

BAB 3

PERANCANGAN DAN PEMBUATAN SISTEM

Dalam bab ini akan dibahas mengenai perancangan dan pembuatan sistem aplikasi yang digunakan sebagai *user interface* untuk menangkap citra ikan, mengolahnya dan menampilkan hasilnya. Pengaturan keseluruhan sistem aplikasi dengan integrasinya terhadap *database* untuk melakukan penyimpanan data, serta pengaturan sistem agar dapat melakukan pengolahan gambar sehingga didapatkan output yang diinginkan.

3.1. Perancangan Data

Pada tahap awal yang dikerjakan pada perencanaan data gambar yang digunakan, meliputi :

1. Ukuran objek gambar relatif sama
2. Jumlah gambar yang ada di *database*
3. Format file gambar

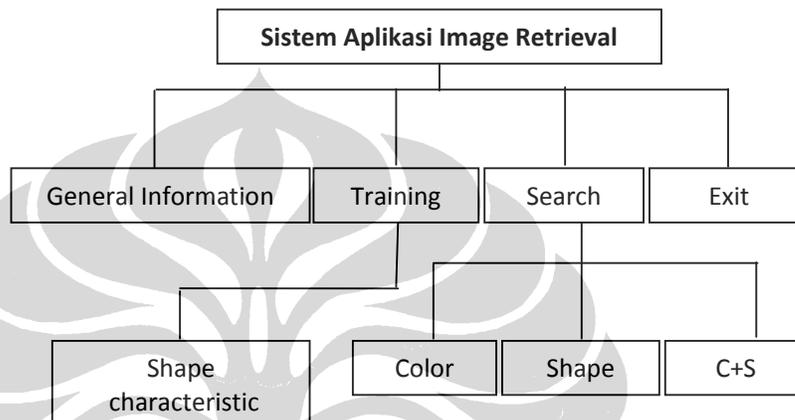
Dari jenis-jenis ekstensi format file gambar yang ada, pada perencanaan sistem ini penulis hanya menggunakan format file gambar yang umum digunakan untuk digital image yaitu dengan ekstensi JPEG (*Join Photographics Expert's Group*), jumlah gambar yang ada didalam image *database* sekitar 100 gambar. Dalam proses pencarian, user dapat dengan bebas menginput jumlah *database* gambar yang diinginkan dengan jumlah maksimal yang sudah ditentukan dan sistem aplikasi secara otomatis menghitung jumlah yang sudah ditentukan. Pada direktori *database* terdiri dari gambar yang akan digunakan sebagai gambar *database*. Adapun objek gambar yang digunakan adalah kategori ikan yaitu ikan cupang, ikan arwana, ikan Louhan, ikan oscar.

Gambar query disebut juga dengan gambar acuan atau gambar template adalah suatu gambar yang dijadikan acuan informasi (*content*) dalam proses pencarian. Sedangkan image *database* atau kumpulan gambar target adalah sekumpulan gambar yang akan digunakan sebagai *database* gambar.

3.2. Perancangan Sistem Aplikasi

Seperti halnya pembuatan sistem aplikasi pada umumnya, pembuatan sistem untuk tugas akhir (tesis) ini membutuhkan perencanaan yang optimal, terutama mengenai proses berjalannya suatu sistem.

3.2.1. Skenario Desain Perancangan Sistem



Gambar 3.1. Map Perancangan Sistem

Jika digambarkan dalam bentuk *user interface software pengolahan data matematis* adalah sebagaimana yang terlihat pada Gambar 3.2 dibawah ini

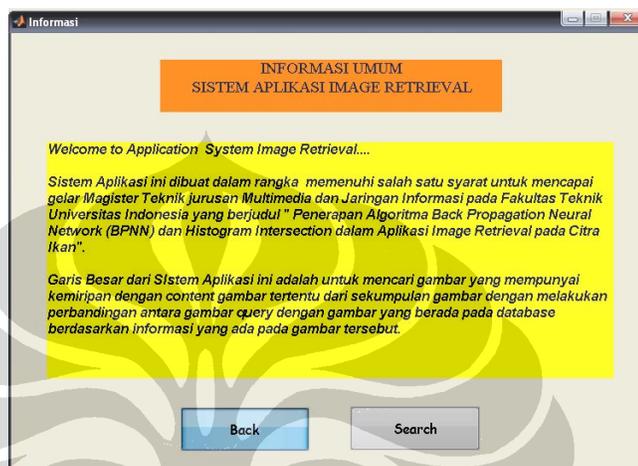


Gambar 3.2. Tampilan Utama *User Interface*

Deskripsi Umum dari Desain Perancangan sistem Aplikasi diatas adalah sebagai berikut:

1. Informasi Umum

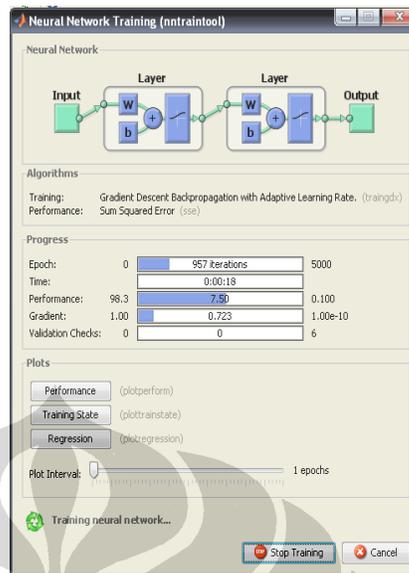
Digunakan untuk menampilkan maksud dan tujuan dari sistem aplikasi ini dibuat. Halaman ini hanya berisi informasi tentang layanan yang akan disediakan pada sistem aplikasi ini. Ketika *button general information* di klik maka akan muncul seperti pada Gambar 3.3 :



Gambar 3.3. Tampilan Informasi Umum

2. Training

Digunakan untuk melakukan proses pelatihan *Neural Network* (Jaringan Syaraf Tiruan). Dimana tujuan dari proses training tersebut adalah mengetahui jenis-jenis ikan berdasarkan pola bentuk (*shape*). gambar yang menjadi karakteristik dari kumpulan *database* gambar yang ada harus dilatih dengan Jaringan Syaraf Tiruan (JST), sebelum melakukan pencarian/*search* berdasarkan bentuk, jenis-jenis spesies ikan harus di *training* terlebih dahulu dengan sistem *Neural Network* (Jaringan Syaraf Tiruan) menggunakan metode *Back Propagation*. Pada proses pelatihan ini memerlukan proses yang cukup lama kurang lebih sekitar 1 s/d 3 jam. Sehingga nantinya dengan menggunakan sistem JST diharapkan untuk proses pencarian ikan berdasarkan bentuk (*shape*) dapat mencapai keakuratan diatas 95%. Gambar 3.4 menunjukkan tampilan *capture training*



Gambar 3.4. Tampilan *Training*

3. Search

Digunakan untuk melakukan proses pencarian gambar ikan. Di dalam halaman ini berisikan tiga pilihan kategori dalam pencarian gambar ikan, dimana masing-masing kategori menampilkan hasil sesuai dengan keinginan user. Sebagaimana terlihat pada Gambar 3.5 :



Gambar 3.5. Tampilan *Search*

Tiga kategori yang disediakan adalah :

1. Berdasarkan Warna

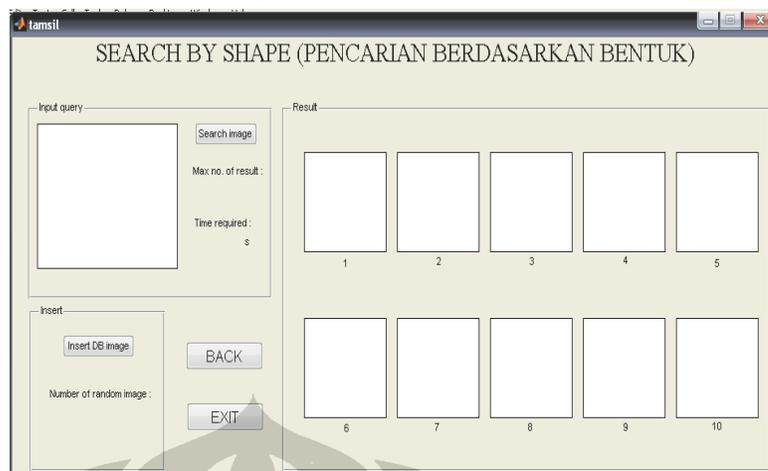
Halaman ini digunakan untuk mencari citra ikan yang disesuaikan dengan fitur warnanya. Dalam halaman ini ada dua tombol yang membantu *user* dalam melakukan proses pencarian warna citra ikan, diantaranya insert *database* (DB) *image* merupakan kumpulan dari data data gambar target ikan, pada *button* ini user bebas memasukkan jumlah DB image yang diinginkan dan tombol search untuk memulai pencarian warna gambar ikan sekaligus untuk menampilkan maksimal 10 gambar ikan yang sesuai dengan fitur ciri warna dan kategorinya. Gambar 3.6 menunjukkan pencarian berdasarkan warna



Gambar 3.6. Tampilan *Search* Berdasarkan Warna

2. Berdasarkan Bentuk

Sama halnya dengan halaman dalam pencarian warna. halaman ini digunakan untuk mencari gambar ikan yang disesuaikan dengan pola bentuk ikan. Dalam halaman ini menyediakan dua tombol yang membantu user melakukan proses pencarian gambar ikan, diantaranya *insert database* (DB) *image* merupakan kumpulan dari data data gambar target ikan dan tombol search untuk memulai pencarian pola bentuk ikan sekaligus untuk menampilkan 10 gambar ikan yang sesuai dengan pengenalan pola bentuk dan kategorinya. Gambar 3.7 menunjukkan pencarian berdasarkan pola bentuk



Gambar 3.7. Tampilan *Search* Berdasarkan Bentuk

3. Berdasarkan keduanya (Warna dan Bentuk)

Halaman ini digunakan untuk mencari gambar ikan yang disesuaikan dengan ciri fitur warna dan pola bentuknya. Dalam halaman ini menyediakan dua tombol untuk membantu user dalam melakukan proses pencarian gambar ikan berdasarkan ciri warna dan pola bentuk diantaranya tombol image DB untuk input jumlah gambar DB ikan yang diinginkan dan tombol search untuk memulai pencarian gambar ikan sekaligus untuk menampilkan 10 gambar ikan yang sesuai dengan ciri warna dan pola bentuk serta kategorinya



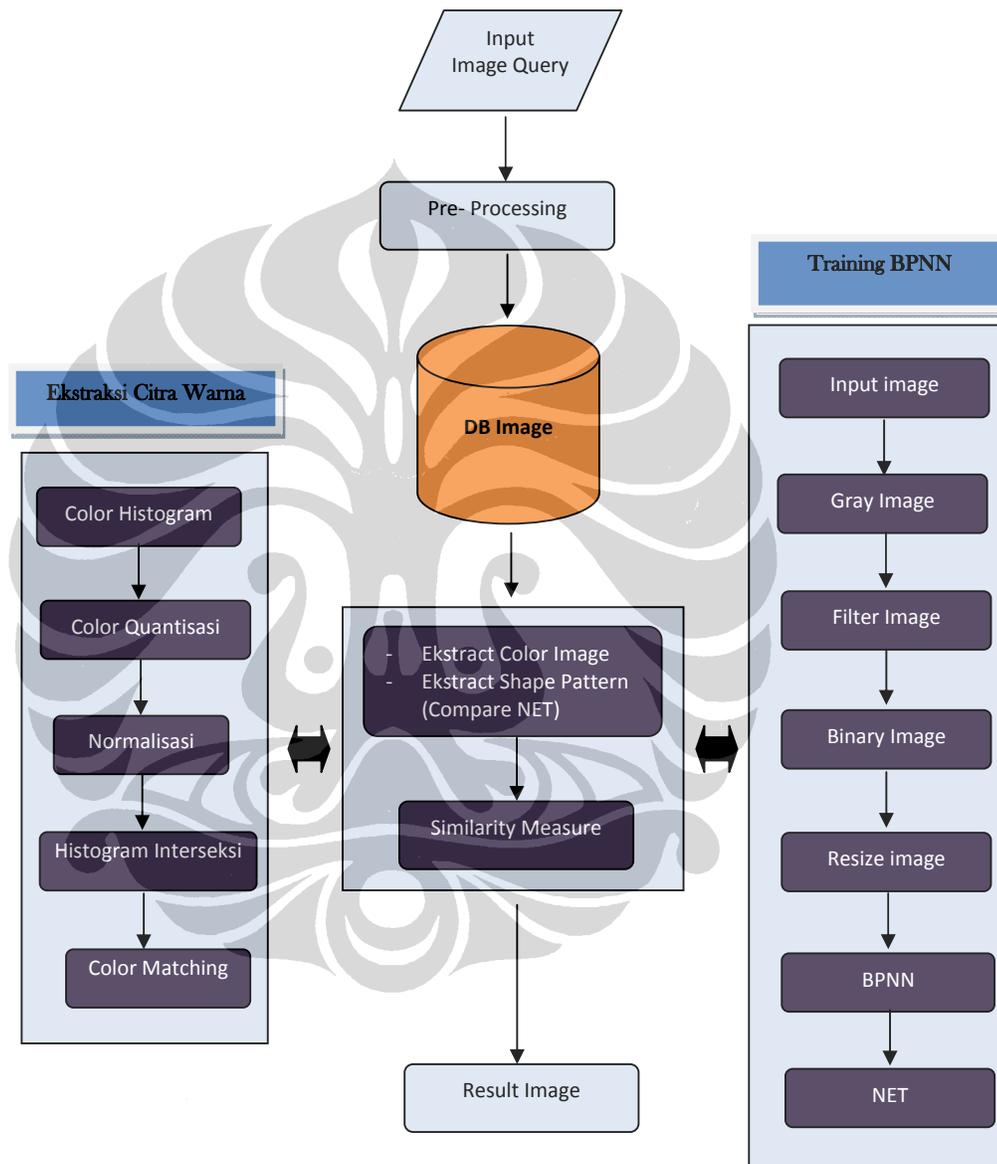
Gambar 3.8. Tampilan *Search* Warna dan Bentuk

4. Exit

Digunakan untuk keluar dari sistem aplikasi *image retrieval*.

3.3. Skema Umum Konfigurasi Sistem Proses Ekstraksi

Disini diterangkan secara umum skema konfigurasi sistem pada proses ekstraksi bagaimana merencanakan program untuk identifikasi ikan. Langkah pembuatannya seperti yang terlihat pada Gambar 3.9 dibawah ini



Gambar 3.9. Skema Umum Konfigurasi Sistem

Secara garis besar konfigurasi sistem pada proses ekstraksi diatas merupakan acuan untuk perancangan proses. Langkah dalam perancangan proses seperti dalam blok diagram adalah sebagai berikut :

1. Gambar query merupakan suatu gambar yang dijadikan acuan informasi dalam proses pencarian. Direktori *database* image digunakan sebagai kumpulan gambar-gambar yang ada di dalam *database*.
2. Selanjutnya dalam pencarian berdasarkan citra warna dilakukan ekstraksi ciri warna dengan dengan color histogram dengan menggunakan tipe *Global Color Histogram* (GCH) dilanjutkan dengan histogram interseksi yang tentunya sudah dilakukan terlebih dahulu proses *preprocessing* dari dua gambar yang telah ditentukan yaitu gambar query dan gambar *database*. langkah terakhir dalam perencanaan proses ini adalah mencari jarak minimum sebagai nilai kedekatan antara gambar query dengan gambar *database* sehingga akan didapatkan citra ikan yang sesuai antara gambar query dan gambar *database*.(khusus untuk ekstraksi warna)
3. Selanjutnya dalam pencarian berdasarkan pengenalan pola bentuk dilakukan dengan menggunakan jaringan syaraf tiruan dengan metode propagasi balik. sebelum melakukan pencarian gambar query, terlebih dahulu dilakukan proses pelatihan/*training* berdasarkan perbedaan jenis ikan, yang sebelum sudah dilakukan proses *preprocessing* (*grayscale, filtering*, citra biner), dari hasil proses Back Propagation Neural Network (BPNN) nantinya akan dihasilkan NET BPNN. ketika terjadi proses *searching* NET BPNN dari hasil training di bandingkan dengan NET BPNN dari gambar Query dengan membandingkan error toleran terkecil. Jadi pada proses pengenalan pola bentuk proses training sangat menentukan keakuratan proses pencarian.
4. Langkah terakhir dalam perencanaan proses ini adalah mengintegrasikan pencarian berdasarkan citra warna dan pengenalan pola bentuk ikan berdasarkan algoritma dimaksud.

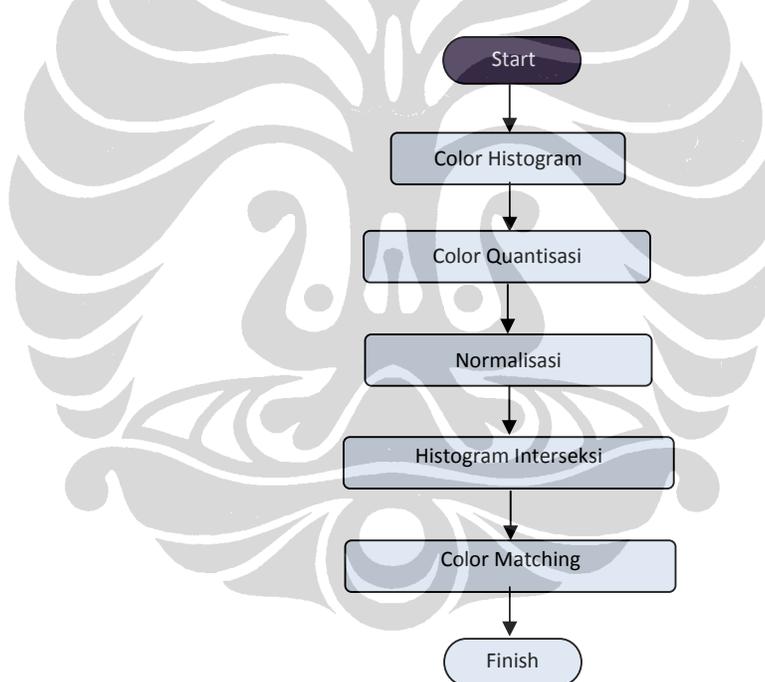
3.4. Ekstraksi Ciri

3.4.1. Ciri Citra Warna (Color Image)

Untuk metode yang digunakan pada ekstraksi ciri warna dalam seminar ini adalah metode histogram interseksi. Akan tetapi sebelum menggunakan metode ini terlebih dahulu dilakukan beberapa tahapan proses diantaranya [9] :

- Histogram Warna (*Color Histogram*)
- Kuantisasi Warna (*Color Quantization*)
- Normalisasi
- Histogram Interseksi (*Histogram Intersection*)

Untuk lebih jelasnya langkah-langkah dalam proses ekstraksi ciri warna akan di gambarkan dalam suatu diagram alir. Berikut ini Diagram alir proses ekstraksi ciri citra warna [9] :



Gambar 3.10. Flowchart Ekstraksi Citra Warna

Diagram alir / flowchart pada gambar 3.10 menunjukkan proses ekstraksi ciri warna dengan beberapa proses, antara lain seperti yang dijelaskan berikut ini :

Model warna Red, Green, Blue (RGB) merupakan yang paling banyak digunakan pada sistem Content Based Image Retrieval (CBIR). Pada model ini, warna

direpresentasikan menjadi tiga warna primer, yaitu : merah, hijau, dan biru. Nilai masing-masing warna primer itu berkisar antara 0 – 255

1. *Color Histogram*

Color histogram adalah representasi distribusi warna dalam sebuah gambar yang didapatkan dengan menghitung jumlah pixel dari setiap bagian range warna secara tipikal dalam dua dimensi atau tiga dimensi. Distribusi warna ini dimodelkan dengan *color histogram*.

Color histogram tersebut didefinisikan sebagai berikut :

$$H[r,g,b] = N \cdot \text{Prob} \{ R=r, G=g, B=b \}$$

dimana :

R,G,B merupakan tiga macam warna

N adalah jumlah pixel pada gambar.

Misalnya ada sebuah gambar berukuran 3x3 pixel dengan nilai RGB sebagai berikut:

(1,1,1) (1,2,0) (1,2,0)

(1,1,0) (2,1,0) (2,3,1)

(3,2,1) (2,2,1) (2,1,0)

Bila yang digunakan adalah format H(r,g,b) dimulai dari H(0,0,0) s/d H(3,3,3).

maka histogram gambar tersebut adalah sebagai berikut :

$$H(0,0,0)=0 \quad H(0,0,1)=0 \quad H(0,0,2)=0 \quad H(0,0,3)=0$$

$$H(0,1,0)=0 \quad H(0,1,1)=0 \quad H(0,1,2)=0 \quad H(0,1,3)=0$$

$$H(0,2,0)=0 \quad H(0,2,1)=0 \quad H(0,2,2)=0 \quad H(0,2,3)=0$$

$$H(0,3,0)=0 \quad H(0,3,1)=0 \quad H(0,3,2)=0 \quad H(0,3,3)=0$$

$$H(1,0,0)=0 \quad H(1,0,1)=0 \quad H(1,0,2)=0 \quad H(1,0,3)=0$$

$$H(1,1,0)=1 \quad H(1,1,1)=1 \quad H(1,1,2)=0 \quad H(1,1,3)=0$$

$$H(1,2,0)=1 \quad H(1,2,1)=0 \quad H(1,2,2)=0 \quad H(1,2,3)=0$$

$$H(1,3,0)=0 \quad H(1,3,1)=0 \quad H(1,3,2)=0 \quad H(1,3,3)=0$$

$$H(2,0,0)=0 \quad H(2,0,1)=0 \quad H(2,0,2)=0 \quad H(2,0,3)=0$$

$$H(2,1,0)=2 \quad H(2,1,1)=0 \quad H(2,1,2)=0 \quad H(2,1,3)=0$$

$$H(2,2,0)=0 \quad H(2,2,1)=1 \quad H(2,2,2)=0 \quad H(2,2,3)=0$$

Tabel 3.1. Color Histogram Gambar

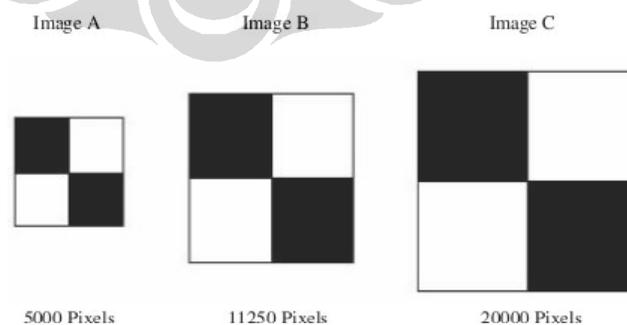
Color	Number of pixels
(0,0,0)	230
(0,0,1)	40
(0,0,3)	677
...	...
...	...
...	...
(3,3,3)	4128

Seperti yang ditunjukkan pada Gambar 3.11 dan tabel 3.1 menunjukkan bahwa gambar tersebut yang telah melalui proses kuantisasi menjadi 64 warna. Sesuai dengan distribusi warna pada tiap pixel, *color histogram* gambar adalah sebagai berikut: $H = \{230, 40, 677, \dots, 4128\}$.

3. Normalisasi

Penggunaan nilai-nilai aktual distribusi warna pada histogram, membuat untuk dipahami. Namun pemakaian dengan cara ini akan menimbulkan masalah jika diterapkan pada gambar yang mempunyai ukuran berbeda namun sebenarnya mempunyai distribusi warna yang sama.

Sebagai contoh, misalnya ada 3 gambar dengan ukuran berbeda yang terkuantisasi menjadi 2 warna (hitam dan putih) seperti yang diperlihatkan pada Gambar 3.12



Gambar 3.12. Ukuran Berbeda, Tetapi Distribusi Warna Sama

Histogram 3 gambar ini adalah:

$$H^A = \{2500, 2500\}$$

$$H^B = \{5625, 5625\}$$

$$H^C = \{10000, 10000\}$$

Seperti dilihat, meskipun ketiga gambar tadi mempunyai distribusi warna yang sama, tapi mempunyai histogram yang berbeda. Ini dikarenakan perbedaan ukuran gambar (dan tentu saja jumlah pixel). Oleh karena itu, untuk membuat histogram tetap sama pada gambar yang mempunyai kesamaan distribusi warna, maka diperlukan suatu normalisasi histogram.

Dengan cara ini, selama distribusi warna pada gambar sama, histogram warnanya akan sama, tidak tergantung lagi pada ukuran gambar. Berikut adalah hasil histogram warna pada gambar 2 yang sudah ternormalisasi:

$$H^A = \{50\%, 50\%\}$$

$$H^B = \{50\%, 50\%\}$$

$$H^C = \{50\%, 50\%\}$$

4. Histogram Interseksi

Histogram interseksi digunakan untuk menentukan jarak antara dua histogram yaitu histogram dari gambar query dengan gambar *database*. Jarak minimum ini menentukan kedekatan antara dua obyek. Gambar yang memiliki jarak paling kecil, merupakan solusinya.

Sebagai penjelasan, dimisalkan ada dua gambar dengan histogram 4 warna yang sudah terkuantisasi sebagai berikut:

$$H_A = \{20\%, 30\%, 10\%, 40\%\}$$

$$H_B = \{10\%, 10\%, 50\%, 30\%\}$$

Cara menghitungnya yaitu dengan menggunakan rumus :

$$d(A, B) = \sqrt{\sum_{j=1}^n (H_j^A - H_j^B)^2} \quad (3.1)$$

Jika nilai 2 histogram tersebut dimasukkan ke dalam rumus diatas, maka hasilnya adalah sebagai berikut:

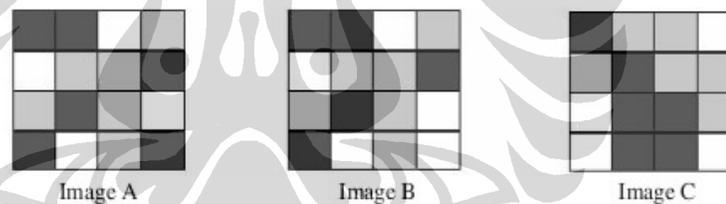
$$d(A, B) = |0.2 - 0.1| + |0.3 - 0.1| + |0.1 - 0.5| + |0.4 - 0.3| = 0.8$$

Tingkat similarity (kemiripan) yang tinggi akan tercapai jika $d(A,B)$ yang dihasilkan adalah 0.

5. Tipe Histogram Warna

Histogram warna terdiri dari dua tipe, *Global Color Histograms* (GCH) dan *Local Color Histograms* (LCH). Pada penggunaan GCH, distribusi warna global suatu gambar diambil dan digunakan sebagai metadata. Jika user mencari gambar dengan yang dalam sistem *databasenya* hanya memperhatikan distribusi warna global suatu citra, GCH adalah pilihan terbaik. Walaupun demikian, karena GCH hanya mengambil distribusi warna global suatu gambar sebagai pertimbangan untuk membandingkan gambar, ini bisa mengembalikan hasil yang tidak sesuai dengan persepsi visual manusia

Misalkan ada tiga gambar yang telah dikuantisasi menjadi tiga warna: hitam, abu-abu, dan putih seperti yang diperlihatkan pada Gambar 3.13. Misalkan gambar A adalah query image, sedangkan gambar B dan C adalah gambar-gambar dalam database.



Gambar 3.13. Gambar Yang Terkuantisasi Menjadi 3 Warna

Tabel 3.2. GCH Image A, B, dan C

Image	Hitam	Abu-abu	Putih
A	37.5%	37.5%	25%
B	31.25%	37.5%	31.25%
C	37.5%	37.5%	25%

Sedangkan Distribusi warna (GCH) tiga gambar diatas adalah seperti pada Tabel 3.2. Maka, jarak antara gambar A dengan gambar B dan C adalah :

$$d(A,B) = |0.375 - 0.3125| + |0.375 - 0.375| + |0.25 - 0.3125| = 0.125$$

$$d(A,C) = |0.375 - 0.375| + |0.375 - 0.375| + |0.25 - 0.25| = 0$$

Dari hasil perbandingan, **gambar C** ternyata ditemukan lebih mirip daripada gambar B (karena jarak C lebih kecil). Padahal, sesuai dengan persepsi manusia, yang lebih mirip dengan **gambar A** sebenarnya adalah **gambar B**.

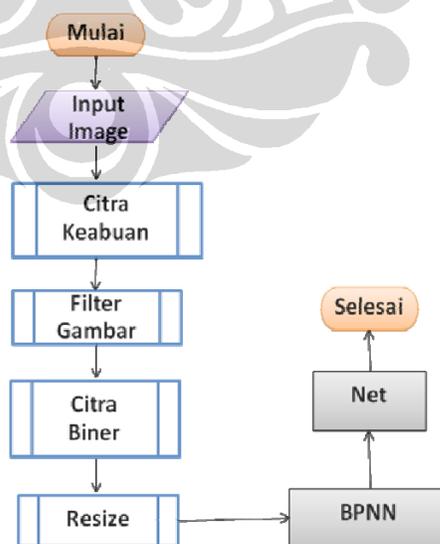
Global Color Histogram (GCH) merepresentasikan keseluruhan bagian gambar dengan satu histogram.

Sedangkan *Local Color Histogram* (LCH) membagi gambar menjadi beberapa bagian dan kemudian mengambil histogram warna tiap bagian tadi. LCH memang berisi lebih banyak informasi tentang gambar, namun metode ini membutuhkan lebih banyak proses komputasi.

3.4.2. Pengenalan Pola Bentuk

Selain menggunakan ekstraksi ciri warna, metode yang digunakan untuk pencarian citra/gambar ikan adalah dengan pengenalan pola berdasarkan bentuk menggunakan *Artificial Neural Network* (Jaringan Saraf Tiruan) dengan metode backpropagation. Adapun flowchart untuk pengenalan ciri berdasarkan bentuk dengan menggunakan Jaringan Saraf Tiruan (Neural Network) terbagi 2 yaitu flowchart untuk *training* karakteristik ikan yang ada di DB (*Database*) dan flowchart untuk *pencarian berdasarkan bentuk*.

Diagram alir/flowchart untuk *training Neural Network* adalah seperti yang diperlihatkan pada Gambar 3.14



Gambar 3.14. Diagram Alir *Training Neural Network*

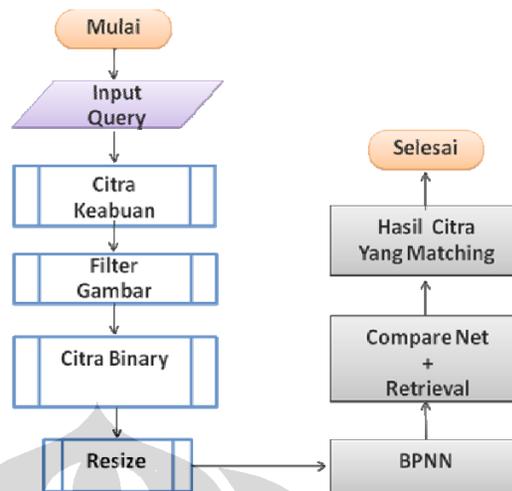
Flowchart (Diagram Alir) pada gambar 3.14 menunjukkan proses pengenalan pola bentuk dengan beberapa proses, yang pertama dilakukan adalah dengan pelatihan (*training*) pada Jaringan Syaraf Tiruan (JST), antara lain seperti yang dijelaskan berikut ini :

Sebelum dilakukan pelatihan pada Jaringan Syaraf Tiruan (JST), terlebih dahulu *image* yang akan diinput sebagai pola latihan pada Jaringan Syaraf Tiruan (JST) dilakukan proses pengolahan gambar (*image Processing*) yang terdiri dari proses Image RGB ke *grayscale*, *filter noise*, binerisasi/citra biner dan di resize kemudian hasil dari *image processing* tersebut dilakukan proses pelatihan (*training*) dengan metode *Back Propagation Neural Network* (BPNN) dengan setting konfigurasi parameter sebagai berikut :

- Jumlah lapisan tersembunyi sebanyak 20.
- Fungsi aktivasi yang digunakan adalah logsig (sigmoid biner) artinya suatu fungsi yang continue dan terdiferensial dengan mudah, memiliki range (0,1).
- Metode *learning* yang digunakan adalah *traingdx* artinya metode BPNN yang dipercepat dengan *learning rate* variabel dan momentum.
- Jumlah error minimum adalah 0,1
- Jumlah *epoch* (iterasi) learning maksimal BPNN adalah 5000 (defaultnya 100)
- Jumlah momentum sebesar 0,95.
Momentum adalah perubahan bobot yang didasarkan atas arah perubahan gradient pola terakhir dan pola sebelumnya.

Proses setting konfigurasi parameter adalah proses yang sangat menentukan keakuratan retrieval dalam membedakan jenis ikan, didalam parameter ini membutuhkan penelitian yang cukup lama. Kemudian hasil keluaran dari proses *training* Jaringan Saraf Tiruan disimpan dalam bentuk net.

Sedangkan flowchart untuk pencarian berdasarkan bentuk adalah seperti yang diperlihatkan pada Gambar 3.15



Gambar 3.15. Diagram Alir Pencarian Pola Bentuk

Flowchart (diagram alir) pada gambar 3.15 menunjukkan proses pencarian pengenalan pola bentuk dengan dengan beberapa proses, antara lain seperti yang dijelaskan berikut ini :

Khusus untuk proses pengolahan gambar (*image processing*) pada pencarian berdasarkan pola bentuk pada prinsipnya sama dengan yang ada pada proses pelatihan (*training*) sebagaimana yang dijelaskan sebelumnya, yaitu terdiri dari *grayscale*, *noise filtering* dan binerisasi atau citra biner dan *resize image* kemudian di proses pengenalan bentuk dengan metode *Back Propagation Neural Network* (BPNN). Hasil keluaran dari proses Jaringan Saraf Tiruan image query (gambar yang menjadi acuan informasi/gambar yang ingin dicari) disimpan dalam bentuk net, hasil dari net query di *compare* (dibandingkan) dengan data net yang ada pada proses *training* yang sudah dilakukan sebelumnya, kemudian dicari *similarity* nya (kemiripan) berdasarkan pola bentuk.

Jadi penelitian di dalam mengenali pola dengan metode *Back Propagation Neural Network*, proses yang sangat menentukan adalah pada proses training *Back Propagation Neural Network* (BPNN).

3.4.3. Integrasi Citra Warna dan Pengenalan Pola Bentuk

Langkah terakhir dalam perencanaan proses ini adalah mengintegrasikan pencarian berdasarkan citra warna dan pengenalan pola bentuk ikan berdasarkan algoritma histogram interseksi dan *Back Propagation Neural Network* (BPNN)