

IMPLEMENTASI MASK R-CNN PADA PERHITUNGAN PERSENTASE TUTUPAN KARANG UNTUK MEMANTAU EKOSISTEM TERUMBU KARANG

Cut Nadilla Maret¹, Husaini^{*2}, Haekal Azief Haridhi³, Naufal Alkhalis⁴, Nur Fadli⁵, Yudi Hadit⁶, Muhammad Nanda⁷, Maria Ulfah⁸, Kris Handoko⁹, Intan Malayana¹⁰, Arsa Cindy Safitri¹¹

^{1,2,3,4,5,6,7,8,10,11}Universitas Syiah Kuala, Banda Aceh, ⁹Kementerian Kelautan dan Perikanan, Jakarta Pusat
Email: ¹cut_nm@mhs.usk.ac.id, ²husaini.muhammad@usk.ac.id, ³haekal.azief.haridhi@usk.ac.id, ⁴naufal_alk@mhs.usk.ac.id, ⁵nurfadli@usk.ac.id, ⁶yudi@usk.ac.id, ⁷m.nanda@usk.ac.id, ⁸mariaulfah@usk.ac.id, ⁹krishandoko.ahlimadya@gmail.com, ¹⁰intanmalayana2003@gmail.com, ¹¹cindyarsa07@gmail.com
*Penulis Korespondensi

(Naskah masuk: 26 Desember 2023, diterima untuk diterbitkan: 27 Agustus 2025)

Abstrak

Terumbu karang memiliki peranan penting bagi kehidupan di laut. Namun, ekosistem terumbu karang sangat rentan mengalami kerusakan karena sejumlah faktor seperti aktivitas manusia, perubahan iklim, lambatnya laju pertumbuhan dan sebagainya. Upaya pelestarian terumbu karang telah dilakukan, namun pemantauan masih minim. Oleh karena itu, pemantauan ekosistem terumbu karang perlu ditingkatkan untuk mengetahui kondisi terumbu karang sebenarnya. Persentaseutupan karang adalah indikator yang perlu diketahui sebagai penentuan tingkat kehidupan terumbu karang. Proses pemantauan terumbu karang saat ini masih dilakukan secara konvensional, sehingga proses pemantauan tidak efisien dan perolehan informasi mengenai persentaseutupan karang membutuhkan waktu yang panjang. Penelitian ini mengimplementasikan algoritma *Mask Region Convolutional Neural Network* (Mask R-CNN) pada *library* Detectron2 untuk melakukan deteksi dan segmentasi objekutupan karang pada ekosistem terumbu karang dengan menggunakan citra terumbu karang sebagai *input*. Model yang digunakan untuk segmentasi *instance* pada citra terumbu karang ini dilatih dengan menggunakan *backbone Residual Network* (ResNet) dan *Residual Networks Next* (ResNeXt) yang terdapat pada *library* Detectron2. Model *backbone* dievaluasi berdasarkan matriks presisi dan *recall*. Hasil penelitian menunjukkan ResNeXt101-FPN merupakan *backbone* terbaik dalam menghasilkan segmentasi. Hasil proses segmentasi tersebut kemudian digunakan untuk menghitung persentaseutupan karang. Berdasarkan hasil perhitungan, persentaseutupan karang dengan data yang diuji adalah sebesar 86,06%. Dengan demikian, proses perhitungan persentaseutupan karang untuk memantau ekosistem terumbu karang dapat dilakukan dengan efisien dan informasi mengenai persentaseutupan karang dapat diperoleh dalam waktu yang singkat.

Kata kunci: Terumbu Karang, Mask R-CNN, Segmentasi, Detectron2, Tutupan Karang

IMPLEMENTATION OF R-CNN MASK ON CORAL COVER PERCENTAGE CALCULATION FOR MONITORING CORAL REEF ECOSYSTEMS

Abstract

Coral reefs have an important role for life in the sea. However, coral reef ecosystems are very vulnerable to damage due to a number of factors such as human activities, climate change, slow growth rates and so on. Efforts to preserve coral reefs have been made, but monitoring remains minimal. Therefore, coral reef ecosystem monitoring needs to be enhanced to assess their actual condition. The percentage of coral cover is an indicator that needs to be known as a determination of the life rate of coral reefs. The current coral reef monitoring process is still carried out conventionally, so the monitoring process is inefficient and obtaining information about the percentage of coral cover takes a long time. This study implements the *Mask Region Convolutional Neural Network* (Mask R-CNN) algorithm in the *Detectron2* library to detect and segment coral cover objects in coral reef ecosystems using coral reef images as *input*. The model used for *instance* segmentation on coral reef images was trained using the *Residual Network* (ResNet) and *Residual Networks Next* (ResNeXt) backbones, which are available in the *Detectron2* library. The backbone model is evaluated based on precision and recall matrices. The results show that ResNeXt101-FPN is the best backbone in producing segmentation. The results of the segmentation process are then used to calculate the percentage of coral cover. Based on the calculation results, the percentage of coral cover with the tested data was 86.06%. Thus, the process of calculating the coral cover percentage to monitor coral reef ecosystems can be carried out efficiently and information about the coral cover percentage can be obtained in a short time.

Keywords: Coral Reef, R-CNN Mask, Segmentation, Detectron2, Coral Cover

1. PENDAHULUAN

Terumbu karang adalah sistem kompleks yang terbentuk oleh organisme penghasil kalsium karbonat bersama dengan makhluk hidup lainnya yang mendiami dasar laut dan wilayah perairannya (Tatipata dkk, 2019). Terumbu karang memiliki peranan yang besar dalam ekosistem laut baik secara fisiologi sebagai pemecah ombak dan secara ekologi sebagai tempat berlangsungnya kehidupan biota-biota laut (Rumkorem dkk, 2019). Namun, terumbu karang sangat rentan terhadap ancaman yang menyebabkan kehidupannya terganggu. Faktor ancaman ini dapat berupa ancaman global dan ancaman lokal (Jubaedah & Anas, 2019). Ancaman global disebabkan karena adanya perubahan kondisi bumi seperti perubahan iklim dan pengasaman laut. Ancaman lokal disebabkan karena faktor-faktor di sekitar ekosistem terumbu karang seperti polusi laut, metode penangkapan ikan yang merusak, serta kegiatan manusia lainnya yang dapat mengganggu kehidupan terumbu karang (Souter dkk, 2020).

Pentingnya peranan terumbu karang pada kehidupan di laut serta besarnya faktor ancamannya sehingga diperlukan upaya untuk mempertahankan kelestarian ekosistem terumbu karang secara efisien dan efektif. Sebagai upaya mempertahankan kehidupan terumbu karang perlu dilakukan pemantauan perubahan yang terjadi pada terumbu karang (Isdianto dkk, 2022). Aktivitas pemantauan dilakukan untuk menilai keadaan terumbu karang dengan melihat persentase tutupan karang (Alamsyah dkk, 2019). Indikator untuk mengetahui tingkat kehidupan ekosistem terumbu karang dilihat berdasarkan luas persentase tutupan karang (Raditya Kesatria Sinaga dkk, 2023). Pendekatan persentase tutupan karang hidup digunakan sebagai indikasi atau petunjuk untuk menduga kondisi terumbu karang (Yunilda dkk, 2022). Terdapat hubungan antara tutupan karang hidup dengan masalah kesehatan terumbu karang (Mahatir dkk, 2022).

Metode pemantauan terumbu karang dilakukan oleh konservasionis dengan melakukan penyelaman langsung dan pengambilan gambar terumbu karang di laut. Gambar-gambar tersebut digunakan sebagai data untuk mendapatkan informasi persentase tutupan karang. Perhitungan persentase tutupan karang yang dilakukan saat ini menggunakan sebuah aplikasi dengan gambar-gambar terumbu karang sebagai *input*. Selanjutnya, aplikasi tersebut akan memberikan titik-titik acak di seluruh area gambar-gambar tersebut. Berikutnya titik-titik acak tersebut diidentifikasi secara manual titik per titik pada semua gambar. Titik-titik yang berada pada bagian tutupan karang akan diidentifikasi sebagai karang sedangkan titik-titik yang berada pada bagian bukan karang akan diidentifikasi sebagai bukan karang. Proses tersebut membutuhkan waktu yang relatif panjang dan tidak efisien untuk mendapatkan informasi tutupan karang sehingga dapat menghambat proses pemantauan terumbu karang.

Untuk menyelesaikan permasalahan tersebut penerapan *deep learning* menjadi salah satu solusi dan inovasi yang dapat diterapkan. *Deep learning* memiliki keunggulan dalam menangani data yang tidak terstruktur dan memungkinkan model melakukan komputasi untuk memahami ciri-ciri dari data yang digunakan (Ahmad dkk, 2019). Salah satu arsitektur *deep learning* yang telah dikembangkan adalah Algoritma *Mask Region Convolutional Neural Network* (Mask R-CNN). Mask R-CNN merupakan arsitektur jaringan saraf tiruan dengan segmentasi *instance* yang mampu memisahkan objek dan memberikan masker piksel untuk setiap objek terpisah (Bellynza & Syaputra, 2022). Mask R-CNN menemukan objek dengan kotak pembatas (*bounding box*) dan segmentasi objek dengan masker piksel (Budi & Setyadi, 2023).

Mask R-CNN pada *library* Detectron2 yang dikembangkan oleh *Facebook Artificial Intelligence Research* (FAIR) mendukung dataset dan model *custom* serta memiliki *backbone* yang telah dilatih sebelumnya dengan dataset yang besar sehingga memiliki kinerja yang unggul pada permasalahan tugas visi komputer seperti deteksi dan segmentasi *instance* (Akshayanivashini & Krisvanth, 2023).

Penelitian terkait dengan implementasi Mask R-CNN telah dilakukan oleh (Alkhalis dkk, 2024) pada proses perhitungan tinggi dan lebar terumbu karang. Namun, penelitian ini hanya menerapkan Mask R-CNN untuk mengukur laju pertumbuhan karang transplantasi melalui perhitungan tinggi dan lebar karang.

Pada penelitian ini, Mask R-CNN dengan memanfaatkan variasi *backbone* ResNet dan ResNeXt yang terdapat pada *library* Detectron2 digunakan untuk membangun model yang mampu melakukan deteksi dan segmentasi *instance* tutupan karang berupa masker piksel. Dengan memanfaatkan *backbone* tersebut, model yang dibangun memiliki kinerja yang optimal dan mampu melakukan segmentasi dengan baik sehingga dapat digunakan untuk menghitung persentase tutupan karang.

Penelitian ini memberikan keterbaharuan pada proses perhitungan persentase tutupan karang untuk memantau ekosistem terumbu karang. Disamping itu, hasil penelitian ini diharapkan dapat menjadi solusi terhadap permasalahan terkait keterbatasan sumber daya serta mempersingkat waktu yang diperlukan untuk memperoleh informasi hasil perhitungan persentase tutupan karang.

1. METODE PENELITIAN

Metode yang dilakukan untuk menyelesaikan penelitian ini terdiri dari tahap pengumpulan dataset, pembagian dataset, anotasi dataset, implementasi model, analisis model dan menghitung persentase tutupan karang.

2.1. Pengumpulan Data

Penelitian ini menggunakan data yang diambil langsung dari lokasi penelitian agar penerapan model yang dibangun relevan dengan keadaan di lokasi penelitian, yaitu di perairan Pulau Gosong, Aceh Barat Daya. Pengambilan data gambar terumbu karang dilakukan pada pagi dan siang hari untuk mendapatkan pencahayaan yang cukup. Pengambilan data menggunakan kamera bawah air Olympus PT-059 *Underwater Camera* yang menghasilkan data citra berukuran 4000 x 3000 piksel. Pengambilan gambar dilakukan dengan memperhatikan jarak kamera, yaitu 100 cm dari objek. Contoh hasil pengumpulan data dapat dilihat pada Gambar 1.



Gambar 1. Contoh citra ekosistem terumbu karang

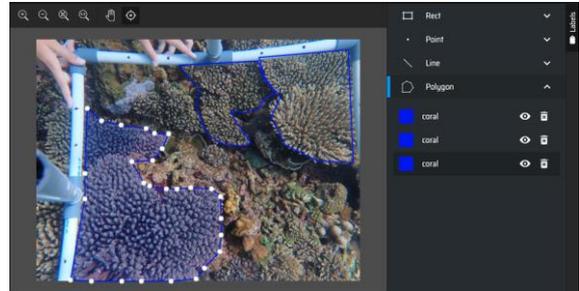
2.2. Pembagian Dataset

Pembagian dataset adalah proses membagi data menjadi beberapa *subset* atau bagian. Penelitian ini menerapkan pembagian dataset menjadi bagian pelatihan, validasi, dan pengujian. Setiap bagian dengan rasio masing-masing 80:10:10 untuk data pelatihan:validasi:pengujian.

2.3. Anotasi Dataset

Detectron2 yang digunakan pada penelitian ini menerima data *Common Objects in Context* (COCO) untuk digunakan dalam tahap implementasi model. Anotasi dilakukan agar informasi sesuai dengan format COCO dan data gambar dapat digunakan. Model sangat dipengaruhi oleh variasi gambar pada dataset dan proses anotasi gambar di dalam dataset (Khatib Sulaiman dkk, 2023). Data pada masing-masing bagian ditempatkan pada folder yang terpisah. Anotasi dataset dilakukan pada bagian dataset pelatihan dan dataset validasi. Proses anotasi dataset dilakukan menggunakan aplikasi MakeSense.ai dengan menggunakan digitasi *polygon* untuk menentukan area *ground truth*. Titik-titik *polygon* dihubungkan di sekeliling area wilayah yang

ditentukan sebagai objek (Agnew dkk, 2023). Proses anotasi dataset dapat dilihat pada Gambar 2.



Gambar 2. Proses anotasi

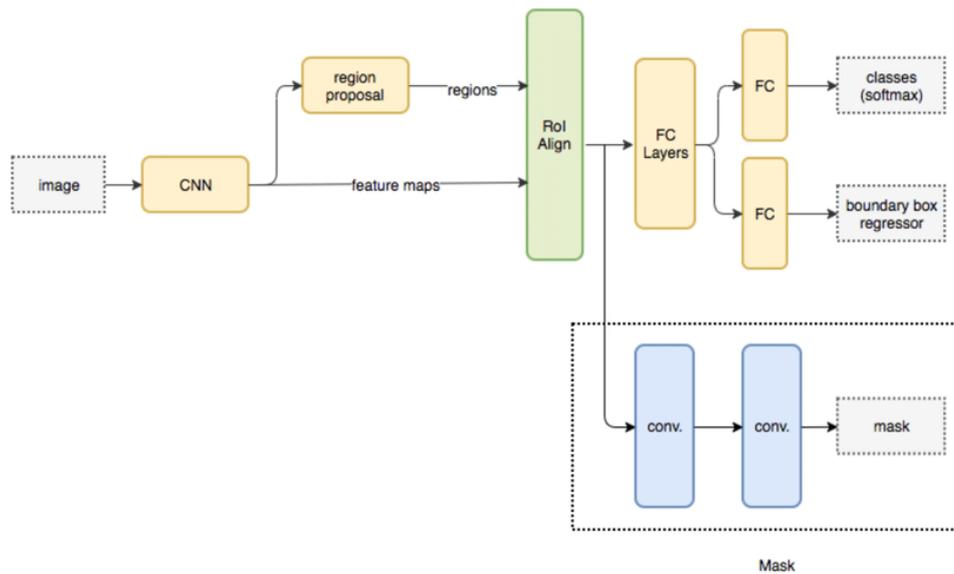
2.4. Implementasi Model

Gambar 3 menunjukkan proses yang dilakukan pada Mask R-CNN yang diawali dengan menerima gambar sebagai masukan. Gambar akan melalui sebagai lapisan konvolusional. Dalam penelitian ini, digunakan *backbone* ResNet dan ResNext yang telah tersedia pada Detectron2. *Backbone* sebagai ekstraktor fitur, memungkinkan model untuk memahami pola-pola kompleks dalam gambar. Proses konvolusi yang terjadi menghasilkan *feature map* yang merupakan representasi visual yang akan digunakan pada tahap *Region Proposal Network* (RPN).

Langkah berikutnya dalam proses ini melibatkan RPN sebagai pendorong untuk mengusulkan wilayah yang dianggap sebagai objek. RPN melakukan pemeriksaan pada setiap titik pada *feature map* dengan tujuan menentukan keberadaan objek pada titik tersebut. Pada tahap ini, *anchor* mendapatkan wilayah berdasarkan nilai *Intersection over Union* (IoU) yang digunakan untuk menentukan suatu wilayah pada gambar masukan dapat dianggap sebagai objek atau *proposed region*.

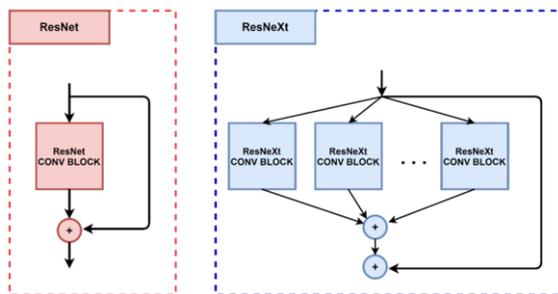
Selanjutnya, *proposed region* melalui proses *Region of Interest* (RoI) *Align* yang berfungsi untuk mengekstraksi fitur yang akan diteruskan pada tahap berikutnya yaitu, *Fully Convolutional Network* (FCN). Tahap FCN dalam proses arsitektur Mask R-CNN menghasilkan *output* klasifikasi objek, kotak pembatas objek dan segmentasi objek.

Penelitian ini menggunakan *backbone* ResNet dan ResNext yang telah disediakan pada *baseline* segmentasi *instance* COCO pada Detectron2. ResNet mempergunakan *shortcut connections* atau *skip connections* melalui blok-blok yang dikenal sebagai *residual blocks*. ResNeXt memperluas prinsip ResNet dengan mengenalkan struktur yang disebut *cardinality* dalam arsitekturnya. Pada ResNeXt, konsep *cardinality* mengacu pada jumlah jalur paralel atau *cardinalities* yang membentuk satu blok.



Gambar 3. Ilustrasi Arsitektur Mask R-CNN (Gonzalez dkk, 2019)

Di dalam blok ResNeXt, informasi melalui beberapa jalur paralel. Pendekatan ini memungkinkan pengaturan kapasitas jaringan dengan lebih efisien tanpa menambah jumlah parameter secara linier, seperti yang ditunjukkan pada Gambar 4 (Zhou dkk, 2021).



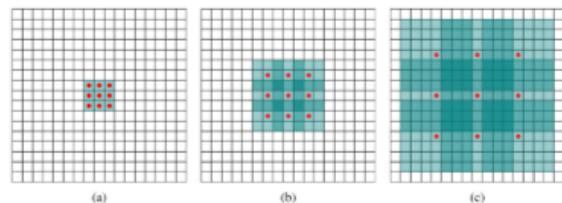
Gambar 4. Ilustrasi ResNet dan ResNext (de Carvalho dkk, 2021)

Penelitian ini menggunakan tujuh *backbone* yaitu ResNet50-FPN, ResNet50-DC5, ResNet50-C4, ResNet101-FPN, ResNet101-DC5, ResNet101-C4, dan ResNeXt101-FPN. Setiap komposisi modul konvolusional terdiri dari banyak lapisan yang berbeda-beda tergantung pada konfigurasi yang digunakan pada kedalaman lapisan blok residual (*residual block*) dalam jaringannya. ResNet-50 menggunakan 50 lapisan blok residual, ResNet-101 menggunakan 101 lapisan blok residual, dan ResNeXt101 menggunakan 101 lapisan blok residual dengan *cardinality*. Dalam blok ResNeXt mengizinkan informasi untuk mengalir melalui beberapa jalur paralel di dalam satu blok (Ma dkk, 2022).

Perbedaan berikutnya antara setiap *backbone* terletak pada kombinasi komponen *Conv-4* (C4), *Dilated Conv-5* (DC5), dan *Feature Pyramid Network* (FPN). Dalam arsitektur ResNet, *Conv4* (C4) berperan sebagai lapisan konvolusi keempat yang

mengekstraksi fitur-fitur kompleks dari gambar. C4 menerapkan konvolusi pada representasi fitur yang telah dibentuk oleh blok residual sebelumnya, membantu model dalam memahami pola dan fitur yang semakin kompleks. Peran penting C4 adalah kemampuannya untuk menangani informasi yang kompleks dan abstrak, serta membantu model dalam memahami konteks dan pola yang lebih mendalam. Dilasi pada DC5 merupakan proses menyisipkan ruang di antara elemen-elemen filter konvolusional, memungkinkan filter memiliki bidang reseptif yang lebih besar tanpa meningkatkan jumlah parameter (Lei dkk, 2019).

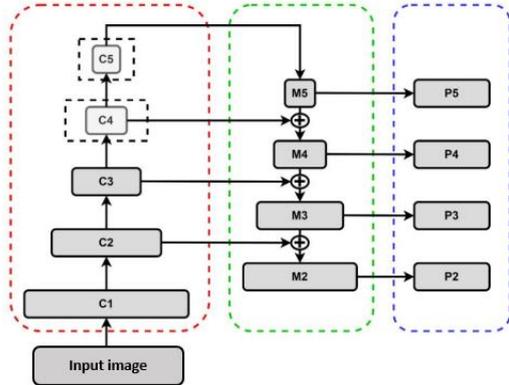
Gambar 5 menunjukkan proses rekonstruksi dari ResNet yang telah mengalami dilasi. Serupa dengan ResNet asli, namun disini *max pooling* dihilangkan dan digantikan oleh operasi *pooling* lainnya, sementara dilasi diterapkan pada blok konvolusi kelima. Selain itu, FPN digunakan dalam *backbone* untuk mengatasi tantangan variasi skala.



Gambar 5. Dilasi (Lei dkk, 2019)

Gambar 6 memperlihatkan struktur *bottom-up* pada bagian garis putus-putus merah, sementara bagian *top-down*, yang berfungsi sebagai ekstraktor fitur, menampilkan empat modul (M5, M4, M3 dan M2). Setiap modul ini mengalami proses penggabungan ganda dari satu modul ke modul berikutnya. Selain berfungsi sebagai ekstraktor fitur, setiap modul pada bagian *top-down* juga memberikan prediksi (P5, P4, P3 dan P2). Prediksi ini digunakan dalam RPN, mengilustrasikan kompleksitas dan

keterkaitan antara bagian-bagian dalam keseluruhan arsitektur.



Gambar 6. Struktur *bottom-up* dan *top-down* (de Carvalho dkk, 2021)

2.5. Analisis Model

Setiap model dari ketujuh *backbone* dievaluasi berdasarkan hasil prediksi model. Evaluasi dilakukan dengan memeriksa presisi dan *recall* dari setiap model yang menggunakan setiap *backbone*. Hal ini bertujuan untuk memahami seberapa baik setiap model mampu melakukan deteksi objek dengan tingkat ketepatan dan kemampuan untuk mengingat kembali (*recall*) objek. Dalam konteks model segmentasi *instance*, dua metrik utama yang diperhatikan adalah presisi (Persamaan (1)) dan *recall* (Persamaan (2)). Presisi mengukur jumlah instan positif yang diidentifikasi dengan benar (TP) dibagi oleh total prediksi yang positif (TP + FP), di mana FP adalah jumlah instan negatif yang salah diidentifikasi sebagai positif oleh model. *Recall* mengukur jumlah instan positif yang diidentifikasi dengan benar dibagi oleh total jumlah instan positif yang seharusnya terdeteksi (TP + FN), di mana FN adalah jumlah instan positif yang tidak terdeteksi atau salah diidentifikasi sebagai negatif oleh model.

$$\text{Presisi} = \frac{TP}{(TP+FP)} \quad (1)$$

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (2)$$

2.6. Menghitung Persentase Tutupan Karang

Perhitungan persentase tutupan karang dilakukan dengan menghitung area masker piksel yang menunjukkan objek karang dibanding dengan keseluruhan area pada gambar. Untuk memperoleh persentase kemudian perbandingan dikali dengan 100, menunjukkan persentase area karang.

Perhitungan ini dimulai dengan mengambil masker piksel pada gambar hasil *output* implementasi model dengan fungsi prediksi masker kemudian dilakukan perhitungan piksel pada masker yang diambil. Keseluruhan piksel dari setiap masker pada gambar dijumlahkan. Hasil penjumlahan setiap piksel pada keseluruhan masker disimpan pada variabel *total_coral_area*. *Coral mask* adalah tensor dengan

dimensi N, H, W , dimana N adalah jumlah gambar, H (*height*) adalah tinggi setiap gambar dan W (*width*) adalah lebar setiap gambar. Di mana *coral mask* $_{n,i,j}$ adalah nilai dari mask di posisi (i,j) pada gambar ke- n (Persamaan (3)).

$$\text{total coral area} = \sum_{n=1}^N \sum_{i=1}^H \sum_{j=1}^W \text{coral mask}_{n,i,j} \quad (3)$$

Berikutnya dilakukan perhitungan jumlah keseluruhan piksel sebagai total area. Hasil perhitungan ini disimpan pada variabel *total_image_area*. H_n adalah tinggi dari gambar ke- n , W_n adalah lebar dari gambar ke- n , dan N adalah jumlah total gambar dalam dataset. (Persamaan (4)). Kemudian perhitungan persentase akan membandingkan variabel *total_image_area* dan *total_image_area* dikali dengan seratus persen dan disimpan pada variabel *coral coverage* (Persamaan (5)).

$$\text{total image area} = \sum_{n=1}^N W_n \times H_n \quad (4)$$

$$\text{coral coverage} = \frac{\text{total coral area}}{\text{total image area}} \times 100\% \quad (5)$$

Coral coverage menunjukkan persentase tutupan coral yang terdapat dalam satu perhitungan persentase tutupan karang. Pada implementasi yang dilakukan, perhitungan persentase tutupan karang terhadap sejumlah gambar yang diambil pada satu area transek, sehingga terdapat beberapa gambar yang akan dilihat rata-rata tutupan karang.

2. HASIL DAN PEMBAHASAN

Penelitian ini dimulai dengan pembagian dataset sebanyak 358 data gambar bawah air ekosistem terumbu karang yang terdiri dari tiga partisi dataset yaitu data pelatihan, data validasi dan data pengujian dengan rasio perbandingan 80:10:10. Pembagian dataset dapat dilihat pada Tabel 1.

bagian data	jumlah gambar
Pelatihan	286
Validasi	36
Pengujian	36

Tahap selanjutnya dilakukan anotasi terhadap data pelatihan dan data validasi. *Output* dari proses anotasi berupa file COCO JSON untuk masing-masing partisi data yaitu data pelatihan dan data validasi. File COCO JSON terdiri dari informasi gambar setelah dilakukan anotasi. Informasi yang terdapat pada file COCO JSON berupa ID data, lebar dan tinggi gambar, kategori objek, dan koordinat dari anotasi yang diterapkan pada setiap gambar. COCO JSON dari data pelatihan dan data validasi dilakukan *register* dengan menggunakan fungsi *register* pada

Detectron2 yang menghasilkan metadata untuk digunakan pada proses pelatihan model.

Pada penelitian ini menunjukkan bahwa dibutuhkan waktu komputasi yang relatif berbeda pada saat dilakukan proses pelatihan yang menggunakan *Graphics Processing Unit* (GPU) V100 pada Google Colaboratory Pro Plus. Waktu komputasi pelatihan setiap *backbone* dapat dilihat pada Tabel 2.

Tabel 2. Waktu Pelatihan

<i>backbone</i>	total waktu komputasi (menit)	waktu per iterasi (detik/it)
ResNet50-C4	09,10	0,5515
ResNet50-DC5	09,16	0,5578
ResNet50-FPN	09,33	0,5750
ResNet101-C4	07,57	0,4784
ResNet101-DC5	08,43	0,5244
ResNet101-FPN	09,36	0,5779
ResNeXt101-FPN	10,49	0,6505

Rata-rata waktu komputasi dari pelatihan yaitu selama sekitar 10 menit, pelatihan model dengan menggunakan *backbone* ResNet101-C4 mencapai waktu pelatihan paling singkat, yaitu pada 7,57 menit. Waktu pelatihan paling singkat berikutnya disusul *backbone* ResNet101-DC5 pada 8,43 menit. Ini menunjukkan bahwa proses pelatihan dengan kedua *backbone* tersebut dibutuhkan waktu komputasi yang lebih sedikit jika dibandingkan dengan *backbone* lainnya. Hal ini menjadi pertimbangan efisiensi model terbaik pada penelitian ini. *Backbone* ResNeXt101-FPN merupakan *backbone* dengan waktu komputasi pelatihan paling lama mencapai 10,49 menit. Antara *backbone* ResNet101-C4 dan ResNeXt101-FPN selisih total waktu komputasi keduanya adalah 2,87 menit dan selisih waktu per iterasi adalah 0,1721 detik/it.

Pada penelitian ini setiap *backbone* menunjukkan ukuran model yang berbeda. Ukuran model terkecil yang dihasilkan dengan *backbone* ResNet50-C4 dengan ukuran model 273,80 MB dan ukuran model terbesar dengan *backbone* ResNet101-DC5 dengan ukuran model 1491,37 MB. Ukuran setiap model yang dihasilkan dengan menggunakan setiap *backbone* ditunjukkan pada Tabel 3.

Tabel 3. Ukuran Model Setiap Backbone

<i>backbone</i>	ukuran model (MB)
ResNet50-C4	273,80
ResNet50-DC5	1342,89
ResNet50-FPN	342,83
ResNet101-C4	422,28
ResNet101-DC5	1491,37
ResNet101-FPN	491,35
ResNeXt101-FPN	836,16

Implementasi model dilakukan dengan menerapkan algoritma Mask R-CNN menggunakan

backbone diperoleh hasil evaluasi yang dapat dilihat pada Tabel 4. Hasil metrik rata-rata presisi atau *Average Precision* (AP) model dengan masing-masing *backbone*, pada AP hasil dari segmentasi yang telah dilakukan, *backbone* ResNeXt101-FPN menunjukkan hasil segmentasi *mask* dan *box* terbaik dari keseluruhan *backbone* yang digunakan dengan dataset yang sama. Pada urutan kedua ResNet101-C4 menunjukkan hasil terbaik setelahnya. Pada AP50 dimana rata-rata presisi berdasarkan nilai *threshold* IoU sebesar 0,5 menunjukkan segmentasi *mask* dan *box* terbaik dihasilkan ResNeXt101-FPN dan disusul ResNet101-C4. Pada AP75 dimana nilai IoU sebesar 0,75 hasil terbaik dihasilkan ResNeXt101-FPN dan masih disusul ResNet101-C4 setelahnya.

Pada hasil metrik rata-rata *recall* (AR) dengan maxDets 100, ResNet101-C4 memberikan nilai AR terbaik dengan hasil 43,8 *mask* dan 42,1 *box*, kemudian dengan hasil AR 41,8 *mask* dan 39,5 *box* disusul ResNeXt101-FPN. *Backbone* terbaik yang dilihat dengan metrik rata-rata *recall* berbeda dengan *backbone* terbaik yang dilihat pada metrik rata-rata presisi. Namun, baik dari metrik rata-rata presisi dan rata-rata *recall* masih menyajikan kedua *backbone* tersebut sebagai hasil terbaik. Berdasarkan evaluasi metrik rata-rata presisi dan rata-rata *recall* pada pengujian keseluruhan *backbone* pada penelitian ini diperoleh *backbone* terbaik yang menghasilkan model untuk melakukan segmentasi *instance* terhadap objek karang adalah ResNeXt101-FPN dan ResNet101-C4.

Tabel 4. Hasil Evaluasi

<i>backbone</i>	tipe	AP	AP50	AP75	AR100
ResNet50-C4	Mask	19,1	35,7	20,1	38,2
	Box	19,8	33,5	22,1	38,2
ResNet50-DC5	Mask	22,2	41,4	23,5	39,1
	Box	21,8	36,0	23,0	37,7
ResNet50-FPN	Mask	22,0	36,3	26,2	34,8
	Box	22,5	36,1	26,8	34,8
ResNet101-C4	Mask	25,6	44,9	27,8	43,8
	Box	25,2	43,5	24,6	42,1
ResNet101-C5	Mask	21,2	38,5	22,8	41,9
	Box	21,0	36,5	22,4	41,0
ResNet101-FPN	Mask	23,3	40,4	24,1	41,0
	Box	22,9	39,8	25,5	39,4
ResNeXt101-FPN	Mask	29,1	46,4	31,5	41,4
	Box	28,3	45,8	30,8	39,5

Hasil dari proses pelatihan menghasilkan model untuk setiap *backbone* yang digunakan. Dengan menggunakan model dari setiap *backbone* yang ada kemudian dilakukan pengujian segmentasi pada gambar. Pengujian kemudian dihasilkan visualisasi segmentasi sebagai *output*. Wilayah pada gambar yang diidentifikasi sebagai objek ditunjukkan dengan adanya penambahan masker. Hasil visualisasi dapat dilihat pada Tabel 5. Setiap objek yang dianggap sebagai karang akan ditunjukkan dengan masker biner dan *bounding box* yang meyakinkan bahwa objek adalah kelas yang diinginkan. Dari hasil pengujian yang dilakukan pada gambar yang sama untuk setiap model dapat dilihat ketepatan masker

dalam melakukan tugas regresi segmentasi objek. Ketepatan ini dinilai dari benar atau tidaknya masker yang divisualisasikan untuk menunjukkan objek karang. Hasil visualisasi masker karang pada gambar berbanding lurus dengan hasil evaluasi metrik rata-rata presisi (AP). *Output* yang dihasilkan X101-FPN menunjukkan visualisasi segmentasi objek karang lebih baik untuk rata-rata keseluruhan karang jika dibandingkan dengan hasil visualisasi masker yang dihasilkan pada model lainnya.

Tabel 5. Visualisasi Hasil

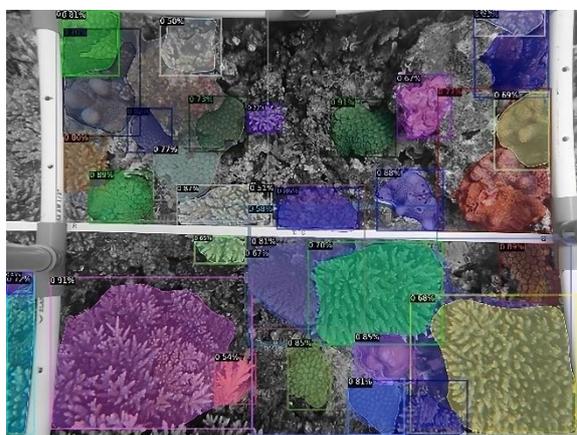
<i>backbone</i>	visualisasi
ResNet50-C4	
ResNet50-DC5	
ResNet50-FPN	
ResNet101-C4	

<i>backbone</i>	visualisasi
ResNet101-DC5	
ResNet101-FPN	
ResNeXt101-FPN	

Dengan merujuk penjelasan sebelumnya, diperoleh dua *backbone* terbaik yaitu ResNeXt101-FPN dan ResNet101-FPN. Penentuan *backbone* terbaik untuk penerapan segmentasi objek karang melihat perbandingan keduanya pada hasil *output* visualisasi masker yang dihasilkan. Visualisasi yang dihasilkan dengan pengujian gambar pada model dengan ResNeXt101-FPN menunjukkan hasil visualisasi lebih baik dibanding ResNet101-FPN. Hal ini ditunjukkan dengan ketepatan masker pada objek karang yang terdapat pada gambar. Hasil ini juga didukung dengan nilai tertinggi pada AP, AP50 dan AP75. Penerapan *backbone* ResNeXt101-FPN ini dilanjutkan dalam melakukan perhitungan persentase tutupan karang.

Perhitungan persentase tutupan karang dilakukan dengan menerapkan fungsi prediksi masker mengambil masker pada gambar hasil *output*. Kemudian dilakukan perhitungan piksel pada masker yang diambil. Keseluruhan piksel dari setiap masker pada satu gambar dijumlahkan. Hasil penjumlahan setiap piksel pada keseluruhan masker disimpan pada variabel ``coral_total_area``. Berikutnya dilakukan perhitungan jumlah keseluruhan piksel sebagai total

area. Hasil perhitungan ini kemudian disimpan pada variabel `total_image_area`. Nilai pada variabel `coral_total_area` banding nilai pada variabel `total_image_area` memberikan hasil nilai total piksel masker pada gambar uji. Hasil visualisasi dari penerapan perhitungan persentase dapat dilihat pada Gambar 7. Dengan menggunakan sumber daya yang tersedia pada saat dilakukan penelitian ini waktu yang dihabiskan untuk melakukan pengujian melakukan perhitungan membutuhkan waktu sekitar 32 detik per gambar dan menghasilkan informasi persentase tutupan karang dengan hasil persentase tutupan karang sebesar 86,06%. Hasil pengujian menunjukkan bahwa segmentasi *instance* menggunakan algoritma Mask R-CNN dengan menerapkan *backbone* ResNeXt101-FPN dapat diimplementasikan untuk perhitungan luasan tutupan karang.



Gambar 7. Visualisasi hasil

3. KESIMPULAN

Dari hasil penelitian yang telah dilakukan, diperoleh bahwa algoritma Mask R-CNN menggunakan *backbone* yang terdapat pada *library* Detectron2 dapat digunakan untuk deteksi dan segmentasi tutupan karang. Oleh karena itu, Mask R-CNN berhasil diimplementasikan dalam perhitungan persentase tutupan karang dengan menggunakan gambar ekosistem karang sebagai *input*. Dengan menggunakan arsitektur *backbone* ResNeXt101-FPN dihasilkan AP 29,1 untuk *mask* dan 28,3 untuk *box*. Dengan total waktu komputasi 10,49 menit dan waktu per iterasi 0,605 detik per iterasi pada proses pelatihan model, dan 32 detik untuk melakukan penerapan model menghitung persentase tutupan karang pada *input* gambar. Model berhasil melakukan perhitungan persentase terumbu karang dengan menunjukkan hasil persentase luas tutupan karang sebesar 86,06%. Dengan demikian penelitian ini dapat menjadi solusi permasalahan terkait keterbatasan sumber daya dan waktu dalam aktivitas pemantauan ekosistem terumbu karang. Penelitian ini memberikan kontribusi dalam pengembangan penerapan teknologi dan metode yang efisien dalam perhitungan

persentase tutupan karang untuk memantau ekosistem terumbu karang.

Saran untuk penelitian selanjutnya adalah menambah dataset citra ekosistem terumbu karang dari lokasi dan variasi yang lebih beragam agar model dapat mengenali berbagai citra terumbu karang selain dari lokasi ekosistem yang dipelajari saat ini. Upaya peningkatan hasil evaluasi model dapat dilakukan dengan meningkatkan nilai presisi dan *recall* melalui peningkatan kualitas dan kuantitas dataset. Selain itu, melatih model dengan algoritma segmentasi *deep learning* lainnya agar dapat memberikan pemahaman yang lebih mendalam terhadap hasil yang diperoleh. Melakukan proses kuantisasi pada model juga dianjurkan untuk mengurangi ukuran model dan meningkatkan efisiensi komputasi, sehingga mengurangi beban komputasi dan mempercepat proses pemrosesan.

DAFTAR PUSTAKA

- AGNEW, C., EISING, C., DENNY, P., SCANLAN, A., VAN DE VEN, P. AND GRUA, E.M., 2023. Quantifying the Effects of Ground Truth Annotation Quality on Object Detection and Instance Segmentation Performance. IEEE Access, 11, pp.25174–25188. <https://doi.org/10.1109/ACCESS.2023.3256723>.
- AHMAD, J., FARMAN, H. AND JAN, Z., 2019. Deep Learning Methods and Applications. In: SpringerBriefs in Computer Science. Springer. pp.31–42. https://doi.org/10.1007/978-981-13-3459-7_3.
- AKSHAYANIVASHINI AND KRISVANTH, 2023. Deep Learning-Based Instance Segmentation of Aircraft in Aerial Images using Detectron2. [online] Available at: <<https://ssrn.com/abstract=4485468>>.
- ALAMSYAH, R., PERMATASARI, A., PRODI, N., SUMBER, M., PERAIRAN, D., TINGGI, S., PERTANIAN, I. AND SINJAI, M., 2019. Sebaran dan Luasan Terumbu Karang di Perairan Pulau Larearea Menggunakan Citra Landsat 8. Jurnal Agrominansia, .
- ALKHALIS, N., HUSAINI, H., HAEKAL AZIEF HARIDHI, MARETNA, C.N., NUR FADLI, HADITIAH, Y., NANDA, M., ULFAH, M., KRIS HANDOKO, INTAN MALAYANA AND ARSA CINDY SAFITRI, 2024. Implementasi Mask R-CNN pada Perhitungan Tinggi dan Lebar Karang untuk Memantau Pertumbuhan Transplantasi Karang. Jurnal Teknologi Informasi dan Ilmu Komputer, 11(3), pp.603–614. <https://doi.org/10.25126/jtik.938374>.
- BELLYNZA, K.A. AND SYAPUTRA, H., 2022. Objek Deteksi Burung Lovebird Menggunakan Instance Segmentation Mask R-CNN. 4(1).
- BUDI, R., HARIANTO, R.A. AND SETYATI, E.,

2023. Segmentasi Citra Area Tumpukan Sampah Dengan Memanfaatkan Mask R-CNN. *Journal of Intelligent System and Computation*, [online] 5(1), pp.58–64. <https://doi.org/10.52985/insyst.v5i1.305>.
- DE CARVALHO, O.L.F., DE CARVALHO, O.A., ALBUQUERQUE, A.O. DE, BEM, P.P. DE, SILVA, C.R., FERREIRA, P.H.G., DE MOURA, R.D.S., GOMES, R.A.T., GUIMARÃES, R.F. AND BORGES, D.L., 2021. Instance segmentation for large, multi-channel remote sensing imagery using mask-RCNN and a mosaicking approach. *Remote Sensing*, 13(1), pp.1–24. <https://doi.org/10.3390/rs13010039>.
- GONZALEZ, S., ARELLANO, C. AND TAPIA, J., 2019. DeepBlueBerry: Quantification of Blueberries in the Wild Using Instance Segmentation. <https://doi.org/10.1109/ACCESS.2017.DOI>.
- ISDIANTO, A., FAIRUZ HAYKAL, M., MAHARDIKA PUTRI, B., ADIBAH, F., NUR IHZA MARHAENDRA, Q., KHANSA FADHILAH, R., AKBAR ARCHA PRASETYO, K., BIN AHMAD HAIRUDDIN, Q., ANDRIMIDA, A., ZAIN HARDIYAN, F., ILMU KELAUTAN, DAN, BRAWIJAYA, U., STUDI ILMU KELAUTAN, P., PERIKANAN DAN ILMU KELAUTAN, F., BHAKTI ALAM SENDANG BIRU, Y., SENDANG BIRU KAMPUNGBARU, J., MALANG, K., TEKNIS, P. AND PPP PONDOKDADAP, U., 2022. *Prosiding Seminar Nasional Perikanan dan Kelautan dalam Rangka Memperingati Hari Ikan Nasional (HARKANNAS) Tahun 2021 ISBN*.
- JUBAEDAH, I. AND ANAS, P., 2019. Dampak Pariwisata Bahari Terhadap Ekosistem Terumbu Karang di Perairan Nusa Penida, Bali. *Jurnal Penyuluhan Perikanan dan Kelautan*, 13(1), pp.59–75. <https://doi.org/10.33378/jppik.v13i1.124>.
- KHATIB SULAIMAN, J., PERMANA SANUSI, A., FARIZA, A. AND ELEKTRONIKA NEGERI SURABAYA, P., 2023. Klasifikasi Tinggi Badan Menggunakan Metode Mask R-CNN. *Indonesian Journal of Computer Science Attribution*, 12(4), p.2050.
- LEI, X., PAN, H. AND HUANG, X., 2019. A Dilated CNN Model for Image Classification. *IEEE Access*, 7, pp.124087–124095. <https://doi.org/10.1109/ACCESS.2019.2927169>.
- MA, X., LI, Z. AND ZHANG, L., 2022. An Improved ResNet-50 for Garbage Image Classification. *Tehnicki Vjesnik*, 29(5), pp.1552–1559. <https://doi.org/10.17559/TV-20220420124810>.
- MAHATIR, M., ADIBRATA, S., UTAMI, E., MANAJEMEN, J. AND PERAIRANFPPB-UBB, S., 2022. Inventarisasi Gangguan Kesehatan Terumbu Karang di Perairan Perlang Bangka Belitung. [online] Available at: <[https://journal.ipb.ac.id/index.php/coj](https://journal.ipb.ac.id/index.php/coj;)>.
- RADITYA KESATRIA SINAGA, R., MAULID AL-WIRA, R., KURNIAWAN, F., RONI, S. AND RAHMA HIDAYATI, J., 2023. Kondisi Kesehatan Terumbu Karang di Taman Wisata Perairan Kepulauan Anambas. *Akuatiklestari*, [online] 6. <https://doi.org/10.31629/akuatiklestari.v6i.5010>.
- RUMKOREM, O.L.Y., KURNIA, R. AND YULIANDA, F., 2019. Asosiasi Antara Tutupan Komunitas Karang Dengan Komunitas Ikan Terumbu Karang Di Pesisir Timur Pulau Biak, Kabupaten Biak Numfor. *Jurnal Ilmu dan Teknologi Kelautan Tropis*, 11(3), pp.615–625. <https://doi.org/10.29244/jitkt.v11i3.23375>.
- SOUTER, D., PLANES, S., WICQUART, J., LOGAN, M., OBURA, D. AND STAUB, F., 2020. Status of Coral Reefs of the World: 2020 Executive Summary.
- TATIPATA, K.P.B., MASHORENG, S., 2019. The Impact of Coral Conditions on the Structure of the Megabenthos Community Associated with the Coral Reefs of the Spermonde Islands. *Torani: JFMarSci*, 3(1), pp.37–50.
- YUNILDA, I.I., GAZALI, M., SURIANI, M. AND NUFUS, H., 2022. The Coral Reef Cover Percentage at of Pulo Kayee Waters East Bakongan, South Aceh. *Jurnal a'ot Ilmu Kelautan*, [online] IV. Available at: <<http://jurnal.utu.ac.id/JLIK>>.
- ZHOU, T., ZHAO, Y. AND WU, J., 2021. ResNeXt and Res2Net Structures for Speaker Verification. In: *2021 IEEE Spoken Language Technology Workshop, SLT 2021 - Proceedings*. Institute of Electrical and Electronics Engineers Inc. pp.301–307. <https://doi.org/10.1109/SLT48900.2021.9383531>.

