

Otomatisasi Penghitungan Jenis Sel Darah Putih pada Pasien *Acute Lymphoblastic Leukemia* Tipe L1 dengan Deteksi Objek dan Segmentasi *Instance* Berbasis *Mask R-CNN*

Mahatma Ageng Wisesa¹, Nopriadi², Nazrul Effendi³

^{1,2,3} Departemen Teknik Nukir dan Teknik Fisika FT UGM
Jln. Grafika 2 Yogyakarta 55281 INDONESIA

¹mahatma.ageng.wisesa@gmail.com ²nopriadi@ugm.ac.id ³nazrul@ugm.ac.id

Intisari— *Acute Lymphoblastic Leukemia* adalah kanker paling umum untuk anak-anak dan penyebab utama kematian pada anak-anak. Sebanyak 70% pasien penderita ALL termasuk dalam kategori L1. Penyakit tersebut ditandai dengan produksi sel darah putih muda (sel limfoblas) lebih dari 20% dari total sel darah putih di sumsum tulang belakang. Respon cepat dan diagnosis dini sangat dibutuhkan untuk rehabilitasi pasien, terutama pada anak-anak. Metode tradisional untuk penghitungan sel darah bergantung pada bantuan manusia, di mana penghitungan dan klasifikasi sel dilakukan secara manual melalui mikroskop. Oleh karena itu, diperlukan alternatif untuk menggantikan metode tradisional dengan sistem otomatis berbasis *deep learning* yang telah berkembang pesat beberapa tahun terakhir agar menggantikan penghitungan dan klasifikasi sel secara manual. Pada penelitian ini digunakan model *deep learning* bernama *Mask R-CNN* yang melakukan klasifikasi, lokalisasi, dan menghasilkan *segmentation mask* untuk mempermudah visualisasi sel terdeteksi. *Mask R-CNN* dilatih pada sel limfoblas dan non-limfoblas. Teknik pengolahan citra seperti filter bilateral, *thresholding*, dan pemerataan histogram diterapkan pada penelitian ini untuk meningkatkan performa model dan mengurangi waktu pelatihan. Penelitian ini telah berhasil mengimplementasikan *Mask R-CNN* dengan akurasi, *F1-Score*, dan mAP sebesar 91,59%, 95,61%, dan 92,75% yang memiliki performa lebih baik dibandingkan penelitian-penelitian sebelumnya. Hasil implementasi pada penelitian ini memiliki potensi untuk mengurangi kesalahan pemeriksaan manual dan menghemat banyak sumber daya manusia, waktu, dan biaya.

Kata kunci— *computer vision*, pendeteksian objek, *deep learning*, *Mask R-CNN*, pengolahan citra, *acute lymphoblastic leukemia*.

Abstract— *Acute Lymphoblastic Leukemia* is the most common cancer in children and the leading cause of death among children. As much as 70% of patients with ALL are included in the L1 category. The disease is characterized by the production of young white blood cells (lymphoblast cells) more than 20% of the total white blood cells in the bone marrow. Rapid response and early diagnosis are highly needed for the rehabilitation of patients, especially children. The traditional method of human-assisted blood cell counting, in which counting and classification is carried out manually through a microscope. Therefore, an alternative is needed to replace traditional method with deep learning-based automated systems that have developed rapidly in recent years to replace manual cell counting and classification. In this research, a deep learning model called *Mask R-CNN* is used which performs classification, localization, and generates mask segmentation to facilitate easy visualization of the detected cells. The *Mask R-CNN* were trained on lymphoblast and non-lymphoblast cells. Image processing techniques such as bilateral filters, thresholding, and histogram equalization were applied in this work to improve model performance and reduce training time. This research has successfully implemented *Mask R-CNN* with accuracy, *F1-Score*, and mAP of 91.59%, 95.61%, and 92.75% which have better performance than previous research. Results in this research have the potential to reduce manual microscopic examination errors and save a lot of human resources, time, and costs.

Keywords— *computer vision*, object detection, deep learning, *Mask R-CNN*, image processing, *acute lymphoblastic leukemia*.

I. PENDAHULUAN

Sel darah putih berfungsi sebagai sistem kekebalan tubuh yang merupakan pertahanan utama tubuh terhadap infeksi dan penyakit [1]. Leukemia adalah jenis kanker darah atau sumsum tulang yang ditandai dengan peningkatan dramatis yang tidak teratur dalam jumlah sel darah putih yang belum matang yang disebut "sel limfoblas" [2]. Leukemia diklasifikasikan menjadi leukemia akut dengan kemampuan progresif cepat dan leukemia kronis yang berlangsung lambat [3]. Leukemia tipe akut diklasifikasikan menjadi dua kelas berdasarkan model *French-American-British* (FAB) yang

merupakan model klasifikasi leukemia yang paling umum, yaitu *Acute Myeloid leukemia* (AML) dan *Acute Lymphoblastic Leukemia* (ALL) [2,4,5]. *Acute Lymphoblastic Leukemia* adalah kanker paling umum untuk anak-anak [2]. Penyakit kanker adalah penyakit kedua yang paling mematikan pada anak-anak, dan *Acute Lymphoblastic Leukemia* adalah penyebab utama kematian pada anak-anak [6].

French-American-British (FAB) mengklasifikas ALL berdasarkan ukuran sel, sitoplasma, nukleus, vakuola, dan basofilia menjadi tiga sub tipe, yaitu L1 (anak-anak), L2 (anak yang lebih tua dan orang dewasa), dan L3 (pasien dengan

leukemia sekunder akibat limfoma burkitt) [7]. Sebanyak 70% pasien penderita ALL termasuk dalam kategori L1 [8]. Penyakit tersebut ditandai dengan produksi jumlah sel darah putih muda (bernama sel limfoblas) yang berlebihan [2]. *Threshold* jumlah sel limfoblas adalah tidak lebih 20% dari total sel darah putih di sumsum tulang belakang untuk batas normal populasi sel limfoblas [8]. Oleh karena itu, respon cepat dan diagnosis dini sangat dibutuhkan untuk rehabilitasi pasien, terutama pada anak-anak [9].

Teknik penghitungan sel darah tradisional yang digunakan di laboratorium cukup menghabiskan waktu dan tenaga [11]. Metode tradisional untuk penghitungan sel darah bergantung pada bantuan manusia, di mana penghitungan dan klasifikasi sel dilakukan secara manual melalui mikroskop [11]. Pemeriksaan mikroskopis manual terhadap apusan darah menghabiskan banyak waktu dan mengalami bias, seperti faktor pengalaman hematologis, kelelahan, dan faktor manusia umum lainnya yang menyebabkan hasil laporan tidak konsisten dan subjektif [10,11]. Oleh karena itu, muncul kebutuhan untuk sistem deteksi dan penghitungan sel darah otomatis secara hemat biaya, cepat, dan efisien tanpa dipengaruhi oleh kelelahan manusia.

Artificial Intelligence (AI) telah sangat maju dalam beberapa tahun terakhir yang mendapatkan manfaat dari kemajuan teknologi *Graphics Processing Unit* (GPU) [13]. Banyak model *Convolutional Neural Network* (CNN) dengan arsitektur berbeda yang dikembangkan dan diterapkan dalam berbagai bidang [13]. *Deep Learning* (pembelajaran mendalam) dianggap sebagai *state-of-the-art* untuk klasifikasi citra [14,15]. Arsitektur CNN yang merupakan bagian dari *deep learning* dapat menghasilkan penurunan tingkat kesalahan klasifikasi sekitar 25% pada tahun 2011 menjadi 3,6% pada tahun 2015 [14,15].

Pada penelitian ini akan digunakan metode *Mask Region-based Convolutional Neural Network* (R-CNN) untuk deteksi dan segmentasi *instance* dari sel limfoblas dan non limfoblas pada apusan darah pasien ALL tipe L1. *Mask R-CNN* merupakan pengembangan dari *Faster R-CNN* dengan adanya penambahan cabang untuk memprediksi *object mask* secara paralel dari *bounding box* yang telah ada [16]. *Faster R-CNN* tidak dirancang untuk penyalarsan piksel-ke-piksel antara *input* dan *output* jaringan [16]. Oleh karena itu, dibutuhkan penelitian yang dapat mendeteksi dan juga secara tepat mengelompokkan setiap piksel-ke-piksel *instance* dari sel limfoblas dan non-limfoblas pada pasien ALL tipe L1 untuk melakukan otomatisasi perhitungan populasi sel darah putih. Kelebihan dari model ini adalah kemampuannya untuk menghasilkan segmentasi *mask* untuk mempermudah visualisasi sel darah putih yang terdeteksi.

Penelitian-penelitian mengenai klasifikasi, segmentasi, pendeteksian, dan penghitungan otomatis dari jenis sel darah manusia dengan berbagai metode *image processing* dan *deep learning* telah banyak dilakukan. Penelitian-penelitian tersebut bertujuan sebagai alternatif menganalisis kesehatan pasien yang memerlukan pengklasifikasian dan perhitungan jumlah dari jenis sel darah secara manual ataupun menggunakan *analyzer*. Kesulitan dalam mendeteksi sel leukemia dari citra

terdapat pada kesamaan morfologi dan variabilitas subjek, membuat pendefinisian fitur citra yang sesuai menjadi tugas yang sangat menantang [17]. Selain itu, semua citra juga mengandung jenis sel darah lain yang dapat mengganggu klasifikasi sel darah putih [17].

De Rong Loh dkk. [18] pada tahun 2021 melakukan penelitian mengenai penghitungan sel darah otomatis untuk skrining infeksi malaria dengan menggunakan *Mask R-CNN*. *Dataset* penelitian berupa sel darah merah yang tidak terinfeksi dan yang terinfeksi. *Dataset* tersebut terdiri dari 297 citra dalam rasio perkiraan 3:1:1. Kumpulan citra tersebut dianotasi secara manual. Teknik *image processing* dari cv2 dan *scikit-image* digunakan dalam algoritma penghitungan sel tersebut. Kontras citra ditingkatkan dengan menerapkan pemerataan histogram. Operasi morfologi, seperti *opening* dan *erosion* dilakukan untuk menghilangkan *noise* citra. *Hyperparameter* seperti *threshold non-maximum suppression* (NMS) dan *min confidence levels* digunakan untuk lebih selektif dalam mengidentifikasi *regions* dan *bounding boxes* untuk sel darah. Nilai akurasi dan mAP yang diperoleh adalah 82% dan 73,1%.

Najmeddine dkk. [11] pada tahun 2019 melakukan penelitian serupa, yakni penghitungan sel darah otomatis menggunakan *Mask R-CNN*. Sel darah merah dan sel darah putih diidentifikasi, diklasifikasikan, dan dihitung dari gambar apusan darah mikroskopis. *Dataset* citra apusan darah mikroskopis yang mencakup 150 citra dengan lebih dari 24.000 sampel sel. Penelitian tersebut menerapkan teknik augmentasi pada data latih untuk meningkatkan efisiensi model dan mengurangi *overfitting*. Rotasi acak, *zooming*, pergeseran, dan transformasi membalik diterapkan untuk memperluas dan memvariasikan data. Penelitian tersebut digunakan *threshold IoU* bernilai 0,65 untuk memaksimalkan deteksi sel. Hasil eksperimen menunjukkan bahwa model yang diusulkan secara efisien mendeteksi 92% sel darah merah dan 96% sel darah putih.

Yang-Hsien dkk. [19] pada tahun 2020 melakukan penelitian menggunakan *Mask R-CNN* untuk membedakan sel darah merah *thalassemic* (tRBCs) dan sel darah merah sehat (hRBCs). Penelitian tersebut menggunakan 2.001 citra hRBCs dan 4.268 citra tRBCs, oleh karena itu penelitian tersebut mengadopsi augmentasi data dengan rotasi acak, untuk menyeimbangkan data pelatihan. Operasi rotasi acak juga meningkatkan variasi dalam *dataset* yang dapat mencegah *overfitting* selama pelatihan model *Mask R-CNN*. *Dataset* tersebut dibagi dengan rasio 3:1:1. Hasil performa model yang diperoleh adalah 97,5% untuk *F1-Score* dan 97,8% untuk akurasi.

Fajri [20] pada tahun 2020 melakukan penelitian menggunakan *Faster R-CNN* untuk mendeteksi sel limfoblas dan sel non-limfoblas. *Dataset* yang diperoleh dari pasien pengidap ALL tipe L1. *Dataset* dibagi menjadi 70% pelatihan, 15% validasi, dan 15% pengujian. Pelatihan digunakan arsitektur ResNet50 dan ResNet101 dengan *hyperparameter* berupa: *batch size*, *learning rate*, dan *step*. Akurasi dimaksimalkan dengan melakukan penyesuaian *threshold non-max suppression* (NMS). Proses penghitungan otomatis

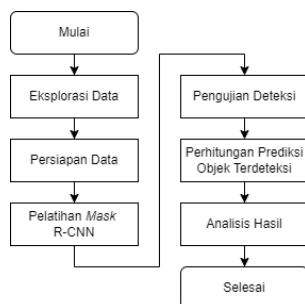
dibandingkan dengan penghitungan secara manual dan dilakukan perhitungan *Mean Absolute Error* (MAE) sebagai pengujian kinerja model. Performa terbaik diperoleh Akurasi, *F1-Score*, dan mAP sebesar 86,2%, 87,9%, dan 90,9%.

Rohan dkk. [21] melakukan penelitian pada tahun 2021 mengenai deteksi sel limfoblas otomatis untuk diagnosis ALL dengan implementasi *You Only Look Once* versi 4 (YOLOv4). Algoritma deteksi objek dilatih dan diuji dengan citra dari *dataset* ALL-IDB1 dan *dataset* C-NMC 2019. *Mean Average Precision* (mAP) yang diperoleh adalah 96,06% untuk *dataset* ALL-IDB1 dan 98,7% untuk *dataset* C-NMC 2019. Augmentasi citra tersebut digunakan untuk meningkatkan properti *dataset* dan membuat algoritma lebih *robust*. Augmentasi citra dilakukan seperti pembalikan horizontal dan vertikal, rotasi 90°, rentang kecerahan $\pm 25\%$, dan *noise* hingga 5% piksel dilakukan secara acak pada citra *dataset* ALL-IDB1. Augmentasi C-NMC dilakukan dengan empat langkah: 1) Pembalikan horizontal dan vertikal; 2) Penyesuaian kecerahan acak $\pm 40\%$; 3) Rotasi acak searah jarum jam dan berlawanan arah jarum jam; dan 4) *Gaussian blur* acak antara 0–3% piksel dan *noise salt* dan *pepper* untuk 5% piksel. *Dataset* dipartisi menjadi pelatihan, validasi, dan pengujian dengan pembagian 70:20:10.

II. METODOLOGI PENELITIAN

Pada penelitian ini, digunakan beberapa perangkat lunak untuk menunjang pelaksanaan penelitian. *Google Colaboratory Pro+* yang dilengkapi Python 3.7.12 dan TensorFlow 2.6.0. *Google Colab Pro+* digunakan sebagai media pelatihan model *Mask R-CNN* dengan GPU NVIDIA Tesla P100. OpenCV 4.5.3 digunakan sebagai pengolahan citra. VGG Image Annotator (VIA 2.0.1.1) digunakan sebagai anotasi citra. *Visual Studio Code* digunakan sebagai *code editor* untuk augmentasi citra, segmentasi citra, penghitungan jumlah sel berdasarkan label, pemisahan data, dan konfigurasi model *Mask R-CNN*. *Microsoft Excel* 2019 digunakan sebagai pemisahan data dan perhitungan MAE. *Google Drive* sebesar 2 TB digunakan sebagai penyimpanan model *Mask R-CNN*, data latih, data validasi, dan *logs* hasil latih model.

Perangkat keras yang digunakan pada penelitian ini berupa komputer personal portable dengan prosesor AMD Ryzen 7 4800HS (2.9 GHz), RAM 16 GB, dan GPU NVIDIA GeForce GTX1650Ti GDDR6 4GB. Perangkat keras tersebut digunakan untuk melakukan *inference* dari model yang telah dilatih pada *Google Colab Pro+*.



Gambar 1. Diagram alir tata laksana penelitian.

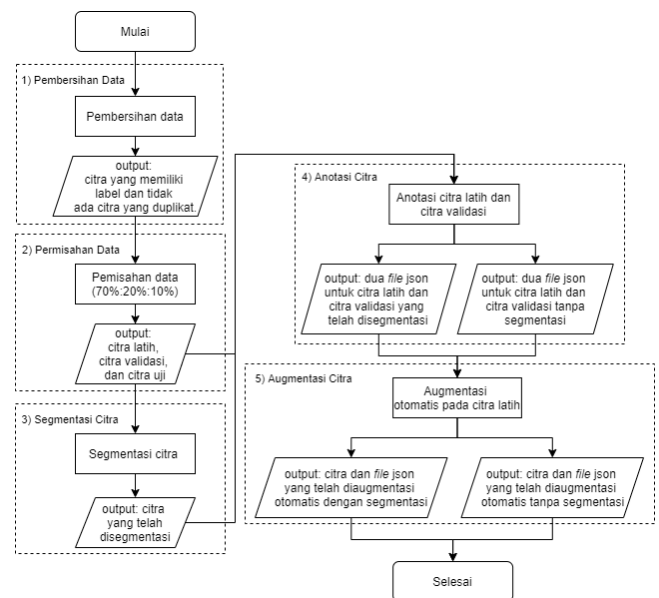
Dataset yang digunakan pada penelitian ini berupa citra digital sel darah dari pasien pengidap *Acute Lymphoblastic Leukemia* tipe L1. Citra sel darah telah diberi label oleh hematologis dari Rumah Sakit Umum Pusat (RSUP) Dr. Sardjito Yogyakarta. Sampel citra sel darah yang digunakan pada penelitian berjumlah 1.461 citra dengan resolusi 4100 × 3075. Proses penelitian dilakukan dengan tahapan sesuai dengan diagram alir pada Gambar 1.

A. Eksplorasi Data

Pada tahapan eksplorasi data merupakan tahap untuk mengetahui jenis sel darah putih apa saja yang terdapat dalam *dataset*. Pada *dataset* penelitian terdapat 1.461 citra yang telah diberi label oleh hematologis RSUP Dr. Sardjito dan terdapat 15 jenis sel darah putih.

B. Persiapan Data

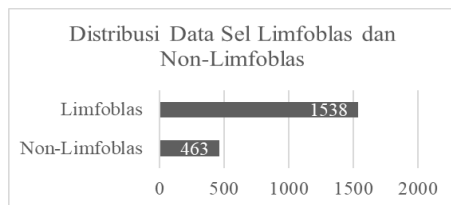
Persiapan data dilakukan dengan lima tahap yang ditampilkan pada diagram alir Gambar 2. Pada tahap pertama, *dataset* perlu dilakukan pembersihan karena tidak semua citra sepenuhnya dapat digunakan untuk penelitian. Sampel citra sel darah awalnya berjumlah jumlah 1.725 citra, setelah dilakukan pembersihan data diperoleh 1.461 citra yang dapat digunakan dalam penelitian ini. Hal yang menentukan apakah citra dapat digunakan pada penelitian adalah citra yang memiliki label untuk setiap sel darah putih dan tidak terjadinya duplikat untuk setiap citra. Setelah proses pembersihan data, maka dapat dilakukan perhitungan otomatis berdasarkan nama *file* citra tersebut menggunakan program Python untuk dilakukan proses pemisahan data.



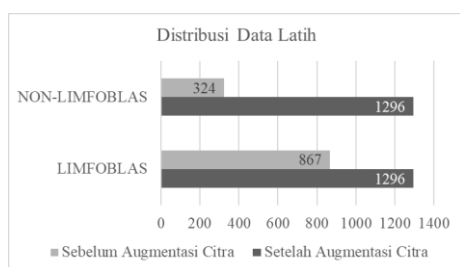
Gambar 2. Diagram alir persiapan data.

Pada tahap kedua, *dataset* dipartisi menjadi 70% data latih, 20% data validasi, dan 10% data uji. Fokus penelitian dipusatkan pada pendeteksian limfoblas dan non-limfoblas dikarenakan untuk menghitung jumlah sel kanker (limfoblas) dari total semua jenis sel darah putih. Jenis sel darah putih

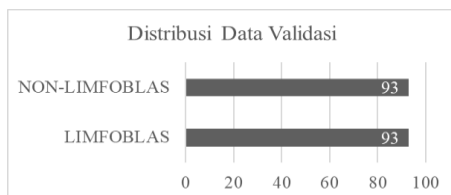
selain sel limfoblas dijadikan satu label, yaitu non-limfoblas. Distribusi awal *dataset* ditampilkan pada Gambar 3. Namun, distribusi setelah partisi ditampilkan pada Gambar 4. (sebelum augmentasi citra), Gambar 5., dan Gambar 6. Pada penelitian ini hanya data latih yang dilakukan augmentasi atau *oversampling* pada kelas minoritas agar tidak terjadi *class imbalance* ataupun *overfitting*. Untuk menghindari *class imbalance* kelas mayoritas dilakukan *undersampling*, sehingga total citra yang digunakan dalam penelitian ini berjumlah 1.056 citra.



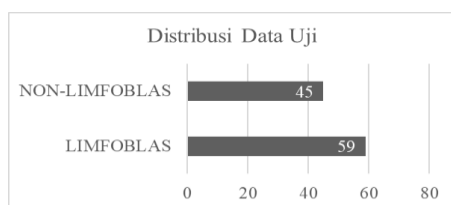
Gambar 3. Distribusi data sel limfoblas dan non-limfoblas.



Gambar 4. Distribusi data latih biner.



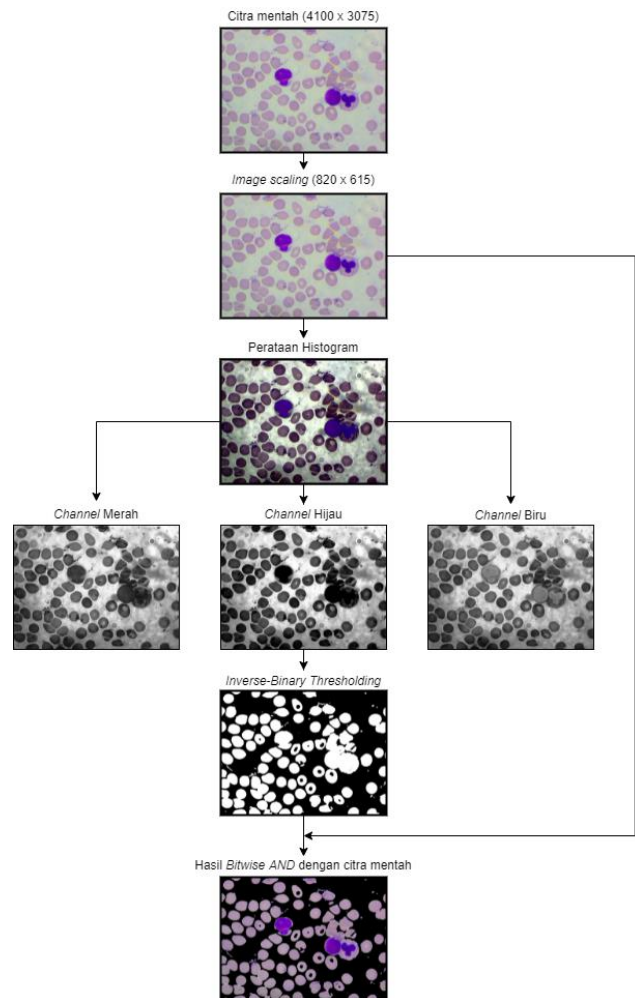
Gambar 5. Distribusi data validasi biner.



Gambar 6. Distribusi data uji biner.

Pada tahap ketiga, digunakan beberapa teknik pengolahan citra untuk melakukan segmentasi citra sel darah dari latar belakangnya. Tujuan dilakukan segmentasi ini adalah untuk membandingkan performa model *Mask R-CNN* dengan adanya segmentasi dan tanpa adanya segmentasi (citra mentah). Ilustrasi untuk proses segmentasi sel ditampilkan pada Gambar 7. Resolusi awal citra adalah 4100×3075, kemudian dilakukan *image scaling* sebesar 0,2 kalinya dari

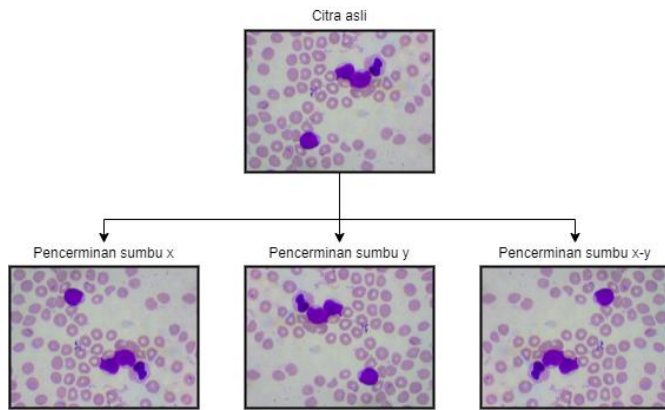
citra asli untuk mengurangi komputasi dalam tahap pelatihan dan deteksi tanpa adanya mengurangi fitur sel darah putih secara signifikan. Pada tahapan pengolahan citra dilakukan segmentasi sel terhadap latar belakangnya dengan melakukan *thresholding* pada *channel* hijau. Sebelum dilakukan *thresholding* perlu adanya pemerataan histogram untuk mendistribusikan intensitas pada citra, sehingga kontras antara sitoplasma dan inti sel terhadap latar belakangnya lebih tinggi. Setelah dilakukan pemerataan histogram, *channel* hijau dari pemerataan tersebut dilakukan *inverse-binary thresholding*. Hasil dari *thresholding* tersebut dapat digunakan sebagai *mask* pada operasi *Bitwise AND* untuk mengekstrak bagian sel darah putih dan sel darah merah dari citra mentah.



Gambar 7. Ilustrasi proses segmentasi sel darah dari citra mentah.

Pada tahap keempat, proses anotasi dilakukan pada citra mentah dan citra disegmentasi. Anotasi dilakukan dengan pembentukan poligon sesuai dengan bentuk sel darah putih yang kemudian diikuti pemberian nama label (limfoblas atau non-limfoblas). Proses anotasi dilakukan dengan perangkat lunak *VGG Image Annotator* (VIA 2.0.1.1) yang dilakukan secara manual untuk setiap sel darah putih.

Pada tahap kelima, data latih dilakukan augmentasi bertujuan untuk menyeimbangkan kelas dan menghindari *overfitting* pada data latih. Augmentasi citra dilakukan dengan pencerminan pada sumbu x, sumbu y, dan kedua sumbu x-y, seperti ditampilkan pada Gambar 8. Penambahan koordinat poligon dan label kelas baru dilakukan secara otomatis (*auto-oriented*) dengan menggunakan OpenCV dan program Python untuk melakukan pencerminan pada koordinat-koordinat poligon dan memberi label kelasnya pada citra hasil augmentasi. Hasil augmentasi memberikan kelas yang seimbang seperti ditunjukkan pada Gambar 4 (setelah augmentasi citra).



Gambar 8. Augmentasi citra.

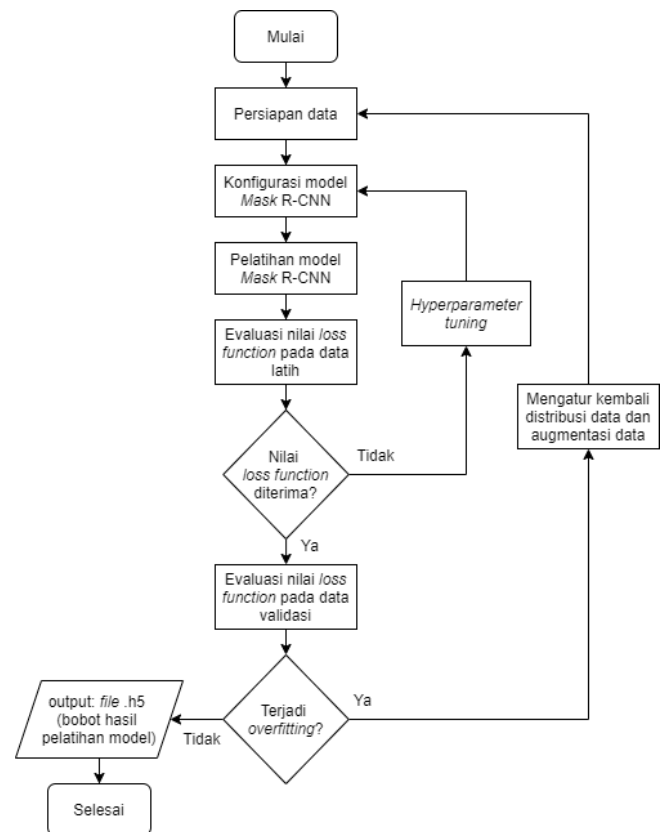
C. Pelatihan Mask R-CNN

Pelatihan model *Mask R-CNN* dilakukan pada *Google Colaboratory Pro+* menggunakan GPU NVIDIA Tesla P100 dan *Google Drive* 2TB sebagai penyimpanan model *Mask R-CNN*, data latih, data validasi, dan *logs* hasil latih model. Pelatihan model digunakan arsitektur *backbone* ResNet101 dikarenakan jumlah lapisan jaringan lebih dalam dibandingkan ResNet50 dan juga ketersediaan perangkat keras untuk melakukan pelatihan dengan arsitektur *backbone* ResNet101. *Pre-train weight* yang digunakan pada pelatihan ini berasal dari bobot COCO (*Common Object in Context*). Diagram alir proses pelatihan model *Mask R-CNN* pada penelitian ditampilkan pada Gambar 9.

Pelatihan model *Mask R-CNN* perlu dilakukan konfigurasi untuk menyesuaikan *dataset* yang digunakan pada penelitian ini. Konfigurasi tersebut berupa: arsitektur *backbone* yang digunakan, jumlah kelas, dan *hyperparameter* untuk pelatihan pada *dataset*. *Hyperparameter* yang diatur pada penelitian ini meliputi: *epochs*, *steps per epoch*, *validation steps*, *learning rate*, *detection minimum confidence*, *detection NMS threshold*, dan *RPN NMS threshold*.

Hasil *loss function* pada data latih dan data validasi dapat dievaluasi menggunakan TensorBoard. Grafik pada TensorBoard dapat dilakukan pengamatan untuk *loss function* data latih maupun data validasi dan mengetahui apakah terjadi *underfitting* atau *overfitting*. Jika model hasil pelatihan *underfitting* maka dapat dilakukan *hyperparameter tuning*. Namun, jika model mengalami *overfitting* dapat melakukan

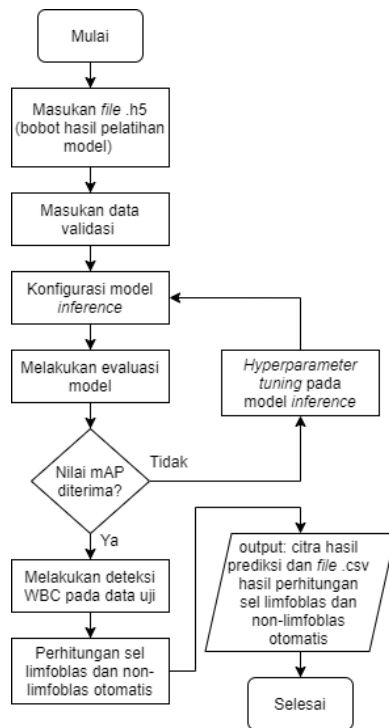
evaluasi ulang pada distribusi data dan augmentasi data untuk memperoleh model yang lebih *generalize* dan *robust*.



Gambar 9. Diagram alir proses pelatihan model *Mask R-CNN*.

D. Pengujian Deteksi

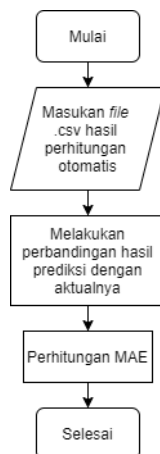
Pengujian algoritma dilakukan berdasarkan hasil pelatihan algoritma sebelumnya. Pada Gambar 10, merupakan diagram alir pengujian algoritma yang digunakan penelitian ini. Keluaran dari hasil pelatihan algoritma sebelumnya berupa *file .h5* yang merupakan bobot hasil pelatihan. *File .h5* tersebut digunakan sebagai proses *inference* pada data validasi dan data uji. Berdasarkan hasil pembobotan tersebut dapat dilakukan perhitungan mAP pada data validasi untuk melakukan evaluasi hasil pelatihan. Jika nilai mAP yang diperoleh tidak maksimal maka dapat dilakukan *hyperparameter tuning* pada konfigurasi *inference* dengan *hyperparameter* sebagai berikut: *detection minimum confidence*, *detection NMS threshold*, dan *RPN NMS threshold*. Hasil konfigurasi *inference* dengan mAP terbaik akan dilakukan deteksi dengan data uji dan melakukan perhitungan sel limfoblas dan non-limfoblas secara otomatis. Keluaran deteksi berupa citra hasil prediksi dan data dengan format *comma separated values* (.csv) yang berisi jumlah jenis sel darah putih pada setiap citra.



Gambar 10. Diagram alir pengujian algoritma.

E. Perhitungan Prediksi Objek Terdeteksi

Hasil proses pada tahap pengujian algoritma adalah berupa citra hasil prediksi (terdapat *bounding box* dan *object mask* pada sel terdeteksi) dan file .csv yang merupakan hasil perhitungan otomatis pada data uji. Kemudian hasil perhitungan otomatis akan dilakukan perbandingan dengan perhitungan aktualnya (bedasarkan pelabelan dari hematologis). Pada Gambar 11. merupakan diagram alir perbandingan antara perhitungan manual dengan hasil prediksi model yang telah dilatih. *Mean absolute error* (MAE) digunakan sebagai metrik evaluasi performa model pada data uji. Proses MAE dilakukan pada *Microsoft Excel* 2019.



Gambar 11. Diagram alir perbandingan perhitungan manual dengan hasil prediksi model.

F. Analisis Hasil

Pada analisis hasil penelitian ini meliputi analisis performa model yang ditinjau dari *loss function* pada data latih dan data validasi berdasarkan *hyperparameter* saat proses pelatihan sebelumnya. Pengujian model dilakukan pada data uji dengan menggunakan *hyperparameter* yang menghasilkan nilai mAP tertinggi pada data validasi dan melakukan evaluasi performa. Kemudian dilakukan perhitungan otomatis pada data uji dan melakukan perbandingan perhitungan secara manual dengan metrik MAE

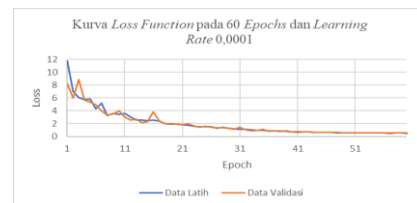
III. HASIL DAN PEMBAHASAN

Pelatihan algoritma pertama kali dilakukan dengan 60 *epochs* yang membutuhkan waktu maksimal 3 jam 23 menit. Pelatihan dengan 60 *epochs* tersebut bertujuan untuk menentukan nilai *learning rate*, *steps per epoch*, dan *validation steps* terbaik dengan memvariasikan nilai-nilainya. Pelatihan ini juga menentukan apakah distribusi data yang digunakan mengalami *overfitting* atau tidak. Nilai *hyperparameter* pada tahap pelatihan awal ini divariasikan seperti berikut:

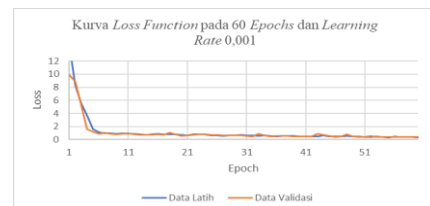
1. *Learning rate* bernilai 0,01; 0,001; dan 0,0001.
2. *Steps per epoch* bernilai 75, 100, 125, dan 150.
3. *Validation steps* bernilai 150, 200, dan 250.
4. *Detection minimum confidence* bernilai 0,70; 0,75; 0,80; 0,85; dan 0,9.

A. Penentuan Nilai Learning Rate

Nilai *learning rate* perlu divariasikan karena jika nilainya terlalu besar dapat terjadi ledakan pembobotan, sehingga kurva *loss function* tidak turun ke nilai minimum. Namun, jika nilai *learning rate* terlalu kecil akan membutuhkan waktu dan komputasi lebih banyak untuk mencapai nilai *loss* yang minimum. Pada Gambar 12. dan Gambar 13. merupakan hasil latih dengan *learning rate* 0,0001 dan 0,001. Dari kedua kurva *loss function* tersebut tidak terjadi *overfitting* yang mengartikan model cukup *generalize* dari distribusi data yang digunakan pada penelitian ini.



Gambar 12. Kurva *loss function* pada 60 *epochs* dengan *learning rate* sebesar 0,0001.



Gambar 13. Kurva *loss function* pada 60 *epochs* dengan *learning rate* sebesar 0,001.

Kurva *loss* pada *learning rate* sebesar 0,01 tidak dapat ditampilkan karena terjadi ledakan pembobotan dan nilai *loss* sebagian besar berupa NaN (*Not a Number*). Oleh karena itu, penelitian ini cukup membandingkan hasil *loss* dengan *learning rate* sebesar 0,001 dan 0,0001. Pada *learning rate* sebesar 0,001 memberikan nilai *loss* akhir 0,3358 untuk data latih dan 0,3749 untuk data validasi. Namun, pada *learning rate* sebesar 0,0001 memberikan nilai *loss* akhir 0,4795 untuk data latih dan 0,5315 untuk data validasi. Pada kedua nilai *learning rate* tersebut dapat disimpulkan bahwa nilai 0,001 merupakan nilai yang optimal dan dapat menurunkan nilai *loss* lebih cepat dibandingkan 0,0001.

B. Penentuan Nilai Steps per Epoch

Hyperparameter selanjutnya yang ditentukan adalah *steps per epoch*. Pelatihan dengan variasi nilai *steps per epoch* dilakukan pada nilai *learning rate* 0,001. Hal yang menjadi pertimbangan dalam menentukan *steps per epoch* adalah nilai *loss* dan waktu pelatihannya. Nilai *loss* akhir yang diperoleh dan waktu pelatihannya ditampilkan pada Tabel I. Dari hasil variasi nilai *steps per epoch* menunjukkan semakin tinggi nilainya semakin baik nilai *loss* yang diperoleh, tetapi membutuhkan waktu lebih lama untuk pelatihan model. Namun, nilai *steps per epoch* yang lebih tinggi tidak menurunkan nilai *loss* secara signifikan. Oleh karena itu, nilai *steps per epoch* yang dipilih adalah bernilai 100.

TABEL I.
HASIL PELATIHAN DENGAN VARIASI STEPS PER EPOCH

<i>Steps per epoch</i>	<i>Loss data latih akhir</i>	<i>Loss data validasi akhir</i>	Waktu pelatihan
75	0,6533	0,5946	2 jam 24 menit
100	0,3358	0,3749	2 jam 43 menit
125	0,2842	0,3110	2 jam 53 menit
150	0,3002	0,3387	3 jam 23 menit

C. Penentuan Nilai Validation Steps

Nilai *validation steps* divariasikan pada *learning rate* dan *steps per epoch* sebesar 0,001 dan 100. Penentuan nilai *validation steps* ditentukan berdasarkan nilai *loss* dan waktu pelatihannya, seperti pada penentuan nilai *steps per epoch* sebelumnya. Variasi nilai *validation steps* memberikan dampak yang sama dengan variasi *steps per epoch*. Nilai *validation steps* yang lebih besar dapat menurunkan nilai *loss*, tetapi jika semakin besar nilainya maka tidak ada perubahan secara signifikan dan membutuhkan waktu yang lebih lama. Nilai *loss* akhir dan waktu pelatihan yang diperoleh pada setiap variasinya ditunjukkan pada Tabel II. Oleh karena itu, nilai *validation steps* yang digunakan pada penelitian ini adalah 200.

TABEL II.
HASIL PELATIHAN DENGAN VARIASI VALIDATION STEPS

<i>Validation steps</i>	<i>Loss data latih akhir</i>	<i>Loss data validasi akhir</i>	Waktu pelatihan
150	0,5805	0,5737	2 jam 12 menit
200	0,3358	0,3749	2 jam 43 menit
250	0,3469	0,3539	3 jam 3 menit

Hasil pelatihan algoritma dengan jumlah *epoch* 60 dapat menentukan nilai *hyperparameter* yang optimal. *Hyperparameter* tersebut berupa *learning rate*, *steps per epoch*, dan *validation steps* yang masing-masing nilainya adalah 0,001, 100, dan 200. Nilai *hyperparameter* yang telah diperoleh tersebut akan digunakan pada pelatihan berikutnya dengan jumlah *epoch* yang lebih besar. Pelatihan algoritma tersebut juga membuktikan bahwa tidak terjadinya *overfitting* pada distribusi data yang digunakan, sehingga distribusi data tersebut dapat digunakan untuk tahap pelatihan selanjutnya.

D. Penentuan Detection Minimum Confidence

Penentuan nilai *detection minimum confidence* dilakukan dengan *hyperparameter* yang telah ditentukan pada A, B, dan C. Nilai *detection minimum confidence* divariasikan dengan nilai 0,7; 0,75; 0,80; 0,85; dan 0,90 untuk citra mentah dan citra disegmentasi. Seluruh pelatihan algoritma dilakukan dengan jumlah *epoch* 220. Hasil pelatihan dievaluasi berdasarkan mAP pada data validasi dengan nilai IoU sebesar 0,5 dan *detection minimum confidence* saat *inference* sebesar 0,4. *Detection minimum confidence* merupakan probabilitas minimum untuk menerima objek yang terdeteksi. Nilai tersebut berpengaruh saat melakukan pendeteksian objek karena menentukan apakah objek akan terdeteksi atau tidak. Oleh karena itu, *detection minimum confidence* saat pelatihan perlu ditentukan secara optimal. Nilai *detection minimum confidence* yang ditentukan akan disesuaikan dengan *dataset* digunakan, yaitu untuk citra mentah dan citra disegmentasi.

TABEL III.
PERFORMA HASIL LATIH PADA CITRA MENTAH

<i>Detection minimum confidence</i>	mAP
0,70	0,9188
0,75	0,8060
0,80	0,8982
0,85	0,8974
0,90	0,8399

TABEL IV.
PERFORMA HASIL LATIH PADA CITRA DISEGMENTASI

<i>Detection minimum confidence</i>	mAP
0,70	0,8818
0,75	0,8171
0,80	0,8070
0,85	0,9225
0,90	0,8512

Berdasarkan hasil pelatihan pada Tabel III. dan Tabel IV. untuk citra mentah dan citra disegmentasi dapat ditentukan nilai *detection minimum confidence* terbaik untuk masing-masing *dataset*. Hal tersebut dilakukan untuk meningkatkan peluang model terbaik saat melakukan pendeteksian pada data uji. Oleh karena itu, nilai *detection minimum confidence* terbaik untuk citra mentah adalah 0,70 dan untuk citra disegmentasi adalah 0,85. Dari hasil pelatihan ini

menunjukkan bahwa nilai *detection minimum confidence* pada citra disegmentasi lebih unggul dibandingkan dengan citra mentah. Nilai *loss* akhir ditampilkan pada Tabel V., *dataset* dengan citra disegmentasi memberikan nilai *loss* sedikit lebih baik dibandingkan citra mentah. Untuk waktu pelatihan, citra disegmentasi membutuhkan waktu lebih sedikit dibandingkan citra mentah, hal ini membuktikan metode-metode pengolahan citra yang digunakan pada penelitian ini dapat meningkatkan performa model lebih baik.

TABEL V.
NILAI *LOSS* AKHIR PADA PELATIHAN

<i>Dataset</i>	<i>Detection minimum confidence</i>	<i>Loss data latih akhir</i>	<i>Loss data validasi akhir</i>	<i>Waktu pelatihan</i>
Citra Mentah	0,70	0,1889	0,2286	11 jam 30 menit
Citra Disegmentasi	0,85	0,1510	0,1965	10 jam 7 menit

E. Evaluasi Performa Hyperparameter Terpilih

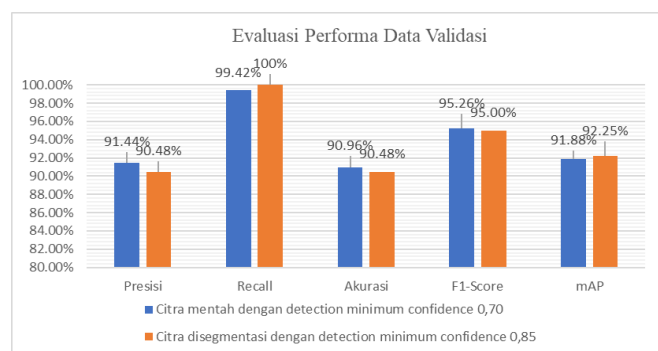
Hyperparameter terbaik untuk kedua *dataset* diukur performanya pada data validasi. Rangkuman *hyperparameter* terbaik yang digunakan dalam pelatihan ditampilkan pada Tabel VI. Metrik evaluasi yang digunakan berupa presisi, *recall*, akurasi, *F1-Score*, dan mAP. Hasil perhitungan metrik evaluasi pada data validasi secara keseluruhan ditampilkan pada Tabel VII. dan Gambar 14.

TABEL VI.
RANGKUMAN *HYPERPARAMETER* TERPILIH

<i>Dataset</i>	<i>Learning Rate</i>	<i>Steps per Epochs</i>	<i>Validation Steps</i>	<i>Detection Minimum Confidence</i>
Citra Mentah	0,001	100	200	0,70
Citra Disegmentasi	0,001	100	200	0,85

TABEL VII.
HASIL PERFORMA PADA DATA VALIDASI

<i>Dataset</i>	<i>Presisi</i>	<i>Recall</i>	<i>Akurasi</i>	<i>F1-Score</i>	<i>mAP</i>
Citra Mentah	0,9144	0,9942	0,9096	0,9526	0,9188
Citra Disegmentasi	0,9048	1,0000	0,9048	0,9500	0,9225



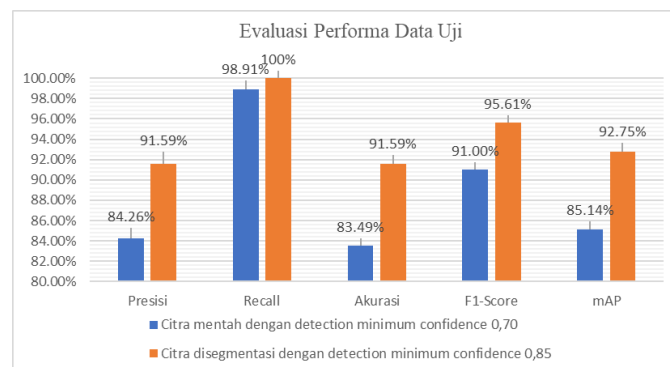
Gambar 14. Evaluasi performa data validasi dengan *hyperparameter* terbaik.

Performa *hyperparameter* yang telah dipilih memberikan nilai *recall* yang hampir 100% untuk citra mentah dan 100% untuk citra disegmentasi. Hal tersebut menunjukkan hampir semua sel darah putih dapat terdeteksi, tetapi dalam klasifikasi belum tentu menunjukkan kelas yang benar. Nilai presisi untuk citra mentah dan citra disegmentasi menunjukkan nilai yang hampir sama. Nilai presisi tersebut menunjukkan model dapat mendeteksi sel darah putih dengan kelas yang sama dengan *ground truth* sebesar 91,44% untuk citra mentah dan 90,48% untuk citra disegmentasi. Untuk nilai akurasi, *F1-Score*, dan mAP masing-masing hampir menunjukkan nilai yang sama, walaupun dilatih pada nilai *detection minimum confidence* yang berbeda.

Evaluasi performa pada data validasi memberikan nilai-nilai metrik yang cukup baik. Hal tersebut dapat dicapai dengan baik karena penentuan *hyperparameter* dilakukan pada data latih dan data validasi, sehingga akan memberikan performa yang baik pula pada data validasi. Oleh karena itu, evaluasi performa selanjutnya digunakan pada data uji yang hasil evaluasi secara keseluruhannya ditampilkan pada Tabel VIII. dan Gambar 15. Data uji merupakan data yang tidak pernah dilihat oleh model sebelumnya, sehingga evaluasi performa pada data uji merupakan performa yang paling dekat pada keadaan sesungguhnya.

TABEL VIII.
HASIL PERFORMA PADA DATA UJI

<i>Dataset</i>	<i>Presisi</i>	<i>Recall</i>	<i>Akurasi</i>	<i>F1-Score</i>	<i>mAP</i>
Citra Mentah	0,8426	0,9891	0,8349	0,9100	0,8514
Citra Disegmentasi	0,9159	1,0000	0,9159	0,9561	0,9275



Gambar 15. Evaluasi performa data uji dengan *hyperparameter* terbaik.

Hasil metrik evaluasi pada data uji memberikan performa yang cukup berbeda untuk kedua *dataset* tidak seperti pada data validasi sebelumnya. Presisi pada citra mentah bernilai 84,26%, sedangkan pada citra disegmentasi bernilai 91,59%. Hal tersebut menunjukkan hanya 84,26% dari sel darah putih yang terdeteksi untuk citra mentah merupakan kelas yang benar atau sesuai dengan *ground truth*. Namun, pada citra disegmentasi memberikan nilai 91,59% yang hampir sama performanya pada data validasi. Nilai *recall* untuk kedua *dataset* memberikan nilai yang hampir sama dengan data

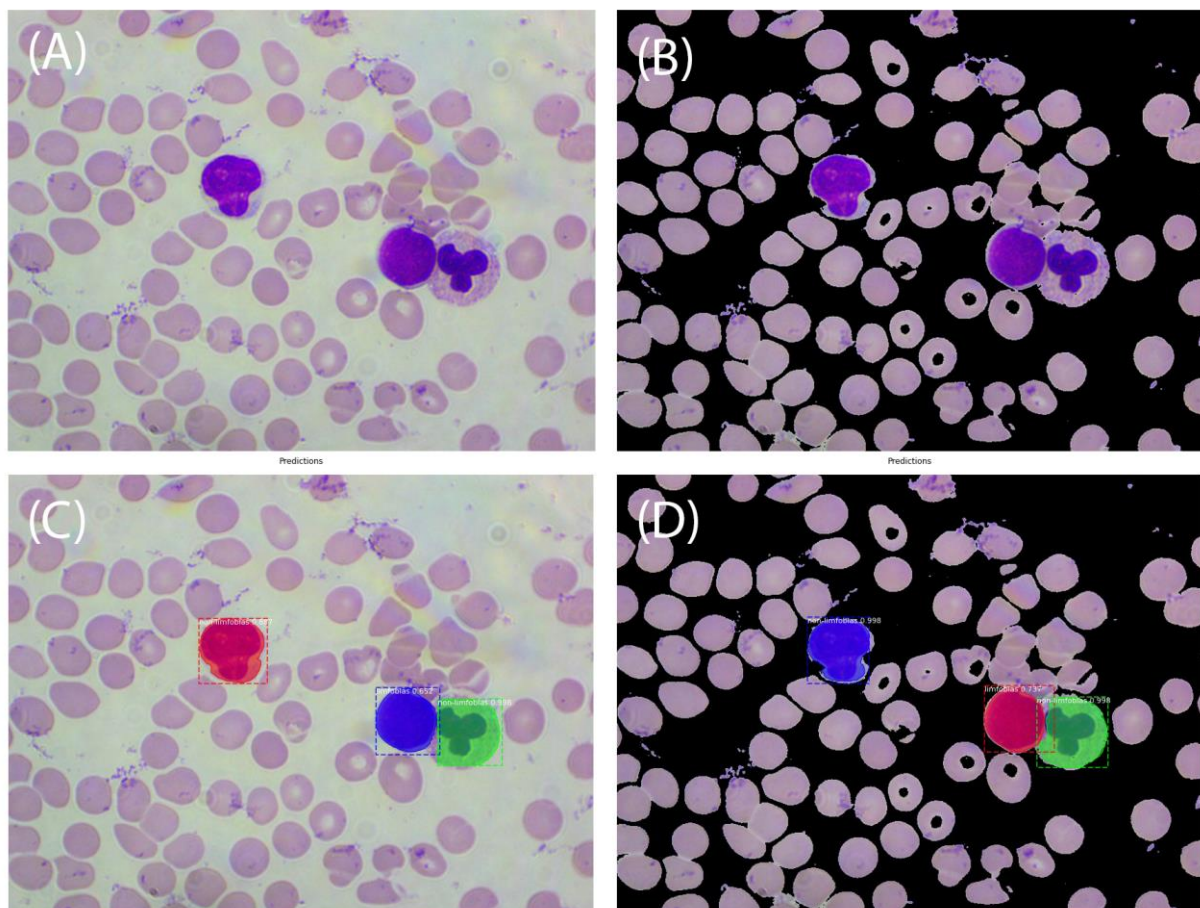
validasi. Nilai *recall* yang hampir 100% untuk citra mentah dan 100% pada citra disegmentasi mengakibatkan nilai akurasi hampir sama dengan nilai presisi. Berdasarkan nilai presisi, *recall*, akurasi, *F1-Score*, dan *mAP dataset* dengan citra disegmentasi memberikan performa yang hampir sama dengan data validasi. Namun, citra mentah memberikan performa lebih turun dibandingkan data validasi. Oleh karena itu, berdasarkan hasil performa data uji dan data validasi, citra yang disegmentasi dapat menghasilkan performa lebih *robust* dan membutuhkan waktu lebih sedikit dalam pelatihan dibandingkan citra tanpa adanya segmentasi.

F. Evaluasi Citra Hasil Prediksi

Hasil proses pendeteksian pada data uji berupa citra yang telah diberi *bounding box*, *mask*, dan *confidence score* pada sel darah putih yang terdeteksi. Pada Gambar 16. dan Gambar 17. merupakan contoh hasil prediksi. Setelah model dilatih, model dapat dijalankan secara *real time* atau mode *inference*. Pada penelitian ini *inference* dilakukan menggunakan data uji. Pada tahap pelatihan model sebelumnya, *hyperparameter* untuk *detection NMS threshold* dan *RPN NMS threshold* digunakan nilai *default* dari Mask R-CNN, yaitu bernilai 0,3 dan 0,7. Nilai *RPN NMS threshold* memiliki nilai yang lebih tinggi bertujuan untuk menghasilkan lebih banyak proposal saat pelatihan. Namun, nilai *detection NMS threshold* yang lebih rendah bertujuan agar tidak banyak *bounding box* yang dihasilkan untuk objek yang sama.

Citra hasil prediksi pada Gambar 16. untuk citra mentah dan citra disegmentasi dapat mendeteksi sel darah putih secara benar dalam hal kelas, *bounding box*, dan *mask*. Namun, hal yang membedakan untuk kedua *dataset* tersebut adalah *confidence score* yang dihasilkan. Dapat dilihat nilai *confidence score* untuk citra mentah yang dihasilkan adalah 0,887; 0,652; dan 0,998, sedangkan untuk citra disegmentasi adalah 0,998; 0,737; dan 0,998. Untuk mendeteksi kelas sel darah putih yang sama citra disegmentasi memiliki *confidence score* yang lebih unggul dibandingkan citra mentah. Hal tersebut terjadi karena model prediksi untuk citra disegmentasi dilatih pada *detection minimum confidence* yang lebih tinggi dibandingkan citra mentah.

Hasil prediksi data uji pada citra mentah terdapat satu *false negative* yang ditampilkan pada Gambar 17. bagian (C). Pada citra tersebut terdapat empat sel limfoblas sebagai *ground truth*, tetapi dua dari sel tersebut berimpit. Model yang dilatih pada citra mentah hanya dapat mendeteksi tiga sel limfoblas dengan *confidence score* masing-masing 0,817; 0,913; dan 0,764. Namun, untuk model yang dilatih pada citra disegmentasi dapat mendeteksi empat sel limfoblas dengan lebih baik, *confidence score* masing-masing deteksi adalah 0,994; 0,995; 0,659; dan 0,915. Sel limfoblas yang tidak dapat dideteksi pada citra mentah, tetapi dapat dideteksi pada citra disegmentasi dengan *confidence score* sebesar 0,659. Hal tersebut dapat terjadi karena *confidence score* sel limfoblas pada citra mentah di bawah nilai 0,4 atau *nilai threshold NMS*



Gambar 16. Perbedaan hasil *confidence score* pada citra mentah dan citra disegmentasi. (A) Citra mentah sebelum prediksi, (B) Citra disegmentasi sebelum prediksi, (C) Citra mentah setelah prediksi, (D) Citra disegmentasi setelah prediksi.

terlalu kecil. Jika *detection minimum confidence* saat *inference* diturunkan kurang dari 0,4 untuk citra mentah maka akan muncul *false positive* yang lebih banyak. Namun, jika nilai *threshold NMS* dinaikkan maka dihasilkan *bounding box* yang lebih banyak pada objek lain. Penentuan nilai *detection NMS threshold* dan *RPN NMS threshold* dilakukan dengan cara melihat langsung hasil prediksi pada data validasi. Citra hasil prediksi tersebut akan terlihat apakah terdapat lebih dari satu *bounding box* untuk setiap sel darah putih.

Hasil evaluasi citra dan performa model deteksi menyimpulkan bahwa nilai *NMS* optimal saat *inference* adalah 0,20 untuk kedua *dataset*. Pada Tabel IX. adalah rangkuman performa model terbaik (dilatih pada citra disegmentasi). Pada dua model yang dilatih pada dua jenis *dataset* yang berbeda, yaitu citra mentah dan citra disegmentasi menunjukkan bahwa dengan adanya segmentasi citra sebelum proses pelatihan dan *inference* dapat membuat model lebih *robust* pada data uji dan membutuhkan waktu lebih sedikit dalam pelatihan (ditunjukkan pada Tabel V.) karena informasi yang tidak diperlukan selama proses pelatihan telah dihilangkan.

TABEL IX.
RANGKUMAN PERFORMA MODEL TERBAIK

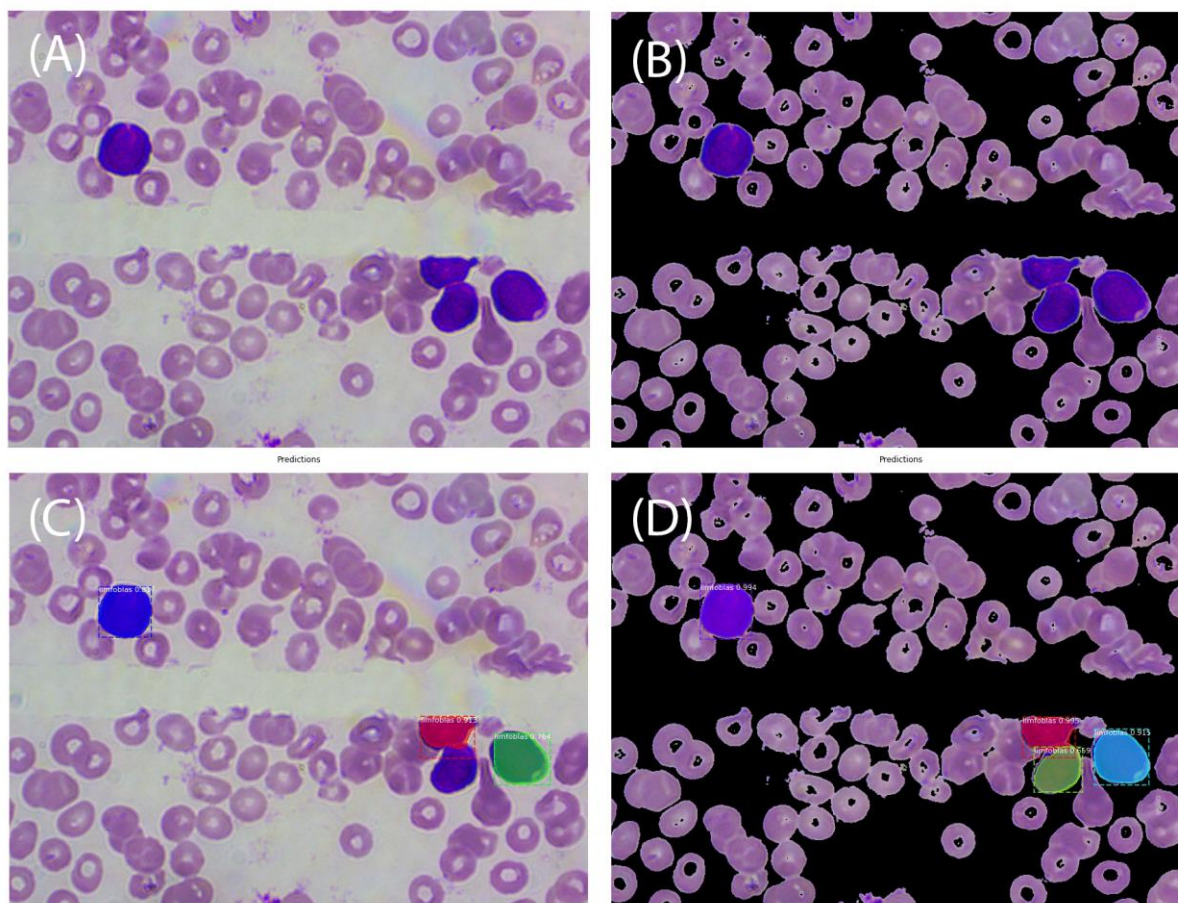
Data	Presisi	Recall	Akurasi	F1-Score	mAP
Data validasi	0,9048	1,0000	0,9048	0,9500	0,9225
Data uji	0,9159	1,0000	0,9159	0,9561	0,9275

G. Evaluasi Perhitungan Otomatis Sel Limfoblas dan Non-Limfoblas

Evaluasi perhitungan otomatis dilakukan pada kedua model untuk *dataset* citra mentah dan citra disegmentasi yang ditampilkan pada Tabel X. Evaluasi perhitungan otomatis dilakukan pada dua kelas, yaitu limfoblas dan non-limfoblas dengan menghitung MAE. Berdasarkan hasil evaluasi MAE pada sel limfoblas dan non-limfoblas tersebut membuktikan bahwa model yang dilatih pada citra disegmentasi memiliki MAE yang lebih kecil dibandingkan pada model yang dilatih pada citra mentah. Oleh karena itu, model terbaik pada penelitian ini adalah model yang dilatih dengan citra disegmentasi.

TABEL X.
EVALUASI MAE

Dataset	Kelas	Hitung Aktual	Hitung Otomatis	MAE
Citra Mentah	Limfoblas	59	64	0,1884
	Non-limfoblas	45	44	0,2464
Citra Disegmentasi	Limfoblas	59	65	0,1159
	Non-limfoblas	45	42	0,1014



Gambar 17. Hasil prediksi citra mentah terdapat *false negative*. (A) Citra mentah sebelum prediksi, (B) Citra disegmentasi sebelum prediksi, (C) Citra mentah setelah prediksi, (D) Citra disegmentasi setelah prediksi.

H. Perbandingan Hasil Deteksi WBC Metode Mask R-CNN dengan Penelitian-Penelitian Sebelumnya pada Pasien ALL Tipe L1

Pendeteksian sel kanker atau sel limfoblas pada pasien ALL tipe L1 merupakan penelitian yang telah dilakukan oleh Tim Riset Studio Sensor Visual, Departemen Teknik Nuklir dan Teknik Fisika (DTNTF), Fakultas Teknik, UGM. Penelitian-penelitian tersebut dilakukan dengan berbagai metode deteksi objek YOLOv2 dan *Faster R-CNN*. Pada Tabel XI. merupakan ringkasan perbandingan hasil pendeteksian sel limfoblas pada pasien ALL tipe L1 menggunakan metode *Mask R-CNN* dengan penelitian sebelumnya. Berdasarkan Tabel XI. menunjukkan bahwa metode *Mask R-CNN* yang telah dikembangkan (diajukan) memiliki nilai performa paling baik jika dibandingkan metode deteksi objek yang dilakukan oleh Fajri [20] dan Prabaswara [22]. Nilai performa yang lebih baik tersebut berupa akurasi, *F1-Score*, dan mAP. Kelebihan fitur deteksi objek yang dikembangkan pada penelitian ini dibandingkan penelitian-penelitian sebelumnya adalah menggunakan pengolahan citra untuk menfokuskan model terhadap sel darah (segmentasi) serta mengurangi waktu pelatihan dan dapat melakukan klasifikasi sel dalam level piksel-ke-piksel (*object mask*).

TABEL XI.
PERBANDINGAN HASIL PENELITIAN OLEH TIM RISET STUDIO
SENSOR VISUAL DTNTF

Nama Peneliti	Tahun	Tujuan	Model	Hasil
Arnanda Prabaswara [22]	2020	Pendeteksian sel limfoblas dan non-limfoblas.	YOLOv2	Nilai akurasi dan mAP adalah 90,9% dan 87,5%.
M. Inska Fajri [20]	2020	Pendeteksian sel limfoblas dan non limfoblas.	<i>Faster R-CNN</i>	Nilai akurasi, <i>F1-Score</i> , dan mAP adalah 86,2%, 87,9%, dan 90,9%.
Mahatma Ageng Wisesa (diajukan)	2021	Pendeteksian dan segmentasi sel limfoblas dan non-limfoblas.	<i>Mask R-CNN</i>	Nilai akurasi, <i>F1-Score</i> , dan mAP adalah 91,59%, 95,61%, dan 92,75%.

IV. KESIMPULAN

Penelitian yang dilakukan telah berhasil mengembangkan model deteksi objek dan segmentasi *instance* untuk sel limfoblas dan non-limfoblas pasien ALL tipe L1. Nilai akurasi, *F1-Score*, dan mAP dapat ditingkatkan dengan melakukan *hyperparameter tuning* pada *detection minimum confidence*, *detection NMS threshold*, dan *RPN NMS threshold*. Teknik pengolahan citra yang digunakan dalam pelatihan *Mask R-CNN* dapat meningkatkan akurasi, *F1-Score*, dan mAP serta mengurangi waktu pelatihan. Model terbaik pada penelitian ini diperoleh nilai akurasi, *F1-Score*, dan mAP sebesar 91,59%, 95,61%, dan 92,75% yang memiliki performa lebih baik dibandingkan penelitian-penelitian sebelumnya.

REFERENSI

- [1] Labati, R. D., Piuri, V., dan Scotti, F. "The acute lymphoblastic leukemia image database for image processing". 2011.
- [2] M.A. Alsalem, A .A . Zaidan, B.B. Zaidan, M. Hashim, H.T. Madhloom, dan N.D. Azeez. "A review of the automated detection and classification of acute leukaemia: Coherent taxonomy, datasets, validation and performance measurements, motivation, open challenges and recommendations". 2018.
- [3] Dipti Patra, Subrajeet Mohapatra, dan Sanghamitra Satpathi. "Image analysis of blood microscopic images for acute leukemia detection". 2010.
- [4] Jakkrich Laosai dan Kosin Chamnongthai. "Acute leukemia classification by using SVM and K-Means clustering". 2014.
- [5] Van-Nhan Tran, Waidah Ismail, Rosline Hassan, dan Atsuo Yoshitaka. "An automated method for the nuclei and cytoplasm of Acute Myeloid Leukemia detection in blood smear images". 2016.
- [6] Kirk D. Wyatt dan Richard J. Bram. "Immunotherapy in pediatric B-cell acute lymphoblastic leukemia". 2019.
- [7] J. M. Bennett, D. Catovsky, Marie-Therese, G. Flandrin, D. A. G. Galton, H. R. Gralnick, dan C. Sultan. "Proposals for the Classification of the Acute Leukaemias". 1976.
- [8] Gamal Abdul-Hamid. "Classification of Acute Leukemia". 2011.
- [9] Arif Muntasaa dan Muhammad Yusuf. "Modeling of the Acute Lymphoblastic Leukemia Detection based on the Principal Object Characteristics of the Color Image". 2019.
- [10] Dipti Patra dan Subrajeet Mohapatra. "Automated Cell Nucleus Segmentation and Acute Leukemia Detection in Blood Microscopic Images". 2010.
- [11] Najmeddine Dhieb, Hakim Ghazzai, Hichem Besbes, dan Yehia Massoud. "An Automated Blood Cells Counting and Classification Framework using Mask R-CNN Deep Learning Model". 2019.
- [12] George P. Brownman, Peter B. Neame, dan Praniti Soamboonsrup. "The contribution of cytochemistry and immunophenotyping to the reproducibility of the FAB classification in acute leukemia". Blood, 68:900-905,1986.
- [13] Nengliang Ouyang, Weijia Wang, Li Ma, Yanfang Wang, Qingwu Chen, Shanhong Yang, Jinye Xie, Shaoshen Su, Yin Cheng, Qiong Cheng, Lei Zheng, dan Yong Yuan. "Diagnosing acute promyelocytic leukemia by using convolutional neural network". 2021.
- [14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, dan Li Fei-Fei. "ImageNet Large Scale Visual Recognition Challenge". 2015.
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, dan Jian Sun. "Deep Residual Learning for Image Recognition". 2015.
- [16] Kaiming He, Georgia Gkioxari, Piotr Dollar, dan Ross Girshick. "Mask R-CNN". 2018.
- [17] Ahmed T. Sahlol , Philip Kollmannsberger, dan Ahmed A. Ewees. "Efficient Classification of White Blood Cell Leukemia with Improved Swarm Optimization of Deep Features". 2020.
- [18] De Rong Loh, Wen Xin Yong, Jullian Yapeter, Jullian Yapeter, Karupppasamy Subburaj, dan Rajesh Chandramohanadas. "A deep learning approach to the screening of malaria infection: Automated and rapid cell counting, object detection and instance segmentation using Mask R-CNN". 2020.
- [19] Yang-Hsien Lin, Ken Y.-K. Liao, dan Kung-Bin Sung. "Automatic detection and characterization of quantitative phase images of thalassemic red blood cells using a mask region-based convolutional neural network". 2020.
- [20] Muhamad Inska Fajri. Pendeteksian Sel Limfoblas pada Kasus Acute Lymphoblastic Leukemia Subtipe L1 Berbasis Convolutional Neural Networks dengan Metode Faster Region-Based Convolutional Neural Networks. Skripsi, Jurusan Teknik Fisika, Fakultas Teknik, Universitas Gadjah Mada, Yogyakarta, 2020.
- [21] Rohan Khandekar, Prakhyha Shastry, Smruthi Jaishankar, Oliver Faust, dan Niranjana Sampathila. "Automated blast cell detection for Acute Lymphoblastic Leukemia diagnosis". 2021.
- [22] Arnanda Prabaswara. Deteksi Sel Limfoblas Pada Kasus Acute Lymphoblastic Leukemia Tipe L1 menggunakan Metode Convolutional Neural Network. Skripsi, Jurusan Teknik Fisika, Fakultas Teknik, Universitas Gadjah Mada, Yogyakarta, 2020.