

KLASIFIKASI SUARA PEMBANGKIT *AUTONOMOUS SENSORY MERIDIAN RESPONSE* (ASMR) DENGAN METODE *K – NEAREST NEIGHBOR* DAN *LINEAR PREDICTIVE CODING*

Ilham Fiqri Muhammad^a, Iwan Iwut Tritoasmoro², Nur Ibrahim³

^{1,2,3} Universitas Telkom, Bandung

ilhamfiqri@student.telkomuniversity.ac.id¹, iwaniwut@telkomuniversity.ac.id²,

nuribrahim@telkomuniversity.ac.id³

Abstrak

Dalam beberapa tahun terakhir, ada sebuah fenomena sensorik yang luput dari perhatian penelitian ilmiah tetapi sudah dikenal oleh banyak orang. Fenomena tersebut dinamai *Autonomous Sensory Meridian Response* atau biasa disebut ASMR. Fenomena ini belum dipelajari sebelumnya terutama oleh teknologi sekarang. Orang – orang berpendapat bahwa ASMR dapat mengurangi depresi dan stress. Bagi orang yang merasakan ASMR akan merasa statis di seluruh tubuh, kulit kepala, belakang leher sebagai respon rangsangan audio dan visual dari ASMR. Penelitian ini mengidentifikasi pemicu ASMR yang biasanya dilakukan oleh orang – orang. Yaitu bisikan, perhatian pribadi, dan suara *Crips*. Sinyal suara yang dihasilkan direpresentasikan dengan fitur *Linear Predictive Coding* (LPC).

Jurnal ini akan menggunakan metode klasifikasi *K-Nearest Neighbor* (K-NN). Metode ini akan diujikan dengan objek berdasarkan data pembelajaran yang ciri-ciri nya paling dekat dengan objek tersebut. Beberapa peraturan ciri-ciri pada K-NN juga berpengaruh terhadap akurasi system pengujian. K-NN juga memiliki kelebihan yaitu lebih mudah dipahami, diimplementasikan dan komputasinya lebih efektif, lebih sederhana dan pengelompokannya lebih baik.

Tujuan dari penelitian ini adalah untuk mengklasifikasikan suara pembangkit ASMR melihat dari parameter yang dicari akurasi terbaik pada sistem. Hasil pengujian menunjukan bahwa akurasi tertinggi yang didapatkan adalah 71.11% dengan menggunakan 6 feature statistik LPC yaitu *Mean*, *Variance*, *Standar Deviasi*, *Skewness*, *Kurtois*, *Entropy* dan menggunakan *Distance Cityblock* pada K-NN dari 90 data latih dan 45 data uji. Parameter untuk klasifikasi menggunakan K-Nearest Neighbor (K-NN) adalah dengan parameter K bernilai 3 (satu) dengan distance *Cityblock*.

Keywords: *K-Nearest Neighbor* (K-NN), *Linear Predictive Coding* (LPC), *Autonomous Sensory Meridian Response* (ASMR)

1. Pendahuluan

Autonomous Sensory Meridian Response (ASMR) adalah fenomena sensorik yang belum dipelajari sebelumnya terutama yang diteliti oleh teknologi zaman sekarang. Dimana individu akan mengalami kesemutan, sensasi seperti statis di seluruh tubuh, kulit kepala, belakang leher dan kadang-kadang area lebih jauh sebagai respons terhadap rangsangan audio dan visual yang memicu spesifik. Sensasi ini secara luas dilaporkan disertai dengan perasaan rileks dan sejahtera. Studi saat ini mengidentifikasi beberapa pemicu umum yang digunakan untuk mencapai ASMR, termasuk bisikan, perhatian pribadi, suara tajam, dan gerakan lambat.[1].

Pada tugas akhir ini akan dilakukan deteksi dan penentuan suara pembangkit ASMR melalui sinyal suara pembangkit ASMR menggunakan ekstraksi ciri *Linear Predictive Coding* (LPC) kemudian akan diklasifikasikan menggunakan *K-Nearest Neighbor* (K-NN). *Linear Predictive Coding* (LPC) adalah teknik untuk menganalisis data persamaan waktu yang didapat dari analisis sistem linear. Dengan menggunakan *Linear Predictive Coding*, parameter dari sistem tersebut dapat ditentukan dengan menganalisis masukan dan keluaran sistem. [6]

Email addresses: ilhamfiqri@student.telkomuniversity.ac.id (Ilham Fiqri Muhammad),
Iwan.tritoasmoro@gmail.com (Iwan Iwut Tritoasmoro, S.T., M.T), nuribrahim@telkomuniversity.ac.id
(Nur Ibrahim, S.T., M.T)

June 14, 2021

K-Nearest Neighbor (K-NN) merupakan pengelompokan suatu data baru berdasarkan jarak data ke beberapa data. Metode ini dapat menyederhanakan algoritma perhitungan sehingga berpengaruh pada efisiensi waktu.

Tujuan Peneliti memilih metode tersebut karena hingga saat ini penelitian menggunakan metode LPC dan K-NN yang menggunakan objek suara pembangkit ASMR masih belum ada. Oleh sebab itu, penelitian ini akan menggunakan kedua metode tersebut, sehingga akan diketahui apakah metode ini dapat diimplementasikan dan memberikan hasil akurasi yang diharapkan.

Peneliti yakin bahwa penelitian ini akan berhasil dengan metode LPC dan K-NN. Karena sudah banyak penelitian yang dilakukan oleh kombinasi metode LPC dan K-NN dengan object yang bermacam – macam.

2. Dasar Teori

2.1. Autonomous Sensory Meridian Response (ASMR)

Autonomous Sensory Meridian Response (ASMR) adalah fenomena sensorik yang biasanya ditimbulkan sebagai respons terhadap rangsangan visual atau audio yang menyebabkan sensasi kesemutan statis yang berasal dari kepala dan sering menyebar ke leher dan kadang-kadang daerah lain dari tubuh. [1]

ASMR mirip dengan fenomena sensorik lainnya, seperti sensasi getaran, sensasi kesemutan yang sering dikaitkan dengan respons emosional terhadap music, dan sinestesia, suatu kondisi neurologis di mana individu mengalami indra secara bersamaan.[3]

2.2. K-Nearest Neighbor (K-NN)

K-Nearest Neighbor (K-NN) adalah sebuah metode yang digunakan untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut [10]. Algoritma ini merupakan jenis instance-based algorithm sehingga pada proses klasifikasi tidak menyimpan model dan hanya berdasarkan memori. Metode ini menyederhanakan algoritma perhitungan sehingga sudah berpengaruh pada efisiensi waktu. Untuk menentukan label data testing maka dilakukan pengambilan sebanyak k data latih terdekat. Parameter k sendiri adalah jumlah tetangga terdekat yang dilibatkan dan memiliki pengaruh dalam menentukan hasil prediksi [4].

Pada K-NN terdapat beberapa aturan jarak yang dapat digunakan, yaitu [4]:

1. Eucliden Distance, dengan rumus:

$$J(V_1, V_2) = \sqrt{\sum_{k=1}^n (V_1(k) - V_2(k))^2} \quad (1)$$

2. Cityblock atau manhattan distance, dengan rumus:

$$J(V_1, V_2) = \sum_{k=1}^n |V_1(k) - V_2(k)| \quad (2)$$

3. Cosine, dengan rumus:

$$\text{Cos}(N_i, N_j) = \frac{\sum_k N_i \cdot N_j}{\sqrt{\sum_k N_i^2} \sqrt{\sum_k N_j^2}} \quad (3)$$

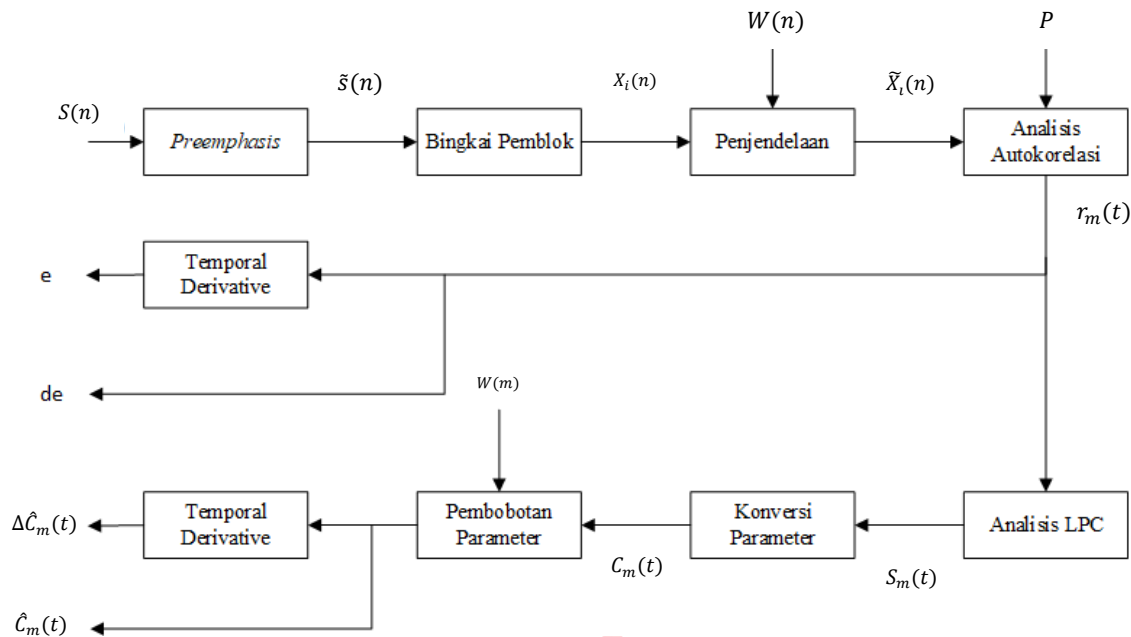
4. Minkowski, dengan rumus:

$$J(V_1, V_2) = \sqrt[\lambda]{\sum_{k=1}^N |V_1(k) - V_2(k)|^\lambda} \quad (4)$$

Dimana, V_1, V_2 merupakan data latih dan data uji, $J(V_1, V_2)$ adalah jarak J dari V_1, V_2 , N adalah nilai dimensi data, dan k adalah jumlah data dari tetangga terdekat.

2.3. Linear Predictive Coding (LPC)

Metode yang digunakan untuk proses ekstraksi ciri ini adalah Linear Prediction Coding (LPC). Diagram blok ekstraksi ciri dapat dilihat pada gambar 1 dengan langkah-langkah analisa dari LPC [5]



Gambar 1 Analisa LPC

1.3.1 Preemphasis

Preemphasis adalah salah satu jenis filter yang sering digunakan sebelum sinyal diproses lebih lanjut. Filter ini mempertahankan frekuensi-frekuensi tinggi pada sebuah spectrum, yang umumnya tereliminasi pada saat produksi suara. Tujuan dari pre-emphasis filter ini adalah:

- Mengurangi Noise ratio pada sinyal, sehingga dapat meningkatkan kualitas sinyal.
- Menyeimbangkan spektrum dari suara

$$y(n) = s(n) - \alpha s(n-1) \quad (5)$$

Dimana, $y(n)$ adalah sinyal preemphasis filter, $s(n)$ adalah sinyal sebelum preemphasis filter dan α merupakan konstanta filter preemphasis yang bernilai 0,97. Dalam bentuk dasar operator s sebagai unit filter, persamaan di atas akan memberikan sebuah transfer function filter preemphasis seperti berikut pada langkah ini sinyal yang telah di preemphasis $s(n)$ akan diblok dalam frame-frame pada tahap frame blocking.

1.3.2 Frame Blocking

Frame blocking digunakan untuk memotong-motong sinyal suara menjadi beberapa frame agar dapat diproses secara short-time untuk memperoleh karakter frekuensi yang relatif stabil. Pada langkah ini, sinyal bicara yang telah di preemphasis, diblok dalam frame-frame dengan N sample dan digeser sebesar M sample dimana: $N = 2 \times M$, sehingga didapat nilai dari sinyal yang baru, rumus nya adalah:

$$xl(n) = y(M.l + n) \quad (6)$$

Dimana, M adalah overlapping frame, n dan l adalah angka yang dimulai dari 0,1,2,3...dan $xl(n)$ adalah nilai dari frame blocking. Proses framing ini dilakukan terus sampai seluruh sinyal dapat terproses. Selain itu, proses ini umumnya dilakukan secara overlapping untuk setiap frame. Panjang daerah overlap yang umumnya digunakan adalah kurang lebih 30% sampai 50% dari panjang frame.

1.3.3 Windowing

Pada langkah ini dilakukan fungsi weighting pada setiap frame yang telah dibentuk pada langkah sebelumnya dengan tujuan untuk meminimalkan discontinuities pada ujung awal dan ujung akhir setiap frame yaitu dengan taper sinyal menuju nol pada ujung-ujungnya. Tipikal window yang digunakan pada metode auto korelasi LPC adalah hamming window yang memiliki bentuk,

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad (7)$$

Dimana, n adalah angka yang dimulai dari 0,1,2,3..., $w(n)$ adalah fungsi window, N adalah jumlah sampel dalam masing-masing frame yang merupakan kelipatan 2 dan π adalah bernilai 3,14

1.3.4 Analisa Auto Korelasi

Pada tahap ini masing – masing frame yang telah dilakukan windowing akan diauto korelasikan untuk mendapatkan:

$$rl(m) = \sum_{n=0}^{N-1-m} \tilde{x}1(n) \cdot \tilde{x}1(n+m) \quad (8)$$

Dimana, $rl(m)$ adalah hasil analisa auto korelasi, N adalah jumlah sampel, m adalah orde dari analisa LPC dan $\tilde{x}1$ adalah nilai dari frame blocking. Sedangkan auto korelasi yang tertinggi pada $m=p$ adalah orde dari analisa LPC, biasanya orde LPC tersebut 8 sampai 16. Auto korelasi ke-0 melambangkan energi dari frame yang bersangkutan dan ini merupakan salah satu keuntungan dari metode auto korelasi.

1.3.5 Analisa LPC

Langkah selanjutnya adalah analisa LPC, di mana pada tahap ini $p+1$ auto korelasi pada setiap frame diubah menjadi satu set LPC parameter yaitu koefisien LPC, koefisien pantulan (reflection coefficient), koefisien perbandingan daerah logaritmis (log area ratio coefficient). Salah satu metode untuk melakukan hal ini ialah metode Durbin yang dinyatakan dalam algoritma di bawah ini:

$$E^0 = r(0) \quad (9)$$

$$k = \frac{r(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} (r(i-j))}{E^{i-1}} \quad 1 \leq i \leq p \quad (10)$$

$$a_i^{(i)} = k_i \quad (11)$$

$$a_j^{(i)} = a_j^{(i-1)} - k_i a_{i-j}^{(i-1)} \quad i \leq j \leq i-1 \quad (12)$$

$$E^i = (1 - k_i^2) E^{i-1} \quad (13)$$

Dimana, a_m adalah koefisien LPC, k adalah koefisien PARCOR (koefisien pantulan), gm adalah koefisien perbandingan daerah logaritmis dan m adalah orde dari analisa LPC frame blocking.

1.3.6 Mengubah LPC ke Koefisien Cepstral

Sekelompok LPC parameter yang sangat penting yang dapat diperoleh dari penurunan koefisien LPC adalah koefisien cepstral $c(m)$. Persamaan yang digunakan adalah:

$$C_m = a_m + \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) \cdot C_m a_m - k \quad (14)$$

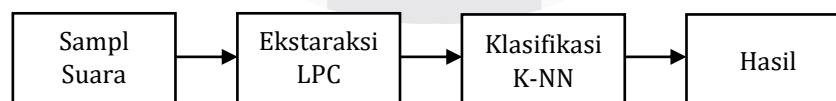
Di mana, C_m adalah koefisien cepstral, a_m adalah koefisien LPC, m adalah orde dari analisa LPC dan k adalah koefisien pantulan.

3. Sistem Model

3.1. Deskripsi Sistem

Desain Sistem

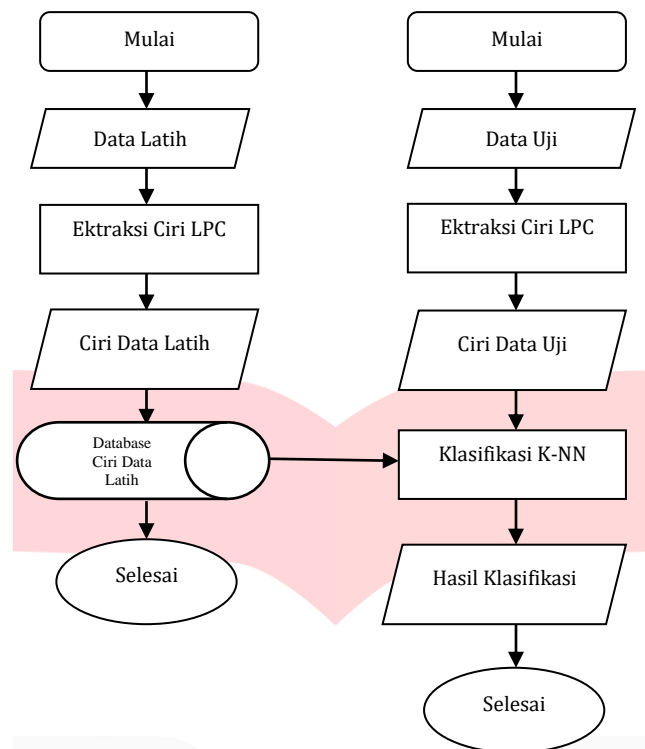
Pada tugas akhir ini penulis melakukan penelitian mengenai klasifikasi suara pembangkit ASMR menggunakan metode *K-Nearest Neighbor* (K-NN) dengan ekstraksi ciri menggunakan *Linear Predictive Coding* (LPC). Secara garis besar sistem klasifikasi ini dibagi menjadi 3 kelompok tahap yaitu tahap pengambilan sample suara, tahap ekstraksi ciri, dan tahap klasifikasi.



Gambar 2 Desain Sistem

Sampel suara merupakan tahap awal untuk mendapatkan data latih dan data uji yang diperoleh dari mengubah video ASMR yang ada di youtube.com menjadi sample suara. Selanjutnya, data latih akan diproses ekstraksi ciri oleh LPC yang nanti digunakan untuk database. Lalu pada tahap klasifikasi, data uji akan menguji sample suara dengan metode K-NN berdasarkan database yang dibuat dari ekstraksi ciri.

Sistem klasifikasi suara pembangkit ASMR ini terdiri dari dua proses yaitu proses data latih dan proses uji. Alur kerja sistem dalam proposal tugas akhir ini dapat dilihat dari gambar 3.



Gambar 3 Perancangan Sistem Data Latih dan Data Uji

Proses latih merupakan proses pembentukan sistem klasifikasi berdasarkan data latih sebagai acuan. Sedangkan proses uji merupakan proses pengujian sistem yang telah dirancang pada proses latih untuk mengklasifikasi jenis emosi dari data uji yang dipilih pada proses uji. Proses latih dan data uji secara garis besar sama hanya saja pada proses latih berarti membangun sistem klasifikasi menggunakan K-Nearest Neighbor (K-NN) yang menghasilkan model klasifikasi K-NN yaitu parameter K-NN yang digunakan pada proses uji klasifikasi.

3.2 Pengambilan Sample Suara

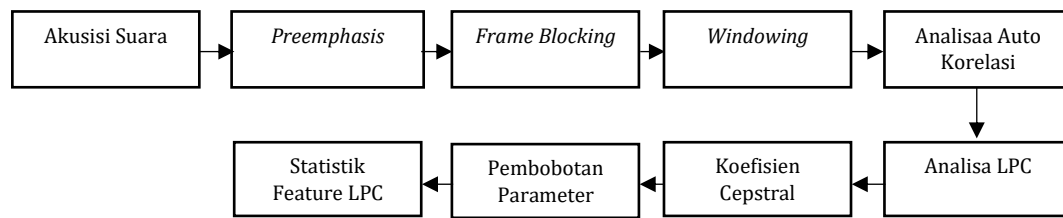
Pengambilan sample suara merupakan tahap awal untuk mendapatkan data latih dan data uji pada frekuensi audio. Pada proses akuisisi ini dilakukan dengan cara mengambil beberapa video ASMR yang ada di youtube.com. Selanjutnya, Video akan dipotong – potong menjadi sample lalu dikonversi ke format WAV. Pengolahan audio yang didapat akan diproses dalam tahapan ekstraksi ciri.

3.3 Ekstraksi Ciri Linear Predictive Coding (LPC)

Untuk menentukan klasifikasi dari suara pembangkit ASMR dibutuhkan data akustik dari sinyal audio tersebut. Ekstraksi ciri menggunakan LPC adalah cara untuk mendapatkan informasi-informasi yang diperlukan dalam perhitungan statistik untuk menentukan suara pembangkit ASMR. LPC biasanya digunakan untuk pengenalan ucapan karena beberapa keuntungan yaitu:

- LPC menyediakan pemodelan yang bagus untuk sinyal ucapan (speech signal)
- LPC mudah diterapkan baik menggunakan perangkat keras maupun perangkat lunak karena perhitungan matematis yang digunakan relative lebih singkat
- Hasil pengenalan ucapan yang didapatkan dengan menggunakan LPC cukup baik, bahkan lebih baik dari metode-metode sebelumnya

Pada gambar 3.3 merupakan diagram alir proses ekstraksi ciri dengan metode LPC.



Gambar 4 Diagram Blok ekstraksi ciri LPC.

Proses diagram alir pada gambar 4 adalah ekstraksi ciri menggunakan LPC, berikut penjelasannya [5]:

3.3.1 Preemphasis

Preemphasis adalah upaya cara penyesuaian agar level base-band pada bagian frekuensi tinggi masih memiliki kualitas sinyal yang baik (carrier to noise ratio yang baik) penyesuaian / penguatan ini diperhitungkan berdasarkan kenaikan noise pada daerah ini sehingga dibutuhkan equalizer (penyelaras) yang dapat mengatur tinggi sinyal terhadap level noise yang ada.

3.3.2 Frame Blocking

Pada proses ini, sinyal hasil *preemphasis* akan dilakukan *framing*. Setiap frame akan berisikan sejumlah sampel dan antara frame yang satu dengan frame yang lainnya akan saling overlapping.

3.3.3 Windowing

Windowing dilakukan untuk mengurangi diskontinuitas sinyal yang dihasilkan dari proses *framing*, *windowing* dilakukan pada setiap *frame* menggunakan *hamming window*, hal ini dikarenakan *hamming window* menghasilkan *sidelobe* yang tidak terlalu tinggi akan tetapi menghasilkan *noise* yang tidak terlalu tinggi

3.3.4 Analisa Auto Korelasi

Analisa auto korelasi pada sistem yang digunakan adalah orde pada LPC yaitu orde 16. Pada analisa ini auto korelasi ke-0 melambangkan energi dari *frame* yang bersangkutan dan ini merupakan salah satu keuntungan dari metode auto korelasi.

3.3.5 Analisa LPC

Langkah selanjutnya adalah analisa LPC, di mana pada tahap ini auto korelasi pada setiap *frame* diubah menjadi satu set LPC parameter yaitu koefisien LPC, koefisien pantulan (*reflection coefficient*), koefisien perbandingan daerah logaritmis (*log area ratio coefficient*).

3.3.6 Mengubah LPC Parameter ke Koefisien Cepstral

Koefisien *cepstral* dilakukan untuk memperoleh penurunan pada koefisien LPC, di mana pada parameter LPC sangat penting untuk mengubah koefisien *cepstral* agar sistem mendapatkan ciri yang tetap.

3.3.7 Pembobotan Parameter

Pembobotan parameter dilakukan karena koefisien *cepstral* orde rendah sensitif terhadap kemiringan spektrum dan koefisien *cepstral* orde tinggi sensitif terhadap derau, maka dilakukan pembobotan koefisien *cepstral* dengan jendela penyedap sehingga meminimalkan sensitivitas tersebut

3.3.8 Statistik Feature LPC

Ekstraksi ciri menggunakan LPC dengan mencari nilai pada metode statistik yaitu *mean*, *variance*, standar deviasi, *skewness*, *kurtois* dan *entropy*. Metode statistik merupakan prosedur-prosedur yang digunakan dalam penyajian data yang meliputi pengumpulan dan penyajian data. Selain penyajian digunakan juga untuk penafsiran data yang meliputi pendugaan, pengujian dugaan dan penarikan kesimpulan [7].

Ekstraksi ciri merupakan suatu proses mengambil karakteristik yang terdapat di dalam sinyal, dan menjadi tahap kunci dari keseluruhan skema klasifikasi dan pengenalan pola [8]. Ciri statistik dapat dilihat pada tabel 3.1 [9].

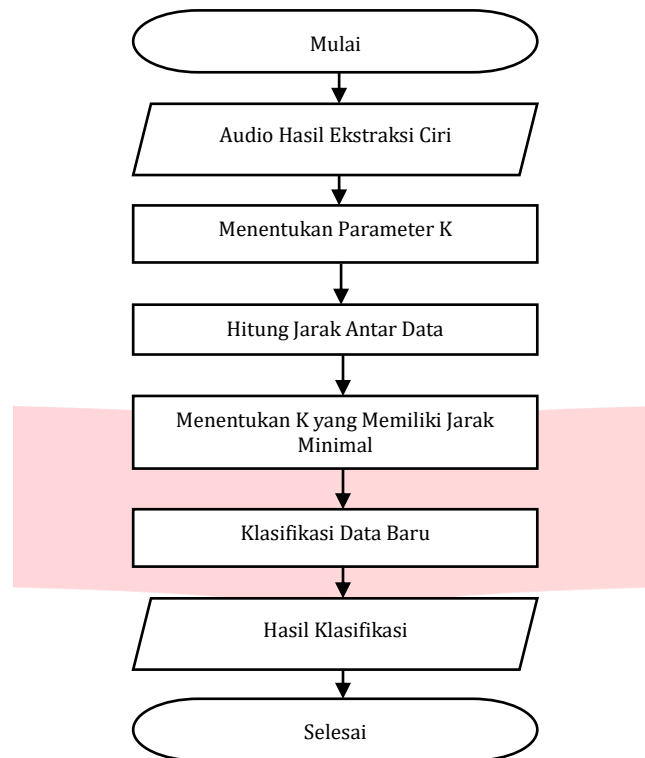
Tabel 1 Feature Statistik

Metode	Formula	Definisi
Mean	$\mu = \frac{1}{N} \sum_{i=1}^N X_i$	Intensitas rata – rata
Variance	$SR = \frac{\sum (X_i - \bar{X})^2}{n}$	Jumlah kuadrat semua deviasi nilai – nilai individual terhadap rata – rata kelompok
Standar Deviasi	$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2}$	Variasi atau disperse dari nilai – nilai data
Skewness	$sk = \frac{E(x - \mu)^3}{\sigma^3} = \frac{1}{\sigma^3} \sum_{i=1}^N (X_i - \mu)^3$	Kecondongan sebaran data
Kurtois	$\sigma^4 = \frac{m_4}{m_2^2}$	Intensitas puncak dari sebaran data
Entropy	$H = - \sum_{i=0}^N p_i \cdot \log(p_i)$	Keacakan data

Dimana N adalah jumlah sampel, X_i adalah nilai data ke-i, \bar{X} adalah nilai rata-rata, n adalah banyaknya data.

3.4 Pengujian Menggunakan K-Nearest Neighbor (K-NN)

Prinsip dasar dari metode *K-Nearest Neighbor* (K-NN) adalah mengukur jarak antara pola sinyal yang tidak diketahui dengan pola sinyal referensi yang ada dalam database. Analisa yang dilakukan pada K-NN adalah pengaruh nilai parameter k yang digunakan dan jenis jarak yang digunakan terhadap akurasi sistem dalam mengklasifikasi suara. Nilai k yang diuji adalah 1, 3, 5, dan 7. Jenis jarak yang diuji adalah *Euclidean*, *Cityblock*, *Cosine*, dan *Correlation*. Diagram alir klasifikasi dalam penelitian ini dapat dilihat pada gambar 4



Gambar 4 Diagram Blok Klasifikasi

Tahap pertama dalam klasifikasi K-NN adalah menentukan parameter k yang digunakan. Parameter k adalah variabel yang bernilai ganjil, dipilihnya nilai parameter k yang ganjil agar mengurangi kesalahan algoritma jika peluang kemiripannya sama. Setelah menentukan nilai parameter k , proses berikutnya adalah menghitung jarak antara data uji ke data latih yang telah disimpan sebelumnya. Banyaknya parameter k sudah mempengaruhi terhadap penentuan jarak ke beberapa data latih sesuai dengan jumlah parameter k . klasifikasi data baru dilakukan berdasarkan hasil perhitungan jarak minimal ke beberapa titik sesuai jumlah parameter k . Klasifikasi K-NN mengklasifikasikan data uji tersebut sesuai dengan banyaknya kelas yang dominan pada saat perhitungan jarak minimal.

3.5 Parameter Performansi Sistem

Untuk mengevaluasi performansi sistem yang dibahas, diperlukan pengujian terhadap data latih dan data uji menggunakan perangkat lunak *Matlab* untuk menentukan klasifikasi suara pembangkit ASMR melalui sinyal suara. Pengujian dilakukan untuk mengetahui kelebihan dan kekurangan sistem. Parameter yang digunakan untuk mengetahui performansi sistem yaitu akurasi dan waktu komputasi.

3.5.1 Akurasi

Akurasi adalah ukuran ketepatan sistem dalam mengenali input yang akan diberikan sehingga menghasilkan keluaran yang benar. Secara sistematis dapat dituliskan sebagai berikut:

$$A = \frac{N_B}{N_T} \times 100 \% \quad (15)$$

Dimana A adalah akurasi, N_B adalah jumlah data benar, N_T adalah jumlah data keseluruhan.

3.5.2 Waktu Komputasi

Waktu komputasi adalah waktu yang dibutuhkan sistem melakukan suatu proses. Secara sistematis dapat dituliskan sebagai berikut:

$$W_k = W_s - W_m \quad (16)$$

Dimana W_k adalah waktu komputasi, W_s adalah waktu selesai, W_m adalah waktu mulai.

4. Simulasi Dan Analisis

4.1. Analisis Pengujian Sistem

Pengujian sistem dilakukan berdasarkan beberapa parameter yang diperoleh selama proses pemograman sistem. Masing-masing pengujian memiliki hasil yang direpresentasikan dalam bentuk table.

Pada pengujian ini dilakukan untuk mendapatkan akurasi dari proses klasifikasi suara pembangkit ASMR dengan metode *Linear Predictive Coding* (LPC) dan *K-Nearest Neighbor* (K-NN).

4.1.1. Pengujian Dengan Metode *Linear Predictive Coding* (LPC)

Pengujian pertama dilakukan untuk mendapatkan informasi – informasi yang diperlukan dalam perhitungan statistik. Pada pengujian ini, data sample yang sudah diambil akan didata yang nantinya oleh *Linear Predictive Coding* (LPC) akan dimunculkan ciri – ciri setiap sample suara. Setiap variabel ucapan diambil sample suara sebanyak 135 sample yang diambil dari video ASMR di Youtube.com yang terdiri dari 90 data latih, dan 45 data uji. lalu 6 ciri yang terdiri dari *Mean*, *Variance*, *Standar Deviasi*, *Skewness*, *Entropi*, dan *Kurtosis* berupa nilai koefisien hasil pengolahan *Linear Predictive Coding* (LPC), sedangkan jumlah variabel terdiri dari 3 variabel yaitu Bisikan (*Whisper*), Perhatian Personal (*Personal Attention*), dan Suara Renyah (*Crisp Sound*). keseluruhan data sample (data latih dan data uji) diekstraksi dengan menggunakan metode *Linear Predictive Coding* (LPC) setiap data dihasilkan 6 ciri dengan nilai koefisiennya. Adapun nilai koefisien dari 6 ciri bisa dilihat pada tabel 4.1 dan tabel 4.2, dimana Table 4.1 adalah hasil dari *Linear Predictive Coding* (LPC) mengekstraksi 90 sample data latih yang satu sample nya memiliki 6 karakter ciri dengan koefisien berbeda setiap sample nya. Dan table 4.2 adalah hasil dari *Linear Predictive Coding* (LPC) mengekstraksi data uji yang satu sample nya memiliki 6 karakter ciri dengan koefien berbeda setiap sample nya.

Tabel 2 Hasil rata-rata dari nilai data latih *Linear Predictive Coding* (LPC)

Jenis Sample	<i>Mean</i>	<i>Variance</i>	<i>Standar Deviasi</i>	<i>Skewness</i>	<i>Entropi</i>	<i>Kurtosis</i>
Chrip	-1,4677	1,1735	-0,5316	-1,4750	1,1574	-0,5087
Personal Attention	-1,6337	1,2821	-0,5367	-1,6200	1,2280	-0,5092
Whisper	-2,3411	2,0599	-0,7081	-2,2289	1,9874	-0,7055

Berdasarkan table 2, setiap jenis sample memiliki nilai rata-rata ciri data latih *Linear Predictive Coding* (LPC). Nilai tersebut nantinya akan dibandingkan dengan data uji yang selanjutnya akan klasifikasikan oleh *K-Nearest Neighbor* (K-NN).

Tabel 3 Hasil rata-rata dari nilai data uji *Linear Predictive Coding* (LPC)

Jenis Sample	Mean	Variance	Standar Deviasi	Skewness	Entropi	Kurtosis
Chrisp	-1,4726	1,1646	-0,5809	-1,5017	1,1381	-0,5255
Personal Attention	-1,4864	1,1921	-0,5602	-1,5409	1,2079	-0,5434
Whisper	-2,2372	1,9278	-0,6810	-2,1639	1,8754	-0,6508

Berdasarkan table 3, setiap jenis sample memiliki nilai rata-rata ciri data uji *Linear Predictive Coding* (LPC). Nilai tersebut nantinya akan menjadi database untuk klasifikasi oleh *K-Nearest Neighbor* (K-NN).

4.1.2 Pengujian Pengaruh Parameter Distance pada Metode *K-Nearest Neighbor* (K-NN) Terhadap Akurasi

Pada pengujian ini dilakukan mencari pengaruh nilai parameter K dan aturan Jarak pada *K-Nearest Neighbor* (K-NN). Nilai K yang akan diujikan adalah 1, 3, 5, 7. Parameter K sendiri adalah jumlah tetangga terdekat yang dilibatkan dan memiliki pengaruh dalam menentukan hasil prediksi [4]. Dipilihnya nilai parameter K yang ganjil agar mengurangi kesalahan algoritma jika peluang kemiripannya sama. Lalu parameter jarak yang digunakan yaitu *Euclidean*, *Cityblock*, *Cosine*, dan *Minkowski*.

Dibawah ini adalah hasil dari klasifikasi yang diproses oleh metode *K-Nearest Neighbor* (K-NN) dengan merubah parameter K dan aturan jarak nya.

Tabel 4 Hasil Akurasi dari pengaruh nilai parameter K dan aturan jarak pada *K-Nearest Neighbor* (K-NN)

Aturan Jarak	Parameter K	Akurasi	Waktu Komputasi (Sec)
Cosine	1	64.4%	1.556
	3	66.7%	1.679
	5	64.4%	1.527
	7	62.2%	1.572
Cityblock	1	68.9%	1.914
	3	71.1%	1.484
	5	68.9%	1.554
	7	66.7%	1.564
Euclidean	1	66.7%	1.920
	3	71.1%	1.540
	5	71.1%	1.529
	7	71.1%	1.511
Minkowski	1	66.7%	1.546
	3	71.1%	1.556
	5	71.1%	1.516
	7	71.1%	1.776

Tabel 4 adalah nilai akurasi yang didapat dari proses klasifikasi menggunakan metode *K-Nearest Neighbor* (K-NN) yang dipengaruhi parameter K dan aturan jarak.

Pada grafik bisa dilihat bahwa block berwarna orange menandakan parameter K bernilai 1. Lalu untuk block berwarna abu – abu menandakan parameter K bernilai 3. Selanjutnya untuk block berwarna kuning menandakan parameter K bernilai 5. Dan terakhir, untuk block berwarna biru menandakan parameter K bernilai 7.

Selanjutnya adalah nilai yang berada diatas block – block adalah nilai akurasi dari setiap kombinasi antara parameter K dan aturan jarak nya yaitu *Cosine*, *Cityblock*, *Euclidean*, dan *Minkowski*.

Dari grafik 4.1 terlihat bahwa saat aturan jarak yang digunakan adalah *Cosine* akurasi nya 64,4% saat parameter K nya bernilai 1, lalu akurasi nya bernilai 66,7% saat parameter K nya bernilai 3, selanjutnya akurasi bernilai 64,4% saat parameter K bernilai 5, dan akurasi bernilai 62,3% saat parameter K bernilai 7.

Pada saat aturan jarak yang digunakan adalah *Cityblock*, akurasi yang didapatkan bernilai 68,9% saat parameter K nya 1, lalu saat parameter K nya bernilai 3, akurasi yang didapatkan adalah 71,1%. Begitupula saat parameter K nya bernilai 5 dan 7, akurasi yang didapatkan sebesar 68,9% dan 66,7%.

Selanjutnya hal yang sama terjadi saat aturan jarak digunakan adalah *Euclidean* dan *Minkowski*. Kedua aturan jarak tersebut menghasilkan akurasi yang sama saat parameter K nya 1, 3, 5, dan 7. Ketika parameter K nya 1 akurasi yang didapatkan sebesar 66,7%. Dan saat parameter K bernilai 3, 5, dan 7 didapatkan akurasi sebesar 71,1%.

Dari grafik 4.1 dapat dianalisis bahwa nilai akurasi tertinggi yaitu 71,1% didapatkan saat aturan jarak *Cityblock* dengan parameter K bernilai 3, aturan jarak *Euclidean* saat parameter K bernilai 3, 5, dan 7, dan saat aturan jarak yang digunakan adalah *Minkowski* pada parameter K bernilai 3, 5, dan 7.

Tetapi menurut waktu komputasi, saat aturan jarak *Cityblock* dengan parameter K nya 3, waktu komputasi nya 1,484 detik. Berbeda Ketika aturan jarak yang digunakan adalah *Euclidean*. Saat aturan jarak yang digunakan adalah *Euclidean* dan parameter K nya bernilai 3, 5, dan 7, waktu komputasi nya 1,540, 1,529, 1,511. Dan juga saat aturan jarak yang digunakan adalah *Minkowski* dan parameter K nya bernilai 3, 5, dan 7, waktu komputasi nya 1,556, 1,516, 1,776.

Dengan begitu, akurasi tertinggi didapat saat aturan jarak yang digunakan adalah *Cityblock* dengan akurasi 71,1% dan waktu komputasi nya 1,484 detik.

5. Kesimpulan

Dari hasil pengujian dan analisis yang telah dilakukan pada sistem deteksi suara pembangkit ASMR, dapat diambil kesimpulan sebagai berikut:

1. Perancangan dan perealisasi sistem untuk mengklasifikasi suara pembangkit ASMR dengan metode Linear Predictive Coding (LPC) dan klasifikasi K-Nearest Neighbor (K-NN) terhadap data yang didapatkan **berhasil dilakukan**. Dengan akurasi yang diperoleh sebesar 71.11 %, dengan kondisi 22 suara teridentifikasi sesuai dengan identitas sebenarnya 8 suara teridentifikasi tetapi tidak sesuai dengan identitas sebenarnya.
2. Pada metode klasifikasi K-NN hasil ekstraksi ciri yang telah didapatkan dengan melihat 6 feature statistic yaitu Mean, Variance, Standar Deviasi, Skewness, Entropi, dan Kurtosis. Akurasi terbaik dicapai dengan menggunakan aturan jarak K-Nearest Neighbor (K-NN) Cityblock dan nilai parameter K nya 3.
3. Terdapat beberapa kombinasi antara aturan jarak dan parameter K yang mendapat akurasi tinggi yaitu 77,11%. Yaitu saat aturan jarak Cityblock dengan parameter K nya 3. Ketika aturan jarak yang digunakan adalah Euclidean, parameter K nya bernilai 3, 5, dan 7. Dan ketika aturan jarak yang digunakan adalah Minkowski, parameter K nya bernilai 3, 5, dan 7. Tetapi aturan jarak Cityblock yang memiliki akurasi paling tinggi karena waktu komputasi yang lebih singkat, yaitu sebesar 1,484 detik.

REFERENSI

- [1] Emma L Barrat dan Nick J Davis, *Autonomous Sensory Meridian Response (ASMR): a flow-like mental state*, peerj.com, 2015.
- [2] Emma L Barrat, Charles Spence, dan Nick J Davis, *Sensory determinants of the autonomous sensory meridian response (ASMR): understanding the triggers*, peerj.com, 2017.
- [3] Del Campo MA, Kehle TJ. *Autonomous Sensory Meridian Response (ASMR) and frisson: mindfully induced sensory phenomena that promote happiness*. International Journal of School & Educational Psychology 4(2):99–105, 2015.
- [4] R. Nurpalida, “Analisis dan Implementasi Prediksi Churn pada Perusahaan Telekomunikasi Menggunakan Metode NCL DAN K-Nearest Neighbor”. Bandung 2017.

- [5] H. Saputra, Sistem Pengenalan Kata dengan Menggunakan, vol. 5, no. September, pp. 1924, 2005.
- [6] A.Yahya and Suryanto, Deteksi Emosi Melalui Pengenalan Suara Menggunakan Linear Predictive Coding (LPC) Dan Hidden Markov Model (HMM), Tugas Akhir, 2012.
- [7] Erdoria Kristina, Jong Jek Siang dan Gunawan Santosa."Penerapan Metode Statistik dan Average Energy Untuk Menguji Tingkat Kemiripan Pada Identifikasi Suara". Jurnal Informatika,vol.7, no.1, Hal.3. 2011.
- [8] B. Boashah, N. A. Khan, and T. Ben-Jabeur,"Time-Frequency Features for pattern recognition using high-resolution TFDs: A tutorial review,"Digit.Signal process. A rev. J., vol.40, no.1, pp.1-30, 2015.
- [9] Domy Kristomo. Seleksi Ciri Suara Jantung Pada Domain Waktu dan Frekuensi. Seminar Riset Teknologi Informasi (SRITI). 2016

