

KLASIFIKASI TUBERKULOSIS DENGAN COMPRESSIVE SENSING DAN SUPPORT VECTOR MACHINE

CLASSIFICATION OF TUBERCULOSIS USING COMPRESSIVE SENSING AND SUPPORT VECTOR MACHINE

Aida Muhdina¹, Ledy Novamizanti², Gelar Budiman³

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom
¹aidamuhdina@student.telkomuniversity.ac.id, ²ledyaldn@telkomuniversity.co.id,
³gelarbudiman@telkomuniversity.ac.id

Abstrak

Tuberculosis (TBC) merupakan sebuah infeksi sistem pernapasan yang disebabkan oleh bakteri *Mycobacterium Tuberculosis*. Salah satu cara untuk mendeteksi bakteri ini yaitu dengan pemeriksaan spesimen dahak. Bakteri tersebut dilihat melalui mikroskop untuk dihitung jumlah bakteri tahan asamnya. Saat ini telah banyak dikembangkan penelitian untuk membantu perhitungan bakteri TBC. Sebagian besar penelitian masih mengklasifikasikan bakteri TBC dengan dua kelas dan menggunakan ekstraksi ciri yang kurang efisien dalam penyimpanan memori.

Sebuah sistem klasifikasi bakteri tuberkulosis dari citra mikroskopis dahak dirancang pada tugas akhir ini. Citra mikroskopis dahak akan melewati tahap pre-processing untuk memisahkan bakteri dari background. Citra yang telah melewati tahap pre-processing akan melewati tahap ekstraksi ciri dan klasifikasi. Sistem ini menggunakan metode Compressive Sensing sebagai ekstraksi ciri dan Support Vector Machine sebagai klasifikasi.

Penelitian ini berhasil mengklasifikasikan spesimen dahak kedalam tiga kelas yaitu positif, negatif dan scanty. Dataset spesimen dahak yang digunakan berisi 90 buah citra yang terbagi menjadi 63 data latih dan 27 data uji. Metode Compressive Sensing pada sistem ini memakai binary random matrix sebagai measurement matrix. Sedangkan untuk metode Support Vector Machine menggunakan kernel linear sebagai fungsi kernel dan One Against One sebagai metode SVM multiclass. Sistem ini dapat mengidentifikasi kelas spesimen dahak dengan akurasi 92,593% dengan waktu komputasi sebanyak 0.519 detik. Kata kunci : bakteri Tuberculosis, Compressive Sensing, Support Vector Machine

Abstract

Tuberculosis (TB) is an infection of the respiratory system caused by the bacteria *Mycobacterium Tuberculosis*. One way to detect these bacteria is by examining sputum specimens. The bacteria were seen through a microscope to count the number of acid-resistant bacteria. Currently, many studies have been developed to help the calculation of TB bacteria. Most of the studies still classify TB bacteria into two classes and use less efficient feature extraction in storage.

A tuberculosis bacteria classification system from sputum images is designed in this final project. This system uses Compressive Sensing method and Support Vector Machine. Compressive sensing is used as feature extraction, while Support Vector Machine is used as classification.

This study succeeded in classifying sputum specimens into three classes: positive, negative and scanty. The sputum specimen dataset used contains 90 images which are divided into 63 training data and 27 test data. The Compressive Sensing method in this system uses binary random matrix as measurement matrix. Meanwhile, the Support Vector Machine method uses linear kernel as kernel function and One Against One as SVM multiclass method. This system can identify the class of sputum specimens with an accuracy of 92.593% with a computation time of 0.519 seconds. Keywords: Detection of TB bacteria, Compressive Sensing, Support Vector Machine.

1. Pendahuluan

Tuberculosis (TBC) merupakan salah satu dari sepuluh penyebab utama kematian secara global. Menurut Global Tuberculosis Report 2019, pada tahun 2018 diperkirakan terdapat 1,23 juta

kematian disebabkan oleh TBC. Tingkat keparahan TBC sangat bervariasi antar negara. Menurut World Health Organization (WHO), Indonesia menempati urutan ketiga kasus TBC tertinggi yaitu sebanyak 8% dari total kasus TBC di dunia[1]. Analisis mikroskopis menggunakan dahak sebagai sampelnya. Sampel dahak kemudian diwarnai menggunakan metode Ziehl-Neelsen dan dihitung bakteri tahan asamnya.

Pada Tugas Akhir ini, penulis membuat suatu sistem untuk mendeteksi bakteri *Mycobacterium Tuberculosis* pada dahak penderita TBC. Sistem ini dibuat dengan metode segmentasi warna HSV dengan *Compressive Sensing* sebagai ekstraksi ciri dan *Support Vector Machine* sebagai sistem klasifikasi. Dalam penelitian ini digunakan beberapa referensi terkait pengolahan citra digital pada dahak TBC maupun metode yang digunakan. Referensi pertama berjudul *A Hybrid Approach to Diagnosis of Tuberculosis from Sputum*. Penelitian yang menggunakan segmentasi warna dengan klasifikasi *Fuzzy* tersebut menghasilkan sistem dengan *sensitivity* sebesar 93.9% dan *specificity* 88,2%[2]. Referensi kedua berasal dari penelitian dengan judul *Diagnosis and Counting of Tuberculosis Bacilli Using Digital Image Processing*. Penelitian tersebut menggunakan pendekatan dengan *thresholding* pada ruang warna HSI khususnya *channel hue*. Penelitian ini menghasilkan tingkat akurasi sebesar 90%[3].

Ekstraksi ciri yang digunakan pada penelitian ini adalah *Compressive Sensing*. *Compressive sensing* merupakan algoritma kompresi dan rekonstruksi. Namun pada penelitian dengan judul *Overlap Aware Compressed Signal Classification*, *Compressive Sensing* digunakan sebagai ekstraksi ciri[4]. Penelitian dengan judul *Image Feature Extraction Using Compressive Sensing* pun menggunakan *Compressive Sensing* sebagai ekstraksi ciri[5]. *Compressive Sensing* digunakan sebagai ekstraksi ciri karena *Compressive Sensing* dapat merekonstruksi sinyal yang telah terkompresi dengan data hasil kompresi yang sedikit. Oleh karena itu *Compressive Sensing* dianggap memiliki informasi deskriptif terhadap sinyal yang telah dikompresi.

2. Konsep Dasar

2.1 Tuberculosis (TBC)

Tuberculosis (TBC) adalah sebuah infeksi sistem pernapasan yang disebabkan oleh bakteri *Mycobacterium Tuberculosis*. TBC merupakan salah satu dari 10 penyebab utama kematian secara global. Menurut *Global Tuberculosis Report 2019*, pada tahun 2018 diperkirakan terdapat 1,23 juta kematian disebabkan oleh TBC[1].

Ada berbagai macam cara untuk mendeteksi TBC, salah satunya adalah analisis mikroskopis. Teknik mikroskopis menggunakan dahak sebagai sampel. Sampel dahak yang diambil, dikumpulkan dalam dua hari kunjungan yang berurutan. Dahak yang diambil kemudian diwarnai menggunakan metode Ziehl-Neelsen. Dengan metode Ziehl-Neelsen bakteri TBC akan berwarna merah, karena bakteri TBC merupakan bakteri tahan asam (BTA)[6].

2.2 Pengolahan Citra Digital

Citra merupakan gambaran dari suatu objek yang dihasilkan oleh peralatan digital dengan sistem sampling dan kuantisasi. Sistem *sampling* merupakan sistem yang merubah citra kontinu menjadi diskrit hingga menjadi citra digital. Sistem *sampling* bekerja dengan cara merubah citra kontinu menjadi M baris dan N kolom. Semakin besar nilai M dan N maka semakin halus citra digital yang dihasilkan. Sedangkan sistem kuantisasi merupakan sistem yang merubah intensitas dari analog atau kontinu menjadi diskrit. Proses tersebut menghasilkan gradasi warna sesuai dengan kebutuhan.

Pengolahan citra digital merupakan suatu proses yang bertujuan untuk mengolah, memanipulasi serta menganalisis citra. Proses pengolahan ini bertujuan untuk menghasilkan data atau informasi yang diinginkan. Salah satu contoh pengolahan citra digital adalah sistem klasifikasi.

2.3 Ruang warna

Ruang warna merupakan sebuah model matematika yang merepresentasikan informasi warna sebagai tiga atau empat komponen warna yang berbeda. Ruang warna yang paling sering digunakan dan biasanya merupakan ruang warna default untuk menyimpan dan merepresentasikan citra digital adalah *Red, Green, Blue* (RGB). Pada penelitian ini citra awal merupakan citra dengan ruang warna RGB. Citra tersebut akan di transformasikan ke ruang warna *Hue, Saturation, Value* (HSV). Pada ruang warna HSV *layer* yang akan diambil adalah *layer hue*. *Layer hue* merepresentasikan cahaya yang tampak seperti merah, hijau, biru, kuning, biru dan lainnya.

2.4 Compressive Sensing (CS)

Compressive Sensing (CS) merupakan teknik pemrosesan sinyal untuk memperoleh dan merekonstruksi sinyal. Hal ini terjadi berdasarkan pada prinsip bahwa *sparsity* sinyal dapat digunakan untuk memulihkan sinyal dengan jumlah bandwidth yang lebih sedikit daripada teorema Nyquist-Shannon[7]. Pada framework CS, sinyal diasumsikan sebagai *K-Sparse* dalam domain transformasi. Sinyal dengan panjang N adalah *K-Sparse* jika memiliki paling banyak K bukan nol dan (N-K) koefisien nol dalam domain transformasi. CS mengambil komposisi nilai-nilai sinyal pada beberapa sampel yang diambil. Sampel-sampel baru ini disebut pengukuran terkompresi y, dengan rumus sebagai berikut:

$$y = \phi \psi x \tag{1}$$

Dimana ϕ adalah matriks pengukuran $M \times N$, $M \ll N$, dan s adalah representasi domain transformasi *K-Sparse* dari sinyal x dalam domain transformasi yang diwakili oleh ψ . Salah satu karakteristik penting dari measurement matrix adalah matriks tersebut tidak perlu memiliki struktur khusus seperti matriks transformasi. Namun dalam beberapa kasus tertentu measurement matrix harus memenuhi *Restricted Isometry Property* (RIP). Dalam penelitian [8] membuktikan bahwa matriks acak seperti *Gaussian matrix* memenuhi RIP.

Pada penelitian ini metode CS yang digunakan hanya cukup pada kompresi CS. Karena CS dapat merekonstruksi secara sempurna sinyal asli yang telah dikompresi, maka dimungkinkan juga untuk menyatakan bahwa kompresi CS memiliki informasi deskriptif tentang sinyal asli. Hasil kompresi inilah yang akan digunakan sebagai ekstraksi ciri dalam proses klasifikasi[5].

2.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan metode yang digunakan untuk mengklasifikasikan berbagai kelompok data. Tujuan dari metode ini adalah mencari *hyperplane* yang dapat memisahkan *cluster* data, sehingga jarak antara rata-rata titik data dan *hyperplane* maksimum. *Hyperplane* adalah sebuah fungsi yang digunakan untuk klasifikasi di dalam ruang kelas dimensi yang tinggi. Selain *hyperplane*, terdapat juga *support vector* yaitu titik data maksimal pada SVM[9]. Untuk klasifikasi, data dipisahkan menjadi set pelatihan dan pengujian. Data yang digunakan untuk klasifikasi terdiri dari ciri yang dipilih dan nilai target yang sesuai atau kelas. Vektor ciri yang dipilih dan nilai target yang sesuai diberikan sebagai input ke model SVM. Ada beberapa data yang mungkin tidak bisa dipisahkan secara linear. Untuk mengatasi itu maka digunakan fungsi kernel. Dengan menambahkan fungsi kernel pada SVM maka data akan dipetakan ke ruang dengan dimensi yang lebih tinggi sehingga *hyperplane* dapat dikonstruksikan. Persamaan dari kernel *linear* adalah sebagai berikut:

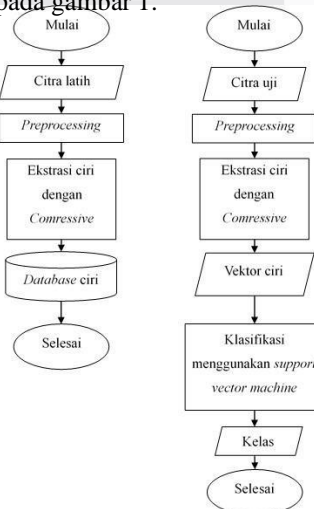
$$k(x_i, x_j) = x_i^T x_j \tag{2}$$

dimana x_i dan x_j adalah pasangan dua data training dan parameter σ dan p merupakan konstanta.

3. Model Sistem dan Perancangan

3.1. Desain Sistem

Tahapan sistem meliputi akuisisi citra, *pre-processing*, ekstraksi ciri dan klasifikasi. Desain sistem secara umum digambarkan pada gambar 1.

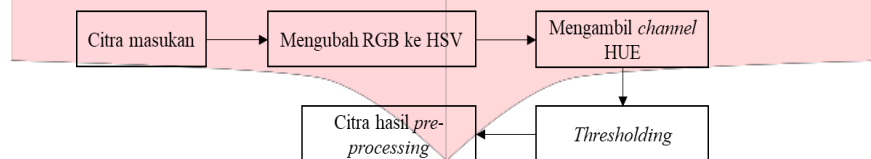


3.1.1 Akuisisi Citra

Akuisisi Citra adalah proses pengambilan sampel citra untuk diproses di dalam sistem. Citra yang diambil merupakan citra mikroskopis dahak. Citra dahak yang digunakan diambil dari penelitian sebelumnya yaitu penelitian [10] dan [11]. Citra dahak tersebut diambil dari Laboratorium daerah Jambi dengan menggunakan mikroskop dan kamera handphone yang beresolusi 13 MP dan 5 MP. Hasil citra yang diambil berformat JPG. Jumlah citra yang digunakan sebanyak 90 citra yang setiap kelasnya terbagi menjadi 30 citra negatif, 30 citra scanty dan 30 citra positif. Ukuran dari citra yang digunakan berbeda-beda dengan range 409×399 sampai 1175×757 .

3.1.2 Pre-processing

Pre-processing merupakan sebuah proses sebelum proses utama sistem dilakukan. *Pre-processing* bertujuan agar citra yang akan diolah seragam dan memiliki kualitas citra yang baik. Berikut adalah langkah-langkah pre-processing dari gambar 2.

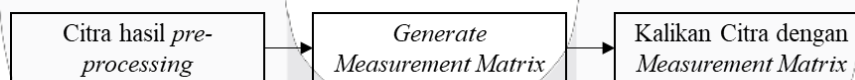


Gambar 2 Alur *pre-processing*

Berikut adalah langkah-langkah pre-processing dari gambar 3.3. Gambar 3.3 menampilkan beberapa blok untuk proses pre-processing. Blok RGB ke HSV berisi proses konversi citra dari ruang warna RGB ke ruang warna HSV. Kemudian diambil layer hue pada citra hasil konversi tersebut. Setelah itu dilakukan proses resizing mengubah ukuran citra menjadi ukuran yang sama. Ukuran yang digunakan adalah 300×300 piksel. Citra yang telah di resize kemudian melalui proses thresholding. Thresholding merupakan proses pemisahan objek dengan background. Proses ini dilakukan dengan memberikan nilai ambang batas sehingga piksel yang nilainya berada diatas ambang batas menjadi putih sedangkan piksel dengan nilai dibawah ambang batas menjadi hitam.

3.1.3 Ekstraksi Ciri

Ekstraksi ciri merupakan proses mengambil ciri-ciri yang dimiliki sebuah citra. Pada sistem ini ekstraksi ciri menggunakan metode Compressive Sensing (CS) dengan citra hasil pre-processing sebagai masukan. Tahap ini akan menghasilkan keluaran berupa matriks ciri berukuran 63×300 . Alur ekstraksi ciri dengan CS tergambar pada gambar 3.

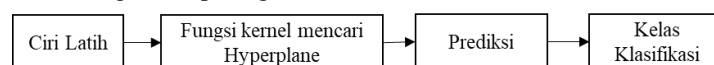


Gambar 3 Diagram alir ekstraksi ciri

Bagian pertama pada ekstraksi ciri dengan CS dimulai dari memasukan citra hasil pre-processing. Citra tersebut kemudian dirubah menjadi vektor baris. Vektor baris tersebut kemudian dikalikan dengan measurement matrix. Proses ini berulang hingga 63 kali pada data latih untuk membentuk database ciri. Hasil dari perkalian matriks ini menjadi ciri latih atau ciri uji yang akan digunakan sebagai masukan untuk proses klasifikasi.

3.1.4 Klasifikasi

Klasifikasi merupakan proses pengelompokan dari citra yang telah melewati proses ekstraksi ciri. Proses klasifikasi pada sistem ini menggunakan metode *Support Vector Machine* (SVM). SVM akan mencari hyperplane yang dapat memisahkan kelompok data ciri dari hasil ekstraksi CS. Proses klasifikasi dengan SVM tergambar pada gambar 4.



Gambar 4 Diagram alir SVM

Blok pertama pada gambar 4 yaitu mengambil matriks ciri sebagai masukan. SVM kemudian menggunakan fungsi kernel untuk mencari hyperplane agar seluruh data dapat dikelompokkan.

Setelah berhasil terkelompokan, SVM menghasilkan kelas atau label beserta anggotanya sehingga proses ini disebut prediksi. Hasil prediksi dari SVM kemudian dibandingkan dengan kelas dan anggota yang di kelompokkan secara manual. Hasil dari klasifikasi ini berupa kelas atau label. Terdapat tiga jenis kelas yaitu positif, negatif, dan scanty. Dimana citra akan dimasukkan ke dalam kelas negatif jika tidak ditemukan bakteri TBC pada dahak. Citra akan dimasukkan ke dalam kelas scanty jika memiliki 1-9 bakteri tahan asam. Citra akan dimasukkan ke dalam kelas positif jika memiliki 10-99 bakteri tahan asam.

3.2 Parameter Performansi

Setelah sistem telah berhasil dibangun, maka dilakukan pengujian untuk menilai performansi sistem. Penilaian performansi sistem memakai parameter sebagai berikut:

1. Akurasi

Akurasi merupakan ukuran ketepatan sistem dalam mengenali masukan, sehingga menghasilkan keluaran yang benar. Secara matematis dapat dirumuskan sebagai berikut:

$$Akurasi = \frac{T}{N} \times 100\% \tag{3}$$

2. Waktu Komputasi

Waktu komputasi adalah waktu yang dibutuhkan sistem untuk melakukan suatu proses. Pada sistem ini, waktu komputasi dihitung dengan cara mengurangi waktu selesai dengan waktu mulai, sehingga akan didapatkan waktu komputasi sistem. Secara matematis dituliskan dengan persamaan berikut:

$$W_k = W_s - W_m \tag{4}$$

Dimana W_k merupakan waktu komputasi, W_s merupakan waktu selesainya proses sistem, dan W_m merupakan waktu mulainya proses sistem.

4. Pengujian Sistem dan Analisis

Pengujian sistem merupakan tahap yang dilakukan dengan menjalankan beberapa skenario untuk mengetahui dan mendapatkan performansi terbaik dari sebuah sistem.

4.1 Pengujian nilai m

Pada skenario ini akan dianalisis performansi sistem terhadap perubahan nilai m. Pengujian ini bertujuan untuk mencari nilai m terbaik untuk dikombinasikan dengan sistem ini. Nilai m merupakan number of measurement yang akan menjadi salah satu ukuran dimensi measurement matrix dan salah satu ukuran dimensi vektor hasil ekstraksi ciri. Berikut tabel hasil pengujian nilai m:

Tabel 1 Hasil pengujian nilai m

| Parameter Performansi | Jenis Data | 300 | 600 | 900 | 1200 |
|-------------------------|------------|--------|--------|--------|--------|
| Rasio Kompresi (%) | | 99.67 | 99.37 | 99.00% | 98.67 |
| Akurasi (%) | Data Latih | 100 | 100 | 100 | 100 |
| | Data Uji | 92.593 | 92.593 | 88.889 | 88.889 |
| Sensitivity (%) | Data Latih | 100 | 100 | 100 | 100 |
| | Data Uji | 92.593 | 92.593 | 85.185 | 88.889 |
| Specificity (%) | Data Latih | 100 | 100 | 100 | 100 |
| | Data Uji | 93.939 | 93.939 | 87.578 | 89.167 |
| Waktu Komputasi (detik) | Data Latih | 0.583 | 0.821 | 1.176 | 1.551 |
| | Data Uji | 0.519 | 0.784 | 1.145 | 1.514 |

4.2 Pengujian Measurement Matrix

Pada skenario ini akan dianalisis performansi sistem terhadap perubahan jenis measurement matrix. Pengujian ini bertujuan untuk mencari measurement matrix terbaik untuk sistem ini. Measurement matrix merupakan matriks yang melewati operasi perkalian dengan vektor citra hasil pre-processing.

Tabel 2 Hasil pengujian measurement matrix

| Parameter Performansi | Jenis Data | Binary random | Gaussian |
|-------------------------|------------|---------------|----------|
| Akurasi (%) | Data Latih | 100 | 100 |
| | Data Uji | 92.593 | 44.44 |
| Sensitivity (%) | Data Latih | 100 | 100 |
| | Data Uji | 92.593 | 44.445 |
| Specificity (%) | Data Latih | 100 | 100 |
| | Data Uji | 93.939 | 48.334 |
| Waktu Komputasi (detik) | Data Latih | 0.583 | 0.454 |
| | Data Uji | 0.519 | 0.425 |

4.3 Pengujian Fungsi Kernel

Pada skenario ini akan dianalisis performansi sistem terhadap perubahan fungsi kernel. Pengujian ini bertujuan untuk mencari kernel terbaik yang akan digunakan pada sistem ini. Tabel hasil pengujian fungsi kernel sebagai berikut:

Tabel 3 Hasil pengujian fungsi kernel

| Parameter Performansi | Jenis Data | <i>Linear</i> | <i>Gaussian</i> | <i>Polynomial</i> |
|-------------------------|------------|---------------|-----------------|-------------------|
| Akurasi (%) | Data Latih | 100 | 100 | 60.317 |
| | Data Uji | 92.593 | 62.963 | 29.63 |
| <i>Sensitivity</i> (%) | Data Latih | 100 | 100 | 90.477 |
| | Data Uji | 92.593 | 62.963 | 29.629 |
| <i>Specificity</i> (%) | Data Latih | 100 | 100 | 45.029 |
| | Data Uji | 93.939 | 49.123 | 11.595 |
| Waktu Komputasi (detik) | Data Latih | 0.583 | 0.444 | 0.483 |
| | Data Uji | 0.519 | 0.419 | 0.416 |

4.4 Pengujian Metode SVM *Multiclass*

Pada skenario ini akan dianalisis performansi sistem terhadap perubahan metode SVM *multiclass*. Skenario ini bertujuan untuk mencari metode yang memiliki akurasi dan waktu komputasi yang terbaik untuk sistem. Metode yang digunakan dalam pengujian metode SVM *multiclass* ini adalah One Against One dan One Against All. Tabel hasil pengujian metode SVM *multiclass* sebagai berikut:

Tabel 4 Hasil pengujian SVM *multiclass*

| Parameter Performansi | Jenis Data | <i>One Against One</i> | <i>One Against All</i> |
|-------------------------|------------|------------------------|------------------------|
| Akurasi (%) | Data Latih | 100 | 100 |
| | Data Uji | 92.593 | 74.074 |
| <i>Sensitivity</i> (%) | Data Latih | 100 | 100 |
| | Data Uji | 92.593 | 74.074 |
| <i>Specificity</i> (%) | Data Latih | 100 | 100 |
| | Data Uji | 93.939 | 73.056 |
| Waktu Komputasi (detik) | Data Latih | 0.583 | 0.441 |
| | Data Uji | 0.519 | 0.429 |

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil analisis terhadap skenario pengujian yang telah dilakukan pada sistem klasifikasi Tuberculosis, maka dapat diambil kesimpulan sebagai berikut:

1. Perancangan dan pengujian sistem klasifikasi bakteri Tuberculosis dengan menggunakan metode *Compressive Sensing* dan *Support Vector Machine* telah berhasil dilakukan.
2. Ukuran m dengan nilai 300 dan binary random matrix merupakan konfigurasi terbaik pada sistem yang telah dibuat.
3. Kernel *linear* merupakan kernel terbaik untuk proses klasifikasi dengan SVM pada sistem ini.
4. Tingkat akurasi pada sistem ini mencapai 100% untuk data latih dan 92,593% untuk data uji.
5. Waktu komputasi pada sistem ini mencapai 29,081 detik untuk data latih dan 11,596 untuk data uji.

5.2 Saran

Berdasarkan kesimpulan diatas, sistem dapat dikembangkan menjadi lebih baik dengan saran sebagai berikut:

1. Melakukan uji coba dengan dataset yang lebih banyak agar sistem memiliki database ciri yang lebih lengkap.
2. Melakukan uji coba dengan measurement matrix lainnya untuk mendapatkan ekstraksi ciri yang lebih baik.
3. Melakukan uji coba dengan menggunakan kernel SVM yang lain agar sistem dapat mengenali data uji dengan lebih baik.
4. Membandingkan dengan metode ekstraksi ciri dan klasifikasi lainnya.

Referensi

- [1] WHO, "Global Tuberculosis Report 2019," 2019.
- [2] P. Ghosh, D. Bhattacharjee, and M. Nasipuri, "A hybrid approach to diagnosis of tuberculosis from sputum," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, pp. 771–776.
- [3] Y. Payasi and S. Patidar, "Diagnosis and counting of tuberculosis bacilli using digital image processing," in *2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC)*, 2017, vol. 2018-Janua, pp. 1–5.
- [4] M. Rani, S. B. Dhok, R. B. Deshmukh, and P. Kumar, "Overlap Aware Compressed Signal Classification," *IEEE Access*, vol. 8, pp. 52950–52967, 2020.
- [5] A. Eleyan, K. Kose, and A. E. Cetin, "Image Feature Extraction Using Compressive Sensing," in *Image Processing and Communications Challenges 5*, vol. 233, R. S. Choras, Ed. Heidelberg: Springer International Publishing, 2014, pp. 177–184.
- [6] Larissa, I. Parwati, and A. K. Sugianli, "Clinical Pathology and Majalah Patologi Klinik Indonesia dan Laboratorium Medik," *J. Indones.*, vol. 21, no. 2, pp. 135–137, 2015.
- [7] V. M. Patel and R. Chellappa, "Sparse Representations, Compressive Sensing and dictionaries for pattern recognition," in *The First Asian Conference on Pattern Recognition*, 2011, no. 5, pp. 325–329.
- [8] E. Candès, "Compressive sampling," in *Proceedings of the International Congress of Mathematicians Madrid, August 22–30, 2006*, Zuerich, Switzerland: European Mathematical Society Publishing House, 2006, pp. 1433–1452.
- [9] U. Asgher, K. Khalil, Y. Ayaz, R. Ahmad, and M. J. Khan, "Classification of Mental Workload (MWL) using Support Vector Machines (SVM) and Convolutional Neural Networks (CNN)," in *2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2020, pp. 1–6.
- [10] H. Mahfuzha, L. Novamizanti, and R. Rahmania, "Deteksi dan Klasifikasi Tuberculosis (TBC) berbasis CIELAB dengan Metode K-Means Clustering dan Support Vector Machine(SVM)," 2020.
- [11] A. N. Prayuni, L. Novamizanti, R. Rahmania, F. T. Elektro, U. Telkom, and N. Bayes, "Klasifikasi Jenis Tuberculosis (Tbc) Berbasis Rgb-Hsv Dengan Metode Naive Bayes," 2020.