

# PERANCANGAN PERANGKAT LUNAK UNTUK MENGIDENTIFIKASI JENIS IKAN MENGGUNAKAN JARINGAN SARAF TIRUAN

*by* Dr. Aris Triwiyatno, S.t., M.t.

---

**Submission date:** 06-Feb-2020 07:17PM (UTC+0700)

**Submission ID:** 1252508408

**File name:** Paper\_C-5-10.pdf (537.61K)

**Word count:** 2833

**Character count:** 17364

# PERANCANGAN PERANGKAT LUNAK UNTUK MENGIDENTIFIKASI JENIS IKAN MENGGUNAKAN JARINGAN SARAF TIRUAN

Dita Marta Dewi Onasiska<sup>\*)</sup>, Achmad Hidayatno, and Aris Triwiyatno

<sup>\*)</sup> Email: ditamarta15@gmail.com

## 1. Pendahuluan

Umumnya semua jenis ikan berenang dengan membentuk barisan kisi-kisi (*lattice*), belah ketupat (*rhombic lattice*), selain itu ada yang membentuk barisan kisi-kisi bujur sangkar (*cubic lattice*), baik secara vertikal maupun

horizontal. Bentuk dan dimensi dari *schooling* ikan dipengaruhi oleh jenis ikan, kedalaman dan penginderaan terhadap serangan predator[1].

Suara-suara yang dikeluarkan ikan terdiri atas : letupan kecil dari berkas suara *noise* yang lebar, sedangkan

lainnya bernada dan mengandung frekuensi serasi/harmonis dan fundamental. Ikan mengeluarkan suara dengan tiga cara utama, yaitu dengan pergerakan tulang (*stridulasi*), vibrasi gelembung renang dan efek yang terjadi karena aktifitasnya yang termasuk kedalam golongan suara hidrodinamik[2].

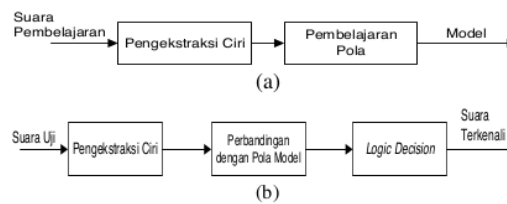
Adapun Yunanto Widyawati pada tahun 2009 telah berhasil merancang perangkat lunak untuk menentukan jenis ikan secara real-time menggunakan metode *Hidden Markov Model*[3]. Berdasarkan penelitian tersebut, pada penelitian ini dirancang sebuah perangkat lunak untuk mengidentifikasi ikan melalui pendeteksian fase penerimaan sinyal akustik dengan menggunakan pencirian *Linear Predictive Coding* dan jaringan saraf tiruan *backpropagation* sebagai algoritma pembelajarannya.

Dengan proses diatas diharapkan mampu untuk mengidentifikasi jenis ikan dari suara yang ditangkap melalui sebuah *Hydrophone*. Dengan menggunakan metode ekstraksi pola suara *Linear Predictive Coding* (LPC)[4] akan memudahkan dalam mendapatkan ciri tiap sinyal sebagai proses awal perancangan program. Dari metode ini akan diperoleh nilai koefisien LPC yang merupakan *feature* (ciri) dari suara yang diucapkan.

Untuk proses pengenalan dan pengambilan keputusan dapat digunakan suatu alogaritma Jaringan Saraf Tiruan (JST)[5]. Dalam merancang suatu jaringan saraf selain memperhatikan struktur hubungan antara simpul masukan dengan simpul keluaran, perlu ditentukan juga cara atau metode pembelajarannya. Belajar bagi jaringan saraf tiruan adalah cara memperbaharui bobot sinapsis yang disesuaikan dengan isyarat masukan dan keluaran yang diharapkan.

**2. Metode**  
**2.1 Pengolahan Suara**

Salah satu bentuk pendekatan pengenalan suara adalah pendekatan pengenalan pola yang terdiri dari dua langkah yaitu pembelajaran pola suara dan pengenalan suara melalui perbandingan pola. Tahap perbandingan pola adalah tahap saat suara yang akan dikenali dibandingkan polanya dengan setiap kemungkinan pola yang telah dipelajari dalam fase pembelajaran, untuk kemudian diklasifikasikan dengan pola terbaik yang cocok. Blok diagram pembelajaran pola dan pengenalan suara ditunjukkan pada gambar 1.



**Gambar 1. (a) Blok diagram pembelajaran pola, (b) Blok diagram pengenalan suara**

Pengenalan suara secara umum dapat dibagi menjadi tiga tahap, yaitu tahap ekstraksi ciri, tahap pemodelan atau pembelajaran dan tahap pengenalan suara. Perancangan sistem merupakan tahap yang penting dalam proses pembuatan program. Perancangan bertujuan agar dalam pembuatannya dapat berjalan secara sistematis, terstruktur, dan rapi sehingga hasil program dapat berjalan sesuai dengan apa yang dikehendaki. Dalam perancangan, aspek yang diperhatikan meliputi kemungkinan pengembangan di masa depan, efektif dan efisien, kemampuan program dan kemudahan untuk dipahami pengguna (*user friendly*) yang diwujudkan dalam tampilan grafis (*Graphical User Interface*). Secara umum pembuatan program ini mengikuti alur sesuai yang ditunjukkan pada Gambar 2.



**Gambar 2. Diagram perancangan sistem**

Pembuatan sistem pengenalan suara jenis ikan ini terdiri dari proses pengelolaan basis data, proses ekstraksi ciri, proses pemodelan dan pelatihan. Ketiga proses tersebut sering disebut dengan proses pembelajaran. Proses pengenalan suara jenis ikan ini merupakan proses perbandingan antara suara uji dengan model suara yang didapat dari proses pembelajaran.

Dalam pembuatan basis data dilakukan perekaman suara dengan 3 jenis ikan yang berbeda yaitu ikan koki (*carrasius auratus*), ikan balon (*poelicia latipinna sailfin molly*) dan ikan patin (*pangsius2*). Setiap jenis ikan melakukan perekaman sebanyak 50 kali masing-masing 35 untuk data latih dan 15 untuk data uji. Sehingga data latih dan data uji jumlahnya sebanyak 150. Perekaman suara dalam pembuatan basis data ini dilakukan di dalam sebuah studio musik selama 10 detik untuk setiap data suara, hal ini bertujuan untuk mengurangi adanya noise saat proses perekaman.

Pencuplikan dilakukan pada frekuensi 44100 Hz dengan resolusi 16 bit. Kecepatan pencuplikan tersebut dilakukan karena spesifikasi dari *sound card* komputer yang dipakai menggunakan pencuplikan pada frekuensi 44100 Hz, dan dengan didasarkan asumsi bahwa sinyal suara ikan berada pada daerah frekuensi 50-3800 Hz sehingga memenuhi kriteria Nyquist yang menyatakan bahwa frekuensi penyampling minimal 2 kali frekuensi maksimal dari sinyal yang akan disampling untuk menghindari *aliasing*. Agar dapat diolah dengan program Matlab, suara yang telah direkam dan disimpan dalam bentuk file '.wav' perlu dipanggil dalam *workspace*.

Suara memiliki amplitudo yang bervariasi tergantung pada keras lemahnya suara tersebut pada saat diucapkan. Normalisasi suara dilakukan untuk menyamakan rentang amplitudo suara dengan membagi nilai-nilai amplitudo dengan nilai amplitudo tertinggi. Sehingga diperoleh nilai hasil normalisasi suara dengan nilai maksimum adalah 1 dan nilai minimum adalah -1.

Dalam suatu masukan data suara, nilai yang akan diproses pada umumnya terletak di tengah-tengah. Untuk mengatasi kelebihan bagian bukan suara (bagiankosong yang terdapat pada awal dan akhir sebuah berkassuara) perlu dilakukan proses deteksi ujung suara.

## 2.2 Ekstraksi Ciri dengan LPC

Langkah-langkah analisis LPC untuk mendapatkan koefisien LPC pada proses ekstraksi ciri suara adalah [4] :

### 1. Preemphasis

Sinyal suara  $s(n)$  dimasukkan ke dalam sistem digital orde rendah (biasanya berupa filter FIR orde satu) yang digunakan untuk meratakan spektrum sinyal. Keluaran dari rangkaian *preemphasis*  $\tilde{s}(n)$  adalah :

$$\tilde{s}(n) = s(n) - \tilde{a}s(n-1), 0,9 \leq \tilde{a} \leq 1,0 \quad (1)$$

### 2. Frame blocking

Sinyal suara hasil dari proses *preemphasis*  $\tilde{s}(n)$  diblok atau dibagi ke dalam beberapa *frame* yang terdiri dari  $N$ -sampel suara, dengan jarak antara *frame* yang berdekatan dipisahkan oleh  $M$ -sampel. Jika  $M \leq N$ , beberapa *frame* yang berdekatan akan saling *overlap* dan hasil estimasi spektral LPC akan berkorelasi dari *frame* ke *frame*. Sebaliknya, jika  $M > N$ , tidak akan ada *overlap* antara *frame* yang berdekatan sehingga beberapa isyarat sinyal suara akan hilang total.

### 3. Windowing (penjendeleaan)

*Windowing* digunakan untuk mengurangi *discontinuitas* sinyal pada awal dan akhir *frame*. Jika *window* didefinisikan sebagai  $w(n)$ , maka hasil dari penjendeleaan sinyal adalah :

$$\tilde{x}_i(n) = \tilde{s}(n)w(n), 0 \leq n \leq N-1 \quad (2)$$

Jenis *window* yang biasa digunakan adalah *Hamming window* yang mempunyai bentuk umum:

$$w(n) = 0,54 - 0,46 \cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1 \quad (3)$$

### 4. Analisis autokorelasi

Setiap *frame* dari sinyal setelah melalui proses *windowing*, kemudian dilakukan analisis autokorelasi sebagai berikut :

$$r_l(m) = \sum_{n=0}^{N-1-m} \tilde{x}_l(n)\tilde{x}_l(n+m), m = 0, 1, \dots, p(4)$$

dengan nilai autokorelasi tertinggi  $p$  adalah orde LPC. Nilai  $p$  biasanya antara 8 sampai 16.

### 5. Analisis LPC

Proses selanjutnya adalah analisis LPC, yang mengubah setiap *frame* autokorelasi  $p+1$  ke dalam bentuk parameter-parameter LPC atau yang biasa disebut dengan koefisien LPC. Metode yang biasa digunakan dalam analisis LPC ini adalah metode Durbin yang mempunyai algoritma:

$$E^{(0)} = r(0) \quad (5)$$

$$k_i = \left\{ r(i) - \sum_{j=1}^{i-1} \alpha_j^{(i-1)} r(i-j) \right\} / E^{(i-1)}, 1 \leq i \leq p \quad (6)$$

$$\alpha_i^{(i)} = k_i \quad (7)$$

$$\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \alpha_{i-j}^{(i-1)}, 1 \leq j \leq i-1 \quad (8)$$

$$E^{(i)} = (1 - k_i^2) E^{(i-1)} \quad (9)$$

Dengan menyelesaikan persamaan 5 sampai 9 secara rekursif untuk  $i = 1, 2, \dots, p$ , koefisien LPC diperoleh :

$$a_m = \alpha_m^{(p)}, 1 \leq m \leq p \quad (10)$$

6. Konversi parameter LPC menjadi koefisien *cepstral* Rangkaian parameter yang sangat penting yang dapat diturunkan secara langsung dari rangkaian koefisien LPC adalah koefisien *cepstral*  $c_m$ , yang ditentukan secara rekursi:

$$c_0 = \ln \sigma^2 \quad (11)$$

$$c_m = a_m + \sum_{k=1}^{m-1} \left( \frac{k}{m} \right) c_k a_{m-k}, \quad 1 \leq m \leq p \quad (12)$$

$$c_m = \sum_{k=1}^{m-1} \left( \frac{k}{m} \right) c_k a_{m-k}, \quad m > p \quad (13)$$

dengan  $\sigma$  adalah *gain* dari LPC. Koefisien *cepstral* yang digunakan direpresentasikan dengan  $Q$ , dimana koefisien yang digunakan biasanya adalah  $Q > p$ , yaitu  $Q = (3/2)p$ .

7. Pembobotan Parameter

Pembobotan dilakukan pada parameter karena sensitifitas koefisien *cepstral* orde rendah pada keseluruhan slope spektral dan sensitifitas koefisien *cepstral* orde tinggi pada derau. Pembobotan dilakukan dengan memberikan jendela tapis pada koefisien *cepstral*. Bentuk koefisien *cepstral* setelah pembobotan adalah :

$$\tilde{c}_m = w_m c_m, \quad 1 \leq m \leq Q \quad (14)$$

dengan  $w_m$  adalah jendela pembobotan atau disebut *bandpass lifter* (tapis pada domain *cepstral*). Besarnya  $w_m$  adalah :

$$w_m = \left[ 1 + \frac{Q}{2} \sin \left( \frac{\pi m}{Q} \right) \right], \quad 1 \leq m \leq Q \quad (15)$$

8. Turunan Temporal Koefisien *Cepstral*

Turunan koefisien *cepstral* (*delta cepstral*) meningkatkan keterwakilan sifat-sifat spectral sinyal yang dianalisis pada parameter. Turunan koefisien *cepstral* dapat dituliskan:

$$\frac{\partial c_m(t)}{\partial t} = \Delta c_m(t) \approx \left( \sum_{k=1}^K k c_m(t+k) \right) / \left( \sum_{k=1}^K k^2 \right) \quad (16)$$

dengan  $(2K + 1)$  adalah jumlah frame.

Pada perancangan ini proses *frame blocking* ditetapkan tiap 30 mili detik dengan jarak antar *frame* 10 mili detik. Jadi dengan kecepatan cuplik sebesar 44100 Hz maka tiap *frame* akan berisi 1323 *byte* data dengan jarak antar *frame* 441 *byte* data atau dengan kata lain *overlap* yang terbentuk sebesar 882 *byte* data. Dengan ketentuan *frame* seperti di atas, maka untuk data hasil cuplik sebanyak 350000 data (pembulatan dari hasil sekitar 350000) maka akan terbentuk  $\frac{350000-882}{441} = 790$  frame (hasil pembulatan dari 789,85). Untuk perhitungan koefisien LPC, digunakan orde LPC 12 sehingga diperoleh data output sebanyak  $(12+1) \times 790 = 10270$  data.

### 2.3 Analisa Frekuensi Sinyal dengan Transformasi Fourier Cepat

Transformasi Fourier Cepat atau *Fast Fourier Transform* (FFT) merupakan penyederhanaan dari *Discrete Fourier Transform* (DFT) [6].

$$X_1(k) = \sum_{n=0}^{N-1} x_n e^{-j2\pi n k / N}, \quad k = 0, 1, \dots, N-1 \quad (17)$$

Faktor  $e^{-j2\pi n k / N}$  akan ditulis sebagai  $W_N$ , maka :

$$W_N = e^{-j2\pi / N} \quad (18)$$

Sehingga persamaan (17) menjadi :

$$X_1(k) = \sum_{n=0}^{N-1} x_n W_N^{kn}, \quad k = 0, 1, \dots, N-1 \quad (19)$$

Proses *Fast Fourier Transform* (FFT) yang digunakan memakai 2048, karena hasil FFT simetris maka keluaran FFT tersebut hanya diambil setengah saja yaitu 1024 yang dianggap sudah dapat merepresentasikan ciri dari sinyal suara dalam domain frekuensi. Dari data FFT ini kemudian dijadikan masukan bagi jaringan saraf tiruan.

### 2.4 Pelatihan Jaringan Saraf Tiruan Backpropagation

Jaringan saraf tiruan perambatan balik merupakan salah satu model dari jaringan saraf tiruan umpan maju dengan menggunakan pelatihan terbimbing yang disusun berdasar pada algoritma *error back propagation*. Pola masukan dan target diberikan sebagai sepasang data. Bobot-bobot awal dilatih dengan melalui tahap maju untuk mendapatkan galat keluaran yang selanjutnya galat ini digunakan dengan tahap mundur untuk memperoleh nilai bobot yang sesuai agar dapat memperkecil nilai galat sehingga target keluaran yang dikehendaki tercapai. Tujuan dari model ini adalah untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama proses pelatihan berlangsung serta kemampuan jaringan memberikan respon yang benar terhadap pola masukan yang berbeda dengan pola masukan selama pelatihan [5].

Setelah melalui tahap ekstraksi ciri LPC selanjutnya parameter-parameter yang dihasilkan dimasukkan ke dalam jaringan saraf tiruan dengan menggunakan metode pembelajaran *Backpropagation*. Proses pelatihan jaringan saraf tiruan dilakukan dengan mengambil *input* hasil proses ekstraksi ciri LPC dari seluruh suara hasil perekaman basis data.

Proses pelatihan diawali dengan pembentukan jaringan serta penentuan struktur jaringan. Dalam melakukan pembentukan jaringan kita perlu menentukan beberapa

parameter jaringan yang dibutuhkan, diantaranya jumlah *layer* yang digunakan, jumlah *neuron* yang digunakan pada tiap *layer*, fungsi aktivasi yang digunakan, fungsi *training* yang digunakan pada saat pelatihan, dan juga parameter performansi jaringan. Adapun *property* jaringan *backpropagation* yang digunakan dalam penelitian ini adalah sebagai berikut:

- Fungsi aktifasi = logsig
- Fungsi pelatihan = trainrp

Pada proses pelatihan jaringan, data akhir hasil transformasi fourier dari setiap sinyal suara pada *database* diformulasikan sebagai vektor masukan pelatihan. Selain menentukan masukan pelatihan kita juga harus menentukan target pelatihan. Dalam sistem ini, fungsi aktivasi yang digunakan ialah *sigmoid biner* yang memiliki range keluaran antara 0 sampai 1, sehingga keluaran yang telah ditentukan juga harus bernilai antara 0 sampai 1. Berikut adalah penentuan target dari setiap jenis ikan.

- Ikan Mas Koki = [1; 0; 0]
- Ikan Balon = [0; 1; 0]
- Ikan Patin = [0; 0; 1]

Selanjutnya keluaran pelatihan tersebut disimpan dalam variabel target. Variabel masukan akan memiliki jumlah baris dan jumlah kolom yang tergantung dari jumlah *database* pelatihan. Begitu juga dengan variabel target akan memiliki 35 baris dan jumlah kolom yang sesuai dengan variabel masukan. Variabel masukan disimpan dan diberi nama 'masukan.mat'. Sedangkan variabel target disimpan dan diberi nama 'target.mat'.

Setelah semua data pelatihan yang terdiri dari masukan dan target telah ditentukan, maka langkah selanjutnya ialah menentukan parameter-parameter pelatihan yang dapat kita atur melalui GUI (*Graphical User Interface*) pada aplikasi bagian 'Latih Jaringan'. Pengaturan parameter diantaranya meliputi penentuan jumlah *hidden layer*, laju pembelajaran, nilai MSE (*Mean Square Error*) pelatihan serta penentuan jumlah iterasi pelatihan.

Parameter jaringan yang berpengaruh dalam mengatur kinerjanya adalah jumlah *hiddenlayer*, jumlah *hidden neuron* serta laju pembelajaran. Tahap selanjutnya menentukan nilai laju pembelajaran. Laju pembelajaran yang digunakan adalah 0,001. Laju pembelajaran yang terlalu kecil akan menyebabkan waktu yang lama untuk mencapai keadaan yang diinginkan, sedangkan laju pembelajaran yang terlalu besar akan menyebabkan jaringan sulit mencapai hasil yang diinginkan.

Pada parameter jaringan, jumlah iterasi yang dipilih adalah 1000. Sedangkan nilai MSE yang ditentukan sebesar  $1e-50$ . Dengan nilai MSE yang sangat kecil,

diharapkan jaringan paling tidak akan mendapatkan nilai MSE yang mendekati nilai MSE yang ditentukan.

## 2.5 Pengambilan Keputusan

Pengambilan keputusan dilakukan pada keluaran dari JST propagasi balik yang akan disimulasikan dengan jaringan yang telah dilatih. Hasil simulasi akan dibandingkan dengan target yang telah ditentukan. Jika hasil simulasi sesuai dengan target maka suara dikenali benar. Namun jika tidak sesuai dengan target akan dikenali salah. Diagram alir proses pengenalan jenis ikan ditunjukkan oleh Gambar 3.



Gambar 3. Diagram alir pengenalan jenis ikan

Proses pengenalan jenis ikan diawali dengan memanggil *file* suara .wav atau dengan pengujian langsung menggunakan *hydrophone*. Kemudian dilakukan proses normalisasi suara, deteksi ujung. Proses selanjutnya

terjadi proses ekstraksi ciri LPC dan FFT dari suara uji hingga diperoleh data akhir (setengah data keluaran FFT) yang akan digunakan sebagai masukan untuk disimulasikan dengan menggunakan jaringan yang telah tersimpan.

### 3. Hasil dan Analisa

Pengujian dilakukan menggunakan data latih, data uji sesuai dan pengujian langsung. Pengujian dilakukan dengan 6 variasi jaringan yaitu menggunakan 3, 4, 5, 6, 7 dan 8 *hidden layer*.

#### 3.1 Pengujian Data Latih

Pada pengujian data latih, data suara yang digunakan untuk melakukan pengujian merupakan data suara dari 3 jenis ikan yang telah direkam sebagai *database*. Perekaman dilakukan selama 10 detik pada frekuensi 44100 Hz dan resolusi 16 bit. Tiap jenis ikan diambil 50 kali data suara. Masing-masing 35 data digunakan sebagai data latih. Hasil pengenalan pada data latih dapat dilihat pada Tabel 1.

Tabel 1. Hasil pengenalan ikan pada pengujian data latih

Jenis Ikan	Nama Jaringan	Persentase Benar
Ikan Koki	jaringansatu	100%
	jaringandua	100%
	jaringantiga	100%
	jaringanempat	100%
	jaringanlima	100%
	jaringanenam	100%

Tabel 1. Hasil pengenalan ikan pada pengujian data latih (lanjutan)

Jenis Ikan	Nama Jaringan	Persentase Benar
Ikan Balon	jaringansatu	100%
	jaringandua	100%
	jaringantiga	100%
	jaringanempat	100%
	jaringanlima	100%
	jaringanenam	100%
Ikan Patin	jaringansatu	100%
	jaringandua	100%
	jaringantiga	100%
	jaringanempat	100%
	jaringanlima	100%
	jaringanenam	100%

Hasil pengujian menggunakan data latih tidak menunjukkan perbedaan hasil pengujian. Hal ini disebabkan karena data yang digunakan untuk menguji jaringan adalah data latih. Selisih nilai MSE akhir dari tiap-tiap jaringan yang sangat kecil juga ikut mempengaruhi hasil pengujian jaringan.

#### 3.2 Pengujian Data Uji

Pada pengujian data uji, data suara yang digunakan untuk melakukan pengujian merupakan data suara dari 3 jenis ikan. Perekaman dilakukan selama 10 detik pada frekuensi 44100 Hz dan resolusi 16 bit. Setiap jenis ikan memiliki 15 data yang telah direkam untuk data uji. Hasil pengenalan pada data uji dapat dilihat pada Tabel 2.

Tabel 2. Hasil pengenalan ikan pada pengujian data uji

Jenis Ikan	Nama Jaringan	Persentase Benar
Ikan Koki	jaringansatu	46,6%
	jaringandua	40%
	jaringantiga	66,6%
	jaringanempat	80%
	jaringanlima	73,3%
	jaringanenam	46,6%
Ikan Balon	jaringansatu	86,6%
	jaringandua	73,3%
	jaringantiga	80%
	jaringanempat	66,6%
	jaringanlima	53,3%
	jaringanenam	66,6%
Ikan Patin	jaringansatu	66,6%
	jaringandua	80%
	jaringantiga	60%
	jaringanempat	73,3%
	jaringanlima	60%
	jaringanenam	73,3%

Hasil pengujian data uji berdasarkan pengenalan jenis ikan nilai tertinggi diperoleh pada pengenalan ikan balon (*poelicia latipinna sailfin molly*) dengan persentase kebenaran 86,6% menggunakan jaringan jaringansatu. Sedangkan hasil pengujian terendah berdasarkan pengenalan jenis ikan adalah pengenalan ikan koki (*carrasius auratus*) dengan persentase kebenaran 40% menggunakan jaringan jaringandua. Hal ini membuktikan penambahan jumlah *hidden layer* tidak terlalu berpengaruh terhadap hasil data uji.

#### 3.3 Pengujian Langsung

Pada pengujian langsung, data suara yang digunakan adalah data suara yang diambil atau direkam secara langsung menggunakan *hydrophone*. Jumlah data suara untuk pengujian langsung sejumlah 10 data suara. Total keseluruhan terdapat 30 data suara. Perekaman dilakukan di dalam ruangan bukan studio musik selama 10 detik pada frekuensi 44100 Hz dan resolusi 16 bit. Hasil pengenalan pengujian langsung dapat dilihat pada Tabel 3.

Tabel 3. Hasil pengenalan ikan pada pengujian langsung

Jenis Ikan	Nama Jaringan	Persentase Benar
Ikan Koki	jaringansatu	50%
	jaringandua	30%
	jaringantiga	40%
	jaringanempat	60%
	jaringanlima	30%
Ikan Balon	jaringanenam	50%
	jaringansatu	40%
	jaringandua	40%
	jaringantiga	40%
	jaringanempat	30%
Ikan Patin	jaringanlima	40%
	jaringanenam	30%
	jaringansatu	40%
	Jaringandua	50%
	jaringantiga	40%
	jaringanempat	60%
	jaringanlima	50%
	jaringanenam	30%

Hasil pengujian langsung memiliki nilai persentasi yang bervariasi. Hasil tertinggi sebesar 50%, sedangkan nilai persentasi terendah sebesar 30% didapatkan pada beberapa jenis ikan. Hal ini membuktikan penambahan jumlah *hidden layer* tidak terlalu berpengaruh terhadap hasil pengujian langsung.

#### 4. Kesimpulan

Dari hasil penelitian dan pembahasan dapat disimpulkan bahwa metode untuk mengenali ikan menggunakan jaringan saraf tiruan sudah cukup baik. Namun karena sistem sangat peka terhadap sinyal derau, maka perlunya ditambahkan pengolahan peredaman derau sehingga data suara akan lebih baik. Dapat ditinjau lebih jauh pula hasil pengenalan yang dihasilkan jika menggunakan algoritma pelatihan selain *trainrp* dan metode pengenalan lain selain *Backpropagation*.



# PERANCANGAN PERANGKAT LUNAK UNTUK MENDENTIFIKASIJENIS IKAN MENGGUNAKAN JARINGAN SARAF TIRUAN

## ORIGINALITY REPORT

6%

SIMILARITY INDEX

5%

INTERNET SOURCES

2%

PUBLICATIONS

0%

STUDENT PAPERS

## PRIMARY SOURCES

1

[elektro.undip.ac.id](http://elektro.undip.ac.id)

Internet Source

5%

2

Mukson Hudi. "PENGUNAAN METODE EKSPERIMEN UNTUK MENINGKATKAN MOTIVASI DAN HASIL BELAJAR PESERTA DIDIK KELAS III PADA MATA PELAJARAN IPA MATERI MAKHLUK HIDUP SD NEGERI KEDUNGBUNGKUS 02 KECAMATAN TARUB KABUPATEN TEGAL TAHUN PELAJARAN 2015/2016", PSEJ (Pancasakti Science Education Journal), 2017

Publication

1%

Exclude quotes  On

Exclude matches  < 15 words

Exclude bibliography  On

# PERANCANGAN PERANGKAT LUNAK UNTUK MENGIDENTIFIKASIJENIS IKAN MENGGUNAKAN JARINGAN SARAF TIRUAN

---

## GRADEMARK REPORT

---

FINAL GRADE

**/0**

GENERAL COMMENTS

**Instructor**

---

PAGE 1

---

PAGE 2

---

PAGE 3

---

PAGE 4

---

PAGE 5

---

PAGE 6

---

PAGE 7

---