BISINDO Sign Letters Recognition Through HOG Features and Bagging Decision Tree

Dolly Indra[1*], Erick Irawadi Alwi[2], Faudiah Anwar[3], St. Nadya Kurnia Prihandani[4]

[1,2,3,4] *Faculty of Computer Science, Universitas Muslim Indonesia, Urip Sumoharjo Street, Km 5, Makassar, Indonesia*

E-mail: [1]dolly.indra@umi.ac.id, [2]erick.alwi@umi.ac.id, [3]faudiahanwar@gmail.com, [4]st.nady4@gmail.com

**Corresponding Author*

Abstrak— **Bahasa Isyarat merupakan salah satu media komunikasi utama bagi penyandang disabilitas rungu. BISINDO (Bahasa isyarat Indonesia) dalam berkomunikasi salah satunya dengan menggunakan gerakan tangan. Salah satu solusi untuk mengatasi masalah ini adalah menggunakan pengolahan citra untuk mengenali huruf BISINDO A-Z berdasarkan gerakan tangannya. Penelitian ini bertujuan membuat sistem pengenalan huruf BISINDO berbasis pengolahan citra dengan menggunakan beberapa tahapan yaitu preprosesing seperti melakukan konversi citra RGB ke citra Grayscale, kemudian kualitas citra ditingkatkan dengan menyesuaikan kontras citra dan menghapus noise dengan median filter, ekstraksi fitur HOG (Histogram of Oriented Gradients) dan klasifikasi Bagging Decision Tree. Jumlah dataset yang digunakan sebanyak 156 citra , dimana 104 citra huruf untuk data latih dan 52 citra huruf untuk data uji. Data tersebut akan di proses dalam sistem sebagai data pelatihan yang kemudian dataset disimpan dalam format 'mat'. Berdasarkan hasil pengujian Klasifikasi menggunakan Bagging Decision Tree yang menghasilkan rata-rata tingkat akurasi mencapai 86.5%. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam upaya pengembangan teknologi pengenalan huruf BISINDO berbasis pengolahan citra digital.**

Kata Kunci — **BISINDO; Grayscale; Median Filter; HOG; Bagging Decision Tree.**

Abstract— **Sign language is one of the primary means of communication for people with hearing disabilities. BISINDO (Indonesian sign language) communicates using hand movements, among other things. One solution to this problem is to use image processing to recognize BISINDO letters A-Z based on hand movements. This study aims to create a BISINDO letter recognition system based on image processing using several stages, namely, preprocessing such as converting RGB images to grayscale images, then improving image quality by adjusting image contrast and removing noise with a median filter, HOG (Histogram of Oriented Gradients) feature extraction, and Bagging Decision Tree classification. A total of 156 images were used in the dataset, consisting of 104 letter images for training data and 52 letter images for test data. The data will be processed in the system as training data, and the dataset will then be stored in 'mat' format. Based on the results of testing Classification using Bagging Decision Tree, which produced an average accuracy rate of 86.5%. Thus, this research is expected to contribute to the development of BISINDO character recognition technology based on digital image processing.**

Keywords — BISINDO**; Grayscale; Median Filter; HOG; Bagging Decision Tree.**

I. INTRODUCTION

Sign language is one of the primary means of communication for people with hearing disabilities [1]. BISINDO uses hand movements, facial expressions, and body positions to convey messages [2], [3]. However, communication barriers still often occur because not all members of the general public understand sign language. Therefore, a system that can recognize BISINDO letters is needed to help bridge communication between the deaf community and the wider community.

Research developments in the field of image processing and artificial intelligence have made significant contributions to sign language recognition [4], [5]. Various machine learning-based approaches [6] have been used, ranging from manual feature extraction techniques combined with classical classification algorithms to the use of deep learning models [7]. Although deep learning algorithms such as Convolutional Neural Networks (CNN) have proven to provide good recognition performance, this approach generally requires a large amount of training data and high computational support. In contrast, manual feature extraction-based machine learning methods combined with appropriate classification algorithms can be a more efficient and lightweight alternative, especially when dealing with limited datasets, such as in the case of BISINDO letter recognition.

According to Goel et al (2025), AEGWO-Net, a method that combines machine learning with swarm intelligence to improve sign language recognition while reducing the problem of excessive feature dimensions, was proposed. Image features are extracted using HOG, reduced with an autoencoder, then optimized using the Improved Grey Wolf Optimizer, and classified with a simple neural network. Test results on six datasets show that AEGWO-Net is capable of achieving a high accuracy of 98.40% [8].

Research conducted by Farhana et al (2025) developed a BISINDO letter detection system based on YOLOv8 with CNN, trained using the Roboflow dataset. The system is capable of recognizing alphabet signs (A–Z) through real-time cameras and image uploads, then translating them into text and voice with gTTS. Evaluation results show an accuracy of 89.74% [9].

Research conducted by Indra et al. (2024) classified BISINDO letters (A–Z) using several stages, including preprocessing (grayscale, thresholding, morphology, and Prewitt) with color (HSV) and shape feature extraction, and using the Random Forest method for classification. Test results show the best accuracy of 94.2% in the 80:20 data scenario, proving the effectiveness of this approach in BISINDO letter recognition [10].

Research conducted by Indra et al. (2024) on BISINDO letter recognition using the Chain Code Contour method and Naive Bayes classification. With 260 images tested in three dataset scenarios, the best results were achieved at a ratio of 80:20 with 100% accuracy, demonstrating the effectiveness of this method in recognizing BISINDO letters [11].

Research conducted by Goenawan and Hartati in 2024 compared the K-NN and Random Forest algorithms for SIBI letter recognition based on gestures. The results show that Random Forest excels in model evaluation with an accuracy of 99.88%. In comparison, K-NN performs good in real-time testing with an average accuracy of 99%, so both have the potential to complement each other in the development of static and dynamic gesture recognition systems [12].

One effective technique for representing hand shapes is Histogram of Oriented Gradients (HOG). This method utilizes the distribution of edge and contour orientations to extract visual patterns from images [13], [14], thereby distinguishing BISINDO letters based on finger configuration and hand shape. However, recognition success is not only determined by feature quality, but also by the classification method used.

This research uses the Bagging Decision Tree algorithm as a classification method. Bagging is an ensemble learning approach that builds several Decision Tree models based on bootstrap samples of training data, then combines the prediction results through majority voting [15], [16]. This approach can improve prediction accuracy and reduce variance, making it more stable than a single Decision Tree.

This research aims to create a BISINDO letters recognition system by combining HOG feature extraction and Bagging Decision Tree classification. The system's performance is evaluated through accuracy measurements, so that the model's accuracy in recognizing BISINDO letters can be determined. Thus, this research is expected to contribute to the development of Indonesian sign language recognition technology based on digital image processing in Indonesia.

II. RESEARCH METHODOLOGY

This research consists of several stages, as shown in Figure 1. Each stage is described in the following section.

Input of BISINDO Letter Images

The initial stage involves capturing images of hands forming letters in BISINDO sign language. Images can be obtained from direct acquisition using a camera. A total of 156 images were used in the dataset, consisting of 104 letter images for training data and 52 letter images for test data. The data was processed in the system as training data, and the dataset was then saved in ‘mat’ format. The 26 BISINDO letters can be seen in Figure 2.

https://doi.org/10.35760/ik.2025.v30i3.57

Figure 1. Proposed System

Figure 2. Bisindo Letters Dataset

Image Preprocessing

The preprocessing stage aims to improve image quality and reduce noise [17], [18]. There are several steps in this stage, namely converting RGB images to grayscale images, then improving image quality by adjusting image contrast and removing noise with a median filter.

HOG Feature Extraction

HOG is used to extract shape [19] and contour [20] information from hand images. This process works by calculating the gradient orientation in each small cell in the image, then forming a gradient direction histogram. The extraction results are feature vectors that represent the edge and contour patterns of BISINDO letters.

Steps in the HOG Feature Extraction Algorithm

1. Convert the image to grayscale. To simplify processing, since HOG only requires pixel intensity.
2. Normalize Image Size to 64x64 pixels.
3. Calculate Gradient. HOG works by calculating pixel intensity gradients to capture local structures (contours and shapes) of BISINDO letters.
4. Divide the image into cells. The image is divided into small cells based on step number 2, resulting in 64 cells.
5. Perform block normalization based on the division of the image into cells, which is 49 blocks
6. Combine all features. All histograms in the block are combined into an HOG feature vector, so that the features per block are 2x2x9 = 36 features, where 2x2 is the cell block and 9 is the orientation bin, so that the total feature vector representing the shape and contour characteristics of the BISINDO letters in the image used is 49 (based on step 5) x 36 (based on step 6) = 1764 features.

Bagging Decision Tree Classification

Bagging Decision Tree is an ensemble learning method that combines several decision trees [21] to improve classification accuracy and stability. In the context of BISINDO character recognition, this algorithm is used to classify feature vectors generated from Histogram of Oriented Gradients (HOG) extraction into BISINDO character classes. The testing stages are as follows:

1. Input of BISINDO Letter Image
```
A = imread(fullfile(nama_folder, nama_file(n).name));
A = imresize(A, [67 67]);
```
2. Convert RGB to Grayscale
```
B = rgb2gray(A);
```
3. Adjusting the contrast
```
adjust = imadjust(B);
```

4. Image Sharpening
   h = fspecial('unsharp');
   gray = imfilter(adjust, h);
5. Smoothing with Filters
   gray1 = imbilatfilt(gray);
6. Performing Contrast Enhancement
   gray2 = adapthisteq(gray1);
7. Performing Noise Removal with Median Filter
   gray3 = medfilt2(gray2, [2 2]);
8. HOG Feature Extraction
   [hogFeatures, ~] = extractHOGFeatures(gray3);
9. Calling the results of the decision tree bagging training
   load Mdl2
10. Reading the output class of the bagging decision tree test results
   [hasil_uji_tree] = predict(Mdl2,features);
11. Set up test target variables
   target(1:2) = {'A'};
   …………
   target(51:52) = {'Z'};
12. Calculating the accuracy of bagging decision tree testing

```
correct_tree_count = 0;
number_of_data = size(features,1);
for k = 1:number_of_data
if isequal (test_result_tree{k},target{k})
    correct_number_tree = correct_number_tree+1;
end
end
tree_test_accuracy = correct_number_tree/number_of_data*100
```

BISINDO Character Recognition Output

This is the final stage where the system displays the classification results using a Bagging Decision Tree based on HOG feature extraction in the form of feature vectors that represent the edge and contour patterns of the letters BISINDO A-Z.

III. RESULTS AND DISCUSSION

Preprocessing Results

During the preprocessing stage, the RGB image is converted to grayscale. Then, the image quality is improved by adjusting the image contrast and removing noise using the median filter method. The results of the preprocessing are shown in Figure 3.

HOG Feature Extraction Results

After preprocessing, the HOG (Histogram of Oriented Gradient) feature extraction process is carried out to capture the features of the BISINDO A-Z letters. These features will later be used as reference data. The results of HOG feature extraction for 104 BISINDO letter training images are shown in Table 1.

(a) (b) (c) (d)

Figure 3. Preprocessing Results (A) Original Image, (B) Image Converted to Grayscale, (C) Image with Contrast Adjustment, (D) Filtered Image

Table 1. HOG Feature Extraction Values for 104 Training Data Images

Training image	Number of Letters	Feature Extraction Values											
		1	…	352	…	704	…	1056	…	1409	…	1763	1764
A.png	1	0,242	…	0,043	…	0,173	…	0,345	…	0,044	…	0,086	0,046
	2	0,151	…	0,180	…	0,259	…	0,407	…	0,149	…	0,234	0,234
	3	0,258	…	0,067	…	0,184	…	0,337	…	0,035	…	0,041	0,017
	4	0,276	…	0,073	…	0,102	…	0,312	…	0,038	…	0,043	0,011
B.png	1	0,277	…	0,409	…	0,031	…	0,373	…	0,145	…	0,337	0,337
	2	0,192	…	0,012	…	0,021	…	0,022	…	0,233	…	0,262	0,199
	3	0,181	…	0,046	…	0,038	…	0,027	…	0,242	…	0,246	0,134
	4	0,263	…	0,432	…	0,056	…	0,450	…	0,054	…	0,197	0,153
C.png	1	0,302	…	0,074	…	0,134	…	0,013	…	0,035	…	0,052	0,148
	2	0,281	…	0,021	…	0,085	…	0,130	…	0,019	…	0,179	0,150
	3	0,292	…	0,004	…	0,049	…	0,053	…	0,010	…	0,070	0,238
	4	0,275	…	0,285	…	0,053	…	0,026	…	0,264	…	0,169	0,111
D.png	1	0,249	…	0,113	…	0,055	…	0,178	…	0,140	…	0,263	0,263
	2	0,404	…	0,126	…	0,065	…	0,307	…	0,082	…	0,200	0,440
	3	0,262	…	0,253	…	0,057	…	0,269	…	0,132	…	0,311	0,384
	4	0,069	…	0,043	…	0,007	…	0,280	…	0,062	…	0,086	0,439
…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…
M.png	1	0,303	…	0,372	…	0,227	…	0,198	…	0,138	…	0,092	0,120
	2	0,281	…	0,108	…	0,091	…	0,098	…	0,193	…	0,183	0,104
	3	0,122	…	0,059	…	0,034	…	0,136	…	0,295	…	0,095	0,281
	4	0,297	…	0,344	…	0,103	…	0,086	…	0,201	…	0,227	0,189
N.png	1	0,295	…	0,276	…	0,011	…	0,131	…	0,291	…	0,286	0,308
	2	0,279	…	0,354	…	0,179	…	0,072	…	0,029	…	0,357	0,302
	3	0,266	…	0,291	…	0,180	…	0,267	…	0,021	…	0,276	0,214
	4	0,276	…	0,316	…	0,078	…	0,082	…	0,289	…	0,174	0,139
O.png	1	0,250	…	0,018	…	0,007	…	0,210	…	0,268	…	0,188	0,145
	2	0,296	…	0,071	…	0,031	…	0,256	…	0,085	…	0,208	0,165
	3	0,271	…	0,254	…	0,419	…	0,334	…	0,094	…	0,146	0,102
	4	0,235	…	0,159	…	0,452	…	0,328	…	0,005	…	0,171	0,143
P.png	1	0,182	…	0,114	…	0,483	…	0,040	…	0,051	…	0,135	0,056
	2	0,067	…	0,126	…	0,335	…	0,186	…	0,062	…	0,267	0,123
	3	0,254	…	0,217	…	0,188	…	0,295	…	0,108	…	0,048	0,071
	4	0,092	…	0,271	…	0,249	…	0,296	…	0,122	…	0,037	0,051
…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…
W.png	1	0,285	…	0,201	…	0,113	…	0,359	…	0,016	…	0,165	0,064
	2	0,082	…	0,153	…	0,009	…	0,310	…	0,059	…	0,152	0,129
	3	0,268	…	0,323	…	0,084	…	0,097	…	0,038	…	0,035	0,009
	4	0,266	…	0,316	…	0,010	…	0,010	…	0,015	…	0,146	0,072
X.png	1	0,292	…	0,124	…	0,002	…	0,242	…	0,227	…	0,118	0,064
	2	0,266	…	0,147	…	0,076	…	0,178	…	0,032	…	0,181	0,019
	3	0,285	…	0,046	…	0,004	…	0,274	…	0,003	…	0,149	0,088
	4	0,215	…	0,129	…	0,023	…	0,262	…	0,122	…	0,114	0,145
Y.png	1	0,286	…	0,277	…	0,267	…	0,093	…	0,285	…	0,107	0,277
	2	0,048	…	0,294	…	0,342	…	0,083	…	0,303	…	0,021	0,043
	3	0,105	…	0,054	…	0,294	…	0,037	…	0,263	…	0,053	0,014
	4	0,282	…	0,364	…	0,090	…	0,109	…	0,370	…	0,198	0,162
Z.png	1	0,218	…	0,025	…	0,056	…	0,352	…	0,091	…	0,011	0,116
	2	0,207	…	0,074	…	0,031	…	0,089	…	0,137	…	0,150	0,175
	3	0,273	…	0,252	…	0,232	…	0,253	…	0,009	…	0,183	0,173
	4	0,279	…	0,116	…	0,054	…	0,309	…	0,006	…	0,069	0,220

Test Results

At this stage, 52 images were tested using the Bagging Decision Tree classification. Tests with a "True" result were correctly recognized, while tests with a "False" result were highlighted in red and were incorrectly identified. The results of testing the BISINDO character recognition system using the Bagging Decision Tree classification are shown in Table 2.

Table 2. System Test Results On 52 Test Images

Test Image	Number of Test Images	Recognition Class	System Recognition Results	Result
A	A1	A	A	True
	A2	A	A	True
B	B1	B	B	True
	B2	B	B	True
C	C1	C	C	True
	C2	C	C	True
D	D1	D	D	True
	D2	D	P	False
E	E1	E	E	True
	E2	E	E	True
F	F1	F	F	True
	F2	F	F	True
G	G1	G	G	True
	G2	G	G	True
H	H1	H	H	True
	H2	H	H	True
I	I1	I	I	True
	I2	I	I	True
J	J1	J	I	False
	J2	J	N	False
K	K1	K	K	True
	K2	K	K	True
L	L1	L	L	True
	L2	L	L	True
M	M1	M	M	True
	M2	M	M	True
N	N1	N	N	True
	N2	N	N	True
O	O1	O	O	True
	O2	O	O	True
P	P1	P	P	True
	P2	P	S	False
Q	Q1	Q	Q	True
	Q2	Q	Q	True
R	R1	R	E	False
	R2	R	R	True
S	S1	S	S	True
	S2	S	S	True
T	T1	T	T	True
	T2	T	T	True
U	U1	U	U	True
	U2	U	U	True
V	V1	V	U	False
	V2	V	V	True
W	W1	W	W	True
	W2	W	W	True
X	X1	X	X	True
	X2	X	X	True
Y	Y1	Y	Y	True
	Y2	Y	M	False
Z	Z1	Z	Z	True
	Z2	Z	Z	True

https://doi.org/10.35760/ik.2025.v30i3.57

Based on Table 2, there are 52 letters, with 7 BISINDO letters detected as incorrect, namely the letters D, two letters J, P, R, V, and Y. Testing with Bagging Decision Tree classification obtained an accuracy of 86.5%.

IV. Conclusion

Based on the results of the research that has been conducted, it can be concluded that the BISINDO letters recognition system with the Histogram of Oriented Gradient (HOG) feature extraction approach and Bagging Decision Tree classification is capable of providing good performance in the letter image identification process. HOG feature extraction has been proven effective in capturing shape patterns and edge structures in BISINDO letter images, resulting in relevant and informative feature representations. Furthermore, the application of the Bagging Decision Tree classification method has successfully utilized these features optimally. This is demonstrated by the system's average accuracy of 86.5%, which confirms that the proposed method is capable of recognizing BISINDO letters quite reliably. However, this accuracy value still has the potential to be improved so that image representation is more in-depth and recognition accuracy can achieve more optimal results.

Acknowledgements

We would like to express our highest appreciation and sincere gratitude to the Institute for Research and Resource Development (LP2S), Universitas Muslim Indonesia, for the financial support provided through the 2025 Faculty Flagship Research Scheme, which enabled this research to be successfully carried out.

REFERENCES

[1] T. Handhika, R. I. M. Zen, D. P. Lestari, and I. Sari, "Gesture recognition for Indonesian sign language (BISINDO)," in *Journal of Physics: Conference Series*, 2018, vol. 1028, no. 1, p. 12173.

[2] L. N. Hayati, A. N. Handayani, W. S. G. Irianto, R. A. Asmara, D. Indra, and N. S. Damanhuri, "Optimizing YOLO-Based Algorithms for Real-Time BISINDO Alphabet Detection Under Varied Lighting and Background Conditions in Computer Vision Systems," *Int. J. Eng. Sci. Inf. Technol.*, vol. 5, no. 3, pp. 285–294, 2025, doi: 10.52088/ijesty.v5i3.948.

[3] A. F. Deleviar, Intan Oktaviani, and Hanifah Permatasari, "Pengembangan Website Speech To Video Bahasa Isyarat Indonesia (Bisindo) Berbasis Algoritma Long Shot Term Memory," *Infotek J. Inform. dan Teknol.*, vol. 8, no. 1, pp. 23–33, 2025, doi: 10.29408/jit.v8i1.26117.

[4] A. Alayed, "Machine Learning and Deep Learning Approaches for Arabic Sign Language Recognition: A Decade Systematic Literature Review," *Sensors*, vol. 24, no. 23, 2024, doi: 10.3390/s24237798.

[5] F. M. Najib, "Sign language interpretation using machine learning and artificial intelligence," *Neural Comput. Appl.*, vol. 37, no. 2, pp. 841–857, 2025, doi: 10.1007/s00521-024-10395-9.

[6] N. Nurhadi, E. A. Winanto, R. M. Said, J. Jasmir, and L. Afuan, "Pattern Classification Sign Language Using Features Descriptors And Machine Learning," *J. Tek. Inform.*, vol. 5, no. 2, pp. 349–356, 2024.

[7] S. Tan, N. Khan, Z. An, Y. Ando, R. Kawakami, and K. Nakadai, "A review of deep learning-based approaches to sign language processing," *Adv. Robot.*, vol. 38, no. 23, pp. 1649–1667, 2024, doi: 10.1080/01691864.2024.2442721.

[8] R. Goel, S. Bansal, and K. Gupta, "Improved feature reduction framework for sign language recognition using autoencoders and adaptive Grey Wolf Optimization," *Sci. Rep.*, vol. 15, no. 1, p. 2300, 2025, doi: 10.1038/s41598-024-82785-x.

[9] F. Farhana, A. Rezha, E. Najaf, and R. Permatasari, "BISINDO Sign Language Interpreter System Using YOLOv8 and CNN," *Bit-Tech (Binary Digit. - Technol.*, vol. 8, no. 1, 2025, doi: 10.32877/bt.v8i1.2638.

[10] D. Indra, L. N. Hayati, M. A. Daris, I. As'ad, and U. Mansyur, "Penerapan Metode Random Forest Dalam Klasifikasi Huruf BISINDO dengan Menggunakan Ekstraksi Fitur Warna dan Bentuk The Application of Random Forest Method in BISINDO Letter Classification Using Color and Shape

Feature Extraction .," *Komputika J. Sist. Komput.*, vol. 9, no. 28, 2024, doi: 10.34010/komputika.v13i1.10363.
[11] D. Indra, L. N. Hayati, and M. C. Irja, "Pengenalan Huruf BISINDO Menggunakan Chain Code Contour dan Naive Bayes BISINDO letter recognition using Chain Code Contour and Naive Bayes," *Komputika J. Sist. Komput.*, vol. 13, no. 148, pp. 11–18, 2024, doi: 10.34010/komputika.v13i1.10360.
[12] A. D. Goenawan and S. Hartati, "The Comparison of K-Nearest Neighbors and Random Forest Algorithm to Recognize Indonesian Sign Language in a Real-Time," *Sci. J. Informatics*, vol. 11, no. 1, pp. 237–244, 2024, doi: 10.15294/sji.v11i1.48475.
[13] G. Joshi, S. Singh, and R. Vig, "Taguchi-TOPSIS based HOG parameter selection for complex background sign language recognition," *J. Vis. Commun. Image Represent.*, vol. 71, p. 102834, 2020, doi: https://doi.org/10.1016/j.jvcir.2020.102834.
[14] F. Jafari and A. Basu, "Saliency-Driven Hand Gesture Recognition Incorporating Histogram of Oriented Gradients (HOG) and Deep Learning," *Sensors*, vol. 23, no. 18, pp. 1–15, 2023, doi: 10.3390/s23187790.
[15] P. Yariyan *et al.*, "Improvement of best first decision trees using bagging and dagging ensembles for flood probability mapping," *Water Resour. Manag.*, vol. 34, no. 9, pp. 3037–3053, 2020.
[16] S. Boukir and W. Feng, "Boundary bagging to address training data issues in ensemble classification," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 9975–9981.
[17] A. Padmo, A. Masa, M. Mushfa, H. Fajri, A. Septiarini, and E. Winarno, "Comparison of Noise Using Reduction Method for Repairing Digital Image," *JOIV Int. J. Informatics Vis.*, vol. 8, no. December, pp. 2189–2194, 2024, [Online]. Available: www.joiv.org/index.php/joiv.
[18] N. T. Singh, C. Kaur, A. Chaudhary, and S. Goyal, "Preprocessing of Medical Images using Deep Learning: A Comprehensive Review," *Proc. 2023 2nd Int. Conf. Augment. Intell. Sustain. Syst. ICAISS 2023*, no. August, pp. 521–527, 2023, doi: 10.1109/ICAISS58487.2023.10250462.
[19] S. Bhave, A. Giri, S. Bhavsar, and G. Chiddarwar, "Effective method for Shape based Image Retrieval using Pyramid of Histogram of Oriented Gradients," in *2019 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)*, 2019, pp. 1–4.
[20] Y. Pang, Y. Yuan, X. Li, and J. Pan, "Efficient HOG human detection," *Signal Processing*, vol. 91, no. 4, pp. 773–781, 2011.
[21] M. Azad, T. H. Nehal, and M. Moshkov, "A novel ensemble learning method using majority based voting of multiple selective decision trees," *Computing*, vol. 107, no. 1, p. 42, 2025.