

Original Research

# YOLOv5 untuk Menghitung Sel Darah Merah dan Sel Darah Putih

Billy Renatasiva <sup>1\*</sup>, Njoto Benarkah <sup>1</sup>, Mohammad Farid Naufal <sup>1</sup>

<sup>1</sup> Teknik Informatika, Fakultas Teknik, Universitas Surabaya, Surabaya-Indonesia

\* corresponding author: [brenatasiva@gmail.com](mailto:brenatasiva@gmail.com)

**Abstract**—Health practitioners use hemocytometer to manually counting the blood cells, and it is considered time-consuming, arduous, and expert-dependent. Automated methods are costly, require meticulous maintenance, can lead to misidentify abnormal cells. This research proposed an application that swiftly, precisely, and easily count red and white blood cells. YOLOv5 is used to detect red and white blood cells in digital images. The model is trained on BCCD dataset and BCCD+ALL-IDB1 using YOLOv5s configuration and 736x736 image input size, and achieve 89.9% mAP50 value for red blood cell, 99.4% for white blood cell, and 93.8% for all classes using BCCD dataset. About 17.7% mean absolute percentage error (MAPE) is obtained using YOLO5x configuration with 416x416 image input size tested on BCCD dataset. The YOLOv5s configuration setup with 736x736 image input size gives 10.9% error rate against ALL-IDB1 dataset. The system is developed using Laravel and Flask, and it proficiently detects and counts red and white blood cells.

**Keywords:** YOLOv5, computer vision, blood count, red blood cell, white blood cell

**Abstrak**—Metode perhitungan sel darah merah secara manual menggunakan hemocytometer membutuhkan waktu yang lama serta melelahkan dan sangat bergantung kepada tenaga ahli di bidang medis, sedangkan perhitungan otomatis menggunakan alat membutuhkan biaya yang mahal dan perawatan ekstra untuk menjaga hasil yang akurat serta apabila dihadapkan dengan sel berukuran abnormal maka alat akan salah mengidentifikasi, maka dibutuhkan suatu sistem yang dapat menghitung jumlah sel darah merah dengan cepat, akurat, dan mudah untuk dioperasikan. YOLOv5 digunakan untuk mendeteksi sel darah merah melalui citra digital. Model dilatih menggunakan BCCD Dataset dan BCCD + ALL-IDB1. Berdasarkan nilai mAP50 didapatkan model terbaik dengan konfigurasi YOLOv5s dengan ukuran masukan 736x736 dengan data latih yaitu BCCD sebesar 89.9% untuk sel darah merah, 99,4% untuk sel darah putih dan 93,8% untuk semua kelas. Berdasarkan *mean absolute percentage error* (MAPE) didapatkan sebesar 17.7% untuk konfigurasi YOLOv5x dengan ukuran citra masukan 416x416 jika diuji menggunakan data BCCD, namun konfigurasi YOLOv5s dengan ukuran citra masukan 736x736 memiliki nilai lebih rendah yaitu 10.9% jika dihadapkan dengan data uji ALL-IDB1. Sistem dibuat dengan menggunakan Laravel dan Flask. Secara keseluruhan, sistem dapat dengan baik mendeteksi serta menghitung sel darah merah maupun sel darah putih.

**Kata kunci:** YOLOv5, visi computer, hitung sel darah, sel darah merah, sel darah putih

## PENDAHULUAN

Sel darah merah atau *erythrocyte* merupakan salah satu dari tiga jenis sel darah, yaitu sel darah merah, sel darah putih, dan trombosit. Sel darah merah atau yang disebut eritrosit merupakan sel darah yang paling umum, sebanyak 40-45% dari volume keseluruhan darah merupakan sel darah merah. Sel darah merah mengandung protein hemoglobin yang membawa oksigen ke seluruh tubuh (Lee, et al., 2022). Sel darah merah memiliki ukuran diameter kurang lebih 7 - 8 mikrometer dengan tebal 2 mikrometer (Bramantya & Suciati, 2022).

Perhitungan sel darah lengkap atau *complete blood count* (CBC), biasa dilakukan untuk menghitung jumlah sel darah merah, sel darah putih, dan trombosit. Tindakan ini dapat membantu dokter dalam mengevaluasi kesehatan keseluruhan pasien. Normalnya, pria memiliki jumlah sel darah merah sebanyak 4,35-5,65 triliun sel/L sedangkan wanita sebanyak 3,92-5,13 triliun sel/L. Banyaknya jumlah sel darah yang terlalu tinggi atau terlalu rendah pada pasien dapat menandakan masalah medis (Mayo Clinic Staff, 2020). Misalnya, rendahnya jumlah sel darah merah bisa jadi pertanda dari anemia, leukemia, malnutrisi, dan lain-lain. Sedangkan tingginya jumlah sel darah merah bisa jadi pertanda dari dehidrasi, penyakit jantung, penyakit paru-paru, kanker ginjal, dan lain-lain (National Library of Medicine, 2021).

Para tenaga medis profesional melakukan pengujian di laboratorium dengan menggunakan mikroskop secara manual untuk menghitung jumlah sel darah merah. Namun pengujian manual memakan waktu dan melelahkan sekalipun bagi tenaga medis profesional.

Selain itu, perhitungan manual kurang dapat diandalkan dikarenakan banyaknya jumlah sel darah pada sampel yang perlu dihitung (Tran et al., 2019). Pada abad ke-20-an para ilmuwan mulai mencoba cara lain yang lebih efektif yaitu dengan menggunakan pemrosesan citra untuk meningkatkan akurasi dari perhitungan sel darah. Namun, citra yang dihasilkan oleh tangkapan dari mikroskop biasanya beresolusi rendah, buram, dan memiliki *noise*. Selain itu, terkadang sel darah saling menumpuk satu dengan yang lain (Lee et al., 2022).

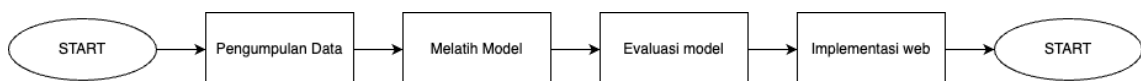
Tran et al (2019) sebelumnya telah melakukan penelitian menghitung sel darah. Penelitian ini menghasilkan akurasi sebesar 93,30% dengan menggunakan Seg-Net yang menghasilkan *segmentation mask* sebagai luaran, namun untuk menanggulangi masalah sel darah yang menumpuk harus dilakukan perhitungan menggunakan *euclidean distance* untuk mendapatkan lokasi dari tiap sel. Alam dan Islam (2019) juga meneliti tentang identifikasi dan menghitung sel darah menggunakan arsitektur YOLO berhasil mendapatkan akurasi sebesar 96.09%, namun arsitektur ini kesusahan dalam mendeteksi objek yang kecil dan berdekatan.

Penelitian serupa telah dilakukan oleh Lee, et al (2022) menggunakan VGG-16 sebagai tulang belakang arsitekturnya. *Dataset* yang digunakan adalah BCCD dengan data latih sel darah merah sebanyak 3.310 dan data validasi sel darah merah sebanyak 843. Secara umum model dapat mendeteksi sel darah merah dengan nilai yang cukup baik, namun model memiliki kelemahan jika diminta untuk mendeteksi sel darah merah yang tidak sempurna (Lee, et al., 2022).

Penelitian ini melakukan penghitungan sel darah secara otomatis sehingga bisa membantu keterbatasan penghitungan secara manual. Berdasar penelitian sebelumnya yang menggunakan Tiny YOLO sehingga mendapatkan performa akurasi tinggi, penelitian ini menggunakan tiga versi YOLO lainnya untuk diimplementasikan dalam suatu bentuk aplikasi *web*, dan membandingkan hasil yang didapat dengan harapan mendapatkan hasil yang sebanding dengan menggunakan tambahan sebuah *dataset* yang berbeda dari penelitian sebelumnya. Berdasar permasalahan tersebut, dibutuhkan suatu metode yang dapat membantu dokter dalam menghitung jumlah sel darah merah pada suatu citra yaitu dengan cara membuat aplikasi untuk otomasi perhitungan sel darah merah menggunakan metode *deep learning*. Sistem dibuat dalam bentuk aplikasi karena penggunaan sistem ini hanyalah untuk keperluan laboratorium sehingga kebutuhan aksesnya hanya melalui lokal.

## METODE

Metode penelitian ini menggunakan empat tahapan utama, mulai dari pengumpulan data, melatih model deteksi objek, mengevaluasi model yang telah dilatih, dan mengimplementasikan model menjadi aplikasi berbasis *web*. Gambar 1 merupakan *flowchart* dari metode yang dipakai pada penelitian ini. Penelitian ini dimulai dengan pengumpulan data. Data yang telah tersedia dilatih dengan menggunakan model YOLOv5s. Performa model dievaluasi dengan menggunakan beberapa pengukuran yang selanjutnya model yang sudah dilatih ini diimplementasikan dalam bentuk *web*.



Gambar 1. Metodologi penelitian terbagi dalam empat tahapan: pengumpulan data, melatih model, evaluasi model, dan implementasi *web*.

### Pengumpulan Data

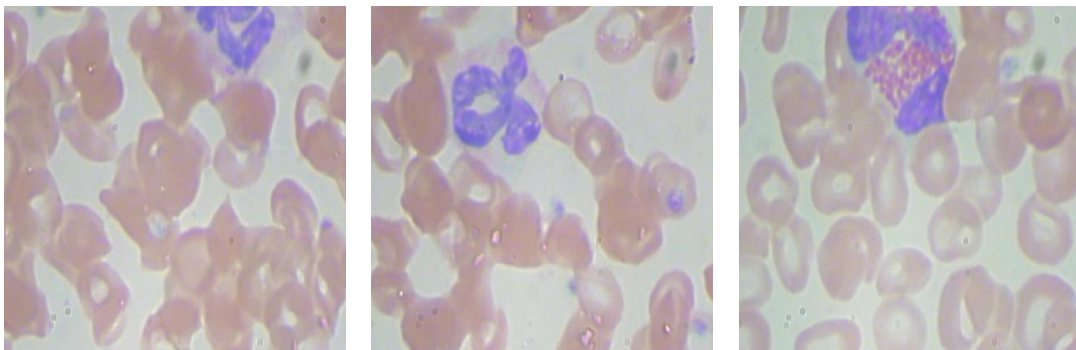
*Dataset* yang digunakan dalam penelitian ini adalah *dataset* publik, yaitu *Blood Cell Detection Image Dataset* (BCCD) dan *Acute Lymphoblastic leukaemia Image Database* (ALL-IDB). *Dataset* BCCD ini memiliki 364 citra yang terbagi dalam tiga kategori label, yaitu sel darah putih (*white blood cell* yang berlabel WBC), sel darah merah (*red blood cell* yang berlabel RBC), dan trombosit (*Platelets*). Tabel 1 berisi detail distribusi label kategori yang dimiliki BCCD.

Sebagian besar label adalah sel darah merah yang menunjukkan dalam setiap citra didominasi oleh sel darah merah. Contoh citra sel darah yang digunakan dalam penelitian ini dapat dilihat di Gambar 2. Penelitian ini menggunakan dataset ALL-IDB hanya proses testing dari model yang telah dilatih.

**Tabel 1**

*Detail Jumlah Untuk Setiap Label Dalam BCCD*

No	Nama Label	Jumlah	%
1	RBC	4.155	85
2	WBC	372	7,61
3	Platelets	361	7,39
Jumlah sel terlabeli		4.888	100



Gambar 2. Contoh citra sel darah.

### Melatih Model

Model yang dipilih untuk penelitian ini adalah YOLOv5, terdapat tiga konfigurasi dari YOLOv5 yang dicoba yakni, YOLOv5s (*small*) dengan ukuran citra sebagai masukan model sebesar 416x416 piksel, YOLOv5s (*small*) dengan ukuran citra sebagai masukan model sebesar 736x736 piksel, dan YOLOv5x (*XLarge*) dengan ukuran citra sebagai masukan model sebesar 416x416 piksel. Model berupa model untuk mendeksi objek, dan untuk melatih model digunakan *dataset* yang sudah diberi label berupa *bounding box* pada tiap sel dengan menggunakan fasilitas yang ada dalam *Roboflow*. *Dataset* yang telah dikumpulkan dibagi menjadi tiga bagian yaitu data latih (*training*), data validasi (*validation*), dan data uji (*test*). Ketiga bagian tersebut akan dibagi menjadi 70% untuk data latih, 20% untuk data validasi, dan 10% untuk data uji. Total tiap data yang didapatkan setelah dibagi adalah 255 citra dipakai sebagai data latih, 73 citra dipakai sebagai data validasi, dan 36 citra dipakai sebagai data uji. Pelatihan model dilakukan dengan menggunakan jumlah *epoch* sebesar 100 dan jumlah *batch* sebesar 16 menggunakan GPU Tesla T4 yang disediakan oleh *Google Colaboratory*.

### Evaluasi Model

Pengukuran performa model menggunakan *mean average precision* (mAP) dan *mean absolute percentage error* (MAPE). *Mean average precision* dipakai untuk mengukur akurasi dari semua label yang ada dengan menghitung nilai rata-rata dari nilai *average precision* (AP) di tiap kelas. *Average precision* merupakan nilai area di bawah kurva *precision-recall*, yakni semakin tinggi nilainya semakin bagus. mAP50 merupakan nilai yang dihitung dari AP dengan nilai *intersection over union* (IOU) 50%, sedangkan mAP50-95 50% hingga 95%. Nilai IoU didapatkan dari menghitung area yang bertumpukan dibagi dengan total area antara *bounding box* dari prediksi dengan *bounding box* dari *ground truth*. *Mean absolute percentage error*

(MAPE) menghitung jumlah kesalahan yang terjadi pada saat perhitungan objek dalam bentuk persentase dan selalu bernilai positif.

### Implementasi Web

Proses yang terakhir adalah pengimplementasian model yang telah dilatih di tahap sebelumnya ke *web*. Model *object detection* digunakan untuk mendeteksi objek sesuai dengan kelas yang telah ditentukan sebelumnya. Proses pendeteksian ini akan menghasilkan sebuah citra dengan *bounding box*. Setelah proses pendeteksian selesai, maka akan dilakukan penghitungan jumlah objek yang berhasil terdeteksi sesuai dengan kelas yang telah ditentukan sebelumnya. Masukan berupa citra sel darah didapatkan menggunakan mikroskop lalu diunggah ke web berupa citra hasil foto. Web dibuat dengan menggunakan *framework* dari *laravel* dan *flask* digabungkan dengan *Python*. Aplikasi dalam web ini memanggil sebuah fungsi yang tersedia di *server* setelah tombol hitung ditekan, lalu hasil citra beserta *bounding box* dan jumlah obyek yang berhasil terdeteksi menurut kelasnya akan dikirimkan kembali ke *web browser* dari *server*.

### HASIL

*Training* model dilakukan sebanyak tiga kali dengan konfigurasi model yang berbeda. Model pertama yang dipakai adalah YOLOv5s dengan citra yang dipakai sebagai masukan memiliki ukuran sebesar 416x416 piksel, kemudian YOLOv5s dengan ukuran citra masukan sebesar 736x736, dan YOLOv5x dengan ukuran citra masukan sebesar 416x416 piksel. Hasil dari tiap konfigurasi model untuk tiap-tiap kelas dapat dilihat pada Tabel 2. Sedangkan hasil MAPE untuk tiap konfigurasi model dapat dilihat pada Tabel 3.

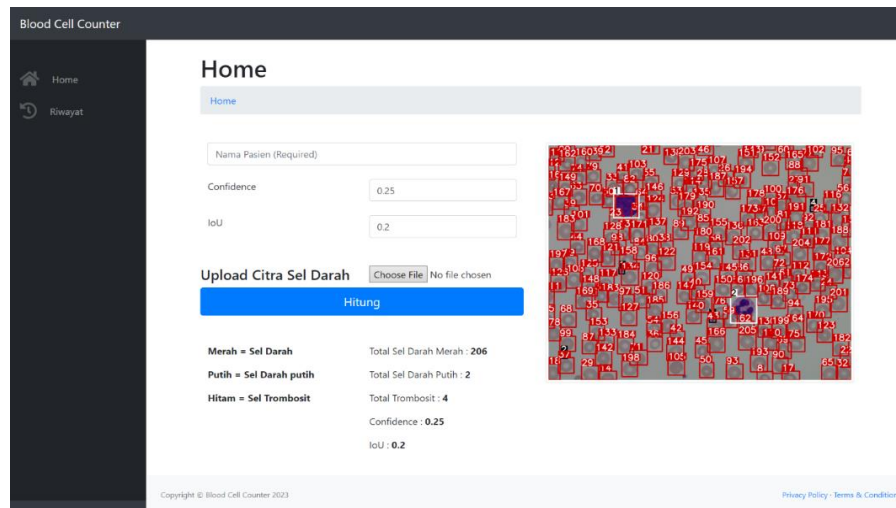
### BAHASAN

Ketiga model dilatih menggunakan *dataset* BCCD dan kombinasi *dataset* BCCD+ALL-IDB. Performa latih diukur dengan menggunakan mAP50 dan mAP50-95. Tabel 2 menunjukkan hasil dari 32 eksperimen dilakukan menggunakan model YOLOv5s dan YOLOv5x dengan dua jenis ukuran citra, yaitu 416x416 piksel dan 736x736 piksel. Hasil latih model dengan menggunakan semua kelas yang ada memberikan nilai mAP50 tertinggi sebesar 93,8%, sedangkan untuk kelas sel darah merah memberikan nilai mAP50 tertinggi sebesar 89,9% dan untuk kelas sel darah putih memberikan nilai mAP50 tertinggi sebesar 99,4%. Ketiga eksperimen tersebut menggunakan data latih BCCD dan ukuran citra 736x736 piksel. Model yang dipakai dari ketiga eksperimen berdasar nilai mAP50 tertinggi tersebut menggunakan YOLOv5s.

Tabel 3 memberikan hasil uji terhadap model YOLOv5s dan YOLOv5x untuk dua jenis ukuran citra yang sama dan *dataset* yang dipakai untuk uji adalah BCCD dan ALL-IDB. Persentase kesalahan uji model diukur menggunakan MAPE untuk mengukur persentase kesalahan dari model yang telah dilatih. Persentase kesalahan terendah didapatkan oleh model YOLOv5s yang dilatih menggunakan citra berukuran 736x736 piksel dan *dataset* kombinasi BCCD+ALL-IDB, *dataset* uji yang dipakai adalah ALL-IDB dengan nilai MAPE sebesar 10,9%. Nilai MAPE 97,6% untuk data latih BCCD dan *dataset* uji ALL-IDB menunjukkan indikasi terjadinya *overfitting* selama proses latih model YOLOv5s dengan ukuran citra sebesar 736x736. Hal ini terlihat dari eksperimen yang menggunakan *dataset* latih BCCD dan data uji ALL-IDB, dibandingkan dengan hasil eksperimen yang menggunakan *dataset* latih BCCD dan *dataset* uji ALL-IDB, maupun hasil eksperimen yang menggunakan *dataset* latih kombinasi BCCD+ALL-IDB dan *dataset* uji BCCD atau ALL-IDB. Hasil latih model menggunakan *dataset* latih kombinasi BCCD+ALL-IDB mempunyai persentase kesalahan kurang dari 30% dan cenderung rendah, sedangkan hasil latih model yang menggunakan data latih BCCD mempunyai nilai MAPE yang bervariasi dan memiliki kecenderungan tidak bisa menangani data baru dari *dataset* yang berbeda kecuali pada eksperimen menggunakan model YOLOv5s dengan citra

berukuran 416x416 piksel yang mempunyai nilai MAPE sekitar 32% baik untuk *dataset* uji BCCD ataupun *dataset* uji ALL-IDB.

Model YOLOv5s dipakai untuk diimplementasikan dalam bentuk *web*. Implementasi *web* dari model yang dilatih dapat melakukan penghitungan jumlah sel darah merah dan sel darah putih serta memberikan *bounding box* pada tiap sel tersebut. Gambar 3 menunjukkan suatu contoh hasil perhitungan menggunakan suatu citra yang mengandung banyak sel darah merah. Setiap sel yang berhasil dideteksi mempunyai bentuk visualisasi berupa sebuah *bounding box* bernomor untuk memudahkan identifikasi dan pengecekan penghitungan.



Gambar 3. Tampilan *web* untuk menghitung sel darah pada suatu citra.

Validasi dilakukan untuk mengetahui apakah tujuan tugas akhir ini tercapai melalui program ini. Proses validasi dilakukan dengan cara wawancara dengan seorang spesialis hematologi dan onkologi dan seorang dokter sekaligus dosen di Fakultas Kedokteran Universitas Surabaya. Berdasarkan hasil wawancara tersebut, ditemukan bahwa dengan adanya program seperti ini, dapat membantu tenaga ahli medis profesional untuk menghitung jumlah sel darah merah. Terjadi beberapa kasus yang perlu dilakukan untuk peningkatan hasil pada pengembangan selanjutnya dalam program menghitung sel yang menumpuk sebagai satu sel.

## SIMPULAN

Nilai mAP50 tertinggi sebesar 93,8% didapatkan dari hasil eksperimen menggunakan semua kelas oleh model YOLOv5s dengan citra berukuran 736x736 piksel dan *dataset* latih BCCD. Hasil uji berdasar nilai MAPE menunjukkan hasil dari latih model menggunakan *dataset* latih BCCD saja mempunyai persentase kesalahan yang lebih besar saat diuji menggunakan *dataset* lain. Model yang dilatih dengan menggunakan *dataset* latih kombinasi BCCD+ALL-IDB menunjukkan persentase kesalahan yang rendah saat diuji menggunakan *dataset* BCCD atau ALL-IDB. Penelitian selanjutnya bisa dilakukan untuk mengurangi persentase kesalahan sehingga kemungkinan terjadi salah deteksi bisa diminimalisasi dan penghitungan sel darah bisa lebih akurat.

Beberapa peningkatan lainnya yang dapat diterapkan pada penelitian selanjutnya, yakni penambahan variasi *dataset* agar performa model dapat dengan baik berjalan pada bentuk data yang berbeda. Kemudian, penambahan pra-proses, yaitu dilakukannya segmentasi sebelum melakukan deteksi objek guna menambah performa pendeteksian dan penghitungan sel yang menumpuk.



## PUSTAKA ACUAN

- Alam, M. M., & Islam, M. T. (2019). Machine learning approach of automatic identification and counting of blood cells. *Healthcare Technology Letters*, 6(4), 103–108. <https://doi.org/10.1049/htl.2018.5098>
- Bramantya, A. A., Faticah, C., & Suciati, N. (2022). Bramantya, Faticah, and Suciati-Detection and Classification of Red Blood Cells Abnormality using Faster R-CNN and Graph Convolutional Networks. *Jurnal Ilmiah Teknologi Informasi*, vol. 20, no. 1, 31 Jan. 2022, pp. 23-44, doi: 10.12962/j24068535.v19i3.a1118
- Lee, S.-J., Chen, P.-Y., & Lin, J.-W. (2022). Complete Blood Cell Detection and Counting Based on Deep Neural Networks. *Applied Sciences*, 12(16), 8140. MDPI AG. <http://dx.doi.org/10.3390/app12168140>
- Mayo Clinic Staff. (2020, December 22). Complete blood count (CBC). Mayo Clinic. Diakses tanggal 19 September 2022, dari <https://www.mayoclinic.org/tests-procedures/complete-blood-count/about/pac-20384919>
- National Library of Medicine. (2021, October 4). Red Blood Cell (RBC) Count: MedlinePlus Medical Test. Diakses tanggal 19 September, 2022, from <https://medlineplus.gov/lab-tests/red-blood-cell-rbc-count/>
- Tran, T., Binh Minh, L., Lee, S., & Kwon, K. (2019). Blood Cell Count Using Deep Learning Semantic Segmentation. *Preprints* 2019, 2019090075. <https://doi.org/10.20944/preprints201909.0075.v1>

## BIBLIOGRAFI

- Acharya, V., & Kumar, P. (2017). Identification and red blood cell automated counting from blood smear images using computer-aided system. *Medical & Biological Engineering & Computing*, 56(3), 483–489. <https://doi.org/10.1007/s11517-017-1708-9>
- Alam, M. (2019). Complete-Blood-Cell-Count-Dataset. Diakses tanggal 6 Januari 2023, dari <https://github.com/MahmudulAlam/Complete-Blood-Cell-Count-Dataset>
- Barbedo, J. G. A. (2013). Automatic Object Counting In Neubauer Chambers. In *Anais de XXXI Simpósio Brasileiro de Telecomunicações*. <https://doi.org/10.14209/sbirt.2013.224>
- Batrah, S. (2018, April 6). Total RBC count using hemocytometer / Neubauer's Chamber (Microdilution & Macrodilution Method): Hematology practicals. *Paramedics World*. Diakses tanggal 6 Januari 2023, dari <https://paramedicsworld.com/hematology-practicals/total-red-blood-cell-rbc-count-using-hemocytometer-neubauer-chamber-microdilution-macrodilution/medical-paramedical-studynotes>
- BIT, H. A. (2019, October 15). Semantic Segmentation. Diakses tanggal 6 Januari 2023, dari <https://medium.com/hackabit/semantic-segmentation-8f2900eff5c8>
- Britannica, T. Editors of Encyclopaedia (2022, October 23). Red Blood Cell. *Encyclopedia Britannica*. Diakses tanggal 6 Januari 2023, dari <https://www.britannica.com/science/red-blood-cell>
- Friebe, M. (2017). International Healthcare Vision 2037 - new technologies, educational goals and entrepreneurial challenges ; proceedings + summary of the 5th BME-IDEA EU Conference ; 11 - 13 June 2017, Magdeburg, Germany. doi:10.25673/4992
- Hemalatha, B., Karthik, B., Krishna Reddy, C., & Latha, A. (2022). Deep learning approach for segmentation and classification of blood cells using enhanced CNN. *Measurement: Sensors*, 24, 100582. <https://doi.org/10.1016/j.measen.2022.100582>
- Katsamenis, I., Karolou, E. E., Davradou, A., Protopapadakis, E., Doulamis, A., Doulamis, A., & Kalogeras, D. (2022). TraCon: A Novel Dataset for Real-Time Traffic Cones Detection Using Deep Learning. In: Krouska, A., Troussas, C., Caro, J. (eds) *Novel & Intelligent Digital Systems: Proceedings of the 2nd International Conference (NiDS 2022)*. NiDS 2022. *Lecture Notes in Networks and Systems*, vol 556. Springer, Cham. [https://doi.org/10.1007/978-3-031-17601-2\\_37](https://doi.org/10.1007/978-3-031-17601-2_37)

- Madan, P., Samaya, S. (2020, March 2). An introduction to deep learning. IBM developer. Diakses tanggal 6 Januari 2023, dari [https://developer.ibm.com/articles/an-introduction-to-deep-learning/?mhsrc=ibmsearch\\_a&mhq=+deep+learning](https://developer.ibm.com/articles/an-introduction-to-deep-learning/?mhsrc=ibmsearch_a&mhq=+deep+learning)
- Medline plus. (2021). Red Blood Cell (RBC) Count: MedlinePlus Medical Test. Diakses tanggal 6 Januari 2023, dari <https://medlineplus.gov/lab-tests/red-blood-cell-rbc-count/>
- Microyn Improved Neubauer Hemocytometer, Cell Counting Chamber with Bright Line. (n.d.). Diakses tanggal 6 Januari 2023, dari <https://www.amazon.com/Microyn-Improved-Neubauer-Hemocytometer-Counting/dp/B01M8KKCT3>
- Mount Sinai. (n.d.). Mount Sinai Health System. Diakses tanggal 6 Januari 2023, dari <https://www.mountsinai.org/health-library/tests/rbc-count>
- Overton, T., & Tucker, A. (2020). DO-U-Net for Segmentation and Counting. In M. R. Berthold, A. Feelders, & G. Kreml (Eds.), *Advances in Intelligent Data Analysis XVIII* (pp. 391–403). Cham: Springer International Publishing.
- Publications, M. (2019, August 18). The Computer Vision Pipeline, part 3: Image preprocessing. Manning. Diakses tanggal 6 Januari 2023, dari <https://freecontent.manning.com/the-computer-vision-pipeline-part-3-image-preprocessing/>
- Rais, M. D. A., Arif, F., Arifuddin, M. F., Muhammad, M., Kaswar, A. B., & Prima Putra, K. (2022). Metode Otomatis untuk Menghitung Sel Darah Merah Menggunakan Image Processing. *Journal of Embedded Systems, Security and Intelligent Systems*, 3(2), 102. <https://doi.org/10.26858/jessi.v3i2.38250>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
- Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. MICCAI 2015. *Lecture Notes in Computer Science()*, vol 9351. Springer, Cham. [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
- Vasković, J. (2022, July 6). Erythrocytes. Kenhub. Diakses tanggal 6 Januari 2023, dari <https://www.kenhub.com/en/library/anatomy/erythrocytes>
- Vlab, A. (2011, January 14). Hemocytometer - Counting of cells - Amrita University. Retrieved January 6, 2023, from <https://www.youtube.com/watch?v=MKSOKM3lr90>

**LAMPIRAN**

**Tabel 2**

*Hasil Pengujian Berdasar Konfigurasi yang Berbeda-beda*

Kelas	Dataset latih	Model	Ukuran citra (piksel)	mAP50 (%)	mAP50-95 (%)
Semua	BCCD	YOLOv5s	416x416	91,7	63,8
			736x736	93,8	65,4
		YOLOv5x	416x416	91,8	58,5
			736x736	91,6	63,5
	BCCD+ALL-IDB	YOLOv5s	416x416	92,1	65,5
			736x736	91,1	63,9
	YOLOv5x	416x416	89,9	62,5	
		736x736	90,2	62,7	
Platelets	BCCD	YOLOv5s	416x416	90,5	61,4
			736x736	92,0	49,1
		YOLOv5x	416x416	89,8	45,4
			736x736	87,3	43,2
	BCCD+ALL-IDB	YOLOv5s	416x416	89,6	51,4
			736x736	87,2	48,8
	YOLOv5x	416x416	86,3	45,5	
		736x736	88,6	48,0	
Sel darah merah	BCCD	YOLOv5s	416x416	87,1	61,8
			736x736	89,9	66,5
		YOLOv5x	416x416	88,4	59,1
			736x736	88,2	64,9
	BCCD+ALL-IDB	YOLOv5s	416x416	88,5	64,0
			736x736	87,2	62,2
	YOLOv5x	416x416	85,2	61,0	
		736x736	84,2	59,8	
Sel darah putih	BCCD	YOLOv5s	416x416	97,5	78,2
			736x736	99,4	80,7
		YOLOv5x	416x416	97,2	71,1
			736x736	99,4	82,4
	BCCD+ALL-IDB	YOLOv5s	416x416	98,3	81,1
			736x736	98,9	80,8
	YOLOv5x	416x416	98,2	81,1	
		736x736	98,0	80,4	



**Tabel 3**

*Hasil pengujian MAPE*

Model	Ukuran citra (piksel)	Dataset latih	Dataset uji	MAPE (%)
YOLOv5s	416x416	BCCD	BCCD	32,0
			ALL-IDB	32,5
		BCCD+ALL-IDB	BCCD	24,6
			ALL-IDB	11,1
	736x736	BCCD	BCCD	26,5
			ALL-IDB	97,6
		BCCD+ALL-IDB	BCCD	25,6
			ALL-IDB	10,9
YOLOv5x	416x416	BCCD	BCCD	17,7
			ALL-IDB	62,5
		BCCD+ALL-IDB	BCCD	23,2
			ALL-IDB	15,7
	736x736	BCCD	BCCD	28,7
			ALL-IDB	92,5
		BCCD+ALL-IDB	BCCD	30,3
			ALL-IDB	11,4