* (HMM) dengan keakurasian 60% hingga 82,76%.

Berdasarkan uraian permasalahan diatas, maka pada laporan penelitian tugas akhir ini mengambil topik "Klasifikasi Penyakit Ikan di Kementerian Kelautan dan Perikanan Badan Karantina Ikan, Pengendalian Mutu dan Keamanan Hasil Perikanan (BKIPM-KKP) Kelas II Bandung Menggunakan Hidden Markov Model"

1.2 Maksud dan Tujuan

Maksud dari penelitian ini adalah bagaimana mengimplementasikan metode *Hidden Markov Model* (HMM) dalam proses mengklasifikasikan penyakit ikan dari hasil mikroskopis laboratorium BKIPM-KKP.

Sedangkan tujuan yang ingin dicapai dari penelitian ini adalah untuk mengetahui apakah penyakit ikan dapat diklasifikasikan dengan menggunakan metode *Hidden Markov Model* (HMM) dan mengetahui seberapa besar akurasi dalam proses klasifikasi penyakit ikan studi kasus BKIPM-KKP.

1.3 Patologi Parasi Protozoa

Protozoa adalah organisme unisellular (bersel satu) mikroskopik yang sangat kecil dan memiliki struktur yang kompleks yang digunakan untuk pergerakan, pelekatan, dan pelindungan. Parasit ini memiliki beberapa kelompok yang ada pada ikan. Parasit ini dapat berkembang biak pada tubuh inang. Golongan protozoa setidaknya memiliki 7 phylum yang merupakan parasit pada ikan, yaitu : *Phylum Amoebozoa, Phylum Ciliophora, phylum Apicomplexa, Phylum Microspora, Phylum Myxozoa.*

Diantara golongan parasit air tawar, *Ichthiophthirius multifillis* telah menyebabkan banyak masalah dalam budidaya ikan air tawar. Pada kondisi budidaya, spesies protozoa tertentu dapat menyebabkan penyakit yang menghasilkan mortalitas tinggi yang berdampak pada kerugian ekonomi yang cukup besar pada ikan air tawar maupun ikan air laut. **[**6**]**

1.4 Pengolahan Citra

Pengolahan citra adalah sebuah ilmu yang mempelajari hal-hal yang berkaitan dengan perbaikan kualitas gambar baik itu meningkatkan kontras, transformasi warna, dan restorasi citra, transformasi gambar seperti rotasi, translasi, skala, dan transformasi geometri, proses pemilihan sebuah citra (feature images) yang optimal untuk tujuan analisis, melakukan penarikan informasi atau dekripsi objek atau pengenalan objek yang terkandung dalam citra, melakukan kompresi atau reduksi sebuah data untuk tujuan penyimpanan data, transmisi data, dan waktu proses data. Input dari pengolahan citra adalah citra, sedangkan output-nya adalah citra hasul pengolahan. **[**8**]**

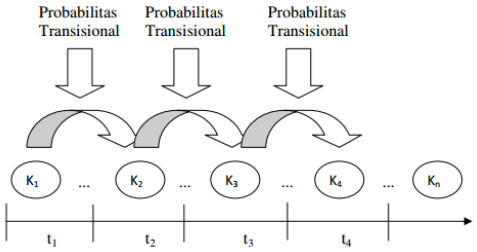
1.5 Ekstraksi

Ektraksi fitur atau featur extraction merupakan suatu pengambilan ciri / feature dari suatu bentuk yang nantinya nilai yang didapatkan akan dianalisis untuk proses selanjutnya. featur extraction dilakukan dengan cara menghitung jumlah titik atau pixel yang ditemukan dalam setiap pengecekan, dimana pengecekan dilakukan dalam berbagai arah tracing pengecekan pada kordinat kartesial dari citra digital yang dianalisis.

1.6 Markov Chain

Rantai Markov *(Markov Chains)* adalah merupakan suatu teknik matematika yang biasa digunakan untuk melakukan pemodelan (modeling) berbagai macam sistem dan proses bisnis. Teknik ini dapat digunakan untuk memperkirakan perubahan-perubahan di waktu yang akan datang dalam variabel-variabel dinamis atas dasar perubahan-perubahan dari variabel-variabel dinamis tersebut di waktu yang lalau. Teknik ini dapat digunakan juga untuk menganalisis kejadia-kejadian di waktu-waktu mendatang secara matematis.

Model Rantai Markov *(Markov Chains)* ditemukan oleh seorang ahli Rusia yang bernama Andrey Markov pada tahun 1906, yaitu: "*Untuk setiap waktu t, ketika kejadian adalah* K_t *dan seluruh kejadian sebelumnya adalah* $K_{t(j)}\dots$, $K_{t(j-n)}$ *yang terjadi dari proses yang diketahui, probabilitas seluruh kejadian yang akan datang Kt(j) hanya bergantung pada kejadian* $K_{t(j-1)}$*dan tidak bergantung pada kejadian-kejadian sebelumnya yaitu* $K_{t(j-2)}$, $K_{t(j-3)}$,..., $K_{t(j-n)}$.*"*.



Gambar 1. Markov Chain

1.7 Hidden Markov Model

Hidden Markov Model (HMM) adalah suatu model probabilitas yang menggambarkan hubungan statistik antara urutan observasi O dan urutan state S yang tidak diobservasi "hidden". [**12**]

Ada 4 procedur HMM yang dapat digunakan didunia nyata, yaitu [10]:

a. Prosedur Maju (*forward procedure*)

Asumsi variabel *forward* $\alpha_t(i)$ pada waktu ke-*t* dan *state* ke-*i* didefinisikan dengan. $\alpha_t(i) = P(O_1,O_2,...,O_T, q_t=i \mid \lambda)$. Fungsi peluang *forward* dapat diselesaikan untuk *N state* dan *T* simbol observasi secara induktif dengan langkah sebagai berikut:

a) Inisialisasi, $\alpha_t(i) = \pi_i b_i(O_1)$,

$1 \le i \le N$ (2.12)

b) Induksi, $\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\alpha_{ij}\right] b_j(Oi+_1)$,

$1 \le t \le T\text{-}1;\ 1 \le j \le N$ (2.13)

c) Terminasi,

$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$ (2.14)

Perhitungan peluang *forward* berdasarkan +pola diagram *trellis*. Terdapat *N* titik tiap *time slot* pada pola, semua kemungkinan deretan *state* digabung sebanyak *N* titik tanpa memperdulikan panjang deretan observasi. Pada saat t=1, dihitung nilai $O_1(i)$, $1 \le i \le N$. Pada waktu t = 2,3,...,*T* hanya diperlukan perhitungan nilai $\alpha_t(j)$ dimana $1 \le j \le N$.

Tiap perhitungan membutuhkan nilai sebelumnya sebanyak *N* dari $\alpha_{t-1}(i)$ karena tiap *N* titik hanya dapat dihubungkan dengan *N* titik pada *time slot* sebelumnya.

b. Prosedur Mundur (*backward procedure*)

Dengan cara yang sama, variabel *backward* $\beta_t(i)$ pada waktu ke-*t* dan *state* ke-*i* didefinisikan dengan $\beta_t(i) = P(O_{t+1}, O_{t+2}, ..., O_T, qt = i \mid \lambda)$. Prosedur *backward* digunakan untuk menghitung $P(O|\lambda)$ secara mundur dari waktu observasi maksimum, $t=T$ sampai waktu observasi awal t=1. Langkah prosedur *backward* adalah sebagai berikut:

a) Inisialisasi, $\beta_T(i) = 1;\ 1 \le i \le N$ (2.15)

b) Induksi, $\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_i + 1)\, \beta_t + 1(j)$, t = *T-1, T-2, ..., 1* (2.16)

Pada inisialisasi didefinisikan $\beta_T(i)$ =1 untuk semua nilai *i*. Untuk mendapatkan *state* ke-*i* waktu ke-*t* dan deretan observasi pada waktu *t*+1, maka diasumsikan semua kemungkinan *state j* pada waktu *t*+1, untuk mendapatkan transisi dari *i* ke *j*, a_{ij} dan deretan observasi o_{t+1} pada *state* ke-*j*, $b_j(O_{t+1})$. Kemudian dihitung bagian observasi dari *state j*, $\beta_{t+1}(j)$.

c. Prosedur Maju – Mundur (*forward – backward procedure*)

Kombinasi dari prosedur *forward* dan *backward* dapat digunakan untuk mendapatkan nilai $P(O|\ \lambda)$. Peluang berada pada *state* ke-*i* waktu ke-*t* dari *N state* sebelumnya waktu *t*-1 dapat dihitung dengan fungsi peluang *forward* $\alpha_t(i)$. Sedangkan peluang transisi dari *state i* waktu ke-*t* menuju *state j* waktu *t*+1 dan mengambil sebuah simbol observasi pada *state j* adalah $a_{ij}b_j(O_{t+1})$. Fungsi peluang *backward* digunakan untuk menghitung peluang munculnya deretan simbol observasi dari waktu *t*+1 sampai *T*.

Secara matematis perhitungan $P(O|\ \lambda)$ menggunakan prosedur *forward-backward* dapat dinyatakan sebagai berikut:

$P(O|\lambda) = \left[\sum_{i=1}^{N}\left[\sum_{j=1}^{N} \alpha_1(i)a_{ij}.\, b_j(O_i + 1)\right]\beta_t + 1(j)\right]$

$P(O|\lambda) = \sum_{i=1}^{N} \alpha_t(i)\, \beta_t(i)$ (2.17)

Keterangan :

$\alpha_t(i)$: merupakan hasil dari observasi prosedur *forward* tahap terminasi.

$\beta_t(i)$: merupakan hasil dari observasi prosedur *backward* tahap terminasi.

d. Prosedur Viterbi (*viterbi procedure*)

Pada algoritma *viterbi*, peluang maksimum berada pada *state* ke-*i* waktu ke-*t*, yang sebelumnya telah menempuh deretan *state* $q=\{q1, q2, ..., qt\text{-}1\}$, jika diberikan deretan observasi *O* dan suatu model λ dinotasikan dengan $\delta_t(i)$.

$\delta t\ (i) = \max P[q_1q_2...q_{t-1},q_t = i,O_1O_2...O_T \mid \lambda]$ (2.19)

Untuk mendapatkan deretan observasi dibutuhkan urutan dari argumen yang maksimal untuk tiap *t* dan *j*. Hal ini dilakukan melalui sebuah fungsi *array* $\psi_t(j)$.Prosedur selengkapnya dari algoritma viterbi adalah sebagai berikut:

1) Inisialisasi, $\delta_1(i) = \pi_i\ b_i\ (o_1),\ 1 \le i \le N;\ \psi_1\ (i) = 0$ (2.20)

2) Rekursi, $\delta_1(j) = \max_{1\le i\le N}\left[\delta_{t-1}(i)a_{ij}\right] b_j(o_t), 2 \le t \le T; 1 \le j \le N$

ψ_t (j) = arg $\max_{1\le i\le N}\left[\delta_{t-1}(i)a_{ij}\right], 2 \le t \le T; 1 \le j \le N$ (2.21)

3) Terminasi, $P^* = \max_{1\le i\le N}[\delta_T(i)]$

$q^*_t = \arg\max_{1\le i\le N}[\delta_T(i)]$ (2.22)

Backtracking, $q^*_t = \psi_{t+1}(q^*_{t+1}), t = T-1, T-2, \dots 1$ (2.23)

1.8 Pengolahan Citra Digital

Pengolahan citra digital adalah istilah umum untuk berbagai teknik yang keberadaannya untuk memanipulasi dan memodifikasi citra dengan berbagai cara. Setiap foto dengan bentuk citra digital dapat diolah menggunakan perangkat lunak tertentu. Foto adalah contoh gambar berdimensi dua yang dapat diolah dengan mudah. Prinsip-prinsip dasar dalam pengolahan citra: [5]

1. Peningkatan kecerahan dan kontras

   Peningkatan kecerahan dan kontras adalah sebuah perbaikan dalam citra yang merubah citra yang memiliki nilai kecerahan dan nilai kontras kurang menjadi nilai kecerahan dan nilai kontras yang lebih baik.
2. Penghilang derau

   Penghilang derau adalah proses penghilangan derau dari sebuah citra dengan keadaan distorsi atau memiliki derau.
3. Pencarian bentuk objek

   Pencarian bentuk objek adalah proses untuk mengenali suatu objek dalam sebuah citra dengan menggunakan pendekatan-pendekatan tertentu untuk mendapatkan nilai dari objek tersebut.

Beberapa jenis operasi dari pengolahan citra adalah sebagai berikut: [6]

1. Modifikasi Kecemerlangan (Brightness Modifikasi)

   Mengubah nilai keabuan atau warna dari gelap menjadi terang ataupun sebaliknya mengubah citra yang terlalu terang/pucat menjadi gelap.
2. Peningkatan Kontras (Contrast Enhancement)

   Dengan meningkatkan kontras dari sebuah citra maka titik yang cenderung gelap menjadi lebih gelap dan yang cenderung terang menjadi lebih terang.
3. Negasi

   Operasi untuk mendapatkan citra negatif (negative image)
4. Pengabuan (Grayscale)

   Merupakan proses konversi citra dengan warna sebenarnya (true color) menjadi citra keabuan (grayscale). Pengubahan gambar ke dalam bentuk grayscale ini dilakukan dengan mengambil nilai pixel dari satu gambar masukan kemudian dihitung menggunakan persamaan sebagai berikut:

$$Grayscale = \frac{(R+G+B)}{3} \quad \text{...(5)}$$

5. Pengambangan (Thresholding)

   Operasi ini digunakan untuk mengubah citra dengan format skala keabuan yang mempunyai nilai lebih dari dua menjadi citra biner yang hanya memiliki dua nilai yaitu 0 dan 1.
6. Pencerminan (Flipping)

   Pencerminan merupakan proses menggambar citra ke bentuk kebalikan dari citra tersebut, seperti ketika orang sedang bercermin.
7. Rotasi (Rotating)

   Rotasi adalah proses memutar koordinat citra sesuai dengan derajat yang ditentukan
8. Pemotongan (Cropping)

   Memotong bagian yang diinginkan dari sebuah citra
9. Pengskalaan (Scaling)

   Mengubah ukuran citra menjadi lebih besar atau lebih kecil
10. Deteksi Tepi (Edge Detection)

    Deteksi tepi adalah suatu proses yang menghasilkan tepi-tepi dari obyek dalam sebuah citra.

1.9 Deteksi Tepi (*Edge Detection*)

Deteksi tepi pada suatu citra adalah proses untuk memperoleh tepi objek. Deteksi tepi memanfaatkan perubahan nilai intensitas yang drastis pada batas dua area. Umumnya deteksi tepi menggunakan dua macam detektor, yaitu detektor baris (Hy) dan detektor kolom (Hx).

Deteksi tepi dibagi menjadi dua golongan. Golongan pertama yaitu disebut deteksi tepi orde pertama, yang bekerja dengan menggunakan turunan atau diferensial orde pertama. Yang termasuk ke dalam golongan ini adalah operator Roberts, Prewitt dan Sobel. Sedangkan golongan dua dinamakan orde kedua, yang menggunakan turunan orde kedua. Yang termasuk golongan orde kedua adalah Laplacian of Gaussian (LoG). Berikut adalah operator dalam proses deteksi tepi yang digunakan pada penelitian: [9]

Gambar 2 berikut adalah operator dari Sobel yang digunakan dalam proses deteksi tepi yang terdiri dari sepasang kernel berukuran 3 x 3:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad H_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Gambar 2 Metode Sobel

1.8 Normalisasi

Normalisasi merupakan proses merubah dimensi region tiap karakter. Pada proses ini menggunakan proses *Thresholding*, merupakan salah satu operasi yang termasuk ke dalam operasi titik dalam pengolahan citra digital. Operasi ini digunakan untuk mengubah citra dengan format skala keabuan yang dimiliki kemungkinan nilai lebih dari 2 ke citra biner yang memiliki 2 buah nilai (0 dan 1).

1.9 Pengujian Akurasi

Akurasi merupakan seberapa dekat suatu angka hasil pengukuran terhadap angka sebenarnya (true value atau reference value). Tingkat akurasi diperoleh dengan perhitungan sesuai dengan persamaan berikut :

$$Akurasi\ Keberhasilan = \frac{\sum Pengujian\ Data\ Benar}{\sum Banyak\ Data\ Uji} \times 100\%$$

2. ISI PENELITIAN

2.1 Analisis dan Perancangan Sistem

Berikut adalah analisis dan perancangan dari sistem yang dibangun.

2.1.1 Analisis Masalah

Berdasarkan dari masalah yang sudah didapat, maka akan dijabarkan secara lebih terperinci tentang masalah yang ditemukan dalam penelitian ini. Berikut adalah masalah yang ditemukan:

1. Dengan menggunakan metode *Hidden Markov Model* (HMM) dalam penelitian ini bagaimana metode ini dapat mengklasifikasi penyakit ikan berdasarkan citra dari hasil mikroskopis.
2. Akurasi dari metode Hidden Markov *Model Markov Model* (HMM).

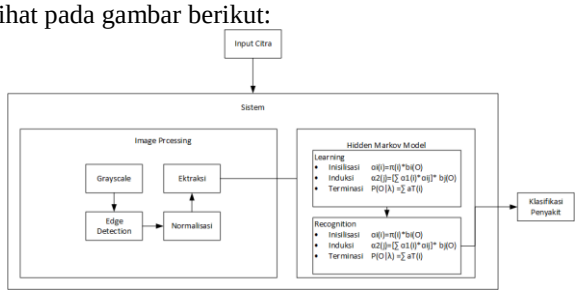
2.1.2 Analisis Data Masukan

Data masukkan yang dibutuhkan untuk aplikasi ini adalah sebuah citra. Citra dapat berupa hasil dari mikroskopis. Citra yang digunakan untuk aplikasi ini berformat '*.jpg*' atau '*.png*' dengan ukuran citra 80x80 pixel, citra tersebut akan menjadi data training setelah proses pengolahan citra. Citra yang dijadikan data masukkan ini akan dibagi menjadi beberapa tipe berdasarkan hasil analisis yang dilakukan oleh sistem, dimana akan dilakukan proses testing untuk menentukan hasil dari klasifikasi Tabel 1 Data Masukan

Parasit	Ciri-Ciri					Spesies	Tipe Prasite
	Filum	Subfilum	Class	Ordo			
1	Vermen	Platyhelminithes	Trematoda	Monogenea		Gyrodactylus sp	Patogen
2	Vermes	Platyhelminithes	Trematoda	Monogenea		Dactylogyrus	Patogen
3	Protozoa	Ciliophora	Ciliata	Osteichthyes		Myxobolus sp	Non-patogen
4	Arthopoda	-	Maxillopoda	Arguloida		Argulus	Patogen
5	Vermen	Platyhelminithes	Trematoda	Turbellaria		Temnocephala	Patogen
6	Protozoa	Ciliophora	Ciliata	Actinopterygi		Trichodina sp	Non-patogen

2.1.3 Analisis Proses

Secara umum aplikasi memiliki beberapa proses dalam melakukan analisis terhadap citra yang dimasukkan kedalam aplikasi. Setiap proses memiliki peran masing-masing untuk dapat mengenali citra yang dimasukkan. Gambar 2 dapat dilihat pada gambar berikut:



Gambar 3 Analisis Proses

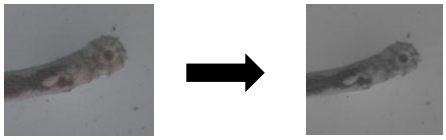
2.1.4 Proses Image Preprocessing

Pada proses ini terdapat beberapa proses pengolahan citra yang dibutuhkan pada simulasi yang akan dibangun mulai dari *grayscaling*, *edge detection*, normalisasi, dan ekstraksi.

2.1.5.1 Grayscale

Proses *graycale* adalah pengolahan citra yang normal (RGB) menjadi citra *grayscale*. Pengubahan gambar ke dalam bentuk *grayscale* ini dilakukan dengan mengambil nilai pixel dari satu gambar masukan kemudian dihitung menggunakan persamaan, berikut adalah citra hasil proses *grayscaling*.

:



Gambar 4 Citra Hasil Grayscale

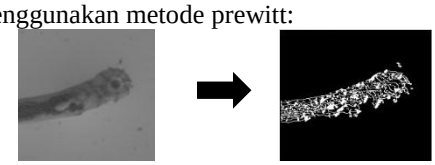
2.1.5.2 Edge Detection

Pada proses ini citra yang digunakan adalah citra berwarna *grayscale.* Proses *edge detection* ini dijalankan dengan menggunakan prewitt, metode prewitt menggunakan dua matriks berukuran 3x3 yaitu matriks Gx dan Gy. Kedua matriks tersebut digunakan untuk menghitung perbedaan warna pada pixel yang sedang dihitung dengan pixel disekitarnya secara horizontal dan vertikal. Matriks Gx digunakan untuk menghitung secara horizontal dan matriks Gy secara matriks vertikal. Gambar 7 berikut adalah operator dari prewitt yang digunakan pada proses *edge detection*:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad H_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Gambar 5 Operator Sobel

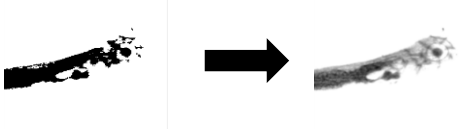
berikut adalah citra hasil *edge detection* dengan menggunakan metode prewitt:



Gambar 6 Citra Hasil Edge Detection

2.1.5.3 Normalisasi

Pada proses ini menggunakan proses *Thresholding*, yang merupakan proses pemisahan pikse-piksel berdasrkan derajat kebuan lebih kecil dari nilai batas yang dimilikinya. Piksel yang memiliki derajat keabuan lebih kecil dari bilai batas yang ditentukan akan diberikan nilai 0, sementara anilai yang memiliki derajat keabuan yang lebih besar dan batas akan diubah menjadi bernilai 1. Metode ini digunakan untuk mengkonversi data image menjadi data biner dengan tujuan agar proses selanjutnya menjadi lebih mu:



Gambar 7 Citra Hasil Normalisasi

2.1.5.4 Ekstraksi

Ekstraksi atau *feature extraction* merupakan proses dimana sample-sample penyakit yang telah melalui proses pengolahan citra. Gambar merupakan citra yang digunakan untuk proses berikutnya.



Gambar 8 Citra yang akan di ekstrak

Hasil ekstraksi dapat dilihat pada Gambar 9 yang merupakan hasil dari Gambar 8 sebagai berikut:**Error! Reference source not found.**

2.1.6 Proses Pembelajaran Learning Vector Quantization

Berikut adalah langkah-langkah yang dilakukan dalam proses pembelajaran *learning vector quantization*:

1. Setiap kelas diambil salah satu array vektornya dan dijadikan sebagai inisialisasi bobot (w). Sedangkan array lainnya menjadi data masukan (x).
2. Tentukan nilai maksimum perulangan, eror minimum, rasio pembelajaran dan pengurangan rasio.
3. Perhitungan nomor 4 sampai nomor 6 dilakukan selama perulangan masih lebih kecil dari maksimum perulangan atau rasio pembelajaran masih lebih besar dari eror minimum.
4. Untuk setiap data masukan (x) dihitung terhadang setiap data (w) dengan menggunakan persamaan 1.
5. Dilihat jarak (C) terkecil yang diperoleh pada perhitungan terhadap x. Kemudian diperiksa apakah target x sesuai dengan kelas bobot dengan jarak terkecil.
6. Dilakukan perhitungan perubahan bobot terhadap bobot dengan jarak terkecil, jika kelas target sesuai dengan bobot maka persamaan yang digunakan persamaan 2, sedangkan jika kelas target tidak sesuai dengan bobot maka persamaan yang digunakan adalah persamaan 3, Setelah itu hasil perhitungan bobot pada iterasi awal akan ditemukan.
7. Langkah nomor 4 sampai dengan nomor 7 dikerjakan terhadap semua masukan, setelah selesai dilakukan perhitungan rubah nilai rasio dengan menggunakan persamaan 4.
8. Perhitungan diulang hingga perulangan maksimum dan mendapatkan nilai bobot akhir dan dilakukan pembaruan bobot.

2.1.7 Proses Pengenalan Learning Vector Quantization

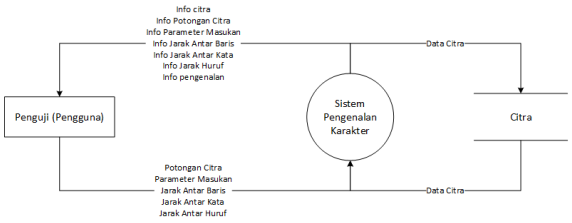
Proses pengenalan dengan metode *learning vector quantization* menggunakan rumus yang sama dengan perhitungan bobot pada proses pembelajaran.

Berikut langkah-langkah pengenalan:

1. Data *test* dihitung terhadang setiap data bobot (w) dengan persamaan 1:
2. Lihat nilai (C) terkecil yang diperoleh pada perhitungan terhadap data *test* . Kelas bobot yang memiliki jarak terkecil menjadi hasil dari proses pengenalan ini.

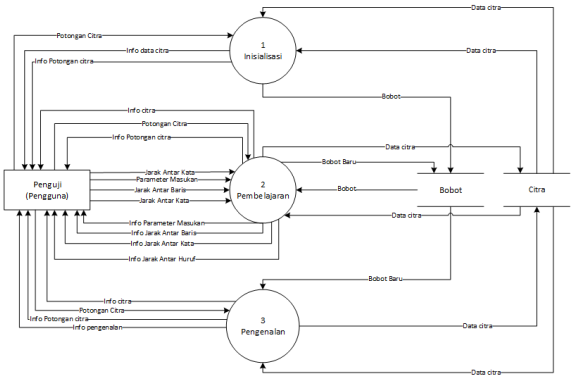
2.1.8 Analisis Kebutuhan Fungsionalitas Sistem

Berikut adalah diagram konteks dari sistem:



Gambar 1 Diagram Konteks

Berikut adalah DFD level 1 dari sistem:



Gambar 2 Data Flow Diagram Level 1

2.2 Implementasi dan Pengujian Sistem

Berikut akan dipaparkan mengenai implementasi dan pengujian sistem.

2.2.1 Lingkungan Implementasi

Lingkungan implementasi merupakan spesifikasi perangkat keras dan perangkat lunak dimana sistem ini akan digunakan. Berikut adalah lingkungan implementasinya:

1. Lingkungan Perangkat Keras

   Berikut adalah spesifikasi perangkat keras dalam pembangunan simulator ini:

   a. *Processor* : Intel Core i5-4210u 2,7 GHz
   b. *Memory* : RAM DDR 3 4 GB
   c. *Harddisk* : 1 TB
   d. *VGA* : NVIDIA GeForce 840m
   e. *Monitor* 14” dengan resolusi 1366 x 768
   f. *Keyboard*
   g. *Mouse*
   h. *Scanner*

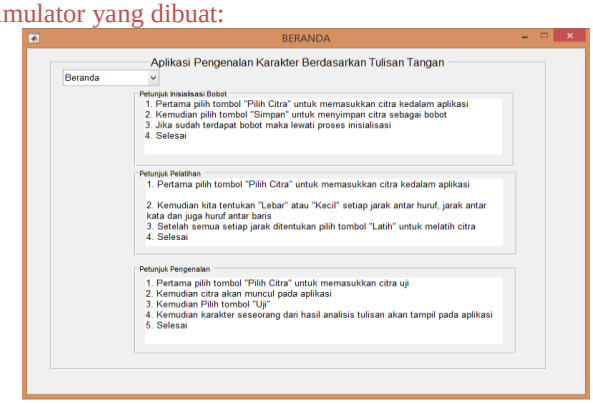
2. Lingkungan Perangkat Lunak

   Berikut adalah perangkat lunak yang digunakan dalam pembangunan simulator:

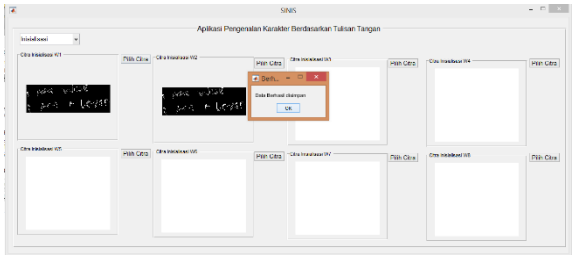
   a. Sistem Operasi : Microsoft Windows 8.1
   b. Pemrograman : MATLAB R2014b

2.2.2 Implementasi Antarmuka

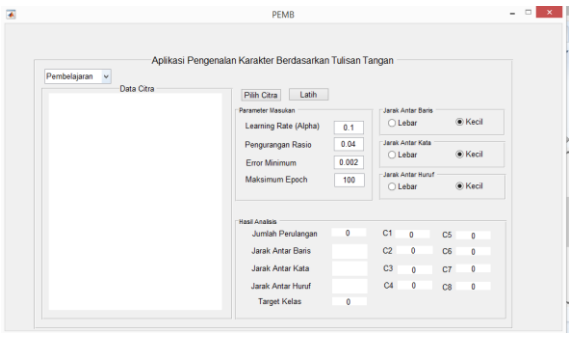
Berikut adalah implementasi antarmuka dari simulator yang dibuat:



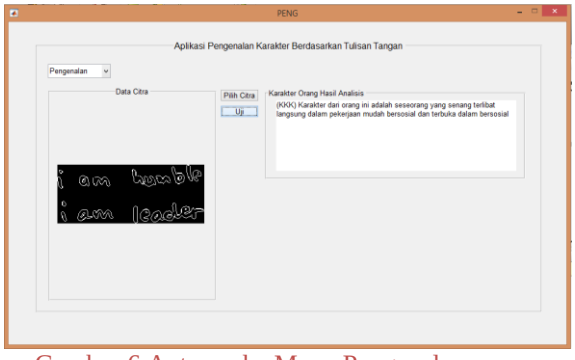
Gambar 3 Antarmuka Menu Beranda



Gambar 4 Antarmuka Menu Inisialisasi



Gambar 5 Antarmuka Menu Pembelajaran



Gambar 6 Antarmuka Menu Pengenalan

2.2.3 Pengujian

Berikut adalah hasil pengujian dari pembangunan simulator:

1. Evaluasi pengujian fungsional

Berdasarkan hasil pengujian fungsional yang dilakukan maka dapat disimpulkan bahwa simulator yang dibangun sudah berjalan sesuai dengan yang diharapkan.

2. Evaluasi Pengujian Akurasi

Dari lima pengujian akurasi yang dilakukan akurasi paling tinggi didapat dari pengujian akurasi ke-5. Pengujian akurasi 5 dilakukan dengan menggunakan kombinasi parameter masukan sebagai berikut:

a. *Learning Rate* (α) : 0.05
b. Pengurangan rasio : 0.1
c. Error Minimum : 0.01
d. Maksimum Epoh : 5

Berikut adalah hasil pengujian akurasi ke-5:

Berikut adalah akurasi yang dihasilkan dari proses pengujian akurasi 5 menggunakan persamaan 8:

$$Akurasi = \frac{69}{100} \times 100\% = 69\%$$

3. PENUTUP

3.1 Kesimpulan dan Saran

Berdasarkan dari hasil penelitian, analisis, perancangan sistem dan pembuatan simulator mengklasifikasi penyakit ikan dengan menggunakan metode *Hidden Markov Model* (HMM) maka kesimpulannya adalah metode *Hidden Markov Model* (HMM) dapat dipergunakan untuk mengklasifikasi penyakit ikan berdasarkan citra penyakit hasil dari mikrosopis, karena akurasi yang didapatkan sudah baik yaitu akurasi sebesar 69%.

Dari hasil penelitian, analisis, perancangan, pembuatan hingga pengujian, permasalahan yang terjadi pada saat pengolahan citra yang digunakan untuk mencari ciri dari citra hasil dari mikroskopis, mengakibatkan banyaknya ciri citra yang didapat dari pengolahan citra, sehingga vektor yang didapat akan sama dengan jumlah ciri yang didapat dari hasil pengolahan citra. Maka saran untuk meningkatkan penelitian selanjutnya mengenai analisis *Hidden Markov Model* (HMM) untuk mengklasifikasi penyakit ikan hasil dari mikrosopis adalah karena data citra masukan sangat berpengaruh dalam proses pembelajaran, disarankan untuk citra yang digunakan lebih baik bila terlebih dahulu menggunakan metode untuk mecari nilai vektor terhadap citra gambar yang akan digunakan sebagai data, baik itu data training maupun data uji agar ciri yang didapat tidak terlalu banyak, sehingga meningkatkan keakurasian dari metode *Hidden Markov Model* (HMM).

DAFTAR PUSTAKA

[1] Feliatra, "Identifikasi Bakteri Patogen (Vibrio Sp) di Perairan Nongso Batam Propinsi Riau," Jurnal Natur Indonesia II, vol. 1, pp. 28-33, 12 Juli 1999.

[2] A Hatmanti, R Nuchsin, and Y Darmayati, "Studi Penyakit Bakterial Pada Budidaya Ikan Kerapu dan Bakteri Penghambatnya di Perairan Teluk Lampung," Jurnal Akuakultur Indonesia, vol. 51-58, no. 7(1), 2008.

[3] Gusrina , Budidaya Ikan untuk SMK. Jakarta: Pusat Perbukuan Departemen Pendidikan, 2008.

[4] Afita Putri Lestari, Rencang Bangun Pengenalan Penyakit Darah Menggunakan Hidden Markov Model. Depok: Skripsi Fakultas Teknik Prgram Studi Teknik Elektro Fakultas Teknik Universitas Indonesia, 2008.

[5] Apoorva Mishra and Deepty Dubey, "A Comparative Study of Different Software," International Journal of Advance Research in, vol. 1, no. 5, p. 64, October 2013.

[6] R Usman, Parasit dan Penyakit Ikan. Padang: Fakutas Perikanan dan Ilmu Kelautan Universitas Bung Hatta, 2007.

[7] Keith D Griffiths, Histopathology Services in Wales.: Nasional Pathology Programme, 2010.

[8] Putra D, Pengolahan Citra. Yogyakarta: Andi Offset, 2010.

[9] Cristopher Danil, "Edge Detection dengan Algoritma Canny," Deteksi Tepi.

[10] V B Rao and H V Rao, Neural Networks and Fuzzy Logic. New York: Management Information Source, 1993.

[11] Riyanto Sigit, Robotika Sensor dan Aktuator. Jakarta: Graha Ilmu, 2009.

[12] P Dymarski, Hidden Markov Model, Theory and Aplications. India: InTech, 2011.

[13] Robert G Fichman and Chris F Kemerer, "Object-Oriented and Conventional Analysis and Design Methodologies," Massachusetts Institute of Technology, vol. 92, pp. 0018-9162, 1992.

[14] Kiyofumi Tanaka, Embedded Systems - Theory and Design Methodology, Kiyofumi Tanaka ed.: InTech, March 02, 2012.

[15] Rany Zuriatna Utami, Made Budi Suksmadana, and Bulkis Kanata, "MENENTUKAN LUAS OBJEK CITRA DENGAN TEKNIK DETEKSI TEPI," Dielektrika, vol. 2, no. 2086-9487, pp. 1 :11 - 17, Februari 2015.

[16] Dwijanto, "Bab 7 Rantai Markov," Riset Operasi , vol. 7.

[17] Kadir A and Susanto A, Teori dan Aplikasi Pengolahan Citra. Yogyakarta: Andi Offset, 2013.

[18] Nugroho Bonafir, Membuat Aplikasi Sistem Pakar Dengan PHP dan Editor Dreamwever. Yogyakarta: Gava Media, 2009.
[19] S Maria Yulianti, Pengenalan Retina Dengan Hidden Markov Model. Depok: Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia, 2008.
[20] Evi Andriani, Analisis dan Identifikasi Berbagai Penyakit Paru-Paru Dengan Metode Hidden Markov Model. Depok: Skripsi Fakultas Teknik Prgram Studi Teknik Elektro Fakultas Teknik Universitas Indonesia, 2009.
[21] Nisa Pandu Pertiwi, APLIKASI ALGORITMA VITERBI UNTUK MENENTUKAN BARISAN KEADAAN TERSEMBUNYI PADA HIDDEN MARKOV MODELS. Bandung: Universitas Perguruan Indonesia, 2011.
[22] Jia Li, Ami Najmi, and Robert M Gray, Image Classification by a Two Dimensional Hidden Markov Model.: The Pennsylvania State University , 2000.
[23] Firdaniza , Nurul Gusriani, and Akmal , "Hidden Markov Model," Trend Penelitian dan Pembelajaran Matematika di Era ICT, 24 Nopember 2006.
[24] Marc Mouret, Christine Solnon, and Christian Wolf, "Classification of images based on Hidden Markov Models," Classification of images based on Hidden Markov Models, vol. F-69622, p. CNRS, Februari 2005.
[25] Hanif Al Fatta, "Seminar Nasional Teknologi 2007 (SNT 2007) ," Konversi Format Citra RGB ke Format Grayscale Menggunakan Visual Basic, pp. 1978-9777, 24 November 2007.
[26] Angela Irfani, Ratih Amelia, and Dyah Saptanti, "Algoritma Viterbi dalam Metode Hidden Markov Models pada Teknologi," Makalah Stmik, 2006.
[27] Muhammad Eko Budi Prasetyo, "Departemen Teknik Informatika, Institut Teknologi Bandung," Teori Dasar Hidden Markov Model, vol. 025, 2010.
[28] Howard Anton and Chris Rorres, Aljabar Linear Elementer, 8th ed., Amelia Safitri, Ed. Jakarta: Erlangga, 2005.