

Model Object Detection Neural Network Berbasis Hand Gesture Recognition sebagai Kontrol Prostesis Tangan

I.M.P. Arya Winata¹, Karuna K.S Prasad², Ilham Fauzi³, I.M.E. Darmayasa Adiputra⁴, I.W. Widhiada⁵

[Submission: 13-02-2023, Accepted: 29-03-2023]

Abstract— Most people with disabilities due to accident injuries are patients after an arm amputation which can cause psychological disorders and even major trauma. The urgency of hand prosthesis functionality is increasingly needed. Hand gesture recognition (HGR) can be used to control hand prostheses, judging from the similarity in shape of objects/objects that tend to make the same hand movements. This development uses three common types of movement, namely pinch, pick, and grab. Developing a neural network model capable of implementing this concept is necessary. The neural network model developed uses the YOLOV7 and YOLOV7tiny pre-trained networks with datasets collected through the public image data scrapping method. The dataset is 317 images and 2278 object labels with a training ratio of 80:20 testing. The training process uses the Pytorch framework with 300 epochs. The results of the loss values for each epoch show that the model is trainable in the given dataset. The training results are then evaluated by evaluating number of parameters, frame per seconds (FPS) and mean average precision (mAP) using a testing dataset. The overall results show the highest evaluation metrics in the model, with YOLOV7 pretrained with parameter number of 36,9 million, FPS of 161, and mAP of 98,11%. The model has the potential to be developed and implemented as a support for the control functionality of hand prostheses.

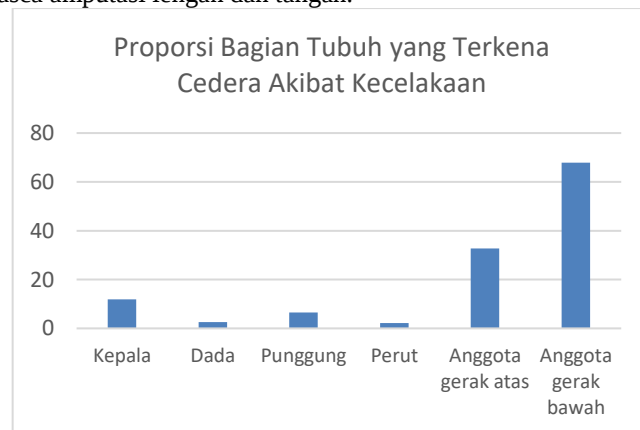
Intisari— Penyandang disabilitas akibat cedera kecelakaan sebagian besar merupakan pasien pasca amputasi lengan yang dapat menyebabkan gangguan psikologis, bahkan trauma besar. Dengan demikian, urgensi fungsionalitas prostesis tangan semakin dibutuhkan. Hand gesture recognition (HGR) dapat dimanfaatkan sebagai kontrol prostesis tangan, meninjau dari kesamaan bentuk objek/benda memberikan kecenderungan terhadap gerakan tangan yang sama. Pengembangan ini menggunakan tiga jenis gerakan yang umum, yaitu pinch, pick, dan grab. Diperlukan pengembangan model neural network yang mampu mengimplementasikan konsep tersebut. Model neural network yang dikembangkan menggunakan pretrained network YOLOV7 dan YOLOV7tiny dengan dataset yang dikumpulkan melalui metode scrapping data gambar publik. Dataset yang diperoleh sejumlah 317 gambar, 2278 label objek dengan rasio training dengan testing 80:20. Proses training menggunakan framework Pytorch dengan 300 epoch. Hasil nilai loss tiap epoch menunjukkan model trainable pada dataset yang diberikan. Hasil training selanjutnya dievaluasi dengan evaluation metrics jumlah

parameter, frame per seconds (FPS) dan mean average precision (mAP) menggunakan testing dataset. Hasil overall menunjukkan evaluation metrics tertinggi pada model dengan pretrained YOLOV7 dengan jumlah parameter 36,9 juta, FPS 161, dan mAP 98,11%. Dengan hasil ini, model memiliki potensi untuk dikembangkan dan diimplementasikan sebagai penunjang fungsionalitas kontrol prostesis tangan.

Kata Kunci— hand gesture recognition, kontrol prostesis tangan, object detection, neural network.

I. PENDAHULUAN

Tidak sedikit penyandang disabilitas yang merupakan pasien pasca amputasi lengan dan tangan akibat cedera dari kecelakaan besar. Pada tahun 2022, Asia Tenggara merupakan salah satu kawasan dengan angka amputasi tinggi [1]. Pasien pasca amputasi lengan dan tangan harus menyesuaikan keterbatasan fisiknya dalam berkegiatan. Hal ini dapat mempengaruhi kondisi psikologis pasien, khususnya pasien dengan trauma besar yang belum mampu menerima diri sepenuhnya. Tingkat depresi dan kecemasan pada pasien pasca amputasi menunjukkan nilai yang relatif tinggi antara 20,6% hingga 63% untuk depresi dan 25,45% hingga 57% untuk kecemasan [2]. Hal ini diperparah dengan fakta bahwa anggota gerak menjadi bagian tubuh yang mengalami cedera terbanyak akibat kecelakaan seperti yang ditunjukkan pada Gambar 1: [3]. Merujuk permasalahan ini, suatu alat bantu prostesis dibutuhkan untuk memperbaiki kualitas hidup pasien pasca amputasi lengan dan tangan.



Gambar 1: Proporsi bagian tubuh yang terkena cedera akibat kecelakaan

Dalam pengembangan prostesis tangan, aspek fungsional menjadi salah satu perhatian yang penting. Hal ini disebabkan karena prostesis harus mampu mensubstitusi semirip mungkin dengan tangan asli yang memiliki gerakan kompleks dan mengambil andil penting dalam kegiatan sehari-hari manusia

p-ISSN:1693 – 2951; e-ISSN: 2503-2372

^{1,2,3,4}Mahasiswa, Jurusan Teknik Mesin Fakultas Teknik Universitas Udayana, Jln. Raya Kampus Unud Bukit Jimbaran 08361 INDONESIA (telp: 0361-703321; e-mail: aryaw2001@gmail.com, iamkaruna09@gmail.com, ilhamg2216@gmail.com, esa.darmayasa@gmail.com)

²Dosen, Jurusan Teknik Mesin Fakultas Teknik Universitas Udayana, Jln. Raya Kampus Unud Bukit Jimbaran 08361 INDONESIA (telp: 0361-703321; e-mail: wynwidhiada@unud.ac.id)



[4]. Merujuk pada permasalahan tersebut, dibutuhkan suatu konsep yang mampu memberikan pemodelan jelas terhadap gerakan tangan pada prostesis tangan.

Dalam aplikasinya, gerakan tangan yang presisi selalu memiliki tantangan dengan adanya background noise yang menyebabkan perlunya desain sistem pengenalan gerakan yang efisien. Oleh karena itu, ketepatan sistem *hand gesture recognition (HGR)* masih memberikan beberapa tantangan bagi para peneliti. Salah satu pendekatan umum untuk rekognisi gerakan tangan statis adalah dengan berbasis computer vision melalui akuisisi data, segmentasi wilayah tangan, ekstraksi fitur, dan klasifikasi gerakan berdasarkan fitur yang teridentifikasi [4,5,6]. Proses akuisisi data untuk pengenalan gerakan tangan berbasis penglihatan dilakukan dengan menggunakan sensor seperti kamera web [7].

Merujuk pada kondisi tersebut, *HGR* memiliki potensi untuk diimplementasikan pada kontrol prostesis tangan, mengingat jenis gerakan tangan cenderung akan memiliki pola tertentu. Pola yang dimaksud ini salah satunya merujuk pada bentuk objek yang hendak diambil. Dengan kata lain, bentuk objek/benda yang mirip akan memiliki kecenderungan gerakan tangan yang sama untuk mengambilnya [8]. Pada penelitian ini, pengembangan model *neural network* dilakukan untuk mencapai fungsionalitas *HGR* tersebut. Hal ini dikarenakan model *neural network* telah menunjukkan kemampuan yang sangat baik dalam menginterpretasikan gambar ke dalam suatu klasifikasi [9]. Terlebih lagi, model *neural network* dalam hal deteksi objek (*object detection*) telah dikembangkan secara lebih spesifik. Pada penelitian ini, pengembangan model *object detection* akan dikaji.

II. PENELITIAN TERKAIT

A. Object Detection Neural Network

Makna dari *object detection* adalah membuat algoritma dalam mengidentifikasi dan mengklasifikasi objek dari sebuah frame/gambar. *Object detection* adalah area penting dari *computer vision* dan memiliki aplikasi penting dalam penelitian ilmiah dan produksi industri praktis, seperti deteksi wajah, deteksi teks, deteksi manusia, deteksi kendaraan, dan deteksi citra medis. Perkembangan dari *object detection* ini dipengaruhi oleh beberapa faktor, seperti kemampuan komputasi dan dataset dalam penerapan *artificial neural network*. Hal ini menyebabkan, algoritma *object detection* tradisional masih sering digunakan, dalam bidang *computer vision*, seperti: *deformable parts model (DPM)* dan *selective search* [10,11]. Arsitektur dasar dari algoritma *object detection* tradisional tersebut telah mampu memberikan interpretasi dasar yang dapat dimanfaatkan dalam bidang bidang yang disebutkan sebelumnya namun tentu masih dapat dikembangkan lagi.

Peningkatan kemampuan komputasi dan ketersediaan jumlah dataset gambar memberikan dampak yang baik dalam pengembangan *object detection* dengan basis *Deep Convolution Neural Network (DCNN)*. Sebelum adanya DCNN, sebuah model CNN diajukan dengan sebutan AlexNet, mampu memberikan nilai terbaik pada *metrics evaluation top-5 score* dengan menggunakan ImageNet dataset [12]. *Top-5 score* ini merupakan metode evaluasi model dengan melihat nilai sebenarnya (*ground truth*) terdapat pada 5 prediksi teratas oleh model. Setelah melihat dampak dari penerapan AlexNet

pada dataset tersebut, pengembangan dari *object detection* semakin pesat dalam penerapan konsep *neural network*. Pengembangan selanjutnya adalah model RCNN (*Region based-CNN*) [13], yang mengajukan konsep *selective search* dalam menentukan objek (*region proposal*) dan dilanjutkan dengan klasifikasi oleh CNN sebagai *feature extraction* pada setiap objek yang didapat. Metode ini dianggap sebagai multi-stage detector karena melalui dua tahapan utama, yaitu penentuan region objek dan training klasifikasi objek. Sebagai optimalisasi, sistem *object detection* dengan konsep single-stage detector diajukan dengan sebutan YOLO (*You Only Look Once: Unified, Real-Time Object Detection*) [14,15]. *Single-stage detector* YOLO ini dibuat dengan konsep tanpa adanya *region proposal* seperti pada *multi-stage detector*, melainkan hanya dengan membagi gambar menjadi beberapa grid dan menentukan sebuah objek terdapat pada grid yang mana. Konsep ini memberikan hasil implementasi *object detection* yang sangat cepat.

B. Real-Time Inference Object Detection

Object detection telah menawarkan akurasi tinggi dalam melakukan penentuan dan klasifikasi objek sehingga hal ini dapat dimanfaatkan dalam proses otomatisasi. Namun, sebagai suatu solusi dalam otomatisasi, model *object detection* harus mampu beradaptasi pada faktor *affordability* dan *compatibility* [16]. *Affordability* menjelaskan bahwa model *object detection* yang baik mampu dijangkau oleh masyarakat dalam hal biaya dan material, sehingga mampu dikomersialisasi secara lebih komprehensif. Sementara itu, *compatibility* memiliki peran yang lebih penting, yaitu mampu menyesuaikan model agar dapat berjalan pada komputer yang sesuai dengan kebutuhan bentuk, yang cenderung kecil. Hal ini disebabkan oleh kecenderungan implementasi *object detection* digunakan dalam menentukan posisi dan jenis objek secara langsung (*real-time*), seperti aplikasi pada pendeteksi penggunaan masker medis [17]. Sehingga menjadi sebuah tantangan ketika implementasi ini diterapkan di tempat deteksi secara langsung yang membutuhkan penyesuaian pada bentuk yang relatif kecil.

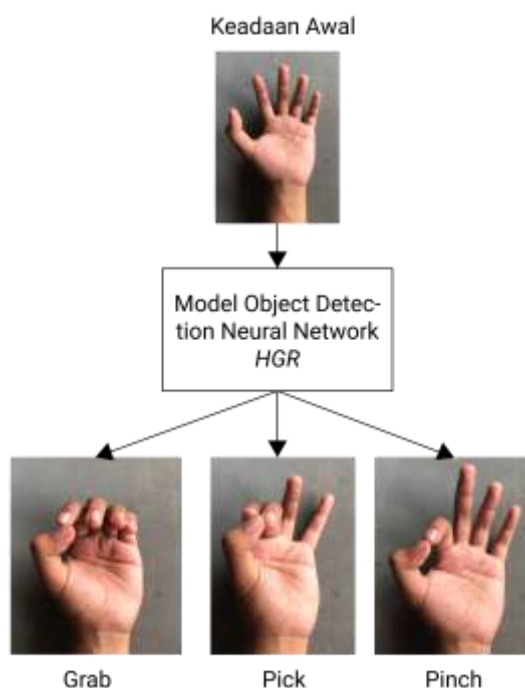
Merujuk pada faktor *compatibility*, model *object detection* juga harus mampu memberikan otomatisasi yang beresesuaian dengan yang dibutuhkan, terutama pada waktu *inference* model. Waktu *inference* ini akan berdampak pada kelancaran dari menginterpretasikan hasil posisi dan klasifikasi objek. Waktu *inference* pada model *object detection* besar dipengaruhi oleh jumlah parameter yang dimuat oleh model. Hal ini dikarenakan jumlah parameter menunjukkan jumlah komputasi yang dilakukan oleh model dalam interpretasi sebuah gambar. Terkait kajian antara *multi-stage detector* dengan *single-stage detector* pernah dilakukan dalam parameter inference time. Secara garis besar, *single-stage detector* memberikan waktu *inference* yang lebih cepat dibandingkan dengan multi-stage detector. Sehingga penerapan *single-stage detector* lebih banyak dilakukan pada bidang deteksi objek yang cenderung dinamis/bergerak.

III. METODE

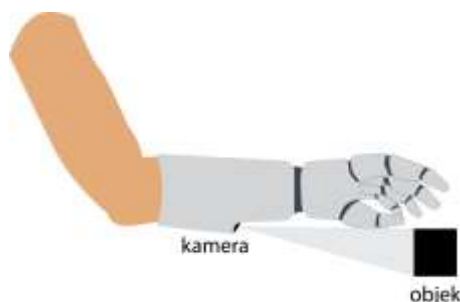
A. Kerangka Berpikir

Pada prinsipnya, *object detection* yang hendak diterapkan memberikan keputusan untuk melakukan *HGR* yang sering

digunakan dalam kegiatan-kegiatan dasar. Dari sisi medis, terdapat lima gerakan dasar jari yang menjadi fokus pengembangan presisi gerakan protesis lengan dan tangan, yakni: (1) gerakan mendekatkan jari-jari ke telapak tangan (fleksi), (2) gerakan meluruskan jari (ekstensi), (3) gerakan mendekatkan jari-jari ke jari tengah (adduksi), (4) gerakan menjauhkan jari-jari dari jari tengah (abduksi), dan (5) gerakan menjauhkan ibu jari dari jari lainnya (oposisi). Kelima gerakan dasar ini dikombinasikan menjadi tiga jenis HGR sebagaimana ditunjukkan pada Gambar 2: , antara lain: 1) gerakan mencubit (*pinch*), 2) gerakan memegang pensil (*pick*), dan 3) gerakan mengambil dengan seluruh telapak tangan (*grab*) [18]. Dalam mendeteksi objek nantinya, protesis dibubuhkan dengan kamera dengan rangkaian divisualisasikan oleh Gambar 3: sehingga secara langsung dapat menentukan jenis HGR berdasarkan bentuk dari objek.



Gambar 2: Mekanisme model *object detection* berbasis HGR



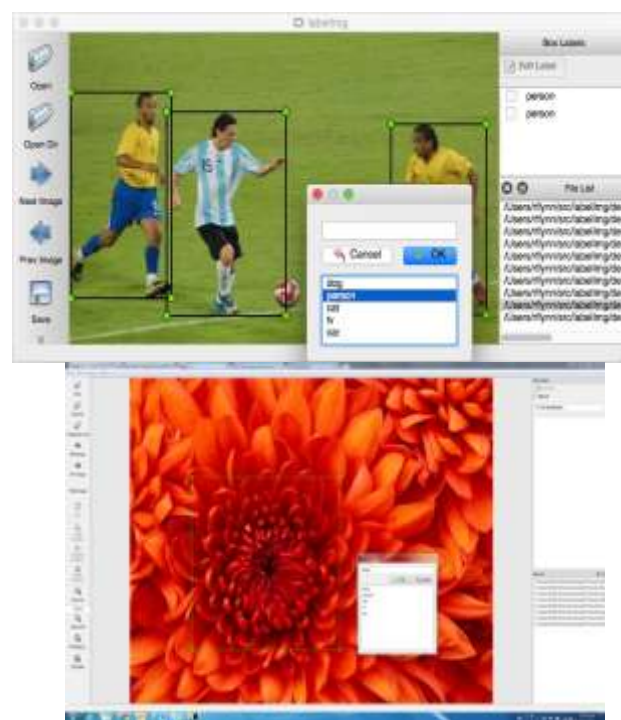
Gambar 3: Implementasi computer vision pada protesis tangan

B. Pengumpulan dan Processing Dataset

Pengumpulan *dataset* dilakukan dengan menggunakan metode *data scrapping* berupa gambar (*image*) yang diperoleh

I.M.P. Arya Winata: Model *Object Detection Neural*...

dari data publik. Hal ini dilakukan untuk memberikan gambaran umum pada objek yang hendak ditargetkan pada saat implementasi sehingga menyerupai keadaan sebenarnya saat *deployment*. Secara lebih spesifik, *image data scrapping* dilakukan dengan menggunakan modul *Selenium* pada *Python*, yang memberikan otomatisasi untuk mengunduh gambar publik pada *Google Image* serta mengumpulkannya sebagai *dataset* [19]. *Dataset* yang telah terkumpul selanjutnya dilakukan *labelling* berupa *bounding box* sebagai *object localization* pada gambar serta identifikasi objek sebagai tiga jenis HGR yang diharapkan pada lengan protesis. Keseluruhan proses *labelling* dilakukan dengan menggunakan *open-source framework LabelImg* yang dikembangkan secara spesifik untuk membantu memberikan *annotation* pada *dataset*, berupa *labels file* (Gambar 4:) [20]. *Dataset* dan *labels file* dilakukan proses *splitting* sebagai usaha untuk mengevaluasi (*testing*) model yang diajukan nanti pada data yang belum digunakan *training* oleh model. Rasio antara data *training* dengan *testing* yang digunakan yaitu 80:20. Secara kuantitas, jumlah *dataset* yang digunakan adalah total 317 gambar dengan 2278 label objek untuk ketiga jenis label.



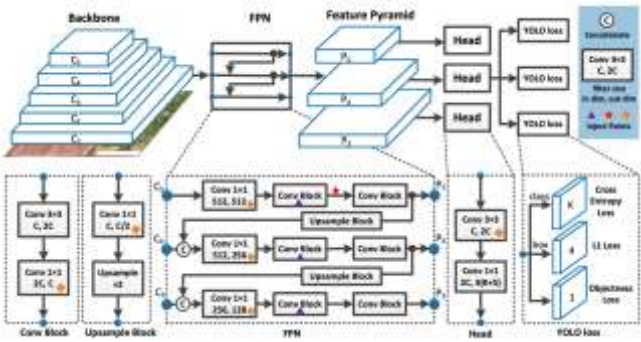
Gambar 4: Ilustrasi demo dari proses persiapan dataset pada *framework LabelImg* [11]

C. Penyusunan Model Neural Network Object Detection Berbasis HGR

Dalam membangun model *neural network object detection*, *pretrained model YOLOV7* digunakan sebagai basis dalam *transfer learning* untuk *dataset* yang telah dikumpulkan. *YOLOV7* menggunakan algoritma *convolutional* pada arsitekturnya sehingga memudahkan dalam *feature extraction*. Arsitektur model *YOLO* dapat dilihat pada Gambar 5:

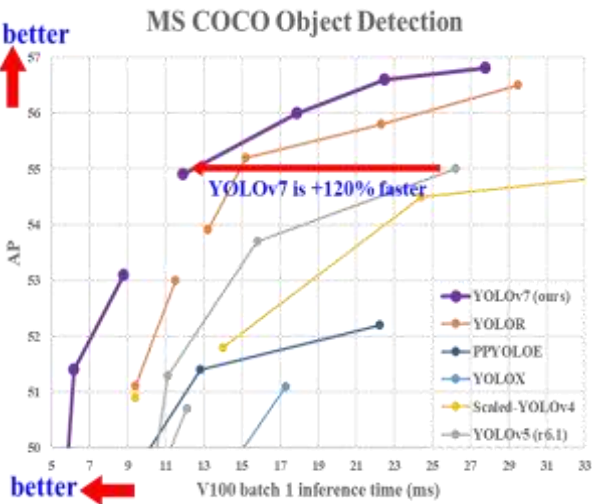
p-ISSN:1693 – 2951; e-ISSN: 2503-2372





Gambar 5: Arsitektur model YOLO [15]

Selain itu, *YOLOV7* merupakan model yang termasuk *one-stage object detector* yang mampu memberikan secara langsung nilai posisi objek dan klasifikasi objek secara langsung pada data gambar. Merujuk pada keunggulan ini, pengembangan model *neural network* untuk prostesis lengan ini dilakukan dengan menggunakan basis *YOLOV7* beserta turunannya [14]. Pemilihan dari *YOLOV7* beserta turunannya turut dipertimbangkan mengingat proses dari training dataset yang mengkonsumsi waktu relatif besar serta menghemat dari proses komputasi yang dilakukan. Hal tersebut dikaitkan dengan beberapa aspek, antara lain: *average precision (AP)* serta parameter yang dapat di-training (*trainable parameters*) pada tiap jenis model turunan *YOLOV7*. Selain itu, terdapat pula aspek lainnya seperti waktu inferensi model yang secara langsung dipengaruhi oleh jumlah *trainable parameter*. Gambar 6: menunjukkan grafik hubungan kedua aspek tersebut komparasi dari model turunan *YOLOV7* dan model lainnya.



Gambar 6: Perbandingan YOLOV7 dengan model *object detection* lain yang memperoleh *state-of-the-arts* pada aspek performansi

D. Evaluasi Model Neural Network Object Detection

Sebagai bahan evaluasi, perbandingan *evaluation metrics* pada beberapa turunan *YOLOV7* turut dilakukan untuk mengobservasi model yang sesuai dengan *task* yang diberikan. Beberapa *evaluation metrics* yang dimaksud berupa *precision*, *recall*, dan *mean average precision (mAP)*. Secara konsep,

precision dan *recall* dihasilkan melalui *cross-validation actual* dan *predicted value* yang ditunjukkan pada Tabel 1:

TABEL I
CROSS-VALIDATION NILAI SEBENARNYA (ACTUAL VALUE) DAN NILAI PREDIKSI (PREDICTED VALUE)

		Actual Value	
		True	False
Predicted Value	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

Precision dan *recall* selanjutnya dirumuskan dalam (1) dan (2) secara berturut-turut. Sementara itu, *mAP* merupakan hasil dari analisis rata-rata yang dihasilkan dari hubungan *precision-recall* akan digunakan sebagai akurasi model keseluruhan [22].

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots (1)$$

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

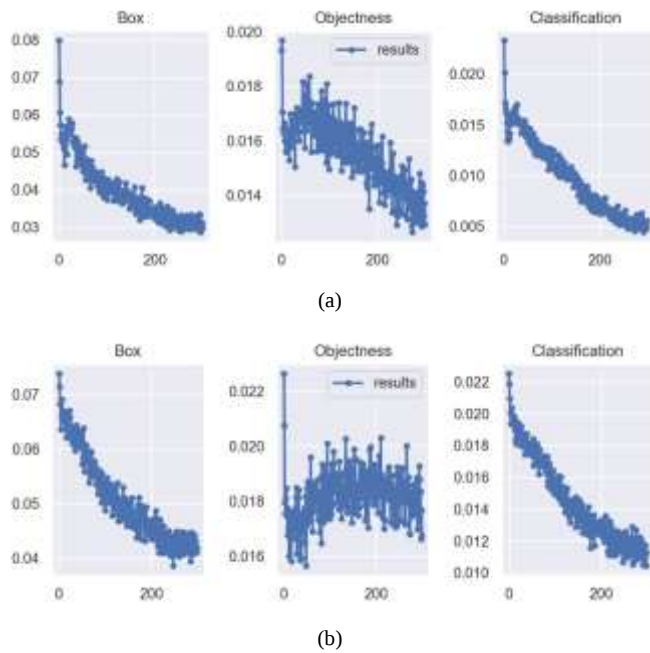
IV. HASIL DAN PEMBAHASAN

Model *neural network* dilakukan *training* dengan konfigurasi *set-up* komputer sebagaimana ditunjukkan Tabel 2:

TABEL III
KONFIGURASI TRAINING MODEL OBJECT DETECTION PROSTESIS TANGAN BERBASIS HGR

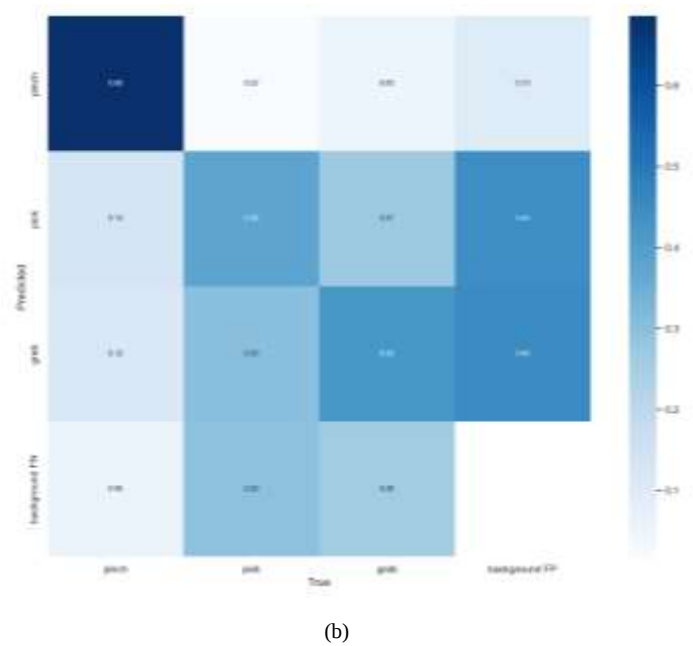
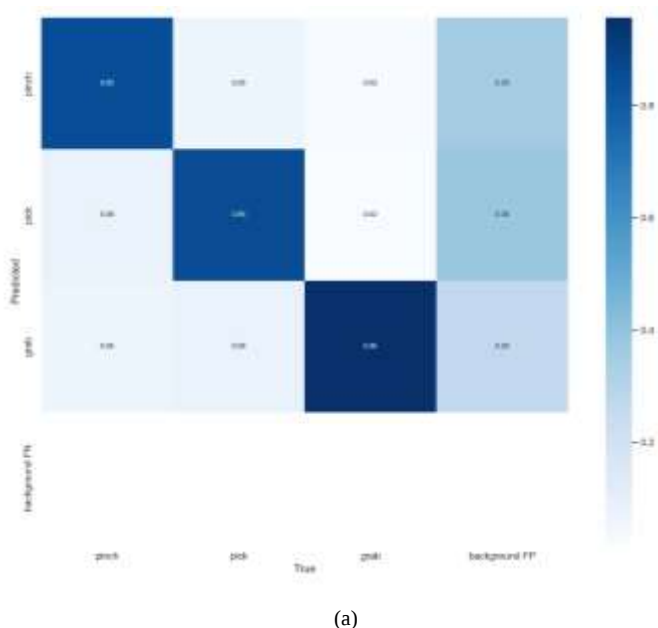
Parameter	Configuration
CPU	10 th Gen Intel® Core i5-10400F
GPU	NVIDIA GeForce GTX1650 Super (4G)
Operating System	Windows10
CUDA	11.7
Python	3.8.9
Torch	1.13.1
Momentum	0.937
Weight decay	0.0005
Batch size	4
Learning rate	0.01
Image size	640 x 640
Epochs	300

Training ini berlangsung selama 6 jam dengan masing-masing model *YOLOV7* dan *YOLOV7tiny* berdurasi 3 jam. Prosedur *training* dengan *set up* yang ditunjukkan pada Tabel 2: menghasilkan grafik *loss function* tiap epoch-nya yang menunjukkan bahwa model *trainable* pada *dataset* yang digunakan (Gambar 7:).



Gambar 7: Grafik nilai loss model tiap epoch training pada (a) YOLOV7 dan (b) YOLOV7tiny

Untuk memperoleh evaluasi yang komprehensif dan objektif pada performa dari model yang dibuat, prosedur dilanjutkan dengan menggunakan *confussion matrix*. Sebagai luaran evaluasi, *confussion matrix model* hasil training pada *testing dataset* ditunjukkan pada Gambar 8:



Gambar 8: *Confusion matrix* hasil training model (a) YOLOV7 dan (b) YOLOV7tiny pada dataset HGR

Melalui *framework Pytorch*, hasil *confussion matrix* tersebut dapat dianalisis *evaluation metrics*-nya yang secara detail ditunjukkan pada Tabel 3:

TABEL IIIII
NILAI *EVALUATION METRICS* PADA *TESTING DATASET*

Model <i>Pre-trained</i>	Jumlah Parameter (juta)	Frame per second (FPS)	mAP (%)
YOLOV7	36,9	161	98,11
YOLOV7-tiny	6,2	286	72,43

Tabel 3: menunjukkan bahwa model YOLOV7 menghasilkan nilai yang lebih baik pada *mAP* relatif terhadap YOLOV7tiny. Hal ini disebabkan karena model *pretrained* YOLOV7 memiliki lebih banyak *trainable parameter*, sejumlah 36,9 juta parameter, sementara YOLOV7tiny hanya memiliki 6,2 juta *trainable parameter* [14]. Tentu dengan adanya perbedaan jumlah parameter ini membuat YOLOV7tiny membuat kesalahan interpretasi yang lebih banyak. Namun, pengembangan YOLOV7tiny tetap harus dipertimbangkan apabila *deployment model* dilakukan karena model yang lebih sedikit parameternya akan meningkatkan kecepatan proses gambar, dengan kata lain FPS (*frame per second*) dari proses gambar semakin meningkat. Hal ini tentu berdampak baik untuk kondisi *real-time* [21]. Selain itu, berdasarkan kelayakan implementasi, nilai dari *mAP* YOLOV7 dapat dikatakan telah memenuhi kategori sangat baik (*excellent*) karena memiliki nilai *mAP* di atas 90%. Sementara itu, YOLOV7tiny, dengan



ambang batas 80%, belum memenuhi sebagai model yang optimal untuk diimplementasikan [22]. Sehingga, YOLOV7 akan lebih baik dan layak saat implementasi model *object detection neural network* pada prostesis tangan.

Peninjauan terhadap masing-masing jenis label juga dapat diamati melalui Tabel 3:. Label yang selalu memiliki nilai *metrics* yang paling rendah adalah label *pinch*. Hal ini dikarenakan model ini masih memiliki jumlah label yang sedikit dan secara pengamatan kasat mata memiliki bentuk yang hampir sejenis dengan objek pada label *pinch*, berbentuk panjang dengan lebar relatif kecil (Gambar 9:). Peningkatan akurasi model berpotensi dapat dikembangkan dengan metode *segmentation* yang memberikan analisis lebih mendetail terhadap bentuk objek [23].

Terkait aspek dataset, pengembangan model juga dapat diperkuat akurasi dengan penambahan pada variasi dataset yang mampu merepresentasikan objek sesuai dengan konsep HGR. Pada dataset yang digunakan saat ini dapat dikatakan masih realtif kurang dalam distribusi jenis objek karena pada saat pengumpulan dataset proses *web scrapping*, objek yang diperoleh terbatas pada keyword yang diberikan.



Gambar 9. Hasil prediksi model *object detection* pada *batch testing dataset*

V. KESIMPULAN

Dengan hasil yang dicapai, pengembangan model *object detection neural network* yang telah dilakukan menunjukkan bahwa pendekatan ini sangat memungkinkan untuk

diimplementasikan, secara khususnya dengan model YOLOV7 dan turunannya. Hal ini secara langsung ditunjukkan oleh nilai *evaluation metrics* yang dihasilkan pada *dataset* objek yang selanjutnya memiliki relasi dengan HGR yang bersesuaian. Sebagai tahapan pengembangan lanjutan, pengumpulan *dataset* yang lebih banyak perlu dilakukan agar mampu merepresentasikan keadaan sebagaimana saat di-deploy serta implementasi metode *segmentation* untuk mencapai ketelitian deteksi dan klasifikasi objek.

UCAPAN TERIMA KASIH

Ucapan terima kasih diberikan kepada Universitas Udayana melalui Lembaga Penelitian dan Pengabdian kepada Masyarakat (LPPM) atas pemberian hibah pada kegiatan Merdeka Belajar Kampus Merdeka (MBKM) Riset dengan Nomor Keputusan 921/UN14/HK/2022 serta Mitra MBKM Riset Tim Prostesis Tangan Puspadi Bali atas bantuan dan arahan terhadap pengembangan prostesis tangan dari perspektif praktisi. Sehingga, penelitian ini dapat berjalan dengan semestinya.

REFERENSI

- [1] I. A. Satam, "Review Studying of the Latest Development of Prosthetic Limbs Technologies," *Int. J. Sci. Eng. Res.*, vol. 12, no. 12, pp. 721–731, 2022.
- [2] P. S. Mckechnie and A. John, "Anxiety and depression following traumatic limb amputation : A systematic review," *Injury*, vol. 45, no. 12, pp. 1859–1866, 2014, doi: 10.1016/j.injury.2014.09.015.
- [3] Kemenkes RI, "Hasil Riset Kesehatan Dasar Tahun 2018," *Kementrian Kesehat. RI*, vol. 53, no. 9, pp. 1689–1699, 2018.
- [4] D. Dosen, S., Prahm, C., Amsüss, S., Vujaklija, I., & Farina, "Prosthetic Feedback Systems," in *Bionic Limb Reconstruction*, Springer Nature Switzerland AG, 2021, pp. 147–170.
- [5] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust Part-Based Hand Gesture Recognition Using Kinect Sensor," *IEEE Trans. Multimed.*, vol. 15, no. 5, pp. 1–11, 2013.
- [6] B. Feng, F. He, X. Wang, Y. Wu, and H. Wang, "Depth-Projection-Map-Based Bag of Contour Fragments for Robust Hand Gesture Recognition," *IEEE Trans. Hum.-Mach. Syst.*, vol. 47, pp. 1–13, 2016.
- [7] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition : A review," *Comput. Vis. Image Underst.*, vol. 141, pp. 152–165, 2015, doi: 10.1016/j.cviu.2015.08.004.
- [8] M. Garcia, J. Cruz, C. Garza, P. DeLucia, and J. Yang, "The Effect of Object Surfaces and Shapes on Hand Grip Function for Heavy Objects," in *Proceedings of the AHFE 2018 International Conferences on Human Factors and Simulation and Digital Human Modeling and Applied Optimization*, 2018, pp. 446–452.
- [9] I. P. G. S. Andisana, M. Sudarma, and I. M. O. Widyantara, "Pengenalan Dan Klasifikasi Citra Tekstil Tradisional Berbasis Web Menggunakan Deteksi Tepi Canny, Local Color Histogram Dan Co-Occurrence Matrix," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 3, p. 401, 2018, doi: 10.24843/mite.2018.v17i03.p15.
- [10] X. Zhang, M. Shen, X. Li, and F. Feng, "A deformable CNN-based triplet model for fine-grained sketch-based image retrieval," *Pattern Recognit.*, vol. 125, p. 108508, 2022, doi: 10.1016/j.patcog.2021.108508.
- [11] N. Dong, Y. Zhang, M. Ding, and G. H. Lee, "Incremental-DETR: Incremental Few-Shot Object Detection via Self-Supervised Learning," 2022, [Online]. Available: <http://arxiv.org/abs/2205.04042>.
- [12] M. Z. Alom et al., "The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches," 2018, [Online]. Available: <http://arxiv.org/abs/1803.01164>.
- [13] P. Bharati and A. Pramanik, "Deep Learning Techniques—R-CNN to Mask R-CNN: A Survey," in *Advances in Intelligent Systems and Computing*, 2020, vol. 999, p. 7, doi: 10.1142/S0218001402001976.
- [14] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, [Online]. Available:

- <http://arxiv.org/abs/2207.02696>.
- [15] J. Solawetz, "YOLOv7 - A Breakdown of How it Works," *Roboflow*, 2022. <https://blog.roboflow.com/yolov7-breakdown/> (accessed Jan. 10, 2023).
 - [16] S. Mittal, "A Survey on optimized implementation of deep learning models on the NVIDIA Jetson platform," *J. Syst. Archit.*, vol. 97, no. December 2018, pp. 428–442, 2019, doi: 10.1016/j.sysarc.2019.01.011.
 - [17] C. G. I. Raditya, P. A. S. Dharma, K. A. Widyatmika, I. N. Suparta, I. M. S. Yasa, and A. A. N. G. Sapteka, "Pendeteksi Penggunaan Masker Wajah dengan ESP32Cam Menggunakan OpenCV dan Tensorflow," *Maj. Ilm. Teknol. Elektro*, vol. 21, no. 2, p. 155, 2022, doi: 10.24843/mite.2022.v21i02.p01.
 - [18] A. Schafer, G. Reis, and D. Stricker, "Comparing Controller With the Hand Gestures Pinch and Grab for Picking Up and Placing Virtual Objects," in *IEEE VR*, 2022, pp. 1–2, doi: 10.3390/mti4040091.
 - [19] K. U. Manjari, S. Rousha, D. Sumanth, and J. Sirisha Devi, "Extractive Text Summarization from Web pages using Selenium and TF-IDF algorithm," *Proc. 4th Int. Conf. Trends Electron. Informatics, ICOEI 2020*, pp. 648–652, 2020, doi: 10.1109/ICOEI48184.2020.9142938.
 - [20] Tzatalin, "LabelImg," *Git code*, 2015. <https://github.com/heartexlabs/labelimg> (accessed Jan. 10, 2023).
 - [21] J. Zheng, H. Wu, H. Zhang, Z. Wang, and W. Xu, "Insulator-Defect Detection Algorithm Based on Improved YOLOv7," *Sensors*, vol. 22, no. 22, pp. 1–23, 2022, doi: 10.3390/s22228801.
 - [22] K. M. Kuo, P. C. Talley, C. H. Huang, and L. C. Cheng, "Predicting hospital-acquired pneumonia among schizophrenic patients: A machine learning approach," *BMC Med. Inform. Decis. Mak.*, vol. 19, no. 1, pp. 1–8, 2019, doi: 10.1186/s12911-019-0792-1.
 - [23] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep Learning Based Text Classification: A Comprehensive Review," *ACM Comput. Surv.*, vol. 54, no. 3, p. Article 62, 2021, [Online]. Available: <http://arxiv.org/abs/2004.03705>.



{Halaman ini sengaja dikosongkan}