

# KLASIFIKASI DATA EEG UNTUK MENDETEKSI KEADAAN TIDUR DAN BANGUN MENGGUNAKAN AUTOREGRESSIVE MODEL DAN SUPPORT VECTOR MACHINE

Yunan Helmi Mahendra<sup>1)</sup>, Handayani Tjandrasa<sup>2)</sup>, dan Chastine Fatichah<sup>3)</sup>

<sup>1,2,3)</sup>Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

e-mail: [yunan12@mhs.if.its.ac.id](mailto:yunan12@mhs.if.its.ac.id)<sup>1)</sup>, [handatj@its.ac.id](mailto:handatj@its.ac.id)<sup>2)</sup>, [chastine@cs.its.ac.id](mailto:chastine@cs.its.ac.id)<sup>3)</sup>

## ABSTRAK

Tidur merupakan kebutuhan dasar manusia. Salah satu gangguan tidur yang cukup berbahaya adalah narkolepsi, yaitu gangguan tidur kronis yang ditandai dengan rasa kantuk yang luar biasa di siang hari dan serangan tidur yang terjadi secara tiba-tiba. Salah satu metode dokter untuk mendiagnosis penyakit narkolepsi adalah dengan melihat aktivitas gelombang otak (melalui sinyal EEG) pasien. Penelitian ini bertujuan untuk mengembangkan perangkat lunak yang dapat mengklasifikasikan keadaan tidur dan bangun melalui sinyal EEG secara otomatis. Dataset EEG yang digunakan tersedia di Physionet. Pertama-tama data EEG yang menjadi masukan dilakukan normalisasi dan filtering. Proses filtering dilakukan untuk membagi data menjadi 3 subband yaitu theta, alpha, dan beta. Setelah itu pada masing-masing subband dilakukan tahap ekstraksi fitur menggunakan Autoregressive Model. Hasil estimasi koefisien AR model digunakan sebagai fitur. Metode yang digunakan untuk mengestimasi koefisien AR model yaitu metode Yule-Walker dan metode Burg. Dataset dibagi menjadi data latih dan data uji menggunakan 10-fold cross validation. Data training digunakan untuk membuat SVM Model. SVM Model digunakan untuk mengklasifikasikan data testing sehingga menghasilkan keluaran label 1 untuk tidur dan label 0 untuk bangun. Untuk menentukan kelas final dilakukan majority vote dari hasil klasifikasi masing-masing subband. Performa sistem diperoleh dengan menghitung akurasi, presisi, dan sensitivitas pada setiap skenario uji coba. Skenario uji coba yang dilakukan antara lain dengan memvariasikan order AR, fungsi kernel, dan parameter C pada SVM. Dari hasil uji coba yang dilakukan, metode Yule-Walker menghasilkan rata-rata akurasi 80.60%, presisi 78.19%, dan sensitivitas 77.56%. Metode Burg menghasilkan akurasi 94.01%, presisi 95.70%, dan sensitivitas 93.39%. Hasil tersebut menunjukkan metode Burg memiliki performa lebih baik dibandingkan dengan metode Yule-Walker.

**Kata Kunci:** Autoregressive Model, Elektroensefalografi, Support Vector Machine, Tidur.

## ABSTRACT

Sleep is basic human necessity. Sleep function is for energy recovery, restoration of the brain, and boost immunity. People who have sleep disorder will lead to decreased immune system. One of the dangerous sleep disorders is narcolepsy. Narcolepsy is a chronic sleep disorder characterized by excessive sleepiness during the day and sleep attacks that occur suddenly. Usually, doctors diagnose narcolepsy by looking at brain wave activity (through EEG signal) of patients during sleep. The purpose of this final project is to develop software for classification of EEG data to detect sleep and awake condition through EEG signals using autoregressive (AR) model as a feature extraction method and support vector machine (SVM) as a classification method. EEG sleep dataset was provided by Physionet. First, EEG data are normalized and filtered. The aim of filtering process is to split data into three subbands: theta, alpha, beta. For each subband, feature extraction is done using by autoregressive (AR) model. The results of estimated coefficients of AR model are used as features. The methods used to estimate coefficients of AR model are Yule-Walker method and Burg method. Dataset is divided into training data and testing data using 10-fold cross validation. Training data is used to create SVM model. SVM Model is used to classify testing data which will produce output label 1 for sleep and label 0 for awake. Final class will be determined by majority voting of the classification results of each subband. System performance is obtained by calculating accuracy, precision, and sensitivity in each experiments. Experiments were performed by varying order AR model, kernel function of SVM, and parameter C of SVM. The average of accuracy, precision, and sensitivity that obtained when using Yule-Walker method respectively are 80.60%, 78.19%, and 77.56%. The average of accuracy, precision, and sensitivity that obtained when using Burg method respectively are 94.01%, 95.70%, and 93.39%. These result showed that Burg method outperformed Yule-Walker method.

**Keywords:** Autoregressive Model, Electroencephalography, Sleep, Support Vector Machine.

## I. PENDAHULUAN

**T**IDUR merupakan kebutuhan dasar manusia. Fungsi tidur antara lain untuk pemulihan energi, pemugaran otak, meningkatkan fungsi imunitas. Orang yang mengalami gangguan tidur, akan mengakibatkan konsentrasi menjadi menurun, daya tahan melemah, juga memicu terserangnya berbagai penyakit berbahaya seperti kanker, stroke, dan narcolepsy [1]. Selama tertidur, seseorang mengistirahatkan beberapa organ tubuhnya, salah satunya adalah otak. Sehingga terdapat perbedaan aktivitas gelombang otak pada orang yang terjaga (bangun)

dengan orang yang tidur. Perbedaan tersebut dapat diidentifikasi menggunakan *electroencephalography* (EEG) [2]. Sinyal EEG juga dapat digunakan untuk pengenalan sandi morse melalui kedipan mata [3], untuk mendeteksi penyakit epilepsi [4]. Selama ini metode yang digunakan untuk mendeteksi keadaan tidur melalui sinyal EEG sebagian besar masih dilakukan secara manual oleh seorang ahli [5]. Hal tersebut membutuhkan banyak biaya dan waktu. Oleh karena itu dibutuhkan suatu perangkat lunak untuk mendeteksi secara otomatis keadaan tidur dan bangun (terjaga) melalui sinyal EEG [6].

Terdapat beberapa penelitian yang dilakukan untuk mengklasifikasikan keadaan tidur dan bangun menggunakan sinyal EEG antara lain metode *Support Vector Machine (SVM)* yang digunakan sebagai metode klasifikasi berdasarkan energy, entropi, standar deviasi yang digunakan sebagai fiturnya [7]. Penelitian yang lainnya yaitu *Autoregressive Model* (metode *Burg*) yang digunakan sebagai metode ekstraksi fitur yang dikombinasikan dengan *Partial Least Square (PLS)*, *k-Nearest Neighbor (k-NN)*, *Bayes algorithm*, *Linear Discriminant Classifier (LDC)* sebagai metode klasifikasinya [7].

Pada penelitian ini metode yang diusulkan yaitu *Autoregressive Model* (metode *Yule-Walker* dan metode *Burg*) sebagai metode ekstraksi fitur dan *Support Vector Machine (SVM)* sebagai metode klasifikasi. Penelitian ini juga membandingkan performa (yang meliputi akurasi, presisi, sensitivitas) dari metode *Yule-Walker* dan *Burg* pada proses ekstraksi fitur.

## II. DASAR TEORI

### A. *Electroencephalography*

*Electroencephalography* (EEG) adalah suatu metode untuk merekam aktivitas elektrik di sepanjang kulit kepala dengan mengukur fluktuasi tegangan yang dihasilkan arus ion di dalam neuron otak. Sinyal EEG adalah rekaman sinyal elektronik otak selama periode waktu tertentu yang berasal dari perangkat yang disebut *Brain Computer Interface (BCI)*. Sinyal-sinyal listrik yang dihasilkan oleh otak akan ditangkap oleh *channel* (berupa elektroda) pada BCI [2].

Pengukuran sinyal EEG dilakukan dengan cara meletakkan elektroda-elektroda pada kulit kepala (*scalp*). Setiap elektroda dilambangkan huruf dan angka. Huruf menandakan area dari kepala pada elektrode, semisal F ialah *Frontal Lobe* dan T ialah *Temporal Lobe* [7]. Angka genap menandakan bagian kanan dari otak dan angka ganjil menandakan bagian kiri otak .

Berdasarkan frekuensi, amplitudo tegangan, dan kondisi objek, sinyal EEG dapat dibagi menjadi 4 gelombang, yaitu gelombang delta (kurang dari 4 Hz), theta (4 – 7 Hz), alpha (8 – 12 Hz), dan beta (13 – 49 Hz) [8].

### B. *Butterworth Bandpass Filtering*

*Bandpass filter* adalah filter yang hanya melewatkan sinyal-sinyal yang frekuensinya tercantum dalam pita frekuensi atau *pass band* tertentu. Frekuensi dari sinyal yang berada di bawah maupun di atas pita frekuensi tidak dapat dilewatkan oleh rangkaian *bandpass filter* [9]. *Butterworth filter* merupakan jenis *filter* pemrosesan sinyal yang didesain untuk memiliki respon frekuensi sedatar (rata) mungkin dalam *passband*. *Filter Butterworth* didefinisikan melalui persamaan *magnitude function*  $H(\omega)$  (1).

$$|H(\omega)|^2 = \frac{1}{1 + \omega^{2n}} \quad (1)$$

dimana  $\omega$  adalah frekuensi angular dalam radian per detik, dan  $n$  adalah *order filter* [10].

### C. *Autoregressive Model*

*Autoregressive (AR) Model* adalah model yang menggambarkan bahwa variabel dependen dipengaruhi oleh variabel dependen itu sendiri pada periode-periode dan waktu-waktu sebelumnya [11]. Secara umum, *autoregressive model* memiliki bentuk persamaan (2).

$$y(t) = - \sum_{i=1}^p \alpha_i y(t-i) + e(t) \quad (2)$$

dimana  $N$  adalah panjang data pada sinyal  $y$ ,  $p$  adalah *order autoregressive model*,  $\alpha$  adalah parameter atau koefisien *autoregressive model*,  $y(t)$  adalah nilai sekarang,  $y(t-i)$  adalah nilai sebelumnya,  $e(t)$  adalah *white gaussian noise with zero mean and variance*.

Metode untuk mengestimasi nilai parameter atau koefisien *autoregressive model* menggunakan metode *Yule-Walker* dan metode *Burg*.

*Metode Yule-Walker*

Persamaan (3) adalah persamaan *Yule-Walker* untuk mendapatkan nilai parameter  $\alpha$  (koefisien AR model),

$$R_{xx}(k) + \sum_{i=1}^p \alpha_i R_{xx}(k-i) = 0 \quad (3)$$

dengan *error* model didefinisikan dengan rumus (4).

$$e = R_{xx}(0) + \sum_{i=1}^p \alpha_i R_{xx}(k-i) \quad (4)$$

dimana  $R_{xx}$  adalah *autocorrelation coefficient* [12].

Untuk mempermudah perhitungan, (3) dan (4) dikombinasikan kemudian diubah ke dalam persamaan matriks menjadi:

$$\begin{bmatrix} R_{xx}(0) & R_{xx}^*(1) & \dots & R_{xx}^*(p-1) & R_{xx}^*(p) \\ R_{xx}(1) & R_{xx}(0) & & R_{xx}^*(p-2) & R_{xx}^*(p-1) \\ \vdots & \vdots & & \vdots & \vdots \\ R_{xx}(p-1) & R_{xx}(p-2) & & R_{xx}(0) & R_{xx}^*(1) \\ R_{xx}(p) & R_{xx}(p-1) & \dots & R_{xx}(1) & R_{xx}(0) \end{bmatrix} \begin{bmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} e \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Perhitungan persamaan matriks di atas dapat diselesaikan menggunakan algoritma *Levinson-Durbin recursion*. Penyelesaian *Levinson-Durbin recursion* dilakukan dengan menambahkan parameter  $\gamma$  dan konstanta  $\Gamma$  yang didefinisikan melalui (5) dan (6) [13].

$$\gamma = R_{xx}(j+1) + \sum_{i=2}^j \alpha_i R_{xx}(j-i+2) \quad (5)$$

$$\Gamma = \frac{-\gamma}{error} \quad (6)$$

Berikut adalah algoritma *Levinson-Durbin recursion*:

1. Inisialisasi
  - a)  $\alpha_1 = 1$
  - b)  $error = R_{xx}(0)$
2. For  $j = 1$  to order
  - a) Hitung parameter  $\gamma$
  - b) Hitung konstanta  $\Gamma$
  - c) for  $i = 2$  to  $j$ 

$$\alpha_i = \alpha_i + \Gamma \alpha_{(j-i+2)}^*$$
  - d)  $\alpha_{j+1} = \Gamma$
  - e)  $error = error(1 - |\Gamma|^2)$

*Metode Burg*

Pada tahun 1975, Burg mengembangkan suatu metode untuk mengestimasi parameter AR Model yang didasarkan pada prediksi linier *forward* dan *backward* [14]. Misalkan diketahui data signal  $y(t)$  dengan  $t = 1, 2, \dots, N$ . Maka kita bisa menghitung prediksi linier *forward* dan *backward* menggunakan (7) dan (8) [13].

$$y'(t) = - \sum_{i=1}^p \alpha_{p,i} y(t-i) \quad (7)$$

$$y'(t - p) = - \sum_{i=1}^p \alpha^*_{p,i} y(t - i) \quad (8)$$

dimana  $\alpha$  dan  $\alpha^*$  secara berurutan adalah koefisien prediksi linier *forward* dan *backward* dan  $p$  adalah order AR model.  $\alpha^*$  didefinisikan sebagai kompleks konjugasi dari  $\alpha$ . Sedangkan prediksi *error forward* dan *error backward* secara berurutan dapat dihitung dengan (9) dan (10).

$$ef_p(t) = y(t) - y'(t) = y(t) + \sum_{i=1}^p \alpha_{p,i} y(t - i) \quad (9)$$

$$eb_p(t) = y(t - p) + \sum_{i=1}^p \alpha^*_{p,i} y(t - p + i) \quad (10)$$

$ef_p(t)$  dan  $ef_b(t)$  berurutan adalah prediksi *error forward* dan *backward*.  $ef_p(t)$  dan  $ef_b(t)$  bergantung pada nilai  $p+1$ , sehingga nilai  $p$  harus lebih kecil dari  $N$ .  $ef_p(t)$  dan  $ef_b(t)$  dapat diminimalisasi dengan menambahkan suatu koefisien *reflection*  $k_p$  yang memenuhi (11).

$$k_p = \frac{-2 \sum_{t=p+1}^N ef_{p-1}(t) + eb^*_{p-1}(t-1)}{\sum_{t=p+1}^N (|ef_{p-1}(t)|^2 + |eb_{p-1}(t-1)|^2)} \quad (11)$$

Berikut adalah algoritma *Burg* untuk mengestimasi parameter AR model:

1. Inisialisasi  $ef_0(t) = eb_0(t) = y(t)$
2. For  $p=1$  to order AR
  - a) Hitung *reflection coefficient*  $k_p$
  - b) Perbaharui  $ef(t)$ ,  $eb(t)$  dengan rumus (12) dan (13)
 
$$ef_p(t) = ef_{p-1}(t) + k_p eb_{p-1}(t-1) \quad (12)$$

$$eb_p(t) = eb_{p-1}(t-1) + k^*_p ef_{p-1}(t) \quad (13)$$
  - c) Hitung  $\alpha_{p,i}$  untuk  $i = 1, \dots, p$  dengan rumus (14) dan (15)
 
$$\alpha_{p,i} = \alpha_{p-1,i} + k_p \alpha^*_{p-1,p-i}, \text{ untuk } i = 1, \dots, p-1 \quad (14)$$

$$\alpha_{p,i} = k_p, \text{ untuk } i = p \quad (15)$$
3.  $\alpha_{p,1}, \dots, \alpha_{p,p}$  adalah hasil estimasi koefisien AR model

#### D. Support Vector Machine

*Support Vector Machine* (SVM) dikembangkan oleh Oser, Guyon, Vapnik dan pertama kali dipresentasikan pada tahun 1992. Prinsip dasar SVM adalah *linear classifier*, dan selanjutnya dikembangkan agar dapat bekerja pada *problem non-linear* dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi [7].

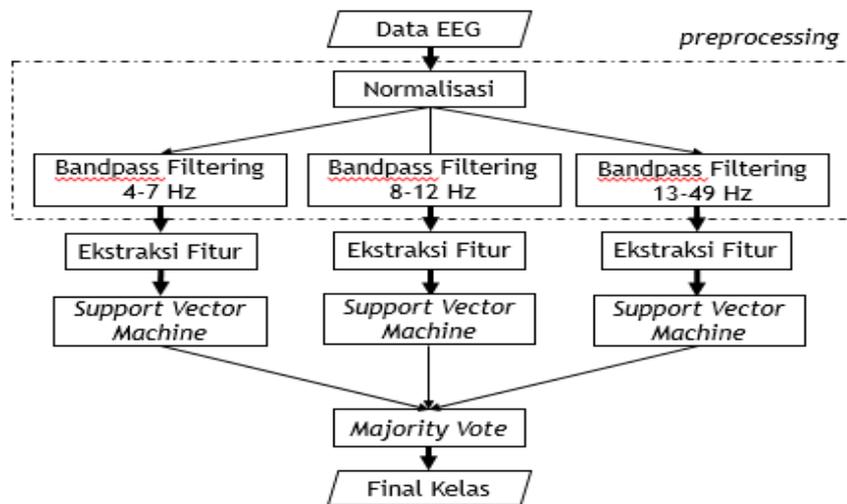
Konsep dasar SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua kelas pada *input space*. *Hyperplane* pemisah terbaik antara kedua kelas dapat ditemukan dengan mengukur *margin hyperplane* dan mencari titik maksimalnya. *Margin* adalah jarak antara *hyperplane* dengan data terdekat dari masing-masing kelas. Data yang paling dekat dengan *hyperplane* disebut sebagai *support vectors* [15].

### III. METODOLOGI PENELITIAN

Rancangan perangkat lunak dimulai dengan membaca masukan berupa data sinyal EEG berupa file txt. Proses deteksi keadaan tidur dan bangun terdiri dari empat proses utama, yaitu *preprocessing*, ekstraksi fitur, klasifikasi, dan *majority vote*. Diagram alir desain umum perangkat lunak ditunjukkan pada Gambar 1.

#### A. Dataset

Dataset yang digunakan adalah data sinyal otak (EEG) yang diunduh dari [www.physionet.org/cgi-bin/atm/ATM](http://www.physionet.org/cgi-bin/atm/ATM) (*Sleep-EDF Database, v1 [deprecated, use sleep-edfx]*). Dataset terdiri 8 subjek (yang memiliki durasi rekaman berbeda-beda), masing-masing subjek terdiri dari 2 tipe yaitu data hasil rekaman EEG *channel Fpz-Cz* (*Channel Fpz-Cz* dapat didefinisikan sebagai perbedaan tegangan antara elektrode *Fpz* dan elektrode *Cz*) dan data *hypnogram* (grafik yang menunjukkan tahapan tidur sebagai fungsi waktu). Data *hypnogram* berisi *ground truth class* yang diperlukan untuk proses latih pada klasifikasi, keterangan kelas tersebut antara lain: 0



Gambar 1. Diagram Alur Sistem yang Diusulkan

untuk kelas bangun, 1 untuk kelas tidur NREM1, 2 untuk kelas tidur NREM2, 3 untuk kelas tidur NREM3, 4 untuk kelas tidur NREM4, dan 5 untuk kelas tidur REM. Data hasil rekaman EEG dengan format EDF (European Define Format) terlebih dahulu diubah ke dalam format file text (.txt) menggunakan aplikasi *EDF Browser*.

### B. Preprocessing

Pada tahap *preprocessing*, diberikan data masukan sinyal EEG  $X_i$  dimana  $i=1, 2, \dots$ , panjang data. Data masukan dinormalisasikan dengan *range* [-1 1]. Hal tersebut dilakukan agar data memiliki skala yang sama. Normalisasi data dilakukan dengan rumus (16).

$$X_{new} = \frac{(X_{old} - \min(X_{old})) * 2}{\max(X_{old}) - \min(X_{old})} - 1 \quad (16)$$

Setelah dilakukan normalisasi, data di-*filter* menggunakan *Butterworth bandpass filtering* untuk memisahkan data menjadi 3 sub band yaitu *theta*, *alpha*, dan *beta*. *Order* 8 dan batas frekuensi 4 - 7 Hz untuk *theta*, *order* 10 dan batas frekuensi 8 - 12 Hz untuk *alpha*, *order* 11 dan batas frekuensi 13 - 49 Hz untuk *beta*. Hasil dari proses *filtering* akan menjadi data masukan pada tahap ekstraksi fitur [8].

### C. Ekstraksi Fitur

Setelah terbagi menjadi *theta*, *alpha*, dan *beta*, pada masing-masing sub band tersebut akan dilakukan tahap ekstraksi fitur menggunakan metode *autoregressive model*. Koefisien *autoregressive model* akan menjadi fitur yang merupakan data masukan pada tahap klasifikasi. Jumlah koefisien *autoregressive model* bergantung pada nilai *order*-nya. Sehingga jumlah fitur sama dengan nilai *order* AR model.

Metode yang digunakan untuk mengestimasi koefisien *autoregressive model* yaitu metode *Yule-Walker* dan metode *Burg*. Metode *Yule-Walker* dalam mengestimasi koefisien atau parameter *autoregressive* (AR) model didasarkan pada koefisien *autocorrelation* dan *Levinson-Durbin recursion*, sedangkan metode *Burg* dalam mengestimasi koefisien atau parameter *autoregressive* (AR) model didasarkan pada meminimalkan prediksi *error forward* dan prediksi *error backward*. Prediksi *error forward* dan prediksi *error backward* dapat diminimalkan dengan menambahkan koefisien *reflection*.

### D. Klasifikasi

Atribut *theta*, *alpha*, dan *beta* hasil tahap ekstraksi fitur menjadi masukan pada tahap klasifikasi. Sebelum dilakukan proses klasifikasi, *theta*, *alpha*, dan *beta* masing-masing dibagi menjadi data training dan data *testing*. Data *training* adalah data yang digunakan untuk membuat model klasifikasi yang digunakan. Sedangkan data *testing* digunakan untuk proses uji coba model klasifikasi yang telah dibuat. Metode klasifikasi yang digunakan adalah *support vector machine*. Hasil dari tahap klasifikasi adalah kelas bangun yang direpresentasikan dengan angka 0 dan kelas tidur dengan angka 1 untuk masing-masing *theta*, *alpha*, *beta*.

### E. Majority Vote

Ketiga hasil klasifikasi dari masing-masing *subband* akan dilakukan *majority vote* sehingga menghasilkan final kelas. *Majority vote* dilakukan dengan memilih kelas yang paling banyak dihasilkan dari ketiga *subband* yang ada

TABEL I  
PROSES PENENTUAN FINAL KELAS MENGGUNAKAN MAJORITY VOTE

Kelas pada <i>thetaband</i>	Kelas pada <i>alphaband</i>	Kelas pada <i>betaband</i>	Final Kelas
0	0	0	<b>0</b>
0	0	1	<b>0</b>
0	1	0	<b>0</b>
0	1	1	<b>1</b>
1	0	0	<b>0</b>
1	0	1	<b>1</b>
1	1	0	<b>1</b>
1	1	1	<b>1</b>

(*thetaband*, *alphaband*, dan *betaband*). Tabel I adalah contoh proses penentuan final kelas menggunakan *majority vote* berdasarkan hasil klasifikasi ketiga *subband*.

#### IV. UJI COBA

Uji coba dilakukan pada setiap subjek menggunakan skema *10 fold cross-validation*, dimana setiap dataset semua percobaan dibagi menjadi sepuluh bagian yang berjumlah relatif sama. Sembilan bagian digunakan untuk data latih dan satu bagian sisanya digunakan untuk data uji. Hasil uji coba pada bab ini merupakan hasil rata-rata dari sepuluh iterasi *10 fold cross-validation*. Terdapat tiga skenario dalam uji coba perangkat lunak ini. Pada setiap skenario uji coba dilakukan perbandingan performa (yang meliputi akurasi, presisi, sensitivitas) metode *Yule-Walker* dan *Burg*. Hasil performa setiap skenario uji coba terdapat pada bagian lampiran.

Skenario uji coba pertama adalah perhitungan performa dengan variasi nilai *order (p) autoregressive model*. Nilai *p* yang diuji coba antara lain: 10, 20, 30, 40, 50, 60, 70, 80, 90, dan 100. Nilai order AR merepresentasikan

TABEL II  
HASIL AKURASI UJI COBA VARIASI NILAI ORDER AR MODEL

Order AR	Metode <i>Yule-Walker</i> (%)	Metode <i>Burg</i> (%)
10	73.91	89.60
20	77.80	92.58
30	78.12	93.27
40	78.82	93.90
50	81.21	94.13
60	79.77	94.58
70	80.91	94.84
80	78.85	94.20
90	80.04	93.47
100	79.03	91.17
rata-rata	78.85	93.17

TABEL III  
HASIL PRESISI UJI COBA VARIASI NILAI ORDER AR MODEL

Order AR	Metode <i>Yule-Walker</i> (%)	Metode <i>Burg</i> (%)
10	75.32	93.17
20	77.79	97.08
30	78.19	97.06
40	77.22	96.83
50	77.19	96.75
60	78.00	95.58
70	78.33	95.38
80	77.30	94.60
90	77.32	94.40
100	77.45	95.21
rata-rata	77.41	95.61

TABEL IV  
HASIL SENSITIVITAS UJI COBA VARIASI NILAI ORDER AR MODEL

Order AR	Metode <i>Yule-Walker</i> (%)	Metode <i>Burg</i> (%)
10	78.20	87.21
20	78.51	89.96
30	77.38	91.39
40	78.75	92.26
50	80.47	92.80
60	77.51	94.46
70	77.91	95.40
80	75.51	95.28
90	76.99	93.44
100	75.18	85.67
rata-rata	77.64	91.79

TABEL V  
HASIL UJI COBA VARIASI FUNGSI KERNEL SVM  
DENGAN METODE YULE-WALKER

Fungsi Kernel	Akurasi (%)	Presisi (%)	Sensitivitas (%)
Linear	78.47	78.46	78.71
RBF (gamma=0.5)	82.96	78.76	75.96
RBF (gamma=1)	78.30	76.21	80.54
RBF (gamma=1.5)	79.84	76.78	81.10
RBF (gamma=2)	78.48	76.15	81.02
rata-rata	79.61	77.27	79.47

TABEL VI  
HASIL UJI COBA VARIASI FUNGSI KERNEL SVM  
DENGAN METODE BURG

Fungsi Kernel	Akurasi (%)	Presisi (%)	Sensitivitas (%)
Linear	93.51	97.38	92.05
RBF (gamma=0.5)	92.64	94.64	89.90
RBF (gamma=1)	94.93	95.46	95.41
RBF (gamma=1.5)	94.88	96.64	94.09
RBF (gamma=2)	94.35	96.98	93.10
rata-rata	94.06	96.22	92.91

banyaknya fitur yang digunakan pada tahap klasifikasi. Pada skenario uji coba ini, jenis *kernel function* yang digunakan pada SVM adalah RBF (*Radial Basis Function*) dengan parameter  $\gamma=1$ . Hasil performa dari masing-masing nilai  $p$  dapat dilihat pada Tabel II, III, dan IV.

Skenario uji coba kedua adalah perhitungan performa yang meliputi akurasi, presisi, sensitivitas dengan memvariasikan fungsi *kernel support vector machine*. Fungsi *kernel* yang diuji coba antara lain: *linear*, RBF (*Radial Basis Function*) dengan nilai  $\gamma$  antara lain: 0.5, 1, 1.5, dan 2. Nilai  $\gamma$  pada RBF mempengaruhi besar kecilnya *variance*. Semakin besar nilai  $\gamma$  mengakibatkan semakin kecil nilai *variance*, sehingga *support vectors* memiliki pengaruh yang besar dalam menentukan kelas dari data  $x_i$ . Sebaliknya, jika nilai  $\gamma$  semakin kecil mengakibatkan semakin besar nilai *variance*, sehingga pengaruh *support vectors* dalam menentukan kelas dari data  $x_i$  sangat kecil. Pada skenario uji coba ini, nilai parameter C pada *support vector machine* yang digunakan adalah 1. *Order autoregressive (AR) model* yang digunakan untuk metode *Yule-Walker* adalah 50 dan 70 untuk metode *Burg*. Penentuan nilai *order* tersebut didasarkan pada hasil uji coba pada skenario 1 Hasil performa dari masing-masing variasi fungsi *kernel* dapat dilihat pada Tabel V dan VI.

Skenario uji coba ketiga adalah perhitungan performa yang meliputi akurasi, presisi, sensitivitas dengan variasi nilai parameter C pada *support vector machine*. Nilai parameter C yang diujikan antara lain: 1, 2, 3, 4, dan 5. Nilai parameter C mempengaruhi besar kecilnya *margin hyperplane*. Semakin besar nilai C, maka *margin hyperplane* akan semakin kecil. Sebaliknya, nilai C yang besar mengakibatkan semakin kecilnya *margin hyperplane*. Pada skenario uji coba ini, *Order AR model* yang digunakan untuk metode *Yule-Walker* adalah 50, dan 70 untuk metode *Burg*. Fungsi *kernel SVM* yang digunakan adalah RBF dengan  $\gamma=0.5$  untuk metode *Yule-Walker*, sedangkan untuk metode *Burg* digunakan RBF dengan  $\gamma=1$ . Hasil performa dari skenario uji coba ini dapat dilihat pada Tabel VII dan VIII.

TABEL VII  
HASIL UJI COBA VARIASI NILAI PARAMETER C  
DENGAN METODE YULE WALKER

Nilai C	Akurasi (%)	Presisi (%)	Sensitivitas (%)
1	82.96	78.76	75.96
2	82.84	79.27	74.98
3	83.29	79.88	75.69
4	83.67	80.44	75.61
5	83.93	81.06	75.54
rata-rata	83.34	79.88	75.56

TABEL VIII  
HASIL UJI COBA VARIASI NILAI PARAMETER C  
DENGAN METODE BURG

Nilai C	Akurasi (%)	Presisi (%)	Sensitivitas (%)
1	94.93	95.46	95.41
2	94.81	95.29	95.37
3	94.83	95.26	95.57
4	94.70	95.11	95.49
5	94.77	95.20	95.44
rata-rata	94.81	95.26	95.46

## V. KESIMPULAN

Dari hasil uji coba yang dilakukan, dapat diambil kesimpulan yaitu metode *autoregressive* (AR) model dan *support vector machine* (SVM) dapat dijadikan metode untuk ekstraksi fitur dan klasifikasi data EEG. Metode *Burg* untuk mengestimasi koefisien AR model menghasilkan performa yang lebih baik daripada metode *Yule-Walker*. Metode *Yule-Walker* menghasilkan rata-rata akurasi 80.60%, presisi 78.19%, dan sensitivitas 77.56%. Sedangkan metode *Burg* menghasilkan rata-rata akurasi 94.01%, presisi 95.70%, dan sensitivitas 93.39%. Akurasi tertinggi metode *Yule-Walker* diperoleh ketika nilai *order* AR = 50, fungsi kernel menggunakan *Radian Basis Function* (RBF) dengan parameter  $\gamma = 0.5$  dan parameter  $C = 5$ . Akurasi tertinggi metode *Burg* diperoleh ketika nilai *order* AR = 70, fungsi kernel menggunakan *Radian Basis Function* (RBF) dengan parameter  $\gamma = 1$  dan parameter  $C = 1$ . Presisi tertinggi metode *Yule-Walker* diperoleh ketika nilai *order* AR = 50, fungsi kernel menggunakan *Radian Basis Function* (RBF) dengan parameter  $\gamma = 0.5$  dan parameter  $C = 5$ . Presisi tertinggi metode *Burg* diperoleh ketika nilai *order* AR = 70, fungsi kernel menggunakan *linear*. Sensitivitas tertinggi metode *Yule-Walker* diperoleh ketika nilai *order* AR = 50, fungsi kernel menggunakan *Radian Basis Function* (RBF) dengan parameter  $\gamma = 1.5$  dan parameter  $C = 1$ . Sensitivitas tertinggi metode *Burg* diperoleh ketika nilai *order* AR = 70, fungsi kernel menggunakan *Radian Basis Function* (RBF) dengan parameter  $\gamma = 1$  dan parameter  $C = 3$ .

## DAFTAR PUSTAKA

- [1] C. R. Burgess and T. E. Scammell, "Narcolepsy: Neural Mechanisms of Sleepiness and Cataplexy," *The Journal of Neuroscience*, 2012.
- [2] M. Telpin, "Fundamental of EEG Measurement," *Measurement Science Review*, Vols. 2, Section 2, 2002.
- [3] A. H. Brilian, H. Tjandrasa and C. Fatchah, "Pengenalan Sandi Morse dari Sinyal Electroencephalogram yang Direkam Perangkat Neurosky Mindwave Menggunakan Dinamic Time Warping," *JUTI*, vol. 14, pp. 63-71, 2016.
- [4] H. Tjandrasa and S. Djanali, "Classification of EEG Signals Using Single Channel Independent Component Analysis, Power Spectrum, and Linear," *Lecture Notes in Electrical Engineering (LNEE)*, Springer, vol. 387, pp. 259-268, 2016.
- [5] A. Rechtschaffen and A. Kales, *A Manual of Standardized Terminology Techniques and Scoring System for Sleep Stages of Human Subjects*, Los Angeles, USA: Brain Research Institute, UCLA, 1968.
- [6] M. S. Scher, "Automated EEG-sleep analyses and neonatal neurointensive care," *Sleep Medicine*, vol. 5, pp. 533-540, 2004.
- [7] K. A. I. Aboalayon, H. T. Ocbagabir and M. Faezipour, "Efficient Sleep Stage Classification Based on EEG Signals," *Institute of Electrical and Electronics Engineers*, 2014.
- [8] T. Kayikcioglu, M. Maleki and K. Eroglu, "Fast and Accurate PLS-Based Classification of EEG Sleep Using Single Channel Data," *Expert System with Applications*, pp. 7825-7830, 2015.
- [9] G. J. McLachlan, K.-A. Do and C. Ambroise, *Analyzing Microarray Gene Expression Data*, Wiley, 2004.
- [10] B. A. Shenoi, *Introduction To Digital Signal Processing and Filter Design*, New Jersey: Wiley Interscience, 2006.
- [11] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing*, Upper Saddle River, New Jersey: Prentice Hall, 1996.
- [12] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*, New York: John Wiley & Sons, 1996.
- [13] P. Stoica and R. L. Moses, *Introduction of Spectral Analysis*, Upper Saddle River, New Jersey: Prentice-Hall, 1997.
- [14] K. Kazlauskas, "The Burg Algorithm with Extrapolation for Improving the Frequency Estimation," *INFORMATICA*, pp. 177-188, 2011.
- [15] A. S. Nugroho, A. B. Witarto and D. Handoko, "Support Vector Machine-Teori dan Aplikasinya dalam Bioinformatika," *IlmuKomputer.Com*, 2003.