

PENGENALAN POLA SUKU KATA AKSARA BIMA DENGAN BARIS TANDA BUNYI MENGGUNAKAN EKSTRAKSI CIRI MOMENT INVARIANT DENGAN METODE ANN

(The Recognition of Bimanese Script with Diacritics Syllable Pattern using Artificial Neural Network from Feature Extraction Based on Invariant Moment)

Muhammad Naufal Rizqullah*^[1], Ramaditia Dwiyanaputra^[1], Fitri Bimantoro^[1]

^[1] Dept Informatics Engineering, Mataram University
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA

Email: m.naufalrizqullah17@gmail.com, [rama, bimo]@unram.ac.id

Abstract

The Bimanese script is one of the archipelago's cultural heritage that needs to be preserved. The problem arose when some Bimanese people doubted the existence of the Bimanese script. Therefore, it is essential to safeguard the Bimanese script and learn the Bimanese script starting from reading and then understanding the letters. After that, add a line of sound marks to entirely understand the Bimanese script's meaning. This study aims to build an Artificial Neural Network (ANN) model to recognize the Bimanese Script Syllable Pattern with Sound Sign Lines by using Moment Invariant feature extraction. Before doing the training, first, determine the parameters on the ANN using the Tuning Hyperparameter, in the test, using a dataset of 2250 images of the Bimanese script. Based on the results of the tests carried out based on the optimal parameters, the accuracy is 77.59%, precision is 78.44%, recall is 77.61%, and F1-Score is 77.33%. Then for testing using K-Fold cross-validation, the best results were obtained using K = 9 with a ratio of 8:1 where the resulting accuracy was 79.74%. Overall the results of this study are expected to preserve the Bimanese script and are developed more widely.

Keywords: Bimanese script, Pattern Recognition, Artificial Neural Network, Backpropagation, Moment Invariant, Tuning Hyperparameter

*Penulis Korespondensi

1. PENDAHULUAN

Bima merupakan kota yang terletak di pulau Sumbawa Provinsi Nusa Tenggara Barat. Berbagai peradaban yang luar biasa dimiliki oleh kota ini seperti tugu, monumental/candi dan aksara, tepatnya Aksara Bima [1]. Aksara Bima merupakan salah satu warisan budaya Nusantara yang perlu dilestarikan. Dalam beberapa penelitian yang mempelajari tentang aksara dalam hal ini naskah Aksara Bima, telah menjadi suatu topik yang menarik untuk dipelajari lebih dalam lagi. Dikarenakan dalam naskah aksara bima tersebut tersimpan berbagai informasi terkait berbagai sejarah baik bima itu sendiri maupun Indonesia [2].

Berdasarkan hasil survei *online* dengan jumlah responden sebanyak 87 orang dengan rentang usia 17 sampai 38 tahun, dimana terdapat 81 Orang yang berasal dari bima, sebanyak 45.7% masih belum mengetahui tentang keberadaan Aksara Bima, dan sebanyak 48.1% belum pernah mempelajari Aksara

Bima [3]. Hal ini dapat menyebabkan hilangnya aksara bima sebagai warisan budaya

Dalam pengembangannya, untuk membaca naskah atau aksara Bima, proses membacanya dimulai dari arah kiri ke kanan, mirip dengan membaca huruf Latin. Untuk dapat membaca dan memahaminya, pengamatan menunjukkan bahwa langkah pertama adalah menghafal huruf-hurufnya kemudian baru menambahkan Baris Tanda Bunyi (Harakat) agar dapat membaca naskah aksara bima. Maka dari itu, pentingnya mengenali Baris Tanda Bunyi adalah agar kita dapat memperoleh makna yang utuh. Hal ini diperparah oleh kenyataan bahwa sistem penulisan aksara Bima tidak menggunakan spasi atau pemisah baris baru [4].

Pentingnya penelitian ini adalah untuk mencegah hilangnya warisan budaya aksara bima. Salah satu alternatif lain adalah media pembelajaran digital, untuk membantu masyarakat dalam mengenali dan memahami aksara bima ini. Pengenalan Pola Suku Kata Aksara Bima dengan Baris Tanda Bunyi bisa menjadi

salah satu upaya dalam melestarikan aksara bima ini dengan membuat pembelajaran mesin yang dapat mengenali Pola Suku Kata aksara bima yang nantinya didapat model pelatihan yang bisa digunakan sebagai sarana pembelajaran, aplikasi berbasis edukasi maupun sebagai terjemahan awal untuk tiap suku kata aksara (yang bisa dikembangkan menjadi terjemahan aksara).

Pengenalan pola bertujuan untuk mengenali atau mengklasifikasikan suatu citra tertentu dalam suatu kelas, dimana pengenalan pola terbagi menjadi tiga tahapan, yaitu pemrosesan awal, ekstraksi fitur, dan klasifikasi atau pengenalan [5]. Sebelumnya, penelitian tentang pengenalan pola suku kata aksara bima sudah ada dilakukan yaitu penelitian Pengenalan Pola Tulisan Tangan Bima [3], lalu penelitian Pengenalan Pola Tulisan Tangan Aksara Bima menggunakan Ciri Tekstur dan KNN [6]. Namun, untuk penelitian yang sejenis tentang pengenalan pola suku kata ada beberapa penelitian yang dilakukan yaitu Penerapan Teknik Deteksi Tepi (Canny) dalam Sistem Pengenalan Tulisan Tangan Aksara Lampung [7]. Dari beberapa penelitian sebelumnya tersebut, belum ada yang melakukan penelitian tentang pengenalan suku kata aksara bima dengan Baris Tanda Bunyi. Kendala yang menjadi penelitian ini adalah sumber referensi yang merujuk pada aksara bima Baris Tanda Bunyi ini masih sangat minim, dikarenakan belum berkembang pesat.

Artificial Neural Network atau ANN (Jaringan syaraf tiruan) merupakan sistem komputasi yang didasarkan pada analogi dengan jaringan syaraf biologis. Model ANN ini dipengaruhi oleh struktur otak, sehingga banyak terminologi yang dipinjam dari ilmu neurosains. ANN terdiri dari unit-unit sederhana yang disebut neuron, yang terhubung melalui koneksi dengan bobot tertentu. [8]. ANN adalah sebuah mesin yang dirancang untuk memodelkan cara kerja atau cara berperilaku otak manusia dalam menyelesaikan fungsi atau tugas tertentu. Dengan mengambil inspirasi dari mekanisme kerja otak manusia, ANN terdiri dari unit pemroses yang disebut neuron. Setiap neuron memiliki penambah, fungsi aktivasi, sejumlah bobot, dan sejumlah vektor masukan [8]. Beberapa penelitian terkait dengan ANN ini yang menunjukkan performa baik yakni Studi Perbandingan Pengenalan Karakter Aksara Lampung yang menghasilkan performa akurasi menggunakan metode roberts sebesar 71.5% dan metode sobel sebesar 85.5% [9]. Lalu, Penerapan Metode Linear Discriminant Analysis dalam studi Pengenalan Pola Tulisan Tangan Aksara Sasak yang Mencapai Akurasi 92.20% [10], dan juga penelitian tentang Penggunaan ANN dalam Implementasi

Pengenalan Pola Notasi Balok yang menghasilkan akurasi tertinggi sebesar 91.20% [8].

Moment Invariant merupakan fungsi non-linear yang tetap konsisten terhadap rotasi, translasi, dan skala dalam konteks momen geometri citra. Apabila kita memiliki sebuah citra dengan nilai intensitas yang direpresentasikan sebagai $f(x,y)$, di mana x mewakili baris dan y mewakili kolom, maka Moment Invariant yang mengubah fungsi citra $f(x,y)$ dalam sistem diskrit [11]. Beberapa penelitian terkait dengan metode Moment Invariant termasuk Pengenalan Pola Tulisan Tangan Suku Kata Aksara Sasak, yang menggabungkan klasifikasi SVM dengan ekstraksi ciri *Moment Invariant* dengan menggunakan 7 fiturnya menghasilkan akurasi sebesar 89.76% [12], dan penelitian Pengenalan karakter Jawi berbasis fitur invariant moment dimana menghasilkan akurasi sebesar 90.4% [13].

Berdasarkan penjelasan sebelumnya, peneliti mengusulkan penelitian tentang Pengenalan Pola Suku Kata Aksara Bima dengan Baris Tanda Bunyi menggunakan ekstraksi ciri *Moment Invariant* dengan metode ANN.

2. TINJAUAN PUSTAKA

Penelitian mengenai Pengenalan Pola Suku Kata Aksara Bima dengan Baris Tanda Bunyi belum ditemukan ketika penelitian ini dilakukan. Adapun penelitian terkait dengan pengenalan aksara dengan suku kata yaitu, Penelitian yang bertujuan untuk menggunakan deteksi tepi (Canny) dalam sistem pengenalan tulisan tangan aksara Lampung yang menggunakan pendekatan berbasis ANN dengan akurasi sebesar 78% dengan jumlah data yang digunakan sebanyak 480 data aksara. Adapun dalam penelitian digunakan Metode Canny sebagai ekstraksi fitur yang digunakan untuk mendeteksi tepi yang dikombinasikan dengan ANN sebagai classifier untuk pengenalan [7]. Kemudian penelitian yang berkaitan dengan aksara bima ada yang penulis temukan yaitu penelitian Pengenalan Pola Tulisan Tangan Bima [3], dimana pada penelitian ini menggabungkan GLCM dan Zoning sebagai ekstraksi ciri dan Probabilistic Neural Network (PNN) sebagai klasifiernya. Hasil dari penelitian ini, menggunakan dataset sebanyak 2640 citra yang berasal dari 22 jenis aksara dengan ukuran 64x64 dan didapat hasil akurasi terbaik sebesar 81.35%.

Beberapa penelitian sebelumnya yang telah menggunakan metode ANN sebagai metode pengklasifikasian antara lain, penelitian pengembangan aplikasi untuk pengenalan aksara katana dengan metode backpropagation dengan menggunakan 3 jenis aksara yaitu aksara gojūon,

dakuon, dan handakuon. Hasil yang diperoleh adalah akurasi sebesar 44,06% untuk aksara gojūon, 48,75% untuk aksara dakuon, dan 82,5% untuk aksara handakuon dengan jumlah data yang digunakan sebesar 1420 [14]. Selanjutnya penelitian pengenalan karakter aksara lampung, penelitian ini menggunakan ANN dengan metode backpropagation dan metode deteksi tepi untuk pola garis dan didapat pada metode Roberts, terdapat persentase error sebesar 28.5%, sementara dalam metode Sobel, persentase error yang didapatkan adalah sebesar 14.5% dengan jumlah data yang digunakan sebanyak 20 data aksara [9]. Lalu penelitian Prediksi Alfabeta Bahasa Inggris menggunakan ANN, penelitian ini menggunakan ANN dengan metode backpropagation untuk memprediksi kata dari data sebanyak 1000 kata yang berasal dari catatan dengan akurasi sebesar 85.08% [15]. Selanjutnya penelitian tentang Klasifikasi Jenis Kacang-Kacangan Berdasarkan Tekstur menggunakan ANN dimana data yang digunakan sebanyak 3 jenis kacang dengan setiap jenis 35 gambar dan pada penelitian ini menggunakan ANN dengan metode backpropagation dan Fitur tekstur yang diekstraksi menggunakan Gray Level Co-occurrence Matrix (GLCM) menghasilkan akurasi sebesar 99.6% [16], dan penelitian tentang Prediksi Kategori Tumor menggunakan ANN menggunakan dataset tumor yang dimiliki oleh Igor Kononenko dan Bojan Cestnik yang pada penelitian ini menggunakan ANN dengan metode backpropagation yang mendapat akurasi sebesar 76.67% [17]. Adapun pertimbangan penulis memilih ANN sebagai classifier dikarenakan ANN dapat digunakan untuk pengenalan banyak jenis class untuk aksara bima yang ditunjukkan pada penelitian Aksara Lampung [9] dan Pengenalan Alfabeta Bahasa Inggris [15].

Dalam penelitian yang menggunakan Moment Invariant sebagai metode ekstraksi ciri, ditemukan bahwa sudah ada digunakan oleh beberapa penelitian yaitu, penelitian Pengenalan karakter Jawi berbasis fitur invariant moment dimana data yang digunakan 125 data pada penelitian ini Moment Invariant sebagai fitur yang karakternya aksaranya dirotasi menjadi 90 derajat dan 180 derajat dan skala sebesar 0.5 lalu didapat akurasi sebesar 90.4% untuk aksara yang tidak diberikan rotasi maupun skala dan akurasi sebesar 82,14% untuk aksara yang di rotasi dan diberi skala [13]. Selanjutnya penelitian Image Retrieval Arca Ganesha dengan menggunakan Metode Ekstraksi Fitur Moment Invariant Dan Deteksi Tepi Canny dimana 50 gambar digunakan dan dalam penelitian ini, metode Moment Invariant diadopsi sebagai fitur ekstraksi dan dikombinasikan dengan metode deteksi tepi canny

dengan akurasi yang didapat sebesar 74% [18]. Selanjutnya penelitian Kombinasi Gray-Level dan Moment Invariant untuk Otomasi Darah pada Deteksi Pembuluh Darah di Gambar Retina, pada penelitian ini menggabungkan metode Moment Invariant dan Gray-Level (GLCM) sebagai fitur ekstraksi dan Tree Decision sebagai klasifikasinya dengan akurasi yang didapat 96.197% [19], dan Penelitian ini membandingkan GLCM dan Moment Invariant sebagai metode ekstraksi fitur, dalam penelitian tersebut, menerapkan Teknik Pengklasifikasi Linear Discriminant Analysis dengan menggabungkan metode GLCM dan Moment Invariant sebagai metode ekstraksi fitur, dengan data yang sebanyak 1000 citra songket dan didapat akurasi tertinggi sebesar 98.33% [11]. Pemilihan Moment Invariant sebagai ekstraksi fitur dikarenakan fitur yang dihasilkan oleh Moment Invariant tidak terpengaruh oleh perlakuan rotasi, translasi maupun skala yang merujuk pada penelitian Pengenalan karakter Jawi [13] yang karakter aksaranya dirotasi sebesar 90 dan 180 derajat, lalu Image Retrieval Arca Ganesha [18] dengan menggunakan Metode Ekstraksi Fitur Moment Invariant dan Deteksi Tepi Canny.

Berdasarkan yang telah diuraikan diatas, dilakukan penelitian dengan menggunakan klasifikasi ANN dan ekstraksi fitur Moment Invariant. Maka dari itu dilakukan penelitian dengan classifier ANN dengan harapan nantinya dapat mengenali banyak class Aksara Bima yang dikombinasikan dengan ekstraksi fitur Moment Invariant yang mampu ekstraksi ciri yang baik terhadap rotasi, translasi, maupun skala yang nantinya diharapkan dapat memperoleh hasil yang baik untuk pengenalan pola suku kata aksara Bima.

2.1 Aksara Bima

Aksara Bima merupakan salah satu warisan budaya Masyarakat suku Mbojo di Bima Nusa Tenggara Barat. Aksara Bima ini sempat "hilang" terpendam ratusan tahun lamanya. Berdasarkan catatan Thomas Stanford Raffles (1978) tentang Aksara Bima lama, secara naskah, tidak pernah ditemukan sampai sekarang, sedangkan Aksara Bima baru dapat ditemukan dalam beberapa naskah Samparaja di kota Bima, Leiden di negeri Belanda dan Perpustakaan Jerman [1].

Aksara Bima sendiri memiliki dua jenis aksara, yaitu aksara huruf lengkung (aksara Bima Lama) dan aksara huruf garis-garis (aksara Bima baru). Aksara Bima lama ini menurut peneliti tidak dikembangkan lagi secara profesional dikarenakan hanya diajarkan di tingkat SD dan SMP saja, lalu faktor lainnya adalah aksara Bima catatan berdasarkan catatan Raffles tersebut telah musnah terbakar bersamaan dengan

terbakarnya istana Bima Tempo dulu sehingga hilangnya informasi akan aksara bima lama tersebut.

Adapun untuk lebih jelasnya, bentuk satu huruf aksara Bima yang berjumlah 22 huruf (17 karakter dan 5 vokal) dan bentuk sengau yang berjumlah 8 huruf sehingga semuanya berjumlah 30 huruf. Aksara Bima memiliki huruf vokal dan konsonan. Untuk huruf vokal hanya satu yang berdiri sendiri yaitu huruf “a” sedangkan huruf vokal yang lain yakni “i”, “u”, “e” dan “o” bentuk hurufnya sama dengan “a” namun harus diberi tanda agar berbeda dengan “a”, sehingga didapat aksara bima dengan tanda bunyi sebesar 90 huruf. Untuk memperjelas penjelasan di atas, dapat dilihat pada Gambar 1.

A	ꦲ	I	ꦲꦶ	U	ꦲꦸ	E	ꦲꦺ	O	ꦲꦺ
BA	ꦧꦲ	BI	ꦧꦲꦶ	BU	ꦧꦲꦸ	BE	ꦧꦲꦺ	BO	ꦧꦲꦺ
CA	ꦕꦲ	CI	ꦕꦲꦶ	CU	ꦕꦲꦸ	CE	ꦕꦲꦺ	CO	ꦕꦲꦺ
DA	ꦢꦲ	DI	ꦢꦲꦶ	DU	ꦢꦲꦸ	DE	ꦢꦲꦺ	DO	ꦢꦲꦺ
FA	ꦑꦲ	FI	ꦑꦲꦶ	FU	ꦑꦲꦸ	FE	ꦑꦲꦺ	FO	ꦑꦲꦺ
GA	ꦒꦲ	GI	ꦒꦲꦶ	GU	ꦒꦲꦸ	GE	ꦒꦲꦺ	GO	ꦒꦲꦺ
HA	ꦱꦲ	HI	ꦱꦲꦶ	HU	ꦱꦲꦸ	HE	ꦱꦲꦺ	HO	ꦱꦲꦺ
JA	ꦗꦲ	JI	ꦗꦲꦶ	JU	ꦗꦲꦸ	JE	ꦗꦲꦺ	JO	ꦗꦲꦺ

Gambar 1. Contoh Aksara Bima dengan tanda bunyi

2.2 Ekstraksi Ciri

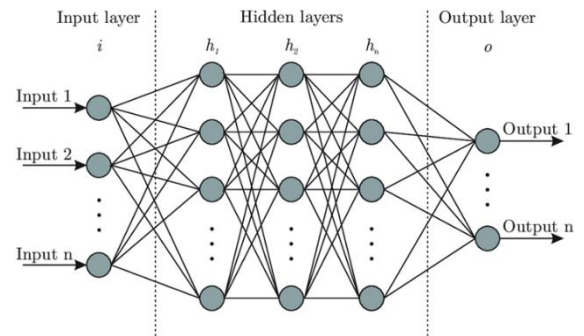
Ekstraksi ciri adalah proses di mana informasi relevan diambil dari hasil akhir berbagai metode, kemudian disimpan dalam bentuk tabel atau struktur data lainnya. Setiap tabel ini berisi informasi yang dapat menggambarkan karakteristik objek dalam sebuah citra. Tujuan dari ekstraksi ciri adalah untuk mendefinisikan sifat-sifat dari setiap objek yang terdapat dalam citra tersebut [20].

2.3 Artificial Neural Network (ANN)

ANN merupakan sistem komputasi memiliki dasar analogi jaringan syaraf biologis. ANN dimodelkan berdasarkan otak dan kerna itu banyak terminologi dipinjam dari neuroscience. ANN dari prosesor – prosesor sederhana yang terhubung lewat koneksi berbobot [21].

Dasar dari setiap Artificial Neural Network (ANN) adalah unit dasar yang disebut artificial neuron. Model ini mengikuti tiga aturan sederhana: perkalian, penjumlahan, dan aktivasi. Pada artificial neuron, input dikalikan dengan bobot yang sesuai, yang berarti setiap nilai input akan dikalikan dengan bobot individu. Langkah selanjutnya adalah penjumlahan dari hasil

perkalian tersebut dengan bias. Hasil penjumlahan ini kemudian melewati fungsi aktivasi, yang menentukan apakah neuron tersebut akan menghasilkan output atau tidak. Meskipun prinsip kerja dan aturan sederhana ANN terlihat sederhana, potensi dan kekuatan kalkulasi dari model ini menjadi kuat ketika ANN disusun secara terhubung satu sama lain [22]. Gambar 2 menunjukkan arsitektur dari ANN.



Gambar 2. Arsitektur dari ANN

2.4 Artificial Neural Network Backpropagation

Metode backpropagation digunakan untuk melatih atau menyesuaikan jaringan agar dapat mencapai keseimbangan antara kemampuan jaringan dalam mengenali pola selama pelatihan dan kemampuan jaringan dalam merespons secara akurat terhadap pola input yang mirip tetapi tidak identik dengan pola yang digunakan selama pelatihan. Backpropagation melibatkan tiga lapisan utama dalam jaringan, yaitu lapisan masukan (*input*), lapisan keluaran (*output*), dan lapisan tersembunyi (*hidden layer*) [23]. ANN Backpropagation memiliki 3 fase yaitu: Propagasi Maju, Propagasi Mundur, dan Propagasi Bobot. Secara umum, langkah-langkah dari algoritma backpropagation dijelaskan sebagai berikut [24]:
 Langkah 1: Menentukan epoch maximum, jumlah hidden neuron, learning rate dan nilai toleransi.
 Langkah 2: Lakukan langkah 3-8 jika kondisi yang diinginkan belum tercapai.

Fase 1: Feed Forward

Langkah 3: Menghitung output dari unit tersembunyi Z_j untuk setiap $j=1, 2, \dots, p$.

$$z_{netj} = v_{jo} + \sum x_i v_{ij} \quad (1)$$

$$Z_j = f(z_{netj}) = \frac{1}{1 + e^{-z_{netj}}} \quad (2)$$

Langkah 4: Menghitung output pada unit output y_k untuk setiap $k=1, 2, \dots, m$.

$$Y_{in_k} = W_{ko} + \sum_{j=1}^p Z_j W_{kj} \quad (3)$$

$$Y_{out} = f(Z_{in_k}) = \frac{1}{a + e^{-y_{in_k}}} \quad (4)$$

Fase 2: Back Forward

Langkah 5: Tiap-tiap output Y_{out} ($k=1, 2, \dots, p$) menerima target pola yang berhubungan dengan pola inputan pelatihan, hitung informasi *error*-nya.

$$\delta_k = (\delta_k - Y_{out})Y_{out}(1 - Y_{out})s \tag{5}$$

$$\Delta W_{kj} = \alpha \delta_k z_j \tag{6}$$

Langkah 6: Menghitung faktor delta untuk setiap unit tersembunyi berdasarkan kesalahan pada unit tersembunyi z_j .

$$\delta_{in_j} = \sum_{k=1}^m \delta_k W_{kj} \tag{7}$$

$$\delta_j = \delta_{in_j} z_j Z(1 - z_j)A \tag{8}$$

$$\Delta v_{ji} = \alpha \delta_j x_i \tag{9}$$

$$\Delta w_{kj} = \alpha \delta_k z_j \tag{10}$$

Fase 3: Perubahan Bobot

Langkah 7: Perubahan bobot yang menuju unit keluaran.

$$w(\text{baru}) = w_{kj}(\text{lama}) + \Delta W_{kj} \tag{11}$$

$$v(\text{baru}) = v_{ji}(\text{lama}) + \Delta W_{ji} \tag{12}$$

Langkah 8 : Menghitung nilai kesalahan dengan MSE (*Mean Squared Error*)

$$MSE = \frac{1}{n_{Pola}} \sum_k^{n_{Pola}} (t_k - Y_{out})^2 \tag{13}$$

Langkah 9: Uji kondisi berhenti, yaitu jika sudah mencapai batas kesalahan yang diharapkan atau batas iterasi maksimal.

2.5 Moment Invariant

Moment Invariant merupakan sebuah metode pengambilan ciri dari sebuah objek. Ciri yang diambil dapat berupa posisi, area, orientasi dan ciri lainnya. Metode ini dikenalkan oleh Hu pada tahun 1961 [25]. Masing-masing nilai merepresentasikan karakteristik bentuk sebuah objek. Kelebihan utama dari metode *Moment Invariant* adalah ketidaktergantungannya terhadap posisi, skala, maupun rotasi objek. Ini berarti jika dua objek memiliki bentuk yang serupa tetapi berbeda dalam hal posisi, skala, atau rotasi, nilai fiturnya akan tetap konsisten [26]. Rumus *Moment Invariant* ditunjukkan pada persamaan dibawah ini.

$$M_{pq} = \sum_x \sum_y x^p y^q I(x, y) \Delta A \tag{14}$$

Dengan,

M_{pq} = *Moment Citra*

p, q = *Orde momen*

$I(x, y)$ = Nilai intensitas pada koordinat piksel

ΔA = Area dari piksel

Melalui persamaan (14), dapat dihitung nilai translasi invariant seperti yang ditunjukkan dibawah ini.

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y) \Delta A \tag{15}$$

Berdasarkan,

$$\bar{x} = \frac{M_{10}}{M_{00}}, \text{ dan } \bar{y} = \frac{M_{01}}{M_{00}} \tag{16}$$

μ = *Moment Pusat*

\bar{x} = *Pusat sumbu x*

\bar{y} = *Pusat sumbu y*

Dari nilai translasi invariant pada persamaan (16), kita dapat menghitung nilai central moment menggunakan persamaan (17) dan (18).

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{pq})^\gamma} \tag{17}$$

$$\gamma = \frac{p+q}{2} + 1 \tag{18}$$

Dengan menggunakan nilai central moment dari persamaan (17) dan (18), kita dapat menghitung nilai ketujuh moment invariant melalui persamaan (19).

$$\begin{aligned} M1 &= \eta_{20} + \eta_{02} \\ M2 &= (\eta_{20} - \eta_{02})^2 + \eta_{11}^2 \\ M3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ M4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \\ M5 &= (\eta_{30} - 3\eta_{12})^2(\eta_{30} + \eta_{12})^2 \\ &\quad + ((\eta_{30} - \eta_{12})^2 \\ &\quad - 3(\eta_{21} - \eta_{03})^2) \\ &\quad + 3(\eta_{21} - \eta_{03})(\eta_{21} \\ &\quad + \eta_{03})(3(\eta_{30} + \eta_{12})^2 \\ &\quad - (\eta_{21} - \eta_{03})^2) \\ M6 &= (\eta_{20} + \eta_{02})(\eta_{30} + \eta_{12})^2 \\ &\quad - (\eta_{21} + \eta_{03})^2 \\ &\quad + 4\eta_{11}(\eta_{30} - \eta_{12})(\eta_{21} \\ &\quad - \eta_{03}) \\ M7 &= (3\eta_{21} - \eta_{03})(\eta_{30} \\ &\quad - \eta_{12})(\eta_{30} + \eta_{12})^2 \\ &\quad + 3(\eta_{21} + \eta_{03})^2 \\ &\quad + (3\eta_{21} - \eta_{03})(\eta_{21} \\ &\quad + \eta_{03})(3(\eta_{12} + \eta_{30})^2 \\ &\quad - (\eta_{21} + \eta_{03})^2) \end{aligned} \tag{19}$$

2.6 Confusion Matrix

Confusion Matrix adalah sebuah metode perhitungan performansi untuk permasalahan klasifikasi, metode ini sangatlah berguna untuk mengukur nilai *Recall*, *Precision*, *Accuracy*, *F-Measure* dan yang terpenting *Confusion Matrix* dapat memberikan informasi perihal jumlah class prediksi

yang diberikan terlepas dari benar atau salah prediksi tersebut [27].

TABEL I. CONFUSION MATRIX

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Akurasi adalah pengukuran seberapa benar sebuah sistem dapat mengklasifikasikan keseluruhan data [28]. Akurasi dapat dihitung menggunakan rumus dari persamaan (20).

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} \quad (20)$$

Presisi adalah rasio dari true positive terhadap jumlah true positive dan false positive. Precision yang tinggi menunjukkan bahwa contoh yang di-label positif adalah benar-benar positif. Precision dapat dihitung menggunakan rumus dari persamaan (21)

$$Presisi = \frac{TP}{FP + TP} \quad (21)$$

Recall adalah rasio dari true positive terhadap jumlah true positive dan false negative. Recall yang tinggi menunjukkan bahwa kelas berhasil dikenali dengan baik. Recall dapat dihitung menggunakan rumus dari persamaan (22).

$$Recall = \frac{TP}{TP + FN} \quad (22)$$

Selain akurasi, precision, dan recall, confusion matrix juga dapat digunakan untuk menghitung F-Score, yang bertujuan untuk menggabungkan precision dan recall. F-Score menggunakan harmonic mean dari precision dan recall, dan dapat dihitung menggunakan rumus dari persamaan (23).

$$F - Score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (23)$$

3. METODE PENELITIAN

3.1 Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini meliputi perangkat keras dan perangkat lunak, serta data berupa citra aksara Bima dalam format digital.

1. Alat Penelitian

Alat-alat yang digunakan pada penelitian ini yang berupa hardware dan software sebagai berikut:

1. Laptop Asus A455LF Intel® Core™ i3-4005U @1.70 GHz (4 CPUs), dengan RAM 10 GB

dengan GPU NVIDIA GeForce 930M (2 GB VRam)

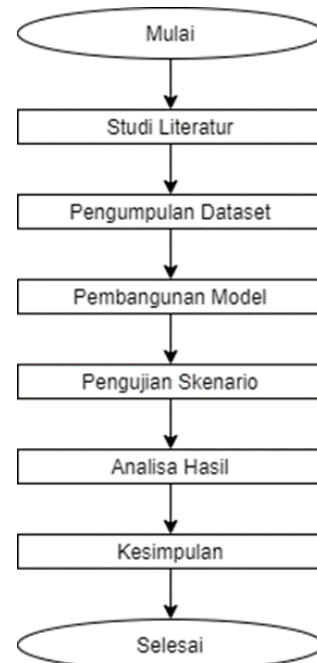
2. Main OS Windows 10 Pro 64-bit
3. Second Sistem Operasi Ubuntu Budgie 20.04 LTS
4. Jupyter Lab
5. Microsoft Office 2016

2. Bahan Penelitian

Bahan Penelitian yang digunakan pada penelitian ini adalah citra tulisan tangan digital Aksara Baris Tanda Bunyi Bima, yang di mana data didapatkan dari sukarelawan masyarakat umum sebanyak 5 orang. Selanjutnya, dimana para sukarelawan akan menuliskan aksara bima sebanyak 1 set (90 gambar aksara Baris Tanda Bunyi bima).

3.2 Alur Penelitian

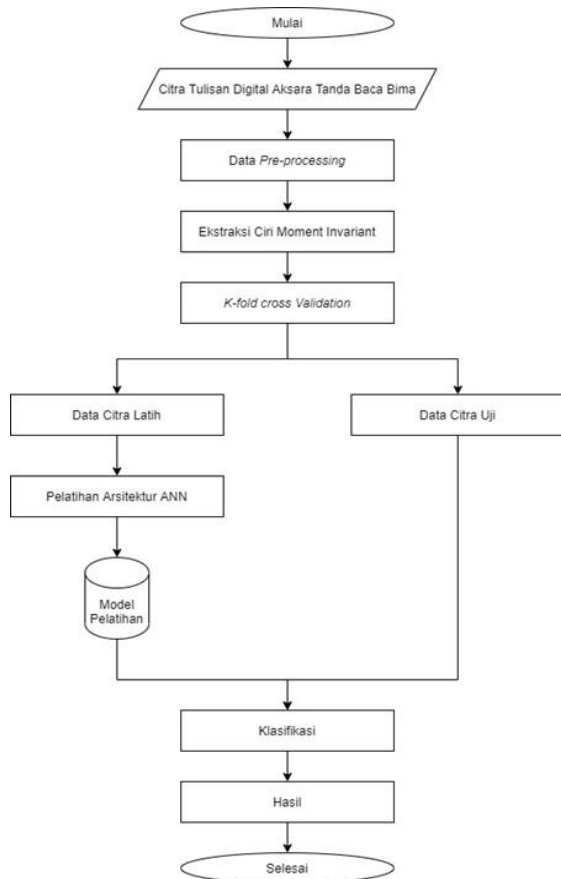
Dalam penelitian ini, diagram alir penelitian digambarkan pada Gambar 3.



Gambar 3. Diagram Alir Penelitian

3.3 Perancangan Sistem

Dalam Pembangunan model pengenalan pola menggunakan machine learning yang digunakan untuk mengenali aksara bima menggunakan Baris Tanda Bunyi dengan menggunakan metode ekstraksi ciri moment invariant dan ANN, terdapat beberapa tahapan yang harus dilakukan yaitu, data aksara akan pre-processing dulu, setelah itu dilakukan ekstraksi ciri, pembagian data menggunakan k-fold, serta tahap pengenalan menggunakan ANN. Diagram alir Pembangunan model ini dapat ditemukan pada Gambar 4.



Gambar 4. Diagram Alir Pembangunan Model

3.4 Data Pre-processing

Pada tahap pre-processing ini dilakukan untuk memanipulasi citra sehingga dapat digunakan dan didapat hasil yang optimal. Tahap pre-processing dimulai dengan merubah citra asli menjadi grayscale yang dimana ini merubah citra dari 3 dimensi (RGB) menjadi 2 dimensi. Kemudian dilakukan crop untuk mengambil bagian citra yang dibutuhkan saja, lalu dilanjutkan ke tahap resize yang dimana kedalam ukuran 64x64 pixel, penggunaan resize ini ditujukan agar semua citra memiliki ukuran yang sama dan tetap menjaga bentuk citra dan juga matriks citra yang tidak terlalu besar sehingga dapat meringankan kinerja komputasi.

3.5 Skenario Pengujian

Pada skenario pengujian pada sistem pengenalan pola Baris Tanda Bunyi aksara bima yang menggunakan metode ANN backpropagation untuk tahap pengujiannya dan moment invariant sebagai ekstraksi cirinya. Pada tahap ini model yang telah terbentuk akan di uji dan dievaluasi menggunakan beberapa parameter yaitu:

1. Jumlah hidden layer: 1, atau 2, atau 3 (dipilih melalui proses trial dan error)

2. Batas learning rate: 0.01, 0.05, dan 0.1
3. Batas epoch : 500, 1000, dan 1500
4. Variasi untuk jumlah neuron pada hidden layer (64, 128, 256)

4. HASIL DAN PEMBAHASAN

Pada tahap pengujian, dilakukan kombinasi berdasarkan parameter-parameter yang telah disebutkan sebelumnya. Pada mekanisme pengujian, hasil dari pengujian semua kombinasi tersebut nantinya akan di dapat model berdasarkan parameter terbaik, ini biasa disebut dengan *Tuning Hyperparameter*. Selanjutnya setelah di dapat parameter terbaik untuk model, maka akan dilakukan pengujian berdasarkan pembagian data menggunakan *K-fold cross validation*. Berikut ini hasil dari pengujian yang dilakukan.

4.1 Pengujian Kombinasi dengan Semua Parameter

Pada pengujian kombinasi (*Hyperparameter*) ini, untuk dapat membangun model *neural network* yang terbaik maka diperlukan tuning *hyperparameter* yang sesuai dengan cara memvariasikan nilai – nilai *hyperparameter* tersebut yang nantinya akan digunakan untuk pelatihan model [29]. Semua parameter, termasuk jumlah *neuron*, jumlah *hidden layer*, nilai *learning rate*, dan jumlah iterasi *epoch* digunakan untuk mengetahui performa model yang terbaik berdasarkan akurasi yang dihasilkan dari kombinasi tersebut. Adapun dalam proses ini dipisahkan berdasarkan jumlah *hidden layer* dikarenakan untuk meringankan beban kinerja komputer saat melakukan pelatihan. Berikut ini adalah performa dari pengujian ini adalah sebagai berikut.

TABEL II. PENGUJIAN KOMBINASI DENGAN *HIDDEN LAYER* 1

Validation Accuracy	Epoch	Neuron	Lr
0.682	500	128	0.01
0.611	500	128	0.05
0.714	500	256	0.01
0.653	500	256	0.05
0.628	1000	256	0.01
0.596	1500	64	0.01
0.695	1500	128	0.01
0.642	1500	128	0.05
0.740	1500	256	0.01
0.672	1500	256	0.05

TABEL III. PENGUJIAN KOMBINASI DENGAN *HIDDEN LAYER 2*

Validation Accuracy	Epoch	HL1	Lr	HL2
0.748971	500	64	0.01	128
0.776955	500	64	0.01	256
0.793416	500	128	0.01	256
0.751852	500	256	0.01	256
0.748148	1500	64	0.01	128
0.776543	1500	64	0.01	256
0.750206	1500	128	0.01	128
0.789300	1500	128	0.01	256
0.752263	1500	256	0.01	128
0.773663	1500	256	0.01	256

TABEL IV. PENGUJIAN KOMBINASI DENGAN *HIDDEN LAYER 3*

Validation Accuracy	Epoch	HL1	Lr	HL2	HL3
0.7617	500	64	0.01	128	256
0.7629	500	64	0.01	256	128
0.7580	500	128	0.01	128	256
0.7604	500	128	0.01	256	128
0.7543	500	128	0.01	256	256
0.7572	1500	64	0.01	128	256
0.7576	1500	64	0.01	256	128
0.7637	1500	128	0.01	128	256
0.7658	1500	128	0.01	256	128
0.7604	1500	128	0.01	256	256

TABEL V. HASIL TERBAIK BERDASARKAN *HIDDEN LAYER*

Lr	Epoch	Neuron HL1	Neuron HL2	Neuron HL3	Validasi Akurasi
0.01	1500	256	-	-	0.7407
0.01	500	128	256	-	0.7934
0.01	1500	128	256	128	0.7658

Setelah didapat semua hasil pengujian, berdasarkan kombinasi *tuning hyperparameter* maka didapatkan performa model terbaik pada *Epoch* sebesar 500, *Neuron hidden layer 1* sebesar 128, *Neuron hidden layer 2* sebesar 256 dan *learning rate* sebesar 0.01.

4.2 Pengujian Hasil *Confusion Matrix*

Setelah didapat parameter terbaik dari *tuning hyperparameter*, didapat model yang nantinya akan dilakukan evaluasi untuk melihat akurasi, *presisi*, *recall*, *f-score* dan juga *confusion matrix*. Data yang digunakan dalam pengujian ini adalah data yang telah

dikumpulkan dan telah mengalami proses augmentasi. Adapun *confusion matrix* ini peneliti rangkum dikarenakan terlalu besar untuk ditampilkan. Berikut ini hasil dari pengujian *confusion matrix*.

TABEL VI. RANGKUMAN HASIL DARI *CONFUSION MATRIX*

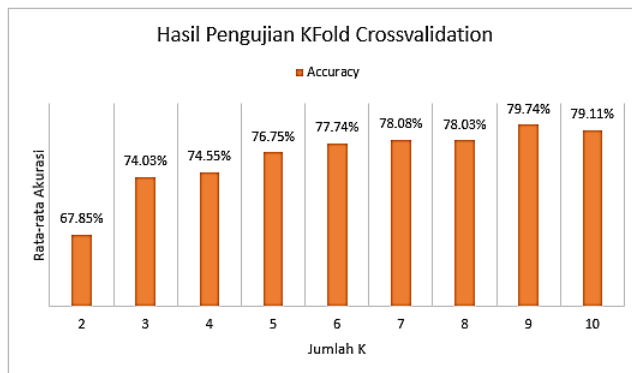
Huruf Aksara yang tertinggi berdasarkan precision, recall dan f1-score			
Aksara	Precision	Recall	F1-Score
ja	1.0	1.000000	1.000000
jo	1.0	1.000000	1.000000
wo	1.0	0.888889	0.941176
ke	1.0	0.833333	0.909091
ko	1.0	0.833333	0.909091
Huruf Aksara yang terendah berdasarkan precision, recall dan f1-score			
Aksara	Precision	Recall	F1-Score
bu	0.437500	0.388889	0.411765
ge	0.500000	0.333333	0.400000
ga	0.520000	0.722222	0.604651
ki	0.571429	0.666667	0.615385
ca	0.583333	0.777778	0.666667
Rata-rata precision, recall, dan F1-Score			
	0.784426	0.77613	0.773317
Akurasi Rata-rata			0.7759
Loss			1.619

Secara keseluruhan pengujian yang dilakukan didapat akurasi sebesar 77.59% dengan loss sebesar 1.6198. Kemudian hasil pengujian untuk *precision*, *recall* dan juga *f1-score* dapat dilihat untuk huruf aksara "ja" dan "jo" dengan keakuratan seluruhnya adalah 100%, ini menjadikan huruf aksara tersebut sebagai huruf yang paling tinggi keakuratannya untuk dikenali maupun dibedakan. Selanjutnya untuk huruf aksara "bu" paling sedikit dikenali ataupun dibedakan dengan total *precision* sebesar 43%, *recall* sebesar 38% dan untuk *f1-score* sebesar 41%. Kemudian untuk rata-rata secara keseluruhan untuk huruf aksara didapat *precision* sebesar 78.44%, lalu untuk *recall* sebesar 77.61% dan *F1-score* sebesar 77.33%.

4.3 Pengujian *K-Fold Cross Validation*

Selanjutnya pengujian dilakukan dengan menggunakan metode Pembagian Data berdasarkan *K-Fold Cross Validation*, dimana berdasarkan pengujian sebelumnya yang menggunakan *hold-out validation* (dimana data dibagi menjadi 2 bagian yaitu *train* dan *test set*) hanya sekali saja dan ini memiliki kelemahan jika set pelatihan dirubah maka menyebabkan hasil

akurasi yang berbeda. Maka dari itu dilakukan dilakukan pengujian *K-Fold Cross Validation* yang dimana menghindari bias dalam pelatihan. Adapun dalam pengujian pembagian data ini menggunakan nilai K (atau nilai fold) dimana parameter yang digunakan untuk nilai K bervariasi yaitu 2, 3, 4, 5, 6, 7, 8, 9 dan 10. Hasil dari pengujian pembagian data antara lain sebagai berikut.



Gambar 5. Diagram Alir Pembangunan Model

Berdasarkan hasil pengujian pada Gambar 5, diperoleh untuk nilai rata-rata akurasi terendah terjadi pada $K=2$, dimana akurasi sebesar 67.85% dengan rasio pembagian kelompok datanya sebanyak 1:1. Selanjutnya nilai rata-rata akurasi tertinggi terjadi pada nilai $K=9$ dengan akurasi sebesar 79.74%, dimana rasio pembagian kelompok datanya sebesar 8:1 untuk *train* dan *test* set.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, terdapat beberapa hal yang dapat disimpulkan oleh penulis, antara lain sebagai berikut.

1. Pada penelitian ini diperoleh model terbaik untuk pengenalan Pola menggunakan *Tuning Hyperparameter* pada *Epoch* sebesar 500, *Neuron hidden layer 1* sebesar 128, *Neuron hidden layer 2* sebesar 256 dan *learning rate* sebesar 0.01.
2. Pengujian menggunakan model terbaik untuk klasifikasi menggunakan ANN *Backpropagation* dengan metode *Moment Invariant* pada Aksara Bima dengan Tanda Bunyi menghasilkan akurasi sebesar 77.59%, *precision* sebesar 78.44%, *recall* sebesar 77.61%, dan *F1-Score* sebesar 77.33%.
3. Berdasarkan hasil pengujian data menggunakan *K-Fold Cross Validation* didapatkan K terbaik yaitu 9

dengan rasio pembagian kelompok data sebanyak 8:1, dimana model yang dihasilkan mendapatkan akurasi rata-rata sebesar 79.74%.

5.2 Saran

Berdasarkan hasil penelitian yang dilakukan, penelitian ini masih sangat jauh dari kata sempurna. Maka dari itu adapun saran yang dapat penulis berikan untuk perbaikan maupun pengembangan terhadap penelitian ini, antara lain sebagai berikut.

1. Penambahan kombinasi metode ekstraksi ciri selain *Moment Invariant* seperti *zoning*, GLCM, LBP, HOG ataupun metode lainnya agar dikombinasikan untuk menambah jumlah fitur pada gambar sehingga lebih bisa dikenali oleh *classifier*.
2. Mencoba menggunakan metode *Deep Learning* seperti CNN ataupun RNN.
3. Penambahan Dataset untuk Aksara Bima dengan Tanda Bunyi dengan variasi yang lebih beragam.
4. Pengembangan penelitian lebih lanjut dari pengenalan pola aksara menjadi terjemahan aksara yang sekiranya bisa di implementasikan langsung ke masyarakat.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada semua pihak yang telah memberikan bantuan dalam menjalankan proses penelitian ini serta tidak lupa pula kepada sukarelawan yang telah membantu dalam proses pengumpulan data Aksara Bima dengan Tanda Bunyi serta rekan-rekan saya yang banyak membantu sehingga dapat menyelesaikan penelitian ini.

DAFTAR PUSTAKA

- [1] S. Abubakar, "Aksara Bima Usaha Menemukanali dan Mengembangkannya," *STIT Sunan Giri Bima*, pp. 1–30, 2018.
- [2] A. Aranta, F. Bimantoro, and I. P. T. Putrawan, "Implementation of Rule Base Algorithm with Hexadecimal Approach to Bima Aksara Transliteration to Latin Letter," *Jtika*, vol. 2, no. 1, pp. 130–141, 2020.
- [3] M. Naufal, "Pengenalan Pola Tulisan Tangan Aksara Bima dengan Ekstraksi Ciri GLCM dan Zoning & Klasifikasi Probabilistic Neural Network," Mataram, 2020.
- [4] S. M. R. Salahuddin, M. Sulaiman, and S. Abubakar, *Aksara Bima : Peradaban Lokal Yang Sempat Hilang*, Cetakan Pe. Bima: Mataram Alam

- Tara Institute, 2013.
- [5] R. R. Riansyah, Y. I. Nurhasanah, and I. A. Dewi, "Sistem Pengenalan Aksara Sunda Menggunakan Metode Modified Direction Feature dan Learning Vector Quantization," *J. Tek. Inform. dan Sist. Inf.*, vol. 3, no. 1, pp. 17–30, 2017, doi: 10.28932/jutisi.v3i1.563.
- [6] F. Bimantoro, A. Aranta, G. S. Nugraha, R. Dwiyanaputra, and A. Y. Husodo, "Pengenalan Pola Tulisan Tangan Aksara Bima menggunakan Ciri Tekstur dan KNN," *J-COSINE*, vol. 5, no. 1, pp. 60–67, 2021.
- [7] E. Hara, H. Fitriawan, and Y. Mulyani, "Penggunaan Deteksi Tepi (Canny) pada Sistem Pengenalan Tulisan Tangan Aksara Lampung Berbasis Jaringan Syaraf Tiruan," *Electr. – J. Rekayasa dan Teknol. Elektro*, vol. 10, no. 3, p. 8, 2016.
- [8] J. P. Haumahu, "Implementasi Jaringan Syaraf Tiruan Untuk Pengenalan Pola Notasi Balok Menggunakan Metode Backpropagation," *J. Ris. Komput.*, vol. 6, no. 3, pp. 255–259, 2019, [Online]. Available: <http://ejurnal.stmik-budidarma.ac.id/index.php/jurikom>.
- [9] H. A. Sholeh, Y. Mulyani, and H. D. Septama, "Sobel a Comparative Study Lampung Script Characters Recogniton Based Edge Detection Method of Roberts," vol. 6, no. 3, pp. 261–272, 2018.
- [10] A. A. S. M. K. Maharani and F. Bimantoro, "Pengenalan Pola Tulisan Tangan Aksara Sasak Menggunakan Metode Linear Discriminant Analysis dan Jaringan Syaraf Tiruan Jenis Backpropagation," *J. Teknol. Informasi, Komputer, dan Apl. (JTika)*, vol. 2, no. 2, pp. 237–247, 2020, doi: 10.29303/jtika.v2i2.105.
- [11] and I. G. . P. S. W. F. Bimantoro, Nurhalimah, "Glcm Dan Moment Invariant Dengan Teknik Pengklasifikasian Linear Discriminant Analysis (LDA)," *J-Cosine*, vol. 2, no. 1, pp. 173–183, 2020.
- [12] R. Yulianti, I. G. P. S. Wijaya, and F. Bimantoro, "Pengenalan Pola Tulisan Tangan Suku Kata Aksara Sasak Menggunakan Metode Moment Invariant dan Support Vector Machine," *J. Comput. Sci. Informatics Eng.*, vol. 3, no. 2, pp. 91–98, 2019, doi: 10.29303/jcosine.v3i2.181.
- [13] F. Arnia, K. Saddami, and K. Munadi, "Moment invariant-based features for Jawi character recognition," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 3, pp. 1711–1719, 2019, doi: 10.11591/ijece.v9i3.pp1711-1719.
- [14] M. F. B. Arfianto, "Implementasi Jaringan Syaraf Tiruan Back Propagation dalam Pengembangan Aplikasi untuk Mengidentifikasi Aksara Katakana," *J. Inform. Upgris*, vol. 5, no. 1, pp. 65–72, 2019, doi: 10.26877/jiu.v5i1.3412.
- [15] H. H. Heriz, H. M. Salah, S. Bashir, A. Abdu, M. M. El Sbihi, and S. S. Abu-Naser, "English Alphabet Prediction Using Artificial Neural Networks," *Int. J. Acad. Pedagog. Res.*, vol. 2, no. 11, pp. 8–14, 2018.
- [16] M. E. Al Rivan, N. Rachmat, and M. R. Ayustin, "Klasifikasi Jenis Kacang-Kacangan Berdasarkan Tekstur Menggunakan Jaringan Syaraf Tiruan," *J. Komput. Terap.*, vol. 6, no. 1, pp. 89–98, 2020.
- [17] I. M. Nasser and S. S. Abu-Naser, "Predicting Tumor Category Using Artificial Neural Networks," *Int. J. Acad. Heal. Med. Res.*, vol. 3, no. 2, pp. 1–7, 2019, [Online]. Available: www.ijeais.org/ijahmr.
- [18] H. Nugroho and E. P. Mandyartha, "Image Retrieval Arca Ganesha Dengan Menggunakan Metode Ekstrasi Fitur Moment," *J. Ilm. NERO*, vol. 3, no. 2, pp. 93–99, 2017.
- [19] W. A. Mustafa, H. Yazid, and W. Kamaruddin, "Combination of gray-level and moment invariant for automatic blood vessel detection on retinal image," *J. Biomimetics, Biomater. Biomed. Eng.*, vol. 34, pp. 10–19, 2017, doi: 10.4028/www.scientific.net/JBBBE.34.10.
- [20] N. A. Fadil and I. Muhimmah, "Pengenalan Citra Aksara Jawa Pada Plang Jalan," *Autom. Ull*, vol. 1, no. 2, 2020.
- [21] J. P. Haumahu, "Implementasi Jaringan SyarafTiruan Untuk Pengenalan Pola Notasi Balok Menggunakan Metode Backpropagation," *JITE (Journal Informatics Telecommun. Eng. Available*, vol. 3, no. 1, pp. 41–47, 2019.
- [22] H. D. Bhakti, "Aplikasi Artificial Neural Network (ANN) untuk Memprediksi Masa Studi Mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Gresik," *Eksplora Inform.*, vol. 9, no. 1, pp. 88–95, 2019, doi: 10.30864/eksplora.v9i1.234.
- [23] Z. Hasanati and Dwiny Meidelfi, "Kajian Implementasi Jaringan Syaraf Tiruan Metode Backpropagation Untuk Deteksi Bau," *J. Appl. Comput. Sci. Technol.*, vol. 1, no. 2, pp. 90–95, 2020, doi: 10.52158/jacost.v1i2.113.
- [24] S. R. Suhartanto, C. Dewi, and L. Muflikhah, "Implementasi Jaringan Syaraf Tiruan Backpropagation untuk Mendiagnosis Penyakit Kulit pada Anak," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 7, pp. 555–562, 2017.
- [25] H. Y. Susetya, A. Rachmat, and K. A. Nugraha, "Implementasi Moment Invariant Untuk Pengenalan Label Buku Perpustakaan Berbasis Android," *J. Terap. Teknol. Inf.*, vol. 1, no. 1, pp. 21–30, 2017, doi: 10.21460/jutei.2017.11.13.
- [26] K. Adi Nugraha, "Deteksi Area Parkir Mobil

- Berbasis Marker Menggunakan Moment Invariants dan K-NN," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 1, pp. 112–121, 2019, doi: 10.28932/jutisi.v5i1.1586.
- [27] M. Z. Eryad, K. N. Ramadhani, A. Arifianto, and L. Belakang, "Pengenalan Bentuk Tangan dengan Convolutional Neural Network (CNN)," *e-Proceeding Eng.*, vol. 7, no. 2, pp. 8212–8222, 2020.
- [28] R. Herwanto, K. Gunadi, and E. Setyati, "Pengenalan Golongan Jenis Kendaraan Bermotor pada Ruas Jalan Tol Menggunakan CNN," *J. Infra Petra*, vol. 8, no. 1, pp. 196–202, 2020.
- [29] I. B. Pakpahan and I. C. Dewi, "Pendeteksian Lubang Pada Jalanan Menggunakan Metode SSD-MobileNet," *IJEIS*, vol. 11, no. 2, pp. 1–11, 2021, doi: <https://doi.org/10.22146/ijeis.60157>.