Performance Evaluation of NVIDIA Jetson Nano on Automated Machine Learning (AutoML) for Contrast-Enhanced Triphasic CT Scan Analysis in Focal Liver Lesion Detection

Angelique P. Bagtas[1], Khaye Q. Arellano[1], Eliezer Gale C. Bador[1], Bianca D. Duyag[1], Albert V. Josef Jr.[1],Janine Caile M. Polecina[1], Justin Ryan L. Tan[2]

[1]*Department of Electronics Engineering, Technological University of the Philippines, Manila, Philippines*
[2]*Department of Internal Medicine, Section of Gastroenterology, Chinese General Hospital and Medical Center*

{angelique.bagtas, khaye.arellano, eliezergale.bador, bianca.duyag, albert.josef, janinecaile.polecina}@tup.edu.ph}@tup.edu.ph

***Abstract*— This study evaluates the performance of the NVIDIA Jetson Nano in implementing Automated Machine Learning (AutoML) for the analysis of Contrast-Enhanced Computed Tomography (CECT) scans in detecting focal liver lesions (FLLs). FLL detection is critical for the early diagnosis and treatment of liver conditions, but manual analysis of CECT scans is time-consuming and error-prone. The Jetson Nano, a compact and powerful edge computing device, offers potential for real-time processing and analysis of medical imaging data. By utilizing the AutoML capabilities, specifically Roboflow Train, the study aims to assess the computational efficiency, speed, and accuracy of the Jetson Nano compared to traditional diagnostic methods. The findings indicate that the Jetson Nano can significantly enhance lesion detection, utilizing AutoML on a Jetson Nano-equipped prototype with a processing speed of 5.09 seconds when connected to Wi-Fi and 8.02 seconds using a mobile data connection, compared to a laptop. This approach reduces processing time and variability between different doctors' analyses, thereby offering a cost-effective and energy-efficient solution for healthcare applications.**

***Keywords*—NVIDIA Jetson Nano, Automated Machine Learning (AutoML), Contrast-Enhanced Computed Tomography (CECT) Scan, Focal Liver Lesions (FLLs), Medical Imaging, Real-time Processing, Embedded AI Computing, Roboflow.**

I. INTRODUCTION

The liver is located under the ribs on the right side of the upper abdomen, acts as the body's largest solid organ, performing vital functions like detoxification, blood sugar regulation, and blood clotting [1]. Liver masses, known as focal liver lesions (FLLs), can be benign or malignant, may not require intervention in benign cases, but immediate medical attention is essential for malignant masses [2,3,4]. These lesions include liver cysts, often benign unless they grow significantly, liver abscesses necessitating prompt treatment due to their potential mortality risk if left untreated, and solid tumors like hepatocellular carcinoma (HCC) and cholangiocarcinoma, both benign and malignant in nature, with benign tumors frequently asymptomatic and detected via imaging scans [5,6,7, 8]. Imaging modalities like computer tomography (CT) scans play a significant role in diagnosing and characterizing FLLs, with contrast-enhanced triple-phase CT scans being particularly useful [9,10]. Artificial intelligence (AI), particularly convolutional neural networks (CNNs), has been employed in various studies to enhance the detection and diagnosis of FLLs, utilizing imaging modalities such as CT scans.

The NVIDIA Jetson Nano opens opportunities for creators, self-taught programmers, and enthusiasts in embedded technology worldwide to leverage the capabilities of artificial intelligence. This user-friendly yet powerful computing device empowers users to execute numerous neural networks simultaneously, enabling tasks like image classification, object recognition, segmentation, and audio processing [11]. Essentially, it functions as a compact computer equipped with a modest graphics processing unit. By supporting CUDA, NVIDIA's general-purpose computing language for GPUs, the Nano facilitates a wide range of GPU-optimized machine learning techniques [12]. The Jetson Nano development board revolutionizes the concept of artificial intelligence by enabling the creation of cost-effective and energy-efficient AI systems. Featuring four ARM cores and a Maxwell GPU serving as a CUDA computing accelerator and video engine, this single-board computer broadens the horizons for projects demanding intense graphics and computation. As a production-ready System-on-Module (SOM) boasting 472 GFLOPs, it expedites the implementation of contemporary AI algorithms [13]. Utilizing AutoML technology, specifically the Roboflow train, this study seeks to assess the performance of NVIDIA jetson nano on AutoML for contrast-enhanced triphasic CT scan analysis in FLL detection.

II. BACKGROUND OF THE PROBLEM

In the healthcare industry, there have been a few studies focused on implementing Automated Machine Learning (AutoML) systems that are specialized for health services, but no study focuses on detecting FLLs [10]. Accurate detection of focal liver lesions is very important for early diagnosis and treatment of liver cancer and other hepatic conditions. Triphasic contrast-enhanced computed tomography (CECT), which captures images of the liver in three distinct phases after intravenous contrast injection, is commonly used for this purpose. However, manually analyzing these large, complex imaging datasets is time-consuming and susceptible to human error and inconsistencies. Automated machine learning (AutoML) platform can extract complex patterns and features from vast amounts of data, which may be missed by human observers. It can automate the selection and optimization of machine learning algorithms and hyperparameters, leading to models with higher performance and generalizability. Radiologists could enhance lesion detection accuracy while reducing processing time and inter-observer variability by applying AutoML to triphasic CECT data.

The NVIDIA Jetson Nano, a compact and powerful edge computing platform, holds promise for real-time processing and analysis of medical imaging data. It is a powerful yet energy-efficient embedded AI computing device that could enable deployment of AutoML models for focal liver lesion detection. The application of machine learning methods has shown promise in enhancing healthcare outcomes, lowering medical expenses, and advancing clinical research. A study states that despite these benefits, many hospitals currently refrain from implementing machine learning solutions [14]. Understanding the Jetson Nano's computational efficiency, speed, and accuracy in handling the complex algorithms involved in automated machine learning for medical imaging is crucial for determining its viability as an edge device in healthcare applications. This

research aims to bridge the existing gap by conducting assessment of the Jetson Nano's performance, providing valuable insights into its potential role in enhancing the efficiency of triphasic CECT scan analysis for focal liver lesion detection through automated machine learning techniques.

III. Objectives

1. To assess the speed of automated machine learning (AutoML), particularly the Roboflow train, when utilized on an NVIDIA Jetson Nano compared to a laptop for image segmentation and classification of the focal liver lesions.

2. To evaluate the performance of the software operated with NVIDIA Jetson Nano utilizing the ISO 25010 software quality matrix.

IV. Related Works

The performance evaluation of the NVIDIA Jetson Nano in the context of Automated Machine Learning (AutoML) for CECT Scan Analysis in Focal Liver Lesion Detection is a critical endeavor. According to the recent studies of Biglari & Tang [15], the Jetson Nano demonstrates varying average speeds across different applications of computer vision algorithms. These include 30.18ms for face detection, 333.33ms for autonomous vehicle obstacle recognition, 476ms for medical diagnosis and disability assistance, 250ms for safety and security, 48ms for smart city management, 2.78ms for LiDar, radar, audio, and motion recognition models, 13.74ms for embedded machine learning optimization papers, and 0.56s for benchmark and review papers.

Heldenkombinat Technologies [16] evaluated the transfer-learning performance of Jetson Nano utilizing ResNet18, recording an average time of 12.6s during a delivery task. Researchers have used this opportunity to look at the summary of real-time ML applications and the practical capabilities of the Jetson Nano. Valladares et al. [17] pursued a study where the Jetson Nano platform was tested through Dew Computing implementations. Concerning the Jetson Nano's performance in ML task execution, the GPU consistently operated at 99% utilization throughout all tests, as it handles vehicle identification computations. RAM consumption remained at 2.1 GB, approximately 50% of its capacity. Energy consumption is directly related to vehicular flow.

A study by Prashanthi S.K et al. [18] concluded that despite Jetson Nano's lower CPU frequency compared to higher-end models like the AGX and NX, the Nano demonstrates the potential for efficient processing through pipelining. While without pipelining, the Nano exhibits longer stall times per epoch at 42.6*s*, with pipelining, this time significantly reduces to 17.2*s*, aligning more closely with the other models' performance. This suggests that by effectively leveraging GPU compute times, the Nano compensates for its CPU limitations, emphasizing the importance of optimizing resource allocation rather than solely relying on individual component speeds when configuring devices for machine learning tasks.

Sahani et al. [19] employed the Jetson Nano to enhance the performance of ARTSENS, a non-invasive A-mode ultrasound-based device used for measuring arterial stiffness (AS). The study focused on automating carotid detection, wall localization, and inner lumen diameter measurement using deep neural network (DNN) models. These DNNs outperformed earlier signal processing methods, achieving significantly lower errors with an accuracy of 90.7% and an inference time of 2.78 ms.

Using deep learning methods, Prabhu & Verma [20] developed a model that automatically identifies between the DFU class and the healthy skin from plantar thermograms. The study also included the Jetson Nano, which operates multiple neural networks concurrently. The suggested work's performance is evaluated and contrasted with current techniques, including DenseNet, VGGNet, and MatConvNet architectures. The results demonstrated a noteworthy enhancement with a maximum accuracy of 97.9%.

Baghersalimi et al. [21] proposed an approach of the NVIDIA Jetson Nano Developer Kit. The study tested its accuracy based on the EPILEPSIAE database, one of the most extensive public epilepsy datasets for seizure detection. The study observed that personalized federated learning increases all the performance metrics, with an accuracy of 91.58% with 1.86 hrs of the training process.

Table 1. Performance of Jetson Nano

Authors	Similarities	Difference	Speed Rate
Biglari & Tang [15]	The speed of the Jetson Nano was tested.	It was tested across different applications of computer vision algorithms.	476ms
Helden-kombinat Technolo-gies [16]	Transfer-learning performance of Jetson Nano	The ResNet18 was utilized.	12.6s
Valladares et al. [17]	Performance evaluation of the Jetson Nano on ML application.	The Jetson Nano's performance was tested based on the GPU utilization, RAM consumption, and Energy Consumption.	-
PrashanthiS.K et al. [18]	Performance of Jetson Nano devices for training deep learning models.	The Jetson Nano was tested for its potential efficient processing through pipelining.	42.6s reduced to 17.2s (stall time)
Sahani et al. [19]	Utilization of the Jetson Nano on ML application.	The Jetson Nano was employed to enhance the performance of the ARTSENS.	27.8ms
Prabhu & Verma [20]	Utilized the Jetson Nano on ML application in Biomed.	The Jetson Nano was employed to enhance diabetic ulcer detection.	-
Baghersa-limi et al. [21]	Usage of Jetson Nano for ML and training process.	The Jetson was used in the context of epileptic seizure detection.	1.86 hrs (training process)

Krüger et al. [22] addressed the critical interplay between GUI applications and resource-limited embedded devices. Based on their GUI rendering capabilities, the study compared the NVIDIA Jetson Nano and Raspberry Pi 3B+. Ten (10) cases were conducted, and in all the cases, the NVIDIA Jetson Nano's GUI rendering capabilities are significantly higher than that of the Raspberry Pi 3B+.

Yose et al. [23] compared the performance, frame capture per second, and execution load time of the GUI-based portable

smart attendance checker using the personal computer versus the Jetson Nano. The execution time on the Jetson Nano ranges from 15s to 22s, while on Windows, it takes 14 to 18 seconds. Notably, the performance and capture rate are significantly affected when there are available faces to extract and compare, especially on the Jetson Nano platform.

According to the study of Ooi et al. [24], there is much opportunity for developing novel sensor-driven applications with hierarchical FL systems. However, because these applications are diverse and have similarly diverse performance needs, the technology may face many challenges. In order to systematically address the needs of these systems, the study advised employing accepted software standards. The ISO/IEC 25012 and ISO/IEC 25010 frameworks are concerned with quality and data quality.

Bautista et al. studied the design of a performance measurement framework for cloud computing. The framework's design draws upon metrology concepts and aspects of software quality directly linked to performance. These aspects align with the ISO 25010 international standard. Our literature review revealed a close association between performance efficiency, reliability, and Jain's measurement perspective. Consequently, this research proposal integrates ISO 25010 concepts into Jain's framework for measuring information system performance.

TABLE 2. PERFORMANCE OF SOFTWARE

Authors	Similarities	Difference
Krüger et al. [22]	The Jetson Nano was used to run the GUI application.	The Jetson Nano was compared to another embedded device: Raspberry Pi 3B+.
Yose et al. [23]	The Jetson Nano was used to run the GUI-based detector.	The performance of the Jetson Nano and personal computer compared.
Ooi et al. [24]	The ISO/IEC 25010 Standard was employed as a software evaluator.	The software that they used utilizes IoT sensors rather than ML.
Bautista et al. [25]	The ISO/IEC 25010 Standard was employed as a software evaluator.	They used the ISO 25010 in Cloud Computing rather than ML.

V. METHODOLOGY

A. Conceptual Framework

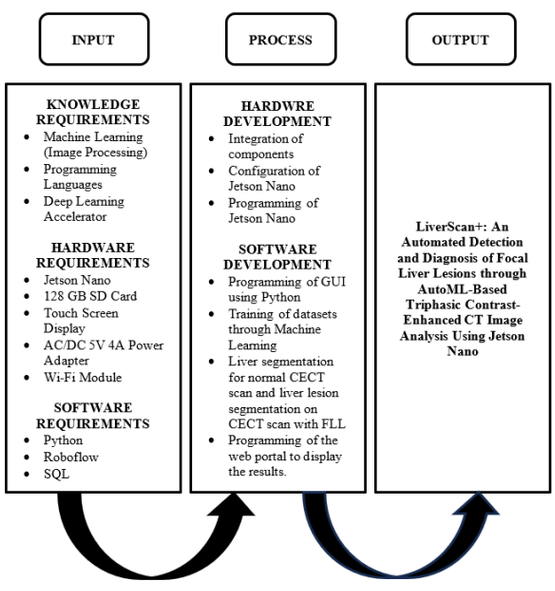

FIG. 1. INPUT PROCESS OUTPUT (IPO) MODEL

Figure 1 shows the Input Process Output Model of the study. For the knowledge criteria, it requires the researcher to have a knowledge of Machine Learning (ML) especially on image processing technique. The collected CECT images undergo annotation, where FLLs are highlighted and assigned to their respective classes. This annotation process provides the algorithm with the knowledge needed to classify classes based on their differences. Subsequently, the dataset is balanced and divided into training, validation, and test sets. This standard machine learning practice ensures that the model goes beyond memorizing training data and learns underlying patterns applicable to new, similar data. The training set is utilized to train the machine learning model, allowing it to discern patterns and relationships within the data. The validation set evaluates the model's performance during training, aiding in hyperparameter adjustments to prevent overfitting. After training and tuning with the training and validation sets, the model's performance is assessed on entirely new and unseen data using the test set. This step provides an unbiased estimate of the model's real-world performance. Following this, dataset preprocessing occurs, involving steps to create a clean, well-structured, and appropriately formatted dataset that facilitates effective training and generalization of the machine learning model. After preprocessing is the augmentation of the dataset, artificially expanding its size through various transformations to enhance the model's generalization and robustness. Lastly, in the training phase, the dataset passes through the model to generate predictions. The model compares its predictions with actual labels using the chosen loss function to calculate the error. The optimizer is then employed to adjust the model's weights and biases, minimizing the loss function. Iterative passes through the training dataset (epochs) refining the model, completing the training process.

When it comes to the hardware of the prototype, the Jetson Nano has the role of a primary processor, adept at handling complex machine learning tasks. Additional components are connected to it to provide power, enable Wi-Fi and Bluetooth connectivity, prevent thermal issues in the prototype, offer storage for applications and the operating system on the Jetson Nano, and facilitate GUI display. Once the prototype is assembled, the GUI becomes accessible for viewing, allowing doctors to interact with its interface for predictions. The results, categorizing tumors, cysts, or abscesses based on the trained dataset, can be observed. Furthermore, patient records are accessible through the web portal. Developing both a GUI and a web portal necessitates a proficiency in programming to enable them to utilize a common database.

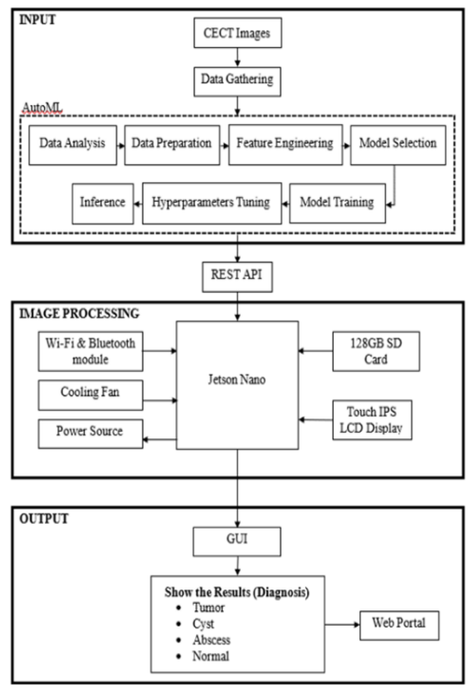

FIG. 2. BLOCK DIAGRAM OF THE DEVICE

Figure 2 outlined the figures illustrate the overall plan for developing hardware, GUI, and a Web portal. The dataset, consisting of CECT images or necessary input for pre

dictions, undergoes processing via AutoML. This process leads to the creation of a REST API crucial for deploying the processed dataset to the GUI. Following this, the hardware setup, primarily featuring the Jetson Nano as the primary processor, is configured to display predictions. Additional components are connected to ensure functionality, including power, connectivity, thermal management, and storage. Once assembled, doctors can utilize the GUI to view predictions categorizing tumors, cysts, or abscesses based on the trained dataset. Patient records are also accessible through the web portal.

B. Research Process Flow

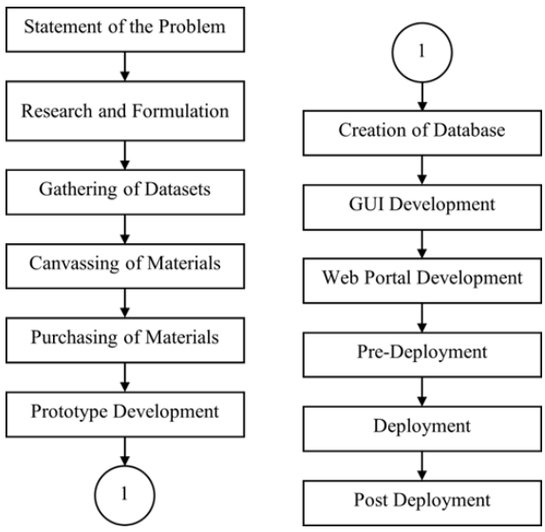

Fig. 3. Process flow of the Study

Figure 3 shows the process flow that will serve as a guide for researchers as they follow the study's progress and complete the necessary processes to reach the final stage of verification. This arrangement will aid in meeting the study's objectives and completing the research.

C. Materials and Equipment

TABLE 3. BILL OF MATERIALS

COMPONENTS	QUANTITY	UNIT COST (Php)	TOTAL COST (Php)
Jetson Nano	1	13,000	13,000
128 GB SD Card	1	700	700
Touchscreen Display	1	8,968	8,968
U-Shaped Mini HDMI to HDMI Adapter	1	98	98
HDMI to HDMI Cord	1	151	151
U-Shaped USB C to USB C Adapter	2	159	324
Intel 8265 NGW Wi-Fi & Bluetooth Module for Jetson Nano	1	877	877
Extension Cord	1	117	117
Acrylic Sheet	5	129	645
Acrylic Stand	1	1785	1785
Aluminum Profile	1	500	500
Barrel Jack Power Adapter	1	217	217
Cooling Fan for Jetson Nano	1	202	202
USB Hub	1	280	280
VPS	1	3,662	3,662
Domain	1	560	560
Plesk	1	9,732	9,732
TOTAL			**₱41,818**

Table 3 shows the list of material and the corresponding prices that will be used in the construction of hardware and software.

D. Hardware Construction

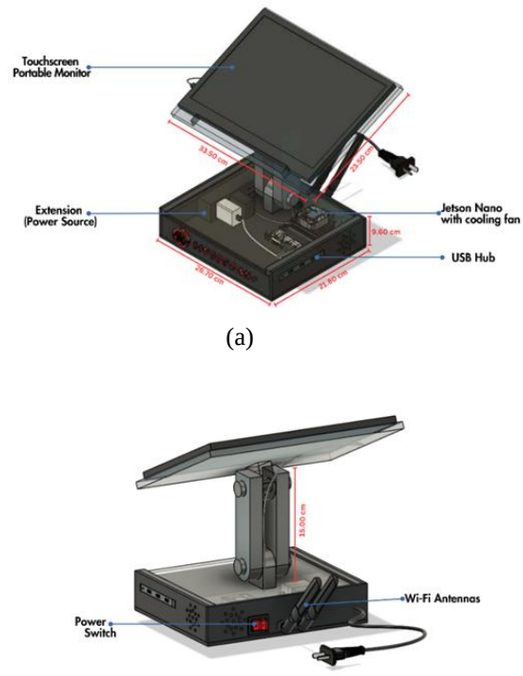

FIG. 4. 3D MODEL OF A DEVICE: (A) SIDE VIEW (B) BACK VIEW)

Figure 4 displays both the side view and the back view of the device, showcasing some of its components along with the dimensions of the upper plate, acrylic box, and the holder. It illustrated the touchscreen portable monitor, power source, USB hub and the Jetson Nano with a cooling fan. Additionally, it included the dimensions of the upper plate and the acrylic box. The length and width of the upper plate are 33.50 cm and 23.50 cm, respectively. The dimensions of the acrylic box are 26.70 cm in length, 21.80 cm in width, and 9.60 cm in height.It also depicts the back view of the 3D model of the device. It illustrates the power switch and the Wi-Fi antennas. Additionally, it includes the height of the holder, which is 15.00 cm.

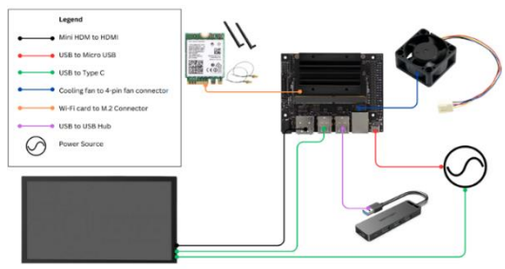

FIG. 5. DEVICE COMPONENT CONNECTION

Figure 5 illustrates the connection of components within the device, including a legend for their connections. An HDMI cable and a USB to Type-C cable connect the touchscreen portable monitor to the Jetson Nano for output display and touchscreen purposes. A Type-C cable connects the monitor to the power source. A micro-USB to USB connector connects the Jetson Nano to the power source. A USB hub is connected to the Jetson Nano's USB port. The Wi-Fi card is connected to the M.2 connector of the Jetson Nano, and the cooling fan is connected to the pins on the Jetson Nano.

E. Hardware Materials

The integration of diverse materials in the design, processing, and analysis will bring about the creation of the device. Listed below are the components that will be utilized in the development of the hardware specifically for detecting FLLs—cysts, abscesses, and solid tumors—through CECT scan images across different phases: unenhanced, arterial, and portal venous.

a.) Jetson Nano

Jetson Nano is a powerful single-board computer designed for AI and machine learning applications. It features a GPU, CPU, memory, and various interfaces for external devices. Its purpose is to provide an affordable platform for developing and deploying AI in various fields, such as robotics, healthcare, and smart cities. This will be used to speed up the computation-intensive tasks involved in training and running deep learning models for image processing.

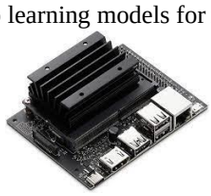

FIG. 9. JETSON NANO

b.) SD Card

The smallest flash memory card alternative for portable electronic devices is the Micro Secure Digital (SD) Card. This is an 11mm x 15.0mm sized card that serves as storage for holding different volumes of data in tech devices. It has different specifications, memory sizes, as well as speed classes. Stores the Operating System (OS) and the images collected through trained data.

FIG. 6. SD CARD

c.) Intel 8265 NGW Wi-Fi & Bluetooth Module

The Intel 8265 NGW supports the 802.11ac Wi-Fi standard, offering high-speed wireless connectivity. Additionally, it features Bluetooth 4.2, enabling wireless communication with devices like keyboards, mice, headphones, and other peripherals. This module will be used to connect the Jetson Nano to Wi-Fi networks and communicate with Bluetooth-enabled devices such as phones.

FIG. 7. INTEL 8265 NGW WI-FI & BLUETOOTH MODULE

d.) IPS Touch Display

The touch panel is a 15.6-inch display screen with a capacitive touch interface, 1920x1080 HD resolution, and IPS LCD technology for better color reproduction and viewing angles. It has an aspect ratio of 16:10 which is commonly used for tablets and laptops. It will be used to display visuals and manipulate the hardware device.

FIG. 8. TOUCHSCREEN PORTABLE MONITOR 15.6 INCH

e.) 5V, 4 Pin Cooling Fan

The 5V, 4 pin cooling fan is a small fan that is used to cool the NVIDIA Jetson Nano developer kit. The fan is powered by 5 volts and has a 4-pin connector. The fan is typically mounted on the top of the Jetson Nano, where it blows air across the heat sink.

FIG. 9 5V, 4 PIN COOLING FAN

f.) USB Hub

The USB hub is a device that expands a single Universal Serial Bus (USB) port into multiple ports, providing additional USB connectivity options for the Jetson Nano. Specifically, it allows for connecting a keyboard and a mouse to the Jetson Nano.

FIG. 10. USB HUB

B. Software Construction

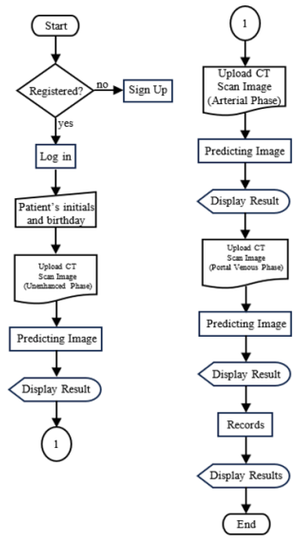

FIG. 11. FLOWCHART OF THE GUI

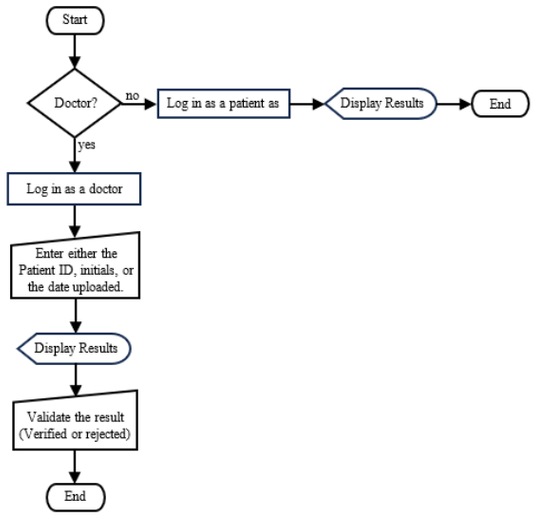

FIG. 12. FLOWCHART OF THE WEB PORTAL

Figures 11 and 12 presented the flowcharts of the study, with a specific focus on the GUI application and the Web portal. The GUI application serves as a platform for doctors to upload CT images in various phases and view the corresponding prediction

results. Within this application, doctors have the option to create an account or log in if they already have an existing account registered in the database.

Conversely, the web portal provides doctors with the capability to access and review patient records. Additionally, it facilitates communication between doctors for discussions on specific results. The comments section is integrated into the patient login segment, where patients are required to input a unique patient ID. Upon logging in, patients gain access solely to the prediction results generated from their CT images and discussions among doctors.

This dual-system approach, encompassing the GUI application and the Web portal, enhances the overall functionality of the study, providing a comprehensive platform for both doctors and patients

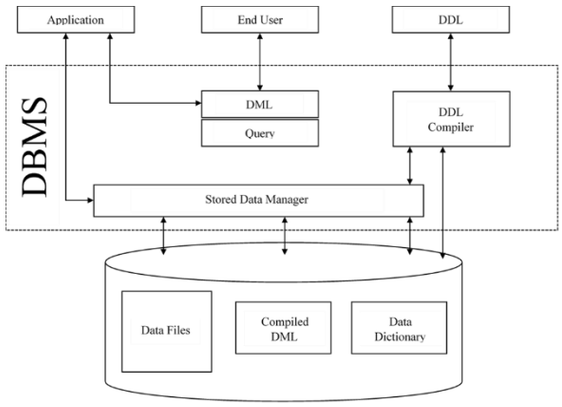

FIG. 13. DBMS OF THE STUDY

Figure 13 illustrates how the Database Management System manages validated dataset that comes from the doctors. The data files include the non-image information from the patient. The GUI's data is stored in a phpMyAdmin database, which makes use of the MySQL database management system. A web portal can operate properly because the default MySQL structure is present in all phpMyAdmin databases. After then, it is managed and under the supervision of a database management system (DBMS). To better organize complex data, this software divides it into smaller pieces. Saving files and folders on a computer drive is similar to how DBMS functions. Specific file categories can be found in each folder. This makes it simple for users to find, alter, and delete any desired data.

F. Software Development

The software comprises a graphical user interface (GUI) application seamlessly integrated with hardware, along with a web portal. The GUI application allows doctors to create and log in to their accounts, where they can upload CECT scan images with various phases and view predictions. The web portal supports two login modes – one for doctors and another for patients. Doctors can access and manage their patients' records, engaging in discussions through a comment system. Patients, on the other hand, have limited access, allowing them to view only their results and discussions initiated by doctors regarding their comments.

a.) Roboflow

This is where the CECT images or the dataset have been annotated, pre-processed, augmented, and trained. This is used for the application of computer vision techniques used in the prediction of FLLs in different phases.

FIG. 14. ROBOFLOW LOGO

b.) Python

The rest API produced by RoboFlow has been implemented in Python. This serves as a programming language used in building a GUI application. Libraries like custom Tkinter have been installed for the design of GUI applications.

FIG. 15. PYTHON LOGO

c.) Hypertext Markup Language (HTML)

This programming language is used for the construction of Web portal as it controls the layout of the content of the webpage and provides structure for web page design.

FIG. 16. HTML LOGO

d.) Cascading Style Sheet (CSS)

This is a custom stylesheet language used to apply style to the web page elements and make it responsive. It handles the user's interface experience.

FIG. 17. CSS LOGO

e.) JavaScript

This serves as the back-end language of the Web portal. It adds interactivity to web page. It handles complex functions and features.

FIG. 18. JAVASCRIPT LOGO

f.) Hypertext Preprocessor (PHP)

It is a server scripting language used to create the website more interactive. This is used in our Web portal to connect to its database.

FIG. 19. PHP LOGO

g.) Structured query language (SQL)

It is a programming language for storing and processing information in the database. It is used to communicate and exchange information from GUI application to Web portal.

FIG. 20. SQL LOGO

h.) Hostinger

It is the software used to set up a domain on the Web Portal. The researchers use VPS hosting to connect the database to the GUI application and Web Portal, resulting in a shared database across both technologies.

FIG. 21. HOSTINGER LOGO

i.) Plesk

A web hosting control panel for virtual private or dedicated servers with an easy-to-use UI and automation tools. It is used to upload an existing database and website to connect it to Hostinger.

FIG. 22. PLESK LOGO

G. Speed Testing

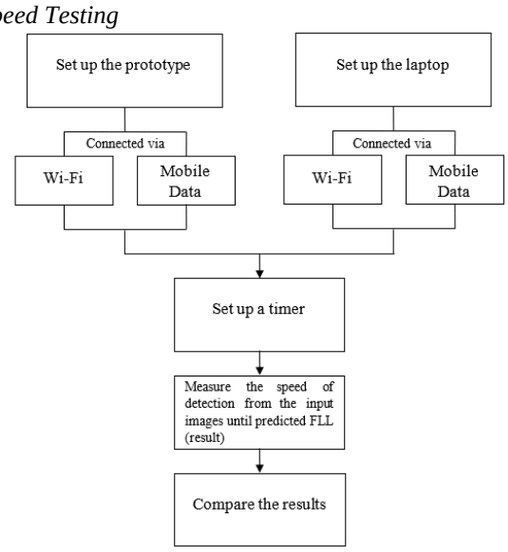

FIG. 23. SPEED TESTING PROCESS FLOW

In figure 23, the process flow is outlined for measuring the speed of a device with Jetson Nano which is the prototype versus a device without this technology, such as a laptop. This test aims to determine how quickly the device provides results upon uploading CECT images as an input. Since the detection depends on the device's internet connectivity, the speed testing was divided into four segments:

1. Comparison of speed of AutoML in prototype between Wi-Fi and mobile data connectivity

2. Comparison of speed of AutoML in laptop between Wi-Fi and mobile data connectivity

3. Comparison of speed of AutoML using Wi-Fi connectivity between prototype and laptop

4. Comparison of speed of AutoML using mobile connectivity between prototype and laptop

The study utilized two distinct methods of internet connectivity, Wi-Fi and mobile data. Both the prototype device and the laptop were subjected to these connectivity modes. Data collection proceeded initially with the prototype connected to Wi-Fi, followed by the same device connected to mobile data, facilitating subsequent comparison of results. Subsequently, the laptop underwent a similar process, initially connected to Wi-Fi and then to mobile data, allowing for a comparative analysis.

A further step involved the comparison of data gathered from the preceding procedures, particularly focusing on the utilization of Wi-Fi by both the prototype and the laptop. Lastly, an examination was conducted to compare data collected from the utilization of mobile data on both devices. This procedure was crucial for several reasons. Primarily, understanding the impact of different types of internet connectivity on device performance can help optimize the conditions under which the prototype and laptop are used. For example, if one type of connectivity significantly enhances detection speed, it could guide future implementations and improve efficiency. Additionally, this comparison provides insights into the relative capabilities of the prototype versus the laptop, highlighting any potential advantages or limitations of each device. Collecting data across these four segments was essential for determining which option is more suitable for achieving fast detection, ultimately aiming to enhance the overall performance and reliability of the detection system.

The researchers utilized a timer to track the time from when the user input a CECT image into the system until the result of lesion detection per phase was displayed. After tracking the individual time per phase, it was summed up to obtain the time taken to produce a result per patient, which represented its speed. To compute the speed per patient, the formula was:

$$\sum x = U_s + A_s + PV_s$$

Where:
$\sum x$ = Speed per patient
U_s = the time in seconds in unenhanced phase
A_s = the time in seconds in arterial phase
PV_s = the time in seconds in portal venous phase

Upon obtaining the speed for each patient, the researchers calculated the average speed per lesion and determined the overall average speed required for a device with Jetson Nano and without this technology to process the detection or machine learning algorithm applied. The formula used for average speed per FLL was:

$$x = \frac{\sum x}{n}$$

Where:
x = average speed per FLL
$\sum x$ = Speed per patient
n = No. of Patient per FLL

The formula used for overall average speed (S) is:

$$S = \frac{x}{N}$$

Where:
x = Average Speed per FLL
N = Total no. of Patient

These findings from the study of Heldenkombinat Technologies [26] indicate that Jetson Nano generally exhibits lower inference times when embedded with machine learning capabilities, particularly for various image vision tasks that the datasets were pretrained under different CNN models. However, the current study diverges from previous research by focusing on a distinct application: detecting FLLs in CECT images inputted from patients, which occurs in three phases and is also influenced by internet speed. Given these parameters, the researchers aim for a maximum processing time of 12.6s to complete all operations efficiently.

This aims to evaluate whether the integration of Jetson Nano technology into the study's prototype operates and maximizes its functionalities, particularly in optimizing and operating multiple neural networks in parallel for tasks such as image classification and segmentation.

H. Statistical Treatment

The statistical method used to compare the speed of the device with a Jetson Nano versus the device without it, such as a laptop using Wi-Fi and mobile data, is the independent

samples t-test. The independent samples t-test is a parametric test that determine if there are statistically significant differences between the means of two independent groups. There is not an assumption of normal distribution, but there is an assumption that the two standard deviations are equal [27]. P-values less than 0.05 lead to the conclusion of rejecting the null hypothesis, indicating a significant difference. Conversely, p-values greater than 0.05 result in accepting the null hypothesis, indicating no significant difference. The comparison of speeds is part of the first objective, and it is divided into four (4) categories: Speed of Prototype (Wi-Fi versus Mobile Data), Speed of Laptop (Wi-Fi versus Mobile Data), Speed of AutoML using Wi-Fi (Prototype versus Laptop), Speed of AutoML using Mobile Data (Prototype versus Laptop).

For the speed of prototype and laptop, both the prototype and the laptop are evaluated under two distinct network conditions: Wi-Fi and mobile data. The performance measurements taken under Wi-Fi are independent of those taken under mobile data. This independence holds true whether comparing the prototype or the laptop. The data points for each condition do not influence each other, making them suitable for an independent samples t-test.

For the speed of AutoML using Wi-Fi and Mobile Data (Prototype versus Laptop), the two distinct devices are the prototype and the laptop. The performance measurements of the prototype and the laptop are independent of each other, meaning the speed of one device does not influence the speed of the other. This independence holds true under both network conditions (Wi-Fi and mobile data). The speed data for each device (prototype and laptop) are collected independently, making them suitable for an independent samples t-test.
The general formula for the t-value when the sample sizes are the same is:

$$t = \frac{\overline{X_1} - \overline{X_2}}{\sqrt{\left(\frac{{s_1}^2}{n_1} + \frac{{s_2}^2}{n_2}\right)}}$$

Where:
t = t-value
$\overline{X_1}$ = Mean value of the first group
$\overline{X_1}$ = Mean value of the second group
s_1= Standard deviation of the first group
s_2= Standard deviation of the second group

I. Performance Evaluation

The ISO 25010 will be used an evaluation standard to evaluate the performance of the device that addressed one of the goals of the study's fourth objective which is to validate the effectiveness of the device and its performance using the ISO 25010 software quality matrix. An evaluation survey is distributed to the doctors who operated the prototype during the deployment. In general, system quality standards in the form of web applications can assessed from different attributes [28].

The purpose of the survey to the study was to measure the quality of LiverScan+ in terms of the quality of information system development and to measure the quality of the system from the user side of the system. The ISO 25010 is a software quality standard defining models for software product and quality in use [29]. It is used to identify software quality standards related to functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, and portability [30]. The researchers used seven categories of ISO 25010 such as functional suitability, reliability, usability, performance efficiency, maintainability, security, and compatibility since these are the categories suitable for the assessment of the device.

The following activities were undertaken to evaluate the performance of the device's system:

1. The device was deployed in the selected hospital, and developers invited evaluators—doctors from the hepatology/gastroenterologydepartment—to demonstrate how the device operates.
2. The device features a GUI that detects FLLs using CECT scan images.
3. Sufficient time was provided for evaluators to thoroughly explore the system's features.
4. Evaluators were instructed to assess the application using the ISO 25010 Software Materials Evaluation. The survey is composed of seven (7) categories with different number of indicators. It is a 5-point Likert scale, indicating 5 as the highest and 1 as the lowest.
5. The rating scale for evaluation, shown in Table 4, serves as a standard tool commonly employed in questionnaire-based research. Its primary purpose is to interpret responses collected through survey research.

TABLE 4. RATING SCALE FOR EVALUATION

Rating	Description
5	Excellent
4	Very High
3	High
2	Medium
1	Low

6. The data was collected and organized and tabulated to calculate the mean per indicator and the overall mean per category.

   The formula used for the mean per indicator (x) is:

$$\text{X} = \frac{\text{Total Score per Indicator}}{\text{No. of Respondents}}$$

   To get the overall mean per category ($\bar{x}$), the formula is:

$$\bar{x} = \frac{\text{Total Weighted Mean per Indicator}}{\text{Total No. of Respondents}}$$

7. The numerical ratings were interpreted using the scale provided in Table 5 to assign corresponding descriptive ratings. The numerical scale was used to assess the performance of the device's system based on the ratings obtained during the survey.

TABLE 5. NUMERICAL SCALE AND ITS DESCRIPTIVE RATING

Numerical Scale	Description
4.21 – 5.00	Very Effective
3.41 - 4.20	Effective
2.61 - 3.40	Moderately Effective
1.81 - 2.60	Ineffective
1.00 - 1.80	Very Ineffective

VI. RESULTS AND DISCUSSION

The results in testing process were obtained using a testing dataset, which include the speed tests for the prototype and laptop utilizing AutoML, and the system performance evaluation, respectively.

Speed Testing

The comparison involved measuring the speed of both the prototype and a laptop using a timer, tracking the time taken from inputting images into the device until the prediction was shown. Both devices were tested using Wi-Fi and mobile data connections. For the Wi-Fi connection, both the prototype and the laptop were operated under a 400 Mbps connection speed. In contrast, during the mobile data connection test, the prototype operated under a 13 Mbps speed, while the laptop operated at 21 Mbps.

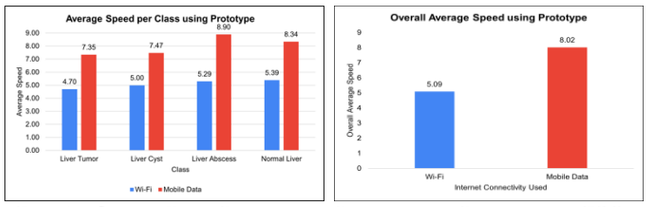

FIG. 24 SPEED TEST RESULT OF PROTYPE IN DETECTING FLLS BETWEEN WI-FI AND MOBILE DATA

Figure 24 displays the average speed results for each class using a prototype that utilizes both Wi-Fi and mobile data connectivity. When using Wi-Fi, the average detection speeds were 4.70 seconds for liver tumors, 5.00 seconds for liver cysts, 5.29 seconds for liver abscesses, and 5.39 seconds for normal liver, resulting in an overall average speed of 5.09 seconds. In contrast, when using mobile data, the detection speeds increased to 7.35 seconds for liver tumors, 7.47 seconds for liver cysts, 8.90 seconds for liver abscesses, and 8.34 seconds for normal liver, with an overall average speed of 8.02 seconds.

TABLE 8. COMPARISON OF SPEED OF AUTOML IN PROTOTYPE BETWEEN WI-FI AND MOBILE DATA CONNECTIVITY

Tested Variables	Statistic (t-test)	p-value	Decision	Conclusion
Tumor	-6.22	0.000	Reject Null	HAS a Significant Difference
Cyst	-3.48	0.056	Accept Null	NO Significant Difference
Abscess	-4.25	0.041	Reject Null	HAS a Significant Difference
Normal	-5.41	0.006	Reject Null	HAS a Significant Difference

Table 8 summarizes the comparison analyses which tested if there are significant differences in speed of detecting focal liver lesions between wireless fidelity connection and mobile data. Independent samples t-test indicated significant differences in speed in detecting tumors ($t = -6.22$, $p = 0.000 < 0.01$), abscess ($t = -4.25$, $p = 0.041 < 0.05$), and normal status livers ($t = -5.41$, $p = 0.006 < 0.01$). Meanwhile, no significant difference was established in detecting cysts ($t = -3.48$, $p = 0.056 > 0.05$).

Null Hypothesis (H_0):

There is no significant difference in speed in detecting tumor, cyst, abscess, and normal livers between Wi-Fi and Mobile Data.

Alternative Hypothesis (H_a):

There is no significant difference in speed in detecting tumor, cyst, abscess, and normal livers between Wi-Fi and Mobile Data.

Decision:

As a result, while using the prototype, the data in this table indicates a significant difference in speed between mobile data and Wi-Fi across most classes. Therefore, it can be concluded that when comparing the speed of the prototype in detecting FLLs between Wi-Fi and mobile data, Wi-Fi enhances the detection speed more than mobile data.

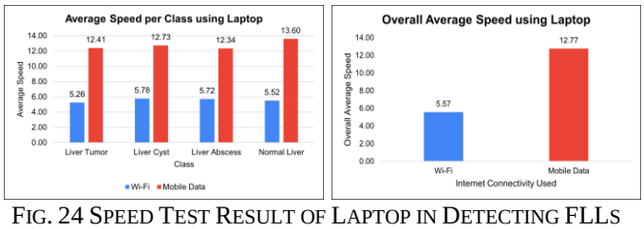

FIG. 24 SPEED TEST RESULT OF LAPTOP IN DETECTING FLLS BETWEEN WI-FI AND MOBILE DATA

Figure 24 displays the average speed results for each class using a laptop with both Wi-Fi and mobile data connectivity. Based on the data gathered from Table 7, it was observed that when the laptop was connected via Wi-Fi, the average speeds for detecting liver tumors, liver cysts, liver abscesses, and normal liver were 5.26 seconds, 5.78 seconds, 5.72 seconds, and 5.52 seconds, respectively, resulting in an overall average speed of 5.57 seconds. Conversely, when the laptop operated on mobile data, the average speeds increased to 12.41 seconds for liver tumors, 12.73 seconds for liver cysts, 12.34 seconds for liver abscesses, and 13.60 seconds for normal liver, culminating in an overall average speed of 12.77 seconds.

TABLE 6. COMPARISON OF SPEED OF AUTOML IN LAPTOP BETWEEN WI-FI AND MOBILE DATA CONNECTIVITY

Tested Variables	Statistic (t-test)	p-value	Decision	Conclusion
Tumor	-8.59	0.000	Reject Null	HAS a Significant Difference
Cyst	-19.28	0.000	Reject Null	HAS a Significant Difference
Abscess	-13.22	0.001	Reject Null	HAS a Significant Difference
Normal	-11.26	0.000	Reject Null	HAS a Significant Difference

Table 6 summarizes the comparison analyses which tested if there are significant differences in speed of detecting focal liver lesions between wireless fidelity connection and mobile data. Independent samples t-test indicated significant differences in speed in detecting tumors ($t = -8.59$, $p = 0.000 < 0.01$), cysts ($t = -19.28$, $p = 0.000 < 0.01$), abscess ($t = -13.22$, $p = 0.001 < 0.01$), and normal status livers ($t = -11.26$, $p = 0.000 < 0.01$).

Null Hypothesis (H_0):

There is no significant difference in speed in detecting tumor, cyst, abscess, and normal livers between Wi-Fi and Mobile Data.

Alternative Hypothesis (H_a):

There is significant difference in speed in detecting tumor, cyst, abscess, and normal livers between Wi-Fi and Mobile Data.

Decision:

Thus, all the tested hypotheses are rejected. As a result, the utilization of a laptop for detecting FLLs using mobile data and Wi-Fi access indicated a significant difference. Therefore, it can be concluded that when comparing the laptop's speed in detecting FLLs using Wi-Fi versus mobile data, Wi-Fi provides a faster detection speed than mobile data.

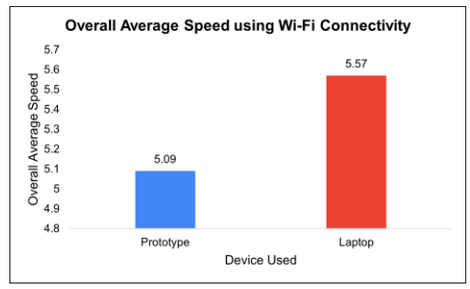

FIG. 25 SPEED TEST RESULT OF PROTOTYPE AND LAPTOP IN DETECTING FLLS USING WI-FI

Figure 25 shows the average speed results for each class using a prototype and a laptop, both utilizing Wi-Fi connectivity. Based on the gathered data in Tables 6 and 7, the prototype achieved an overall average speed of 5.09 seconds with Wi-Fi connectivity, which is faster than the laptop's average speed of 5.57 seconds.

TABLE 7. COMPARISON OF SPEED OF AUTOML USING WI-FI CONNECTIVITY BETWEEN PROTOTYPE AND LAPTOP

Tested Variables	Statistic (t-test)	p-value	Decision	Conclusion
Tumor	-2.54	0.026	Reject Null	HAS a Significant Difference
Cyst	-2.20	0.097	Accept Null	NO Significant Difference
Abscess	-1.33	0.254	Accept Null	NO Significant Difference
Normal	-0.22	0.836	Accept Null	NO Significant Difference

Table 7 summarizes the comparison analyses which tested if there are significant differences in speed of detecting focal liver lesions between Prototype and Laptop on wireless fidelity connection. Using t-test, a significant difference in speed is found for tumor ($t = -2.54$, $p = 0.026 < 0.05$). Meanwhile, no significant differences are found for cyst ($t = -2.20$, $p = 0.097 > 0.05$), abscess ($t = -1.33$, $p = 0.254 > 0.05$), and normal livers ($t = -0.22$, $p = 0.836 > 0.05$).

Null Hypothesis (H_0):

There is no significant difference in speed in detecting tumor, cyst, abscess, and normal livers between prototype and laptop using Wi-Fi connectivity.

Alternative Hypothesis (H_a):

There is significant difference in speed in detecting tumor, cyst, abscess, and normal livers between prototype and laptop using Wi-Fi connectivity.

Decision:

Therefore, across most classes, there was no significant difference between employing the prototype or laptop when operating via Wi-Fi connection.

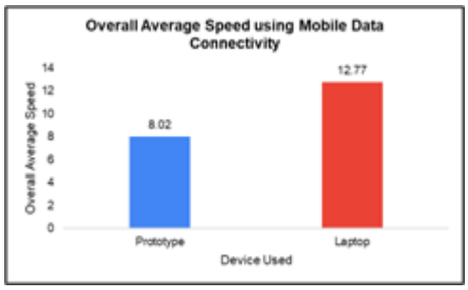

FIG. 26 SPEED TEST RESULT OF PROTOTYPE AND LAPTOP IN DETECTING FLLS USING MOBILE DATA

Figure 26 shows the average speed results for each class using a prototype and a laptop, both utilizing mobile data connectivity. Based on the gathered data in Tables 6 and 7, the prototype achieved an overall average speed of 8.02 seconds with mobile data connectivity, which is faster than the laptop's average speed of 12.77 seconds.

TABLE 8. COMPARISON OF SPEED OF AUTOML USING MOBILE DATA CONNECTIVITY BETWEEN PROTOTYPE AND LAPTOP

Tested Variables	Statistic (t-test)	p-value	Decision	Conclusion
Tumor	-5.58	0.000	Reject Null	HAS a Significant Difference
Cyst	-7.36	0.010	Reject Null	HAS a Significant Difference
Abscess	-3.70	0.033	Reject Null	HAS a Significant Difference
Normal	-7.53	0.003	Reject Null	HAS a Significant Difference

Table 8 summarizes the comparison analyses which tested if there are significant differences in speed of detecting focal liver lesions between Prototype and Laptop on mobile data connection. Independent samples t-test indicated significant differences for tumor ($t = -5.58$, $p = 0.000 < 0.01$), cyst ($t = -7.36$, $p = 0.010 < 0.05$), abscess ($t = -3.70$, $p = 0.033 < 0.05$), and normal livers ($t = -7.53$, $p = 0.003 < 0.01$).

Null Hypothesis (H_0):

There is no significant difference in speed in detecting tumor, cyst, abscess, and normal livers between prototype and laptop using Mobile Data connectivity.

Alternative Hypothesis (H_a):

There is no significant difference in speed in detecting tumor, cyst, abscess, and normal livers between prototype and laptop using Mobile Data connectivity.

Decision:

Thus, all the tested null hypotheses are rejected. Consequently, a significant difference emerged when employing either the prototype or laptop on mobile data, suggesting that the prototype outperformed the laptop in this regard.

System Performance Evaluation

The validation process assessed the prototype's accuracy in diagnosing FLLs using CECT scan images from six (6) patients – one (1) patient each FLL and three (3) patients for normal liver at JJASGH. Four validators – three gastroenterologists and one radiologist – independently reviewed the prototype's performance with ten (10) trials each classification.

TABLE 12. EVALUATION OF PROTOTYPE PERFORMANCE IN DETECTING FOCAL LIVER LESIONS

Criteria	Weighted Mean	Interpretation
Functionality and Sustainability	3.83	Effective
Reliability	3.88	Effective
Usability	4.08	Effective
Performance Efficiency	3.5	Effective
Maintainability	3.6	Effective
Security	4.45	Very Effective
Compatibility	4.25	Very Effective
Grand Mean	**3.9**	**Effective**

**Range of values - 4.21 – 5.00 = Very Effective; 3.41 - 4.20 = Effective; 2.61 - 3.40 = Moderately Effective; 1.81 - 2.60 = Ineffective; 1.00 - 1.80 = Very Ineffective*

Table 12 presents the evaluation results of the prototype's performance in detecting FLLs is conducted to reveal its effectiveness across multiple criteria. It demonstrates completeness, correctness, and appropriateness in functionality and sustainability, with a mean of 3.83. In terms of reliability, it excels in maturity, availability, and recoverability, and is highly effective in fault tolerance, with a mean of 3.88. Usability is highlighted by its recognition, learnability, and operability, with a mean of 4.08. Performance efficiency is noted in time-behavior and interoperability, with a of 3.5. Maintainability is ensured through modularity, reusability, and analyzability, with a grand mean of 3.6. Security features are robust, with high ratings for confidentiality, integrity, non-repudiation, accountability, and authenticity, achieving a of 4.45. Compatibility is demonstrated in effective co-existence, with a mean of 4.25. Overall, grand mean is 3.9 showcasing the prototype's effectiveness in various aspects potential for practical implementation.

VII. CONCLUSION

This study aimed to assess the speed and performance of an Automated Machine Learning (AutoML) system for image segmentation and classification of focal liver lesions (FLLs) on an NVIDIA Jetson Nano, compared to a laptop. The findings indicate that the Jetson Nano-equipped prototype, connected via Wi-Fi, outperformed the laptop in terms of speed for detecting FLL based on comprehensive speed tests conducted with both the prototype and the laptop using AutoML.

The evaluation demonstrated that the prototype's performance was not only fast but also accurate, as it consistently matched the official contrast-enhanced computed tomography (CECT) scan reports, producing true positive results. This indicates that the prototype's detection capabilities are reliable and precise.

Moreover, the prototype's software was evaluated using the ISO 25010 software quality model, which considers various criteria such as functionality, reliability, usability, efficiency, maintainability, security, and compatibility. The assessment revealed that the prototype performed effectively across all these dimensions. It was tested with six diagnosed CT images of patients, encompassing a total of 180 slices from institutional datasets at JJASGH, and the results were validated by a resident gastroenterologist.

REFERENCES

[1] S. R. Z. Abdel-Misih and M. Bloomston, "Liver anatomy," Surgical Clinics of North America, vol. 90, no. 4, pp. 643–653, 2010. doi:10.1016/j.suc.2010.04.017

[2] S. K. Venkatesh, V. Chandan, and L. R. Roberts, "Liver masses: A clinical, radiologic, and pathologic perspective," Clinical Gastroenterology and Hepatology, vol. 12, no. 9, pp. 1414–1429, 2014. doi:10.1016/j.cgh.2013.09.017

[3] Z. Sheikh and A. Nelson, "Liver lesions: Symptoms, Causes, Treatment, and More," WebMD, https://www.webmd.com/hepatitis/liver-lesions.

[4] R. W. Laing and D. F. Mirza, "Liver lesions: types, risk factors, investigations and treatment," Trends in Urology & Men's Health, vol. 11, no. 6, pp. 20–25, 2020. doi:10.1002/tre.777

[5] R. Chojniak, "The diagnosis of focal liver lesions in pediatric patients," Radiologia Brasileira, vol. 53, no. 3, 2020. doi:10.1590/0100-3984.2020.53.3e3

[6] A. Alshaikhli and A. Al-Hillan, "Liver Cystic Disease", https://www.ncbi.nlm.nih.gov/books/NBK567739/.

[7] H. Akhondi and D. Sabih, "Liver Abscess," National Library of Medicine, https://www.ncbi.nlm.nih.gov/books/NBK538230/

[8] Bethesda, "LiverTox: Clinical and Research Information on Drug-Induced Liver Injury," National Institute of Diabetes and Digestive and Kidney Diseases, https://www.ncbi.nlm.nih.gov/books/NBK548322/ (accessed Sep. 25, 2023).

[9] A. Murphy, "CT Triple-Phase Liver (Protocol)," Radiopaedia, https://radiopaedia.org/articles/ct-triple-phase-liver-protocol.

[10] A. Mustafa and M.R. Azghadi, "Automated Machine Learning for Healthcare and Clinical Notes Analysis," Computers, vol. 10, no. 2, pp. 24–24, Feb. 2021, doi: https://doi.org/10.3390/computers10020024.

[11] "Courses – NVIDIA," Nvidia.com, 2022. https://courses.nvidia.com/courses/course-v1:DLI+S-RX-02+V2/.

[12] ShawnHymel, "Getting Started with the NVIDIA Jetson Nano - Part 1: Setup," DigiKey, Dec. 10, 2019. https://www.digikey.ph/en/maker/projects/getting-started-with-the-nvidia-jetson-nano-part-1-setup/2f497bb88c6f4688b9774a81b80b8ec2.

[13] fionawarrings, "Get started with AI with the NVIDIA Jetson Nano," reichelt Magazin, Dec. 15, 2021. https://www.reichelt.com/magazin/en/projects/get-started-with-ai-with-the-nvidia-jetson-nano/.

[14] J. Waring, C. Lindvall, and R. Umeton, "Automated Machine Learning: Review of the State-of-the-Art and Opportunities for Healthcare," Artificial Intelligence in Medicine, vol. 104, p. 101822, Feb. 2020, doi: https://doi.org/10.1016/j.artmed.2020.101822.

[15] A. Biglari and W. Tang, "A review of embedded machine learning based on hardware, application, and sensing scheme," Sensors, vol. 23, no. 4, p. 2131, Feb. 2023. doi:10.3390/s23042131

[16] H. Technologies, "Image recognition with pytorch on the Jetson Nano," Medium, https://medium.com/@heldenkombinat/image-recognition-with-pytorch-on-the-jetson-nano-fd858a5686aa

[17] Huanga, "Performance evaluation of the nvidia jetson nano through a real-time machine learning application," SpringerLink, https://link.springer.com/chapter/10.1007/978-3-030-68017-6_51.

[18] P. S.K, S. A. Kesanapalli, and Y. Simmhan, "Characterizing the performance of Accelerated Jetson Edge devices for training deep learning models," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 6, no. 3, pp. 1–26, Dec. 2022. doi:10.1145/3570604

[19] A. K. Sahani, D. Srivastava, M. Sivaprakasam, and J. Joseph, "A machine learning pipeline for measurement of arterial stiffness in A-mode ultrasound," IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 69, no. 1, pp. 106–113, Jan. 2022. doi:10.1109/tuffc.2021.3109117

[20] M. S. Prabhu and S. Verma, "A deep learning framework and its implementation for diabetic foot ulcer classification," 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Sep. 2021. doi:10.1109/icrito51393.2021.9596380

[21] S. Baghersalimi, T. Teijeiro, D. Atienza, and A. Aminifar, "Personalized real-time federated learning for epileptic seizure detection," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 2, pp. 898–909, Feb. 2022. doi:10.1109/jbhi.2021.3096127

[22] M. Krüger, B. Vogel-Heuser, and S. Vollmann, "Investigating the rendering capability of embedded devices for graphical-user-interfaces in Mobile Machines," De Gruyter, https://doi.org/10.1515/auto-2023-0043 (accessed Mar. 17, 2024).

[23] E. Yose, V. Victor, and N. Surantha, "Portable smart attendance system on Jetson Nano," Bulletin of Electrical Engineering and Informatics, https://doi.org/10.11591/eei.v13i2.6061 (accessed Mar. 17, 2024).

[24] M. P.-L. Ooi et al., "Measurement and applications: Exploring the challenges and opportunities of hierarchical federated learning in Sensor Applications," IEEE Instrumentation & Measurement Magazine, vol. 26, no. 9, pp. 21–31, Dec. 2023. doi:10.1109/mim.2023.10328671

[25] L. Bautista, A. Abran, and A. April, "Design of a performance measurement framework for cloud computing," *Journal of Software Engineering and Applications*, vol. 05, no. 02, pp. 69–75, Jan. 2012. doi:10.4236/jsea.2012.52011

[26] Heldenkombinat Technologies, “Image recognition with PyTorch on the Jetson Nano - Heldenkombinat Technologies - Medium,” Medium, May 22, 2019. https://medium.com/@heldenkombinat/image-recognition-with-pytorch-on-the-jetson-nano-fd858a5686aa (accessed Jun. 09, 2024).

[27] Ross, A., & Willson, V. L. (1970, January 1). Independent Samples T-test. SpringerLink. https://link.springer.com/chapter/10.1007/978-94-6351-086-8_3

[28] Ertie Cusipag Abana, & Benedict Descargar Sy. (2022, November). ISO/IEC 25010 based evaluation of rice seed analyzer: a machine vision application using image processing... ResearchGate; Institute of Advanced Engineering and Science. https://www.researchgate.net/publication/364961963_ISOIEC_25010_based_evaluation_of_rice_seed_analyzer_a_machine_vision_application_using_image_processing_technique

[29] Hussain, A., & Mkpojiogu, E. (2015, November 5). *An application of the ISO/IEC 25010 standard in the quality-in-use assessment of an online health...* ResearchGate; Penerbit UTM Press. https://www.researchgate.net/publication/283767184_An_application_of_the_ISOIEC_25010_standard_in_the_quality-in-use_assessment_of_an_online_health_awareness_system

[30] Tacuban, T. (2020). Facial Recognition Using Segmented Facial Points Algorithm for Intelligent Surveillance System. *Asia Pacific Journal of Multidisciplinary Research*, *8*(2), 158–166. https://www.apjmr.com/wp-content/uploads/2020/04/APJMR-2020.8.2.19.pdf