

## IMPLEMENTASI ALGORITMA BFCC DAN kNN PADA *EMBEDDED SYSTEM* UNTUK DETEKSI DINI *BRONCHITIS*

Septiyo Budi Perkasa<sup>1</sup>, Barlian Henryranu Prasetio<sup>\*2</sup>, Eko Setiawan<sup>3</sup>,  
Edita Rosana Widasari<sup>4</sup>, Dahnia Syauqy<sup>5</sup>

<sup>1,2,3,4,5</sup>Universitas Brawijaya, Malang

Email: <sup>1</sup>septyoperkasa0@student.ub.ac.id, <sup>2</sup>barlian@ub.ac.id, <sup>3</sup>ekosetiawan@ub.ac.id

<sup>4</sup>editarosana@ub.ac.id, <sup>5</sup>dahnial87@ub.ac.id

\*Penulis Korespondensi

(Naskah masuk: 27 September 2022, diterima untuk diterbitkan: 20 Juni 2023)

### Abstrak

World Health Organization (WHO) menyatakan bahwa sebanyak 600 juta orang di dunia menderita *bronchitis*. *Bronchitis* merupakan salah satu penyakit pernafasan yang dapat disebabkan oleh virus *Respiratory Syncytial Virus (RSV)* dan *Rhinovirus*. Gejala umum *bronchitis* adalah seseorang akan mengalami kesulitan bernafas dengan disertai batuk. Namun, tidak sedikit orang mengabaikan gejala umum ini sehingga berindikasi mengalami *bronchitis* tingkat berat ataupun berpotensi kematian. Oleh karena itu, dalam paper ini mengusulkan sistem deteksi dini *bronchitis* berdasarkan suara batuk berbasis *embedded system*. Ini merupakan terobosan baru pada dunia medis dengan desain alat kesehatan yang portabel. Sistem yang diusulkan menerapkan algoritma *Bark Frequency Cepstral Coefficients (BFCC)* dan *K-Nearest Neighbor (kNN)*. BFCC merupakan algoritma yang digunakan untuk mengekstraksi fitur suara batuk dan menghasilkan nilai koefisien *cepstral*. Selanjutnya, nilai koefisien *cepstral* tersebut dihitung jarak *Euclidean*-nya untuk dapat diklasifikasikan menggunakan kNN. Algoritma BFCC dan kNN diimplementasikan pada perangkat Mini Komputer *Raspberry Pi 3 Model B+* dengan mikrofon sebagai perangkat masukan suara dan perangkat LCD *touchscreen 3.5 inchi* untuk sebagai antarmuka yang menampilkan keluaran hasil deteksi. Hasil pengujian menunjukkan rata-rata waktu komputasi sebesar 4,452 detik dan penggunaan CPU sebesar 26%, serta akurasi kNN sebesar 73% untuk perhitungan jarak *Euclidean* dengan nilai *neighbor = 5*.

**Kata kunci:** BFCC, Bronchitis, Embedded system, KNN, Suara Batuk

## EARLY BRONCHITIS DETECTION SYSTEM USING EMBEDDED SYSTEM BASED ON BFCC AND KNN ALGORITHM

### Abstract

The World Health Organization (WHO) states that as many as 600 million people in the world suffer from bronchitis. Bronchitis is a disease that can be caused by respiratory syncytial virus (RSV) and rhinovirus. Symptoms of common bronchitis a person will experience difficulty breathing accompanied by coughing. Unfortunately, many people underestimate this common symptom. Even though, it is indicating that they have severe bronchitis or possibly death. Therefore, this study proposes an early detection system for bronchitis based on cough e sounds based on an embedded system. This is a new breakthrough in the medical world with a portable medical device design. The proposed system applied the Bark Frequency Cepstral Coefficients (BFCC) and K-Nearest Neighbor (kNN) algorithms. BFCC is an algorithm that is used to extract cough sound features and produce cepstral coefficient values. Furthermore, the value of the cepstral coefficient will be calculated for the Euclidean distance to be classified using kNN. The implementation of the BFCC and kNN algorithms is carried out on a Raspberry Pi 3 Mini Computer Model B+ with a microphone as a voice input device and a 3.5-inch LCD touchscreen device to display the resulting output interface. The results obtained an average computation time of 4.452 seconds and CPU usage of 26%, and kNN accuracy of 73% from the calculation of the Euclidean distance with a neighbor value = 5.

**Keywords:** BFCC, Bronchitis, Embedded system, KNN, Cough Sound

## 1. PENDAHULUAN

*Bronchitis* merupakan penyakit tidak menular yang menyerang saluran pernapasan dengan disebabkan oleh virus *Respiratory Syncytial Virus (RSV)* dan *Rhinovirus*. Ada beberapa faktor lain yang menyebabkan seseorang terjangkit penyakit *bronchitis* seperti merokok aktif atau pasif, menghirup asap atau debu, dan polusi udara sekitar yang mengakibatkan iritasi pada saluran pernapasan (Maghfiroh dkk, 2021).

*World Health Organization (WHO)* menyatakan bahwa sebanyak 600 juta orang di dunia menderita *bronchitis* (WHO, 2021) (Dotan dkk, 2019). 65 juta orang diantaranya menderita *bronchitis* tingkat sedang hingga berat (Hidayatulloh, 2018). Tingginya angka penderita *bronchitis* tingkat sedang hingga berat tersebut dikarenakan masih banyak orang di dunia yang menyepelekan gejala umum penyakit *bronchitis* sehingga berindikasi mengalami *bronchitis* tingkat berat ataupun berpotensi kematian (Mathew, 2022). Gejala awal *bronchitis* adalah seseorang akan mengalami hidung berair, tidak enak badan, menggigil, pegal-pegal, sakit kepala, dan sakit tenggorokan. Selanjutnya, seseorang akan mengalami kesulitan bernafas dengan disertai batuk yang mana merupakan gejala umum dari *bronchitis* (Singh, 2022). Oleh karena itu, sistem deteksi dini *bronchitis* berdasarkan suara batuk dengan berbasis *embedded system* diusulkan guna mendeteksi ataupun mencegah penyakit *bronchitis* semakin memburuk.

Pada penelitian (Belkacem dkk, 2021) telah membahas hipotesis medis dari sistem cerdas yang digunakan untuk merekam data dari pasien dengan gejala batuk yang dialami. Batuk tersebut di ubah menjadi data medis untuk mendeteksi terkena penyakit pernafasan atau tidak. Penelitian ini menjelaskan bagaimana dapat membangun sebuah sistem pendeteksian penyakit melalui suara batuk. Perangkat keras dirancang setelah suara batuk diproses melalui perangkat lunak dengan algoritma prediksi yang bergantung dari model kecerdasan buatan yang diusulkan. Selanjutnya (Kaminska dkk, 2017) telah melakukan parameterisasi ucapan menggunakan ekstraksi fitur *Bark Frequency Cepstral Coefficients (BFCC)*. Kemudian, dua algoritma klasifikasi yaitu *K-Nearest Neighbor (kNN)* dan *Support Vector Machine (SVM)* dibandingkan untuk mengetahui hasil akurasi pengenalan ucapan tersebut. kNN dengan ekstraksi fitur BFCC mendapatkan hasil akurasi lebih dari 60%, sedangkan klasifikasi SVM dengan BFCC mendapatkan hasil akurasi 35%. Hal ini menunjukkan penguasaan algoritma BFCC pada sistem pengenalan ucapan dan klasifikasi kNN mendapatkan hasil yang lebih baik dari SVM.

(Putra, 2017) juga telah menyatakan bahwa penggunaan algoritma BFCC untuk ekstraksi fitur suara lebih baik daripada *Mel-Frequency Cepstral Coefficient (MFCC)*. Hal ini dikarenakan dari segi

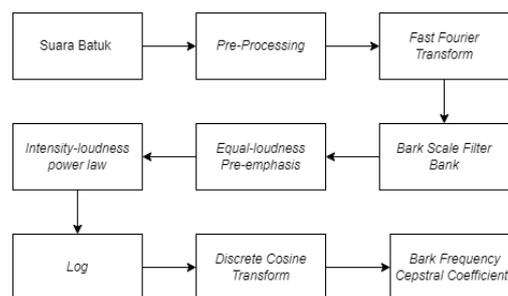
pemrosesan BFCC sangat sederhana dan tidak menggunakan *resource* yang cukup besar. Oleh karena itu, sistem yang diusulkan menerapkan algoritma BFCC dan kNN.

Pertama, sistem melakukan pengenalan suara batuk, dimana sinyal yang dihasilkan oleh suara batuk akan diekstraksi fiturnya menggunakan algoritma BFCC. Kemudian, akan menghasilkan nilai koefisien *cepstral*. Selanjutnya, nilai koefisien *cepstral* tersebut dihitung jarak *Euclidean*-nya untuk dapat diklasifikasikan menggunakan kNN dan dilakukan pengklasifikasian berdasarkan kelas yang ditentukan: *bronchitis* dan *non-bronchitis*. Implementasi algoritma BFCC dan kNN tersebut dilakukan pada perangkat Mini Komputer *Raspberry Pi 3 Model B+* dengan mikrofon sebagai perangkat masukan suara dan perangkat LCD *touchscreen 3.5 inchi* untuk menampilkan antarmuka keluaran yang dihasilkan.

## 2. METODE PENELITIAN

### 2.1 Bark Frequency Cepstral Coefficients (BFCC)

BFCC merupakan salah satu algoritma yang biasa digunakan untuk mengekstraksi fitur sinyal suara. Sinyal suara yang keluar dari sistem dikarenakan oleh ekstraksi masukan dan juga respon dari sistem itu sendiri. Algoritma ini memadukan antara spektrum dengan transformasi cos untuk menghasilkan nilai koefisien *cepstral* (Putra, 2017). Proses BFCC terdiri dari perekaman suara batuk *pre-processing*, *fast fourier transform*, *bark filter bank*, *intensity power*, *log*, *discrete cosine transforms*, dan *bark-frequency cepstral coefficients* seperti yang ditunjukkan pada Gambar 1.

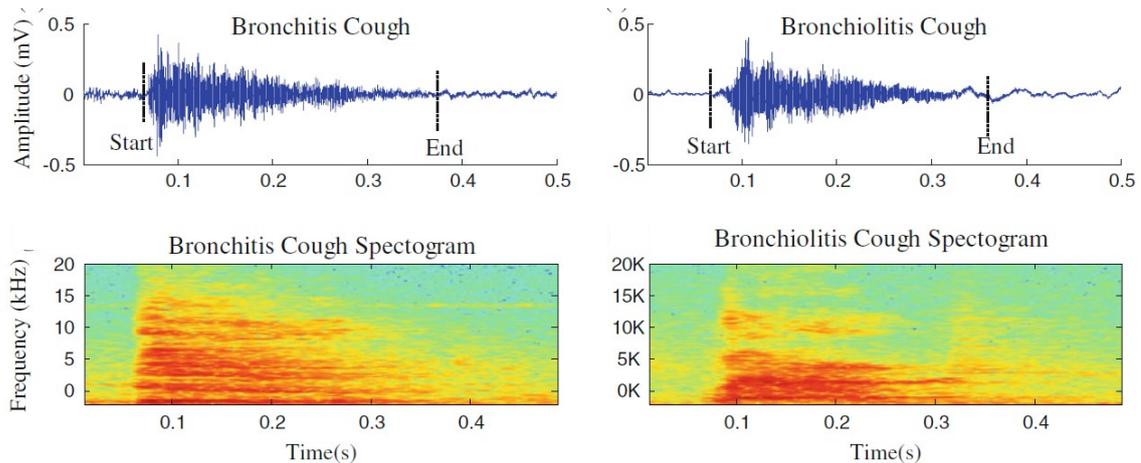


Gambar 1. Diagram blok proses BFCC

#### 2.1.1 Suara Batuk

Pada tahap ini suara batuk yang akan direkam dan disimpan dengan format *.wav* file. Kemudian, suara batuk diproses sebagai indikator deteksi dini penyakit *bronchitis*.

Suara batuk yang disebabkan oleh penyakit tertentu memiliki karakteristik yang berbeda, seperti pneumonia, asma, bronchiolitis maupun bronchitis, seperti ditunjukkan pada Gambar 2 (Abeyratne dkk, 2013).



Gambar 2. Frekuensi spektrogram suara batuk

Gambar 2 menunjukkan perbedaan di antara suara batuk yang mendukung hipotesis bahwa batuk membawa informasi tentang keadaan saluran pernapasan bagian bawah. Untuk suara batuk

Dalam penelitian ini, sistem dilatih menggunakan dataset COUGHVID (Lara dkk, 2020). Dataset COUGHVID menyediakan lebih dari 20.000 rekaman batuk bersumber dari banyak orang yang mewakili berbagai usia subjek, jenis kelamin, lokasi geografis, dan berbagai penyakit, termasuk Bronchitis. Pertama, dataset difilter menggunakan algoritma deteksi batuk. Kedua, ahli paru berpengalaman memberi label lebih dari 2.000 rekaman untuk mendiagnosis kelainan medis yang ada pada batuk, sehingga memberikan kontribusi sebagai dataset batuk berlabel ahli terbesar yang ada yang dapat digunakan untuk banyak tugas klasifikasi audio batuk.

### 2.1.2 Pre-Processing

Tahap *pre-processing* merupakan tahap persiapan saat ingin melakukan ekstraksi sebuah fitur. *Pre-processing* ini terdiri dari tiga proses lain seperti normalisasi, *framing*, dan *windowing*.

#### 1) Normalisasi

Proses ini melakukan normalisasi pada file rekaman suara batuk yang disimpan menggunakan format *.wav* supaya amplitudo yang dihasilkan lebih baik dan melakukan perubahan file suara *.wav* dengan stereo nanti akan diubah menjadi mono.

#### 2) Framing

Proses *framing* ini bermaksud menentukan suara yang akan digunakan dari sebuah suara yang panjang. Setelah itu untuk ukuran frame harus cukup pendek untuk menghindari hilang karakteristik suara tersebut. Hal ini guna mendapatkan karakteristik sinyal suara yang lebih stabil.

#### 3) Windowing

bronchitis cenderung lebih panjang dengan spektrum daya yang meluas hingga 15kHz. Durasi rata-rata batuk bronchitis adalah 2.7 detik.

Proses *framing* menyebabkan kebocoran *spectral (magnitude leak)* atau *aliasing*. *Aliasing* adalah sinyal baru yang frekuensinya berbeda dengan sinyal aslinya. Efek ini mungkin disebabkan oleh *framing* yang menyebabkan sinyal menjadi terputus. Oleh karena itu, diperlukan proses *windowing* yang bertujuan untuk mengurangi terjadinya *aliasing* tersebut. Proses *windowing* sesuai dengan Persamaan (1), dimana  $M$  merupakan jumlah sampel (sebesar ukuran FFT = 2048) dan  $N$  merupakan sampel ke- $n$  (Kumar dkk, 2018).

$$w(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{M-1}\right); 0 \leq n \leq M-1 \quad (1)$$

### 2.1.3 Fast Fourier Transform

Tahap *Fast Fourier Transform* (FFT) merupakan metode transformasi yang mempunyai kecepatan dan efisiensi lebih baik dari *fourier* diskrit saat memproses data dan biasa digunakan untuk menganalisa frekuensi. Penggunaan FFT dalam pemrosesan sinyal suara cukup baik, dikarenakan sinyal suara akan diubah menjadi bentuk spektrum frekuensi agar memudahkan saat menganalisa dari suara yang sudah dilakukan perekaman suara batuk. FFT jika digunakan untuk jumlah data yang begitu besar juga dapat mengurangi waktu komputasi menjadi lebih cepat. Proses FFT sesuai dengan Persamaan (2), dimana  $s(t)$  adalah sinyal domain waktu,  $s(f)$  adalah sinyal domain frekuensi,  $t$  adalah waktu,  $f$  adalah frekuensi, dan  $e$  adalah konstanta *euler* dari sinyal (Kumar dkk, 2018).

$$S(f) = \int_{-\infty}^{\infty} s(t) e^{-j2\pi ft} dt \quad (2)$$

2.1.4 Bark Scale Filter Bank

Pada tahap ini dilakukan untuk mengetahui ukuran frekuensi pada *band* khusus di sinyal suara, biasanya *bark scale filter bank* digunakan pada sinyal dalam domain frekuensi. Terdapat dua proses yaitu masukan dengan nilai frekuensi (Hz) dan keluaran yang akan dihasilkan merupakan frekuensi dalam *bark*, sesuai yang ditunjukkan pada Persamaan (3) dan (4) (Putra, 2017).

$$f_{bark} = 6.0 * \sinh^{-1} \left( \frac{F(Hz)}{600} \right) \tag{3}$$

$$f_{bark} = 6.0 * \operatorname{arcsinh} \left( \frac{F(Hz)}{600} \right) \tag{4}$$

Selanjutnya, pada frekuensi yang sudah diubah bentuk *bark*, lalu dikembalikan kedalam bentuk frekuensi dalam Hz sesuai Persamaan (5) berikut.

$$F_{Hz} = 600 * \sinh \left( \frac{F(bark)}{6.0} \right) \tag{5}$$

2.1.5 Intensity Loudness Power Law

Pada tahap ini biasa digunakan untuk penerapan dari hukum *intensity power* atau biasa disebut kekuatan intensitas. Proses *intensity loudness power* sesuai dengan Persamaan (6), dimana *E* merupakan nilai array sesudah melakukan *intensity power* dan *w* merupakan nilai array dari sinyal informasi (Putra, 2017).

$$E = \frac{(w^2 + (56.8 * 10^6)) * w^4}{(w^2 + (6.3 * 10^6)) * (w^2 + (0.38 * 10^9)) * (w^6 + 9.58 * 10^26)} \tag{6}$$

2.1.6 Log

*Log* merupakan tahapan yang biasa digunakan untuk menangani terjadinya *zero handling* pada fitur yang diubah menjadi audio *log*, dan juga dapat memberikan limit di setiap frame yang didalam terdapat sinyal informasi untuk mengurangi terjadinya *error relative* saat ada pembulatan data dengan tipe float.

*Zero handling* merupakan cara untuk menangani hasil pembagian yang bernilai nol, dimana dalam elemen *probability* pada *log* dapat teratasi permasalahan tersebut.

2.1.7 Discrete Cosine Transform

*Discrete cosine transforms* (DCT) ini digunakan untuk mendapatkan kembali sinyal domain waktu berupa *cepstrum* yang bertujuan mengubah sinyal ke dalam komponen frekuensi dasar. Pada tahap ini menggunakan DCT III sesuai Persamaan (7), dimana *N* merupakan jumlah frame, *n* merupakan frame ke-*n*, dan *k* merupakan konstanta dengan interval  $0 > k > N - 1$  (Heredia, 2019).

$$X_k = \sum_{n=0}^{N-1} x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) \left( k + \frac{1}{2} \right) \right]; k = 0, \dots, N - 1 \tag{7}$$

2.1.8 Bark Frequency Cepstral Coefficients Filtering

*Bark frequency cepstral coefficients* merupakan hasil dari tahapan sebelumnya, BFCC ini nanti akan dijadikan sebagai fitur yang diklasifikasi menggunakan algoritma klasifikasi. Pada proses ini diperlukan proses filtering lagi untuk menghasilkan nilai cepstral yang lebih halus atau baik, Hal ini untuk memudahkan pengklasifikasian. Proses *bark frequency cepstral coefficients* untuk mendapatkan nilai koefisien *cepstral* sesuai Persamaan (8), dimana *w* (*n*) merupakan fungsi window terhadap fitur cepstral, *c* merupakan koefisien *cepstral*, dan *n* merupakan indeks dari koefisien *cepstral* (Putra, 2017).

$$w(n) = \left( 1 + \frac{c}{2} \sin \left( \frac{n\pi}{c} \right) \right), n = 1, 2, \dots, c \tag{8}$$

2.2 K-Nearest Neighbor (kNN)

Klasifikasi kNN merupakan suatu algoritma mesin pembelajaran dengan membandingkan sebuah data uji dengan data latih. Atribut dari data latih, dengan setiap data menunjukkan titik sebuah ruang berdimensi *n* dan semua data latih disimpan pada dimensi *n*. Jika ada sebuah data yang tidak dikenali kelasnya, maka kNN akan mencari pola ruang untuk data latih *k* yang terdekat dengan menghitung jarak *Euclidean* (*d<sub>st</sub>*) seperti yang ditunjukkan pada Persamaan (9), dimana *x* merupakan data latih, *y* merupakan data uji, dan *n* merupakan jumlah data (Taunk, 2019).

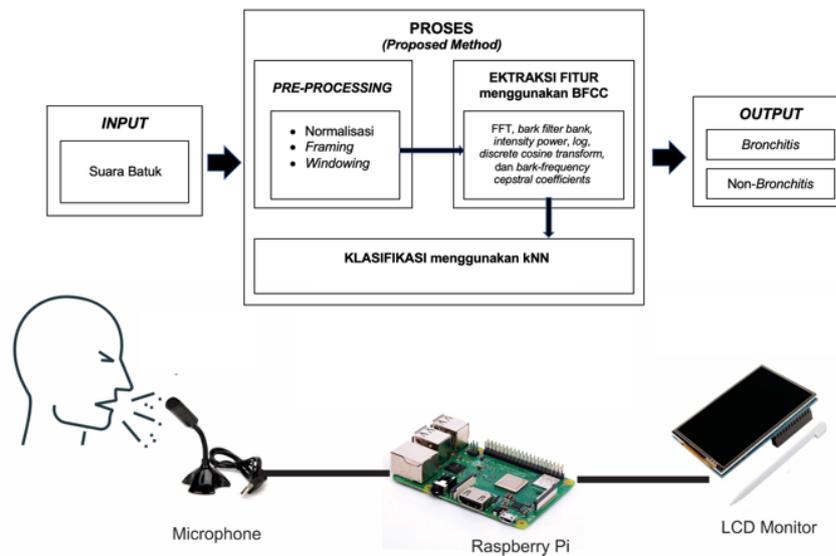
$$d_{st} = \sqrt{\sum_{j=1}^n (x - y)^2} \tag{9}$$

3. PERANCANGAN DAN IMPLENTASI SISTEM

	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP
1	fitur33	fitur34	fitur35	fitur36	fitur37	fitur38	fitur39	fitur40	klasifikasi	
2	-0,01054	-0,0073	-0,11741	0,027358	-0,04035	0,538173	-0,43406	2,561931	Bronchitis	
3	0,278455	0,04517	0,241195	0,40295	-0,3075	0,654381	-1,00433	2,335597	Bronchitis	
4	-0,13477	0,063417	-0,06056	-0,09692	-0,14596	0,312984	-0,53103	2,99794	Bronchitis	
5	-0,03114	0,211437	0,02977	0,391941	0,130819	0,061584	-0,00511	2,297754	Bronchitis	
6	-0,00664	-0,11028	-0,06359	0,256193	0,043345	0,663425	-0,24964	1,24197	Bronchitis	
7	-0,04778	0,115292	-0,0032	0,274784	-0,0102	0,328922	-0,23826	0,988807	Non-Bronchitis	
8	0,090258	0,03846	0,076297	0,401531	-0,23982	0,435804	-0,87587	1,789425	Non-Bronchitis	
9	-0,03141	0,000351	-0,01899	0,279961	-0,27534	0,457693	-0,97786	2,227928	Non-Bronchitis	
10	0,31985	-0,11983	-0,18892	0,261238	-0,28025	0,487998	-0,56589	1,325116	Non-Bronchitis	
11	-0,08075	0,173159	-0,09286	0,435237	-0,13903	0,517359	-0,45159	1,239181	Non-Bronchitis	

Gambar 3. Data fitur suara batuk

Sistem deteksi dini *bronchitis* yang di bangun pada penelitian ini berdasarkan pengenalan suara batuk.



Gambar 4. Gambaran umum sistem

Proses awal pengambilan suara batuk di ambil menggunakan mikrofon *Raspberry Pi*, lalu suara batuk akan di proses oleh *Raspberry Pi* dengan menggunakan ekstraksi fitur suara menggunakan teknik BFCC. Ekstraksi suara batuk menggunakan teknik BFCC menggunakan ukuran folter Bark dengan ukuran 40. Hasil ekstraksi fitur BFCC ditunjukkan pada Gambar 3.

Langkah selanjutnya adalah dilakukan pengkasifikasi dengan penggunaan algoritma *kNN*, setelah itu hasil yang di dapatkan *Raspberry Pi* akan ditampilkan pada layer LCD *Raspberry Pi* yang terdapat pilihan menu, seperti yang ditunjukkan pada Gambar 4.

Perangkat keras mikrofon dihubungkan dengan *Raspberry Pi* 3B+ melalui port USB pada *Raspberry*, sedangkan perangkat LCD *touchscreen* dihubungkan dengan *Raspberry Pi* melalui 13 Pin GPIO agar dapat menampilkan tampilan *Raspberry Pi*. Selanjutnya, *Raspberry Pi* perlu dihubungkan dengan adaptor USB *Micro* seperti yang ditunjukkan pada Gambar 5. Hasil implementasi dari perancangan keras tersebut ditunjukkan pada Gambar 6.



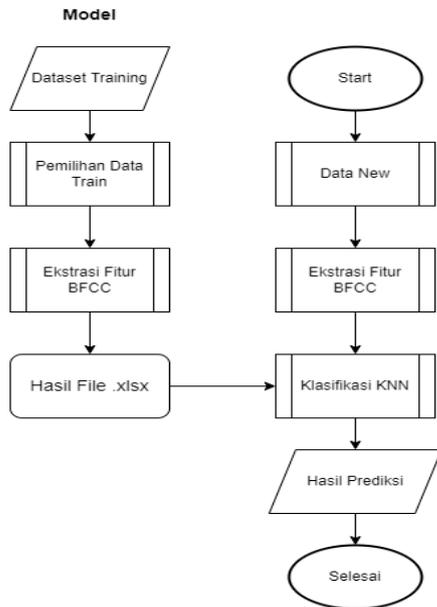
Gambar 6. Implementasi perangkat keras sistem



Gambar 5. Perancangan perangkat keras sistem

Perancangan perangkat lunak keseluruhan program utama meliputi semua proses program seperti program ekstraksi fitur BFCC, program *data new*, dan program klasifikasi *kNN*. Pada model terdapat pemilahan data *training* (latih) seperti yang digambarkan pada Gambar 7.

Terdapat beberapa proses pada keseluruhan program perancangan perangkat lunak seperti program *data new* dimana mengolah *input* suara batuk yang masuk dari mikrofon setelah itu suara batuk disimpan kedalam file dengan format *wav*. Kemudian, proses dilanjutkan pada program ekstraksi fitur dimana file *wav* akan diolah untuk menghasilkan nilai koefisien *cepstral* menggunakan algoritma BFCC. Pada model berisi dataset suara batuk yang nantinya akan masuk kedalam program anotasi dataset dimana dataset suara batuk tersebut akan dimasukkan kedalam dua kategori batuk *bronchitis* dan *non-bronchitis*, kemudian masuk kedalam program ekstraksi fitur dimana file *wav* akan diolah untuk menghasilkan nilai koefisien *cepstral* dari dataset. Jika sudah didapatkan hasil nilai koefisien *cepstral* dari program *data new*, proses klasifikasi menggunakan algoritma *kNN* dilakukan.



Gambar 7. Perancangan perangkat lunak sistem

#### 4. HASIL DAN PEMBAHASAN

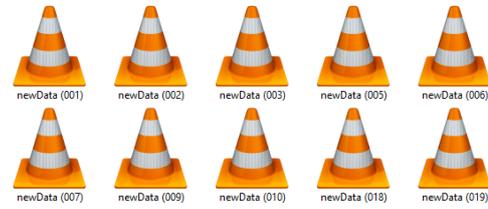
Hasil didapatkan dari pengujian yang telah dilakukan yaitu pengujian fungsional mikrofon terhadap suara batuk, yang digunakan untuk mengetahui apakah mikrofon dapat melakukan perekaman sesuai dengan fungsional mikrofon tersebut. Kemudian, pengujian waktu komputasi dan penggunaan CPU saat melakukan komputasi pendeteksian *bronchitis*. Selanjutnya, pengujian akurasi dilakukan untuk mencari nilai akurasi yang didapat dengan penggunaan kNN.

Hasil pengujian fungsional mikrofon terhadap suara batuk ditunjukkan pada Tabel 1. Hasil menunjukkan saat melakukan perekaman sebanyak sepuluh kali, mikrofon dapat berjalan dengan baik. Selanjutnya, dari sepuluh kali percobaan suara batuk terdeteksi tersebut berhasil masuk kedalam file *data new* seperti yang ditunjukkan pada Gambar 8. Hasil rata-rata keberhasilan dari percobaan mikrofon adalah 100 % dengan jenis suara yang berbeda. Sehingga, tidak terjadi kegagalan pada pengujian mikrofon terhadap suara batuk.

Tabel 1. Hasil pengujian fungsional mikrofon

Variabel Suara	Keterangan	
	Terdeteksi	Tidak Terdeteksi
Subjek1	✓	
Subjek2	✓	
Subjek3	✓	
Subjek4	✓	
Subjek5	✓	
Subjek6	✓	
Subjek7	✓	
Subjek8	✓	
Subjek9	✓	
Subjek10	✓	

Rata-rata keberhasilan =  $10/10 \times 100 = 100\%$



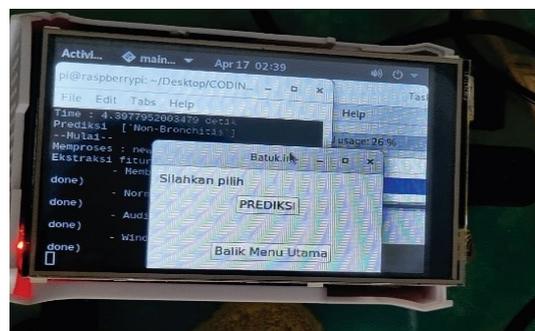
Gambar 8. Hasil deteksi suara batuk berhasil masuk kedalam file *data new*

Hasil pengujian waktu komputasi dan penggunaan CPU saat melakukan komputasi pendeteksian *bronchitis* ditunjukkan pada Tabel 2.

Tabel 2. Hasil pengujian waktu komputasi dan penggunaan CPU

Variabel Suara	Hasil Prediksi	Waktu Komputasi	Penggunaan CPU
Subjek1	Non-bronchitis	4,441s	26%
Subjek2	Non-bronchitis	4,382s	26%
Subjek3	Non-bronchitis	4,466s	28%
Subjek4	Non-bronchitis	4,321s	26%
Subjek5	Non-bronchitis	4,397s	26%
Subjek6	Non-bronchitis	4,327s	25%
Subjek7	Non-bronchitis	4,379s	26%
Subjek8	Non-bronchitis	4,365s	22%
Subjek9	Bronchitis	4,556s	26%
Subjek10	Non-bronchitis	4,500s	28%
Subjek11	Bronchitis	4,355s	25%
Subjek12	Bronchitis	4,448s	28%
Subjek13	Non-bronchitis	4,430s	28%
Subjek14	Non-bronchitis	4,455s	27%
Subjek15	Non-bronchitis	4,673s	27%
Subjek16	Bronchitis	4,461s	28%
Subjek17	Non-bronchitis	4,572s	26%
Subjek18	Bronchitis	4,451s	24%
Subjek19	Bronchitis	4,553s	27%
Subjek20	Non-bronchitis	4,518s	27%
<b>Rata-rata</b>		<b>4,452 s</b>	<b>26%</b>

Pengujian ini untuk mengetahui hasil dari sistem saat melakukan komputasi pendeteksian penyakit *bronchitis* dan berapa lama waktu komputasi yang digunakan dan penggunaan CPU raspberry. Tampilan hasil waktu komputasi dan penggunaan CPU ditunjukkan pada Gambar 9.



Gambar 9. Tampilan hasil waktu komputasi dan penggunaan CPU

Penggunaan CPU mendapatkan hasil yang cukup rendah sebesar 26%, dilihat dari waktu komputasi yang didapat cukup ekuivalen dengan

penggunaan metode BFCC yaitu sebesar 4.452 detik. Hal ini dikarenakan algoritma BFCC melakukan eksekusi secara bertahap atau serial.

Hasil pengujian akurasi dari penggunaan kNN dilakukan dengan data latih terhadap data uji seperti yang ditunjukkan pada Tabel 3. Data tersebut sudah disiapkan dan diketahui label asli, sehingga penulis ingin mencocokkan prediksi dengan label aslinya dengan menggunakan nilai neighbor = 5 diperhitungkan jarak *Euclidean*-nya. Nilai neighbor = 5 merupakan nilai yang paling cocok dan umum digunakan pada bagian implementasi dengan klasifikasi kNN untuk memprediksi kelas penyakit *bronchitis* dan *non-bronchitis*. Pengujian ini bertujuan untuk mencari tingkat keberhasilan prediksi kelas yang didapat.

Tabel 3. Hasil pengujian akurasi kNN

No	Data Latih	Data Uji	Hasil Klasifikasi		Akurasi
			Sesuai	Tidak Sesuai	
1	92	38	26	12	68%
2	108	38	27	11	71%
3	124	38	28	10	73%

Jumlah data yang di uji sebanyak 19 dengan masing masing label 2 ditotal sebanyak 38, Kemudian data uji tersebut akan dicocokkan dengan data latih untuk pengetesan prediksi akurasi yang didapat. Pengurangan data latih dilakukan karena ingin mengetahui peningkatan akurasi dimana tingkat akurasi ini dipengaruhi dari kuantitas dan semakin banyak data latih akurasinya akan meningkat.

## 5. KESIMPULAN

Implementasi algoritma BFCC dan kNN pada *embedded system* untuk deteksi dini *bronchitis* berdasarkan pengenalan suara telah berhasil dilakukan. Hasil pengujian terhadap perangkat Mikrofon USB untuk perekaman suara batuk dapat berfungsi 100% dari percobaan yang sudah dilakukan dan suara batuk berhasil disimpan pada mini komputer *Raspberry Pi*. Selanjutnya, hasil pengujian dari penggunaan CPU dengan BFCC pada mini komputer *Raspberry Pi* mendapatkan hasil *resource*-nya sebesar 26%, sedangkan pengujian waktu komputasi terhadap penggunaan ekstraksi BFCC memerlukan waktu 4,452 detik dari rata-rata keseluruhan pengujian. Kemudian, penggunaan algoritma BFCC dan KNN mendapatkan akurasi sebesar 73% dengan 124 data latih dan nilai jarak *Euclidean neighbors* = 5.

Keterbatasan pada penelitian ini adalah tidak adanya perangkat *audio speaker* untuk justifikasi saat pengenalan suara batuk masuk. Diharapkan kedepannya dapat membuat pemberitahuan justifikasi dengan dibantu perangkat *audio speaker* saat pengenalan suara batuk yang masuk pada mikrofon. Kemudian, penambahan data latih juga kedepannya akan dapat menghasilkan pengklasifikasian jauh lebih baik.

## DAFTAR PUSTAKA

- MAGHFIROH, DWIRAHAYU Y., dan MASHUDI S., 2021. Studi Literatur: Asuhan Keperawatan Pada Pasien Anak Dengan Bronchitis Dengan Masalah Keperawatan Bersihan Jalan Nafas Tidak Efektif. *Health Science Journal*, 5(1), pp. 35-43.
- WORLD HELATH ORGANIZATION, 2021. *Chronic obstructive pulmonary disease*. World Health Organization.
- DOTAN Y., JENNIFER Y., dan KIM V., 2019. *Chronic Bronchitis: Where Are We Now?*. *Chronic Obstructive Pulmonary Diseases*, 6(2), pp. 178-192.
- HIDAYATULLOH, A. 2018. Asuhan Keperawatan Pada Klien Bronchitis Kronik dengan Masalah Ketidakefektifan Bersihan Jalan Nafas di Ruang Teratai RSUD Bangil. Jombang: Insan Cendekia Medika.
- WIDYSANTO, A. dan MATHEW G., 2022. *Chronic Bronchitis*. National Center for Biotechnology Information: National Library of Medicine.
- SINGH, A. AVULA A. dan ZAHN E., 2022. *Acute Bronchitis*. National Center for Biotechnology Information: National Library of Medicine.
- BELKACEM, A. N., SOFIA O. A. L., ELHADJ BENKHELIFA, dan CHAO C., 2021. *End-to-End AI-Based Point-of-Care Diagnosis System for Classifying Respiratory Illnesses and Early Detection of COVID-19: A Theoretical Framework*. *Journal COVID-19: Integrating Artificial Intelligence, Data Science, Mathematics, Medicine and Public Health, Epidemiology, Neuroscience, and Biomedical Science in Pandemic Management*, 2021. arXiv: Electrical Engineering and System Science, Signal Processing Version. 1.
- KAMINSKA D., SAPINSKI T., dan ANBARJAFARRI G., 2017. *Efficiency of chosen speech descriptors in relation to emotion recognition*. *EURASIP Journal on Audio, Speech, and Music Processing*. No. 3.
- PUTRA K. T., 2017. *Voice Verification System Based on Bark Frequency Cepstral Coefficient*. *Journal of Electrical Technology UMY*. Vol. 1, No. 1.
- BAYRATNE, U., SWARNKAR, V., SETYAWATI, A., dan TRIASIH, R., 2013. *Cough Sound Analysis Can Rapidly Diagnose Childhood Pneumonia*. *Annals of Biomedical Engineering*.
- LARA, O., TEIJEIRO, T., DAN ATIENZA, D., 2020. *The CHOUGHVID crowdsourcing dataset: A Corpus for the study of large-scale cough analysis algorithms*. *Scientific*

*Data*, 8:156. Tersedia di:  
<https://doi.org/10.1038/s41597-021-00937-4>

- KUMAR C., REHMAN F. U., KUMAR S., MEHMOOD A., dan SHABIR G., 2018. *Analysis of MFCC and BFCC in a speaker identification system. International Conference on Computing, Mathematics and Engineering Technologies.*
- ROLON-HEREDIA J.A., GARRIDO-AREVALO V.M., and MARULANDA J., 2019. Voice compression using discrete cosine transform and wavelet transform. *Journal of Physics*, 1403.
- TAUNK K., DE S., VERMA S., 2019. *A Brief Review of Nearest Neighbor Algorithm for Learning and Classification.* International Conference on Intelligent Computing and Control Systems (ICCS), May 2019.