

Klasifikasi Status Erupsi Gunung Berapi dengan menggunakan Analisa *Template Matching* dan Algoritma *K-Nearest Neighbors*

Fridy Mandita¹, Lintang Ragadanu Arizona²

^{1,2}Program Studi Teknik Informatika, Fakultas Teknik,
Universitas 17 Agustus 1945 Surabaya
Jln. Semolawaru No.45, Surabaya

[1fridymandita@untag-sby.ac.id](mailto:fridymandita@untag-sby.ac.id), [2lintangarizona1212@gmail.com](mailto:lintangarizona1212@gmail.com)

Abstrak

Salah satu jenis bencana alam yang terjadi di Indonesia adalah erupsi gunung vulkanik. Hal ini disebabkan Indonesia mempunyai ratusan gunung yang aktif dengan tipe erupsi dan status yang berbeda-beda untuk setiap gunung berapinya, dimana Indonesia merupakan negara yang dianggap sebagai pulau dengan jumlah gunung berapi aktif paling banyak di dunia. Meskipun seismograf digunakan untuk mencatat aktivitas gunung berapi, alat tersebut masih belum dapat mengklasifikasikan jenis tipe gempa dan status erupsi gunung berapi. Untuk mengatasi hal ini, peneliti mengusulkan penggunaan *Machine Learning* dalam mengklasifikasikan status erupsi gunung berapi. *Machine Learning* adalah salah satu cabang *Artificial Intelligence* (AI) yang memudahkan kinerja manusia. Terdapat tujuh cabang AI, termasuk *machine learning*. Dalam penelitian ini, digunakan metode *Machine Learning* berupa *Template Matching* dan *K-Nearest Neighbor* (KNN) untuk mengklasifikasikan tipe dan status erupsi gunung berapi. Hasil pengujian berupa akurasi matriks kebingungan, *recall*, presisi, dan *score* F1 dimana untuk menemukan K terbaik dalam model KNN dibagi menjadi K=1, 3, 5, 7, 11. Hasil yang didapat K terbaik pada rasio data 90:10 yaitu K=1 dimana akurasi pada data besar (mayoritas) sebesar 91% dan akurasi pada *all data* (data keseluruhan) sebesar 87%. Hasil pengujian juga berupa bagaimana model KNN memprediksi data baru menggunakan KNN.

Kata kunci—Erupsi gunung berapi, klasifikasi citra seismograf, *machine learning*, *template matching*, KNN

Abstract

One type of natural disaster that occurs in Indonesia is a volcanic eruption. This is because Indonesia has hundreds of active volcanoes with different types of eruptions and status for each volcano, Indonesia is a country that is considered to be the island with the largest number of active volcanoes in the world. Even though seismographs are used to record volcanic activity, these tools can still not classify the type of earthquake and the status of volcanic eruptions. To overcome this, researchers propose using Machine Learning to classify volcanic eruption status. Machine Learning is a branch of Artificial Intelligence (AI) that facilitates human performance. There are seven branches of AI, including machine learning. In this research, Machine Learning methods were used as Template Matching and K-Nearest Neighbor (KNN) to classify the type and status of volcanic eruptions. The test results are in the form of confusion matrix accuracy, recall, precision, and F1 score where to find the best K in the KNN model is divided into K=1, 3, 5, 7, 11. The results obtained are the best K at a data ratio of 90:10, namely K=1 where accuracy on big data (majority) is 91% and accuracy on all data (total data) is 87%. The test results also include how the KNN model predicts new data using KNN.

Keywords—*Volcanic eruptions, seismograph image classification, machine learning, template matching, KNN*

1. PENDAHULUAN

Erupsi gunung api merupakan bentuk bencana geologis yang sering terjadi di Indonesia, dimana magma keluar dari gunung menuju permukaan bumi melalui erupsi efusif atau eksplosif. Erupsi efusif terjadi ketika lava keluar secara perlahan dan mengalir tanpa ledakan, sedangkan erupsi eksplosif terjadi ketika lava keluar melalui ledakan dan diikuti oleh gempa. Indonesia adalah negara dengan jumlah gunung berapi aktif terbanyak di dunia, dan gunung-gunung tersebut termasuk dalam rangkaian gunung api aktif yang disebut sebagai “*ring of fire*” (Pratama, 2014). Indonesia terletak di antara cincin api, sehingga terdapat banyak gunung berapi. Banyaknya gunung berapi aktif di Indonesia membuat Indonesia sebagai negara dengan gunung berapi teraktif terbanyak di dunia (Adri et al., 2020). Gunung berapi biasanya meledak dalam periode pendek ada yang di setiap antara 1 - 6 tahun, sedangkan periode tengah tiap 6 – 8 tahun dan periode terpanjang sudah terjadi di beberapa gunung berapi yaitu 30 tahun (Rahayu et al., 2014).

Pusat Vulkanologi dan Mitigasi Bencana Geologi (PVMBG) di Indonesia sering memberikan masukan terkait kondisi gunung api. PVMBG membuat rekomendasi tentang keadaan aktivitas gunung berapi berdasarkan data pemantauan aktivitas masing-masing gunung berapi. Ada dua jenis pengamatan berdasarkan pengamatan visual dan faktor seismik. Adapun terkait lembaga yang salah satu tugasnya mengawasi aktivitas gunung api di negara yaitu Badan Meteorologi, Klimatologi, dan Geofisik (BMKG) (Firman et al., 2018)(Hafida, 2019).

Machine Learning (ML) telah digunakan secara luas dalam banyak penelitian untuk mengklasifikasikan letusan gunung berapi, tetapi terdapat masalah dalam mengklasifikasikan letusan gunung berapi menggunakan ML karena peneliti tidak mengetahui algoritma mana yang terbaik untuk mendeteksi sinyal seismik. Dalam upaya mengantisipasi situasi letusan gunung berapi vulkanik dan algoritma apapun memiliki akurasi yang tinggi dalam mengklasifikasikan sinyal seismik, pengujian dengan seismograf masih belum memberikan klasifikasi menurut jenis gempa dan jenis bahayanya, dan beberapa peneliti selalu menerapkan metode ini dengan menggunakan metode perhitungan manual atau ML tidak membantu mereka (Kom et al., 2021).

Penelitian yang dilakukan oleh (Firman et al., 2018) dalam hal klasifikasi penentuan status erupsi berapi dengan algoritma Naïve Bayes dan KNN, mendapat akurasi 79,71% pada algoritma Naïve Bayes dan 63,68% pada algoritma KNN. Penelitian ini menggunakan deviasi *K-Fold Cross Validation* dimana jarak akurasi data training status gunung api setiap eksperimen diungguli oleh algoritma Naïve Bayes. Penelitian lain terkait eksperimen klasifikasi status gunung api juga pernah dilakukan oleh (Wijaya et al., 2020). Eksperimen ini menggunakan metode *template matching* dimana akurasi klasifikasi peneliti didapat tajak akurasi sebesar 92,33% hanya saja dalam penelitian ini Sebagian besar tahap perhitungan dilakukan dengan cara manual, dengan demikian banyak algoritma yang bisa digunakan dalam klasifikasi status erupsi gunung api ini.

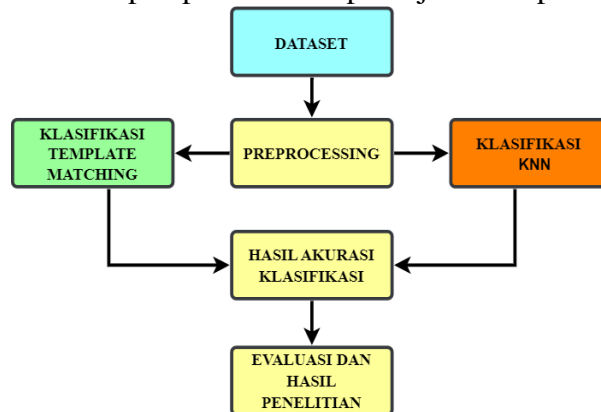
Berdasarkan permasalahan yang telah dijabarkan dari penelitian sebelumnya, penelitian ini bertujuan untuk mengklasifikasikan sinyal seismik dengan menggunakan metode *template matching* dan KNN. Peneliti menggunakan model *fitting analysis* dan

metode *K-Nearest Neighbor* (KNN) yang dipilih untuk klasifikasikan objek yang berbeda dan mendeteksi sinyal seismik. KNN sendiri merupakan metode klasifikasi data berbasis kelas yang tujuannya untuk mengklasifikasikan fitur baru berdasarkan atribut dan bahan ajar (Suwirmayanti et al., 2017). *Template matching* adalah teknik pengenalan pola yang kuno dan berguna dalam klasifikasi sinyal seismik. Algoritma ini bekerja dengan cara mengevaluasi model pola terhadap model citra dalam *database*. Kelemahan algoritma ini terletak pada terbatasnya model yang digunakan sebagai model acuan dalam basis data, menurut ukuran, bentuk dan kedudukannya (Bahri & Maliki, 2012). Dalam penelitian ini peneliti merumuskan masalahnya yaitu proses klasifikasi dari sebuah sinyal seismik untuk mengetahui tingkat akurasi.

2. METODE PENELITIAN

2.1. Tahapan Penelitian

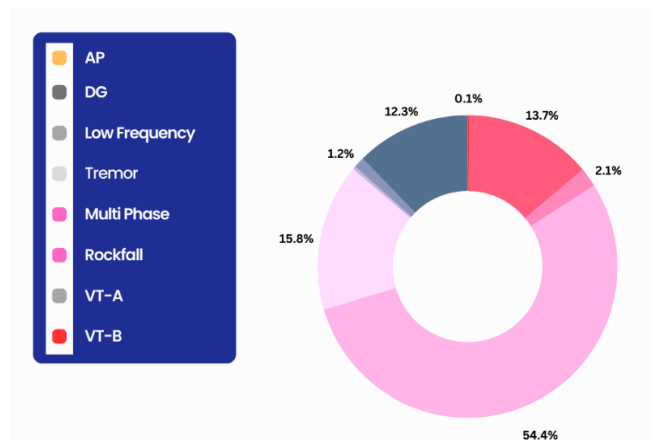
Adapun keseluruhan tahapan penelitian dapat dijabarkan pada gambar 1.



Gambar 1. Tahapan Penelitian

Terkait detail tahapan penelitian dijabarkan dalam subbab dalam artikel ini.

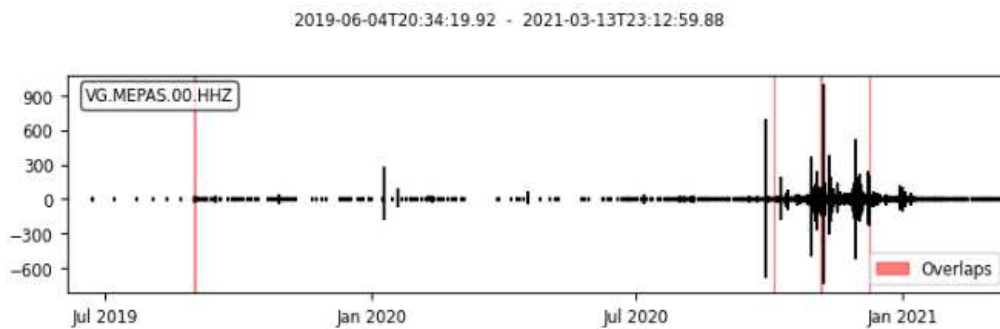
2.1.1 Dataset



Gambar 2. Diagram Dataset

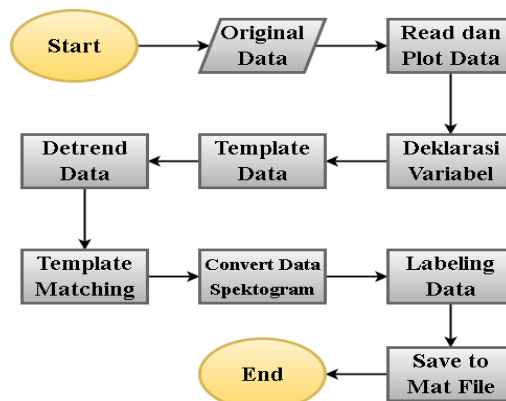
Gambar 2 dijelaskan bahwa *dataset* untuk saat ini yang digunakan dalam penelitian ini diambil dari citra seismograf dari Juli 2019 - Januari 2021 (*dataset* bisa bertambah),

dimana *dataset* yang digunakan dibagi atau diklasifikasi secara manual menjadi 8 kelas dimana masing-masing kelas berjumlah 2023 data *mseed* yaitu, *AP* berjumlah 3 data, *DG* berjumlah 278 data, *Low-Frequency* berjumlah 42 data, *Multi Phase* berjumlah 1101, *Rockfall* berjumlah 319 data, *Tremor* berjumlah 7 data, *VT-A* berjumlah 24 data, *VT-B* berjumlah 249 data. *Dataset* yang digunakan tersedia secara *online* dari *website* <https://www.fdsn.org/networks/?search=Merapi> untuk gunung Merapi. Penelitian akan dibagi menjadi dua eksperimen yaitu menggunakan data keseluruhan dan menggunakan data besar atau mayoritas (*dg*, *multiple phase*, *rockfall*, *vt-b*). Gambar 3 merupakan visualisasi data seismograf yang digunakan dalam penelitian ini dari Juli tahun 2019 sampai januari tahun 2021.



Gambar 3. Plot *dataset* keseluruhan penelitian

2.1.2 Preprocessing Data



Gambar 4. *Preprocessing Data*

Gambar 4 menampilkan beberapa tahap *preprocessing data* yang akan dijabarkan sebagai berikut:

1) *Original Data*

Original data merupakan data yang belum melalui proses atau manipulasi apa pun. Menurut Garno et al. (2019), *original data* dalam penelitian ini merupakan data *mseed* yang tidak bisa langsung digunakan dalam proses penelitian karena harus dilakukan pembacaan terlebih dahulu, kemudian dilakukan *preprocessing* agar data tersebut bisa terbaca dan dapat digunakan dalam proses selanjutnya. *Original data* berbentuk *binary file* sehingga tidak bisa langsung dilakukan eksperimen terhadap data tersebut ke dalam *google collab*.

2) *Read dan plot data*

Agar *original data* dapat terbaca, dilakukan proses membaca atau *mengimpor* data dari sumber data (*google drive*) ke dalam *collab* peneliti untuk kemudian diproses dan dianalisis lebih lanjut. Menurut Arrais et al. (2022), sumber data ini berbentuk *mseed* yaitu format standar untuk data seismik digital yang dikompresi secara efisien. Format *Mseed* dikompresi menjadi ukuran yang lebih kecil dari format SEED yang lebih lama, dan memungkinkan transfer data yang lebih cepat dan efisien melalui jaringan seismik. Setelah itu baru data *mseed* tersebut *diplot* berdasarkan masing-masing kelas dan diolah di *google collab*.

3) *Deklarasi Variabel*

Tahap ini merupakan proses untuk menamai variabel sesuai dengan jenis dan tipenya yang nantinya bertujuan agar memudahkan dalam proses penelitian yang akan dilakukan.

4) *Template Data*

Tahap selanjutnya yaitu melakukan pembuatan pola pada masing-masing kelas yang digunakan untuk mencari kemiripan atau pengenalan pola pada data *match*.

5) *Detrend Data*

Selanjutnya yaitu proses untuk menghapus atau mengurangi tren atau kecenderungan yang ada pada suatu rangkaian data. Menurut Rogers et al. (2021), *detrend data* dilakukan untuk menghilangkan atau mengurangi *trend* yang tidak relevan atau tidak berkontribusi terhadap informasi seismik yang ingin dicari. Metode yang digunakan yaitu regresi linier (*Linear regression*): Metode ini dilakukan dengan memperkirakan persamaan garis *trend* dari data dan mengurangi pengaruh *trend* tersebut dari data. Setelah *detrend* data dilakukan, data yang dihasilkan menjadi lebih stabil dan dapat memberikan informasi yang lebih akurat pada analisis seismik. *Detrend* data merupakan langkah awal yang penting dalam pengolahan data seismik dan diperlukan untuk memperoleh hasil analisis seismik yang lebih baik.

6) *Template Matching*

Setelah proses *detrend* selesai dilakukan, maka rumus *template matching* dapat diterapkan dimana data yang sudah *template* sebelumnya *dimatch* sesuai dengan pola target sesuai dengan kecocokan serta kemiripan data.

7) *Convert data to spektrogram*

Tahap selanjutnya merupakan proses untuk mengubah sinyal data dari domain waktu menjadi domain frekuensi dengan tujuan untuk memvisualisasikan informasi frekuensi dari sinyal tersebut. Spektrogram adalah gambar dua dimensi yang menunjukkan bagaimana intensitas sinyal bervariasi pada berbagai frekuensi dalam suatu rentang waktu tertentu.

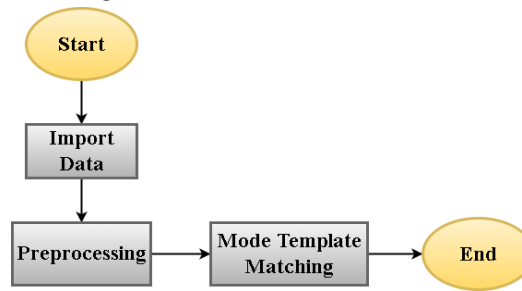
8) *Labeling data*

Tahap ini merupakan proses memberikan label atau kategori pada setiap data atau informasi yang ada dalam *dataset*. Proses *labelling* dilakukan secara otomatis dengan mengubah jenis seismik menjadi angka. Dimana jenis seismik yang akan digunakan dalam penelitian ini berjumlah 8 jenis atau tipe seismik.

9) *Save to mat.file*

Setelah semua data telah diproses menjadi data bersih maka hasil data tersebut disimpan bentuk *file* MATLAB. *File* matlab adalah format *file* yang digunakan oleh program matlab untuk menyimpan data numerik dalam bentuk matriks atau *array*.

2.1.3 Klasifikasi *Template Matching*



Gambar 5. Klasifikasi *Template Matching* (Bahri et al., 2012)

Gambar 5 menampilkan beberapa tahap klasifikasi *template matching* yang akan dijabarkan sebagai berikut:

1) *Import Dataset*

Tahap pertama dalam klasifikasi menggunakan *template matching* adalah mengimport *dataset* berbentuk *Mseed* terlebih dahulu.

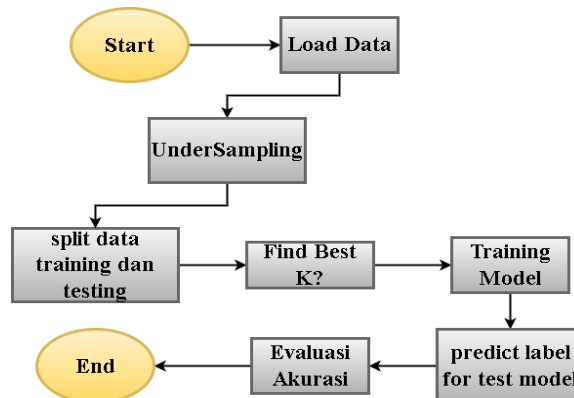
2) *Preprocessing*

Tahap ini akan dilakukan pembuatan deklarasi variabel, *template data*, *detrend data*, dan *labelling data*.

3) *Mode Template Matching*

Terakhir yaitu penerapan *template matching* yang digunakan untuk mencocokkan suatu sinyal dengan sebuah pola referensi atau *template matching* yang digunakan untuk membandingkan setiap bagian sinyal dengan *template* dan menghitung kesamaan atau perbedaan antara keduanya. Kemudian, teknik ini akan menghitung korelasi antara *template* dan sinyal yang sedang dianalisis untuk menentukan sejauh mana pola referensi cocok dengan sinyal.

2.1.4 Klasifikasi KNN



Gambar 6. Klasifikasi KNN (Firman et al., 2018)

Gambar 6 menampilkan beberapa tahapan Klasifikasi KNN yang akan dijabarkan sebagai berikut:

1) *Load Data*

Tahap pertama dalam klasifikasi menggunakan KNN adalah *load data* yang sudah selesai dilakukan pada tahap *preprocessing* menggunakan *template matching*. Dimana data yang akan digunakan yaitu data yang sudah tersimpan didalam *file* matlab yang telah diolah sebelumnya.

2) *Undersampling*

Dikutip dari Magnolia et al. (2022), tahap selanjutnya yaitu teknik pengolahan data yang digunakan untuk mengurangi jumlah data pada kelas mayoritas atau yang lebih banyak dalam sebuah *dataset*, sehingga dapat menyamakan jumlah data antara kelas mayoritas dan minoritas atau yang lebih sedikit. *Undersampling* digunakan karena dalam klasifikasi data, terdapat data yang tidak seimbang (*imbalanced data*). Pada *dataset* yang tidak seimbang, kelas mayoritas cenderung memiliki lebih banyak data dibandingkan kelas minoritas. Hal ini dapat mengakibatkan performa model klasifikasi menurun, karena model cenderung akan memperhatikan kelas mayoritas dan mengabaikan kelas minoritas. Oleh karena itu, dengan menggunakan teknik *undersampling*, data pada kelas mayoritas dapat dikurangi hingga jumlah data pada kelas mayoritas dan minoritas menjadi seimbang, sehingga model klasifikasi dapat lebih fokus pada kelas minoritas.

3) *Split data training dan data testing*

Tahap selanjutnya yaitu dilakukan untuk membagi data menjadi dua bagian yaitu data *training* dan data *testing*. Data *training* digunakan untuk melatih model KNN, sedangkan data *testing* digunakan untuk mengevaluasi performa model yang telah dilatih. Proporsi masing-masing yaitu data *training* sebesar 90% sedangkan data *testing* sebesar 10%.

4) *Find best K?*

Pada penelitian ini, nilai *k* merupakan parameter yang menentukan jumlah tetangga terdekat yang digunakan untuk membuat prediksi. Nilai *K* yang akan digunakan dalam penelitian ini yaitu 1,3,5,7,dan 11.

a. *Training Model*

Setelah proses mencari *best k* selanjutnya dilakukan melatih model knn menggunakan data yang telah diproses dan siap digunakan. Pada tahap ini, model akan belajar dari data yang telah diberikan dan menghasilkan parameter atau bobot yang dapat digunakan untuk membuat prediksi serta optimasi pada data baru. *Training* model adalah tahap kritis dalam penelitian ini karena keakuratan dan efektivitas model bergantung pada kualitas data, dan parameter yang dioptimalkan dengan baik. Oleh karena itu, perlu melakukan evaluasi model secara terus-menerus dan memperbarui model saat diperlukan untuk memastikan model yang dihasilkan dapat menghasilkan prediksi yang akurat.

b. *Test Model*

Pada tahap ini proses pengujian kinerja model KNN pada data yang belum pernah dilihat sebelumnya. Tujuannya adalah untuk mengetahui sejauh mana model dapat menggeneralisasi dan melakukan prediksi yang akurat pada data baru yang belum pernah dilihat. Proses test model dilakukan setelah model dilatih dengan data *training* untuk memprediksi tipe seismik yang baru yang ada pada label.

5) *Evaluasi Akurasi*

Terakhir yaitu evaluasi akurasi merupakan proses mengukur seberapa baik model knn dapat memprediksi kelas atau label pada data yang belum dilihat sebelumnya (*data testing*). Akurasi adalah salah satu metrik evaluasi yang umum digunakan untuk mengukur performa model, khususnya pada masalah klasifikasi. Akurasi dihitung dengan membandingkan jumlah prediksi yang benar (*true positives*) dengan jumlah total data yang diuji (*true positives + false positives*). Secara matematis, akurasi dapat dinyatakan sebagai: $akurasi = (true\ positives + true\ negatives) / total\ data$ Nilai akurasi berkisar

antara 0 hingga 1, dengan nilai 1 menunjukkan bahwa model melakukan prediksi dengan sempurna dan nilai 0 menunjukkan bahwa model tidak dapat membedakan antara kelas yang berbeda sama sekali.

Namun, penggunaan akurasi sebagai satu-satunya matrik evaluasi memiliki kelemahan pada kasus-kasus di mana data tidak seimbang (*imbalanced data*). Sebagai contoh, jika kelas minoritas memiliki jumlah data yang sedikit dibandingkan kelas mayoritas, akurasi yang tinggi dapat diperoleh hanya dengan memprediksi semua data sebagai kelas mayoritas. Oleh karena itu, pada kasus seismik ini data yang digunakan *imbalance* maka peneliti akan membandingkan hasil akurasi data besar (data mayoritas) dengan data keseluruhan menggunakan K yang berbeda-beda, sehingga bisa diketahui performa model dalam melakukan klasifikasi seismik.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 7. *confusion matrix* (Santra & Christy, 2012)

- True Positive (TP)* Hasil yang diperoleh jika kelas 1 bertemu dengan kelas 1
- Negative (TN)* Hasil yang diperoleh jika kelas 0 bertemu dengan kelas 0
- Positive (FP)* Hasil yang diperoleh jika kelas 0 bertemu dengan kelas 1
- Negative (FN)* Hasil yang diperoleh jika kelas 0 bertemu dengan kelas 0
- TP** Jumlah data dari kelas 1 yang benar diklasifikasikan sebagai kelas 1
- TN** Jumlah data dari kelas 0 yang benar diklasifikasikan sebagai kelas 0
- FP** Jumlah data dari kelas 0 yang salah diklasifikasikan sebagai kelas 1
- FN** Jumlah data dari kelas 1 yang salah diklasifikasikan sebagai kelas 0

Rumus dari *confusion matrix* untuk menghitung *accuracy*, *precision*, *recall*, dan *f1-score* seperti berikut:

1) *Accuracy*

Rasio prediksi yang benar dengan keseluruhan data. Adapun rumus untuk menghitung nilai akurasi adalah sebagai berikut:

$$\text{Akurasi} = \frac{TP+TN}{TP+FN+FP+TN} \times 100\% \quad (1)$$

2) *Precision*

Rasio dari prediksi benar positif dengan keseluruhan data yang diprediksi sebagai positif. Adapun rumus untuk menghitung nilai *precision* adalah sebagai berikut:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

3) *Recall*

Rasio dari prediksi benar positif dengan keseluruhan data yang benar positif. Adapun rumus untuk menghitung nilai *recall* adalah sebagai berikut:

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

4) *F1-score*

Pembobotan dari perbandingan rata – rata sebuah *precision* dan *recall*. Adapun rumus untuk menghitung nilai *f1-score* adalah sebagai berikut:

$$F1\text{-score} = 2 * \frac{\text{recall} * \text{precision}}{\text{recall} + \text{precision}} \quad (4)$$

3. HASIL DAN PEMBAHASAN

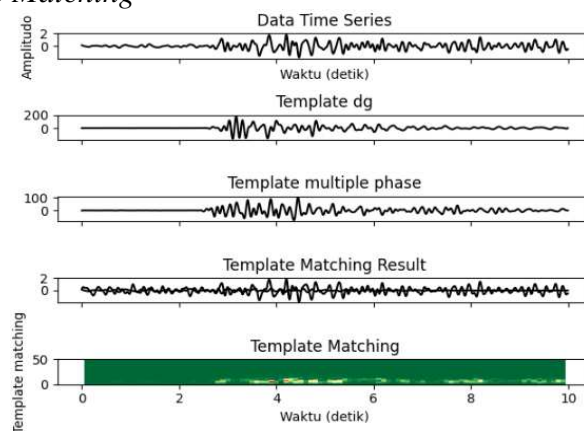
3.1. Data Penelitian

Dataset untuk saat ini yang digunakan dalam riset ini diambil dari citra seismograf (*dataset* bisa bertambah). Dimana *dataset* yang digunakan dibagi atau diklasifikasi secara manual menjadi 8 kelas yang mana tiap - tiap kelas sebanyak 2023 data mseed yaitu, *AP* berjumlah 3 data, *DG* berjumlah 278 data, *Low-Frequency* berjumlah 42 data, *Multi Phase* berjumlah 1101, *Rockfall* berjumlah 319 data, *Tremor* berjumlah 7 data, *VT-A* berjumlah 24 data, *VT-B* berjumlah 249 data. *Dataset* yang digunakan tersedia secara *online* dari *website* yang tersedia dan dapat diakses kapanpun di <https://www.fdsn.org/networks/?search=Merapi> untuk gunung Merapi. Penelitian akan dibagi menjadi dua eksperimen yaitu menggunakan data keseluruhan lalu menggunakan data besar atau mayoritas (*dg*, *multiple phase*, *rockfall*, *vt-b*).

Tabel 1. Jumlah Data *Seismic Events*

No.	Kategori Semua Kelas	Jumlah File
1.	<i>AP</i>	3
2.	<i>DG</i>	278
3.	<i>Low Frequency</i>	42
4.	<i>Multi Phase</i>	1101
5.	<i>Rockfall</i>	319
6.	<i>Tremor</i>	7
7.	<i>VT-A</i>	24
8.	<i>VT-B</i>	249

3.2. Hasil *Template Matching*



Gambar 8. Hasil *template matching*

Gambar 8 merupakan hasil visualisasi *template matching* dimana gambar pertama paling atas merupakan *data time series* yang akan dicocokkan pada data di bawahnya yang merupakan *template* dari *DG* dan *Multi-Phase* data tersebut di *plot* berdasarkan panjang amplitudo, selanjutnya dari hasil pencocokkan *data time series* tersebut akan muncul *template matching result* dari hasil pencocokan *data match* dan *data template*. Gambar bawah sendiri yang berwarna hijau merupakan hasil *spectrogram* dimana jika terdapat warna kuning atau merah pada data itu merupakan *noise* dan indikasi adanya gelombang gempa yang terjadi pada hasil data, yang dihitung setiap 10 detik per-data.

Tabel 2. Hasil Pengujian Data besar (Mayoritas)

K	Accuracy	Precision	Recall	F1 Score
K = 1	0.91	0.91	0.91	0.91
K = 3	0.87	0.87	0.87	0.87
K = 5	0.85	0.85	0.85	0.85
K = 7	0.82	0.80	0.82	0.80
K = 11	0.81	0.78	0.81	0.79

Tabel 2 menampilkan hasil pengujian klasifikasi pada data mayoritas dalam lima perhitungan KNN dengan $k=1$, $k=3$, $k=5$, $k=7$, dan $k=11$, semakin besar nilai k maka performa model akan semakin menurun. Pada kasus ini, nilai **K=1** memberikan performa yang paling baik dengan *accuracy* sebesar 91%, *precision* sebesar 91%, *recall* sebesar 91%, dan *f1-score* sebesar 91%. Namun, pada kasus kedua dan seterusnya akurasi model terlihat menurun, sehingga model menjadi kurang baik untuk mengklasifikasikan data baru. Oleh karena itu, dapat disimpulkan bahwa nilai k yang optimal untuk model KNN pada data mayoritas adalah $k=1$. Namun, hal ini dapat berbeda-beda tergantung pada *dataset* yang digunakan dan perlu dilakukan evaluasi untuk menentukan nilai k yang optimal.

Tabel 3. Hasil Pengujian *All Data*

K	Accuracy	Precision	Recall	F1 Score
K = 1	0.87	0.88	0.87	0.87
K = 3	0.79	0.79	0.79	0.79
K = 5	0.75	0.72	0.75	0.73
K = 7	0.77	0.71	0.77	0.74
K = 11	0.76	0.71	0.76	0.73

Tabel 3 menampilkan hasil pengujian klasifikasi pada *all data* dalam lima perhitungan KNN dengan $k=1$, $k=3$, $k=5$, $k=7$, dan $k=11$, sama seperti pada pengujian data mayoritas semakin besar nilai k maka performa model akan semakin menurun. Pada kasus ini, nilai **K=1** memberikan performa yang paling baik dengan *accuracy* sebesar 87%, *precision* sebesar 88%, *recall* sebesar 87%, dan *f1-score* sebesar 87%. Namun, pada

kasus kedua dan seterusnya akurasi model terlihat menurun, sehingga model menjadi kurang baik untuk mengklasifikasikan data baru. Oleh karena itu, dapat disimpulkan bahwa nilai k yang optimal untuk model KNN pada *data all* adalah **K=1**. Namun, hal ini dapat berbeda-beda tergantung pada *dataset* yang digunakan dan perlu dilakukan evaluasi untuk menentukan nilai k yang optimal.

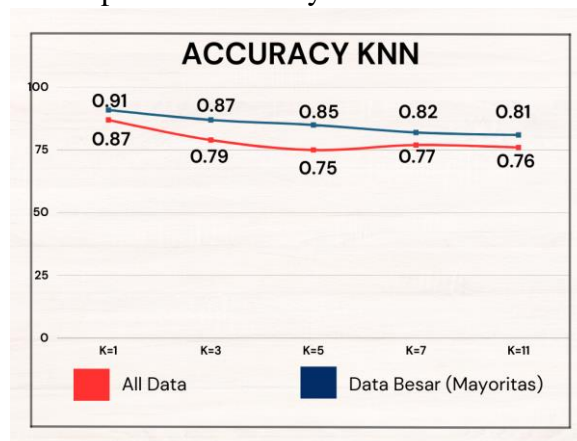
Tabel 4. Hasil Pengujian *cross-validation* data mayoritas

5 Fold Test	Score test 1	Score test 2	Score test 3	Score test 4	Score test 5	Rata-rata
K=1	0.86	0.82	0.84	0.84	0.87	0.84
K=3	0.80	0.81	0.78	0.81	0.81	0.80
K=5	0.80	0.81	0.79	0.82	0.79	0.80
K=7	0.81	0.78	0.80	0.81	0.80	0.80
K=11	0.80	0.83	0.76	0.79	0.80	0.76

Tabel 5 Hasil Pengujian *cross-validation* keseluruhan data

5 Fold Test	Score test 1	Score test 2	Score test 3	Score test 4	Score test 5	Rata-rata
K=1	0.83	0.79	0.83	0.84	0.80	0.82
K=3	0.76	0.73	0.80	0.78	0.73	0.76
K=5	0.75	0.71	0.78	0.77	0.75	0.75
K=7	0.75	0.74	0.80	0.78	0.71	0.76
K=11	0.77	0.76	0.79	0.78	0.70	0.76

Tabel 4 dan 5 dapat dilihat bahwa akurasi performa model tiap lipatan dan tiap K mengalami penurunan ketika diterapkan *cross validation* pada data sebanyak 5 *fold* (lipatan), meskipun begitu dapat disimpulkan dalam penelitian ini model KNN mampu dalam melakukan klasifikasi sinyal gempa baik menggunakan *cross validation* maupun tidak dengan akurasi rata-rata diatas 0.75 yang menandakan model KNN mampu digunakan dalam klasifikasi pada *dataset* sinyal seismik.



Gambar 9. Hasil perbandingan akurasi antar K

Gambar 9 dapat disimpulkan bahwa baik pada data mayoritas maupun *data all*, model KNN mampu melakukan klasifikasi data seismik dengan baik meskipun ketika penambahan jumlah K, akurasi model dalam memprediksi data baru semakin menurun.

4. KESIMPULAN

Dari penelitian yang telah dilakukan dan juga pembahasan sebelumnya dihasilkan beberapa kesimpulan sebagai berikut:

1. Berdasarkan penelitian yang telah dilakukan, hasil menunjukkan bahwa *template matching* dan model KNN cocok digunakan dalam klasifikasi sinyal seismik karena memiliki akurasi tinggi dengan *dataset* yang digunakan berjumlah 2023 dengan tipe Mseed. Dari percobaan terhadap 5 K pada model KNN didapat hasil akurasi pada data mayoritas $K1=0.91$, $K3=0.87$, $K5=0.85$, $K7=0.82$, $K11=0.81$ sedangkan pada keseluruhan data di dapat akurasi masing-masing $K1=0.87$, $K3=0.79$, $K5=0.75$, $K7=0.77$, $K11=0.76$. Didapatkan bahwa hasil akurasi terbaik terletak pada $K=1$ pada masing-masing *dataset*, meskipun terjadi penurunan Ketika K yang di uji semakin besar tetapi kecenderungan penurunan masih stabil dan tidak ada lonjakan akurasi yang signifikan. Hal ini juga dapat memberikan kesimpulan bahwa model KNN lebih baik dalam mengklasifikasikan sinyal ketika *dataset* yang dipakai *balance* pada setiap kelas daripada data yang *imbalance*.
2. Setelah melakukan pengujian terhadap rasio data 90:10 pada masing-masing *dataset*, parameter $K=1$ memberikan akurasi tertinggi dibandingkan dengan K lainnya. Hal ini dikarenakan pada rasio data 90:10, didapatkan *data training* yang lebih besar sehingga model dapat mengenali pola data lebih baik. Hal ini juga terjadi karena model KNN bekerja dengan mengenali tetangga terdekatnya, sehingga pada *dataset* yang terpisah relatif jelas dan terpisah secara baik pada saat *preprocessing*, tetangga terdekat secara efektif mewakili kelas yang sesuai.
3. Penggunaan *cross-validation* dengan 5 fold (lipatan) secara konsisten menghasilkan akurasi yang lebih rendah dibandingkan dengan pengujian tanpa menggunakan *cross-validation* pada model KNN dalam kasus klasifikasi sinyal seismik.

Dari beberapa poin diatas dapat disimpulkan bahwa analisa *Template Matching* dan model KNN cocok digunakan pada kasus klasifikasi sinyal seismik terutama pada data yang *balance*.

5. SARAN

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat diberikan untuk mengembangkan penelitian terkait klasifikasi sinyal seismik menggunakan model KNN yang telah dibuat:

1. Memperluas penelitian dengan membandingkan model KNN dengan algoritma klasifikasi lainnya, seperti *Decision Tree*, *Neural Networks* atau *Random Forest*, untuk menentukan metode mana yang paling cocok untuk klasifikasi, khususnya menggunakan *dataset Mseed* (sinyal seismik).
2. Melakukan eksperimen lain dengan menggunakan parameter K yang berbeda - beda dengan tipe *dataset* baik Mseed maupun image pada kategori sinyal seismik. Uji juga

model pada *data balance* maupun *imbalance* dengan jumlah *dataset* diatas 2000 untuk melihat apakah KNN masih mampu dalam melakukan klasifikasi dengan jumlah *dataset* yang lebih banyak.

3. Melakukan eksplorasi teknik *resampling* lainnya selain *cross-validation*, pertimbangkan untuk menggunakan teknik *resampling* lainnya seperti *Leave-One-Out Cross-Validation* (LOOCV), *Bootstrap*, *Holdout Validation*. Eksplorasi juga teknik pada model yang akan diterapkan, pertimbangkan untuk menggunakan *Synthetic Minority Over-sampling Technique* (SMOTE) atau *Adaptive Synthetic Sampling* (ADASYN). Bandingkan hasil akurasi dan evaluasi performa model dari teknik *resampling* yang berbeda untuk melihat apakah ada perbedaan yang signifikan.

DAFTAR PUSTAKA

- Adri, W., Sabri, L. ., & Wahyuddin, Y. (2020). Pembuatan Peta Jalur Evakuasi Bencana Gunung Api Dan Persebaran Lokasi Shelter Menggunakan Metode Network Analyst (Studi Kasus : Gunung Merapi, Boyolali-Magelang). *Jurnal Geodesi UNDIP*, 10(1), 189–196. <https://ejournal3.undip.ac.id/index.php/geodesi/article/view/29693>
- Adrianus, A., Manik, H. M., & Nainggolan, T. B. (2021). *Atenuasi Derau Acak Dan Koheren Pada Data Seismik 2d Di Perairan Aru , Papua Attenuation Of Random And Coherent Noise On 2d Seismic Data Of Aru Waters ., April*, 57–70.
- Andry, et al . (2022). *Perbandingan metode Template Matching dengan K-Nearest Neighbor dalam identifikasi karakter studi kasus plat nomor kendaraan*. 10(01).
- Arrais, S., Urquiza-Aguiar, L., & Tripp-Barba, C. (2022). Analysis of Information Availability for Seismic and Volcanic Monitoring Systems: A Review. *Sensors*, 22(14), 1–27. <https://doi.org/10.3390/s22145186>
- Bahri, R. S., & Maliki, I. (2012). Perbandingan Algoritma Template Matching Dan Feature Extraction Pada Optical Character Recognition. *Urnal Komputer Dan Informatika (KOMPUTA)*, 1(1), 187–198. http://repo.pens.ac.id/1324/1/Paper_TA_MBAH.pdf
- Evan ESDM, et al. (2012). Pengenalan Gunungapi, kepada masyarakat agar lebih waspada terhadap sekitar terutama lokasi warga yang dekat dengan aktivitas gunung berapi. *VSI Departemen ESDM*, 12.
- Firman, Muhammad, M., & Khairan, A. (2018). Perbandingan Klasifikasi Antara KNN dan Naive Bayes pada Penentuan Status Gunung Berapi dengan K-Fold Cross Validation. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(5), 577. <https://doi.org/10.25126/jtiik.201855983>
- Garno, G., & Adam, R. I. (2019). Skema Penyembunyian Data pada Gambar Berbasis Interpolasi Kubik B-Spline Menggunakan Metode Least Significant Bit (LSB). *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, 5(3), 255. <https://doi.org/10.26418/jp.v5i3.37584>
- Hafida, S. H. N. (2019). Perubahan Kesiapan Masyarakat dalam Menghadapi Bencana Erupsi Gunung Berapi. *Jupiis: Jurnal Pendidikan Ilmu-Ilmu Sosial*, 11(2), 396. <https://doi.org/10.24114/jupiis.v11i2.13955>
- Iman, S. G., & Syamsuddin, E. Y. (2020). Perancangan Algoritma Optimasi Pada Pengenalan Karakter Plat Nomor Kendaraan Menggunakan Pengolahan Citra. *Komputika : Jurnal Sistem Komputer*, 9(2), 155–164.

- <https://doi.org/10.34010/komputika.v9i2.3682>
- Isman, Andani Ahmad, & Abdul Latief. (2021). Perbandingan Metode KNN Dan LBPH Pada Klasifikasi Daun Herbal. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(3), 557–564. <https://doi.org/10.29207/resti.v5i3.3006>
- Jeklin, A. (2016). *Implementasi Template Matching Correlation Dan K – Nearest Neighbor Untuk Mendeteksi Hukum Bacaan Tajwid Pada Citra Tulisan Al-Qur'an*. July, 1–23.
- Kom, M., Ekonomi, F., Uin, I., & Utara, S. (2021). *Nabillah Purba*. 9(2), 91–98.
- Lailatul et al. (2020). Analisis Spektrum Sinyal Gempa Vulkano-Tektonik (VT-A dan VT-B) di Gunung Kelud. *Kaos GL Dergisi*, 8(75), 147–154. <https://doi.org/10.1016/j.jnc.2020.125798><https://doi.org/10.1016/j.smr.2020.02.002><http://www.ncbi.nlm.nih.gov/pubmed/810049><http://doi.wiley.com/10.1002/anie.197505391><http://www.sciencedirect.com/science/article/pii/B9780857090409500205>
- Magnolia, C., Nurhopipah, A., & Kusuma, A. (2022). Penanganan Imbalanced Dataset untuk Klasifikasi Komentar Program Kampus Merdeka Pada Aplikasi Twitter. *Edu Komputika*, 9(2), 105–113.
- Pratama, A. (2014). Pemodelan Kawasan Rawan. *Jurnal Geodesi Undip*, 3(April), 28–43.
- Putri, D. N., Rafmadhanty, F., Megantara, I. P., Jayanti, I. N., & Hapsari, K. D. (2014). *Klasifikasi Huruf Korea (Hangul) dengan Metode Template Matching Correlation*. 1–5.
- Rahayu, R., Ariyanto, D. P., Komariah, K., Hartati, S., Syamsiyah, J., & Dewi, W. S. (2014). Dampak Erupsi Gunung Merapi Terhadap Lahan Dan Upaya-Upaya Pemulihannya. *Caraka Tani: Journal of Sustainable Agriculture*, 29(1), 61. <https://doi.org/10.20961/carakatani.v29i1.13320>
- Reza Noviansyah, M., Rismawan, T., & Marisa Midyanti, D. (2018). Penerapan Data Mining Menggunakan Metode K-Nearest Neighbor Untuk Klasifikasi Indeks Cuaca Kebakaran Berdasarkan Data Aws (Automatic Weather Station) (Studi Kasus: Kabupaten Kubu Raya). *Jurnal Coding, Sistem Komputer Untan*, 06(2), 48–56.
- Rogers, B., Giles, D., Draper, N., Mourot, L., & Gronwald, T. (2021). Influence of artefact correction and recording device type on the practical application of a non-linear heart rate variability biomarker for aerobic threshold determination. *Sensors (Switzerland)*, 21(3), 1–16. <https://doi.org/10.3390/s21030821>
- Rosyadi, M. D. (2017). Pengenalan Motif Dasar Pada Kain Sasirangan. *Technologia*, 8(2), 53–61.
- Santra, a. K., & Christy, C. J. (2012). Genetic Algorithm and Confusion Matrix for Document Clustering. *International Journal of Computer Science*, 9(1), 322–328. <http://ijcsi.org/papers/IJCSI-9-1-2-322-328.pdf>
- Saputra, A. D., Jayanta, & Pangaribuan, I. A. B. (2020). Klasifikasi Alfabet Bahasa Isyarat Indonesia (BISINDO) Dengan Metode Template Matching dan K-Nearest Neighbors (KNN). *Seminar Nasional Mahasiswa Ilmu Komputer Dan Aplikasinya (SENAMIKA)*, 1(2), 747–760. <https://conference.upnvj.ac.id/index.php/senamika/article/view/563>
- Suwirmayanti, N. L. G. P., Rahardja, C. A., Juardi, T., Agung, H., Bachtiar, A. M., Rivki, M., Permana Putra1), A. M. H. P. 2), Siswan Syahputra 3), Novita, S., Harsani, P., & Qur'ania, A. (2017). Analisis Metode K-Nearest Neighbour (Knn) Dalam Klasifikasi. *Jurnal Buana Informatika*, 10(1), 118–125.

- Wijaya, N., Endah, M., Feliati, M., Studi, P., Program, I., & Korespondensi, P. (2020). *Penerapan Algoritma Decision Tree C.45 Untuk Klasifikasi Data Status Huni Rumah Rehabilitasi Pasca Erupsi Merapi*. 424–430.
- Wildani, A., Maryanto, S., & Susilo, A. (2015). Analisis Sinyal Seismik Tremor Harmonik dan Tremor Spasmodik Gunungapi Semeru, Jawa Timur – Indonesia. *Wacana Didaktika*, 3(2), 140–150.
<https://doi.org/10.31102/wacanadidaktika.3.2.140-150>