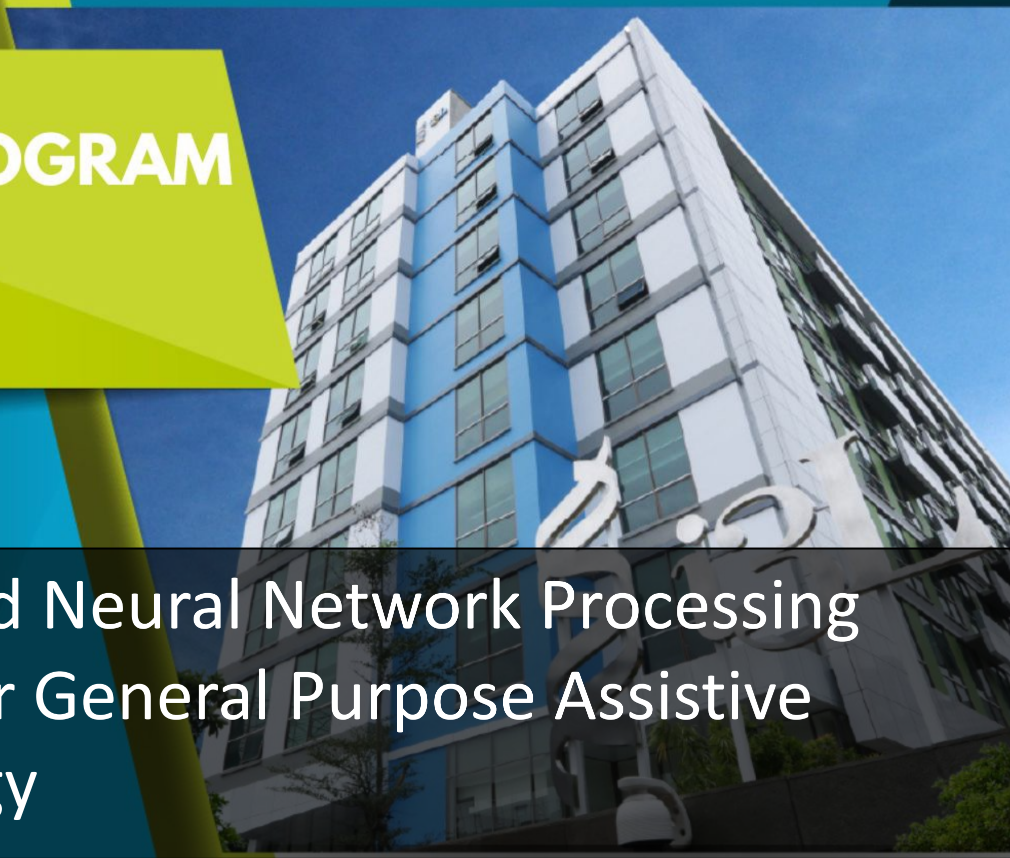
ENRICHMENT PROGRAM REPORT



Embedded Neural Network Processing System for General Purpose Assistive Technology

STUDY PROGRAM
Bioinformatics

JEREMIE THEDDY DARMAWAN
19010195

Xanno Kharis Sigalingging, S.Si., S.T., M.Sc.
Nanda Rizqia Pradana Ratnasari, S.Si, M.Stat.

INDONESIA INTERNATIONAL INSTITUTE FOR LIFE SCIENCES (i3L)
YEAR OF SUBMISSION

INTERNSHIP REPORT

Embedded Neural Network Processing System for General Purpose Assistive Technology

By
Jeremie Theddy Darmawan
19010195

Submitted to

i3L – Indonesia International Institute for Life Sciences
School of Life Sciences

in partial fulfilment of the enrichment program for the Bachelor of Science in
Bioinformatics

Internship Project Supervisor: Nanda Rizqia Pradana Ratnasari, S.Si, M.Stat.
Internship Project Field Supervisor: Xanno Kharis Sigalingging, S.Si., S.T., M.Sc.

Taipei, Taiwan
2022

INSTITUT BIO SCIENTIA INTERNASIONAL INDONESIA

Jl. Pulomas Barat Kav. 88 Jakarta Timur 13210 Indonesia
+6221 295 67888, +6221 295 67899, +6221 296 17296
www.i3l.ac.id

Certificate of Approval

Student	: Jeremie Theddy Darmawan
Cohort	: 2019
Title of final thesis project	: Embedded Neural Network Processing System for General Purpose Assistive Technology *Embedded Neural Network Processing System for General Purpose Assistive Technology*

We hereby declare that this final thesis project is from student's own work. The final project/thesis has been read and presented to i3L's Examination Committee. The final project/thesis has been found to be satisfactory and accepted as part of the requirements needed to obtain an i3L bachelor's degree.

Names and signature of examination committee members present:

1	Thesis Supervisor	: Nanda R.P.R. S.Si., M.Sc.	Approved
2	Field Supervisor	: Xanno	Approved
3	Lead Assessor	: Muammar S. , B.S., M.S., Ph.D.	Approved
4	Assessor 2	: Puspa S.P. ,S.Kom., IMSME, Ph.D.	Approved

Acknowledged by,

Head of Study Program,

David Agustriawan, S.Kom., M.Sc., Ph.D.

This is a form-based authentication form, gaining access to this form is a method of signer validation, therefore, this form does not require a signature.
Scan the QR code to verify the document validity.

COPYRIGHT NOTICE

STATEMENT OF ORIGINALITY

I, Jeremie Theddy Darmawan, hereby declare that the material contained in my thesis titled "**Embedded Neural Network Processing System for General Purpose Assistive Technology**" is my original work, completed under the guidance and advice of my Thesis Advisors, Nanda Rizqia Pradana Ratnasari, S.Si, M.Stat. and Xanno Kharis Sigalingging, S.Si., S.T., M.Sc. I have read and understand the definition and guidelines for proper use of sources and citation style published by the Indonesia International Institute for Life Sciences (i3L). By signing this statement, I confirm that my thesis conforms to these guidelines.

I give i3L permission to submit an electronic copy of my thesis to a commercial document screening service, including my name. If you check "No," my name will be removed before the document is submitted for screening.

☒ Yes
☐ No

Name of student: Jeremie Theddy Darmawan
Student ID: 19010195
Study Program: Bioinformatics

Signature: Date: 22 December 2022

ABSTRACT

Assistive technology (AT) is invaluable to people with special educational needs and disabilities, enabling them to function more efficiently. Designed with low power consumption and a GPU capable of running AI frameworks, NVIDIA's Jetson Nano is an affordable embedded system device. With this system, users will be mobile and connected to any computer with an appropriate interface, and able to interact with that computer. Furthermore, Jetson Nano has modularity, which could solve other issues related to AT in the future. For this study, the FCOS and TOOD models were used for the cursor object detection along with the SAHI algorithm. This study combines a Speech-to-Intent module and a Mouse Cursor Detection module to move the computer mouse using speech commands. A USB Gadget API can be developed in the future in order to extend the assistive technology device to provide assistance to other USB-equipped devices using the Linux operating system in the future, as well as using powerful embedded devices, like the Jetson Nano or Xavier, that are able to combine additional deep learning models to enhance their functionality.

Keywords: Data science applications in education; Improving classroom teaching; Teaching/learning strategies; 21st century abilities; Architectures for educational technology system

ACKNOWLEDGEMENTS

My first and foremost thanks to Almighty God for His showers of blessings throughout my research that enabled me to complete the project successfully.

I would like to express my deep and sincere gratitude to Dr. Jenq-Shiou Leu, B.Sc., M.Sc., Ph.D., SM IEEE, Professor and Head, Mobilizing Information Technology Laboratory, National Taiwan University of Science and Technology, Taipei, for giving me the opportunity to do a research internship and providing invaluable support and guidance throughout this program. I am also thankful to my research supervisor for this research, Xanno Kharis Sigalingging, S.Si., S.T., M.Sc., who provided practical guidance in conducting and writing the research. I am extremely grateful for what they both offered me.

I am extremely grateful to my parents for their love, prayers, support, caring, and sacrifices for educating and preparing me for my future. In addition, I would like to thank my sister for her support and prayers.

I would like to say thanks to my i3L supervisor, Nanda Rizqia Pradana Ratnasari, S.Si., M.Stat., for her guidance in conducting and creating the research manuscripts as well as liaison with the other departments in i3L to ensure a smooth process.

I am extending my thanks to the members at MIT Lab that have given support throughout this internship period for this project and many others.

Finally, my thanks are extended to all the people who have supported me to complete the research work directly or indirectly.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

***AT**: Assistive Technology*

***ASR**: Automatic Speech Recognition*

***FCOS**: Fully Convolutional One-Stage*

***TOOD**: Task-Aligned One-Stage*

***VFNet**: Varifocal Net*

***mAP**: mean Average Precision*

***ARM**: Advanced RISC Machine*

***SAHI**: Slicing Aided Hyper Inference*

I. INTRODUCTION

As the National Taiwan Institute of Technology (NTIT), the National Taiwan University of Science and Technology was founded on August 1, 1974. Among the technical and vocational education systems of Taiwan, it was the first institution of higher education of its kind. Taiwan's rapid economic and industrial development created a need for highly trained engineers and managers at the bachelor's degree level, so we extended this system to the highest level. Master's and doctoral degrees were added to NTIT in 1979 and 1982, respectively. NTIT became a university on August 1, 1997, and its name was changed to "National Taiwan University of Science and Technology", also known as "Taiwan Tech." The school reorganized itself into five colleges at that time. Now, Taiwan Tech currently has six colleges, including engineering, electrical and computer engineering, management, design, liberal arts & social sciences, and applied science.

The university’s vision as described on its website: “To be the leading industrial engineering and management department, possessing excellent record as an outstanding research and educational institution in Taiwan and Asia.” With the mission of: “To teach our students about applied science and technology as well as to conduct researches in IE&M, and to train them managerial capability required for enterprise development and become the person in charge on it in the future.”

The organizational structure of the university is led by the President followed by Vice-President 1 and Vice-President 2, Secretary, Treasurer 1, and Treasurer 2. There are six colleges, including Engineering, Electrical and Computer Engineering, Management, Design, Liberal Arts & Social Sciences, and Applied Science. Each of the six colleges has its own Dean and a set of departments. This program was conducted within the College of Electrical Engineering and Computer Science with the Department of Electronic and Computer Engineering. This department’s research focus is on realizing advanced technologies that will be applied to the industry through the strong cooperation between academia and industry that will strike a balance between theory and practice. The Dean of this college is Professor Jiann-Liang Chen, Ph.D., and the professor that the student has undergone the program was Professor Jenq-Shiou Leu. During the internship program, the student was placed in the Mobilizing Information Technology (MIT) Laboratory. The research focus of this laboratory is on the interdisciplinary application of Deep Learning.

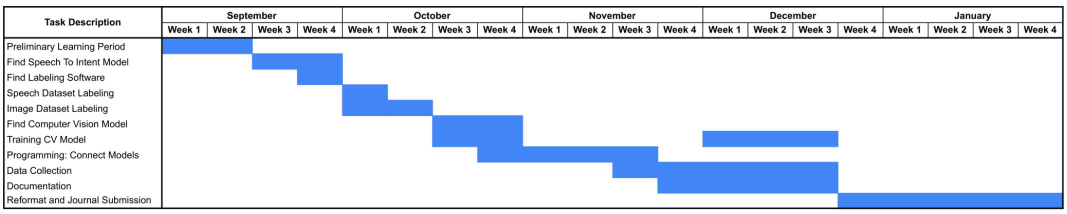
II. INTERNSHIP ACTIVITIES

The internship program was run for five months starting from the 1st of September until the end of January 2023. There are pre-defined working hours and the laboratory is open 24 hours. Students may come at any time and leave at any time. Remote work is also allowed. However, since the Professor's expectation of the laboratory is goal-oriented there should be progress on the work done every week.

During the internship, the task given to the student was not defined at the beginning of the internship program. It was formulated after a settling-in period where the student was encouraged to learn the techniques and skills that will be used during the task. The task was defined as research on Neural Networks and their application as a General-Purpose Assistive Technology which was then titled "Embedded Neural Network Processing System for General Purpose Assistive Technology." As how research should be conducted, the student undergone literature review, problem formulation, method formulation, experimentation, and manuscript or report writing in no particular order. During the research, the student was required to conduct the process by themselves and queries should be attempted to be solved by themselves at first and further inability to solve would then be guided by the field supervisor. Initially, this research was conducted without thought of publication. However, once the research was completed, the results were sufficient for an attempt at publication. The internship timeline that lasted for 5 months, from September 2022 to January 2023, can be referred to in Table 1. Among the tasks that was assigned, it could be outlined with the following descriptions: preliminary learning period, finding speech-to-intent model, finding labelling software, labelling of both speech dataset and image dataset, finding the working computer vision model, training the CV model, programming to combine all the models together, and data collection. Finally, a period of documentation for reporting and further reformatting for journal submission was also present at the end of the timeline. While some may be done simultaneously, most of the tasks was done in sequential order as seen on Table 1.

Table 1

Internship Project Timeline



The closest theory that was provided in i3L on that was relevant to the research task was regarding programming and machine learning. This was taught in the fourth semester as Programming in Python and Discrete Mathematics in the third semester, respectively. However, this was not sufficient to conduct the research as deep learning can be taught as a more advanced version of machine learning that is based on artificial neural networks. Deep

Learning is made up of layered structured algorithms which require a larger amount of data than machine learning but require little human intervention for problems like feature selection.

Due to the unencountered topic of deep learning during university, there was a need to learn the basics of deep learning model creation, and optimization, among others. The lab members in the laboratory, the Ph.D. students under the supervision of Professor Leu, advised YouTube videos and research papers to read on to understand the basics of deep learning. GitHub Repositories and practice during the preparatory period helped apply what was little known and translating it into experience relatively quickly.

III. PROJECT DESCRIPTION

A. Introduction

Technology has become a crucial part of education, communication, as well as interpersonal human interaction. The utilization of technological devices is a cornerstone of many aspects of quality life ((Boni) Li & Perkins, 2007). Assistive technology (AT) has an important role in facilitating persons with special educational needs or disabilities to overcome struggles in daily life when interacting with technological devices, such as computers and mobile devices (Borg & Östergren, 2020). Usage of AT could significantly improve the health condition and well-being of persons with a disability whereas in a more specific life aspect namely education, it could provide improved academic engagement and social participation (McNicholl et al., 2021; Shi et al., 2022). Hence, reducing stigma and facilitating inclusion for those affected people. The need for specialized caregivers and long-term care could also be reduced as an outcome of AT implementation.

A personal assistant that can act as an intermediary between the computer device and human is proposed to solve the aforementioned problem. As communication among human beings is primarily done using spoken language rather than typing on a keyboard, it is a natural expectation to have a voice interface with a computer for interaction (Das et al., 2015). This can be achieved using an automatic speech recognition (ASR) system, such as speech-to-text, that converts spoken commands into text. Such a system can be implemented by using an acoustic signal input and deep learning algorithm to map the signals into words. Further computer functions can be called upon from the transcribed words.

The Jetson Nano is an affordable embedded system device from NVIDIA that has a small form factor with low power consumption as well as being compatible with various AI frameworks (Valladares et al., 2021). It has a 1.43 GHz ARM architecture CPU with a 128-core NVIDIA Maxwell GPU and 4GB DDR4 RAM (Tolmacheva et al., 2020). Using an embedded device such as the Jetson Nano enables computing on the edge without compromising too much on computational capabilities. Deep learning algorithms can be developed through the likes of TensorFlow, PyTorch, Caffe, Torch, and Keras open-source frameworks (Wu et al., 2022). Both of which supported by the Jetson device through the TensorRT optimization technology for accelerating edge computing (Shin & Kim, 2022).

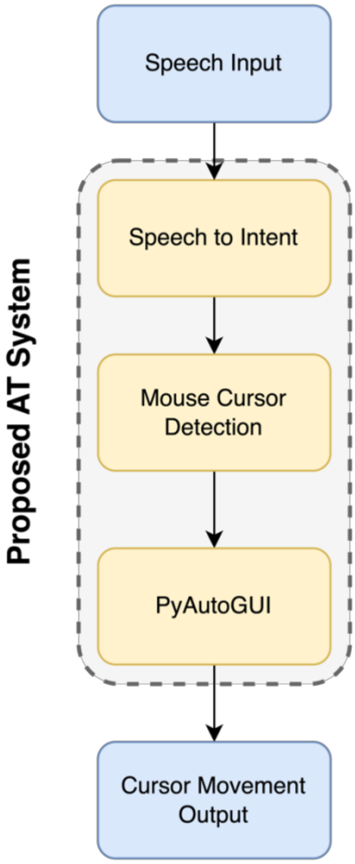
By implementing the AT to control the mouse cursor on a computer, persons with a physical disability can effectively interact with the computer device using only their voice. A small form factor is important to have in order to enable interactive usage, mobility, and allow them to use any computer device with the necessary interface for connection. The Jetson Nano is a perfect platform that provides both portability and the necessary computational capacity to execute the technology with virtually no latency issues (Shin & Kim, 2022). The proposed system offers a general-purpose AT device that could provide affordable access and feature upgradability to the technology using a small embedded device and deep learning models for functionality. By adding more modules, this device can target more than individuals with physical disabilities and reach those with visual or hearing impairments. Despite the necessity of other modules to make the whole system function effectively, further work would largely depend on the cursor detection module to enable the system in providing its function to external computer systems. It works by identify the location of the cursor without actually having access to the main computer system. Without this module, the AT would not be able to identify the location of the cursor and further implications would arise in the cursor movements. Detecting the cursor is essential in this assistive device as computer devices based on the Linux, Windows and MacOS operating systems primarily use the cursor to execute user's commands. This method of controlling the computer device also requires the least amount of permission access in which the connected device could act as a Human Interface Device (HID) that is similar to a mouse. As the permission granted by the computer to this assistive device is minimal, the cursor detection module relies on object detection to identify the location of the cursor and estimate its position on the screen which otherwise would be impossible. Hence, this study focuses on the cursor detection module of the system where several feasible models are introduced and discussed.

B. Results and Discussions

a. Results

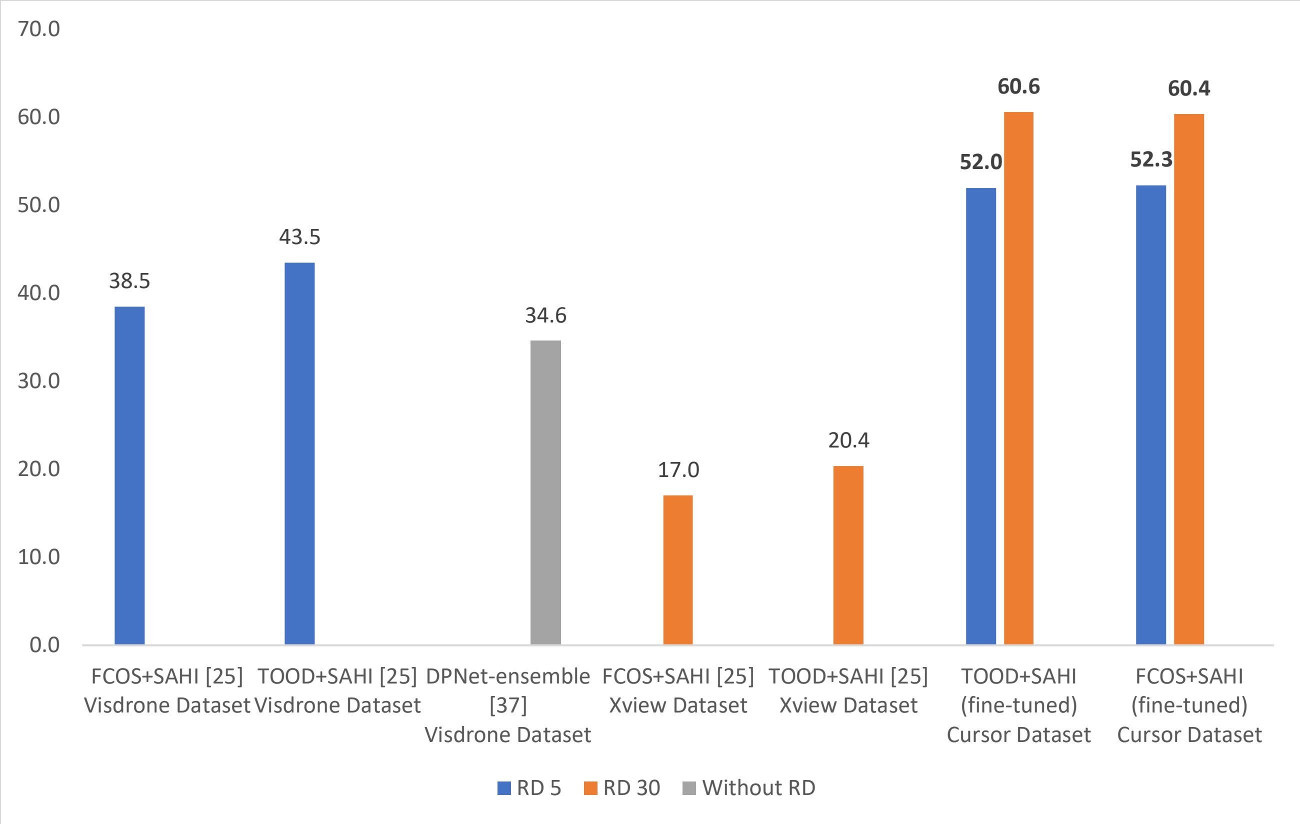
In order for the computer to do what is requested by the user, it needs commands that are understood by the machine while humans do not regularly communicate using machine language. Therefore, a mechanism to translate human language, in this case, spoken, to a language that machines are able to understand is necessary. In this project, the stages to implement the AT are described as beginning with the Speech-to-Intent module, Mouse Cursor Detection module, and finally patching it all together with code to move the computer mouse by command as shown in the schematic representation of the process in **Fig 1**.

Figure 1
Proposed AT System



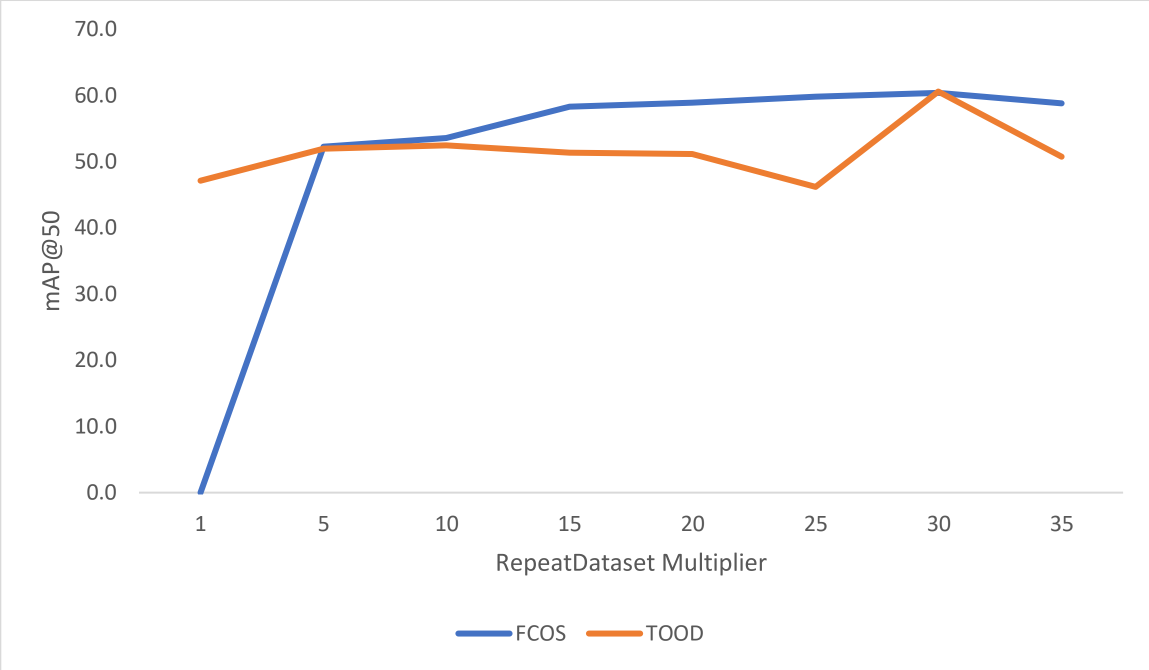
Since the metric mAP requires a threshold for the determination of correct and incorrect detection there can be various mAP values (Padilla et al., 2020). This is generally denoted as mAP@*t*, where *t* refers to the threshold value. The value of *t* usually ranges from 50 to 95 in increments of 5. Based on the SAHI study, the mAP@50 for FCOS, VFNet, and TOOD enhanced with the SAHI pipeline were 38.5, 42.2, and 43.5, respectively (Akyon et al., 2022). As of November 19, 2022, the leader boards, found in http://aiskyeye.com/leaderboard/, for the small object detection challenge based on the visdrone dataset has a maximum mAP@50:5:95 of 40.35 using an Inception model which falls below the SAHI-enhanced VFNet and TOOD results (Zhu et al., 2021). **Fig. 2** below shows the mAP@50 comparison between the fine-tuned model against models that were used in the original SAHI publication on 5 and 30 RepeatDatasets. The fine-tuned models were able to perform better than the TOOD and FCOS that were trained on the visdrone and xview datasets. This is due to the usage of pre-trained weights and further fine-tuning of the network which causes an increase in performance as also observed in a previous study using fine-tuned models (Kim et al., 2022; Lalor et al., 2017).

Figure 2
Comparison of fine-tuned FCOS and TOOD cursor detection model to other FCOS and TOOD models



Based on the above results where both fine-tuned models have very similar performances, further investigation on the effect of dataset size on performance was explored. By using the parameter "RepeatDataset" which is available to artificially manipulate the size of the dataset, several models were trained with the full dataset using various values of this parameter to evaluate the effect on model performance. Generally, larger dataset sizes would provide better performance as the models are able to train from a larger collection of data (Shorten & Khoshgoftaar, 2019). To calculate the total amount of training data that was received by the model, the raw training data should be simply multiplied by the RepeatDataset multiplier. Hence, the full training data of 119 images and a multiplier of 5 and 25 would generate around 1190 and 2975 images for model training, respectively. The multiplier parameter that was tested was 1, 5, 10, 15, 20, 25, 30, and 35. Since more training data needs to be processed in each epoch, the training duration increases as the multiplier increases. For a multiplier of 3, the training time could be as little as 30 minutes while using the multiplier value 35 would increase the training time to over 3 hours with the full dataset.

Figure 3
Effect of RepeatDataset on fine-tuned models' mAP@50 performance

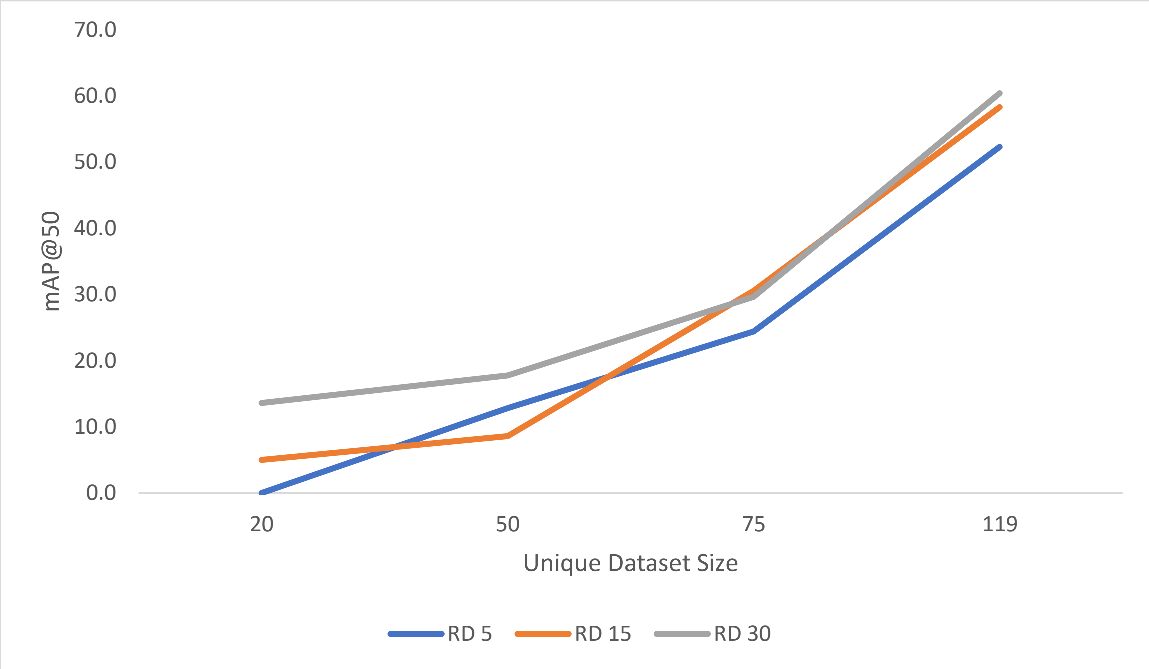


Despite having a small dataset, the usage of RepeatDataset which increases the dataset size does not significantly improve the performance of the models. In terms of mAP, both models experience a slight increase as the parameter escalates before slightly declining back as the parameter value approaches 25. **Fig. 3** shows the impact of increasing the RepeatDataset parameter and consequently duplicating the dataset, onto the mAP of the model. Increasing the parameter value increases the performance slightly where it starts to plateau above RepeatDataset 5 and 15 for the TOOD and FCOS models, respectively. This corresponds to 595 and 1785 image and annotation data. The FCOS model was not able to train effectively with too little data but it has a more stable performance as duplicate data increases. For the TOOD model, it did achieve higher peak performance on RepeatDataset 30 and was able to learn from fewer data however, its performance started to decline earlier, starting from RepeatDataset 5, and has more performance fluctuations on higher values of RepeatDataset.

At first glance, the model has reached its peak capability of detecting the objects with the plateauing. This can also be referred to as the saturation point of the model where supplying more data does not significantly result in better performance (Adadi, 2021). However, due to the nature of the RepeatDataset parameter that creates duplicates of the same image to increase the dataset, it actually gives a false sense of performance plateau. There are dangers when implementing this method excessively. Some of those dangers include introducing training bias and would negatively impact the generalization performance of the model on new unseen images (Manjusha & Suryanarayana, 2022). Some of these impacts can be observed on the larger values where the performance deteriorates. While the models experience slight improvements as the RepeatDatasets are increased, the impact is transient and does not persist as the parameter continues to be increased. From the results, it seems that duplicating data could produce improved results when used in moderation, but it has adverse effects as it becomes too large. The deterioration of model performance also seems to vary from one model to another with the TOOD model rather taking a larger drop than FCOS. Rather than increasing the value of this parameter, it might yield a better outcome to augment the dataset with new, unique data instead.

Figure 4

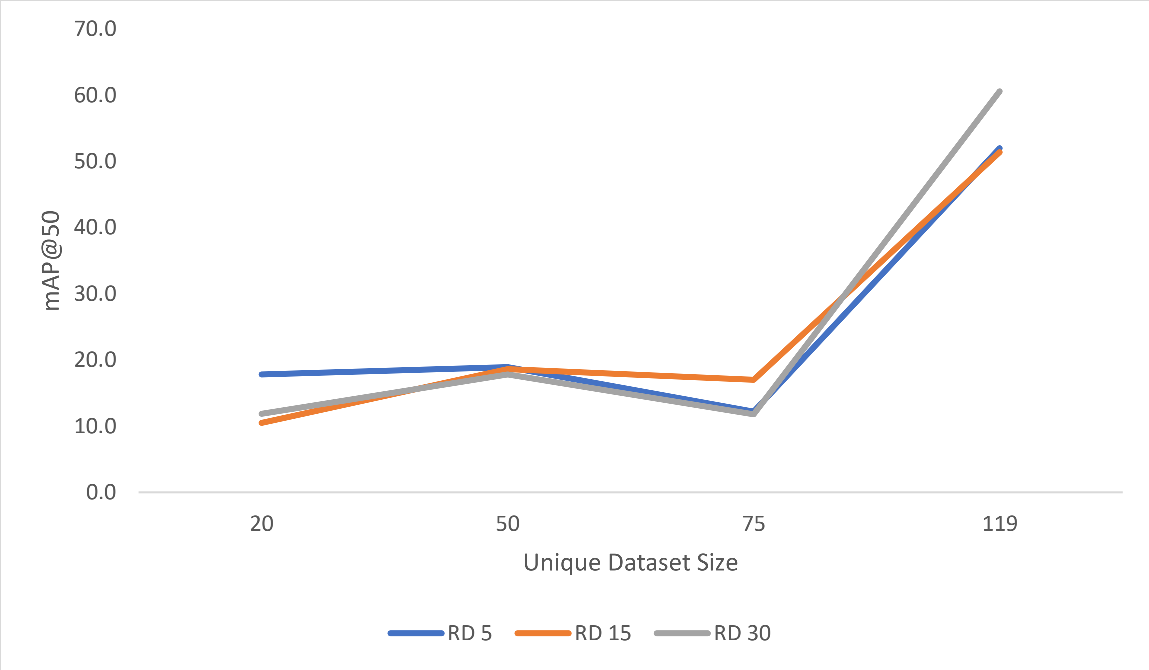
Effect of Unique Dataset Size and RepeatDataset Parameter on fine-tuned FCOS model's mAP@50 performance; RD: RepeatDataset



As seen in **Fig. 4**, the FCOS model with a dataset of 20 unique images and only 5 RepeatDatasets, caused the model to not be able to infer the validation images. It was also observed in **Fig. 3** where it fails to predict the validation images with the full dataset and 1 RepeatDataset. Specifically, the FCOS model seems to require more data than the TOOD model to produce results. Adding new and unique data into the dataset for training does improve the performance significantly while increasing the RepeatDataset parameter did not have a large impact on the mAP of the models. These findings are consistent with the previous finding and literature (Manjusha & Suryanarayana, 2022). We also explored the impact of unique data on the TOOD model. The unstableness of the TOOD model can be observed with the turbulent performances, as seen in both **Fig. 3** above and **Fig. 5** below. In **Fig. 5**, the performance dropped for all RepeatDataset values with the unique dataset containing 75 images. This may be something inherent to the model as adding new and duplicate images both showed an early decline in performance while rising back to its peak on more images. As mentioned in other research, higher learning rates are prone to larger covariant errors and make training unfeasible at times at the expense of fast convergence (Hsieh & Shanechi, 2018). Reducing the learning rate by another factor of 0.1 or more for training may produce more consistent results albeit resulting in lower performance for the same number of epochs.

Figure 5

Effect of Unique Dataset Size and RepeatDataset Parameter on fine-tuned TOOD model's mAP@50 performance; RD: RepeatDataset



With the same amount of data as the FCOS model, the TOOD model falls short on performance however it does achieve something that the FCOS model could not. It was able to train using less data, namely on the 20 image dataset with 5 RepeatDataset where the FCOS fails to produce results, this model was able to train and subsequently infer the validation image better than other TOOD models that were trained with larger RepeatDataset. When it comes to the impact of duplicate images, the performance of models trained on smaller unique datasets is penalized more with the introduction of more duplicate images. As the data increases, this performance penalty decreases. These models should ideally be trained with more new images, rather than relying on the RepeatDataset parameter, which will improve its performance tremendously.

b. Discussions

A proof-of-concept system to allow cursor movement by using speech-to-intent recognition and object detection to identify the cursor's coordinate was developed in this study. Commands that were trained on the Speech-to-Intent module can be categorized for the purpose of moving the cursor up, down, left, and right. The module was also trained to recognize the command click and tap which corresponds to a double and single mouse click. This module would recognize speeches from users and infer their intent based on the set of trained intents. A regular ASR would output a text of transcribed speech however since the module combines an intent classification module, the output instead is the intent of the transcribed speech. This intent could then be used for the algorithm to move the cursor.

The crucial module of this system lies in the cursor detection module. However in the current system, it is mainly acting as a foundation for future works and extension into application between the Jetson Nano and other devices, the computer vision used to detect the cursor presents a method to externally, without access to the device system, obtain the coordinates of the mouse cursor. By fine-tuning the pre-trained weights onto the newly gathered training and validation dataset, the TOOD and FCOS were able to detect cursors with relatively high performance. We compare the fine-tuned model against the same models on different datasets that were done by other researchers. This would also provide insights into our results and whether it has performed adequately or not. By comparing the size and characteristics of the dataset used to train the two models, it can be inferred that the introduction of duplicate images when used in moderation with a large number of unique images could be beneficial to the model. Additionally, adding more objects for the model to recognize other than cursors may be beneficial to the AT's user experience and expand the functionality of the system.

On the final module of the system that comes directly after the cursor detection module, it uses the PyAutoGUI package to move the cursor according to the command deciphered from the speech-to-intent module. Since the speech intents are only limited to "up", "down", "left", and "right", the cursor movements are also programmed to that same limitation. When the user speaks the command "up", the cursor would move up by several pixels that are hard-coded by the algorithm. The same value is applied to all other movement directions and as a result consecutive commands in the opposite direction would return the cursor to the initial position. Further works may find that moving the cursor towards a specific coordinate, by adding a calibration stage between the camera and screen resolution, could be more intuitive and adds more user-friendliness.

All of the modules should be interconnected following the pipeline representation above in **Fig. 1**. During initialization, the cursor detection module would be called upon from the saved fine-tuned models and a camera calibration step has been added to calibrate the camera's view with the display. Next, the speech-to-intent module would be initialized from the saved model. Once all the models are initialized and the camera has been calibrated, the system would be ready for hearing the user's commands. This is done iteratively until an interruption is invoked. In every loop, the system would first recognize the user's commands and output the corresponding intent, the camera would capture a single frame and run it through the detection model to output the coordinates, and finally, the cursor will move using the PyAutoGUI package according to the inferred intent.

Future works on this system would greatly benefit persons with special needs and enable them to effectively interact with computer devices based on cursor movement and USB interface just by using their voices. Each module of the system could also be improved upon by simply adding more utterances to the speech-to-intent module to improve the capability of the model, or adding more new, and unique data onto the cursor detection. The system could be extended to allow the Jetson Nano as a USB Gadget that will be recognized as a mouse HID when connected by USB. By doing this, the Jetson can move cursors of other devices as long as it has a compatible USB interface. Even more, improvements to extend its function using various deep learning algorithms can be done since the Jetson Nano could operate multiple deep learning models at the same time without any significant issues. Other peripheral equipment may also be connected to the Nano, such as sensors or motors. There could be various additions to the functions that the system can offer and with the availability of more powerful embedded devices, namely the Jetson Xavier, more computational power could be solved relatively easily.

C. Conclusion and Recommendation

Persons with special educational needs and disabilities benefit greatly from assistive technology (AT), which enables them to function more efficiently in daily life. NVIDIA's Jetson Nano is an affordable embedded system device with low power consumption and a GPU capable of operating AI frameworks. The user will be able to interact with this equipment, be mobile, and be able to connect to any computer with the appropriate interface. In addition to being easily accessible, Jetson Nano has a modularity that has the potential to solve other existing AT issues. By patching together a Speech-to-Intent module and a Mouse Cursor Detection module, this study moves the computer mouse with speech commands. The challenge with cursor detection lies in the small size of the cursor relative to the background and SAHI offers a suitable pipeline for use for this purpose. The FCOS and TOOD model presents a possible framework for detecting small objects, in this case cursors, and its performance would benefit greatly with the addition of new images into its datasets. Furthermore, through the use of the Linux operating system, future works can extend the functionality of the assistive technology device to include other USB-equipped devices with the USB Gadget API using a device based on embedded systems, such as the Jetson Nano or Xavier, which is capable of combining multiple deep learning models that are connected into this system.

IV. SELF REFLECTION

This internship program was heavily involved in deep learning research that uses the TensorFlow Framework as well as NVIDIA TensorRT optimization techniques. Although they were not directly applied into this reported research, these skills are essential in understanding the development of neural networks and for another research that the student was involved in. By using these skills, the student may be able to apply them to research or work involved in Artificial Intelligence in various fields and especially since deep learning form the basis to many A.I. advancements, such as ChatGPT that was developed and released recently by OpenAI (Haque et al., 2022).

Through this program, the student was able to identify their strength in programming and artificial intelligence development in research. Among the strengths showcased during program was that ability to learn and apply taught information within the program period. Through support from other people who are more experienced, the learning curve for new skills are drastically reduced. Collaborative works also outputs a greater impact on results than individual efforts. Weaknesses that occur during the internship is time and resource management that was not carefully monitored where sometimes other lab members were not able to utilize the resources available due to the student's lack of experience in resource management. Since there was not a fixed time on working hours, often times the student worked over normal working hours (9 A.M. – 5 P.M.) in order to satisfy the curiosities that linger. However, this was not sustainable for health and psychological reasons.

The values that i3L possess were directly translated during the research was done. Grit was necessary to resolve every problem that the student stumble upon without giving up where as the role of Role-Models in the program, in which case was the Ph.D. students, provided inspiration to excel in doing the work. Integrity was exercised during research and manuscript development in providing actual research results and also not plagiarizing texts.

Classes at i3L provided an important foundation in understnading machine learning however, the skills for deep learning were not provided at i3L. The most influential role from i3L classes was in python programming, which is the main programming language used for deep learning development, and manuscript writing. The student was able to conduct research of their own as well as aid the other Ph.D. student in several of their research. As a result, several research results were generated with the students collaborative efforts.

V. CONCLUSION & RECOMMENDATION

The internship program that was conducted was based on research. It was completed successfully and a manuscript was developed which will be attempted to be submitted for publication. Since the output of research are usually in the form of manuscripts and an attempt at publication is a natural proceeding, the student has fulfilled the goals of the internship and exceeded the initial expectation following the results of the research. This experience has increased the specialization of the study program into a more advanced field and the practical experience was very enriching to the theories that was taught by the i3L.

REFERENCES

Adadi, A. (2021). A survey on data-efficient algorithms in big data era. *Journal of Big Data 2021 8:1*, *8*(1), 1–54. https://doi.org/10.1186/S40537-021-00419-9

Akyon, F. C., Onur Altinuc, S., & Temizel, A. (2022). Slicing Aided Hyper Inference and Fine-Tuning for Small Object Detection. *2022 IEEE International Conference on Image Processing (ICIP)*, 966–970. https://doi.org/10.1109/ICIP46576.2022.9897990

(Boni) Li, Y., & Perkins, A. (2007). The impact of technological developments on the daily life of the elderly. *Technology in Society*, *29*(3), 361–368. https://doi.org/10.1016/J.TECHSOC.2007.04.004

Borg, J., & Östergren, P. O. (2020). Assistive Technology Facilities for the Students with Intellectual Disability in the Buds Special Schools in Kannur District of Kerala Anjali Rajeevan. *Studies in Indian Place Names*, *10*(4), 301–308. https://doi.org/10.3109/17483107.2014.974221

Das, P., Acharjee, K., Das, P., & Prasad, V. (2015). VOICE RECOGNITION SYSTEM: SPEECH-TO-TEXT. *Journal of Applied and Fundamental Sciences*, *1*, 2395–5562.

Haque, M. U., Dharmadasa, I., Sworna, Z. T., Rajapakse, R. N., & Ahmad, H. (2022). *“I think this is the most disruptive technology”: Exploring Sentiments of ChatGPT Early Adopters using Twitter Data*. https://doi.org/10.48550/arxiv.2212.05856

Hsieh, H. L., & Shanechi, M. M. (2018). Optimizing the learning rate for adaptive estimation of neural encoding models. *PLOS Computational Biology*, *14*(5), e1006168. https://doi.org/10.1371/JOURNAL.PCBI.1006168

Kim, M. G., Kim, M., Kim, J. H., & Kim, K. (2022). Fine-Tuning BERT Models to Classify Misinformation on Garlic and COVID-19 on Twitter. *International Journal of Environmental Research and Public Health*, *19*(9). https://doi.org/10.3390/IJERPH19095126

Lalor, J., Wu, H., & Yu, H. (2017). *Improving Machine Learning Ability with Fine-Tuning*.

Manjusha, L., & Suryanarayana, V. (2022). Detect /Remove Duplicate Images from a Dataset for Deep Learning. *Journal of Positive School Psychology* , *6*(2), 606–609. https://journalppw.com/index.php/jpsp/article/view/1547

McNicholl, A., Casey, H., Desmond, D., & Gallagher, P. (2021). The impact of assistive technology use for students with disabilities in higher education: a systematic review. *Disability and Rehabilitation. Assistive Technology*, *16*(2), 130–143. https://doi.org/10.1080/17483107.2019.1642395

Padilla, R., Netto, S. L., & da Silva, E. A. B. (2020). A Survey on Performance Metrics for Object-Detection Algorithms. *International Conference on Systems, Signals, and Image Processing*, *2020-July*, 237–242. https://doi.org/10.1109/IWSSIP48289.2020.9145130

Shi, G., Ke, S., & Banozic, A. (2022). The Role of Assistive Technology in Advancing Sustainable Development Goals. *Frontiers in Political Science*, *4*, 24. https://doi.org/10.3389/FPOS.2022.859272/BIBTEX

Shin, D.-J., & Kim, J.-J. (2022). A Deep Learning Framework Performance Evaluation to Use YOLO in Nvidia Jetson Platform. *Applied Sciences*, *12*(8), 3734. https://doi.org/10.3390/app12083734

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, *6*(1), 1–48. https://doi.org/10.1186/S40537-019-0197-0/FIGURES/33

Tolmacheva, A. E., Ogurtsov, D. A., & Dorrer, M. G. (2020). Justification for choosing a single-board hardware computing platform for a neural network performing image processing. *IOP Conference Series: Materials Science and Engineering*, *734*(1), 012130. https://doi.org/10.1088/1757-899X/734/1/012130

Valladares, S., Toscano, M., Tufiño, R., Morillo, P., & Vallejo-Huanga, D. (2021). *Performance Evaluation of the Nvidia Jetson Nano Through a Real-Time Machine Learning Application* (pp. 343–349). https://doi.org/10.1007/978-3-030-68017-6_51

Wu, Y., Liu, L., Pu, C., Cao, W., Sahin, S., Wei, W., & Zhang, Q. (2022). A Comparative Measurement Study of Deep Learning as a Service Framework. *IEEE Transactions on Services Computing*, *15*(1), 551–566. https://doi.org/10.1109/TSC.2019.2928551

Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., & Ling, H. (2021). Detection and Tracking Meet Drones Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2021.3119563