

ANALISIS PENGGUNAAN ALGORITMA GENETIKA UNTUK MENINGKATKAN PERFORMANSI DARI KLASIFIKASI GENRE MUSIK BERBASIS JARINGAN SYARAF TIRUAN BACK-PROPAGATION

ANALYSIS OF THE USE OF GENETIC ALGORITHM TO IMPROVE THE PERFORMANCE OF MUSICAL GENRE CLASSIFICATION BASED NEURAL NETWORK BACK-PROPAGATION

Atiffan Ramadhiah

Iwan Iwut Tritoasmoro
,IR., MT²Inung Wijayanto, ST., MT³¹ Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom^{2,3} Fakultas Teknik Elektro, Universitas Telkom

Jl. Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia

1) ABSTRAK

Pengolahan Sinyal *Digital* pada sinyal audio berkembang pesat untuk menghasilkan sebuah sistem yang bekerja otomatis. Sehingga diperlukan suatu pengembangan metode dan algoritma yang dapat mengklasifikasi genre secara tepat. Beberapa penelitian sebelumnya sudah menggunakan Jaringan Syaraf Tiruan, *Support Vector Machine*, *Hidden Markov Model*, dan *Continous Density Hidden Markov Model* sebagai metode klasifikasi. Pada penelitian sebelumnya, digunakan Jaringan Syaraf Tiruan *Backpropagation* dengan akurasi 67%. Selanjutnya digunakan algoritma genetika dalam tugas akhir ini untuk klasifikasi *genre* yang memiliki kualitas yang baik dalam ketepatan klasifikasinya dengan menggunakan ciri konten frekuensi dan klasifikasi menggunakan jaringan syaraf tiruan *backpropagation*. Setelah dilakukan pengujian terhadap sistem yang telah dirancang. Parameter yang menghasilkan akurasi maksimal yaitu jumlah *hidden layer* 1, jumlah *neuron* tiap *layer* 20, nilai *learning rate* 0.05, fungsi aktivasi *tansig* untuk *hidden layer*, fungsi aktivasi *purelin* untuk *output layer*, algoritma pembelajaran *trainrp* dengan akurasi 77.77% dari data latih sebanyak 150 data dan 150 data uji. Parameter Algoritma Genetika meningkatkan akurasi menjadi 85,55% dengan parameter jumlah generasi 100, jumlah individu 50, peluang crossover 0.6, dan peluang permutasi 0.01.

Kata Kunci: Musik, Genre, Jaringan Syaraf Tiruan *back-propagation*, Algoritma Genetika.

ABSTRACT

Music has a wide range of genres and some examples are Pop, Rock, and Dance. Man in distinguishing the genre typically see with the characteristics of the music and the type of instrument being played. He sometimes easy to distinguish a genre of music, but a system or machine is sometimes difficult to distinguish the genre of a music file. Digital Signal Processing in the rapidly evolving audio signal to produce a system that works automatically. So we need a development of methods and algorithms that can accurately classify genre. Several previous studies have used Artificial Neural Networks, Support Vector Machine, Hidden Markov Models, and Continuous Density Hidden Markov Model as a method of classification. In previous studies, use Neural Network Backpropagation with an accuracy of 67%. Furthermore, genetic algorithm is used in the final work ini untuk genre classification that has good quality in classification accuracy by using a characteristic frequency content and classification using neural networks backpropagation. After testing the system that has been designed. Parameters that produces maximum accuracy ie the number of hidden layer 1, the number of neurons of each layer 20, the value of learning rate 0.05 activation function *tansig* for the hidden layer, the activation function *purelin* for the output layer, learning algorithm *trainrp* with an accuracy of 77.77% of the training data 150 training data and 150 testing data. Parameter Genetic Algorithms improve accuracy becomes 85.55% by the parameter generation number 100, the sheer number of people 50, 0.6 crossover opportunities, and opportunities permutation 0:01.

Keywords : Music, Genre, backpropagation Neural Networks, Genetic Algorithms

1. Pendahuluan

Perkembangan yang pesat pada audio *processing* dirasakan banyak membantu dalam memajukan perkembangan musik digital. Musik terdiri dari berbagai macam *genre* dan jenis sesuai dengan konten musik tersebut. Perkembangan musik digital terutama pada klasifikasi *genre* dirasakan telah membantu dalam kemudahan mempelajari dan mencari suatu lagu. Hal tersebut mendorong diciptakannya kemudahan dalam variasi klasifikasi *genre* yang mampu mengoptimalkan proses pembelajaran yang dapat dilakukan dengan mudah, *simple* dan memiliki kualitas yang baik dalam ketepatan pencarian suatu lagu. Sehingga diperlukan suatu pengembangan proses pembelajaran tersebut dengan berbagai metode dan algoritma yang lebih baik. Dan dalam perkembangannya dibatasi terlebih dahulu hanya pada klasifikasi *genre* yang memiliki kualitas yang baik dalam ketepatan klasifikasinya.

Dalam beberapa tahun terakhir, penelitian tentang Algoritma Genetika (AG) banyak ditemukan dalam makalah penelitian (Chu, 2003a, Chen, 2003; Chu, 2003b). Pada makalah tersebut ditunjukkan karakteristik yang berbeda dalam algoritma genetika dibandingkan yang lainnya. Dalam operasi algoritma genetika, hanya perlu menetapkan fungsi tujuan tanpa operasi tambahan, seperti operasi differensial. Oleh karena itu, dapat digunakan untuk fungsi obyektif

untuk semua jenis masalah. Jaringan syaraf tiruan yang paling sering digunakan pada pengenalan suara pada penelitian sebelumnya adalah jaringan syaraf tiruan *back-propagation* / propagasi balik.

Pada penelitian sebelumnya, digunakan Jaringan Syaraf Tiruan *Backpropagation* dengan akurasi 67% [6]. Selanjutnya digunakan algoritma genetika dalam tugas akhir ini untuk klasifikasi *genre* yang memiliki kualitas yang baik dalam ketepatan klasifikasinya dengan menggunakan ciri konten frekuensi dan klasifikasi menggunakan jaringan syaraf tiruan *backpropagation*. Penggunaan metode ini dapat menghasilkan akurasi terbaik yang akan didapat dari hasil pengujian klasifikasi *genre* lagu dari data pada penelitian sebelumnya yang akan dikelompokkan menjadi data latih yang merupakan database lagu acuan dan data uji yang merupakan data yang akan diuji ketepatan klasifikasi *genre*. Akurasi yang diharapkan adalah diatas 67% terhadap tiga *genre* lagu yaitu *Rock*, *Pop*, dan *Dance*.

2. Perancangan Sistem

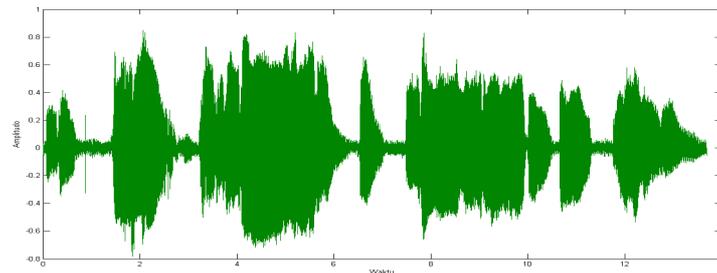
Klasifikasi genre musik berbasis jaringan syaraf tiruan *back propagation* yang dirancang terdiri dari 2 proses yaitu proses latih dan proses uji. Algoritma genetika digunakan untuk mengoptimasi pelatihan jaringan syaraf tiruan *back propagation* untuk mendapat akurasi uji terbaik. Alur kerja sistem dalam tugas akhir ini dapat dilihat dari gambar di bawah ini.

Gambar 2 Perancangan Sistem

Proses latih merupakan proses pembentukan sistem klasifikasi berdasarkan data latih sebagai acuan. Dalam hal ini data latih pada sistem adalah data lagu mp3 yang terdiri dari 300 lagu. Sedangkan proses uji merupakan proses sesungguhnya sistem yang telah dirancang pada proses latih untuk mengklasifikasi ketepatan lagu dari data uji yang dipilih pada proses uji. Proses latih dan uji secara garis besar sama hanya saja pada proses latih berarti membangun sistem klasifikasi menggunakan Jaringan Syaraf Tiruan *Back Propagation* yang menghasilkan parameter JST terlatih yaitu parameter JST yang digunakan pada proses uji klasifikasi. Algoritma Genetika digunakan untuk mengoptimasi pelatihan jaringan syaraf tiruan *back propagation* untuk mendapat akurasi uji terbaik yang terbaik pada proses uji.

2.1 Akuisisi Data

Akuisisi data berupa data lagu dan suara senandung. Data lagu menggunakan frekuensi *sampling* sebesar 44100 sampel/detik sesuai dengan data mp3, Berikut contoh gambar sinyal hasil proses ini:



Gambar 2.1 Sinyal Hasil Tahap Akuisisi Data

2.2 Preprocessing[2]

Setelah diakuisisi, data masuk ke dalam tahap *preprocessing*. Tahap ini bertujuan untuk memperbaiki kualitas data sebelum dilakukan ekstraksi ciri.

2.3 Ekstraksi Ciri[5]

Ekstraksi ciri merupakan proses untuk mendapatkan ciri dari sinyal musik yang menjadi masukan sistem klasifikasi KNN. Ciri yang didapatkan dalam bentuk angka. Dalam tahap ini digunakan konten frekuensi yang terdiri dari:

1. Strength of Half Beat
2. Bass Frequency Variation
3. High Frequency Strength of Half Beat
4. Mid Frequency Beat Likelihood
5. Mid Frequency Beat Offset
6. Mid Frequency Variation
7. Dynamic Range
8. Spectral Power – low
9. Spectral Power – lowmid
10. Spectral Power – mid

11. Spectral Power – high
12. Attack Velocity – fast
13. Attack Velocity – slow

Untuk mendapatkan 13 ciri yang digunakan dilakukan beberapa proses.

2.4 JST-BP

2.4.1 Pelatihan

Dalam tahap ini dilakukan pelatihan JST-BP. Nilai-nilai ciri data latih dipakai untuk melatih JST- BP sehingga jaringan tersebut dapat digunakan untuk pengklasifikasian pada tahap selanjutnya. Dalam pelatihnnya dilakukan inisialisasi jumlah target, *hidden layer*, fungsi aktivasi, algoritma pelatihan dan jumlah *epoch*. Dikarenakan data lagu terdapat 200 lagu maka untuk target akan digunakan bernilai 1 sampai 200.

2.4.2 Pengujian

Dalam tahap ini dilakukan pengklasifikasian data uji. Nilai-nilai ciri data uji diolah ke dalam JST-BP terlatih. Kemudian JST-BP terlatih mengeluarkan hasil kelas lagu untuk data uji tadi.

2.5 Optimasi Algoritma Genetika

Pada penelitian sebelumnya algoritma klasifikasi langsung digunakan dengan penentuan parameter secara trial and error. Pada tugas akhir inidilakukan optimasi menggunakan algoritma genetika pada metode klasifikasi seperti yang terlihat pada gambar 3.10. Penentuan parameter-parameter akan dioptimasi menggunakan algoritma genetika secara tepat untuk masing-masing algoritma klasifikasi diharapkan dapat meningkatkan akurasi uji. Kemudian parameter algoritma klasifikasi hasil dari proses optimasi ini yang akan digunakan pada proses pengujian sistem secara keseluruhan.

Gambar 2.5 Optimasi menggunakan Algoritma Genetika

2.5.1 Optimasi JST

Pada jaringan syaraf tiruan, optimasi dilakukan pada parameter-parameter jaringan sehingga didapatkan jaringan yang optimal. Parameter yang dioptimasi adalah jumlah hidden layer, pada tugas akhir ini dibatasi pada pilihan 1 atau 2 hidden layer, jumlah neuron masing-masing hidden layer dan learning rate.

Pengkodean kromosom menggunakan binary encoding dengan panjang kromosom 24 dengan urutan 2 gen pertama adalah ukuran frame, gen 3 adalah pemilihan banyak ciri, gen ke-4 sampai ke-10 adalah learning rate, 6 nilai biner akan diterjemahkan menjadi nilai real dengan range [0,1]. Gen ke-11 dan gen ke-18 adalah penentu jumlah hidden layer, jika keduanya bernilai 1 maka jumlah hidden layer adalah 2, jika salah satunya bernilai 1 maka jumlah hidden layer yang digunakan 1, namun jika keduanya bernilai 0 maka kromosom tersebut tidak akan digunakan. Gen ke-12 hingga ke-17 dan gen ke-18 hingga gen ke-24 menunjukkan jumlah neuron. Kombinasi 6 nilai biner akan menghasilkan rentang jumlah neuron dari 1 neuron hingga 64 neuron.

Tabel 2.5.1 Desain kromosom parameter JST-BP

Urutan Gen	Definisi
Gen 1-2	Panjang Frame
Gen 3	Jenis Ciri
Gen 4-10	Learning Rate
Gen 11 & 18	Aktivasi Layer
Gen 12-17	Jumlah Neuron
Gen 19-24	

Parameter training lain yang digunakan diantaranya, jumlah epoch=100 dan fungsi aktivasi sigmoid. Parameter algoritma genetika yang digunakan diantaranya, jumlah populasi 25, nilai fitness menggunakan akurasi terhadap data uji, probabilitas pindah silang 0.8, probabilitas mutasi 0.01.

3. Analisis Dan Keluaran Sistem

Pada bab ini dilakukan beberapa pengujian terhadap sistem yang telah dirancang. Setelah dilakukan pengujian maka hasil pengujian tersebut dianalisis dan disimpulkan hasilnya.

Untuk mengetahui performansi sistem yang telah dirancang, maka dilakukan pengujian terhadap sistem dengan beberapa skenario pengujian yaitu:

1. Pengujian dan analisis pengaruh window ekstraksi ciri terhadap akurasi *output* sistem.
2. Pengujian dan analisis pengaruh kombinasi parameter arsitektur JST-BP terhadap akurasi *output* sistem.
3. Pengujian dan analisis pengaruh kombinasi parameter Algoritma Genetika terhadap akurasi *output* sistem.

3.1 Pengaruh Window Ekstraksi Ciri terhadap Akurasi Output Sistem

Dalam skenario ini dilakukan pengujian tiga window ekstraksi ciri *Harmonic* -FFT yaitu 250ms, 500ms, 1000ms, dan 2000ms. Dalam pengujian digunakan data latih sebanyak 150 data lagu dan 150 sebagai data uji. Dari hasil pengujian, dilakukan analisis akurasi menggunakan rumus 3.1. Berikut grafik akurasi *output* sistem dengan menggunakan tiga window ekstraksi ciri:

Gambar 3.1 Pengaruh *Window* Ekstraksi Ciri Terhadap Akurasi *Output* Sistem

Dari gambar 4.7 semakin besar nilai *window* maka nilai akurasi rata-rata semakin naik, hal ini dikarenakan semakin besar nilai *window* maka menghasilkan kecocokan nada yang lebih besar. Dalam penentuan ukuran *window* juga sebaiknya digunakan ukuran *window* yang tepat untuk menghasilkan ekstraksi ciri yang terbaik karena apabila ukuran *window* terlalu kecil menyebabkan waktu proses juga akan semakin lama dan untuk ukuran *window* yang semakin besar maka waktu proses akan cepat namun detail dari ciri suara tiap *frame* akan semakin rendah.

3.2 Analisis Pengaruh Parameter JST Backpropagation

Pada JST *Back Propagation*, terdapat beberapa parameter yang dapat menentukan kinerja JST dalam memproses *input* yang baru. Pada Tugas Akhir parameter JST yang diuji yaitu pengaruh jumlah *hidden layer*, pengaruh *learning rate*, jumlah *neuron* pada masing-masing *layer*, pengaruh fungsi aktivasi pada *hidden layer* dan *outputlayer*, pengaruh algoritma pelatihan (*training*) JST, dan nilai validasi. Dalam pengujian digunakan data latih sebanyak 200 data lagu dan 90 data humming sebagai data uji.

3.2.1 Analisis Pengaruh Jumlah *Hidden Layer* dan *Learning Rate*

Setiap jaringan dapat memiliki lebih dari satu lapisan tersembunyi (*hidden layer*) atau bahkan tidak memilikinya sama sekali. Jika jaringan memiliki beberapa lapisan tersembunyi maka lapisan tersembunyi terbawah berfungsi untuk menerima masukan dari lapisan *input*. Sedangkan *learning rate* digunakan untuk menentukan laju pemahaman. Semakin besar nilai *learning rate* maka akan semakin cepat pula proses pelatihannya. Akan tetapi jika *learning rate* terlalu besar, maka algoritma menjadi tidak stabil. Pada pengujian ini akan diteliti pengaruh dari jumlah *hidden layer* dan *learning rate* pada JST. Dari hasil analisis akan dicari jumlah *hidden layer* dan nilai *learning rate* yang paling cocok untuk diterapkan pada sistem.

Pengujian ini dilakukan dengan mengubah jumlah *hidden layer* dan nilai *learning rate* pada JST yang dibangun. Jenis ekstraksi ciri yang digunakan adalah ciri *window 1000ms*. Parameter JST lain yang digunakan, yaitu fungsi aktivasi *hidden layer* adalah *purelin*, jumlah *neuron* masing-masing *layer* adalah 10, dan fungsi aktivasi *output layer* adalah *purelin*. Algoritma *training* JST yang digunakan yaitu *trainrp*. Parameter lain pada *backpropagation* yang digunakan yaitu nilai maksimum epoch sebanyak 1000, batas toleransi *error* 10^{-5} , gradien minimum 1×10^{-6} . Pada tabel di bawah ini dapat dilihat akurasi yang dihasilkan masing-masing jumlah *hidden layer* dan nilai *learning rate*.

Hidden layer mempengaruhi sistem dalam hal pelatihan jaringan saraf tiruan. Sebuah *hidden layer* sudah cukup bagi *back propagation* untuk mengenali sembarang kesamaan antara masukan dan target dengan tingkat ketelitian tertentu. Penambahan jumlah *hidden layer* seringkali membuat pelatihan menjadi lebih mudah. Namun terkadang semakin besar arsitektur jaringan maka akan menjadi semakin kompleks. Di sisi lain, parameter *learning rate* sangat mempengaruhi proses pelatihan. *Learning rate* yang terlalu besar akan mengakibatkan MSE menurun tajam pada awal iterasi, tetapi akan mengakibatkan MSE menjadi berosilasi atau naik turun tidak terkendali. Sebaliknya, *learning rate* yang terlalu kecil akan mengakibatkan MSE menurun sangat pelan.

Tabel 3.2.1 Akurasi Akibat Pengaruh Jumlah *Hidden layer* dan Nilai *Learning Rate*

<i>Learning Rate</i>	Akurasi (%)		
	<i>Hidden layer</i>		
	1	2	3
0.05	77.77	76.67	70
0.1	76.67	66.67	66.67
0.2	66.67	64.44	66.67
0.3	64.44	60	64.44
0.4	64.44	60	60
0.5	60	60	58

Pada tabel 4.1 terlihat bahwa jumlah *hidden layer* dan nilai *learning rate* dapat mempengaruhi kinerja sistem

sehingga hasil akurasi yang didapatkan cukup beragam. Akurasi tertinggi 77.77% diperoleh saat *hidden layer* berjumlah

1 dan nilai *learning rate* 0.05. Jika dilihat dari grafik, penambahan *hidden layer* ternyata membuat akurasi menjadi semakin menurun. Hal tersebut dipengaruhi oleh arsitektur jaringan yang memiliki lebih dari 1 *hidden layer* membuat sistem menjadi lebih kompleks. Sehingga 1 *hidden layer* sudah cukup untuk membangun sistem dengan akurasi yang tinggi.

3.2.2 Analisis Pengaruh Jumlah Neuron

Pelatihan jaringan *back propagation* yang memakan waktu lama bisa dikurangi dengan menambah jumlah *neuron* pada *hidden layer*. Namun sampai saat ini, belum ada formula khusus yang bisa menemukan jumlah *neuron* pada *hidden layer* yang optimal. Pada pengujian ini akan diteliti pengaruh dari jumlah *neuron* pada JST. Dari hasil analisis akan dicari jumlah *neuron* yang paling cocok untuk diterapkan pada sistem.

Pengujian ini dilakukan dengan mengubah jumlah *neuron* pada JST yang dibangun. Jenis ekstraksi ciri yang digunakan adalah ciri *window 1000ms*. Parameter JST lain yang digunakan, yaitu jumlah *hidden layer* 1, fungsi aktivasi *hidden layer* adalah *tansig*, dan fungsi aktivasi *output layer* adalah *purelin*. Algoritma *training* JST yang digunakan yaitu *trainrp*. Parameter lain pada *back propagation* yang digunakan yaitu nilai maksimum *epoch* sebanyak 1000, batas toleransi *error* 10^{-5} , nilai *learning rate* 0.4, gradien minimum 1×10^{-6} , dan nilai validasi 0.1.

Pada tabel 3.2.2 terlihat bahwa jumlah *neuron* dapat mempengaruhi kinerja sistem sehingga hasil akurasi yang didapatkan cukup beragam. Akurasi tertinggi 77.77% diperoleh saat jumlah *neuron* 20 dan 30. Hal tersebut dipengaruhi oleh besarnya jumlah *neuron* menyebabkan jaringan JST menjadi lebih rumit. Oleh karena itu, pada sistem akan diimplementasikan jumlah *neuron* sebanyak 20 pada *hidden layer* agar tercapai akurasi yang maksimal.

Tabel 3.2.2 Akurasi Akibat Pengaruh Jumlah Neuron

Jumlah Neuron	Akurasi (%)
10	70
20	77.77
30	76.67
40	70
50	66.67

3.2.3 Analisis Pengaruh Fungsi Aktivasi Pada Hidden layer dan OutputLayer

Back Propagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada lapisan tersembunyi. Algoritma *back-propagation* menggunakan *erroroutput* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan maju (*forwardpropagation*) harus dikerjakan terlebih dahulu. Pada saat perambatan maju, *neuron-neuron* diaktifkan dengan menggunakan fungsi aktivasi yang dapat dideferensiasikan seperti *tansig*, *logsig*, dan *purelin*. Pada pengujian ini akan diuji pengaruh dari fungsi aktivasi pada *hidden layer* dan *outputlayer* JST. Dari hasil analisis akan dicari fungsi aktivasi yang paling cocok untuk diterapkan pada sistem.

Pengujian ini dilakukan dengan mengubah fungsi aktivasi pada *hidden layer* dan *outputlayer* pada JST yang dibangun. Jenis ekstraksi ciri yang digunakan adalah ciri *window1000ms*. Parameter JST lain yang digunakan, yaitu jumlah *hidden layer* 1 dan jumlah *neuron* masing-masing *layer* adalah 20. Algoritma *training* JST yang digunakan yaitu *trainrp*. Parameter lain pada *back-propagation* yang digunakan yaitu nilai maksimum *epoch* sebanyak 1000, batas toleransi *error* 10^{-5} , nilai *learning rate* 0.4, gradien minimum 1×10^{-6} .

Tabel 3.2.3 Akurasi Akibat Pengaruh Fungsi Aktivasi

Aktivasi Hidden layer	Akurasi (%)		
	Aktivasi OutputLayer		
	Tansig	Logsig	Purelin
Tansig	30	70	77.77
Logsig	64.44	66.67	70
Purelin	40	64.44	66.67

Pada tabel 3.2.3 dapat dilihat bahwa fungsi aktivasi sangat mempengaruhi kinerja JST. Masing-masing fungsi aktivasi menghasilkan akurasi yang berbeda. Menurut hasil pengujian diperoleh bahwa fungsi aktivasi *hidden layer* tan-sigmoid (*tansig*) dan fungsi aktivasi *outputlayer* pure-linier (*purelin*) menghasilkan akurasi tertinggi yaitu 77.77%. Dapat disimpulkan bahwa fungsi aktivasi yang cocok untuk *hidden layer* adalah fungsi Tan-Sigmoid (*tansig*), dan fungsi aktivasi yang cocok untuk *outputlayer* adalah Pure-Linier (*Purelin*) karena untuk kombinasi tersebut mampu mencapai konvergensi tercepat dan menghasilkan akurasi tertinggi. Dari data yang diperoleh, juga dapat disimpulkan bahwa fungsi aktivasi *tansig* tidak terlalu cocok untuk dipakai pada *outputlayer* dan fungsi aktivasi *purelin* tidak terlalu cocok untuk dipakai pada *hidden layer*, karena sebagian besar fungsi aktivasi ini menghasilkan akurasi rendah.

3.2.4 Analisis Pengaruh Algoritma Pelatihan JST

Sebagian besar algoritma pelatihan untuk jaringan *feedforward* menggunakan gradien dari fungsi kinerja untuk menentukan bagaimana mengatur bobot-bobot dalam rangka meminimumkan kinerja. Gradien ini ditentukan dengan menggunakan suatu teknik yang disebut dengan nama *back-propagation*. Pada dasarnya, algoritma pelatihan *back-propagation* akan menggerakkan bobot dengan arah gradien negatif. Prinsip dasar dari algoritma *back-propagation* sederhana adalah memperbaiki bobot-bobot jaringan dengan arah yang membuat fungsi kinerja menjadi turun dengan cepat.

Pada pengujian ini akan diuji dan dianalisis pengaruh berbagai macam algoritma pelatihan JST. Algoritma yang diuji antara lain *Conjugate gradient back-propagation with Fletcher-Reeves updates (traincgf)*, *Gradient descent with adaptive learning rate back-propagation (traingda)*, *Gradient descent with momentum back-propagation (traingdm)*, *Gradient descent with momentum and adaptive learning rate back-propagation (traingdx)*, *Gradient descent back-propagation (traingd)*, *Resilient back-propagation (trainrp)*, *Conjugate gradient back-propagation with Polak-Ribière updates (traincgp)*, *Conjugate gradient back-propagation with Powell-Beale restarts (traingcb)*.

Pengujian ini dilakukan dengan mengubah algoritma pelatihan JST yang dibangun. Jenis ekstraksi ciri yang digunakan adalah ciri *window 1000ms*. Parameter JST lain yang digunakan, yaitu jumlah *hidden layer* 1, jumlah *neuron* masing-masing *layer* adalah 20, fungsi aktivasi *hidden layer* yang digunakan tansig, dan fungsi aktivasi *outputlayer* yang digunakan adalah purelin. Parameter lain pada *back-propagation* yang digunakan yaitu nilai maksimum epoch sebanyak 1000, batas toleransi *error* 10^{-5} , nilai *learning rate* 0.4, gradien minimum 1×10^{-6} .

Pada tabel 3.2.4 dapat dilihat akurasi yang dihasilkan dari algoritma pelatihan JST yang diuji. Dari tabel 4.4 terlihat bahwa algoritma pelatihan JST dapat mempengaruhi kinerja sistem. Algoritma pelatihan yang menghasilkan akurasi tertinggi 77.77% yaitu algoritma *Resilient backpropagation (trainrp)*. Dengan demikian pada sistem ini digunakan algoritma pelatihan *trainrp*. Dari hasil pengujian juga didapatkan bahwa algoritma pelatihan *traingdm* dan *traingd* meskipun menghasilkan akurasi >60% tetapi tidak cocok untuk digunakan pada sistem karena algoritma pelatihan ini tidak mampu mendeteksi data latih secara keseluruhan dan mengklasifikasikan data tersebut.

Tabel 3.2.4 Akurasi Akibat Pengaruh Algoritma Pelatihan JST

Algoritma Pelatihan	Akurasi (%)
Trainidx	60
Traingd	64.44
Traingdm	64.44
Traingda	70
Trainrp	77.77
Traincgf	64.44
Traincgp	66.67
Traincgb	66.67

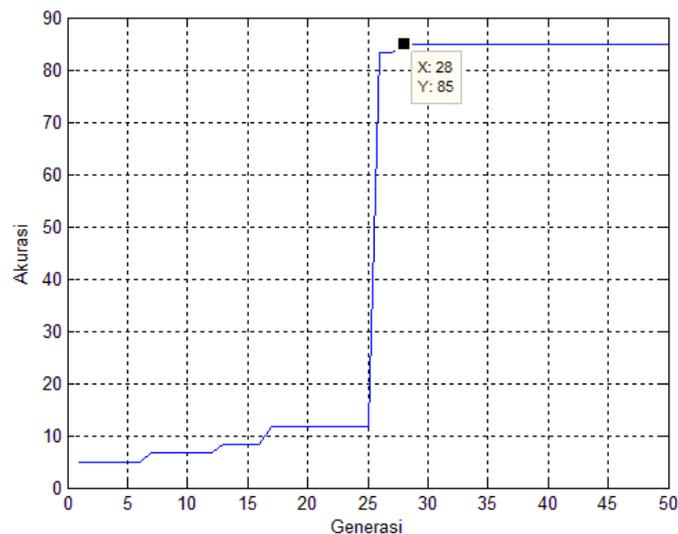
3.3 Optimasi parameter jaringan syaraf tiruan dengan algoritma genetika

Jaringan syaraf tiruan yang optimal didapatkan dengan mengkombinasikan beberapa parameter pelatihan. Sifat algoritma genetika yang umum, dapat digunakan sebagai sarana untuk mendapatkan kombinasi parameter yang optimal. Seperti yang dijelaskan pada sub bab 3.6.1 tentang perancangan kromosom untuk jaringan syaraf tiruan, optimasi dilakukan pada pemilihan jumlah *hidden layer*, jumlah *neuron* setiap *hidden layer* dan nilai *learning rate* yang paling optimal untuk proses pelatihan sehingga nilai akurasi uji mencapai nilai optimal.

Parameter algoritma genetika yang digunakan antara lain:

- Jumlah populasi = 50
- Maksimal Generasi = 100
- Probabilitas Pindah Silang (P_c) = 0.6
- Probabilitas Mutasi (P_m) = 0.5

Dari 3 kali percobaan, didapatkan akurasi data uji tertinggi pada nilai 85.55% dengan *hidden layer* 1 memiliki 43 *layer*, *hidden layer* 2 29 *layer* dan *learning rate* 0.778. Didapatkan nilai *fitness* tertinggi pada generasi ke 28.



Gambar 3.3 Grafik Optimasi JST-BP

4. Kesimpulan

Berdasarkan hasil implementasi, pengujian, dan analisis yang telah dilakukan, maka dapat ditarik kesimpulan sebagai berikut :

1. Perancangan klasifikasi genre musik berbasis pengolahan suara digital dengan metode jaringan syaraf tiruan *back propagation* sudah dirancang dan dioptimasi dengan algoritma genetika. Dari hasil pengujian dapat disimpulkan bahwa metode algoritma genetika dapat digunakan untuk meningkatkan akurasi aplikasi yang dirancang.
2. Dalam analisis pengaruh *window* ekstraksi ciri terhadap kinerja sistem didapat *window* ekstraksi ciri terbaik yaitu *window* 1000ms untuk ekstraksi ciri dalam sistem ini karena mempunyai ciri yang lebih kompleks. Pada proses preprocessing jenis filter yang digunakan adalah Butterworth dengan orde 5 tipe *bandpass* dan nilai *threshold cropping* adalah 0.05 dari hasil normalisasi amplitudo.
3. Parameter JST dapat mempengaruhi kinerja sistem karena dalam hal ini JST berperan sangat penting dalam proses deteksi. Penggunaan parameter JST yang berbeda akan menghasilkan akurasi yang berbeda. Pada tugas akhir ini, parameter yang menghasilkan akurasi maksimal yaitu jumlah *hidden layer* 1, jumlah *neuron* tiap *layer* 20, nilai *learning rate* 0.05, fungsi aktivasi *tansig* untuk *hidden layer*, fungsi aktivasi *purelin* untuk *output layer*, algoritma pembelajaran *trainrp* dengan akurasi 77.77% dari data latih sebanyak 200 data lagu dan 90 data humming sebagai data uji.
4. Parameter Algoritma Genetika meningkatkan akurasi menjadi 85.55% dengan parameter jumlah generasi 100, jumlah populasi 50, peluang crossover 0.6, dan peluang permutasi 0.01.

Daftar Pustaka

- [1] Betteng, Rico Chrisnawan. 2012. "Content Based Filtering Music Information Retrieval Berdasarkan Genre, Mood dan Nada Dasar dengan Inputan Audio". Bandung: Institut Teknologi Telkom
- [2] Fajri Muhammad, Yohan. 2013. "Perancangan Aplikasi Ketepatan Lagu dari Senandung Manusia berbasis Pengolahan Suara Digital dengan Metode Jaringan Saraf Tiruan Back Propagation". Bandung: Institut Teknologi Telkom
- [3] Hunt, Brian R., Ronald L. Lipsman, dan Jonathan M. Rosenberg. 2001. "A Guide to Matlab for Beginners and Experienced Users". New York: Cambridge University Press.
- [4] Hermawan, Arief, 2006, *Jaringan Syaraf Tiruan: Teori dan Aplikasi*, Andi Offset, Yogyakarta.
- [5] Mauludiya, Rosyita Ayuning. 2015. "SIMULASI DAN ANALISIS KLASIFIKASI GENRE MUSIK BERBASIS FFT DAN SUPPORT VECTOR MACHINE". Bandung: Universitas Telkom.
- [6] Petty, Brendan. 2010. "Music Genre Classification using a Backpropagation Neural Network", Labrosa.

- [7] Randy L. Haupt. 2004, "Practical Genetic Algorithms". A John Wiley & Sons, Inc.