

PERANCANGAN SISTEM MONITORING KERUSAKAN MESIN DENGAN PEMISAHAN BANYAK SUMBER SUARA MENGGUNAKAN INDEPENDENT COMPONENT ANALYSIS (ICA)

Design of Machines Fault Monitoring With Sound Separation Using Independent Component Analysis (ICA)

¹Devi Anggraini

²Achmad Rizal, ST., MT.

³Suryo Adhi Wibowo, ST., MT.

^{1,2,3} Fakultas Teknik Elektro – Universitas Telkom

Jl. Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia

¹ devz.dep@gmail.com

² achmadrizal@telkomuniversity.ac.id

³ suryoadhiwibowo@telkomuniversity.ac.id

ABSTRAK

Dalam proses produksi di industri, pemeliharaan mesin-mesin produksi adalah hal yang sangat penting. Pemeliharaan kondisi mesin di industri membutuhkan kecepatan dan kemudahan, salah satu metodenya adalah dengan analisis getaran (analisis vibrasi). Getaran mesin menyebabkan pola suara yang diemisikan mesin, dimana suara mesin yang satu bercampur dengan suara mesin yang lainnya. Perubahan getaran mesin akan mengakibatkan perubahan pola suara yang diemisikan mesin.

Pada Tugas Akhir ini dilakukan perancangan sistem monitoring dan pendeteksian kerusakan mesin dengan pemisahan sinyal suara yang berasal dari banyak sumber suara pada mesin. Beberapa suara motor direkam secara bersamaan melalui *microphone array* (beberapa mikrofon yang disusun). Setiap mikrofon menerima sinyal suara dari beberapa motor tersebut, sehingga sinyal output dari tiap mikrofon merupakan sinyal campuran. Pemisahan sinyal campuran tersebut dilakukan dengan menggunakan *Blind Source Separation* (BSS) dengan metode *Independent Component Analysis* (ICA) dan algoritma *FastICA*. Pada penelitian sebelumnya, digunakan algoritma *natural gradient*, tetapi dalam penerapannya masih memiliki beberapa kelemahan.

Untuk memisahkan sinyal campuran digunakan metode ICA dengan pendekatan dari domain waktu (*Time Domain ICA-TDICA*), pendekatan domain frekuensi (*Frequency Domain-FDICA*), gabungan keduanya secara bertahap (*Multistage ICA-MSICA*), serta *FastICA*. Hasil dalam Tugas Akhir ini yaitu dengan menggunakan algoritma *FastICA* didapatkan performansi sistem yang lebih baik ditandai dengan nilai parameter performansi SNR yang tinggi.

Kata kunci: ICA, *FastICA*, sinyal suara, deteksi kerusakan.

ABSTRACT

In the process of industrial production, the maintenance of production machinery is very important. Maintenance of machinery conditions in industry requires speed and ease, one of the methods is the analysis of vibration. Machine vibration causes the machine emitted sound pattern, wherein the engine sound mixes with the sound of the other machines. The change of machine vibration will cause the change of emitted sound machine pattern.

In this final project, a system for monitoring and detection of fault of the machines is designed with the separation of sound signals from many sources on the machines. Some motor sounds are recorded simultaneously through microphone array (multiple microphones arranged). Each microphone receives sound signals from some of the motors, so that the output of each microphone signal is mixed signal. Separation of mixed signals is done by using Blind Source Separation (BSS) with Independent Component Analysis (ICA) method and FastICA algorithm. In previous study, using the natural gradient algorithm, but the application still has some drawbacks.

To separate mixed signals, using ICA method with the approach of Time Domain ICA (TDICA), the approach of Frequency Domain ICA (FDICA), combination of both gradually (Multistage ICA-MSICA), and FastICA. The result in this final project by using FastICA algorithm obtained a better system performance, characterized by the high SNR value.

Keywords: ICA, *FastICA*, sound signal, fault detection.

1. Pendahuluan

Monitoring dan pendeteksian kerusakan pada mesin bermotor di pabrik dengan menggunakan analisis getaran (vibrasi), merupakan metode yang saat ini biasa dipakai oleh kalangan industri. Analisis getaran ini menganalisis pola getaran berdasarkan beberapa parameter, yaitu amplitudo, frekuensi, dan fasa. Jika timbul perubahan getaran, maka suara yang diemisikan mesin ikut berubah dan ini bisa menjadi indikasi

terjadinya kerusakan pada mesin, karena sinyal suara juga memiliki parameter yang sama seperti sinyal getaran. Menggunakan analisis sinyal suara, memiliki beberapa keunggulan daripada analisis getaran. Diantaranya, mempermudah pengukuran karena tidak harus menempelkan langsung sensor getaran (*accelerometer*) ke badan mesin yang mempunyai suhu dan getaran yang tinggi. Selain itu, efektif dan efisien karena bisa memantau

kondisi mesin secara kontinu dan terus-menerus, sedang pada analisis getaran hanya dilakukan pengukuran setiap periode tertentu, sehingga bila ada kerusakan di luar periode, analisis getaran tidak bisa mendeteksi kerusakan mesin.

Blind Source Separation (BSS) adalah teknik memisahkan sinyal yang lazim dipakai pada telekomunikasi nirkabel, untuk memisahkan antara pengguna satu dengan yang lain agar pesan tidak salah alamat. Secara garis besar, BSS merupakan teknik untuk memisahkan penggabungan sinyal output tanpa mengetahui karakteristik dan jumlah sumbernya. Berdasarkan statistik sinyal masukan (independensi) maka dapat ditentukan komponen-komponen sinyalnya. Metode analisis komponen independen (*Independent Component Analysis* – ICA) sering digunakan untuk menyelesaikan permasalahan BSS.

Pada Tugas Akhir ini, dirancang sebuah sistem monitoring dan pendeteksian kerusakan mesin yang berbasis pemisahan sinyal suara yang berasal dari banyak sumber suara pada mesin dengan menggunakan *microphone array* sebagai perekamnya. Pemisahan sinyal campuran tersebut dilakukan dengan menggunakan *Blind Source Separation* (BSS) dengan metode *Independent Component Analysis* (ICA). Pada penelitian sebelumnya, digunakan algoritma *natural gradient* dalam pemisahan sinyal suara tercampur, tetapi masih memiliki beberapa kelemahan sehingga di sini akan digunakan algoritma *FastICA* karena lebih cepat konvergen.

2. Dasar Teori

2.1 *Independent Component Analysis* (ICA)

2.1.1 Teori ICA^[4]

Prinsip dari metode ICA adalah menghitung secara implisit matriks campuran dari sinyal sehingga dapat ditemukan kembali estimasi sinyal yang diinginkan. ICA merupakan suatu teknik perhitungan statistik untuk menemukan faktor-faktor tersembunyi yang mendasari sekumpulan variabel random, pengukuran, atau sinyal-sinyal. Salah satu aplikasi metode ICA yaitu digunakan untuk memisahkan sinyal-sinyal tercampur yang berasal dari sumber yang saling bebas statistik. Asumsi pengamatan dilakukan pada n linier pencampuran x_1, x_2, \dots, x_n dari n komponen bebas.

Bila ada sumber-sumber yang saling bebas satu sama lain (*independent*) tercampur, maka komponen terukur sensor dalam domain waktu dapat dimodelkan sebagai:

$$x_j(t) = a_{j1}s_1(t) + a_{j2}s_2(t) + \dots + a_{jn}s_n(t) \quad (2.1)$$

Persamaan di atas apabila ditulis dalam bentuk vektor-matriks maka menjadi:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \quad (2.2)$$

atau

$$\mathbf{x} = \sum_{i=1}^n \mathbf{a}_i s_i \quad (2.3)$$

Persamaan (2.2) di atas dikenal sebagai model dari *Independent Component Analysis* (ICA). Model tersebut bersifat *general*. Komponen independen \mathbf{s} , merupakan komponen laten, dimana komponen tersebut tidak dapat langsung diamati. Matriks pencampur \mathbf{A} , juga tidak diketahui, hanya sinyal yang terukur saja, \mathbf{x} , yang diketahui. Komponen \mathbf{A} dan \mathbf{s} diestimasi dari sinyal terukur \mathbf{x} tersebut.

Tujuan dari metode ICA adalah mencari sinyal estimasi \mathbf{y} . Bila persamaan (2.3) di atas ditulis dalam bentuk matriks dua dimensi, maka:

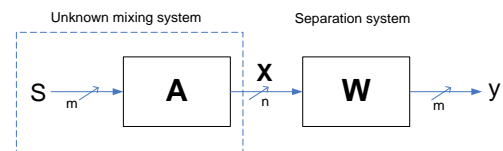
$$\begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} * \begin{bmatrix} \bar{s}_1 \\ \bar{s}_2 \end{bmatrix} \text{ dan } \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{bmatrix} * \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} \quad (2.4)$$

Sehingga sinyal estimasi dapat dinyatakan:

$$\mathbf{y} = \mathbf{W}\mathbf{x} \quad (2.5)$$

dimana $\mathbf{W} = \mathbf{A}^{-1}$.

Permasalahan utama metode ICA adalah mencari model filter linear \mathbf{W} , dimana \mathbf{W} merupakan invers dari matriks pencampur \mathbf{A} . Ketepatan pemilihan model filter ini akan mempengaruhi kualitas pemisahan sinyal. Beberapa metode optimasi untuk mencari nilai \mathbf{W} ini akan dijelaskan kemudian.



Gambar 2.1 Blok Diagram Model ICA

2.1.2 Ambiguitas ICA^[6]

Dalam Algoritma ICA terdapat beberapa hal bersifat ambigu yang muncul, yaitu:

1. Variansi dan energi tidak dapat ditentukan

Setiap sinyal memiliki suatu nilai variansi tertentu yang menyatakan tingkat energi sinyal tersebut, begitu juga dengan sinyal asli pembentuk *mixed signal*. ICA memungkinkan proses pemisahan sejumlah komponen bebas dari suatu sinyal tercampur, namun ICA tidak dapat menentukan nilai variansi sinyal asli. Hal ini disebabkan karena pada saat perekaman data akan terdapat faktor bebas skalar yang menyebabkan pelemahan atau penguatan sinyal hasil rekaman, sehingga nilai variansi sinyal asli tidak dapat ditentukan melalui proses ICA.

- Urutan *Independent Component* tidak dapat ditentukan

Komponen bebas yang dihasilkan melalui kalkulasi ICA dapat berubah-ubah urutannya, hal ini jelas ketika proses simulasi ICA dilakukan.

2.1.3 Optimasi ICA^{[6][7]}

Ada beberapa algoritma ICA yang digunakan. Masing-masing algoritma memiliki kelebihan satu sama lain, dan dari tahun ke tahun algoritma ini terus disempurnakan. Beberapa algoritma untuk optimasi ICA adalah sebagai berikut:

- Infomax*

Algoritma *infomax* mendekati model ICA berdasarkan fungsi *non-linear*.

- Natural Gradient*

Amari, Chicocki, dan Yang (1996) mengusulkan untuk memodifikasi besar pembobotan filter pemisah pada algoritma *Infomax*. Algoritma ini lebih cepat konvergen daripada *infomax*.

- FastICA*

FastICA merupakan suatu algoritma yang ditujukan untuk melakukan proses ekstraksi berdasarkan algoritma ICA. Algoritma *FastICA* ini dikenal lebih cepat konvergen dari algoritma-algoritma sebelumnya. Algoritma *FastICA* memodelkan ICA dengan pendekatan *non-gaussianity* yang didasarkan pada *negentropy* dengan iterasi titik tetap (*fixed-point*). Algoritma ini dapat pula didekati dengan iterasi Newton yang menghasilkan persamaan yang sama pula. Aturan pembobotan filter pemisah pada algoritma *FastICA* ditentukan dengan:

$$\begin{aligned} \mathbf{w}^+ &= \mathbf{w} - \frac{E \{ \mathbf{x}g(\mathbf{w}^T \mathbf{x}) \} - \beta \mathbf{w}}{E \{ g'(\mathbf{w}^T \mathbf{x}) \} - \beta} \\ &= E \{ \mathbf{x}g(\mathbf{w}^T \mathbf{x}) \} - E \{ g'(\mathbf{w}^T \mathbf{x}) \} \mathbf{w} \quad (2.6) \end{aligned}$$

dimana fungsi *g* merupakan turunan dari fungsi kontras.

Proses *FastICA*^[5] terbagi menjadi dua, yaitu *pre-processing* dan *processing*.

- Pre-processing FastICA*

- Remmean*

Remmean atau *centering* adalah tahapan pre-proses yang dilakukan untuk membuang *mean/removing mean*, $m=E$. Pada tahapan ini, data input diolah sehingga didapatkan matriks rata-rata/*mean*. Matriks pusat inilah yang kemudian akan diolah ke tahapan selanjutnya, sedangkan matriks rata-rata akan ditambahkan saat algoritma ICA telah selesai dilakukan.

- Whitenv*

Setelah proses *remmean*, tahapan pre-proses selanjutnya adalah *whitenv*. Pada tahapan ini data ditransformasikan

sehingga didapatkan vektor data baru dengan karakteristik:

- antara baris yang satu dan baris yang lainnya tidak saling berkorelasi
- memiliki variasi yang sama
- kovariansi matriks = matriks identitas [I]

- PCAmAt*

PCAmAt adalah proses yang ditujukan untuk mengurangi/meminimalkan jumlah *redundancy*/pengulangan yang terjadi di dalam data yang diobservasi. Sehingga data yang diolah benar-benar data yang tidak mengandung pengulangan. Pembuangan data redundan oleh *PCAmAt* hanya didasarkan pada sifat korelasi antara elemen-elemen yang terdapat di dalam data tersebut.

- Processing FastICA*

Pada ICA, setiap sinyal yang ditangkap/direkam merupakan hasil dari suatu fungsi linier. Dengan menganggap bahwa S_i bebas statistika, maka invers dari persamaan $\mathbf{x}=\mathbf{A}\mathbf{s}$ dapat ditulis:

$$\mathbf{S}=\mathbf{W}\mathbf{x} \quad (2.7)$$

dimana \mathbf{W} adalah matriks invers dari matriks pencampur \mathbf{A} . Jika y adalah salah satu nilai komponen bebas dari *mixed signal*, atau $y = \mathbf{W}^T \mathbf{x}$ dan w_i merupakan salah satu komponen baris dari matriks invers \mathbf{A} , maka y adalah salah satu komponen bebas dari \mathbf{X} . Permasalahan yang dihadapi sekarang adalah bagaimana cara untuk mendapatkan w_i yang memiliki nilai yang sama dengan salah satu baris dari vektor \mathbf{A} . Untuk melihat konsep dasar ICA, persamaan $y = \mathbf{W}^T \mathbf{x} = \mathbf{A}^T \mathbf{x}$, sehingga didapatkan $y = \mathbf{W}^T \mathbf{x} = \mathbf{W}^T \mathbf{A} \mathbf{s} = \mathbf{z}^T \mathbf{s}$. Terlihat bahwa y merupakan kombinasi linear dari s dengan *weighted factor*. Karena jumlah dari dua komponen bebas atau lebih memiliki sifat *gaussian* yang lebih besar dari sinyal asli s , maka $\mathbf{z}^T \mathbf{s}$ lebih *gaussian* dari semua komponen s yang ada, dan nilai *gaussian* $\mathbf{z}^T \mathbf{s}$ akan semakin kecil dan mendekati nilai s jika $\mathbf{z}^T \mathbf{s}$ mendekati atau sama dengan salah satu variabel bebas s . Berdasarkan hal ini, nilai dari vektor \mathbf{W} dapat dicari dengan cara memaksimalkan *gaussianity* nilai $\mathbf{W}^T \mathbf{x}$.

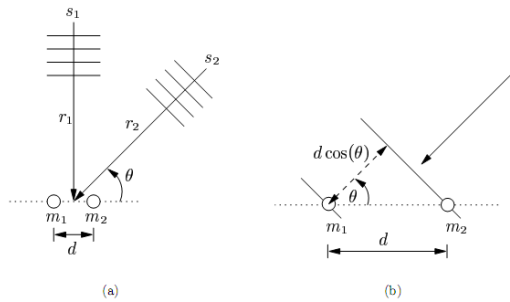
2.2 Microphone Array^{[1][5]}

Mikrofon merupakan transduser yang mengubah sinyal suara menjadi sinyal listrik. Pembagian jenis mikrofon dapat digolongkan

menjadi beberapa macam berdasarkan bahannya, keterarahannya, dan fungsinya.

Microphone array merupakan sekumpulan mikrofon yang terhubung menjadi satu kesatuan untuk menerima serta mentransmisikan sinyal suara. Dengan menggunakan susunan mikrofon, maka dapat dicapai seleksi spasial, memperkuat propagasi sumber dari arah tertentu dan melemahkan propagasi sumber dari arah lainnya.

Konsep pemrosesan *microphone array* dapat diilustrasikan seperti Gambar 2.3. Misalkan dua sumber bunyi s_1 dan s_2 diletakkan pada jarak r_1 dan r_2 dari sepasang mikrofon (m_1 dan m_2). Jarak antara s_1 dan s_2 sama panjang terhadap kedua mikrofon (“on-axis”), sedang s_2 lebih dekat ke mikrofon m_2 daripada mikrofon m_1 (“off-axis”). Karena jarak s_1 terhadap dua mikrofon adalah sama maka akan diterima oleh kedua mikrofon pada fasa yang sama dan kombinasi antar keduanya akan memperkuat sinyal dua kali. Sedangkan jarak antar sumber s_2 dengan kedua mikrofon berbeda sehingga terjadi perbedaan fasa antar keduanya dan kombinasi antar kedua mikrofon akan saling melemahkan. Perbedaan ini bergantung pada sudut datang (θ) dan jarak antar mikrofon (d).



Gambar 2.2 Konsep *microphone array*

Susunan sejumlah mikrofon pada dasarnya mencuplik medan suara pada titik yang berbeda pada suatu ruang. Sebagai akibatnya, pada pemrosesan *array* akan terjadi masalah *spatial aliasing* jika sinyal yang dicuplik sangat lambat. Ketika *spatial aliasing* terjadi maka *array* tidak bisa membedakan di antara sudut datang untuk frekuensi yang ingin ditangkap dan arah yang tidak diinginkan. Untuk menghindari *spatial aliasing*, maka digunakan aturan:

$$d < \frac{\lambda_{\min}}{2} \quad (2.8)$$

dimana λ_{\min} adalah panjang gelombang minimum dari obyek sumber suara (sebanding dengan frekuensi maksimum).

2.3 Frekuensi Sesaat^[2]

Suatu sinyal dapat ditampilkan dalam domain waktu dan domain frekuensi. Pada kedua domain inilah (waktu-frekuensi) biasanya analisa sinyal dilakukan. Pada domain waktu suatu sinyal dapat dituliskan sebagai:

$$x(t) = a \cos \omega_0 t \quad (2.9)$$

dimana,

$$\omega_0 = 2\pi f_i \quad (2.10)$$

Maka didapat:

$$x(t) = a \cos \left(\int_0^t 2\pi f_i(t) dt + \theta \right) \quad (2.11)$$

Frekuensi sesaat dapat diturunkan dari *Short Time Fourier Transform (STFT)*. STFT merupakan modifikasi dari Transformasi *Fourier*, yakni teknik memecah suatu sinyal menjadi komponen-komponen sinusoid-nya dalam domain frekuensi. STFT digunakan untuk menganalisa sinyal yang bersifat non-stasioner, membagi menjadi bagian-bagian kecil yang diasumsikan stasioner. Pembagian ini dilakukan dengan fungsi *window* dengan lebar tertentu yang kemudian digeser dan dikalikan dengan sinyal asli untuk memperoleh sinyal stasioner yang sempit. Setelah sinyal-sinyal tersebut dibagi menjadi sinyal-sinyal stasioner yang sempit, maka sinyal tersebut ditransformasikan dengan Transformasi *Fourier* untuk memperoleh STFT.

Frekuensi sesaat merupakan frekuensi kurva sinus yang sesuai dengan sinyal pengukuran. Nilai frekuensi ini bukan berarti frekuensi dalam *cycle* per satuan waktu, karena dalam frekuensi sesaat ada nilai negatifnya ($-\infty, \infty$). Karena sifat ketidak-elastisan dan tipe kerusakan yang lain dapat dianggap berpengaruh terhadap komposisi frekuensi dari respon, maka frekuensi sesaat ini diharapkan sangat berguna dalam karakterisasi kerusakan.

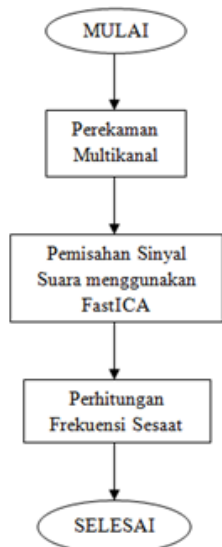
3. Perancangan dan Realisasi Sistem

3.1 Perancangan Sistem

Sistem monitoring dan deteksi kerusakan mesin yang berbasis pemisahan sinyal suara ini dilakukan dalam beberapa tahap, dimulai dari proses perekaman multikanal untuk memperoleh data sinyal suara tercampur. Lalu tahap-tahap selanjutnya adalah pemisahan sinyal suara tercampur, perhitungan frekuensi sesaat untuk mendeteksi kerusakan mesin berdasarkan analisis pola suaranya.



Gambar 3.1 Skema diagram perancangan system

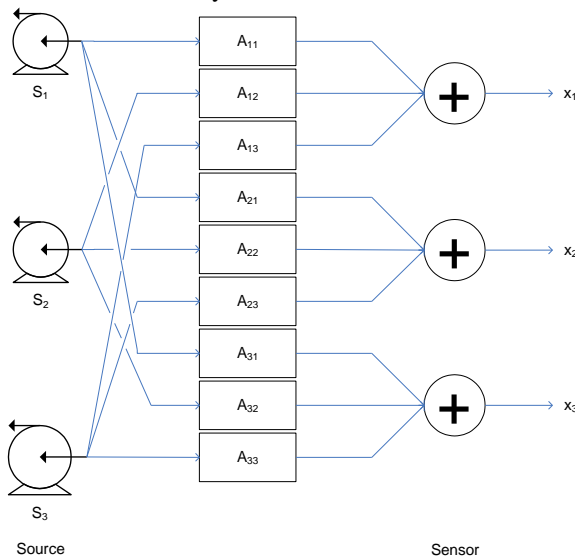


Gambar 3.2 Flowchart perancangan sistem

3.2 Perekaman Multikanal

Perekaman sinyal suara mesin bertujuan untuk memperoleh data suara yang diemisikan motor mesin, dimana nantinya data ini yang akan diolah.

Proses perekaman dilakukan pada tiap kanal. Jumlah sensor yang digunakan harus memenuhi kriteria *microphone array* dimana $m \geq n$ (m = jumlah mikrofon/sensor; n = jumlah motor/sumber). Di sini digunakan dua sensor-dua sumber, tiga sensor-tiga sumber, dan empat sensor-tiga sumber. Pada tiap-tiap kanal dilakukan proses perekaman secara simultan/bersamaan dan tiap *file* disimpan dengan ekstensi '.wav'. Mikrofon diletakkan secara spesifik terhadap kondisi salah satu motor, sehingga sinyal yang diterima mikrofon akan lebih kuat pada motor yang berada di dekatnya namun tercampur dengan sinyal emisi dari motor lainnya.



Gambar 3.3 Blok diagram perekaman multikanal (contoh: 3 sumber 3 sensor)

Keterangan Gambar 3.3:

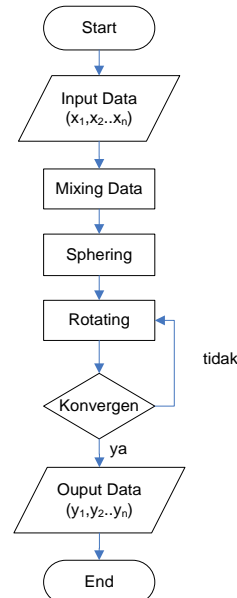
S_1, S_2, S_3 = motor/sumber suara

X_1, X_2, X_3 = mikrofon/sensor

A_{ij} = sinyal suara dari sumber S_i yang diterima oleh mikrofon X_j (contoh: A_{11} = sinyal suara dari sumber S_1 yang diterima oleh mikrofon X_1)

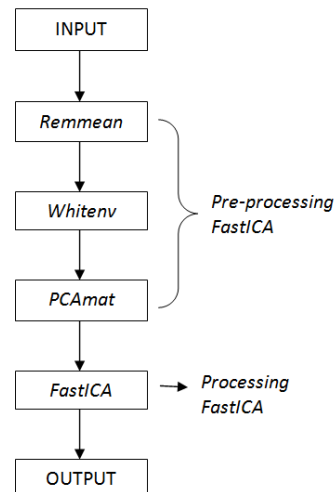
3.3 Pemisahan Sinyal Suara

Pada proses pemisahan sinyal suara, digunakan metode *Independent Component Analysis* (ICA). Proses ICA diawali dengan menginputkan sinyal campuran dari tiap-tiap mikrofon. Sinyal yang sudah dalam bentuk matriks ini kemudian dicampur (*mixing*) satu sama lain untuk menghasilkan matriks campuran dengan orde gabungan dari sinyal input. Matriks campuran akan di-*sphere* untuk mencari arah putar yang tepat. Setelah melalui proses *sphering* matriks akan diputar (*rotating*) untuk mencapai konvergensi melalui beberapa kali iterasi. Diagram alir ICA ditunjukkan oleh Gambar 3.4.



Gambar 3.4 Flowchart ICA

Untuk melakukan proses ekstraksi berdasarkan algoritma ICA, digunakan algoritma *FastICA*.



Gambar 3.5 Diagram alur proses *FastICA*

Tahap awal yaitu *pre-processing*, bertujuan untuk meningkatkan kemampuan ICA dalam mengestimasi matriks dekomposisi agar kualitas output yang akan dihasilkan algoritma ICA dapat ditingkatkan. Tahap ini terdiri atas 3 bagian, yaitu *remmean*, *whitenv*, dan *PCAmat*.

Pada tahap *remmean* (*centering*), data input diolah sehingga didapatkan matriks rata-rata/*mean*. Matriks pusat inilah yang kemudian akan diolah ke tahapan selanjutnya, sedangkan matriks rata-rata akan ditambahkan saat algoritma ICA telah selesai dilakukan.

Setelah proses *remmean*, tahapan selanjutnya adalah *whitenv*. Pada tahapan ini data ditransformasikan sehingga didapatkan vektor data baru dengan beberapa karakteristik yaitu antara baris yang satu dan baris yang lainnya tidak saling berkorelasi, memiliki variasi yang sama, dan kovariansi matriks = matriks identitas [I].

Tahap selanjutnya adalah *PCAmat* yaitu proses yang ditujukan untuk mengurangi/meminimalkan jumlah redundancy/pengulangan yang terjadi di dalam data yang diobservasi sehingga data yang diolah benar-benar tidak mengandung pengulangan.

Setelah tahap *pre-processing*, maka dilanjutkan ke tahap *processing* yaitu proses ICA itu sendiri. Pada ICA, ekstraksi sinyal tidak hanya dilakukan berdasarkan nilai korelasinya, namun juga faktor-faktor lainnya. Data diasumsikan memiliki sifat *non-gaussian* dan tanpa asumsi ini, proses ICA tidak mungkin dilakukan.

3.4 Perhitungan Frekuensi Sesaat

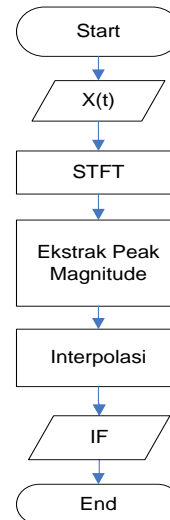
Setelah dilakukan pemisahan sinyal, maka sinyal estimasi yang diperoleh digunakan untuk mendeteksi kerusakan mesin dengan metode frekuensi sesaat yang dapat ditelusuri dari *spectrogram* (*time-frequency representation*) sinyal. *Spectrogram* menunjukkan besarnya nilai amplitudo untuk setiap frekuensi sinyal pada tiap waktu pengukuran.

Perhitungan frekuensi sesaat menggunakan perhitungan *Short Time Fourier Transform* (STFT) yaitu dengan cara membagi sinyal $x(t)$ menjadi beberapa segmen. Frekuensi sesaat ini diperoleh dengan menurunkan fasa dari sinyal terhadap waktu. Untuk mengetahui fasa dari suatu sinyal, maka perlu diketahui bentuk sinyal analisisnya. Turunan dari fasa ini dapat ditentukan dengan persamaan:

$$H[s(t)] = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{s(t)}{t - \tau} dt \quad (3.1)$$

Frekuensi sesaat yang ditelusuri adalah frekuensi sesaat yang memiliki amplitudo terbesar, sehingga diperoleh frekuensi sesaat yang mewakili frekuensi dominan (frekuensi fundamental) dari sinyal. Hasilnya berupa *spectrogram* dengan garis-garis tunggal yang menunjukkan frekuensi dengan amplitudo yang lebih besar dari *threshold* yang telah ditentukan.

Selanjutnya dilakukan interpolasi hasil penelusuran agar diperoleh hasil frekuensi sesaat yang lebih baik, sehingga frekuensi fundamentalnya akan terlihat lebih jelas.



Gambar 3.6 Flowchart perhitungan frekuensi sesaat

3.5 Parameter Performansi

Untuk mengukur tingkat performansi sistem, tentunya diperlukan perhitungan terhadap parameter yang bisa menunjukkan performansi yang dicapai oleh sistem. Di sini dipakai parameter SNR atau Signal-to-Noise Ratio. SNR adalah perbandingan antara daya sinyal yang diinginkan terhadap daya noise yang diterima pada suatu titik pengukuran. SNR digunakan untuk menunjukkan tingkat kualitas sinyal penerimaan, dimana semakin besar harga SNR maka kualitas akan semakin baik. Satuan SNR adalah dB.

$$SNR = \frac{\text{Daya sinyal informasi}}{\text{Daya sinyal noise}} \quad (3.2)$$

4. Analisis Dan Pengujian

4.1 Pengujian Sistem

Data pengujian terdiri dari 4 jenis data rekaman yaitu:

1. Bearing-Normal-Misalignment
2. Bearing-Unbalance-Misalignment
3. Bearing-Unbalance-Normal
4. Misalignment-Unbalance-Normal

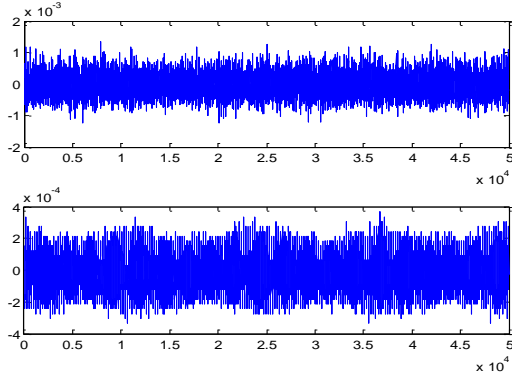
Masing-masing data terdiri dari 3 file rekaman yang merepresentasikan 3 mic saat perekaman.

4.2 Pemisahan Sinyal Suara

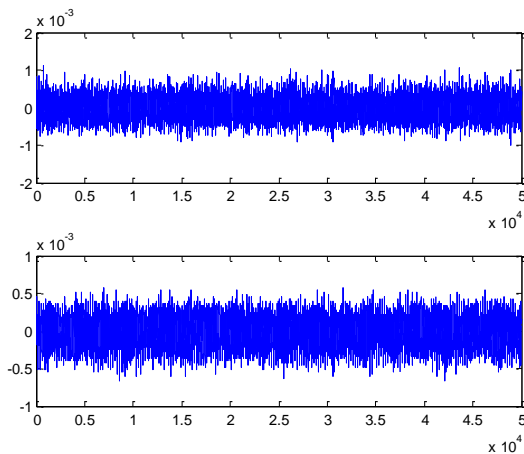
Untuk memisahkan sinyal campuran ini, digunakan metode ICA dengan pendekatan dari domain waktu (*Time Domain ICA-TDICA*), pendekatan domain frekuensi (*FDICA*), gabungan keduanya secara bertahap (*Multistage ICA-MSICA*), serta Fast ICA. Hasil pemisahan sinyal suara mesin dengan beberapa metode ICA dapat dijabarkan pada sub-bab berikut.

4.2.1 Time Domain ICA (TDICA)

Analisa dasar suatu sinyal dapat dilakukan dalam domain waktu (*time-series*). Pada domain ini ditampilkan informasi besarnya amplitudo sinyal tiap waktu sampel. *Time domain ICA* (TDICA) memisahkan sinyal dalam domain waktu. Pada percobaan pertama dengan dua mikrofon-dua motor dikondisikan motor dalam keadaan *bearing fault* dan *misalignmnet*. Berdasarkan perekaman *baseline* penelitian dengan *single mic* diperoleh sinyal asli dan campuran dari kondisi *bearing fault* dan *unbalance* dalam domain waktu sebagai berikut:

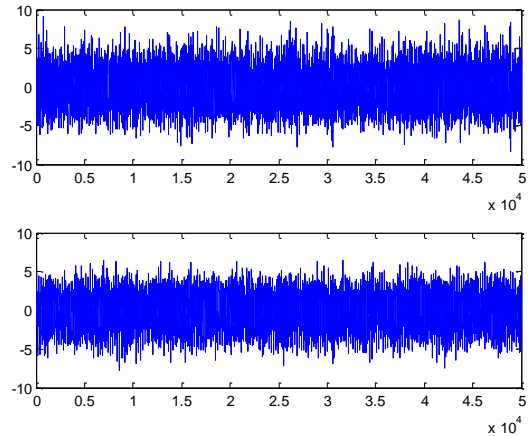


Gambar 4.1 Sinyal Asli Kondisi Bearing Fault dan Misalignment



Gambar 4.2 Sinyal Campuran tiap mikrofon

Hasil pemisahan sinyal dengan TDICA adalah sebagai berikut:



Gambar 4.3 Sinyal Estimasi TDICA

Untuk mengidentifikasi apakah sinyal tersebut adalah sinyal dari *bearing fault* atau *misalignment* sulit dianalisa dari domain waktu. Pola frekuensi sesaat digunakan untuk identifikasi sinyal dan menghasilkan diagram seperti pada Tabel 4.5. Dari pola frekuensi sesaat terlihat bahwa sinyal bagian atas adalah sinyal dari *bearing fault* dan sinyal bagian bawah merupakan sinyal kondisi *misalignment*. Kuantitas hasil pemisahan sinyal suara dalam domain waktu (TDICA) ini dapat ditunjukkan pada tabel di bawah.

Tabel 4.1 Kondisi dan Kualitas TDICA

Parameter	Nilai
Besar Langkah	0.000001
Blok Proses	30
Jumlah Iterasi	100
Angle	0.2
SNR	-4.6321

4.2.2 Frequency Domain ICA (FDICA)

Analisa dalam domain waktu terkadang kurang bisa memberikan informasi yang terkandung dalam suatu sinyal. Begitu juga dalam pemisahan sinyal dengan metode ICA. Maka, pemisahan dalam domain frekuensi menjadi alternatif, dan secara matematik lebih mudah karena hanya melibatkan perkalian *dot product*. Untuk merubah sinyal dalam domain waktu menjadi domain frekuensi, sinyal ditransformasikan dengan *fourier transform* dan secara komputasi dilakukan dengan FFT (*fast fourier transform*). Hasil dari FDICA berupa sinyal estimasi dalam domain frekuensi, sehingga perlu ditransformasi balik dalam domain waktu dengan IFFT (*Inverse FFT*). Sinyal estimasi ini dicari pola frekuensi sesaatnya untuk identifikasi sinyal.

Tabel 4.2 Kondisi dan Kualitas FDICA

Parameter	Nilai
Besar Langkah	0.000001
Blok Proses	30
Jumlah Iterasi	100
Angle	1,4
SNR (dB)	11.9272

4.2.3 Time-Frequency ICA (MSICA 1)

Pemisahan sinyal suara mesin dengan metode ICA pada kedua domain waktu dan frekuensi telah berhasil dilakukan dan diketahui hasil pemisahannya. Pengembangan pendekatan tersebut adalah dengan menggabungkan pemisahan dalam domain waktu dan frekuensi sekaligus secara bertahap (*Multistage ICA – MSICA*). Pada tahap pertama pengembangan ini (MSICA 1), TDICA dilakukan terlebih dahulu daripada FDICA. Output dari TDICA menjadi input bagi TDICA dan sinyal estimasi dari TDICA akan dicari pola frekuensi sesaatnya. Dengan menggunakan nilai parameter yang sama dengan sebelumnya diperoleh sinyal estimasi dengan pola dan kualitas.

Untuk kondisi besar langkah, blok proses dan jumlah iterasi yang sama, MSICA 1 menghasilkan konvergensi yang lebih baik dari metode-metode sebelumnya.

Tabel 4.3 Kondisi dan Kualitas MSICA 1

Parameter	Nilai
Besar Langkah	0.000001
Blok Proses	30
Jumlah Iterasi	100
Angle TDICA	0.0
Angle FDICA	0.0
SNR	12.6752

4.2.4 Frequency-Time ICA (MSICA 2)

Pada pendekatan multi tahap selanjutnya, FDICA dilakukan terlebih dahulu terhadap TDICA. Sinyal input dalam domain waktu ditransformasi fourier dalam domain frekuensi. Sinyal estimasi output FDICA juga dalam domain frekuensi dan ditransformasi balik (IFFT) ke dalam domain waktu untuk di-inputkan pada TDICA. Output dari TDICA berupa sinyal estimasi akhir dalam domain waktu.

Hasil pemisahan sinyal suara mesin dengan MSICA 2 untuk nilai parameter yang sama dengan metode sebelumnya dapat ditunjukkan pada tabel di bawah ini.

Tabel 4.4 Kondisi dan Kualitas MSICA 2

Parameter	Nilai
Besar Langkah	0.000001
Blok Proses	30
Jumlah Iterasi	100
Angle FDICA	1,4
Angle TDICA	0,3
SNR	22.9999

4.2.5 FastICA

Selanjutnya, *FastICA* dilakukan untuk dibandingkan dengan teknik ICA yang lain.

Hasil pemisahan sinyal suara mesin dengan *FastICA* untuk nilai parameter yang sama dengan metode sebelumnya dapat ditunjukkan pada tabel di bawah ini:

Tabel 4.5 Kondisi dan Kualitas FastICA

Parameter	Nilai
Besar Langkah	0.000001
Blok Proses	30
Jumlah Iterasi	100
SNR	26,667

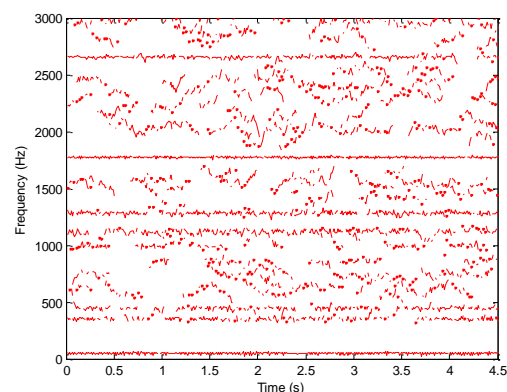
4.3 Perhitungan Frekuensi Sesaat

Pada sub-bab ini data yang ditampilkan dapat digolongkan menjadi dua jenis diagram atau grafik, yakni spektrogram sinyal dan frekuensi sesaatnya. Spektrogram menggambarkan sinyal dalam domain waktu-frekuensi. Diagram ini memperlihatkan perbedaan warna yang menunjukkan besarnya amplitudo (dB SPL) tiap frekuensi waktu. Pola warna biru mewakili amplitudo yang rendah sedang pola garis merah mewakili amplitudo tinggi.

Pola frekuensi sesaat menggambarkan perubahan fekuensi tiap sampel waktu. Grafik atau diagram ini berupa titik-titik serta garis, dimana titik-titik yang terhubung terus sepanjang waktu membentuk garis lurus menunjukkan frekuensi dominan dari kondisi tersebut. Pola frekuensi sesaat ini diperoleh dengan menurunkan fasa sinyal terhadap waktu.

Identifikasi sinyal estimasi terhadap sinyal asli yang telah dilakukan sebelumnya mengacu pada pola frekuensi sesaat kedua sinyal tersebut. Untuk menentukan bahwa pola frekuensi sesaat sinyal tersebut adalah *bearing fault* atau *unbalance* maka perlu ditelusuri pola frekuensi sesat sinyal asli (*baseline* penelitian). Penelusuran frekuensi sesaat dari empat kondisi motor tersebut adalah sebagai berikut:

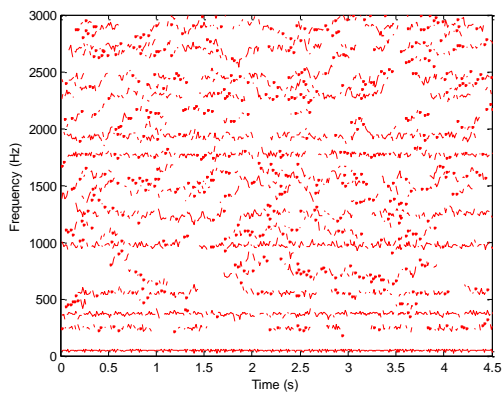
- Kondisi Normal

**Gambar 4.4** Pola Frekuensi Sesaat Sinyal kondisi Normal

Pada spektrogram kondisi normal terlihat pola garis gelap pada frekuensi rendah. Pola ini terlihat kembali di sekitar frekuensi 1000 Hz, 1700 Hz, serta 2600 Hz. Untuk mencari nilai numerik frekuensi tersebut maka spektrogram diubah menjadi diagram pola frekuensi sesaat yang menunjukkan lebih detail nilai frekuensi fundamentalnya. Dari penelusuran frekuensi sesaat akan didapat satu atau beberapa frekuensi fundamental, frekuensi tersebut dikarakterisasi sebagai frekuensi mesin pada kondisi normal.

- **Kondisi *Unbalance***

Pada kondisi *unbalance* terlihat spektrogram mulai berubah warna menjadi lebih merah. Hal ini mengindikasikan peningkatan amplitudo dari kondisi normal. Sedangkan penelusuran frekuensi sesaat menghasilkan perubahan yang tidak mencolok dibandingkan kondisi normal.

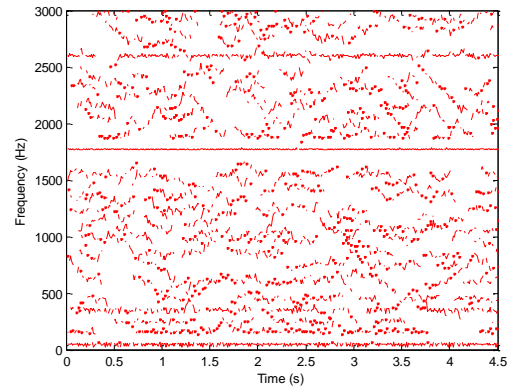


Gambar 4.5 Pola Frekuensi Sesaat Sinyal Kondisi Unbalance

Pada kondisi *unbalance* ini didapatkan frekuensi fundamental yang rendah yakni di sekitar 50 Hz, 360 Hz, 1700 Hz dan 1900 Hz. Frekuensi 2600 Hz yang sebelumnya tampak pada kondisi normal mulai tidak tampak pada kondisi *unbalance* ini. Kondisi *unbalance* pada penelitian ini dilakukan dengan penambahan masa 25 gr pada impeler motor/pompa.

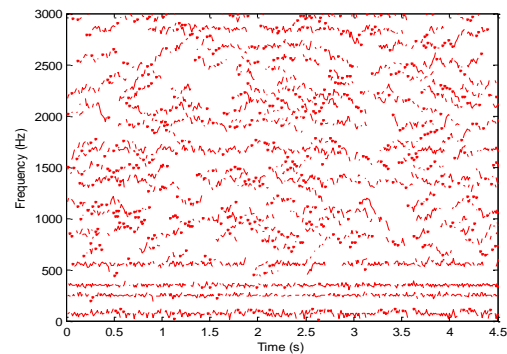
- **Kondisi *Misalignment***

Pada kondisi *misalignment*, motor diletakkan dengan posisi miring 30° dari kondisi normal. Spektrogram yang didapatkan masih mirip dengan kondisi normal namun terjadi peningkatan amplitudo dan juga frekuensi fundamental baru di sekitar 5200 Hz dan 5300 Hz.



Gambar 4.6 Frekuensi Sesaat Sinyal Kondisi Misalignment

- **Kondisi *Bearing Fault***



Gambar 4.7 Pola Frekuensi Sesaat Sinyal Kondisi Bearing Fault

Kondisi *Bearing Fault* atau kerusakan bearing diberikan dengan merusak rumah bearing penahan *ball bearing* atau disebut *ball cage*. Kerusakan ini mengakibatkan pergerakan bearing yang dapat merusak mesin/motor. Hasil citra spektrogram menunjukkan warna yang paling gelap daripada ketiga spektrogram pada kondisi sebelumnya. Hal tersebut menunjukkan bahwa pada kondisi ini suara mesin cenderung bertambah keras. Frekuensi fundamental pada kondisi ini cenderung dominan pada frekuensi-frekuensi rendah dengan amplitudo tinggi pada 70 Hz, 260 Hz, 350 Hz dan 586 Hz.

5. Kesimpulan Dan Saran

5.1 Kesimpulan

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

1. Kualitas pemisahan sinyal suara meningkat dengan metode FastICA dilihat dari nilai SNR yang meningkat yaitu 26,667 dB.
2. Dapat dideteksi masing-masing jenis kerusakan dengan melihat pola frekuensi sesaat tiap-tiap kerusakan.

5.2 Saran

Saran yang dapat digunakan untuk perkembangan penelitian Tugas Akhir selanjutnya, yaitu:

1. Pengembangan metode ekstraksi ciri lain yang dapat meningkatkan akurasi.
2. Tugas Akhir ini juga dapat dikembangkan menggunakan SMS Gateway untuk proses yang lebih *real time*.

DAFTAR PUSTAKA

- [1] Atmaja, B.T. Tugas Akhir. 2009. **Pemisahan Banyak Sumber Suara Mesin dari *Microphone Array* dengan Metode *Independent Component Analysis* (ICA) untuk Deteksi Kerusakan.** Jurusan Teknik Fisika ITS, Surabaya.
- [2] Anda, A R. Tugas Akhir. 2006. **Penggunaan Frekuensi Sesaat untuk Deteksi Pola Suara Kerusakan Motor Listrik.** ITS, Surabaya.
- [3] Antoni J. 2005. Blind Separation of Vibration Component: Principle and Demonstration. **Elsevier Mechanical System and Signal Processing** 19 1166-1180.
- [4] Bell, A J and Sejnowski T J. 1996. **The 'Independent Components' of Natural Scene are Edge Filters.** CNL: The Salk Institute (submitted to vision research).
- [5] Douglas, SC. 2001. Microphone Array Tehcnique and Application: **BSS of Acoustic Signal.** (Bredstein-D.Ware, Eds.). Berlin: Springer Inc, 2001.
- [6] Ihsan, Muhammad. Tugas Akhir. 2008. **Optimasi Metode Pengenalan Individu dengan Analisis Iris Mata Menggunakan Independent Component Analysis.** Jurusan Teknik Telekomunikasi IT Telkom, Bandung.
- [7] Mutiara, Nova Daryanti. Tugas Akhir. 2009. **Watermarking Citra Digital Menggunakan Spread Spectrum Pada Domain ICA (Independent Component Analysis).** Jurusan Teknik Informatika IT Telkom, Bandung.