



## IDENTIFIKASI GERAKAN TANGAN PADA SANDI *SEMAPHORE* PRAMUKA SECARA *REALTIME* MENGGUNAKAN *DECISION TREE*

Arya Sukma Putra Dwika<sup>1</sup>, Asrul Abdullah<sup>2</sup>, Syarifah Putri Agustini Alkadri<sup>3</sup>

<sup>1, 2, 3</sup>Fakultas Teknik dan Ilmu Komputer, Universitas Muhammadiyah Pontianak

email: [arya.sukma.420@gmail.com](mailto:arya.sukma.420@gmail.com)<sup>1</sup>, [asrul.abdullah@unmuhpnk.ac.id](mailto:asrul.abdullah@unmuhpnk.ac.id)<sup>2</sup>, [agustini.putri@unmuhpnk.ac.id](mailto:agustini.putri@unmuhpnk.ac.id)<sup>3</sup>

Informasi Artikel	ABSTRACT
<p><b>Riwayat artikel :</b>            Disubmit : 1 Desember 2024            Direvisi : 4 Desember 2024            Diterima : 10 Desember 2024            Dipublikasi : 20 Desember 2024</p>	<p><i>This research aims to accurately and real-time identify hand gestures in semaphore, especially for Scouts who are new to this skill. Proper identification can minimise errors that could potentially result in inaccurate information received, which in turn can affect the safety and effectiveness of communication. The Decision Tree method is applied to identify hand gestures as a contribution to improving the communication skills of Scouts. Test results show that the model is able to recognise letter classes in semaphore ciphers with a high level of accuracy under normal lighting conditions, which is 94%. However, under low lighting conditions, the performance of the model decreased, with an accuracy of 74% in the first test with 20 recognised classes, and 66% in the second test with 18 recognised classes. Model evaluation was conducted using confusion matrix to measure the accuracy, recall, and precision of the training process using Decision Tree. These results show that the Decision Tree method has the potential to assist Scouts in identifying semaphore ciphers effectively, especially under adequate lighting conditions.</i></p>
<p><b>Keywords:</b>  <i>Semaphore cipher identification, Decision Tree, Scouting, Hand gesture accuracy, Confusion matrix</i></p>	
	ABSTRAK
<p><b>Kata kunci:</b>            Identifikasi sandi <i>semaphore</i>, <i>Decision Tree</i>, Pramuka, Akurasi gerakan tangan, <i>Confusion matrix</i></p>	<p>Penelitian ini bertujuan untuk mengidentifikasi gerakan tangan pada sandi <i>semaphore</i> secara akurat dan <i>real-time</i>, khususnya bagi anggota Pramuka yang baru mempelajari keterampilan ini. Identifikasi yang tepat dapat meminimalkan kesalahan yang berpotensi mengakibatkan informasi yang diterima menjadi tidak akurat, yang pada gilirannya dapat mempengaruhi keselamatan dan efektivitas komunikasi. Metode <i>Decision Tree</i> diterapkan untuk mengidentifikasi gerakan tangan sebagai kontribusi dalam meningkatkan kemampuan komunikasi anggota Pramuka. Hasil pengujian menunjukkan bahwa model ini mampu mengenali kelas huruf pada sandi <i>semaphore</i> dengan tingkat akurasi yang tinggi dalam kondisi pencahayaan normal, yaitu sebesar 94%. Namun, pada kondisi pencahayaan yang rendah, performa model menurun, dengan akurasi sebesar 74% pada uji pertama dengan 20 kelas yang dikenali, dan 66% pada uji kedua dengan 18 kelas yang dikenali. Evaluasi model dilakukan menggunakan <i>confusion matrix</i> untuk mengukur tingkat akurasi, <i>recall</i>, dan <i>precision</i> pada proses pelatihan menggunakan <i>decision tree</i>. Hasil ini menunjukkan bahwa metode <i>decision tree</i> memiliki potensi untuk membantu anggota Pramuka dalam mengidentifikasi sandi <i>semaphore</i> secara efektif, terutama dalam kondisi pencahayaan yang memadai.</p>





## PENDAHULUAN

Gerakan Pramuka Indonesia adalah sebuah organisasi pendidikan non formal yang mengadakan pendidikan kependuan di Indonesia. Pramuka merupakan singkatan dari Praja Muda Karana, yang berarti “Rakyat Muda yang Senang Berkarya” (Anzari & Puriza, 2022). Salah satu keterampilan yang diajarkan dalam Pramuka adalah sandi *semaphore* yang menggunakan gerakan tangan sebagai salah satu komponen utama dalam komunikasi visual. Penyandian Pramuka seperti *semaphore* adalah model bentuk komunikasi digital awal yang disampaikan atas dasar kerahasiaan pesan yang harus dipelajari oleh anggota Pramuka sebagai Syarat Kecakapan Umum. Sandi *semaphore* merupakan suatu cara untuk mengirim dan menerima berita dengan menggunakan bendera, dayung, batang, tangan kosong atau dengan sarung tangan. Informasi yang didapat, dibaca melalui posisi bendera atau tangan tetapi saat ini yang umumnya digunakan ialah bendera, yang dinamakan bendera *semaphore* (Juliatmojo & Aribowo, 2013).

Mengidentifikasi gerakan tangan pada sandi *semaphore* secara akurat dan *realtime* merupakan tantangan yang harus dihadapi, terutama bagi anggota Pramuka yang baru belajar keterampilan ini. Kesalahan dalam mengidentifikasi gerakan tangan dapat mengakibatkan informasi yang salah atau tidak sesuai, yang pada akhirnya dapat mempengaruhi keselamatan dan efektivitas komunikasi. Oleh karena itu, penting untuk mengembangkan pengujian model yang mampu mengidentifikasi gerakan tangan pada sandi *semaphore* secara *realtime* guna meningkatkan kemampuan komunikasi dan keamanan Pramuka.

Dalam situasi ini, teknik pengenalan pola dan pembelajaran mesin, terutama *Decision Tree*, memiliki potensi yang signifikan dalam membantu mengidentifikasi gerakan tangan dalam sandi *semaphore* Pramuka. *Decision Tree* adalah metode yang terkenal dalam pembelajaran mesin untuk pengambilan keputusan (Aradea et al., 2011). Penggunaan *Decision Tree* dalam mengidentifikasi gerakan tangan dapat memberikan kontribusi yang signifikan dalam memungkinkan para anggota Pramuka untuk berkomunikasi dengan lebih efektif.

Pengenalan pola ini menggunakan OpenCV dan MediaPipe dengan bahasa pemrograman Python untuk mengenali pola dari gerakan tangan pada sandi *semaphore*. Penggunaan OpenCV dan MediaPipe yang digabungkan dengan *machine learning* untuk mengolah dan melatih sehingga dapat menirukan sifat manusia dengan baik. Penggunaan *machine learning* dapat mempermudah dalam mengidentifikasi gerakan tangan pada sandi *semaphore* tersebut.

Penelitian sebelumnya berjudul “Penerapan Metode *Ekstraksi Wavelet Haar* Untuk Deteksi Gerakan Sandi *Semaphore* Menggunakan Stick Lalu Lintas” yang menunjukkan sistem deteksi gerakan sandi *semaphore* ini telah berjalan dengan baik. Sistem dapat mendeteksi gerakan peraga





pada variasi jarak 10 meter, 15 meter, dan 20 meter dengan persentasi keberhasilan kumulatif yang mencapai 85% (Hatta et al., 2020).

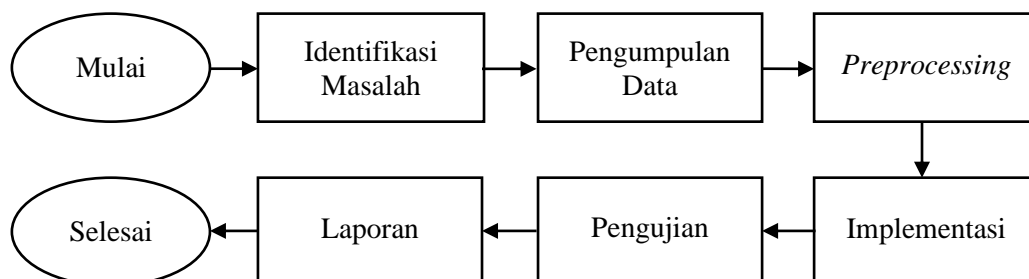
Penelitian selanjutnya berjudul “Media Pembelajaran Sandi *Semaphore* Menggunakan Teknologi *Augmented Reality* Berbasis Android” yang hasil penelitiannya adalah pendeteksian marker berjalan cukup baik namun perlu diperhatikan jarak dan fokus kamera, semakin dekat kamera dengan gambar atau marker semakin baik, marker tidak boleh tertutup lebih dari 80%, tingkat kemiringan kamera pada pendeteksian marker semakin tinggi semakin baik, ketika intensitas cahaya rendah atau terlalu tinggi proses pendeteksian marker semakin melambat (Ahsanandi & Awaludin, 2022).

Penelitian lainnya berjudul “Identifikasi Pola Sidik Bibir Pada Identitas Manusia Menggunakan Metode *Histogram Of Oriented Gradients* (Hog) Dengan Klasifikasi *Decision Tree* Untuk Aplikasi Bidang Forensik Biometrik” dengan hasil penelitian adalah hasil 7 eksperimen menunjukkan bahwa model *Decision Tree* dengan integrasi SMOTE dapat memberikan akurasi yang baik dalam mengklasifikasikan data kecelakaan lalu lintas (Reichenbach et al., 2019).

Berdasarkan uraian di atas, penelitian ini bertujuan untuk mengembangkan pengujian yang dapat mengidentifikasi gerakan tangan dalam sandi *semaphore* pramuka secara *realtime* menggunakan *Decision Tree*. Dengan pengujian model ini, diharapkan anggota Pramuka bisa berkomunikasi dengan lebih efektif, mengurangi resiko kesalahan komunikasi, serta mempercepat proses pembelajaran sandi *semaphore*.

## METODE PENELITIAN

Metode penelitian ini diilustrasikan melalui diagram alir yang menjelaskan langkah-langkah utama yang dapat dilihat pada Gambar 1.



Gambar 1. Diagram Alir



### 1. Identifikasi Masalah

Identifikasi masalah merupakan langkah awal dalam suatu penelitian yang bertujuan untuk mendefinisikan dan menjelaskan masalah yang ingin diselesaikan dengan menggunakan metode penelitian yang tepat.

### 2. Pengumpulan Data

Pada tahap ini diperlukan dataset untuk memenuhi kebutuhan dalam mengidentifikasi dan perhitungan akurasi. Dataset terdapat “A-Z” dan “SPASI” setiap kelas akan mempunyai data sebanyak 500 perkelas. Dataset akan diambil dengan cara merekam titik kordinat *keypoint* tubuh dari kamera *webcam* secara *realtime*, hasil *keypoint* akan disimpan ke bentuk *file Comma Separated Values* (CSV).

### 3. Preprocessing

*Preprocessing* merupakan tahap awal dalam pengolahan data atau analisis data. Dalam pra-premrosesan terdapat beberapa tahap diantaranya:

- 1) Membuat grup hasil dari perekaman data, berupa kelas yang direkam dan jumlah sampel data kelas yang di ambil. Data kelas memiliki kelas 27 kelas, dari kelas “A-Z” dan “SPASI”. Jumlah sampel data yang diambil sebanyak 500 data, setiap kelasnya jadi jumlahnya 27 kelas memiliki sampel data sebanyak 13.500 sampel data.
- 2) Membagi data menjadi data latih dan data uji, dengan mengatur proposisi yang akan di alokasikan pengujian sebesar 30%, lalu data pelatihan 70%.

### 4. Implementasi

Implementasi menggunakan metode *Decision Tree* merupakan proses *meachine learning* untuk dapat menyesuaikan hasil dari dataset. Implementasi bertujuan sebagai pengelompokan dataset menjadi kelas-kelas yang sesuai, klasifikasi juga dapat di artikan sebagai pembelajaran mesin. Implementasi ini akan mempelajari dan mengelompokkan kelas dari hasil dataset yang berbentuk file (CSV), maka metode ini akan mengembangkan *Decision Tree* berdasarkan pemilihan atribut acak pada setiap data untuk menentukan Implementasi.

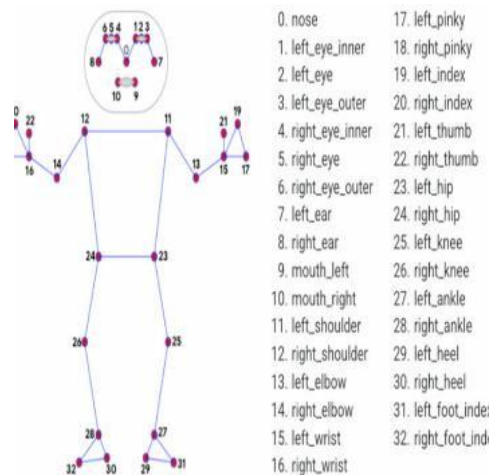
Proses perekaman secara *real-time* dimulai dengan merekam menggunakan kamera dan memprosesnya dengan MediaPipe untuk mendeteksi dan memperkirakan pose gerakan tangan, dengan mengenali *landmark* pada setiap frame. Hasilnya kemudian diproses lebih lanjut menggunakan OpenCV untuk menandai *landmark* atau garis penting tangan. Langkah terakhir melibatkan penggunaan *Decision Tree* untuk mengklasifikasikan pose yang terdeteksi menjadi gerakan tertentu. Kombinasi MediaPipe, OpenCV, dan *Decision Tree* memungkinkan sistem untuk merespons gerakan tangan secara *real-time* dengan tepat dan efisien.





*Open Computer Vision* (OpenCV) merupakan *library open source* yang tujuannya dikhususkan untuk melakukan pengolahan citra agar komputer mempunyai kemampuan yang mirip dengan cara pengolahan visual pada manusia. OpenCV telah menyediakan banyak algoritma visi komputer dasar serta menyediakan modul pendeteksian objek yang menggunakan metode *computer vision* (Zulkhaidi et al., 2019). Sedangkan *MediaPipe* dirancang untuk menerapkan kecerdasan buatan pada aplikasi yang akan dibangun (Convolutional, n.d.). *Framework* ini dapat digunakan untuk membangun model *machine learning* seperti *gesture recognition*.

*MediaPipe Pose Landmark Detection Guide* berfungsi sebagai pengenalan gerakan tubuh manusia secara *real-time*, setiap gerakan memungkinkan akan dapat dikenali dengan memberikan hasil dan akurasi secara *real-time* (Remiro et al., 2023). *Pose Landmark Detection Guide* ini menghasilkan objek yang dapat mendeteksi gerakan untuk setiap pengenalan. Objek hasil tersebut terdiri dari tiga output yang berisi *gestures*, koordinat yang dinormalisasi (*Landmark*) dan koordinat dunia (*WorldLandmark*). *Pose Landmark Detection Guide* memiliki model *landmark* tubuh manusia sebagai media untuk mendeteksi tubuh dengan titik kordinat, terdapat 33 *keypoint body detection* dapat dilihat pada Gambar 2.



**Gambar 2.** Index Keypoint pada *MediaPipe*

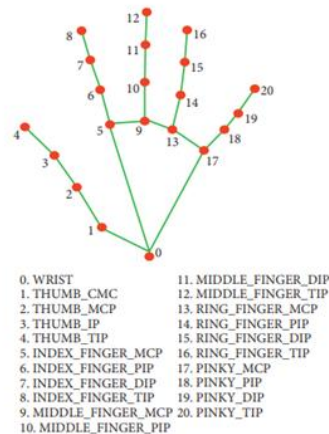
(sumber: *MediaPipe*)

*MediaPipe Gesture Recognition* berfungsi sebagai pengenalan gerakan tangan manusia secara *real-time*. *Gesture Recognizer* ini menghasilkan objek yang dapat mendeteksi gerakan untuk setiap pengenalan. Objek hasil tersebut terdiri dari empat komponen, setiap komponen berbentuk array. Dalam *MediaPipe Gesture Recognition* memiliki model *landmark* tangan manusia sebagai media





untuk mendeteksi tubuh dengan titik kordinat, terdapat 21 *keypoint hand landmark* dapat dilihat pada Gambar 3.



**Gambar 3.** *Keypoint Hand MediaPipe*

(sumber: towardsdatascience.com)

Pada tahap penerapan *Decision Tree* memerlukan *import library* berguna untuk menerapkan algoritma *Decision Tree* dalam proses pembelajaran mesin.

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
```

Setelah *import library Decision Tree* selanjutnya input kode standarisasi pada data dan sebagai penerapan *Decision Tree*.

```
pipelines = {
    'dt': make_pipeline(StandardScaler(), DecisionTreeClassifier()),
}
```

Setelah standarisasi dan penerapan *Decision Tree*, langkah selanjutnya membuat dan melatih model *machine learning* lalu membuat sebuah *pipeline* yang berisi model *Decision Tree*. Model *Decision Tree* tersebut akan dilatih menggunakan data pelatihan ( $X_{train}$  dan  $y_{train}$ ) dengan kriteria gini sebagai metode splitting, maksimum kedalaman pohon adalah 5, dan *random state* adalah 42. Kemudian, model *Decision Tree* tersebut akan dievaluasi menggunakan data uji ( $X_{test}$  dan  $y_{test}$ ) untuk menghitung akurasi.





```

fit_models = { }

for algo, pipeline in pipelines.items():
    model = pipeline.fit(X_train, y_train)
    fit_models[algo] = model

# Membuat pipeline untuk model Decision Tree
dt_pipeline = Pipeline([
    ('dt', DecisionTreeClassifier(criterion='gini', max_depth=5, random_state=42)) # Model Decision Tree
])

# Melatih model Decision Tree dengan pipeline
dt_pipeline.fit(X_train, y_train)

# Evaluasi model
accuracy = dt_pipeline.score(X_test, y_test)
print(f'Akurasi model Decision Tree: {accuracy}')

# fit_models merupakan kamus yang berisi model-model yang telah dilatih untuk setiap algoritma
menggunakan data pelatihan.

fit_models

# Menggunakan model Decision Tree dari fit_models untuk membuat prediksi menggunakan data uji
(X_test).

dt_predictions = fit_models['dt'].predict(X_test)

# Menggunakan model Decision Tree yang telah dilatih menggunakan pipeline untuk membuat
prediksi menggunakan data uji (X_test).

dt_predictions = dt_pipeline.predict(X_test)

```





```
# Mencetak hasil prediksi dari model Decision Tree yang menggunakan pipeline ke layar.

print(dt_predictions)
```

**5. Pengujian**

Proses pengujian ini dilakukan untuk melihat identifikasi dari citra (Hasna et al., 2022). Pengujian dilakukan untuk menentukan keberhasilan dalam mengidentifikasi dan tingkat akurasi terhadap dataset yang diolah. Pengujian ini untuk memastikan bahwa seluruh kebutuhan identifikasi telah terpenuhi setelah pengujian. Pengujian akan dilakukan dengan mengambil hasil pengolahan data yang berbentuk *file pickle* (PKL), dan memunculkan kamera yang memberikan hasil prediksi.

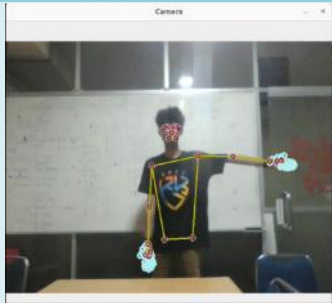

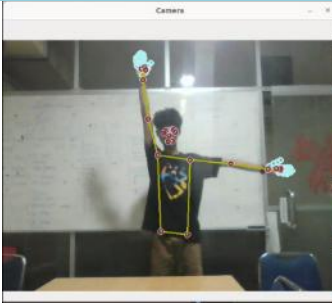

Pengujian akan dilakukan dengan, *confusion matrix* untuk mengetahui hasil *accuracy*, *Recall*, dan *Precision*. Pengujian juga akan dilakukan dengan menguji jarak efisiensi dan pencahayaan dari model, pencahayaan yang ditentukan pencahayaan normal dan kurang pencahayaan.

**HASIL DAN PEMBAHASAN**

Hasil dan pengujian penelitian tentang pengujian model Identifikasi Gerakan Tangan pada Sandi *Semaphore* Pramuka Secara *Realtime* Menggunakan *Decision Tree*. Pengujian pada penelitian ini menggunakan *confusion matrix* dan pengujian pada pencahayaan.

**1. Hasil Perekaman Data**

**Tabel 1.** Hasil Perekaman Data

No	Kelas	Tampilan Hasil Perekaman	No	Kelas	Tampilan Hasil Perekaman
1	B		2	D	
3	P		4	R	





Dataset yang diperoleh untuk penelitian ini, merupakan hasil dari perekaman video secara *realtime*, dataset ini memiliki jumlah kelas sebanyak 27 kelas, dengan total data 1 kelas berjumlah 500 data dan jumlah keseluruhan data yang diperoleh sebanyak 13.500 data.

Dataset yang diperoleh sebanyak 13.500 data, dibagi menjadi data latih dan data uji dengan pembagian 70% data latih, 30% data uji. Hasil dari pembagian data ini diperoleh data latih sebanyak 6.750 data, data uji sebanyak 6.750 data.

**Tabel 2.** Hasil Pembagian Data

Pembagian Data	Jumlah Data
Jumlah Data Total	13.500
Jumlah Data Training Set	9.450
Jumlah Data Testing Set	4.050

Pada tabel di atas, menggambarkan pembagian data dalam dataset menjadi data latih dan data uji, serta jumlah total data yang digunakan dalam penelitian.

## 2. Hasil Pengujian *Confusion matrix*

Proses pengujian yang dilakukan, dengan dataset yang telah di siapkan pada model untuk melihat tingkat *accuracy*, *Recall*, dan *Precision* beserta melihat tingkat keberhasilan dalam mengklasifikasi huruf pada sandi *semaphore* (Krstinić et al., 2020). Keberhasilan dalam mengklasifikasi huruf pada sandi *semaphore* menjadi indikator dalam menentukan model yang dibangun ini berhasil.

Hasil pengujian ini merupakan pengujian dengan pecahayaan normal dan kurang pencahayaan, hasil dari pengujian ini dapat di lihat pada Tabel 3.

**Tabel 3.** Hasil Klasifikasi Huruf pada Sandi *Semaphore*

No	Huruf	Pencahayaan Normal		Kurang Pencahayaan	
		Uji 1	Uji 2	Uji 1	Uji 2
1	A	T	T	T	T
2	B	T	T	T	T
3	C	T	T	TT	T
4	D	TT	T	TT	TT
5	E	T	T	TT	T
6	F	T	T	T	TT
7	G	T	T	T	T





8	H	T	T	T	T
9	I	T	T	T	T
10	J	T	T	T	TT
11	K	T	TT	T	T
12	L	T	T	TT	TT
13	M	T	T	TT	TT
14	N	T	T	T	T
15	O	T	T	T	T
16	P	T	T	TT	T
17	Q	T	T	T	T
18	R	T	T	TT	TT
19	S	T	T	T	TT
20	T	T	T	T	T
21	U	TT	T	T	TT
22	V	T	T	T	TT
23	W	T	T	T	T
24	X	T	T	T	T
25	Y	T	T	T	T
26	Z	T	T	T	T
27	SPASI	T	T	T	T
<b>Total Berhasil</b>		25	26	20	18

Keterangan: T = Terdeteksi

TT = Tidak Terdeteksi

Pada Tabel 3, menjelaskan hasil dari klasifikasi untuk setiap huruf dalam gerakan sandi *semaphore* berdasarkan dua kondisi pencahayaan yaitu pencahayaan normal dan kurang pencahayaan. Huruf A memiliki hasil klasifikasi yang sama baik pada pencahayaan normal maupun kurang pencahayaan sedangkan Huruf J memiliki hasil klasifikasi yang berbeda antara pencahayaan normal dan kurang pencahayaan. Kesimpulannya, tabel tersebut menunjukkan performa model dalam mengklasifikasikan gerakan sandi *semaphore* berdasarkan kondisi pencahayaan yang berbeda.

Hasil dari klasifikasi huruf pada sandi *semaphore* pencahayaan normal dan kurang pencahayaan, menunjukan perbandingan keberhasilan model dalam mengklasifikasi. Untuk menghitung hasil *accuracy* dari pengujian pencahayaan normal dan kurang pencahayaan, dengan penggunaan rumus:





Accuracy uji 1 pecahayaan normal :  $\frac{25}{27} \times 100\% = 92\%$

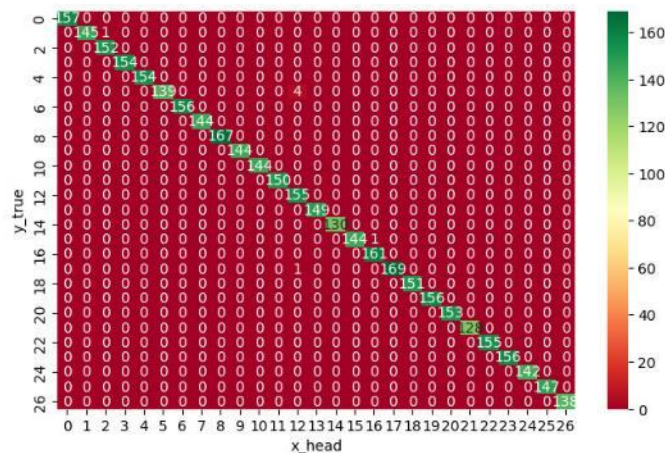
Accuracy uji 2 pecahayaan normal :  $\frac{26}{27} \times 100\% = 96\%$

Accuracy uji 1 kurang pencahayaan :  $\frac{20}{27} \times 100\% = 74\%$

Accuracy uji 2 kurang pencahayaan :  $\frac{18}{27} \times 100\% = 66\%$

Berdasarkan hasil pengujian, model ini dapat mengenali kelas huruf pada sandi *semaphore* dengan pencahayaan normal. Hal ini dibuktikan dengan tingkat akurasi yang lebih tinggi pada pencahayaan normal. Pada uji pertama, model ini menghasilkan akurasi 92% dengan 25 kelas yang dikenali. Pada uji kedua, model ini menghasilkan akurasi 96% dengan 26 kelas yang dikenali. Sehingga, rata-rata akurasi model ini pada cahaya normal adalah 94%.

Pengujian *confusion matrix*, pengujian ini menggunakan dataset hasil dari pelatihan model random forest. Dalam pengujian ini, menghasilkan tingkat *accuracy*, *Recall*, dan *Precision* untuk mengetahui seberapa baik dalam pelatihan model menggunakan *Decision Tree*.



**Gambar 4.** Heatmap *Confusion matrix*

Gambar di atas merupakan plot dua dimensi dengan sumbu x dan y, dan skala warna di sisi kanan. Sumbu x dilabeli sebagai “x\_head” dan berkisar dari 0 hingga 26, sedangkan Sumbu y dilabeli sebagai “y\_true” dan berkisar dari 0 hingga 26. Plot ini memiliki latar belakang berwarna yang berkisar dari merah (nilai rendah) hingga hijau (nilai tinggi), sesuai dengan skala warna di sisi kanan gambar. Ada beberapa nilai numerik tertulis langsung pada plot, yang mewakili data tertentu dalam kumpulan data. Nilai-nilai ini diberi warna putih untuk kontras dengan latar belakang berwarna.

Dari visualisasi pada Gambar 2, didapatkan hasil dari *confusion matrix* yang ditampilkan pada Tabel 4.



Tabel 4. Tabel Hasil *Confusion matrix*

No	<i>Rf</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Support</i>
1	A	1.00	1.00	1.00	358
2	B	1.00	1.00	1.00	350
3	C	0.99	0.99	0.99	257
4	D	1.00	0.99	0.99	348
5	E	0.99	0.99	0.99	353
6	F	0.99	0.99	0.99	354
7	G	1.00	1.00	1.00	344
8	H	1.00	1.00	1.00	369
9	I	1.00	1.00	1.00	354
10	J	1.00	0.94	0.97	356
11	K	1.00	1.00	1.00	352
12	L	0.99	0.99	0.99	350
13	M	0.99	0.98	0.98	343
14	N	0.97	1.00	0.98	332
15	O	1.00	1.00	1.00	351
16	P	1.00	1.00	1.00	361
17	Q	1.00	1.00	1.00	359
18	R	1.00	0.98	0.99	352
19	S	1.00	1.00	1.00	353
20	T	1.00	1.00	1.00	337
21	U	1.00	1.00	1.00	344
22	V	0.94	1.00	0.97	335
23	W	1.00	1.00	1.00	348
24	X	1.00	1.00	1.00	356
25	Y	1.00	1.00	1.00	326
26	Z	1.00	1.00	1.00	342
27	SPASI	1.00	1.00	1.00	342
<i>Accuracy</i>					0.99
<i>Recall</i>					0.99
<i>Precision</i>					0.99





Tabel 4 merupakan hasil *confusion matrix* untuk setiap huruf dalam gerakan sandi *semaphore* sebagai berikut:

1. Huruf (*Letter*): Mempresentasikan huruf-huruf yang diuji dalam gerakan sandi *semaphore*.
2. *Precision*: Menunjukkan seberapa akurat model dalam mengklasifikasikan positif (benar) dari semua prediksi positif. Nilai *Precision* berkisar antara 0 hingga 1, dengan 1 menunjukkan presisi sempurna.
3. *Recall*: Dapat dikenal sebagai *sensitivity* atau *true positive rate*, menunjukkan seberapa baik model dalam mengidentifikasi semua instance positif yang sebenarnya. Nilai *Recall* juga berkisar antara 0 hingga 1, dengan 1 menunjukkan *Recall* sempurna.
4. *F1-Score*: Menggabungkan *Precision* dan *Recall* untuk memberikan nilai yang seimbang antara keduanya. *F1-Score* adalah *harmonic mean* dari *Precision* dan *Recall*.
5. *Support*: Jumlah *instance* (sampel) yang mewakili setiap huruf dalam data uji.

Pada Tabel tersebut dijelaskan bahwa percobaan huruf A memiliki *Precision*, *Recall*, dan *F1-Score* sebesar 1.00, yang menunjukkan hasil yang sangat baik dalam mengklasifikasikan huruf A. Pada percobaan huruf J memiliki *Recall* yang sedikit lebih rendah (0.94), yang berarti model mungkin melewatkan beberapa *instance* huruf J yang sebenarnya positif.

*Accuracy* mengukur seberapa sering model memberikan prediksi yang benar, rumus perhitungan *accuracy* dapat dilihat pada persamaan:

$$Accuracy = \frac{4040}{4047} \times 100\% = 99\%$$

*Precision* mengukur seberapa sering prediksi positif model yang benar, rumus perhitungan *precision* dapat dilihat pada persamaan:

$$Precision = \frac{4040}{4040+6} \times 100\% = 99\%$$

*Recall* mengukur seberapa sering model dapat menemukan kelas positif, rumus perhitungan *recall* dapat dilihat pada persamaan:

$$Recall = \frac{4040}{4040+1} \times 100\% = 99\%$$

Dalam penelitian ini, algoritma *Decision Tree* diaplikasikan untuk mengidentifikasi gerakan tangan pada sandi *semaphore* secara *real-time*, terutama bagi anggota Pramuka. Keandalan algoritma ini dalam mengklasifikasikan huruf-huruf pada sandi *semaphore* terbukti efektif, dengan akurasi rata-rata 94% pada kondisi pencahayaan normal, yang menurun hingga 74% dan 66% pada kondisi kurang





pencahayaan dalam dua uji berturut-turut. Hasil ini konsisten dengan temuan Rouabeh et al. (2014), yang menunjukkan bahwa *Decision Tree* menawarkan akurasi tinggi dengan waktu komputasi rendah pada pengenalan citra, seperti dalam klasifikasi gambar jalan raya. Dalam konteks *real-time*, hasil ini menjadi signifikan karena pencapaian akurasi yang tinggi memperlihatkan kemampuan algoritma ini dalam menangani proses klasifikasi cepat yang diperlukan dalam aplikasi *semaphore* (Rouabeh et al., 2014). Guo et al. (2015) juga menyoroti keefektifan *Decision Tree* dalam pengklasifikasian gambar pada remote sensing, di mana kemampuan adaptasi terhadap data multi-temporal meningkatkan akurasi klasifikasi. Hal ini relevan dengan penelitian ini yang menggunakan berbagai data visual untuk meningkatkan kinerja klasifikasi gerakan tangan *semaphore* dalam kondisi pencahayaan berbeda.

Selain itu, penelitian ini menggunakan *confusion matrix* untuk mengevaluasi nilai akurasi, *precision*, dan *recall* dalam mengukur performa model. Hasil menunjukkan bahwa huruf A memiliki *precision* dan *recall* sempurna (1.00), sedangkan huruf J menunjukkan *recall* yang sedikit lebih rendah pada kondisi kurang pencahayaan, yakni 0.94, menunjukkan adanya keterbatasan dalam menangkap instance positif pada kondisi ini. Naswin (2023) menekankan bahwa *Decision Tree* mampu meningkatkan akurasi melalui integrasi teknik praproses, terutama dalam mengklasifikasikan gambar medis yang kompleks seperti deteksi pneumonia dari citra X-ray. Hasil ini memperkuat gagasan bahwa *Decision Tree* dapat meningkatkan keakuratan dalam klasifikasi citra yang kompleks, termasuk gerakan tangan *semaphore* dalam kondisi pencahayaan variatif. Interpretabilitas algoritma *Decision Tree* juga merupakan salah satu keunggulannya, sebagaimana diutarakan oleh Surekha (2017), yang menyatakan bahwa algoritma ini memberikan jalur keputusan yang jelas, memungkinkan model ini untuk lebih dapat diandalkan dalam konteks aplikasi Pramuka di mana komunikasi yang tepat sangatlah krusial.

Lebih lanjut, penelitian ini menunjukkan bahwa performa model menurun dalam kondisi kurang pencahayaan. Norouzi et al. (2015) membahas bahwa algoritma *Decision Tree* mampu memfasilitasi analisis *real-time* melalui efisiensi komputasi yang tinggi, terutama dalam klasifikasi gerakan manusia. Namun, pada penelitian ini, tantangan dalam pencahayaan rendah menurunkan akurasi, menunjukkan perlunya metode pendukung untuk meningkatkan ketahanan model dalam kondisi lingkungan yang berbeda. Penelitian ini sejalan dengan pendekatan hybrid yang dibahas oleh Battula et al. (2016), yang mengusulkan penggabungan *Decision Tree* dengan *wavelet transform* untuk menangani tantangan pada dataset yang tidak seimbang dan kondisi pengujian yang sulit. Strategi ini dapat diterapkan pada penelitian ini untuk mempertahankan akurasi yang lebih stabil di berbagai kondisi pencahayaan.





Secara keseluruhan, hasil penelitian ini menunjukkan potensi besar dari algoritma *Decision Tree* dalam aplikasi identifikasi gerakan tangan pada sandi *semaphore* secara *real-time*. Tingkat akurasi yang dicapai pada kondisi pencahayaan normal dan pencahayaan rendah memberikan gambaran tentang kelebihan dan keterbatasan *Decision Tree*. Seperti yang diungkapkan oleh berbagai literatur, algoritma ini memberikan keunggulan dalam hal efisiensi komputasi dan keakuratan dalam klasifikasi gambar yang kompleks. Penggunaan *confusion matrix* dalam penelitian ini memberikan evaluasi menyeluruh atas performa model, serta menunjukkan area untuk pengembangan lebih lanjut, seperti meningkatkan robustitas model terhadap kondisi pencahayaan yang bervariasi. Hasil penelitian ini memperkuat literatur yang ada dan memberikan kontribusi penting bagi pengembangan metode komunikasi *semaphore* yang lebih efisien dan efektif bagi anggota Pramuka.

## SIMPULAN

Berdasarkan hasil pengujian klasifikasi huruf pada sandi *semaphore* dengan kondisi pencahayaan normal dan kurang pencahayaan, terdapat perbandingan keberhasilan model dalam mengenali kelas huruf. Ditemukan bahwa model memiliki tingkat akurasi yang lebih tinggi pada kondisi pencahayaan normal dibandingkan dengan kondisi kurang pencahayaan. Pada uji pertama dengan pencahayaan normal, model mencapai akurasi sebesar 92% dengan mengenali 25 kelas. Pada uji kedua, model mencapai akurasi sebesar 96% dengan mengenali 26 kelas. Rata-rata akurasi model pada kondisi pencahayaan normal adalah 94%.

Namun, pada kondisi kurang pencahayaan, model menunjukkan performa yang lebih rendah. Pada uji pertama, model mencapai akurasi sebesar 74% dengan mengenali 20 kelas, sedangkan pada uji kedua, akurasi turun menjadi 66% dengan mengenali 18 kelas. Dalam pengujian *confusion matrix*, digunakan untuk mengevaluasi tingkat akurasi, *Recall*, dan *Precision* dalam pelatihan model menggunakan *Decision Tree*. Hasil pengujian ini memberikan gambaran tentang seberapa baik model dapat mengenali kelas huruf dalam berbagai kondisi.

## DAFTAR RUJUKAN

- Ahmad Naswin and Adityo Permana Wibowo (2023). Performance analysis of the decision tree classification algorithm on the pneumonia dataset. *International Journal of Artificial Intelligence in Medical Issues*, 1(1), 1-9. <https://doi.org/10.56705/ijaimi.v1i1.83>
- Ahsanandi, M. F., & Awaludin, L. (2022). Sistem Peringatan Tingkat Kerentanan Bangunan Berbasis Sensor IMU dengan Metode Fuzzy. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 12(1). <https://doi.org/10.22146/ijeis.70141>





- Anzari, Y., & Puriza, M. Y. (2022). Aplikasi dan Game Edukasi Sandi *Semaphore* Berbasis Multimedia. *EPSILON: Journal of Electrical Engineering and Information Technology*, 19(3). <https://doi.org/10.55893/epsilon.v19i3.67>
- Aradea, S. A., Z, A., & A, Y. (2011). Penerapan *Decision Tree* Untuk Penentuan Pola Data Penerimaan Mahasiswa Baru. *Jurnal Penelitian Sitrotika*, 7(1).
- Battula, B. P., Krishna, K., Bhattacharyya, D., & Kim, T. (2016). A novel decision tree framework using discrete haar wavelet transform. *International Journal of U- And E- Service, Science and Technology*, 9(1), 63-72. <https://doi.org/10.14257/ijunesst.2016.9.1.07>
- Guo, K., Wang, J., & Wang, Y. (2015). Application Of Zy-3 Remote Sensing Image in The Research of Huashan Experimental Watershed. *Proceedings of the International Association of Hydrological Sciences*, 368, 51-56. <https://doi.org/10.5194/piahs-368-51-2015>
- Hasna, U., Siregar, A. C., & Octariadi, B. C. (2022). Klasifikasi Bentuk Daun Tanaman Begonia (Begoniaceae) Menggunakan Metode K-Nearest Neighbor. *Digital Intelligent*, 3(1), 57–64. <https://doi.org/10.29406/diligent.v3i1.4809>
- Hatta, Moch., Mulyono, N. V., & Ontoseno, R. D. H. (2020). Penerapan Metode Ekstraksi Wavelet Haar Untuk Deteksi Gerakan Sandi *Semaphore* Menggunakan Stick Lalu Lintas. *Teknika: Engineering and Sains Journal*, 4(1). <https://doi.org/10.51804/tesj.v4i1.787.13-18>
- Juliatmojo, T., & Aribowo, E. (2013). Pembelajaran Sandi Morse dan Sandi *Semaphore*. *Jurnal Sarjana Teknik Informatika*, 1(1).
- Krstinić, D., Braović, M., Šerić, L., & Božić-Štulić, D. (2020). *Multi-label Classifier Performance Evaluation with Confusion matrix*. 1–14. <https://doi.org/10.5121/csit.2020.100801>
- Norouzi, M., Collins, M. D., Johnson, M. S., Fleet, D. J., & Kohli, P. (2015). Efficient non-greedy optimization of decision trees.. <https://doi.org/10.48550/arxiv.1511.04056>
- Reichenbach, A., Bringmann, A., Reader, E. E., Pournaras, C. J., Rungger-Brändle, E., Riva, C. E., Hardarson, S. H., Stefansson, E., Yard, W. N., Newman, E. A., & Holmes, D. (2019). Identifikasi Pola Sidik Bibir Pada Identitas Manusia Menggunakan Metode Histogram of Oriented Gradients (HOG) Dengan Klasifikasi *Decision Tree* Untuk Aplikasi Bidang Forensik Biometrik. *Progress in Retinal and Eye Research*, 561(3).





- Remiro, M. Á., Gil-Martín, M., & San-Segundo, R. (2023). Improving Hand Pose Recognition Using Localization and Zoom Normalizations over MediaPipe Landmarks. *ECSA 2023*, 69. <https://doi.org/10.3390/ecsa-10-16215>
- Rouabeh, H., Abdelmoula, C., & Masmoudi, M. (2014). Performance evaluation of decision tree and neural network techniques for road scene image classification task. *International Image Processing, Applications and Systems Conference*. <https://doi.org/10.1109/ipas.2014.7043274>
- Rouabeh, H., Abdelmoula, C., & Masmoudi, M. (2014). Performance evaluation of decision tree and neural network techniques for road scene image classification task. *International Image Processing, Applications and Systems Conference*. <https://doi.org/10.1109/ipas.2014.7043274>
- Surekha, S. (2017). A Survey of Various Tree Based Classification Techniques. *International Journal of Advanced Research in Computer Science and Software Engineering*, 7(3), 59-65. <https://doi.org/10.23956/ijarsse/v7i3/0105>
- Zulkhaidi, T. C. A.-S., Maria, E., Lunak, P. S. T. R. P., & Samarinda, P. P. N. (2019). Pengenalan Pola Bentuk Wajah dengan OpenCV. *JURTI*, 3(2).

