

Perbandingan Klasifikasi Status Pengaturan pada Air Conditioner (AC)  
dengan Metode *Backpropagation* dan *Support Vector Machine*

*Comparison of Air Conditioner Setting Classification using  
Backpropagation and Support Vector Machine*

Laily Nur Qomariyati<sup>1\*</sup>, Syafira Tiara Zani<sup>1</sup>, Koredianto Usman<sup>1</sup>, Muhammad Ary Murti<sup>2</sup>

<sup>1</sup>Department of Telecommunication Engineering/ Faculty of Electrical Engineering

<sup>2</sup>Department of Electrical Engineering/ Faculty of Electrical Engineering

Telkom University, Jl. Telekomunikasi Jl. Terusan Buah Batu phone (022) 7565930

lailyng@student.telkomuniversity.ac.id<sup>1\*</sup>, syafiratz@student.telkomuniversity.ac.id<sup>1</sup>,

Korediantousman@telkomuniversity.ac.id<sup>1</sup>, Arymurti@telkomuniversity.ac.id<sup>2</sup>

**Abstrak** – Diantara perangkat elektronik yang digunakan oleh masyarakat modern, AC termasuk perangkat yang tersedia di berbagai negara, termasuk Indonesia. Sifat dari AC yang dioperasikan dalam jangka waktu yang lama menyebabkan AC menjadi salah satu perangkat elektronik yang mengkonsumsi daya terbesar di gedung-gedung perkantoran dan rumah tangga modern. Sifat dari masyarakat yang tidak memperhatikan pengaturan AC yang tepat semakin menambah pemborosan penggunaan AC. Oleh karena itu, dibutuhkan suatu sistem yang dapat mengontrol setting AC sehingga biaya penggunaannya menjadi minimal. Meskipun telah banyak penelitian yang berupaya untuk melakukan efisiensi penggunaan AC, namun penelitian tentang sumber-sumber IoT serta Machine Learning belum banyak dilakukan. Oleh karena itu, pada penelitian ini kami menyelidiki tentang penggunaan *Backpropagation*(BP) dan *Support Vector Machine*(SVM) terkait dengan klasifikasi setting AC dari kondisi setting oleh pengguna menuju setting ideal. Langkah ini diperlukan untuk tahap berikutnya terkait penjadwalan/scheduling dari setting AC yang optimum untuk menghasilkan konsumsi daya yang minimal. Sebagai input dari sistem diantaranya adalah jumlah orang dalam ruangan, temperature dalam dan luar ruangan serta temperature target. Output sistem berupa setting kondisi AC off, low, medium dan high. Data yang dibangkitkan bersifat dummy dengan jumlah dataset sebanyak 1476 data. Klasifikasi low, med, high didefinisikan dengan jumlah orang di dalam ruangan, selisih antara temperature luar dan temperature target. Sebagai contoh jumlah orang kurang dari 5 selisih temperature kurang dari 3°C, maka akan dikategorikan sebagai low effort. Hasil penelitian dengan data tersebut diperoleh akurasi untuk BP=96,1% dan SVM=96.9%. Dengan hasil akurasi tersebut, maka SVM lebih baik dibandingkan dengan BP untuk mengklasifikasikan data yang ada.

**Kata Kunci:** klasifikasi, pengaturan AC, machine learning, BP, SVM.

**Abstract** – Among the electronic devices used by modern society, air conditioning includes devices that are available in offices in big cities and various countries, including Indonesia. The nature of the AC which is turned on for a long time and the use of power has made it one of the electronic devices that consume the largest power in modern office buildings and households. To proper the AC setting that makes more waste when using the AC. Therefore, we need a system that can control AC settings so that the cost of using AC is minimal. Although there have been many studies that have attempted to carry out the process of using AC, there has not been much research on IoT sources and Machine Learning. Therefore, this study is based on the use of ANN BP and SVM related to the classification of AC settings from the regulatory conditions by the user to the ideal setting. This step is needed for the next stage related to scheduling of optimal AC settings to produce minimal power consumption. The input of ANN is the number of people in a room, indoor and outdoor temperatures, and the target temperature. The output of the system is setting the AC conditions off, low, medium, and high. The data generated is dummy with 1476 data. The low, medium, and high classifications are classified by the number of people in the room, the difference between the outside and the inside and target temperatures, and the current time. For

example, the number of people is less than 5, the difference in temperature is less than 3 degrees, it will be categorized as low effort. The results of the research with these data are accurately with ANN BP of 96.1% and SVM of 96.9%. In terms of accuracy the SVM method is better than BP.

**Keywords:** classification, AC's setting, machine learning, BP, SVM

## 1. Pendahuluan

Tidak bisa dipungkiri, saat ini *Air Conditioner* (AC) telah menjadi kebutuhan masyarakat modern. AC telah banyak ditemui di setiap sudut ruangan seperti gedung-gedung bahkan di sebagian besar rumah masyarakat. Energi yang dikonsumsi AC cukup besar jika dibandingkan dengan alat elektronik lainnya yaitu sekitar 350-2220 Watt. Menurut pernyataan Badan Energi Internasional (IEA), dalam 30 tahun ke depan penggunaan mesin pendingin ruangan atau AC akan menjadi penggerak utama permintaan listrik global. Mengacu pada hal tersebut, maka penggunaan AC harus dilakukan secara efektif dan efisien untuk menghemat konsumsi daya listrik.

Penggunaan AC sering tidak disesuaikan dengan kondisi ruangan seperti jumlah orang di dalam ruangan tersebut dan suhu ruangan serta suhu luar ruangan. Klasifikasi status pengaturan pada AC ini akan mengatur *effort* yang dilakukan untuk mencapai suhu target yang telah ditetapkan berdasarkan jumlah orang di dalam ruangan, suhu dalam ruangan, suhu luar ruangan, dan suhu ideal kenyamanan orang bekerja pada ruangan. Dengan adanya klasifikasi status *adjustment* pada AC, penggunaan AC dapat lebih terkendalikan sesuai kebutuhannya.

Beberapa pihak telah melakukan analisa terkait konsumsi energi berbasis *machine learning*. Metode *Decision Tree* digunakan pada penelitian sebelumnya oleh Zhun Yu dkk, dalam kajiannya tentang pemodelan penggunaan energi di perumahan Jepang[1], *decision tree* memiliki kelebihan yaitu komputasinya cenderung cepat karena daerah pengambilan keputusan yang sebelumnya kompleks dapat diubah menjadi lebih singkat namun metode ini lebih mudah untuk memprediksi variable kategori dibandingkan data variabel numerik[2]. *Artificial Neural Network* (ANN) juga pernah diterapkan sebagai algoritma prediksi sebab ANN memiliki kemampuan pemetaan nonlinear yang cukup canggih namun banyak variabel yang harus dipertimbangkan dalam pembangunan algoritmanya[3].

Penelitian ini akan menggunakan algoritma *Backpropagation* (BP) dan *Support Vector Machine* (SVM) untuk mengklasifikasikan *effort* pengaturan AC sebab belum banyak penelitian terdahulu menggunakan kedua metode ini. Pemilihan metode BP dilakukan sebagai representasi dari sistem JST klasik dan pemilihan SVM dilakukan sebagai representasi JST modern. *Backpropagation* adalah salah satu metode pembelajaran yang dapat diterapkan dalam Jaringan Syaraf Tiruan. Metode ini berfungsi untuk melatih jaringan *multilayer* JST. Penemuan *backpropagation* membuat JST kembali diminati setelah sempat terhenti karena kelemahan JST pada saat itu yang hanya memiliki *layer* tunggal. Model jaringan *backpropagation* merupakan suatu pelatihan *supervised learning* yang paling banyak digunakan. Algoritma ini sering dipakai untuk pengklasifikasian yang rumit karena algoritma dilatih menggunakan metode pembelajaran. *Support Vector Machine* (SVM) adalah salah satu metode yang dapat digunakan untuk melakukan prediksi baik dalam kasus klasifikasi maupun regresi. SVM mengadopsi struktur minimalisasi resiko (*Structure Risk Minimization*) yang mampu mengurangi generalisasi eror pada sejumlah data latih dan mampu menyelesaikan masalah non-linier berdimensi tinggi. Pada dasarnya SVM memiliki prinsip untuk melakukan klasifikasi antara dua kelas yang linier namun lebih jauh ia dikembangkan untuk klasifikasi multi kelas. SVM juga termasuk dalam kelas *supervised learning*.

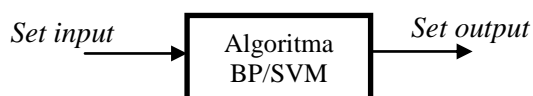
Keluaran yang diharapkan dari penelitian ini yaitu metode yang memiliki performa baik berdasarkan tingkat akurasi untuk memprediksi *setting* pemakaian AC sehingga daya yang digunakan lebih efisien.

## 2. Metode Penelitian

### 2.30. Prosedur Penelitian

#### 2.1.1 Set Data Input dan Data Output

Set data *input* dan set data *output* masing-masing berkorespondensi dengan data *input* dan data *output* pada BP dan SVM seperti Gambar 1.



Gambar 1 SVM/BP

- **Set input**

*Set input* yang menjadi ciri BP dan SVM pada penelitian ini terdiri dari suatu vektor dengan jumlah elemen sebanyak empat. Empat ciri ini terdiri dari (a) jumlah orang dalam suatu ruangan ( $n$ ); (b) temperatur dalam ruangan ( $T_i$ ); (c) temperatur luar ruangan ( $T_o$ ); (d) temperatur target ( $T_t$ ).

Ciri pertama yaitu jumlah orang menyatakan banyaknya orang dalam ruangan pada saat pengamatan. Data jumlah orang diambil dengan menggunakan sensor kamera yang diletakkan di dalam ruangan tersebut. Pada kenyataannya, diperlukan sistem terpisah untuk mengakuisisi citra di setiap interval waktu tertentu. Kemudian dengan menggunakan jaringan IoT, data citra dikirim ke *server* untuk dianalisa menggunakan pengolahan citra digital sehingga diperoleh jumlah orang dalam ruangan. Pada penelitian ini data jumlah orang dibangkitkan secara *dummy* dari 0 sampai 10 orang yang berkorespondensi dengan jumlah orang di dalam ruangan perkantoran kecil sampai sedang.

Temperatur dalam ruangan ( $T_i$ ) dan temperatur luar ruangan ( $T_o$ ) diukur dengan menggunakan sensor suhu yang diletakkan di dalam dan luar ruangan tersebut. Sensor ini juga akan mengirimkan secara berkala hasil pembacaan suhu ke *server* IoT. Selisih yang besar antara temperatur dalam dan luar ruangan menyebabkan AC harus bekerja keras untuk mengatasi perbedaan temperatur tersebut. Kerja keras tersebut dipetakan dengan keluaran bernilai *medium* atau *high* pada BP/SVM. Satuan yang dipakai pada temperatur ini adalah  $^{\circ}\text{Celsius}$ .

Temperatur target pada penelitian ini adalah temperatur ideal yang menyebabkan produktivitas orang bekerja menjadi optimal. Jurnal Sistem Teknik Industri Volume 6, No. 3 Juli 2005, yang berjudul “Menciptakan kenyamanan thermal dalam bangunan”, menulis bahwa suhu nyaman thermal untuk orang Indonesia berada pada rentang suhu  $22,8^{\circ}\text{C}$  -  $25,8^{\circ}\text{C}$  dengan kelembaban 70% [4]. Oleh karena itu, dalam penelitian ini, kami menentukan rentang suhu target ideal berkisar pada suhu  $22-25^{\circ}\text{C}$  disesuaikan dengan suhu luar ruangan.

Contoh vektor input dengan 4 elemen ini adalah

$$\mathbf{x}^T = [3 \quad 21 \quad 27 \quad 24]$$

Contoh di atas menunjukkan jumlah orang sebanyak 3, temperatur dalam  $21^{\circ}\text{C}$  temperatur luar  $27^{\circ}\text{C}$  dan temperatur target  $24^{\circ}\text{C}$ .

Pada penelitian ini digunakan pembangkitan data *dummy* untuk data latih dan data uji dengan jumlah data sebanyak 756 data latih dan 720 data uji. Data ini diletakkan pada *repository* di Google Drive dengan link sebagai berikut : <https://tinyurl.com/DataSetPaper>

- **Set output**

*Output* dari BP dan SVM adalah 4 macam target yaitu 1, 2, 3, dan 4. Ke empat *output* ini berkorespondensi dengan kondisi AC saat ini hingga mencapai temperatur ideal. *Output* 1 menunjukkan bahwa AC harus berada pada kondisi mati. *Output* ini terjadi antara lain apabila jumlah orang di dalam ruangan adalah 0 atau tidak ada orang di dalam ruangan. *Output* 2 berkorespondensi dengan upaya ringan dari AC untuk mencapai temperatur ideal. Kondisi ini berkorespondensi dengan jumlah orang yang sedikit dalam ruangan atau selisih antara temperatur dalam ruangan dan temperatur target kurang dari sama dengan  $2^{\circ}\text{C}$ . Dalam kondisi

keluaran 2 maka putaran *fan* pada AC dapat diatur dalam nilai minimum. *Output* 3 berkorespondensi dengan upaya sedang dari AC untuk mencapai temperatur ideal. Kondisi ini berkorespondensi dengan jumlah orang sedang antara 4 sampai 5 dalam ruangan atau selisih antara temperatur dalam ruangan dan temperature target antara 3 sampai 6°C. Dalam kondisi keluaran 3 maka putaran *fan* pada AC dapat diatur dalam nilai sedang. *Output* 4 berkorespondensi dengan upaya besar dari AC untuk mencapai temperatur ideal. Kondisi ini berkorespondensi dengan jumlah orang lebih dari sama dengan 6 dalam ruangan atau selisih antara temperatur dalam ruangan dan temperature target lebih dari 5°C. Dalam kondisi keluaran 4 maka putaran *fan* pada AC dapat diatur dalam nilai maksimum.

Pemetaan antara *set input* dan *set output* dapat dilihat pada Tabel 1

Tabel 1. Output BP dan SVM

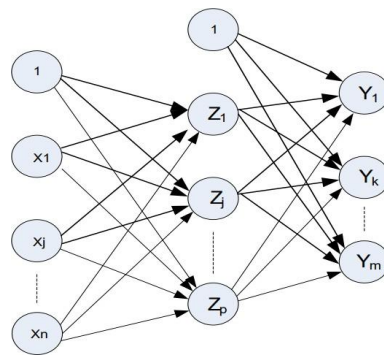
Output BP/SVM	Setting Effort AC	Keadaan	Keterangan
1	<i>Off</i>	Jumlah orang = 0	AC dalam keadaan mati
2	<i>Low</i>	$\Delta T \leq 2^\circ\text{C}$ selisih suhu target dan luar ruangan $N < 3$ , jumlah orang	Dalam waktu 30 menit, level <i>fan</i> AC akan naik 1 <i>step</i> dan suhunya turun 1°C
3	<i>Medium</i>	$3^\circ\text{C} < \Delta T < 6^\circ\text{C}$ , selisih suhu dalam dan luar ruangan $4 < N < 6$ , jumlah orang	Dalam waktu 25 menit, level <i>fan</i> AC akan naik 1 <i>step</i> dan suhunya turun 1°C
4	<i>High</i>	$\Delta T > 5^\circ\text{C}$ , selisih suhu dalam dan luar ruangan $N \geq 6$ , jumlah orang	Dalam waktu 20 menit, level <i>fan</i> AC akan naik 1 <i>step</i> dan suhunya turun 1°C

## 2.2 Algoritma Backpropagation

*Backpropagation* adalah salah satu metode pembelajaran yang dapat diterapkan dalam Jaringan Syaraf Tiruan. Metode ini berfungsi untuk melatih jaringan *multilayer* JST. Penemuan *backpropagation* membuat JST kembali diminati setelah sempat terhenti karena kelemahan JST pada saat itu yang hanya memiliki *layer* tunggal. Model jaringan *backpropagation* merupakan suatu pelatihan *supervised learning* yang paling banyak digunakan. Algoritma ini sering dipakai untuk pengklasifikasian yang rumit karena algoritma dilatih menggunakan metode pembelajaran [5]. Arsitektur algoritma *backpropagation* terdiri atas tiga *layer*, yaitu *input layer*, *hidden layer* dan *output layer*. Setiap *neuron* yang terdapat pada lapisan *input* akan terhubung dengan setiap *neuron* yang berada pada *hidden layer*, demikian juga setiap *neuron hidden layer* akan selalu terhubung pada lapisan *output*. Pada *input layer* tidak terjadi proses komputasi, sedangkan pada *input layer* terjadi pengiriman sinyal input  $X$  ke *hidden layer*. Pada *hidden layer* dan *output layer* terjadi proses komputasi terhadap bobot dan bias serta akan dihitung pula besarnya *output* dari *hidden layer* dan *output layer* tersebut berdasarkan fungsi aktivasi tertentu. Jaringan *backpropagation* terdiri dari *multilayer network* seperti Gambar 2 yaitu [6] :

1. Lapisan *input* (1 buah), yang terdiri dari 1 hingga  $n$  unit *input*.
2. Lapisan tersembunyi (minimal 1 buah), yang terdiri dari 1 hingga  $p$  unit tersembunyi.
3. Lapisan *output* (1 buah), yang terdiri dari 1 hingga  $m$  unit *output*.

Arsitektur JST BP dapat dilihat pada Gambar 2.

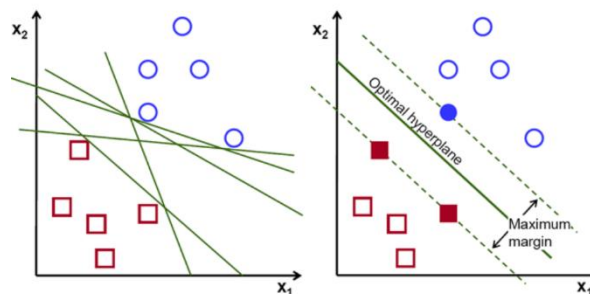


Gambar 2. Arsitektur *Backpropagation* (Jong, 2004)

**2.3 Algoritma Support Vector Machine**

*Support Vector Machine* merupakan suatu metode *machine learning* yang dapat digunakan untuk melakukan prediksi baik dalam kasus klasifikasi maupun regresi. SVM pertama kali diperkenalkan oleh Boser, Guyon, Vapnik pada tahun 1992. SVM memiliki kedudukan setara dengan *Neural Network* (NN) dimana keduanya termasuk pada kelas *supervised learning*. *Supervised learning* adalah sebuah teknik pendekatan dimana dibutuhkan data latih untuk mengelompokkan data tersebut ke dalam kategori yang telah ditentukan.

Pada dasarnya SVM memiliki prinsip untuk melakukan klasifikasi antara dua kelas yang linier namun lebih jauh ia dikembangkan untuk klasifikasi multi kelas. SVM bekerja dengan cara menemukan *hyperplane* terbaik dengan meminimalisir kesalahan dalam analisis serta memaksimalkan margin geometrinya[7]. Dalam SVM dikenal pula istilah *support vector* yakni data yang memiliki jarak terdekat dengan *hyperplane*, *support vector* ini yang akan diperhitungkan untuk menemukan *hyperplane* mengingat posisinya yang cukup sulit untuk diklasifikasikan sebab rawan terjadi *overlap*. Ilustrasi dari SVM dapat dilihat pada Gambar 3.

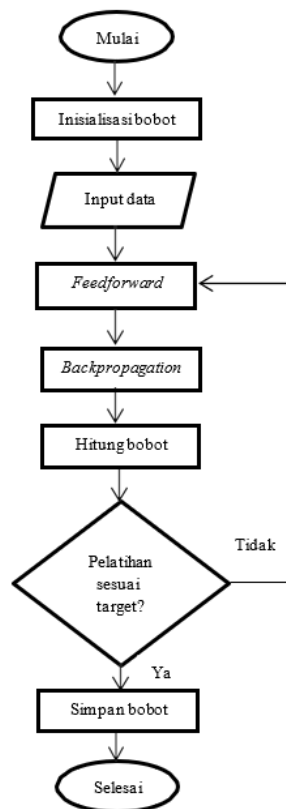


Gambar 3. Ilustrasi SVM

**2.4 Training dan Testing**

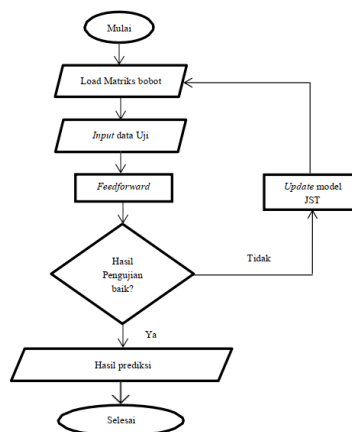
**2.4.1 Training dan Testing Algoritma BP**

Ada tiga tahapan dalam pelatihan JST *backpropagation*, yang pertama yaitu *feedforward* dimana *input* akan dihitung maju sampai *output layer* dengan fungsi aktivasi. Setelah itu dilakukan tahap *backpropagation* untuk mendapatkan nilai *error*. Tahap terakhir yaitu melakukan pelatihan hingga target *error* telah tercapai. Semua proses tersebut akan dilakukan berulang-ulang sampai target terpenuhi dan akan berakhir dengan penyimpanan bobot dari tahap pelatihan. Diagram alir dari proses pelatihan JST *backpropagation* adalah seperti pada Gambar 4 berikut :



Gambar 4. Diagram Alir Pelatihan JST BP

Selanjutnya akan dilakukan pengujian JST *backpropagation* terhadap data uji. Kondisi data uji telah disamakan dengan data latih terlebih dahulu. Selanjutnya akan diambil bobot terbaik yang telah didapatkan dalam proses pelatihan sebelumnya. Bobot tersebut kemudian akan diproses melalui tahap *feedforward* sehingga akan didapat hasil pengaturan *effort* AC. Diagram alir dari proses pengujian JST *backpropagation* adalah seperti pada Gambar 5 berikut:



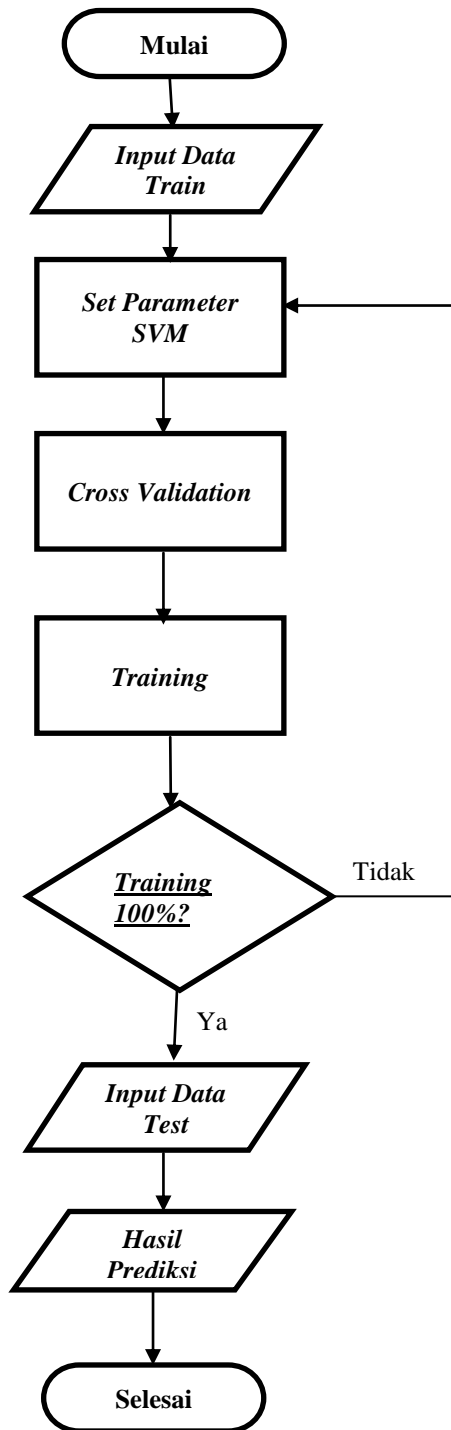
Gambar 5. Diagram Alir Pengujian JST BP

### 2.4.2 Training dan Testing Algoritma SVM

Untuk membuat pemodelan sistem, parameter input disiapkan sebagai data latih. Selanjutnya, parameter tersebut di kuantisasi dan normalisasi untuk meningkatkan penghitungan efisiensi dan mencegah data individu mengalami *overflow* selama penghitungan. Proses

kuantisasinya ialah menyelaraskan antara nilai selisih suhu target dan luar ruangan serta jumlah orang yang ada. Hal ini dilakukan agar tidak timbul kerancuan dalam pemilihan kelas *output*.

Selanjutnya, dilakukan pemilihan fungsi kernel yang akan digunakan. Ada beberapa jenis fungsi *kernel* diantaranya fungsi *linear*, fungsi polinomial, fungsi RBF *Gaussian*. Disamping fungsi *kernel*, juga ditentukan parameter *C* sebesar 0,5 ; 1 dan 10 dan *Gamma* yang akan digunakan dalam mencari akurasi training terbaik pada algoritma SVM. Setelah itu dilakukan validasi dan melatih sistem menggunakan data input yang sudah tersedia. Jika hasil *training* sudah mencapai 100%, maka akan diberikan parameter uji dan menghitung akurasi dari sistem yang sudah terbentuk. Output yang diharapkan ialah perintah *setting effort AC* berdasar kelas yang diberikan dan disesuaikan dengan parameter *input*. Proses yang dikerjakan pada sub blok diagram yang diteliti, ditunjukkan pada Gambar 6.



Gambar 6. Diagram Alir Algoritma SVM

**3. Hasil dan Pembahasan**

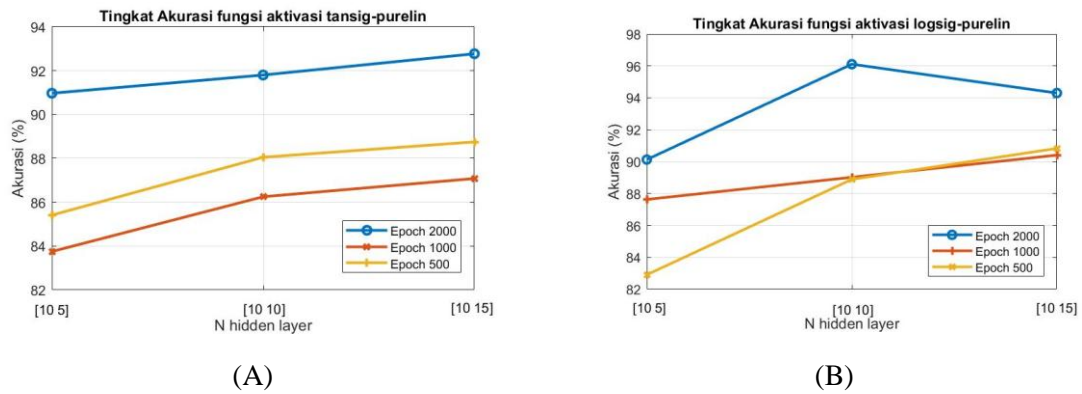
**3.1 Hasil Simulasi Akurasi pada JST BP**

**3.1.1 Akurasi JST BP terhadap *Epoch* dan *Hidden Layer***

Gambar 7(A) menunjukkan hasil simulasi saat dilakukan kombinasi parameter jumlah *hidden layer*, jumlah *epoch*, dan menggunakan fungsi aktivasi *tansig-purelin*. Gambar 7(B)



menunjukkan hasil simulasi saat dilakukan kombinasi parameter jumlah *hidden layer*, jumlah *epoch*, dan menggunakan fungsi aktivasi *logsig-purelin*.

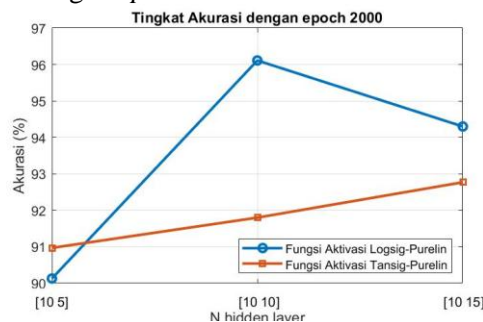


Gambar 7. (A) Tingkat Akurasi JST BP dengan Epoch 500 1000 2000 dengan fungsi aktivasi tansig-purelin (B) Tingkat Akurasi JST BP dengan Epoch 500 1000 2000 dengan fungsi aktivasi logsig-purelin

Dari kurva pada Gambar 7(A) tersebut dapat dilihat secara konsisten bahwa *epoch* 2000 memiliki tingkat akurasi lebih baik dari *epoch* 1000 dan *epoch* 500. Begitupun pada Gambar 7(B), *epoch* 2000 memiliki tingkat akurasi yang lebih baik daripada *epoch* 1000 dan *epoch* 500. Tingkat akurasi terbaik kedua diperoleh pada saat *epoch* 1000. Disisi lain, *epoch* 500 memiliki kinerja paling buruk untuk data yang diujikan. Namun, pada saat menggunakan fungsi aktivasi *logsig-purelin* dan pada saat jumlah *hidden layer* [10 15], *epoch* 500 memiliki tingkat akurasi yang lebih baik daripada *epoch* 1000. Akurasi terbaik dapat dilihat pada Gambar 7(B) saat jumlah *hidden layer* [10 10] yaitu sebesar 96,11%. Akurasi terbaik kedua dapat dilihat pada Gambar 7(A) saat jumlah *hidden layer* [10 15] yaitu sebesar 92,77%. Sedangkan nilai akurasi terendah didapatkan sebesar 82,91% dengan jumlah *hidden layer* [10 5] dan jumlah *epoch* 500. Selisih( $\Delta$ ) antara kinerja terbaik *hidden layer* [10 10] *epoch* 2000 dan kinerja terburuk *hidden layer* [10 5] *epoch* 500 yaitu 13,2%.

### 3.1.2 Akurasi JST BP terhadap Fungsi Aktivasi

Gambar 8 berikut menunjukkan kurva perbandingan hasil simulasi antara fungsi aktivasi *logsig-purelin* dan *tansig-purelin* dengan *epoch* 2000.



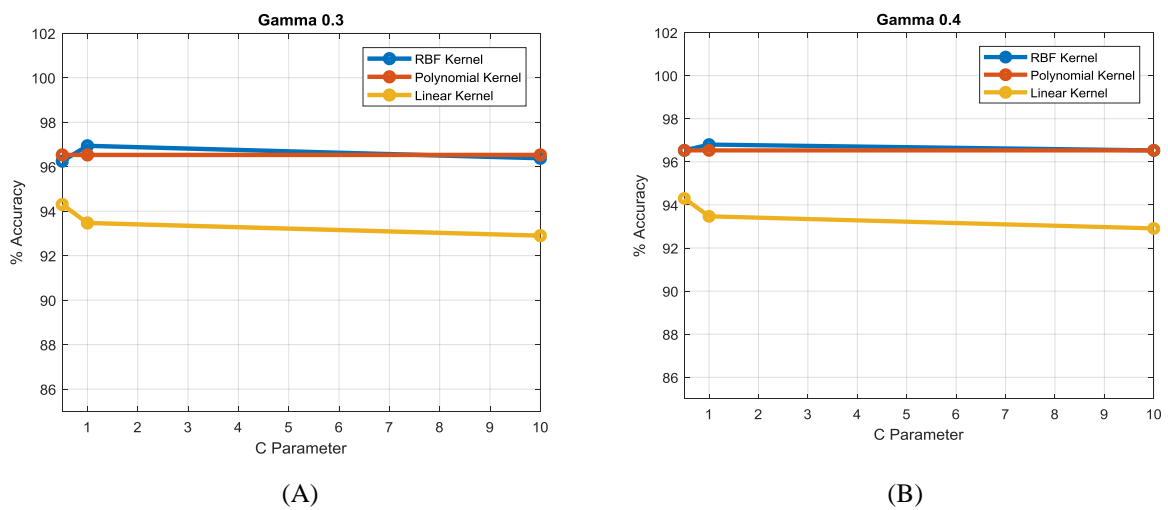
Gambar 8. Tingkat Akurasi JST BP dengan fungsi aktivasi *logsig-purelin* *tansig-purelin* dan *epoch* 2000

Dari kurva pada Gambar 8 dapat dilihat bahwa fungsi aktivasi *logsig-purelin* lebih baik daripada fungsi aktivasi *tansig-purelin*. Maka, performa terbaik didapat saat kombinasi

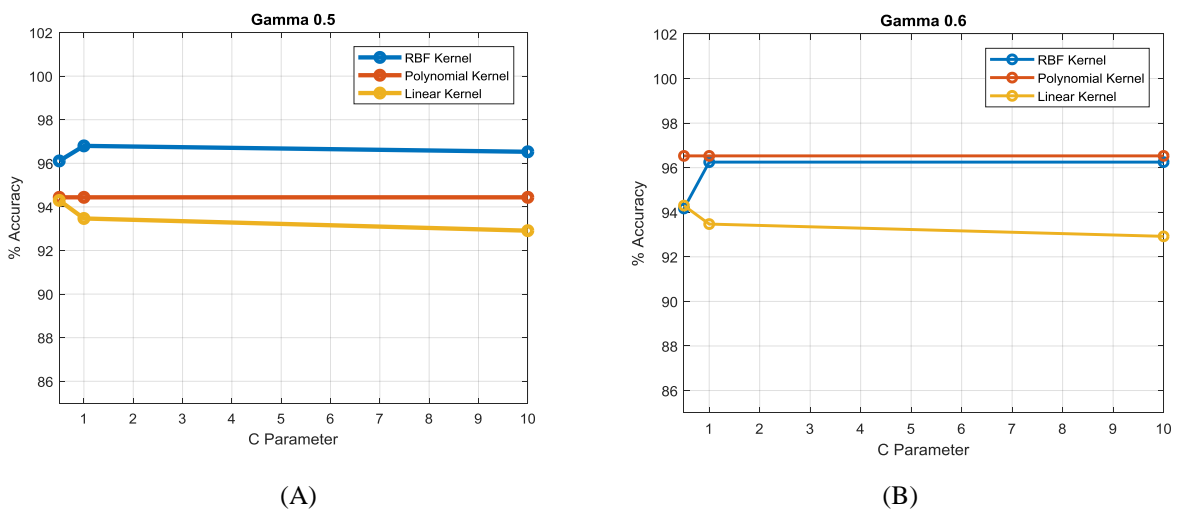
parameter dengan jumlah *hidden layer* [10 10], jumlah *epoch* 2000, dan fungsi aktivasi *logsig-purelin* yang memperoleh akurasi sebesar 96,11%.

### 3.2 Hasil Simulasi Akurasi pada SVM

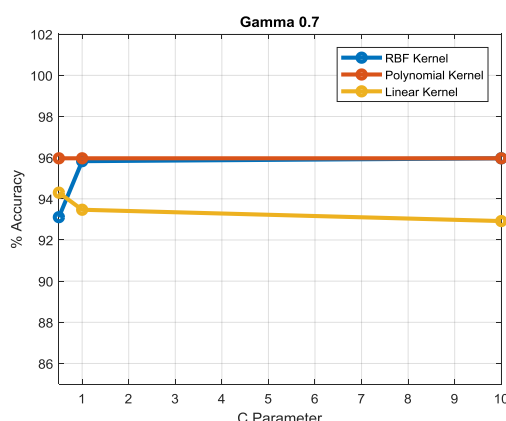
Simulasi pada SVM dilakukan untuk mengetahui pengaruh parameter yang bekerja pada SVM yaitu nilai *Gamma*, Fungsi *Kernel* dan Parameter *C*. Nilai *Gamma* yang akan diujikan antara lain 0,3 ; 0,4 ; 0,5 ; 0,6 ; 0,7. Untuk Fungsi *Kernel* digunakan *Kernel RBF Gaussian*, *Kernel Polynomial* dan *Kernel Linear*. Selanjutnya untuk nilai Parameter *C* yang digunakan ialah 0,5 ; 1 ; 10. Hasil simulasi ditunjukkan pada Gambar 9A, 9B, Gambar 10A, 10B dan Gambar 11 sebagai berikut.



Gambar 9. (A) Tingkat Akurasi SVM dengan Gamma= 0,3 (B) Tingkat Akurasi SVM dengan Gamma 0,4



Gambar 10. (A) Tingkat Akurasi SVM dengan Gamma= 0,5 (B) Tingkat Akurasi SVM dengan Gamma 0,6



Gambar 11. Tingkat Akurasi SVM dengan Gamma= 0,7

Gambar 9A merupakan hasil simulasi SVM dengan nilai  $\Gamma = 0,3$ . Sumbu datar menyatakan nilai  $C$  dari 0,5 hingga 10 dan sumbu tegak menyatakan nilai akurasi yang didapatkan pada saat pengujian sistem. Nilai akurasi tertinggi didapatkan dengan kombinasi  $C=1$  dengan *Kernel RBF Gaussian* yaitu sebesar 96,9%. Kinerja *Kernel Polynomial* mirip dengan *Kernel RBF* yang memiliki akurasi konstan pada nilai  $C = 0,5 ; 1 ; 10$  sebesar 96,5%. Pada sisi lain, *Kernel Linear* memiliki kinerja paling buruk untuk data yang diujikan. Pada nilai  $C=10$  didapatkan akurasi sebesar 92,9%. Lalu Gambar 9B merupakan hasil simulasi dengan nilai  $\Gamma=0,4$ . Kinerja *Kernel Polynomial* menghasilkan akurasi yang sama seperti pada Gambar 9A. *Kernel Linear* memberikan akurasi sebesar 92,91%. Hanya saja, *Kernel RBF Gaussian* pada saat  $C=1$  memiliki nilai yang lebih kecil dibandingkan pada Gambar 9A yakni sebesar 96,8%.

Gambar 10A menyatakan hasil simulasi SVM dengan nilai  $\Gamma$  sebesar 0,5. Nilai akurasi tertinggi didapatkan saat Parameter  $C$  bernilai 1 dengan *Kernel RBF Gaussian* yaitu sebesar 96,8%. *Kernel Polynomial* bekerja dengan konstan dan memiliki akurasi sebesar 94,4% pada nilai  $C = 0,5 ; 1 ; 10$ . Namun, *Kernel Linear* tetap memiliki kinerja terburuk ketika nilai Parameter  $C$  lebih dari 1 dengan akurasi sebesar 92,91%. Gambar 10B merupakan hasil simulasi dengan nilai  $\Gamma=0,6$ . *Kernel RBF* cenderung konstan dengan akurasi sebesar 96,3% pada nilai  $C=1$  dan 10. Namun, kinerja *Kernel RBF* lebih rendah dibandingkan dengan *Kernel Polynomial* yang memiliki akurasi konstan sebesar 96,5%. Untuk *Kernel Linear* menghasilkan akurasi sebesar 92,92%.

Gambar 11 menyatakan hasil simulasi dengan nilai  $\Gamma=0,7$ . Pada kurva tersebut terlihat bahwa akurasi tertinggi diperoleh dengan *Kernel RBF* pada saat nilai  $C$  adalah 10 dan pada *Kernel Polynomial* dengan akurasi sebesar 95,6%. Terlihat bahwa kinerja kedua *kernel* cukup baik dan konstan terutama *kernel polynomial*. *Kernel Linear* memberikan akurasi sebesar 92,92% pada saat parameter  $C$  bernilai 10.

Dari seluruh kurva simulasi SVM, *Kernel RBF Gaussian* cenderung memberika hasil akurasi yang cukup tinggi dibandingkan dengan *kernel* lainnya. Akurasi tertinggi dari seluruh kurva diperoleh sebesar 96,9% dengan  $\Gamma=0,3$ , *Kernel RBF* dan nilai  $C = 1$ . Sementara, untuk nilai akurasi terendah diperoleh sebesar 92,9% pada saat  $\Gamma=0,3$ , *Kernel Linear* dan nilai  $C = 10$ . Diketahui pula bahwa *Kernel Linear* memiliki kinerja paling buruk dibandingkan dengan *kernel* lainnya untuk data yang diujikan. Tren dari masing-masing kurva relatif konstan untuk nilai  $C$  lebih dari 1. Selisih akurasi antara kinerja seluruh kombinasi parameter yang diujikan pada SVM adalah sebesar 4,0%.

#### 4. Kesimpulan

Tingkat keberhasilan klasifikasi tergantung pada kombinasi parameter yang digunakan masing-masing metode. Dari hasil pengujian, metode *backpropagation* mempunyai performansi

terbaik dengan kombinasi parameter jumlah hidden layer [10 10], jumlah epoch 2000, dan fungsi aktivasi *logsig-purelin*. Sedangkan metode SVM mempunyai performansi terbaik dengan kombinasi parameter  $C = 1$ ,  $\text{Gamma} = 0.3$ , dan Fungsi Kernel RBF. Dari segi akurasi, dapat direkomendasikan metode SVM lebih baik dibandingkan metode BP dengan tingkat akurasi sebesar 96,9% dengan  $\Delta$  Akurasi = 4,0%. akurasi SVM lebih konsisten dibanding BP terhadap perubahan parameter.

## Referensi

- [1] Z. Yu, F. Haghghat, B. C. M. Fung, and H. Yoshino, "A decision tree method for building energy demand modeling," *Energy Build.*, vol. 42, no. 10, pp. 1637–1646, 2010, doi: 10.1016/j.enbuild.2010.04.006.
- [2] B. N. Patel, "Efficient Classification of Data Using Decision Tree," *Bonfring Int. J. Data Min.*, vol. 2, no. 1, pp. 06–12, 2012, doi: 10.9756/bijdm.1098.
- [3] Y. Sun, W. Wang, Y. Zhao, and S. Pan, "Predicting cooling loads for the next 24 hours based on general regression neural network: Methods and results," *Adv. Mech. Eng.*, vol. 2013, 2013, doi: 10.1155/2013/954185.
- [4] B. Talarosha, "Menciptakan Kenyamanan Thermal Dalam Bangunan," *J. Sist. Tek. Ind.*, vol. 6, no. 3, pp. 148–158, 2005.
- [5] R. S. Suhartanto, C. Dewi, and L. Muflikhah, "Implementasi Jaringan Syaraf Tiruan Backpropagation untuk Mendiagnosis Penyakit Kulit pada Anak," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 1, no. 7, pp. 555–562, 2017.
- [6] M. Agustin and T. Prahasto, "Penggunaan Jaringan Syaraf Tiruan Backpropagation Untuk Seleksi Penerimaan Mahasiswa Baru Pada Jurusan Teknik Komputer Di Politeknik Negeri Sriwijaya," *J. Sist. Inf. Bisnis*, vol. 2, no. 2, pp. 89–97, 2012, doi: 10.21456/vol2iss2pp089-097.
- [7] A. S. Nugroho, A. B. Witarto, and D. Handoko, "Support Vector Machine Teori Aplikasi dalam Bioinformatika," *Support Vector Mach. Teor. Apl. dalam Bioinformatika*, pp. 842–847, 2003, doi: 10.1109/CCDC.2011.5968300.