JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 12 No. 3S1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v12i3S1.5219

KLASIFIKASI CITRA PLANKTON DENGAN ALGORITMA HIBRIDA CONVOLUTIONAL NEURAL NETWORK DAN EXTREME LEARNING MACHINE

Muhammad Syaugi Shahab[1], Achmad Junaidi[2*], Andreas Nugroho Sihananto[3]

[1,2,3]Universitas Pembangunan Nasional "Veteran" Jawa Timur; Jl. Rungkut Madya, Gn. Anyar, Kec. Gn. Anyar, Surabaya; (031) 8706369

Received: 20 Agustus 2024
Accepted: 5 Oktober 2024
Published: 12 Oktober 2024

Keywords:
CNN; ELM; SMOTE.

Corespondent Email:
achmadjunaidi.if@upnjatim.ac.id

Abstrak. Penelitian ini bertujuan untuk meningkatkan akurasi dalam klasifikasi plankton secara otomatis dengan pendekatan hibrida CNN-ELM. Menggunakan *Convolutional Neural Network* (CNN) untuk ekstraksi fitur dan *Extreme Learning Machine* (ELM) sebagai pengklasifikasi, model ini dirancang untuk mengatasi tantangan citra plankton yang buram, *dataset* kecil, dan ketidakseimbangan kelas. SMOTE digunakan untuk menangani ketidakseimbangan data. Dari implementasi SMOTE dengan metode interpolasi, permasalahan ketidakseimbangan kelas berhasil diatasi dengan menjadikan jumlah data latih sama rata untuk setiap kelas. Dari hasil pengujian, konfigurasi dengan 32 *filter* dan 2000 *hidden node* serta 64 *filter* dan 2000 *hidden node* memberikan performa terbaik dengan akurasi 97,78%. Sebaliknya, model dengan 64 *filter* dan 4000 *hidden node* menunjukkan performa terendah dengan akurasi 82,78% yang diakibatkan *overfitting*. Analisis *confusion matrix* mengungkapkan kinerja tinggi pada beberapa kelas plankton, namun masih kesalahan klasifikasi sering terjadi pada kelas seperti Alexandrium, Noctiluca, dan Nitzschia. Temuan ini menunjukkan bahwa konfigurasi dengan *filter* dan *node* yang lebih kompleks tidak selalu menghasilkan kinerja lebih baik. Penelitian ini diharapkan dapat mendukung pengambilan keputusan di bidang kelautan.

Abstract. *This research aims to improve the accuracy of automatic plankton classification using a hybrid CNN-ELM approach. By utilizing Convolutional Neural Network (CNN) for feature extraction and Extreme Learning Machine (ELM) as the classifier, the model is designed to address challenges such as blurry plankton images, small datasets, and class imbalance. SMOTE with interpolation was employed to tackle data imbalance, effectively equalizing the number of training samples across all classes. The testing results indicated that configurations with 32 filters and 2000 hidden nodes, as well as 64 filters and 2000 hidden nodes, delivered the best performance with an accuracy of 97.78%. In contrast, the model with 64 filters and 4000 hidden nodes showed the lowest performance with 82.78% accuracy due to overfitting. The confusion matrix analysis revealed high performance in classifying some plankton classes; however, frequent misclassifications were observed in classes such as Alexandrium, Noctiluca, and Nitzschia. These findings suggest that more complex filter and node configurations do not always result in better performance. This research is expected to support decision-making in marine science.*

1. PENDAHULUAN

Lautan menutupi hampir 71% dari permukaan Bumi dan merupakan salah satu ekosistem yang paling penting secara biologis [1]. Di dalam ekosistem laut, plankton memainkan peran kunci sebagai dasar rantai makanan dan indikator kualitas lingkungan [2]. Organisme mikroskopis ini berkontribusi pada produksi primer global dan memainkan peran penting dalam siklus karbon [3]. Perubahan lingkungan, baik fisik, kimia, maupun biologi, dapat memengaruhi komposisi dan struktur plankton, menjadikannya indikator penting dalam memantau kondisi laut dan perubahan iklim [2].

Klasifikasi plankton secara tradisional dilakukan melalui metode manual, yang memerlukan observasi langsung dan analisis mendalam oleh para ahli. Meskipun memberikan wawasan yang berharga, metode ini sangat memakan waktu dan tidak praktis untuk diterapkan dalam skala besar. Dengan kemajuan teknologi, metode berbasis pencitraan telah muncul sebagai solusi yang lebih efisien [4]. Namun, klasifikasi otomatis plankton menghadapi tantangan seperti kualitas gambar yang rendah, ukuran dataset yang kecil, dan ketidakseimbangan kelas antar spesies [5].

Untuk mengatasi tantangan ini, penelitian ini mengusulkan pendekatan hibrida yang menggabungkan *Convolutional Neural Network* (CNN) dan *Extreme Learning Machine* (ELM). CNN digunakan untuk ekstraksi fitur dari citra plankton, sedangkan ELM bertugas melakukan klasifikasi berdasarkan fitur yang diekstraksi. Kombinasi ini menawarkan kecepatan pelatihan yang lebih tinggi dan akurasi klasifikasi yang lebih baik dibandingkan dengan metode tradisional, serta tidak memerlukan dataset yang sangat besar [6].

Selain itu, ketidakseimbangan data antar kelas plankton ditangani dengan menggunakan algoritma *Synthetic Minority Oversampling Technique* (SMOTE). SMOTE bekerja dengan menciptakan sampel sintetis yang meniru karakteristik sampel minoritas, sehingga distribusi kelas menjadi lebih seimbang dan meningkatkan kinerja model. Dengan metode ini, tantangan ketidakseimbangan data dapat diatasi, memungkinkan model untuk belajar lebih efektif dari data yang ada [7].

Penelitian ini diharapkan tidak hanya meningkatkan kinerja dalam klasifikasi plankton, tetapi juga memberikan kontribusi signifikan terhadap pemahaman tentang ekosistem laut. Serta, penelitian ini dapat membantu para ahli kelautan dalam pengambilan keputusan yang lebih cepat dan akurat.

2. TINJAUAN PUSTAKA

2.1. Convolutional Neural Network

CNN adalah jaringan *deep learning* yang digunakan dalam visi komputer untuk mengenali dan mengklasifikasikan fitur-fitur dalam gambar. Arsitektur CNN terinspirasi dari cara kerja dan organisasi korteks visual, yang dirancang menyerupai koneksi antara neuron di otak manusia [8]. Secara umum, CNN terdiri dari enam komponen utama, yaitu *input image*, *convolution layer*, *pooling layer*, *activation function*, *fully connected layer* dan *output layer*.

2.2. Extreme Learning Machine

Extreme Learning Machine (ELM) adalah algoritma pembelajaran yang dikembangkan untuk *single-hidden layer feedforward neural network* (SLFN). Berbeda dengan metode konvensional, ELM memilih *node* tersembunyi secara acak dan menghitung bobot keluaran secara analitis. Keunggulan utama ELM adalah kecepatan pembelajarannya yang sangat tinggi, menggunakan metode *single-pass learning*. Algoritma ini umumnya menghasilkan kinerja generalisasi yang baik tanpa memerlukan penyesuaian parameter secara iteratif, seperti yang diperlukan dalam metode tradisional, serta tidak melibatkan proses *backpropagation* [9].

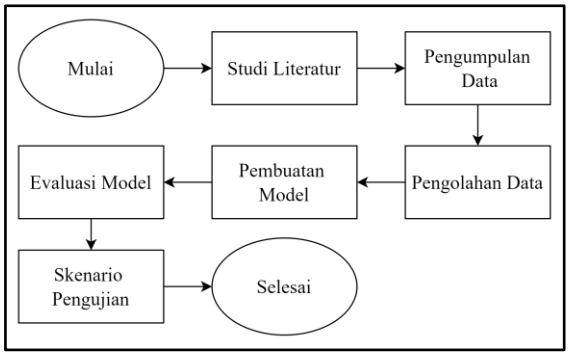
2.3. Synthetic Minority Over-sampling Technique (SMOTE)

Dalam perkembangan *machine learning*, ketidakseimbangan data antar kelas dalam *dataset* menjadi tantangan utama, terutama dalam *supervised learning* untuk klasifikasi atau pengenalan pola. Ketidakseimbangan ini membuat model lebih fokus pada kelas mayoritas, menyebabkan performa yang kurang optimal pada kelas minoritas. Untuk mengatasi masalah ini, [10] mengembangkan metode Synthetic Minority Over-sampling Technique (SMOTE), yang menggunakan teknik *oversampling* untuk menciptakan sampel sintetis dari kelas minoritas. SMOTE bekerja dengan memilih sampel minoritas secara acak,

kemudian menggunakan interpolasi antara sampel tersebut dengan tetangga terdekatnya untuk menghasilkan sampel sintetis. Metode interpolasi seperti *bilinear* dan *bicubic* meningkatkan resolusi gambar tetapi sering menghasilkan tepi kasar dan efek cincin. Sebagai alternatif, pendekatan berbasis pembelajaran meningkatkan kualitas gambar dengan memanfaatkan korelasi antara gambar resolusi rendah dan tinggi yang dipelajari dari data pelatihan [11]. Metode ini bertujuan meningkatkan kinerja klasifikasi pada *dataset* yang tidak seimbang dengan menghasilkan sampel sintetis yang realistis dan mendekati sampel asli dalam ruang fitur sesuai dengan jumlah kelas mayoritas [12].

3. METODE PENELITIAN

Tahapan metode penelitian yang dilakukan dalam penelitian ini dijalankan seperti pada Gambar 1, yang terdiri dari studi literatur, pengumpulan data, pengolahan data, pembuatan model, evaluasi model, dan skenario pengujian pada model yang telah dilatih.



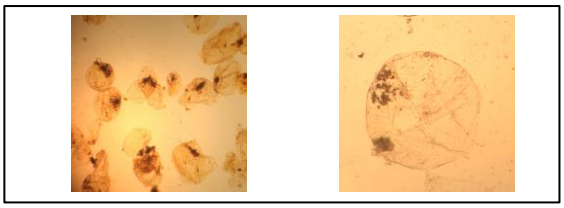
Gambar 1. Langkah-Langkah Penelitian

3.1. Studi Literatur

Untuk memahami dasar teori dan memperkuat landasan penelitian, studi literatur dilakukan dengan fokus pada kajian ilmiah dan publikasi terkait plankton. Penelitian ini mencakup analisis mendalam tentang kajian ilmiah, jurnal kelautan, serta metode penanganan ketidakseimbangan data. Selain itu, studi ini mengeksplorasi penerapan metode CNN-ELM dalam klasifikasi citra plankton dan membahas pentingnya plankton serta tantangan dalam klasifikasinya berdasarkan kajian literatur.

3.2. Pengumpulan Data

Pengumpulan data dilakukan dengan menggunakan *dataset* berisi 897 gambar plankton laut mikroskopis, yang dilabeli oleh peneliti di *Woods Hole Oceanographic Institution* (WHOI). Gambar-gambar ini dikelompokkan ke dalam 18 kategori kelas dan dikumpulkan secara rutin setiap tahun menggunakan *Imaging FlowCytobot* (IFCB) di *Martha's Vineyard Coastal Observatory* (MVCO) [13]. Dataset ini memiliki lisensi MIT, memungkinkan penggunaan bebas dalam penelitian. Gambar 2 menunjukkan contoh citra plankton dengan kelas Asterionellopsis dari *dataset*.

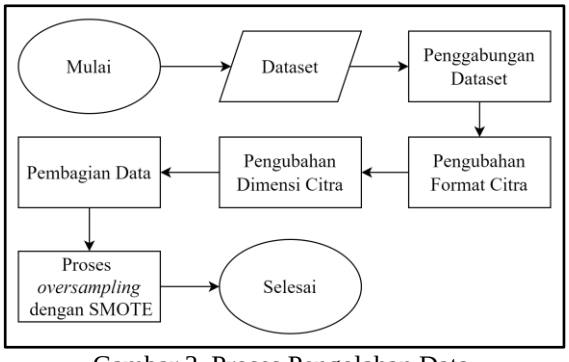


Gambar 2. Asterionellopsis Glacialis

Serta, dapat dilihat pada Gambar 2 bahwa dimensi citra plankton dan warna citra yang diperoleh sangatlah beragam. Oleh karena itu, data citra plankton pada setiap kelas perlu terlebih dahulu dilakukan pengolahan data sebelum dilakukan proses pembuatan model seperti yang telah dipaparkan pada Gambar 1.

3.3. Pengolahan Data

Setelah pengumpulan data citra, tahap pengolahan dilakukan untuk memastikan keberlanjutan dan konsistensi saat membangun model. Langkah ini mencakup normalisasi data, yang diperlukan karena adanya perbedaan format, dimensi citra, dan ketidakseimbangan data dalam dataset. Proses ini dijelaskan lebih lanjut pada Gambar 3.

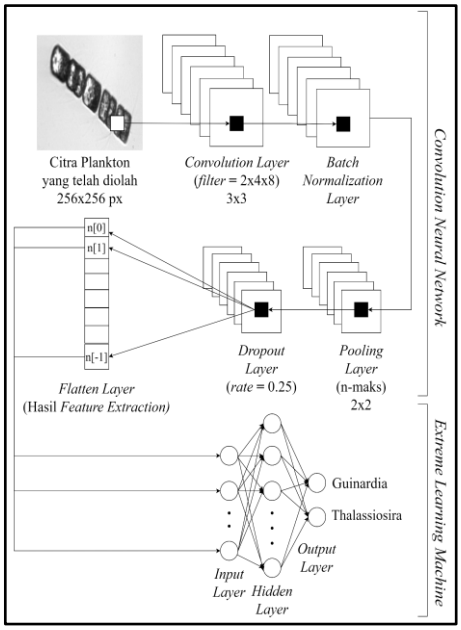


Gambar 3. Proses Pengolahan Data

Langkah-langkah pengolahan data dari Gambar 3 dimulai dengan menggabungkan *dataset* plankton WHOI menjadi satu jalur data pelatihan untuk anotasi kelas plankton. Citra dikonversi menjadi *grayscale* untuk menyederhanakan analisis dan mengurangi kompleksitas data. Dimensi citra diubah menjadi seragam, yaitu 256x256 piksel. Data kemudian dibagi dengan rasio 80:20, di mana 80% digunakan untuk pelatihan model dan 20% untuk pengujian, memastikan jumlah data yang cukup untuk klasifikasi multi-kelas [14]. Karena ketidakseimbangan data antar kelas, SMOTE diterapkan untuk melakukan *oversampling* pada kelas minoritas melalui metode interpolasi, sehingga distribusi data menjadi lebih seimbang.

3.4. Pembuatan Model

Tahapan pembuatan model dimulai dengan merancang arsitektur algoritma hibrida *Convolutional Neural Network* (CNN) dan *Extreme Learning Machine* (ELM) seperti ditunjukkan pada Gambar 4 Arsitektur ini terdiri dari dua tahap utama, yaitu ekstraksi fitur menggunakan CNN dan klasifikasi menggunakan ELM. Model arsitektur yang dikembangkan menerima citra plankton berukuran 256x256 piksel sebagai *input*, setelah melalui tahap pengolahan data. Arsitektur ekstraksi fitur menggunakan CNN terdiri dari tiga lapisan *convolution*, *batch normalization*, *pooling*, dan *dropout*.



Gambar 4. Kerangka Kerja Metode

Convolution layer dengan kernel 3x3 digunakan untuk mengekstrak fitur, menghasilkan *feature map* yang menonjolkan fitur tertentu. *Batch normalization* kemudian menormalkan fitur tersebut, menjaga keluaran dengan rata-rata mendekati 0 dan standar deviasi mendekati 1. *Pooling layer* mengurangi ukuran *feature map*, sementara *dropout layer* membantu mencegah *overfitting* dengan secara acak menonaktifkan sebagian *neuron* selama proses *training*. Fitur yang telah diekstrak kemudian di-*flatten* menjadi *array* 1 dimensi dan dimasukkan ke dalam ELM untuk klasifikasi. Di sini, fitur dikalikan dengan bobot, ditambah bias, dan diproses melalui *feedforward network* untuk mempelajari hubungan non-linier antara fitur dan kelas, hingga akhirnya menghasilkan nilai keluaran untuk klasifikasi di *output layer*.

3.5. Evaluasi Model

Evaluasi model sangat penting untuk mengukur efektivitas dan keberhasilan model pada data baru. *Confusion matrix* digunakan sebagai dasar untuk menghitung nilai *accuracy*, *precision*, *recall*, F1-*score* dan *support*, dengan mempertimbangkan empat aspek utama, yaitu.

1. *True Positive* (TP): Deteksi yang benar sesuai dengan kelas plankton yang seharusnya terdeteksi.
2. *True Negative* (TN): Deteksi yang benar di mana kelas plankton seharusnya tidak terdeteksi.
3. *False Positive* (FP): Deteksi yang salah, di mana model salah mengidentifikasi plankton yang sebenarnya tidak ada.
4. *False Negative* (FN): Plankton yang sebenarnya ada tetapi tidak terdeteksi oleh model.

Evaluasi ini membantu menilai kinerja model dalam klasifikasi yang akurat dan mengidentifikasi area yang memerlukan perbaikan.

3.6. Skenario Pengujian

Tahapan pengujian melibatkan enam skenario yang berbeda dengan menguji beberapa parameter. Parameter yang diuji mencakup jumlah *kernel filter* pada *convolution layer* CNN dan jumlah *node* pada *hidden layer* ELM. Variasi yang digunakan adalah 16, 32, dan 64 kernel filter pada *convolution layer* CNN, serta 2000 dan 4000 *node* pada *hidden*

layer ELM. Rincian parameter pengujian ini disajikan pada Tabel 1.

Tabel 1. Skenario Pengujian

Skenario Ke-	Jumlah Filter (CNN)	Jumlah Node (ELM)
1	16	2000
2	16	4000
3	32	2000
4	32	4000
5	64	2000
6	64	4000

4. HASIL DAN PEMBAHASAN

4.1. Lingkungan Pengembangan

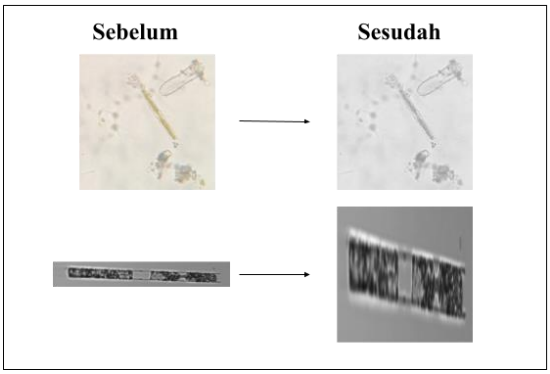
Implementasi dilakukan dengan menggunakan bahasa pemrograman Python yang menggunakan *library* TensorFlow, Keras, scikit-learn, dan numpy. Dengan dijalankan di lingkungan pengembangan Google Colab yang menggunakan *hardware* T4 GPU untuk pembuatan modelnya.

4.2. Pengolahan Data

Proses pengolahan data plankton melibatkan beberapa tahap, termasuk konversi format warna dan dimensi citra, pembagian data, serta oversampling menggunakan SMOTE, seperti yang ditunjukkan pada Gambar 3. Berikut ini adalah pembahasan mengenai implementasi dari setiap langkah tersebut.

4.2.1. Perubahan Format Data Citra

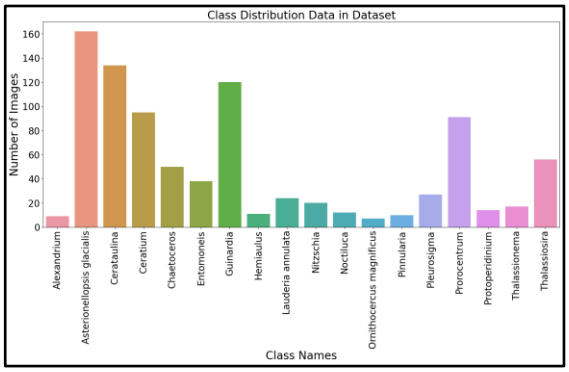
Implementasi perubahan format data citra terdapat pengecekan untuk memeriksa apakah format warna citra sudah merupakan *grayscale* atau belum agar fungsi perubahan dapat dijalankan. Proses tersebut dilakukan dengan mengakses dimensi terakhir dari citra, yang mewakili jumlah channel warna pada gambar, yang dimana jika sama dengan 1, maka gambar tersebut hanya memiliki satu channel, yang berarti gambar adalah *grayscale* dan sebaliknya jika tidak. Dapat dilihat pada Gambar 5 merupakan hasil citra yang telah diubah formatnya agar dapat seragam dalam hal format ukuran dan warna. Dalam proses perubahan dimensi citra, gambar dengan ukuran horizontal yang lebih panjang dari vertikal akan disesuaikan menjadi 256x256 piksel. Penyesuaian ini menyebabkan rasio citra berubah, menghasilkan gambar yang terlihat lebih pipih atau memanjang. Perubahan ini mempermudah model dalam menangkap fitur, karena pada lapisan *convolution* dan *pooling* akan lebih mudah memproses gambar yang telah diubah ukurannya.



Gambar 5. Perbandingan Sesudah Pra-Pemrosesan

4.2.2. Pembagian Data dan Proses Oversampling SMOTE

Dalam pengolahan data, pembagian data dan *oversampling* menggunakan metode SMOTE adalah tahap krusial untuk mengatasi ketidakseimbangan antar kelas. Gambar 6 menunjukkan bahwa distribusi citra plankton antar kelas memiliki kesenjangan signifikan dalam persebaran data.



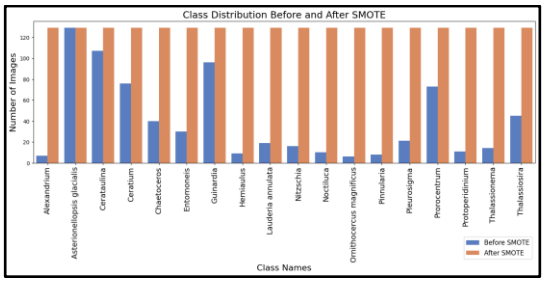
Gambar 6. Distribusi Kelas pada Dataset

Distribusi kelas mayoritas terbesar memiliki jumlah 162 pada Asterionellopsis Glacialis, sedangkan kelas minoritas terkecil berjumlah 7 pada Ornithocercus Magnificus. Kemudian, pembagian data dilakukan dengan rasio 80:20, di mana 80% data digunakan untuk pelatihan dan 20% untuk pengujian dengan hasil seperti pada Tabel 2.

Tabel 2. Jumlah Data *Train* dan *Test*

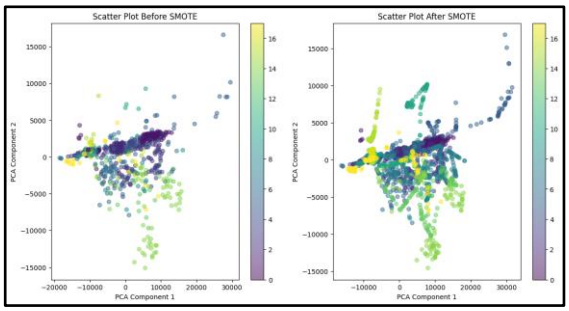
Nama Kelas	Jumlah Data *Train*	Jumlah Data *Test*
Alexandrium	7	2
Asterionellopsis glacialis	129	33
Cerataulina	107	27
Ceratium	76	19
Chaetoceros	40	10
Entomoneis	30	8
Guinardia	96	24
Hemiaulus	9	2
Lauderia annulata	19	5
Nitzschia	16	4
Noctiluca	10	2
Ornithocercus magnificus	6	1
Pinnularia	8	2
Pleurosigma	21	6
Prorocentrum	73	18
Protoperidinium	11	3
Thalassionema	14	3
Thalassiosira	45	11

Dari Tabel 2, kelas Asterionellopsis Glacialis memiliki jumlah sampel data latih terbanyak, yaitu 129 sampel. Kelas ini dijadikan acuan untuk menentukan jumlah sampel maksimal yang akan ditambahkan ke kelas minoritas pada data latih melalui teknik *oversampling* dengan interpolasi. Tujuannya adalah untuk menyeimbangkan distribusi data antar kelas dan meningkatkan akurasi model klasifikasi menggunakan SMOTE. Seperti yang terlihat pada Gambar 7, setelah SMOTE dilakukan masalah ketidakseimbangan kelas diatasi dengan menghasilkan distribusi data yang lebih merata di antara kelas-kelas minoritas dan mayoritas.



Gambar 7. Distribusi *Dataset* Setelah SMOTE

Selain itu, Gambar 8 menunjukkan bahwa hasil *scatter plot* sebelum SMOTE mencerminkan ketidakseimbangan kelas, dengan beberapa kelas yang kurang terwakili dan klaster warna yang jarang, sementara kelas lain lebih padat.



Gambar 8. *Scatter Plot* Sebelum dan Sesudah SMOTE

Hasil *scatter plot* setelah penerapan SMOTE, distribusi kelas menjadi lebih merata dengan peningkatan jumlah titik data pada kelas minoritas, meningkatkan kepadatan klaster dan keseimbangan data. Penerapan SMOTE dengan interpolasi terbukti membantu memperbaiki kinerja model klasifikasi dengan menambah sampel sintetis pada kelas yang kurang terwakili.

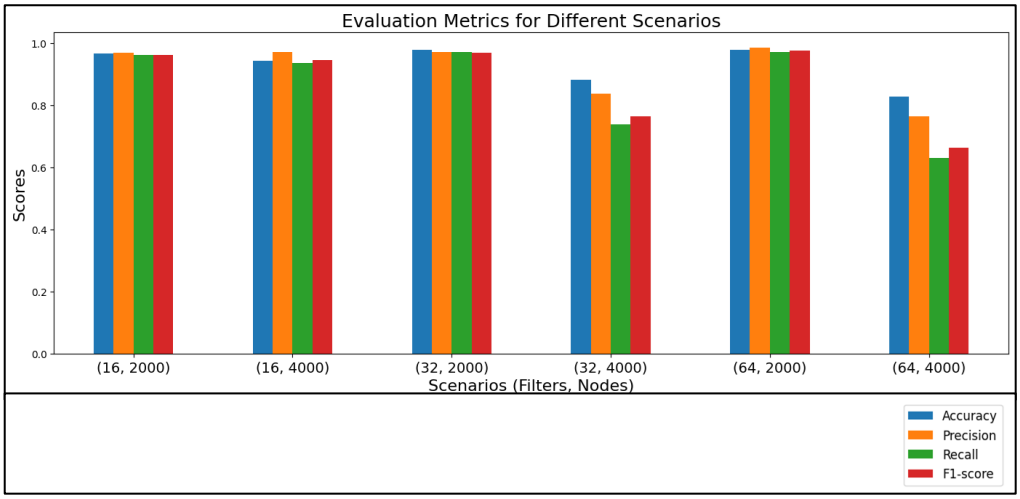
4.3. Hasil Pengujian

Setelah melakukan implementasi untuk membangun model dengan menggunakan algoritma hibrida CNN dan ELM, kemudian dilakukan skenario pengujian dengan menggunakan matriks evaluasi dan *confusion matrix* untuk mengukur kualitas dari hasil klasifikasi model. Pada proses pelatihan, dari 6 skenario pengujian diambil nilai terbaik untuk masing-masing percobaan sehingga didapatkan hasil seperti pada Tabel 3

Tabel 3. Hasil Skenario Pengujian

Skenario	*Accuracy*	*Precision*	*Recall*	*F1-score*	*Support*
Skenario 1	0.966667	0.968750	0.962963	0.963365	180
Skenario 2	0.944444	0.970623	0.936049	0.946885	180
Skenario 3	0.977778	0.970733	0.972222	0.970469	180
Skenario 4	0.883333	0.838301	0.739057	0.764651	180
Skenario 5	0.977778	0.985185	0.972222	0.976051	180
Skenario 6	0.827778	0.763907	0.630877	0.662699	180

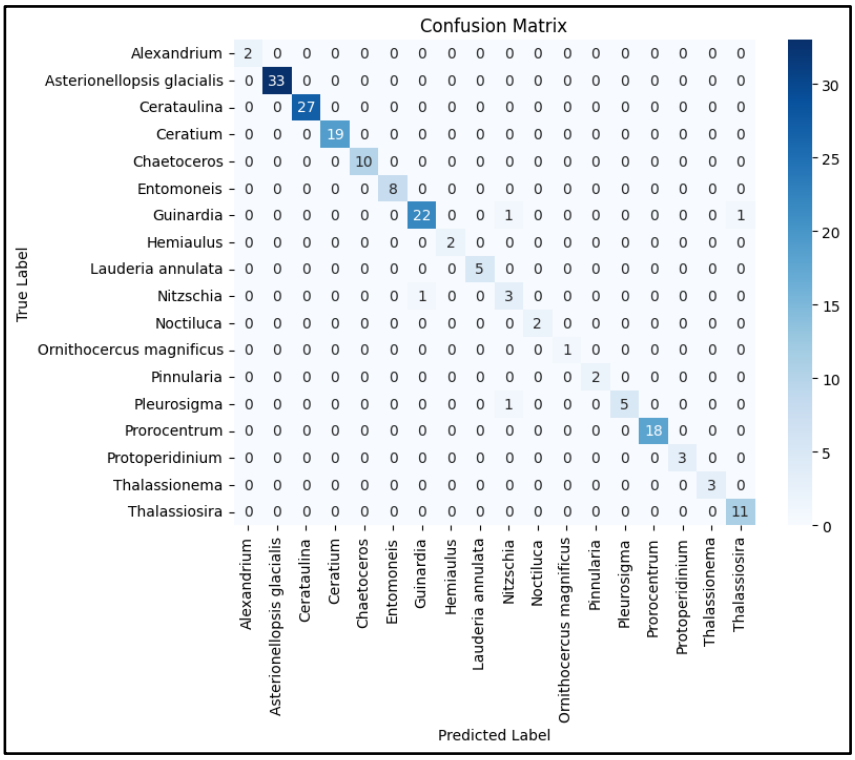
Pada Tabel 3, hasil evaluasi model CNN-ELM dengan berbagai konfigurasi *filter* dan *hidden node* menunjukkan bahwa konfigurasi dengan 32 *filter* dan 2000 *hidden node* serta 64 filter dan 2000 *hidden node* memberikan kinerja terbaik dengan akurasi sebesar 97,78%. Kedua konfigurasi ini juga menunjukkan nilai *precision*, *recall*, dan F1-*score* yang tinggi, menandakan bahwa model mampu mengklasifikasikan data dengan sangat baik pada kedua konfigurasi tersebut. Sebaliknya, konfigurasi dengan 64 *filter* dan 4000 *hidden node* memiliki performa terendah dengan akurasi 82,78%, yang diikuti oleh konfigurasi dengan 32 *filter* dan 4000 *hidden node* dengan akurasi 88,33%. Kedua konfigurasi ini juga menunjukkan penurunan yang signifikan dalam nilai *recall* dan F1-*score*, yang mengindikasikan bahwa model kurang mampu mengenali kelas-kelas dengan tepat ketika menggunakan jumlah *filter* dan *hidden node* yang lebih besar. Konfigurasi dengan 16 *filter* dan 2000 *hidden node* juga memberikan hasil yang cukup baik dengan akurasi 96,67%, sementara penambahan jumlah *hidden node* menjadi 4000 pada konfigurasi ini mengakibatkan sedikit penurunan performa dengan akurasi 94,44%.



Gambar 9. Hasil Perbadingan Nilai Matriks Evaluasi

Selain itu, dapat dilihat pada Gambar 9 bahwa nilai evaluasi matriks dengan model dengan 32 *filter* dan 2000 *hidden node* memberikan performa terbaik di antara semua skenario. Di sisi lain, peningkatan jumlah *node* menjadi 4000 pada model dengan 32 atau 64 *filter* justru menurunkan performa secara signifikan, terutama terlihat pada penurunan *recall* dan F1-*score*. Hasil ini menunjukkan bahwa konfigurasi dengan *filter* dan *node* yang

lebih kompleks tidak selalu menghasilkan performa yang lebih baik, melainkan dapat menyebabkan *overfitting* atau kesulitan dalam generalisasi. Secara keseluruhan, penggunaan 32 *filter* dan 2000 *hidden node* terbukti menjadi pilihan yang paling optimal untuk model CNN-ELM dalam klasifikasi citra plankton pada penelitian ini.



Gambar 9. *Confusion Matrix* Skenario 32 *Filter* dan 2000 *Hidden Node*

Pada Gambar 9, hasil *confusion matrix* untuk parameter terbaik dengan 32 *filter* dan 2000 *hidden node* menunjukkan kinerja model CNN-ELM dalam mengklasifikasikan 18 kelas plankton. Matriks ini menggambarkan seberapa sering prediksi model cocok dengan label sebenarnya. Beberapa kelas, seperti Asterionellopsis Glacialis, Cerataulina, dan Guinardia, memiliki tingkat klasifikasi yang sangat baik dengan sebagian besar prediksi berada di diagonal, menandakan akurasi tinggi. Namun, beberapa kelas lainnya seperti Alexandrium, Noctiluca, dan Nitzschia menunjukkan kesalahan klasifikasi yang lebih sering, dengan adanya prediksi yang tersebar di kolom yang tidak sesuai. Hasil ini menunjukkan bahwa sementara model ini cukup efektif untuk beberapa kelas, masih ada beberapa kelas yang perlu perbaikan dalam hal akurasi klasifikasi.

5. KESIMPULAN

a. Hasil dari implementasi SMOTE dengan interpolasi telah berhasil dalam menyelesaikan permasalahan dalam distribusi data yang tidak seimbang menjadi sama rata pada data latih. Selain itu, dari skenario pengujian yang dilakukan, konfigurasi model CNN-ELM dengan 32 *filter* dan 2000 *hidden nod*e serta 64 *filter* dan 2000 *hidden node* memberikan kinerja terbaik dengan akurasi 97,78% dan nilai *precision*, *recall*, serta F1-*score* yang tinggi.

b. Dari skenario pengujian, model dengan 64 *filter* dan 4000 *hidden node* memiliki performa terendah dengan akurasi 82,78%, disusul oleh model dengan 32 *filter* dan 4000 *hidden node* dengan akurasi 88,33%, menunjukkan penurunan signifikan dalam *recall* dan F1-*score*. Serta, penambahan jumlah *hidden node* dari 2000 menjadi 4000 cenderung menurunkan performa

model, khususnya pada model dengan 32 dan 64 *filter*, yang disebabkan oleh *overfitting* sehingga menunjukkan bahwa konfigurasi dengan *filter* dan *node* yang lebih kompleks tidak selalu menghasilkan performa yang lebih baik.

c. Hasil dari *confusion matrix* menunjukkan kinerja tinggi pada beberapa kelas, namun masih terdapat kesalahan klasifikasi yang sering terjadi pada kelas seperti *Alexandrium*, *Noctiluca*, dan *Nitzschia* yang perlu diperbaiki. Hal itu disebabkan karena kurangnya data citra pada kelas-kelas tersebut sehingga dapat ditambahkan metode augmentasi untuk menambahkan jumlah sampel citra sebelum di SMOTE.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Allah SWT yang telah melimpahkan rahmat dan hidayahnya sehingga penelitian ini dapat diselesaikan dengan baik. Selain itu penulis juga mengucapkan terimakasih kepada orang tua dan teman-teman penulis yang selalu memberikan dukungan dalam pengerjaan penelitian ini.

DAFTAR PUSTAKA

[1] A. C. Duxbury dan C. Cenedese, "Ocean | Definition, Distribution, Map, Formation, & Facts | Britannica." Diakses: 4 Maret 2024. [Daring]. Tersedia pada: https://www.britannica.com/science/ocean

[2] D. Dzikrina, M. Rahman, A. R. M. Akbar, dan R. Yunita, "KEANEKARAGAMAN PLANKTON DAN HUBUNGANNYA DENGAN KUALITAS AIR SUNGAI SEKITAR PABRIK KELAPA SAWIT (STUDI KASUS DI ASAM-ASAM, JORONG KABUPATEN TANAH LAUT)," *EnviroScienteae*, vol. 19, no. 4, Art. no. 4, Nov 2023, doi: 10.20527/es.v19i4.17784.

[3] S. Sunagawa *dkk.*, "Tara Oceans: towards global ocean ecosystems biology," *Nat. Rev. Microbiol.*, vol. 18, no. 8, hlm. 428–445, Agu 2020, doi: 10.1038/s41579-020-0364-5.

[4] G. Gorsky *dkk.*, "Digital zooplankton image analysis using the ZooScan integrated system," *J. Plankton Res.*, vol. 32, no. 3, hlm. 285–303, Mar 2010, doi: 10.1093/plankt/fbp124.

[5] A. Lumini dan L. Nanni, "Ocean Ecosystems Plankton Classification," dalam *Recent Advances in Computer Vision: Theories and Applications*, M. Hassaballah dan K. M. Hosny, Ed., dalam Studies in Computational Intelligence. , Cham: Springer International Publishing, 2019, hlm. 261–280. doi: 10.1007/978-3-030-03000-1_11.

[6] J. Sharma, O.-C. Granmo, dan M. Goodwin, "Deep CNN-ELM Hybrid Models for Fire Detection in Images," dalam *Artificial Neural Networks and Machine Learning – ICANN 2018*, V. Kůrková, Y. Manolopoulos, B. Hammer, L. Iliadis, dan I. Maglogiannis, Ed., dalam Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018, hlm. 245–259. doi: 10.1007/978-3-030-01424-7_25.

[7] D. Elreedy, A. F. Atiya, dan F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Mach. Learn.*, Jan 2023, doi: 10.1007/s10994-022-06296-4.

[8] D. Bhatt *dkk.*, "CNN Variants for Computer Vision: History, Architecture, Application, Challenges and Future Scope," *Electronics*, vol. 10, no. 20, Art. no. 20, Jan 2021, doi: 10.3390/electronics10202470.

[9] G.-B. Huang, Q.-Y. Zhu, dan C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1, hlm. 489–501, Des 2006, doi: 10.1016/j.neucom.2005.12.126.

[10] N. V. Chawla, K. W. Bowyer, L. O. Hall, dan W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res.*, vol. 16, hlm. 321–357, Jun 2002, doi: 10.1613/jair.953.

[11] A. Junaidi, C.-H. Lin, Y.-H. Tseng, L.-H. Chang, dan S.-C. Peng, "Gradient-Based Adaptive Image Super Resolution," dalam *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Jul 2019, hlm. 2774–2777. doi: 10.1109/IGARSS.2019.8900578.

[12] A. F. B. Sajiwo, B. Rahmat, dan A. Junaidi, "KLASIFIKASI INDEKS STANDAR PENCEMARAN UDARAN (ISPU) MENGGUNAKAN ALGORITMA XGBOOST DENGAN TEKNIK IMBALANCED DATA (SMOTE)," *J. Inform. Dan Tek. Elektro Terap.*, vol. 12, no. 3, Art. no. 3, Agu 2024, doi: 10.23960/jitet.v12i3.4699.

[13] E. C. Orenstein, O. Beijbom, E. E. Peacock, dan H. M. Sosik, "WHOI-Plankton- A Large Scale Fine Grained Visual Recognition Benchmark Dataset for Plankton Classification," 2 Oktober 2015, *arXiv*: arXiv:1510.00745. doi:

10.48550/arXiv.1510.00745.

[14] H. Bichri, A. Chergui, dan M. Hain, "Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets. | International Journal of Advanced Computer Science & Applications | EBSCOhost." Diakses: 14 Agustus 2024. [Daring]. Tersedia pada: https://openurl.ebsco.com/contentitem/doi:10.14569%2Fijacsa.2024.0150235?sid=ebsco:plink:crawler&id=ebsco:doi:10.14569%2Fijacsa.2024.0150235