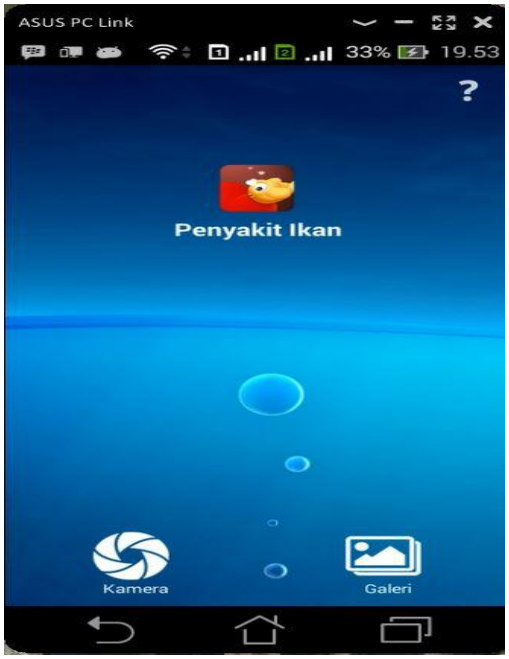
BAB IV
HASIL PENGUJIAN DAN ANALISA

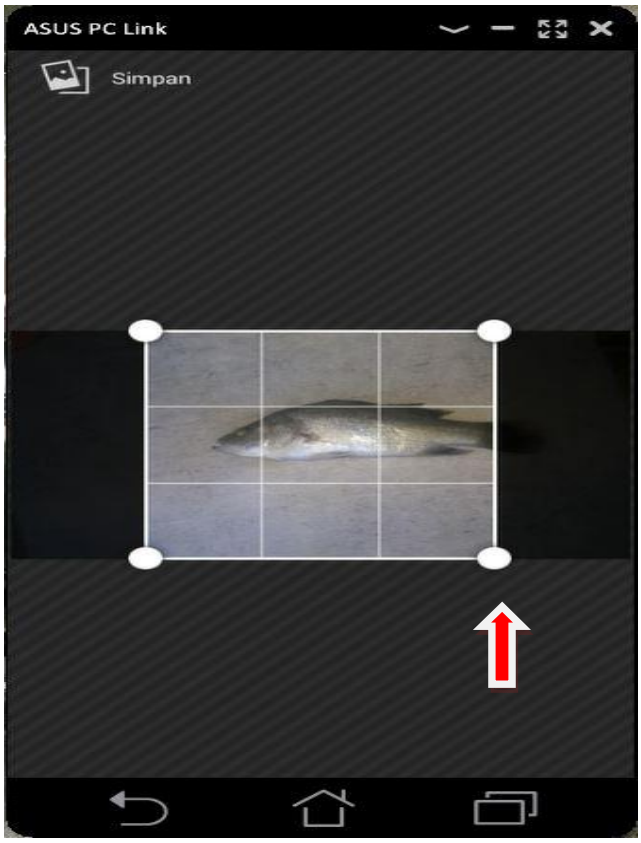
4.1 Cara Kerja dan Penggunaan Aplikasi

Ketika dijalankan program aplikasi akan menampilkan beberapa opsi yaitu akses kamera, akses galeri, atau bantuan. Seperti yang terlihat pada gambar 4.1



Gambar 4.1. Tampilan awal aplikasi

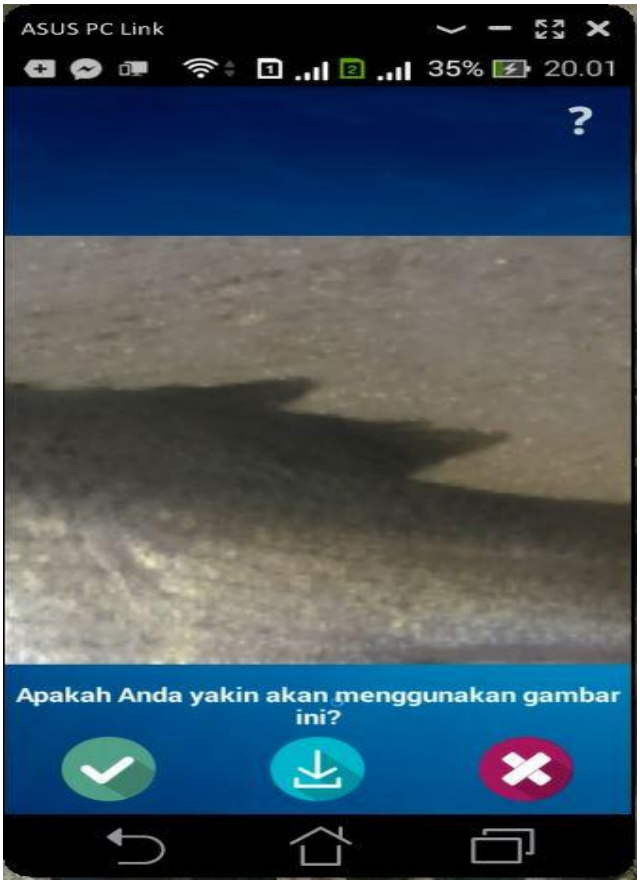
Menu “Kamera” digunakan jika pengguna ingin mengambil foto secara langsung melalui kamera, sedangkan menu “Galeri” digunakan jika user ingin memilih foto yang tersimpan di *memory* eksternal sedangkan symbol tanda Tanya (?) yaitu untuk bantuan. Setelah user menentukan foto yang akan diolah, user akan memilih area yang akan diidentifikasi dan akan ditampilkan pada antarmuka tampilan *Image View* seperti terlihat pada gambar 4.2.



Gambar 4.2. Tampilan Image View

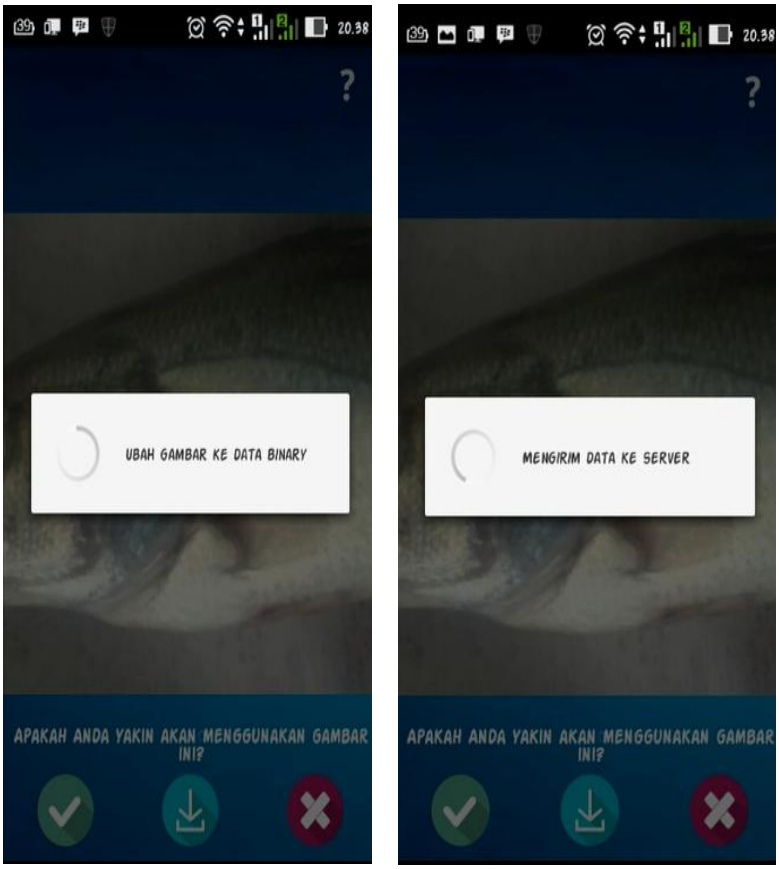
Pada tampilan tersebut, pengguna harus menentukan sendiri potongan gambar yang akan diolah menggunakan pengolahan citra. Proses pemotongan cukup dilakukan dengan cara menyentuh dan memposisikan jari pada titik awal area yang akan diseleksi dan menariknya secara diagonal ke arah yang berlawanan, seperti yang diilustrasikan pada gambar 4.2 diatas. Pada ilustrasi gambar tersebut, posisi penentuan gambar yang akan diseleksi, dapat dilakukan dari beberapa sisi.

Jika pengguna telah melakukan seleksi pada gambar, lau klik simpan pada direktori aplikasi untuk kemudian ditampilkan pada kotak dialog, seperti yang terlihat pada gambar 4.3.



Gambar 4.3. Tampilan Dialog

Pada tampilan kotak dialog diatas, pengguna diminta untuk menentukan pilihan. Jika ingin melanjutkan proses klik symbol  . Jika ingin mengulang pemilihan area pada gambar klik symbol . Sedangkan symbol  digunakan jika ingin menyimpan gambar yang telah di crop. Tampilan aplikasi pada saat melakukan proses pengolahan citra ditunjukkan pada gambar 4.4.



(a) (b)

Gambar 4.4. Tampilan pada saat aplikasi melakukan proses pengolahan citra

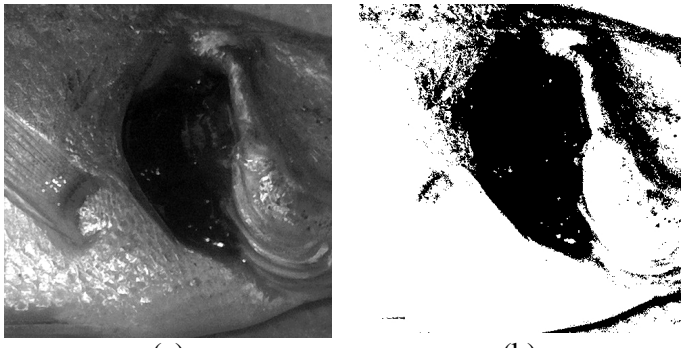
Pada saat terjadi, potongan gambar akan diolah oleh program aplikasi yang kemudian menghasilkan gambar *grayscale* dan gambar biner. Gambar biner yang dihasilkan selanjutnya akan dikirim ke server untuk dikenali. Hasil gambar yang sudah dikenali tersebut akan diunduh oleh program aplikasi.

Selanjutnya, file yang akan dihasilkan dari layanan server akan dibaca oleh program untuk mengambil hasil idenifikasi yang akan diproses oleh server. Hasil identifikasi kemudian akan ditampilkan di layar, seperti yang terlihat pada gambar 4.5.



Gambar 4.5. Tampilan hasil akhir

Gambar yang dihasilkan dari proses pengolahan citra ditunjukkan pada gambar 4.6 berikut.



(a) (b)

Gambar 4.6. Hasil pengolahan citra oleh program aplikasi:

(a) Citra *grayscale*, (b) Citra biner

4.2 Pengujian dan Analisa

Pengujian dilakukan untuk mengetahui kondisi yang baik dari gambar yang diambil untuk kemudian diproses oleh program aplikasi sehingga gambar yang dihasilkan bisa dikenali dengan baik oleh server. Pada tiap pengujian akan dilakukan analisis untuk mengetahui pengaruh gambar yang diuji terhadap hasil pengenalan oleh server.

4.2.1 Pengujian Pengkondisian Warna

Pada pengujian ini, *sample* gambar diperoleh galeri. Pada gambar yang sudah disediakan dibuat warna gambar dan latar belakang pada masing-masing *sample* gambar.

Pengujian dibagi menjadi dua tahap, pengujian tahap pertama ditujukan untuk menguji pengaruh warna latar belakang sedangkan pengujian tahap 2 menguji warna pada gambar terhadap hasil pengolahan gambar sampai dengan hasil identifikasi, pada saat proses pemotongan gambar secara manual. Pengujian pada tahap ini terlampir pada lampiran no.2 dan no.4 yang dilakukan sebanyak 10 kali pengujian pada tiap warna. Dari data pada lampiran no.2 dan no.4 maka didapat kesimpulan:

a. Warna latar biru berhasil teridentifikasi dengan tepat sebanyak 90% dan gagal teridentifikasi sebanyak 10% disebabkan bentuk ekor pada no.2 transparan.

b. Warna latar merah berhasil teridentifikasi dengan tepat sebanyak 90% dan gagal teridentifikasi sebanyak 10% disebabkan bentuk ekor pada no.2 transparan.

c. Warna latar hijau berhasil teridentifikasi dengan tepat sebanyak 90% dan gagal teridentifikasi sebanyak 10% disebabkan bentuk ekor pada no.2 transparan..

d. Data pada no.2 mempunyai nilai diluar jangkauan sehingga tidak teridentifikasi.

Dari ketiga pengujian pengkondisian warna pada tahap pertama diketahui bahwa:

- Warna latar tidak mempengaruhi proses citra.
- Hasil identifikasi gambar oleh server menunjukkan bahwa layanan tersebut dapat mengenali gambar yang dengan berbeda latar asalkan bentuk ekor tidak transparan.

Pada pengujian tahap kedua, menguji pengaruh perbedaan intensitas warna antara gambar dan latar belakang terhadap hasil pengolahan gambar sampai dengan hasil identifikasi gambar dan juga menguji sejauh mana tingkat keakuratan proses binarisasi gambar dengan menggunakan metode Otsu. Pengujian pada tahap ini terlampir pada lampiran no.2 dan no.4 yang dilakukan sebanyak 10 kali pengujian pada tiap warna. Dari data pada lampiran no.2 dan no.4 maka didapat kesimpulan:

a. Warna gambar dan latar biru berhasil teridentifikasi dengan tepat sebanyak 90% dan gagal teridentifikasi sebanyak 10% disebabkan bentuk ekor pada no.2 transparan.

b. Warna gambar dan latar merah berhasil teridentifikasi dengan tepat sebanyak 90% dan gagal teridentifikasi sebanyak 10% disebabkan bentuk ekor pada no.2 transparan.

c. Warna gambar dan latar hijau berhasil teridentifikasi dengan tepat sebanyak 90% dan gagal teridentifikasi sebanyak 10% disebabkan bentuk ekor pada no.2 transparan.

d. Data pada no.2 mempunyai nilai diluar jangkauan sehingga tidak teridentifikasi.

Dari ketiga pengujian pengkondisian warna pada tahap kedua diketahui bahwa:

- Pada percobaan yang kedua untuk pengkodisian warna masih dapat dikenali oleh sistem asalkan bentuk yang di identifikasi tidak transparan.

- Biner yang dihasilkan sangat berpengaruh terhadap hasil identifikasi oleh server. Semakin tinggi tingkat gangguan pada gambar biner, menyebabkan hasil identifikasi sulit dikenali oleh server.

4.2.2 Pengujian Gambar Miring

Pada pengujian ini, sample gambar diperoleh dari galeri dengan pengkondisian gambar yang sengaja dibuat miring. Tingkat kemiringan dibagi menjadi dua kategori, yaitu kemiringan wajar, dan kemiringan tidak wajar.

Pengujian ini ditujukan untuk mengetahui pengaruh kemiringan gambar terhadap hasil pengenalan yang dilakukan oleh server. Pengujian pada tahap ini terlampir pada lampiran no.3 dan no.4 yang dilakukan sebanyak 10 kali pengujian pada tiap gambar. Dari data pada lampiran no.3 dan no.4 maka didapat kesimpulan:

a. Tingkat kemiringan wajar berhasil teridentifikasi dengan tepat sebanyak 80% dan gagal teridentifikasi sebanyak 20% disebabkan pada saat gambar diubah ke dalam gambar biner bentuknya tidak sesuai dengan gambar aslinya.

b. Tingkat kemiringan tidak wajar berhasil teridentifikasi dengan tepat sebanyak 70% dan gagal teridentifikasi sebanyak 30% disebabkan pada saat gambar diubah ke dalam gambar biner bentuknya tidak sesuai dengan gambar aslinya.

c. Ketika sebuah gambar diluar nilai *Similarity* yang seharusnya maka sistem akan mengidentifikasi hasil yang lainnya.

Dari dua pengujian diatas, diketahui bahwa server hanya mampu mengenali sebagian gambar dengan tingkat kemiringan yang wajar dan tidak wajar. Tingkat kemiringan yang tidak wajar lebih sulit dikenali dikarenakan tingkat kemiringan mempengaruhi hasil identifikasi.

4.2.3 Pengujian Gambar Selain Ikan Kakap

Pada pengujian ini, sample gambar diperoleh dari galeri dengan Pengkondisian gambar yang sengaja bukan ikan. Pengujian ini ditujukan agar mengetahui apakah benda selain ikan mempunyai nilai atau tidak.

Tabel 4.1 Pengujian gambar selain ikan

No	Gambar	Hasil
1		

2	Energizer Apakah Anda yakin akan menggunakan gambar ini?	Trematoda Trematoda adalah cacing pipih, biasanya menyerang insang cirinya nya warna insang merah pucat. Cara penanganannya ikan yang sakit direndam larutan formalin 30-50 ppm selama 24 s/d48 jam atau larutan formalin 100-200 ppm selama 30 menit s/d 1 jam dan perendaman H2O2 200 ppm selama 1 jam. Sehat 46.9 Sakit 62.5 Ok Apakah Anda yakin akan menggunakan gambar ini?
3	Apakah Anda yakin akan menggunakan gambar ini?	Tricodiniasis Tricodiniasis penyakit ini biasanya menyerang sirip ikan penyebarannya melalui perairan atau menular dari ikan lain. Cirinya sirip sobek-sobek dan lender meningkat. Cara penanganannya dengan cara direndam dalam air laut yang sudah diberi formalin 25-30 ppm selama 1-2 hari. Sehat 70.3 Sakit 73.4 Ok Apakah Anda yakin akan menggunakan gambar ini?

Dari 3 pengujian diatas terlihat bahwa selain ikan kakap gambar diatas juga mempunyai nilai yang dapat terdeteksi oleh aplikasi, karena gambar diatas mempunyai nilai yang sama dengan yang ada pada database.

4.3 Hasil Pengujian

Gambar yang diupload oleh user dan gambar yang ada di database diubah menjadi *hash image*. *Hash image* dari gambar yang diupload diubah dibandingkan dengan hash image dari gambar yang ada di database untuk mendapatkan informasi index kemiripan (similarity). Nilai dari index kemiripan diurutkan berdasarkan pada nilai yang paling tinggi. Gambar dengan Nilai perbandingan yang paling tinggi dari index kemiripan akan dinyatakan sebagai penyakit yang diderita oleh ikan. Setelah diperoleh nilai perbandingan paling tinggi, maka sistem akan mengunci penyakit ikan yang diderita untuk sementara. Selanjutnya sistem akan mengubah database gambar kondisi sehat menjadi *hash image*. *Hash image* yang diupload kemudian dibandingkan dengan database gambar kondisi sehat. Jika ditemukan *hash image* yang diupload memiliki kemiripan yang lebih tinggi terhadap *hash image* kondisi sehat, maka ikan dinyatakan sehat. Jika ditemukan hash gambar yang diupload memiliki kemiripan yang lebih tinggi terhadap gambar kondisi sakit, maka ikan dinyatakan sakit.

Contoh perhitungan.

Diketahui:

Index_sakit = *Index of Similarity* terbesar dari perbandingan gambar yang diupload dan database gambar sakit.

Index_sehat = *Index of Similarity* terbesar dari perbandingan gambar yang diupload dan database gambar sehat.

Pada contoh ini akan dilakukan pengujian terhadap gambar insang sehat.

Tabel 4.2 Perbandingan *Index similarity* sebelum diurutkan.

No	Nama Penyakit	Index Similarity
1	Ekor Sakit	68
2	Sirip Sakit	65
3	Insang Sakit	74
4	Badan sakit	70
5	Anus sakit	63

Tabel 4.3 Perbandingan *index similarity* setelah diurutkan.

No	Nama Penyakit	Index Similarity
1	Insang Sakit	74
2	Badan sakit	70
3	Ekor Sakit	68
4	Sirip Sakit	65
5	Anus sakit	63

Dari tabel diatas maka dapat disimpulkan penyakit yang paling mungkin diderita oleh ikan tersebut adalah penyakit Insang sakit. Sehinggai diperoleh nilai index_sakit=74. Namun demikian, sebelum diambil kesimpulan gambar yang diupload harus terlebih dahulu dibandingkan dengan gambar sehat. Berikut contoh

dari perhitungan index similarity perbandingan antara gambar yang diupload dengan gambar sehat.

Tabel 4.4 Index similarity insang sebelum diurutkan

No	Nama Penyakit	Index Similarity
1	Insang Sehat 1	84
2	Insang Sehat 2	72
3	Insang Sehat 3	94
4	Insang Sehat 4	82
5	Insang Sehat 5	80
6	Insang Sehat 6	86
7	Insang Sehat 7	90
8	Insang Sehat 8	74
9	Insang Sehat 9	70
10	Insang Sehat 10	88

Tabel4.5 Urutan tabel insang sehat berdasarkan *Index Similarity*

No	Nama Penyakit	Index Similarity
1	Insang Sehat 3	94
2	Insang Sehat 7	90

3	Insang Sehat 10	88
4	Insang Sehat 6	86
5	Insang Sehat 1	84
6	Insang Sehat 4	82
7	Insang Sehat 5	80
8	Insang Sehat 8	74
9	Insang Sehat 2	72
10	Insang Sehat 9	70

Dari tabel tersebut ditemukan index_sehat = 94.

Langkah selanjutnya adalah membandingkan index_sakit dengan index_sehat. Berikut hasil perbandingannya.

```
If (index_sakit>index_sehat)
{
    Print  "Penyakit yang diderita adalah insang sakit"
}else
{
    Print "Ikan dalam kondisi sehat";
}
```

Karena diketahu nilai index_sakit=74 sedangkan nilai index_sehat=94, maka dapat disimpulkan bahwa ikan dalam kondisi sehat.