Important declarations

Associated Data

Data supplied by the author:
Supplementary Data

Required Statements

Competing Interest statement:
The authors declare that they have no competing interests.

Funding statement:
The authors received no funding for this work.

Empowering Vision: Smart Text Translation and Object Identification for the Visually Impaired

Suresh Merugu Corresp., [1] , **Gavin Teo Siu Chyi** [1] , **Himani Maheshwari** [2]

[1] Computer Science, University of Southampton Malaysia, Iskander Puteri, Johor, Malaysia

[2] Computer Science, Graphic Era Hill University, Dehradun, Uttarakhand, India

Corresponding Author: Suresh Merugu
Email address: s.merugu@soton.ac.uk

Visually impaired individuals encounter substantial challenges when attempting to perform everyday tasks independently. Despite technological advancements, effective assistance remains elusive, and a comprehensive device meeting all needs has yet to emerge in the market. This paper proposes an innovative assistive device that harnesses the power of computer vision technology, specifically leveraging TensorFlow Lite and OpenCV, in combination with text recognition capabilities through Pytesseract. The system's objective is to provide audio output in various languages using the googletrans and gtts libraries in Python. This comprehensive approach holds significant promise as a guiding tool for visually impaired individuals, empowering them to navigate daily tasks with enhanced independence and confidence. By utilizing the capabilities of computer vision and text recognition, the proposed device can interpret visual information from the environment, such as objects like chairs, television sets, and printed text. It then converts this information into auditory cues, enabling real-time guidance and informed decision-making for visually impaired individuals. The integration of these technologies aims to address critical challenges faced by this community and enhance their overall quality of life. Through this innovative approach, the proposed system seeks to provide a multi-faceted solution that significantly improves the autonomy and daily experiences of visually impaired individuals.

Empowering Vision: Smart Text Translation and Object Identification for the Visually Impaired

Suresh Merugu [1*], Gavin Teo Siu Chyi[1], and Himani Maheshwari[2]

[1] Department of Computer Science, School of Computer Science, University of Southampton Malaysia, 79100 Iskandar Puteri, Johor, Malaysia.
*Corresponding Author e-mail address: s.merugu@soton.ac.uk
[2] School of Computing, Graphic Era Hill University, Dehradun, India.

Abstract: Visually impaired individuals encounter substantial challenges when attempting to perform everyday tasks independently. Despite technological advancements, effective assistance remains elusive, and a comprehensive device meeting all needs has yet to emerge in the market. This paper proposes an innovative assistive device that harnesses the power of computer vision technology, specifically leveraging TensorFlow Lite and OpenCV, in combination with text recognition capabilities through Pytesseract. The system's objective is to provide audio output in various languages using the googletrans and gtts libraries in Python. This comprehensive approach holds significant promise as a guiding tool for visually impaired individuals, empowering them to navigate daily tasks with enhanced independence and confidence. By utilizing the capabilities of computer vision and text recognition, the proposed device can interpret visual information from the environment, such as objects like chairs, television sets, and printed text. It then converts this information into auditory cues, enabling real-time guidance and informed decision-making for visually impaired individuals. The integration of these technologies aims to address critical challenges faced by this community and enhance their overall quality of life. Through this innovative approach, the proposed system seeks to provide a multi-faceted solution that significantly improves the autonomy and daily experiences of visually impaired individuals.

Keywords—Text to Speech, Computer Vision, Raspberry Pi 4B, Sensors, OpenCV.

1. Introduction

The human body is a fascinating work of art, each body part working in unison to achieve various processes such as energy metabolism, reproduction, cell differentiation and gene expression, digestion, etc. One fundamental biological function is physical movement, these include speech and articulation, reflexive movement, and locomotion and these processes require sensors to be carried out. For instance, the ears pickup sounds from the surroundings, the eyes pickup visual information, the skin detects temperature changes, the nose detects chemical odours, and the tongue perceives taste. Therefore, any damage to these organs will lead to impaired function, this can result in mild inconvenience to a full disability depending on the organ and severity. Among all these organs, the impairment of the eyes is the most common according to the World Health Organization [1] with at least 2.2 billion individuals who are affected globally. The impact of visual impairment ranges from difficulty with daily tasks such as walking, writing, reading, social prejudice like isolation, and emotional/psychological effects such as depression, low self-esteem, and anxiety. Therefore, the goal of this work is to develop a wearable integrated system that would aid visually impaired individuals in detecting objects and reading text through computer vision, with audio feedback for the user.

A. History of Computer Vision

In 1957, Russell Kirsh was an American computer scientist who developed the first digital image scanner with the help of his colleagues [2]. This device, which utilised a rotating drum with photodetectors to capture images, marked a significant advancement in the field of digital imaging. Later, an American engineer named Lawrence Roberts published his Ph.D. thesis “Machine Perception of Three-Dimensional Solids” in 1963 [3]. Roberts explained the process of deriving 3D information about solid objects from 2D photos and the steps involved in converting a 2D structure into a 3D one. His work laid the groundwork for future computer vision development alongside the invention of the neocognitron that was proposed by Kunihiko Fukushima in 1979 which is a multi-layered artificial neural network, this would be the precursor to the modern convolutional neural network (CNN) [4]. Eventually in 2001, two MIT researchers, Paul Viola and Michael Jones, created the first facial detection framework that utilised Adaboost as its learning algorithm [5]. However, from 2001 onwards, there were several notable advancements in computer vision, such as the emergence of AlexNet in 2012, which outperformed previous methods like sparse-coding and Scale-Invariant Feature Transform with Fisher Vectors (SIFT + FVs) [6] and the introduction of the Region-based Convolutional Neural Networks (R-CNN) by Ross Girshick and his colleagues [7]. Eventually, improvements for R-CNN would come in the form of fast R-CNN [8] and faster R-CNN [9]. Soon other popular object detection would be invented like the Single Shot MultiBox Detector (SSD) [10] and You Only Look Once (YOLO) [11].

B. Basics of Computer Vision

The basic steps to achieve successful object detection are as follows: Firstly, image acquisition, image processing, feature extraction, pattern recognition and output respectively [12]. Image acquisition is the process of collecting visual data from a camera. Image processing alters visual data such that it enhances them for further processes through noise reduction, contrast enhancement and filtering, etc. Feature extraction involves identifying and extracting relevant patterns from the processed data such as edges, corners, texture, colour histogram, etc. Pattern recognition is the use of machine learning algorithms to detect patterns and classify them accordingly into different labels/categories such as car, person, chair, etc. The output would usually be a bounding box around one to multiple objects and/or any custom output depending on how the application was designed.

C. Applications of Computer Vision

With computer vision achieving such rapid development, many countries have been utilising them in various sectors. For example, surveillance and security, computer vision is used to maintain constant surveillance on citizens while keeping logs on suspicious activities or wanted individuals. This is done through facial recognition where it scans for facial features of an individual and finds matches in a database. Another example would be autonomous vehicles and aerial vehicles, this is where autonomous vehicles use cameras, sensors and liDAR systems to perceive and respond accordingly to the environment, thus enabling them to navigate safely with little to no human intervention [13]. Aerial vehicles such as unmanned combat aerial vehicles (UCAV) use computer vision to perform military and non-military functions like reconnaissance, bombardment, target acquisition and tracking, to environmental monitoring, policing, and package delivery [14], perhaps future warfare only requires the use of robots only. Lastly, Agriculture, where it can be used to sort various produce, detect conditions of produce (any signs of disease or ripeness) and remote monitoring by farmers [15].

D. Challenges

In Malaysia, there is approximately 415,000 visually impaired individuals in 2023 according to [16]. The assistive device available for visually impaired individuals currently includes such as canes, smart canes, magnifying glass, guide dog, smart readers, and braille. However, they are not without flaws. Canes can only vaguely sense what objects are in front of the user waist down, similarity smart canes emit ultrasonic waves to detect what objects are in front with range of 2 to 4 meters. However, it doesn't specifically tell what objects are present. Magnifying glasses work to some extent by enlarging words or objects for seeing but the effectiveness of it would depend on the distant of the object and the severity of the user's visually impairment. Guide dogs only guides the owner from one destination to another. Smart reader are devices having a similar size to a small tablet, they help visually impaired individuals to read by magnifying the text underneath it. Lastly braille, it's a tactile writing system used by people who are visually impaired. It is based on a series of raised dots arranged in specific patterns within a grid of six dots, allowing users to read with their fingertips. It's often available on elevators and ATMs. However, each of those devices could only achieve one functions only and trying to incorporation everything together would be impractical and troublesome.

2. Previous Work

There are plenty of projects whose objective is to help the visually impaired using embedded systems. For example, a project conducted by [17], developed an assistive system that utilised Raspberry Pi 4, Pi camera along with other software like PyAudio to provide individuals who are visually impaired to describe what their surroundings are like. Furthermore, [18] used a robot instead of a minicomputer to perform text detection where several items were presented in front of the robot and it picked up audio from the user, the items closest to the audio will be pointed at by the robot. [19] designed an experiment which compared various object detection models and dataset to determine which combination performed the best, which would be MS COCO with mobilenet_v1 but this experiment did not use any minicomputers. Lastly, [20] developed an IoT device capable of object detection as well as currency recognition using Raspberry Pi and Pi camera. All the data collected during the operation of the device were sent to a remote server for analysis. [21] published a work using a Raspberry Pi and a Pi camera in 2022. Their main objective was facial recognition using Raspberry Pi and its camera to provide authorization (security).

[22] successfully created a text to speech system that also used a Raspberry Pi, Pi camera and an artificial neural network which is used for text prediction and the output is sent through headphones to the user. Another work conducted by [23] developed a smart reader that's able to convert written and printed text into speech. Furthermore, [24] developed a voice-assisted text reading system for visually impaired individuals. However, it uses Arduino as its processing unit instead of Raspberry Pi, and the camera is attached to the users' finger as this could help visually impaired users to instinctively point to the text that they desired to read and understand. Lastly, the study conducted by [25] explored the use of Raspberry Pi technology in developing a wearable smart cap for individuals with visual impairments. Unlike

previous studies, this showed the most promise for a cost-effective and efficient solution to assist visually impaired individuals as users would instinctively turn their heads to the object they wish to identify.

Limitations of Previous Work

Table 1 shows the limitations of the previous works, most of them were stated by the authors while other have be inferred and some were not suggested.

The development of the smart assistive device consists of several hardware and software components to work synergistically. Firstly, the processor would be the Raspberry Pi 4 Model B Rev 1.2, it is responsible for processing data and executing various tasks such as text translation, object detection and text-to-speech through other modules like Tensorflow Lite, OpenCV, Pytesseract, googletrans, and gtts within the system.

In tandem with the processing capabilities of the Raspberry Pi 4 Model B Rev 1.2, the Pi Camera V2 assumes a crucial role by acquiring essential imagery data, including real-time videos and images, necessary for both object detection and text translation functionalities. For object detection, the Pi Camera V2 captures real-time images, which are then subjected to preprocessing steps like resizing and normalization. These processed images are subsequently fed into a deep learning model, efficientdet_lite0.tflite, implemented with TensorFlow Lite and OpenCV. This pre-trained model efficiently identifies objects within the images, providing bounding boxes along with class labels and confidence scores. The class labels are then translated into a designated language using the Google Translate API (googletrans), and the translated labels are converted into audio output through text-to-speech (TTS) functionality provided by gtts, enabling auditory feedback through earphones. Similarly, for text translation, the Pi Camera V2 captures images containing text, and optical character recognition (OCR) is performed using Pytesseract. By analyzing pixels and identifying character patterns, Pytesseract extracts text from the images, leveraging language-specific models and dictionaries to interpret the text patterns and convert them into machine-readable text. The extracted text is subsequently translated into the desired language using the Google Translate API, and the translated text is outputted as audio through text-to-speech functionality, again enabling auditory output via earphones. The entire system would be powered by a power bank and controlled wireless via VNC and hotspot provided by a smartphone. Figure 2 shows how the developed system would be implemented in the final version.

The Raspberry Pi 4 Model B, used as the main controller, has built-in wireless networking capabilities that facilitate its use as an outdoor IoT monitoring device without additional wireless connection components. Its greater processing power compared to microcontrollers like the Arduino UNO enables more complex data processing and analysis. While the Raspberry Pi 4 Model B is a relatively new model and has not been extensively explored for IoT-based weather monitoring systems, it does have a higher power consumption of 540mA (2.7W) when idle.

The developed system consists of several physical hardware, these include the Raspberry Pi, Pi Camera, cables, monitor, keyboard and mouse, earphone, smartphone, power bank and cables, and a headset. The official operating system for Raspberry PI is Raspbian GNU/Linux 11 (bullseye) and Python is the programming language used for the development of the program. TensorFlow Lite is a lightweight version of TensorFlow, an open-source machine learning framework developed by Google. TensorFlow Lite is specifically designed for mobile and embedded devices, so it’s well-suited for deployment on the Raspberry Pi. It enables efficient execution of machine learning models, allowing the Raspberry Pi to perform tasks such as object detection, image classification, and natural language processing.

OpenCV (Open-Source Computer Vision Library) is a widely used open-source library for computer vision tasks. It provides a comprehensive set of tools and algorithms for image processing, computer vision, and machine learning. OpenCV is utilised on the Raspberry Pi for tasks such as image capture, manipulation, feature detection, and object tracking. Pytesseract is a Python wrapper for Tesseract-OCR, an open-source optical character recognition engine developed by Google. It allows the application to extract text from images using OCR techniques. Pygame is a set of Python modules designed for writing video games and multimedia applications. However, in this case, it’s used to develop the graphical user interface (GUI) for the application. Furthermore, googletrans is a Python wrapper for the Google Translate API, which allows language translation capabilities into their applications. With googletrans, the application can easily translate text between different languages. Next, gtts (Google Text-to-Speech) is used Text-to-Speech once the detected text is translated. gtts is a Python library and CLI tool for converting text into speech. Lastly Python is the primary programming language used for developing applications and integrating these various libraries and tools on the Raspberry Pi.

Figure 2 shows the object detection and text translation framework which will be implemented for the proposed system. While Figure 3 shows the use case a user will encounter while using the proposed system.

2.1. Initial Setup and Object & Text Detection

Figure 4 shows how the initial setup was, it was in the early stages of testing results are shown in Figure 5 and 6.

The above figure shows the successful setup of the proposed system before integration into the headset.

The initial object detection shown in Figure 5 achieved decent results, being able to accurately detect most common objects such as chairs, tv, laptop, bottle, person, etc.

The initial text detection shown in Figure 6 also achieved decent results. However, this made the application to be extremely slow as it continuously captures and detects live footage thus making this not viable for real time text detection.

2.2. Implementation of GUI and Output

In the second iteration, there were 5 modifications made. Firstly, the introduction of a graphical user interface (GUI), the GUI allows the user to easily navigate between each function without the use of the terminal shown in Figure 7 below.

Second, is the inclusion of the "Language" button, which is shown at Figure 8, this enables the user to select five languages (English, Japanese, Hindi, Chinese, Malay) for text-to-speech for both object and text detection instead of only being for the text detection.

Third, a text-to-speech feature was added to both object and text detections as shown in Figure 9 and 10. The language output is determined by the choice made during language selection. After detecting objects or text, the program will translate the objects name's/text into the selected language.

2.3. Improvement of GUI and Language Update

Pytesseract does support text detection of multiple languages. However, in the current situation, photos would be taken from a camera and not from a static image, thus there are various factors that would affect the effectiveness of the text detection.

First, the figure above shows the level of exposure, this refers to the amount of light captured by the camera. The amount will affect the quality of a photo thus the chances of accurate detection. [26] discussed the optimization of image acquisition systems for autonomous driving and found that the mean average precision (mAP) of detection networks SSD-Mobilenet and Resnet drop with over or under exposure. Text detection through the camera will have different lighting thus different results.

Second, motion blur, this refers to the blurring effect observed in images captured while objects are in motion, this is caused by the object moving or the camera moving as shown in Figure 12. The resulting image capture would have a blurry effect, therefore there will be difficulty in detecting. This would be further exacerbated for text detection as trying to detect a group of small words while under motion blur makes it unreadable.

Third, language which uses Latin alphabet and language which doesn't require different OCR. In Figure 13, the language on the left all uses alphabets, each having the identical structure whereas on the right, each language has distinct curves, fonts, and styles. During testing, Japanese, Hindi, Chinese failed to be detected and if a detection were to occur, the original text would be detected as random gibberish and thus the translation and TTS would be also. Therefore, the language options would be changed to English, German, French, Spanish and Malay.

Furthermore, a new menu has been created for better user experience (Figure 14). Also, an OCR button was added, the OCR button allows users to choose what language they wish for text translation. The options are shown in Figure 15.

2.4. Developed System

For the last iteration, the raspberry pi, the camera, and earphones are integrated into a headset thus completing the entire system as shown in Figure 16.

3. Results and Discussion

3.1. Object detection and identification by using the developed system

For object detection, 20 items were used for testing these include chair, keyboard, mouse, laptop, television, backpack, cell phone, cup, scissors, person, potted plant, vase, bucket, card, ping pong bat, pencil case, toy van, wallet, and sign board. 15 out of 20 items were correctly classified while the rest were not. The correct ones include chair, keyboard, mouse, laptop, television, backpack, cell phone, cup, scissors, person, bottle, fork, spoon, potted plant, and vase. The following output in each figure consists of bounding boxes with confidence scores, names of each object detection according to the language selected and an audio output for each object's name according to the language selected. The output is shown from Figures 17 to 25.

Figures 26 to 30 shows the output when the system is used in an outdoors environment. It consists of four correct detection and one incorrect detection.

For text detection, each language is tested with different font styles, font colours and background. The font styles used include Aptos (body), Academy Engraved LET, and Brush Script MT. Aptos (body) is the standard font, Academy Engraved LET is the special font and Brush Script MT is the cursive font, all of which is shown in Figure 31.

As for font colours and background, there will only be five combinations tested. White background with black font, black background with white font, yellow background with red font, grey background with blue font and blue background with green font. Therefore, with the combination of font styles, font colours and background colours, this will result in 15 unique combinations for each language thus ensuring robust testing scenarios. The following output of consist of the sample text, on the left and the output on the right.

First, starting with Aptos (body) font, for all languages, white background with black font, black background with white font, yellow background with red font could successfully be detected as shown from Figures 32 to 37.

Some translations directly translate so it might not convey the correct message to the user grammatically. For example, in Figure 38, the original text meant "Germany, Italy and Japan were best friends from 1939 to 1945" but in the translation it became "Germany, Italy and Japan were from 1939 to 1945 best friends" which is still technically correct.

Another direct translation is shown in Figure 39, the original text meant "Does nasi lemak come from Singapore or Malaysia? what do you think?" in Bahasa Malay but the result was "Do fat rice come from Singapore or Malaysia? what do you think?" But nasi lemak (rice cooked with coconut milk and pandan) is directly translated to fat rice, lemak is fat in Bahasa Malay.

For grey background with blue font and blue background with green font, the system had difficulty detecting the displayed text regardless of the angles used or lighting conditions. The grey background with blue font often had incomplete or incorrect translation while for blue backgrounds with green font, it's always undetectable to the system as shown in Figures 40 and 41.

Next, for Academy Engraved LET, it could successfully detect white background with black font, black background with white font is fine (Figures 42 & 43). However, starting from yellow background and red font (Figure 44) and beyond (Figures 45 & 46), nothing else can be detected successfully regardless of the language used.

Lastly, for Brush Script MT, successful detection was only seen for English, Malay, and Spanish for white background with black font, black background with white font (Figures 47 & 48). Any other combination would result in complete gibberish which is shown in Figures 49, 50 and 51.

3.2. Evaluation

The system is able to detect most common objects but not for items which are uncommon/rare such as voltmeter, microscope, jackhammer, etc. However, this doesn't affect the quality of the system as the main objective is aiding the visually impaired which only includes general object detection only. In terms of text detection, 31 out of 75 combinations were successfully detected and translated. Most of the failures for text detection were attributed to grey background with blue font and blue background with green font as well as the cursive font.

There are various reasons why such background and font colours cause such difficulty for text detection. Firstly, colour similarity, blue and green are adjacent colours on the colour spectrum making them relatively close in terms of hue. This similarity can cause the green text to blend into the blue background, reducing the colour contrast between the text and the background. Grey and blue on the other hand are not adjacent colours on the colour spectrum, they can still share similarities in hue, this similarity can cause the blue text to blend into the grey background, making it harder for the system to differentiate the text from the background based on colour alone. Secondly is lack of contrast, blue and green, grey, and blue doesn't provide enough contrast for clear visibility. Text detection algorithms often rely on detecting significant differences in colour intensity or brightness between the text and the background. As for the cursive font, it causes the structure of the sentence to change and distort such that the system doesn't recognise it as actual words but symbols or punctuation thus resulting in faulty translation.

4. Conclusion

To conclude this paper, the objective of this developed system is to build a wearable assistive device for visually impaired individuals with capabilities for object detection and text translation with audio output. The developed system employs a Raspberry Pi 4 Model B 1.2 as the core computer and a Pi camera V2 for video and image acquisition. Upon obtaining the data, Pytesseract is utilized for text translation to extract the character pattern from an image, thereby determining the actual text. The detected text is then translated into a target language based on the language selected in lang.txt, followed by gtts performing text-to-speech to notify the user. For object detection, the camera continuously captures live video while identifying objects in real-time. The objects are processed through a pre-trained model EfficientDet_lite0.tflite to determine their labels. These labels are then translated similarly to text translation and converted to speech for user notification.

In conclusion, the developed system provides a robust foundational platform for an assistive device aimed at the visually impaired. It encompasses essential functions such as reading, object identification, and audio output, thereby restoring a degree of independence to visually impaired individuals. This integration of various technologies demonstrates significant potential in enhancing the quality of life for visually impaired users, making everyday tasks more manageable and fostering greater autonomy. The developed system is not without flaws. The usability could be made more user-friendly for visually impaired individuals. For example, the implementation of voice recognition, would enable those who are severely visually impaired to navigate the application through their voice instead of mouse and keyboard. This could be achieved with a microphone and some simple python code. Furthermore, the system could include a wider selection of languages such as Russian, Korean, Tamil, etc. to ensure a large range of translation and detection thus making the system more well-rounded. Furthermore, implementation of a custom dataset, enables future systems to be able to detect wider selection of objects

5. Acknowledgment

The authors would like to thank the department of Computer Science, Connected Intelligence Research Group (CIRG) and RMC, University of Southampton Malaysia, 79100 Iskandar Puteri, Johor, Malaysia.

6. References

[1] World Health Organization (2023). Blindness and Vision Impairment. [online] World Health Organization. Available at: https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

[2] First Digital Image. (2022). NIST. [online] Available at: https://www.nist.gov/mathematics-statistics/first-digital-image.

[3] Roberts, L.G. (1963). Machine perception of three-dimensional solids. [online] dspace.mit.edu. Available at: https://dspace.mit.edu/handle/1721.1/11589.

[4] Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics, 36(4), pp.193–202. doi:https://doi.org/10.1007/bf00344251.

[5] Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 1. doi:https://doi.org/10.1109/cvpr.2001.990517.

[6] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Communications of the ACM, [online] 60(6), pp.84–90. doi:https://doi.org/10.1145/3065386.

[7] Gupta, S., Arbeláez, P., Girshick, R. and Malik, J. (2014). Indoor Scene Understanding with RGB-D Images: Bottom-up Segmentation, Object Detection and Semantic Segmentation. International Journal of Computer Vision, [online] 112(2), pp.133–149. doi:https://doi.org/10.1007/s11263-014-0777-6.

[8] He, K., Gkioxari, G., Dollar, P. and Girshick, R. (2018). Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.1–1. doi:https://doi.org/10.1109/tpami.2018.2844175.

[9] Ren, S., He, K., Girshick, R. and Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. [online] arXiv.org. Available at: https://arxiv.org/abs/1506.01497.

[10] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. and Berg, A.C. (2016). SSD: Single Shot MultiBox Detector. Computer Vision – ECCV 2016, [online] 9905, pp.21–37. doi:https://doi.org/10.1007/978-3-319-46448-0_2.

[11] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. [online] arXiv.org. Available at: https://arxiv.org/abs/1506.02640.

[12] www.sama.com. (n.d.). Computer Vision: History and How it Works | Sama. [online] Available at: https://www.sama.com/blog/computer-vision-history-how-it-works#:~:text=The%20History%20of%20Computer%20Vision.

[13] Cortés, I., Beltrán, J., de la Escalera, A. and García, F. (2023). Joint Object Detection and Re-Identification for 3D Obstacle Multi-Camera Systems. Sensors, [online] 23(23), p.9395. doi:https://doi.org/10.3390/s23239395.

[14] Mohsan, S.A.H., Khan, M.A., Noor, F., Ullah, I. and Alsharif, M.H. (2022). Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review. Drones, [online] 6(6), p.147. doi:https://doi.org/10.3390/drones6060147.

[15] Tian, H., Wang, T., Liu, Y., Qiao, X. and Li, Y. (2020). Computer vision technology in agricultural automation —A review. Information Processing in Agriculture, [online] 7(1), pp.1–19. doi:https://doi.org/10.1016/j.inpa.2019.09.006.

[16] BERNAMA (2023). - IS BRAILLE STILL RELEVANT? [online] BERNAMA. Available at: https://www.bernama.com/en/thoughts/news.php?id=2154357#:~:text=In%20a%20world%20that%20depends [Accessed 18 Apr. 2024].

[17] Liu, S., Xu, H., Li, Q., Zhang, F. and Hou, K. (2021). A Robot Object Recognition Method Based on Scene Text Reading in Home Environments. Sensors, 21(5), p.1919. doi:https://doi.org/10.3390/s21051919.

[18] Wahab, F., Ullah, I., Shah, A., Khan, R.A., Choi, A. and Anwar, M.S. (2022). Design and implementation of real-time object detection system based on single-shoot detector and OpenCV. Frontiers in Psychology, 13. doi:https://doi.org/10.3389/fpsyg.2022.1039645.

[19] Rahman, Md.A. and Sadi, M.S. (2021). IoT Enabled Automated Object Recognition for the Visually Impaired. Computer Methods and Programs in Biomedicine Update, 1, p.100015. doi:https://doi.org/10.1016/j.cmpbup.2021.100015.

[20] Islam, R., Iqbal, F., Akhter, S., Rahman, M. and Khan, R. (2023). Deep learning based object detection and surrounding environment description for visually impaired people. Heliyon, 9(6), pp.e16924–e16924. doi:https://doi.org/10.1016/j.heliyon.2023.e16924.

[21] Boxey, A., Jadhav, A., Gade, P., Ghanti, P. and Mulani, Dr.A.O. (2022). Face Recognition using Raspberry Pi. Journal of Image Processing and Intelligent Remote Sensing, (24), pp.15–23. doi:https://doi.org/10.55529/jipirs.24.15.23.

[22] K. S. P, A. A, G. K, 2022, "Raspberry Pi based Smart Assistance for Visually Impaired People," in: 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, pp. 1199-1204. DOI: 10.1109/ICESC54411.2022.9885412.

[23] Sarkar, S., Pansare, G., Patel, B., Gupta, A., Chauhan, A., Yadav, R., & Battula, N. (2021). Smart Reader for Visually Impaired Using Raspberry Pi. International Conference on Innovations in Mechanical Sciences (ICIMS'21), IOP Conf. Series: Materials Science and Engineering, 1132, 012032. doi:10.1088/1757-899X/1132/1/012032

[24] Sanjana, B., & RejinaParvin, J. (2016). Voice Assisted Text Reading System for Visually Impaired Persons Using TTS Method. IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), 6(3), 15-23. DOI: 10.9790/4200-0603031523.

[25] Kusuma, R., & Poornima, K. J. (2023). Raspberry Pi Based Implementation of Wearable Smart Cap for Visually Impaired Person. Eur. Chem. Bull., 12(8), 6000-6004.

[26] Blasinski, H., Farrell, J., Lian, T., Liu, Z. and Wandell, B. (2018). Optimizing Image Acquisition Systems for Autonomous Driving. Electronic Imaging, 30(5), pp.161–1161–7. doi:https://doi.org/10.2352/issn.2470-1173.2018.05.pmii-161.

Table 1(on next page)

Limitations of Previous Work

Empowering Vision: Smart Text Translation and Object Identification for the Visually Impaired

Suresh Merugu [1*], Gavin Teo Siu Chyi[1], and Himani Maheshwari[2]

[1] Department of Computer Science, School of Computer Science, University of Southampton Malaysia, 79100 Iskandar Puteri, Johor, Malaysia.
*Corresponding Author e-mail address: s.merugu@soton.ac.uk
[2] School of Computing, Graphic Era Hill University, Dehradun, India.

TABLE I. LIMITATIONS OF PREVIOUS WORK

Reference	Year	Limitations
25	2023	● No faster GPU. ● No Optical Character recognition and Image Analysis and better text recognition. ● No proximity sensors. ● No regional languages for better user experience.
18	2022	● Unable to detect actions such as writing, running, washing dishes. ● Model can't calculate object's velocity (moving car's speed)
22	2022	● Increase portability using smaller development board and camera.
21	2022	● Unable to detect actions such as writing, running, washing dishes.
23	2021	● No proximity sensors.
19	2021	● Include custom dataset for wider range of object detection
17	2021	● Add Natural Language Processing (NLP) and use more language datasets to improve audio feedback
24	2016	● Model can't calculate object's velocity (moving car's speed)

Figure 1

Block Diagram of Developed System

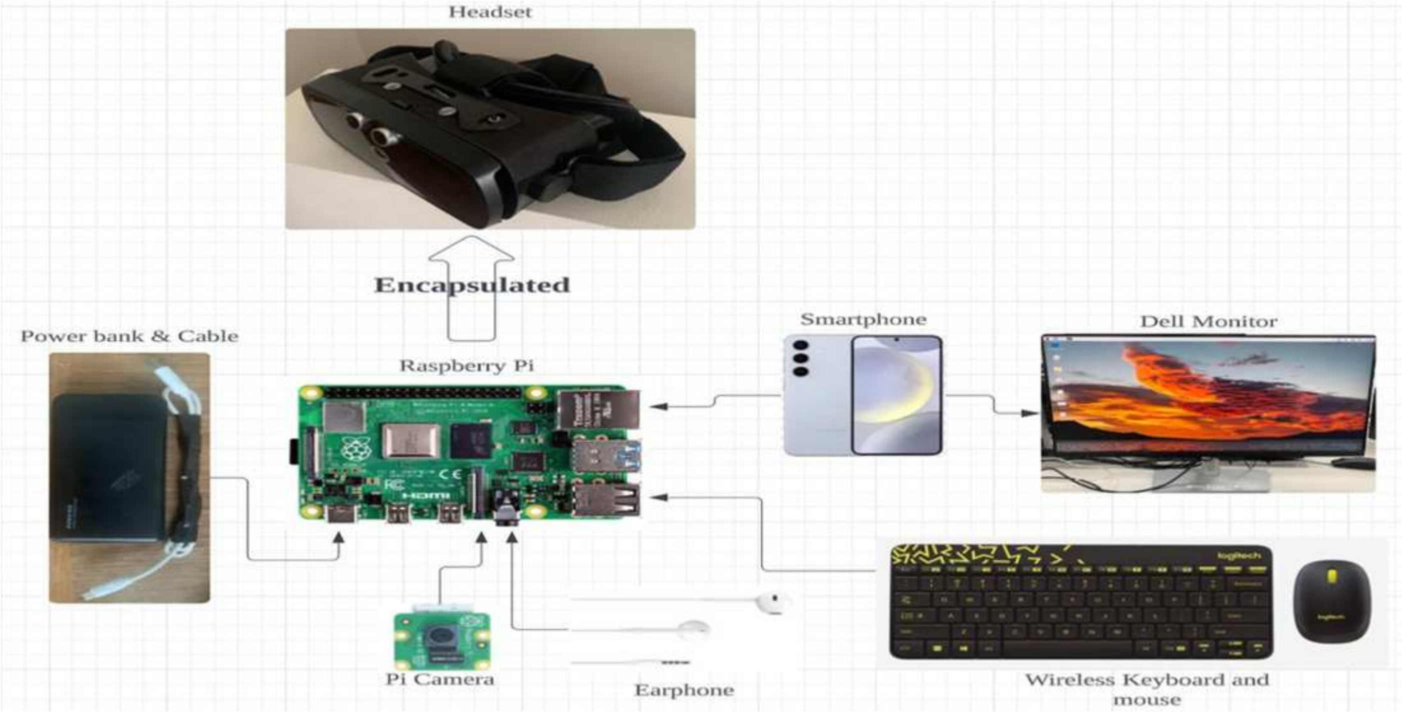


Figure 2

Object Detection & Text Translation Framework

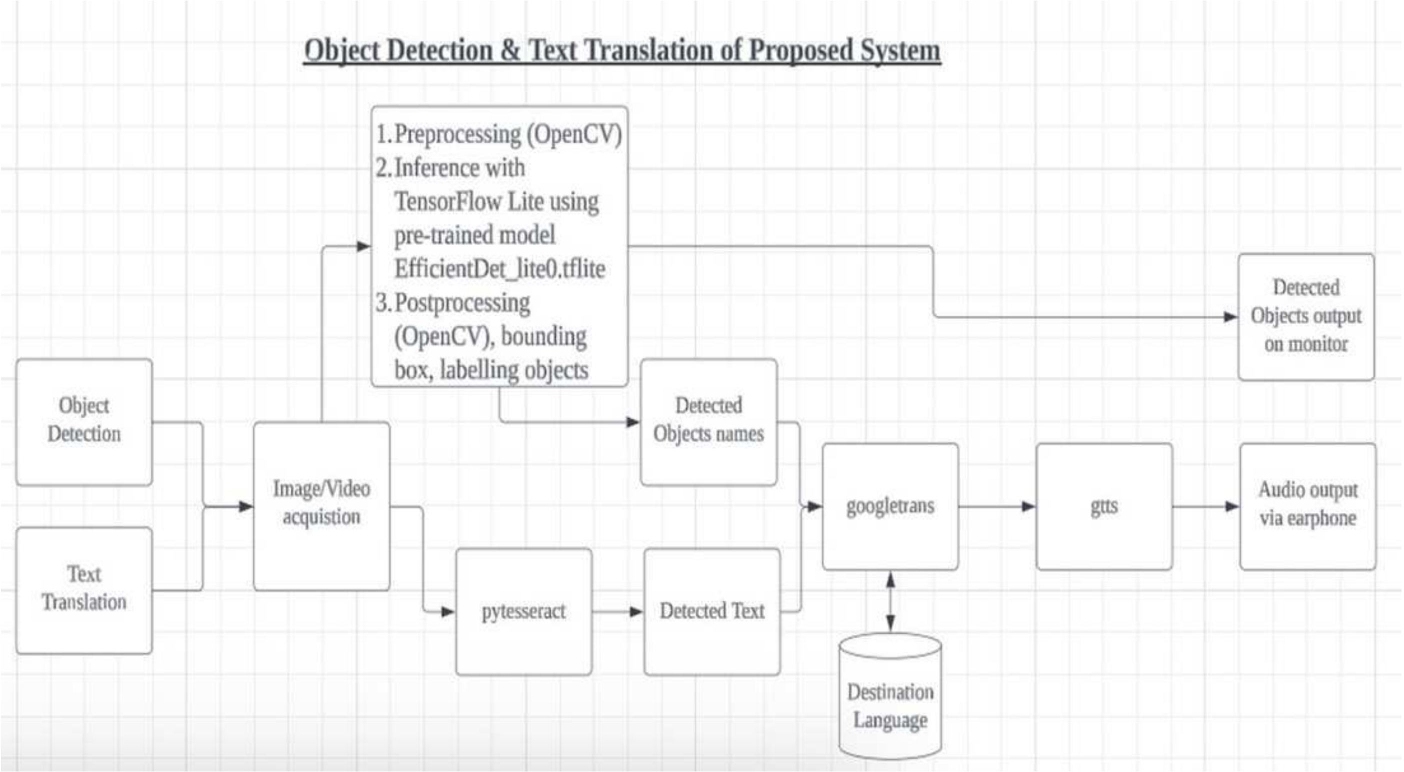

Object Detection & Text Translation of Proposed System
1. Preprocessing (OpenCV)
2. Inference with TensorFlow Lite using pre-trained model EfficientDet_lite0.tflite
3. Postprocessing (OpenCV), bounding box, labelling objects
Detected Objects output on monitor
Object Detection
Text Translation
Image/Video acquistion
Detected Objects names
googletrans
gtts
Audio output via earphone
pytesseract
Detected Text
Destination Language

Figure 3

Use Case Diagram for The Developed System

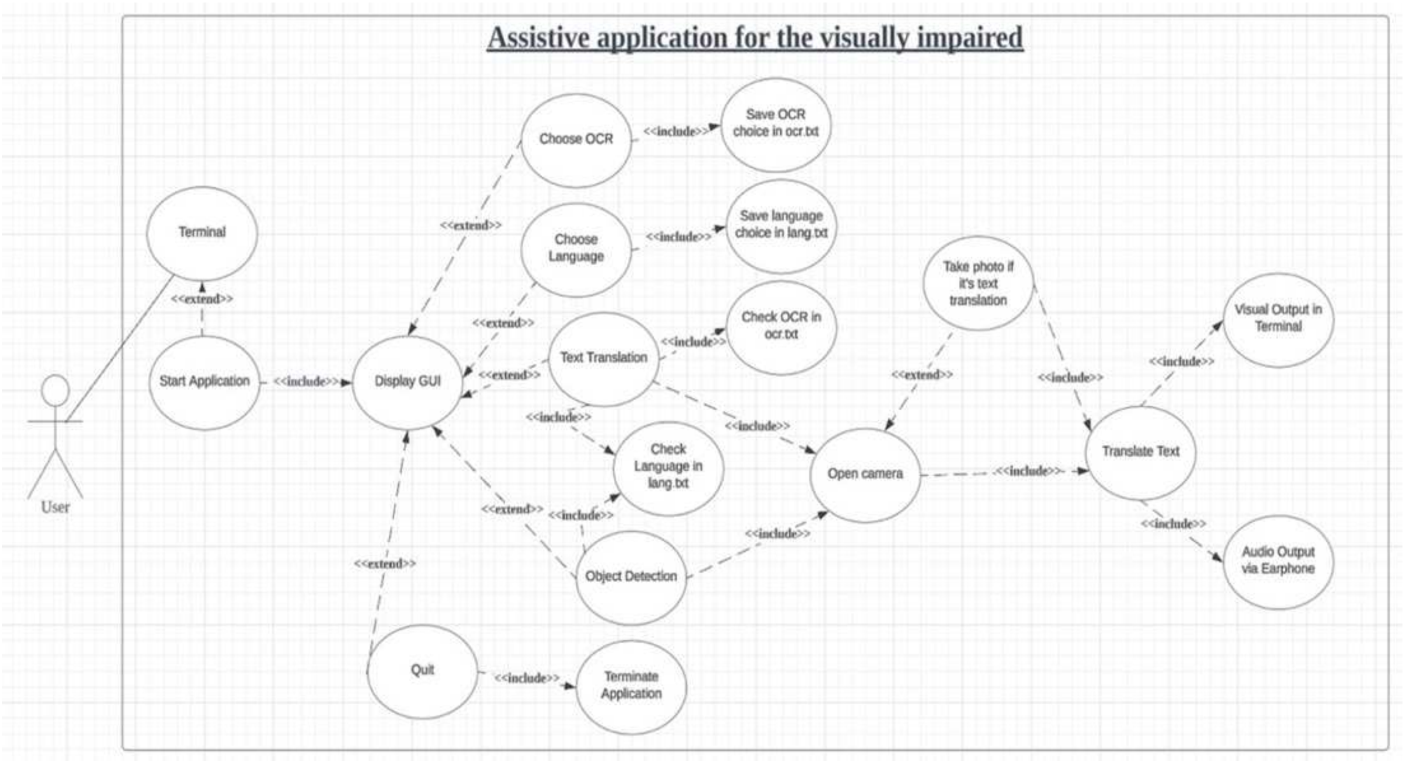


Figure 4

Initial Setup of the Developed System

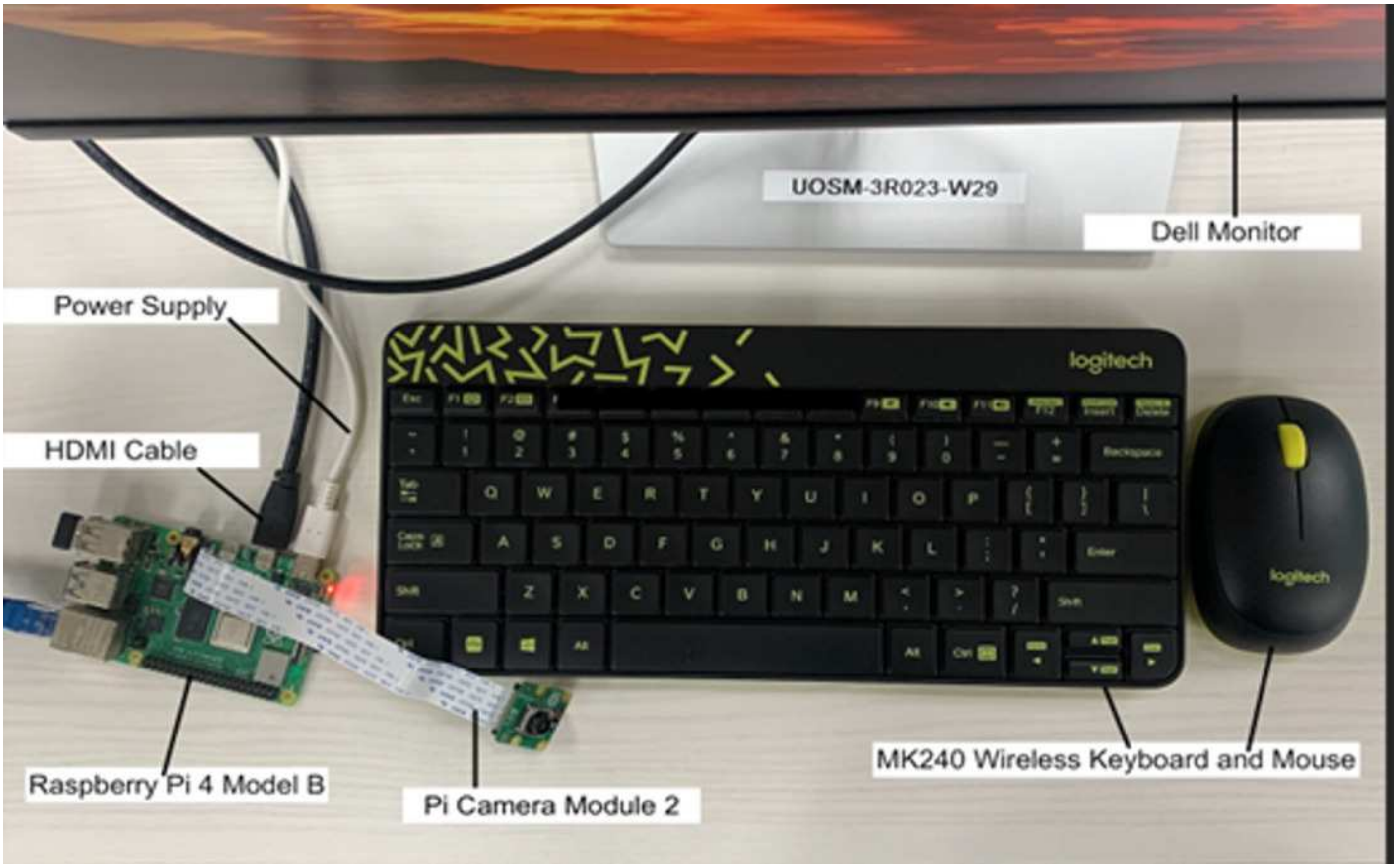
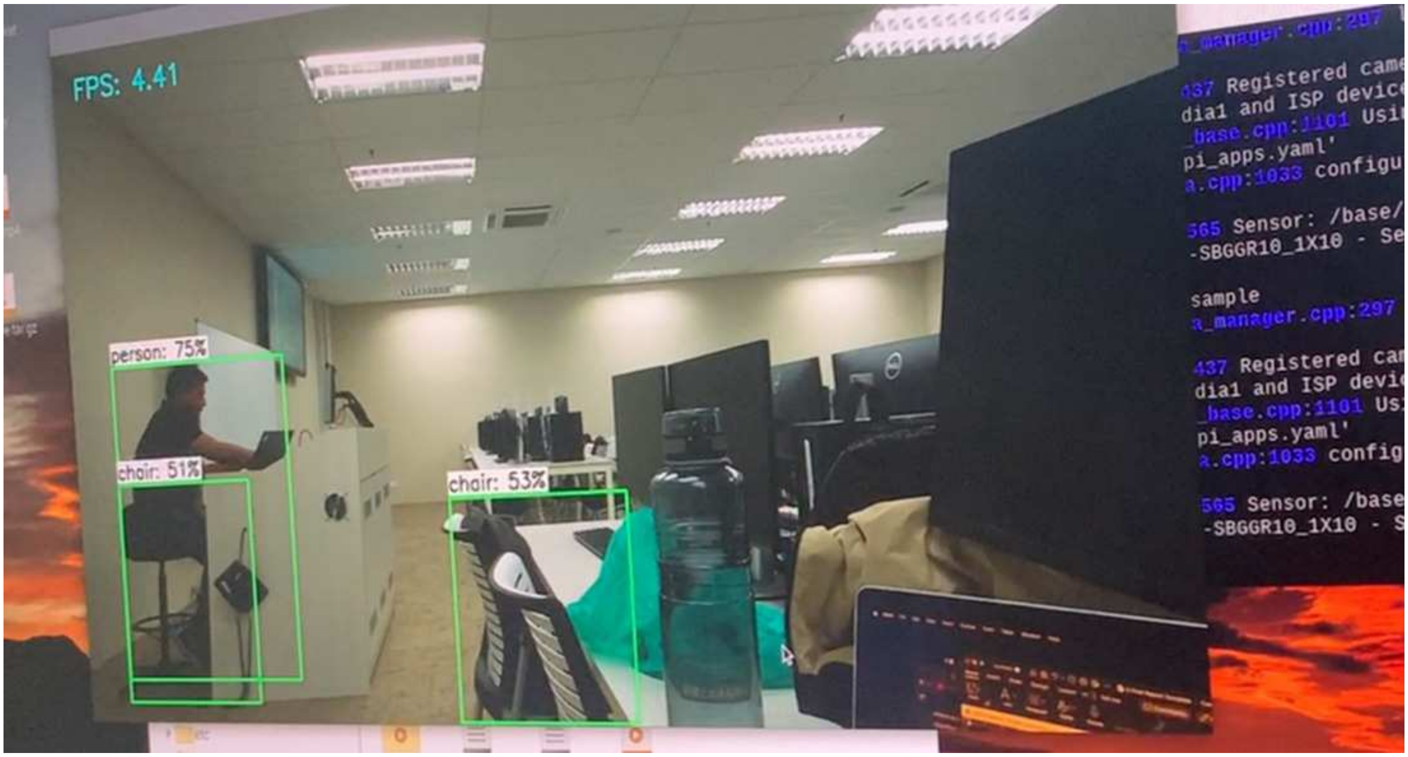


Figure 5

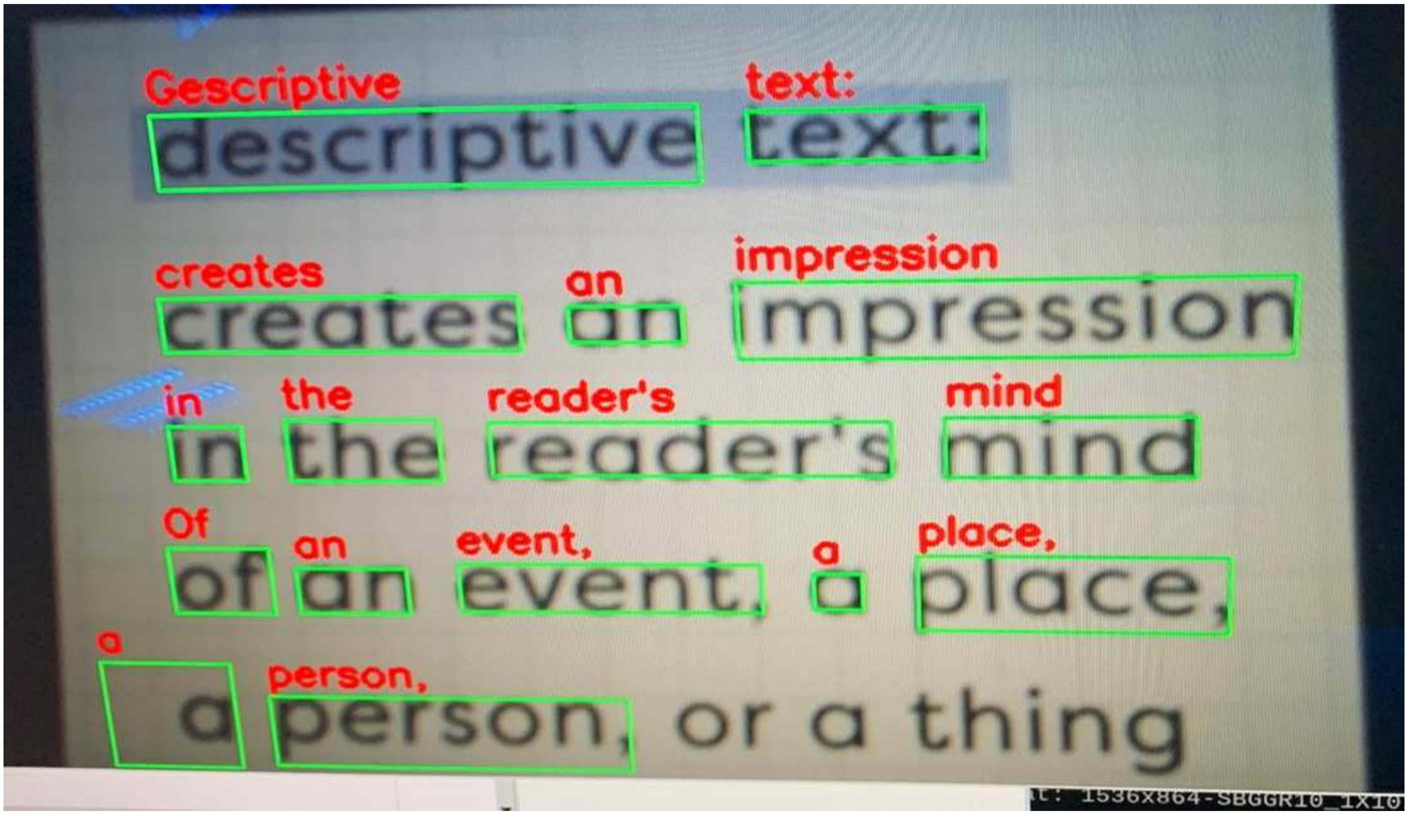
Initial Object Detection Testing



```
a_manager.cpp:297
437 Registered came
dia1 and ISP device
_base.cpp:1101 Usi
pi_apps.yaml'
a.cpp:1033 configu

565 Sensor: /base/
-SBGGR10_1X10 - Se

sample
a_manager.cpp:297

437 Registered ca
dia1 and ISP devic
_base.cpp:1101 Us
pi_apps.yaml'
a.cpp:1033 config

565 Sensor: /base
-SBGGR10_1X10 - S
```

Figure 6

Initial Text Detection Testing



descriptive text:
creates an impression
in the reader's mind
of an event, a place,
a person, or a thing

Figure 7

The GUI for object and text recognition

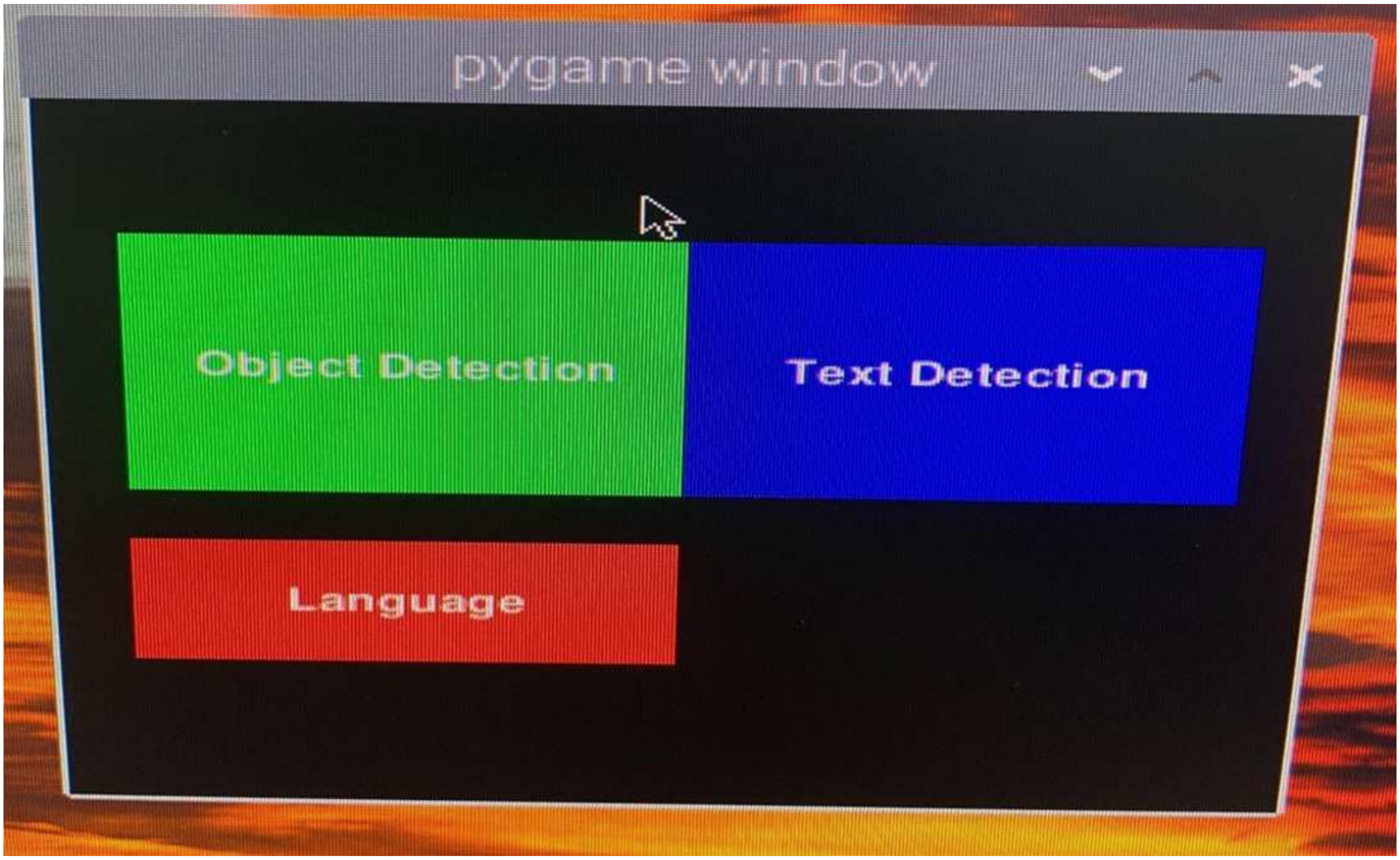


Figure 8

Language Selection Menu

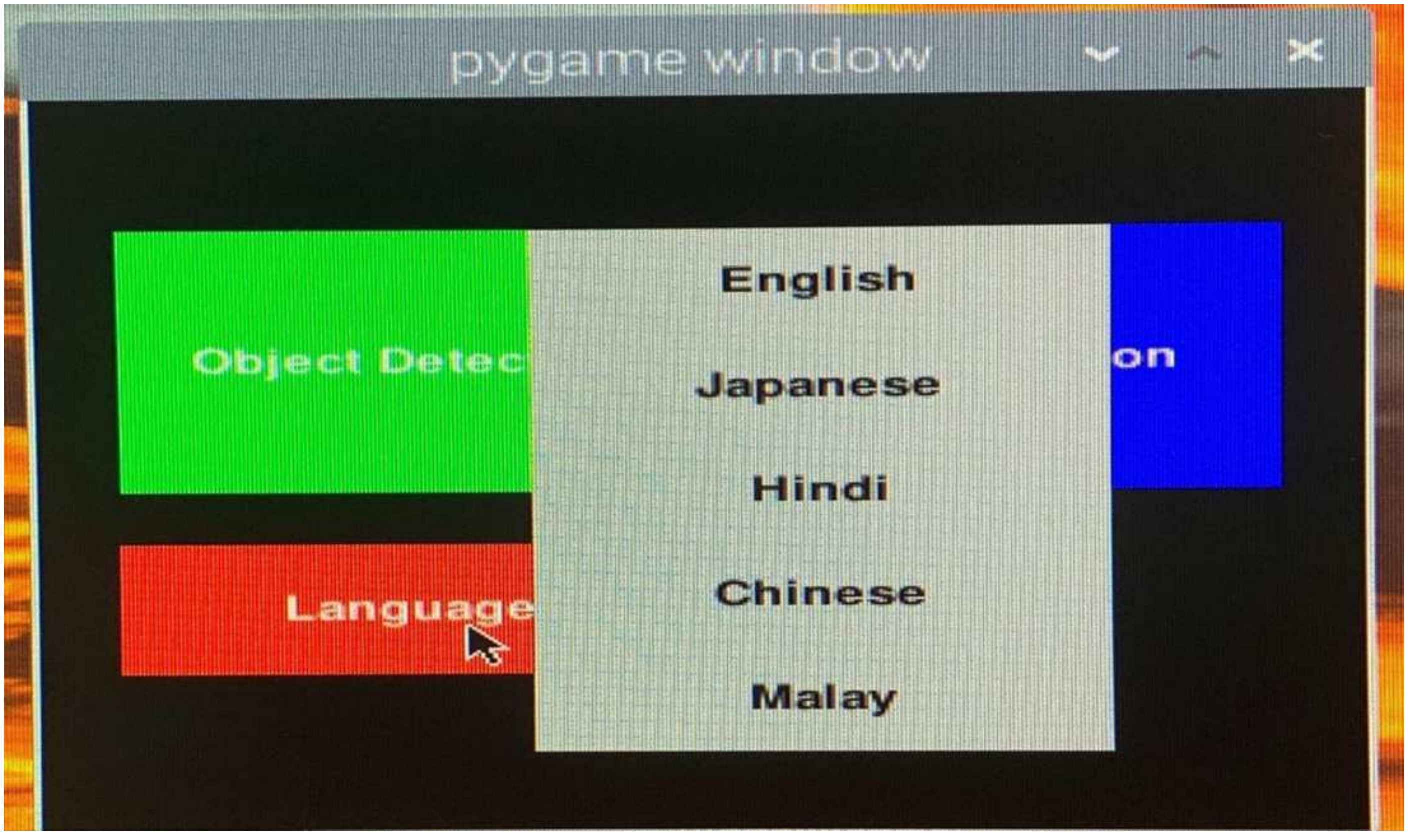
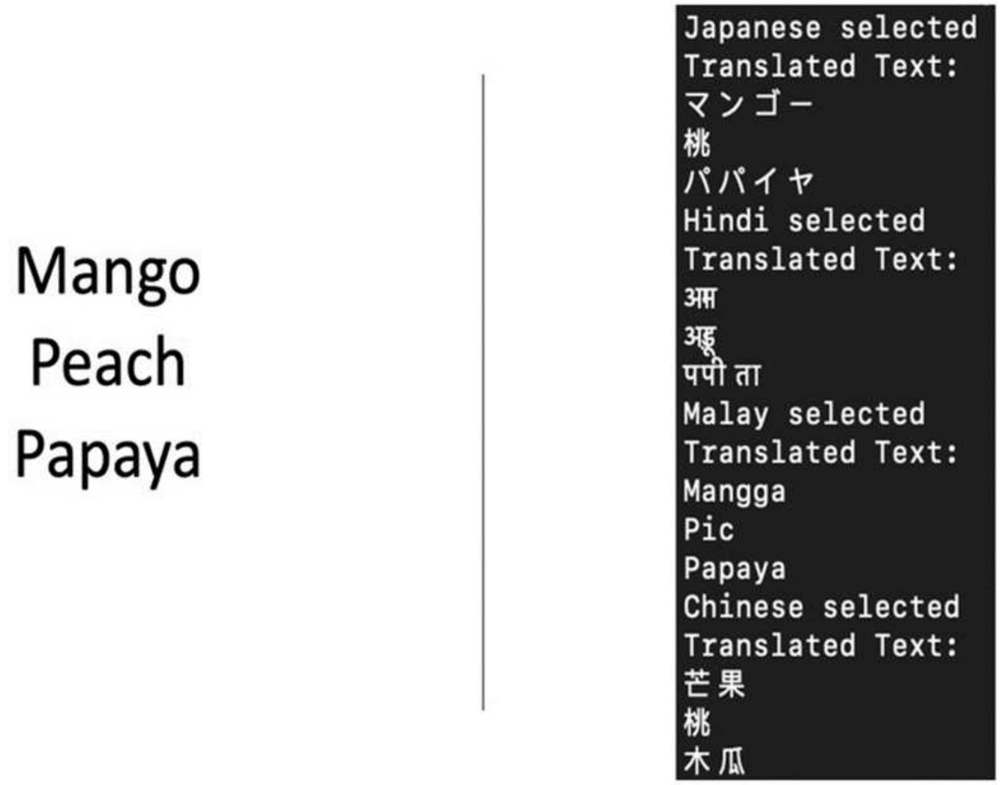


Figure 9

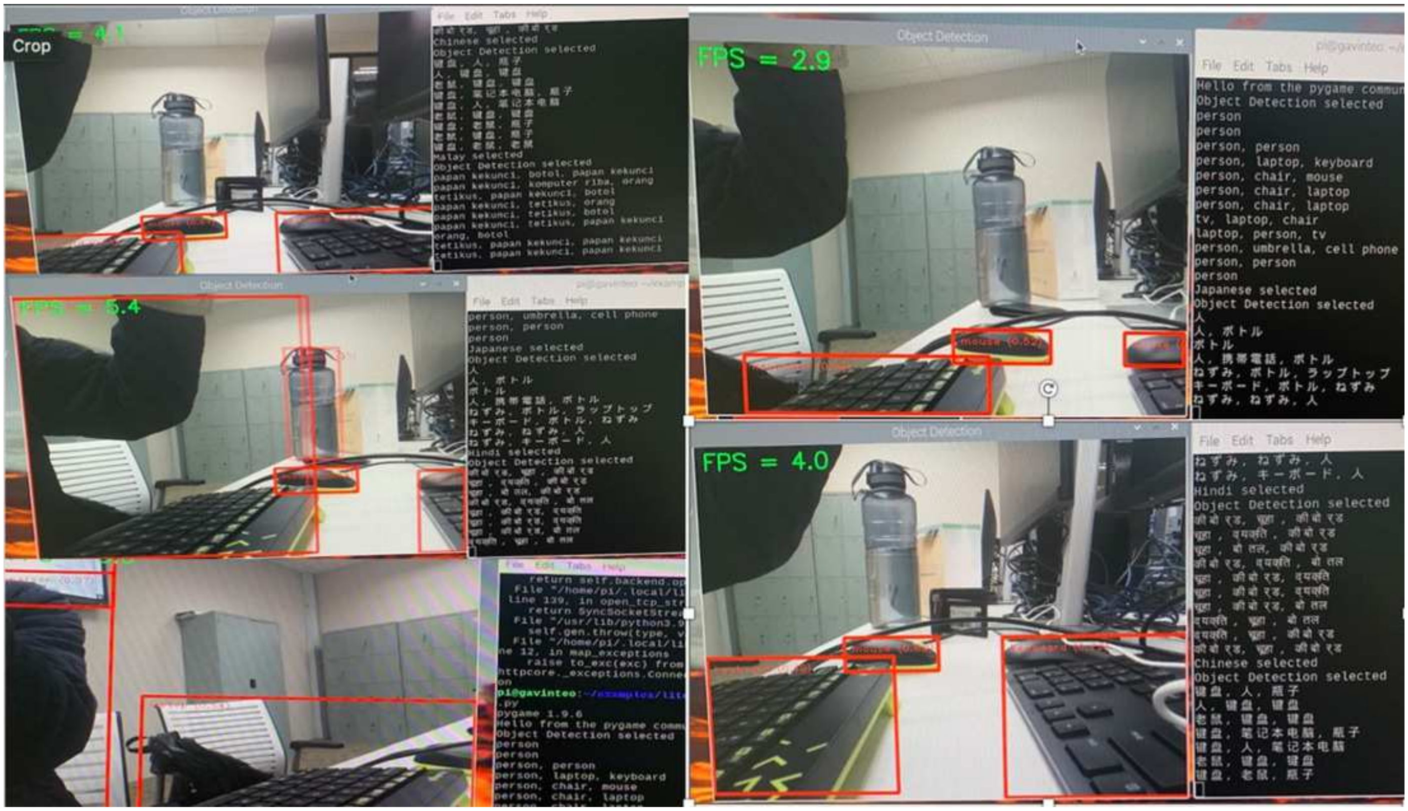
Test Data (Left) & Text Translation and Audio Output (Right)



```
Japanese selected
Translated Text:
マンゴー
桃
パパイヤ
Hindi selected
Translated Text:
आम
आड़ू
पपी ता
Malay selected
Translated Text:
Mangga
Pic
Papaya
Chinese selected
Translated Text:
芒果
桃
木瓜
```

Figure 10

Object detection with translated audio output (5 languages)



```
की बो र्ड, चूहा , की बो र्ड
Chinese selected
Object Detection selected
键盘, 人, 瓶子
人, 键盘, 键盘
老鼠, 键盘, 键盘
键盘, 笔记本电脑, 瓶子
键盘, 人, 笔记本电脑
老鼠, 键盘, 键盘
键盘, 老鼠, 瓶子
老鼠, 键盘, 瓶子
键盘, 老鼠, 老鼠
Malay selected
Object Detection selected
papan kekunci, botol, papan kekunci
papan kekunci, komputer riba, orang
tetikus, papan kekunci, botol
papan kekunci, tetikus, orang
papan kekunci, tetikus, botol
papan kekunci, tetikus, papan kekunci
orang, botol
tetikus, papan kekunci, papan kekunci
tetikus, papan kekunci, papan kekunci
```

```
person, umbrella, cell phone
person, person
person
Japanese selected
Object Detection selected
人
人, ボトル
ボトル
人, 携帯電話, ボトル
ねずみ, ボトル, ラップトップ
キーボード, ボトル, ねずみ
ねずみ, ねずみ, 人
ねずみ, キーボード, 人
Hindi selected
Object Detection selected
की बो र्ड, चूहा , की बो र्ड
चूहा , व्यक्ति , की बो र्ड
चूहा , बो तल, की बो र्ड
की बो र्ड, व्यक्ति , बो तल
चूहा , की बो र्ड, व्यक्ति
चूहा , की बो र्ड, व्यक्ति
चूहा , की बो र्ड, बो तल
व्यक्ति , चूहा , बो तल
```

```
    return self.backend.op
  File "/home/pi/.local/li
 line 139, in open_tcp_str
    return SyncSocketStrea
  File "/usr/lib/python3.9
    self.gen.throw(type, v
  File "/home/pi/.local/li
ne 12, in map_exceptions
    raise to_exc(exc) from
httpcore._exceptions.Conne
on
pi@gavinteo:~/examples/lit
.py
pygame 1.9.6
Hello from the pygame comm
Object Detection selected
person
person
person, person
person, laptop, keyboard
person, chair, mouse
person, chair, laptop
```

```
Hello from the pygame commun
Object Detection selected
person
person
person, person
person, laptop, keyboard
person, chair, mouse
person, chair, laptop
person, chair, laptop
tv, laptop, chair
laptop, person, tv
person, umbrella, cell phone
person, person
person
Japanese selected
Object Detection selected
人
人, ボトル
ボトル
人, 携帯電話, ボトル
ねずみ, ボトル, ラップトップ
キーボード, ボトル, ねずみ
ねずみ, ねずみ, 人
```

```
ねずみ, ねずみ, 人
ねずみ, キーボード, 人
Hindi selected
Object Detection selected
की बो र्ड, चूहा , की बो र्ड
चूहा , व्यक्ति , की बो र्ड
चूहा , बो तल, की बो र्ड
की बो र्ड, व्यक्ति , बो तल
चूहा , की बो र्ड, व्यक्ति
चूहा , की बो र्ड, व्यक्ति
चूहा , की बो र्ड, बो तल
व्यक्ति , चूहा , बो तल
व्यक्ति , चूहा , की बो र्ड
की बो र्ड, चूहा , की बो र्ड
Chinese selected
Object Detection selected
键盘, 人, 瓶子
人, 键盘, 键盘
老鼠, 键盘, 键盘
键盘, 笔记本电脑, 瓶子
键盘, 人, 笔记本电脑
老鼠, 键盘, 键盘
键盘, 老鼠, 瓶子
```

Figure 11

Examples of under exposure (Left), normal (Middle), over exposure (Right)

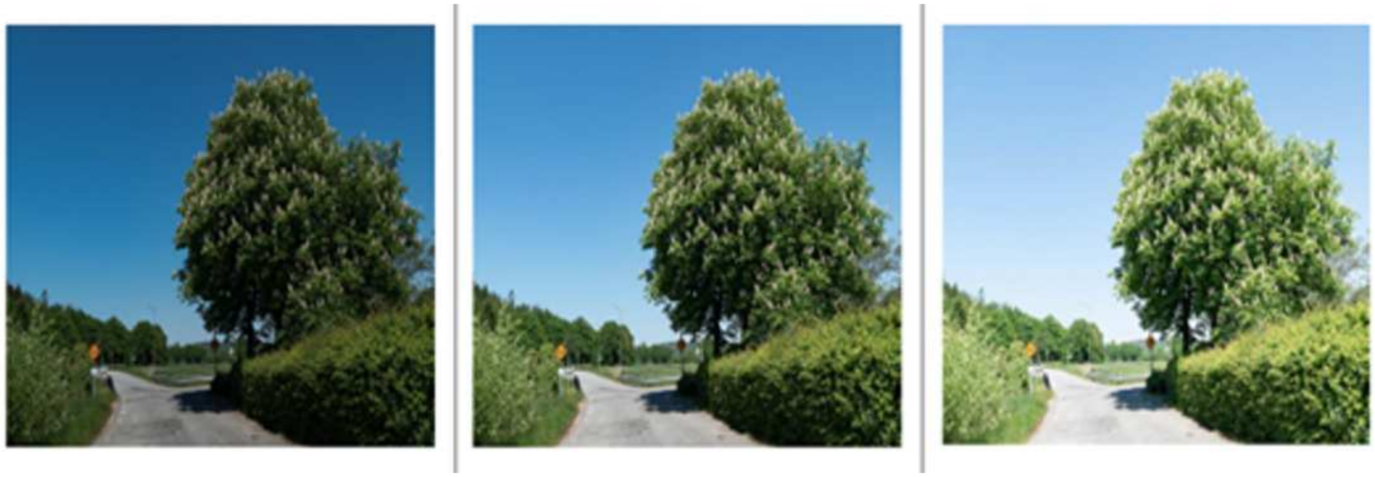


Figure 12

Example of non-Motion blur (Left) vs motion blur (Right)

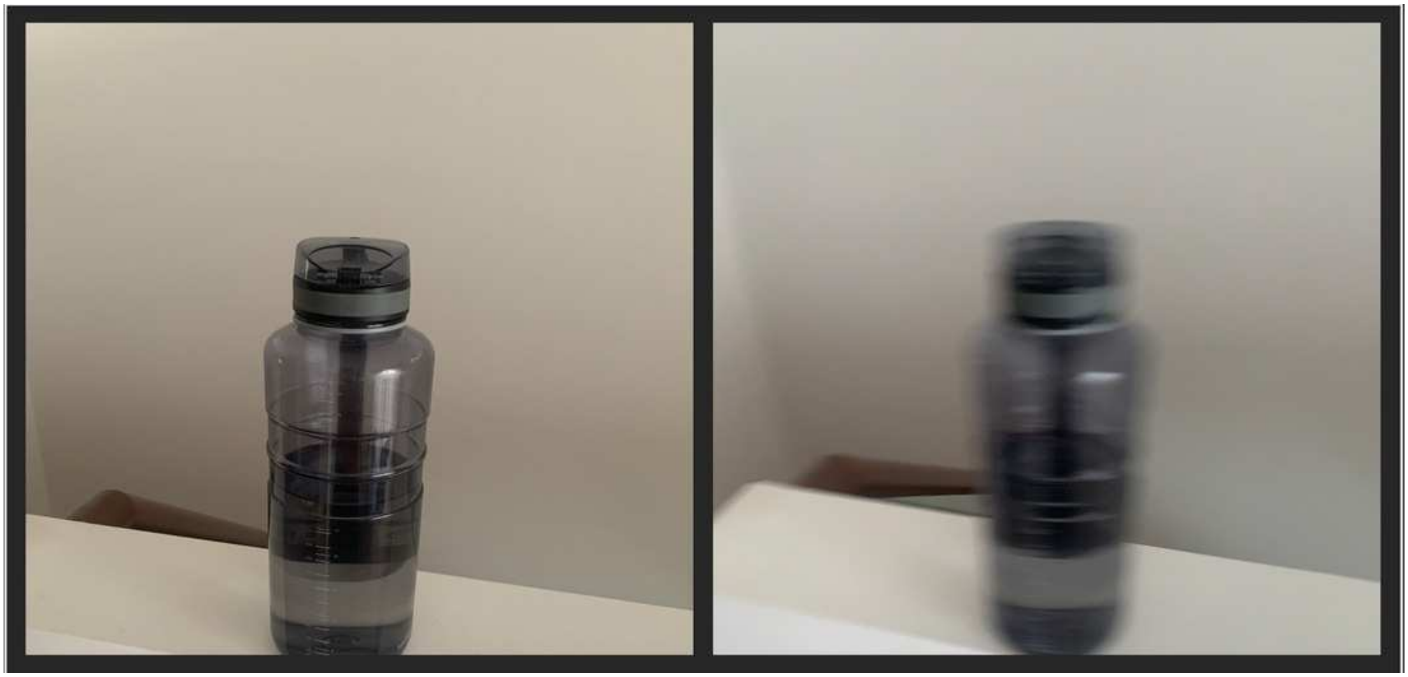
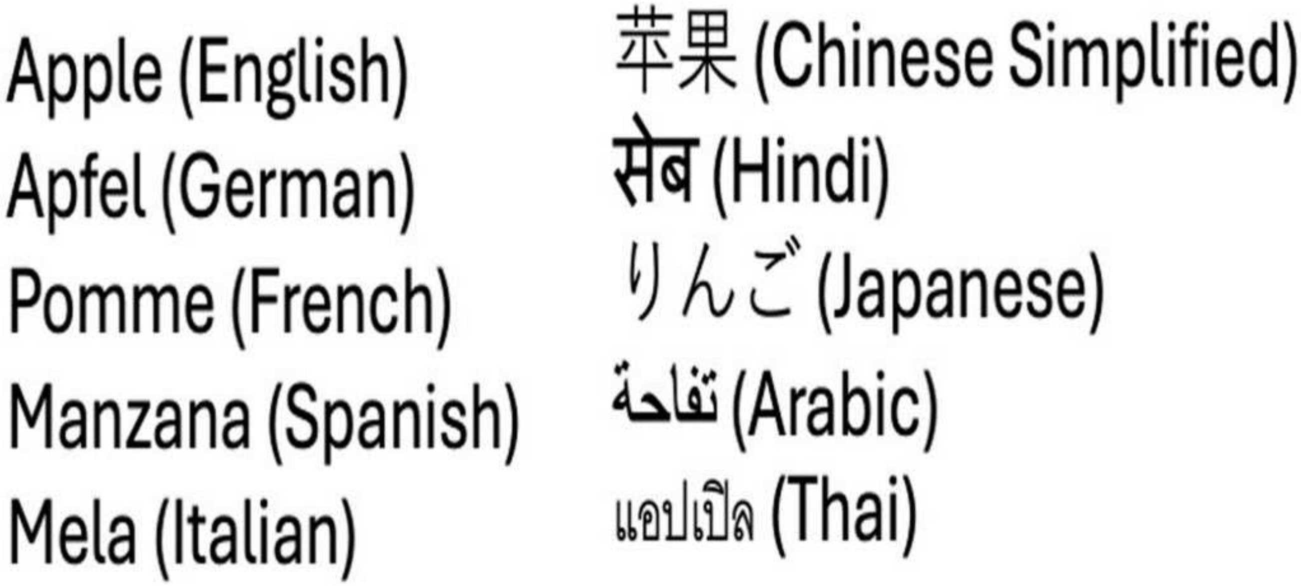


Figure 13

Latin Alphabet (Left) vs Non-Latin Alphabet (Right)



Apple (English)

Apfel (German)

Pomme (French)

Manzana (Spanish)

Mela (Italian)

苹果 (Chinese Simplified)

सेब (Hindi)

りんご (Japanese)

تفاحة (Arabic)

แอปเปิล (Thai)

Figure 14

New and Improved GUI Menu

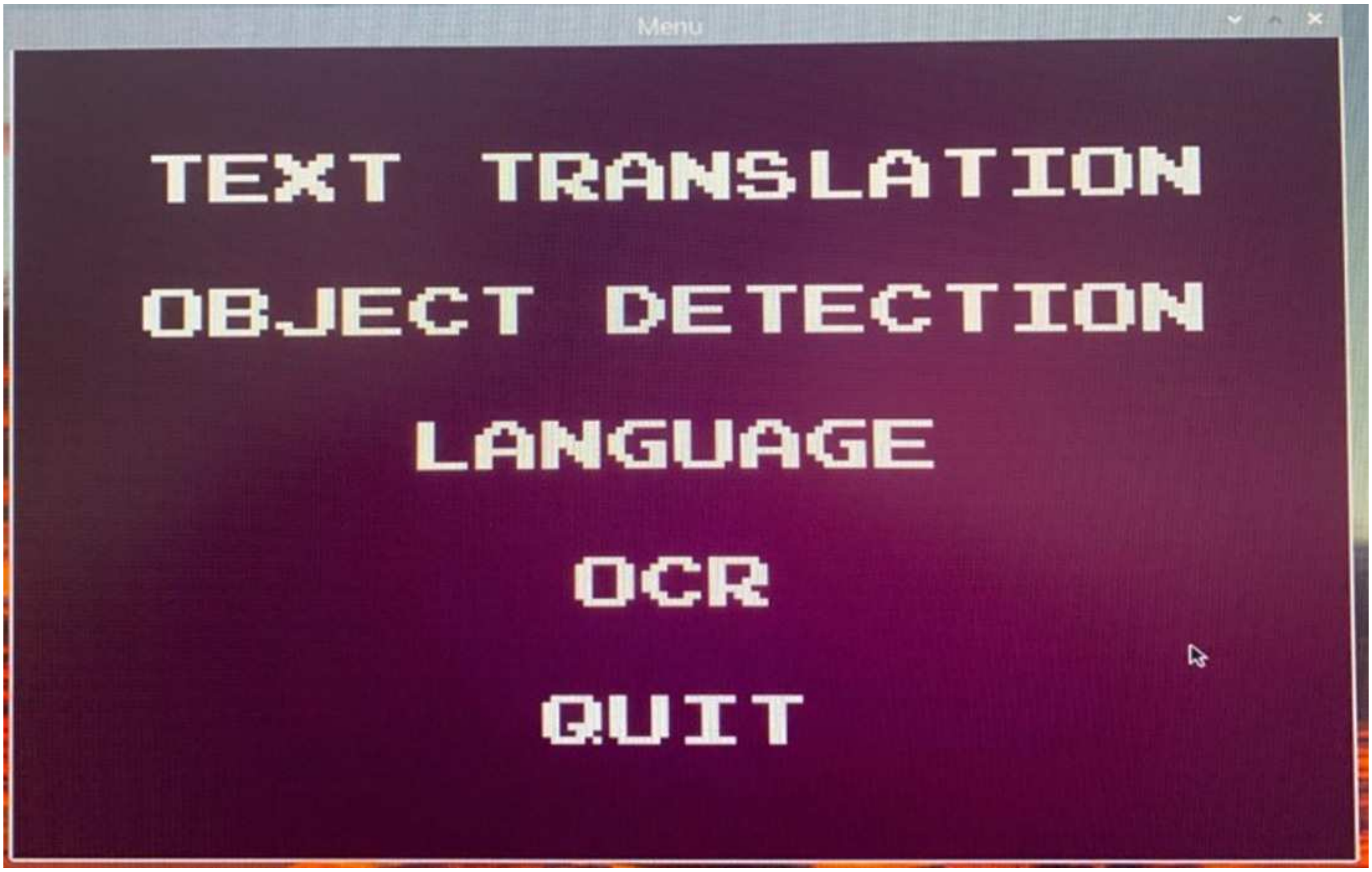


Figure 15

OCR Menu (Determine which language user wants to read)

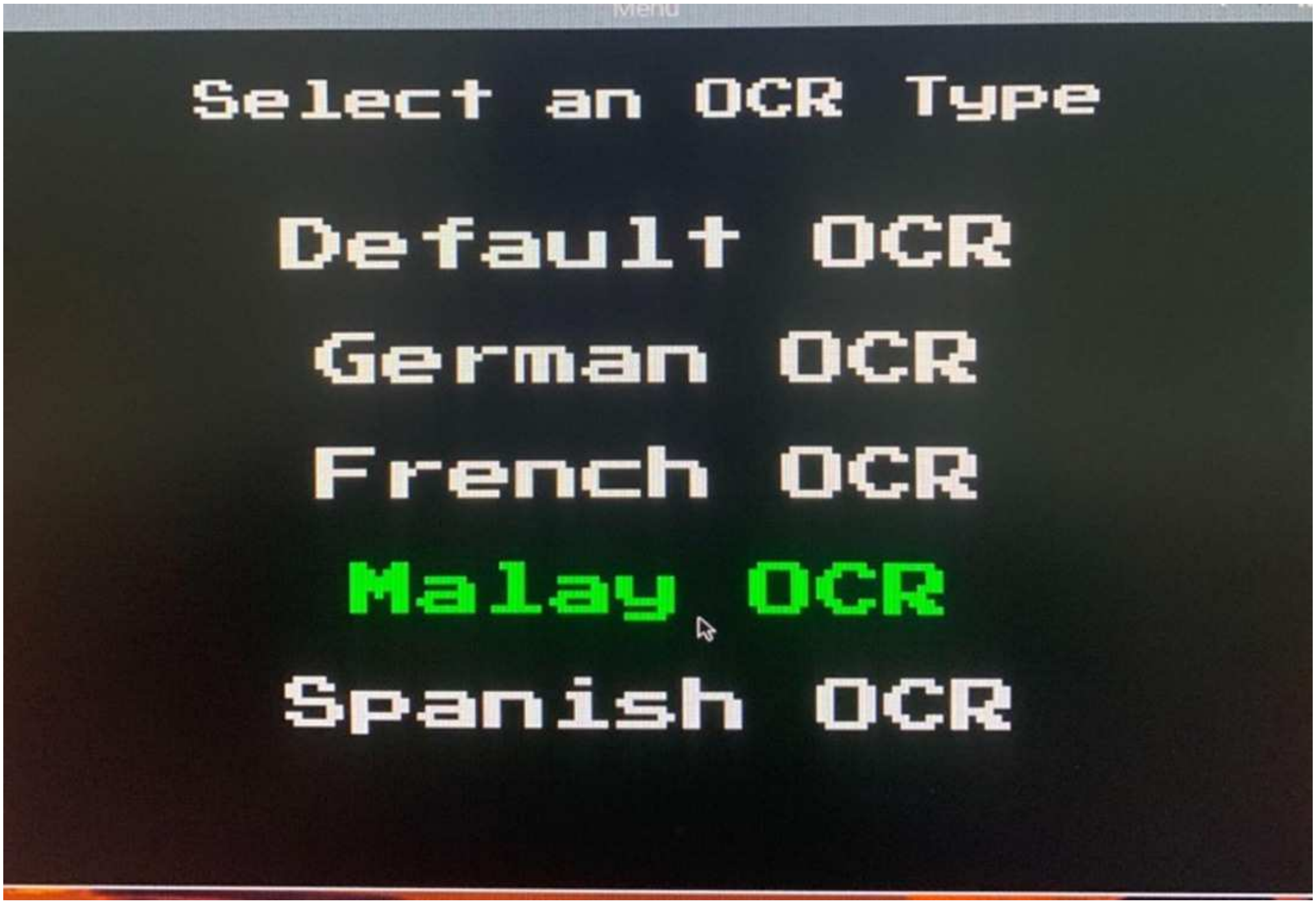


Figure 16

User Wearing the Proposed System

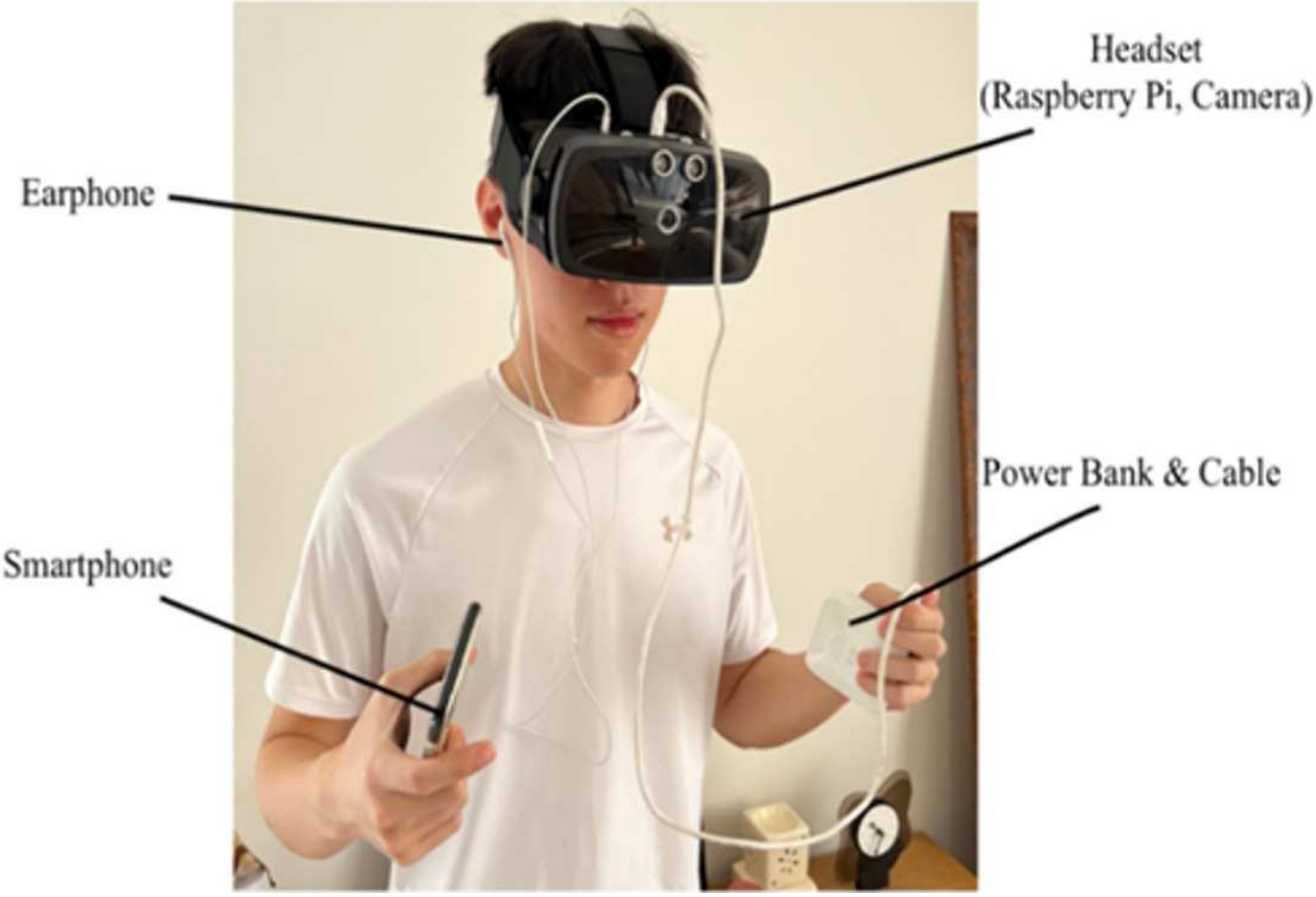
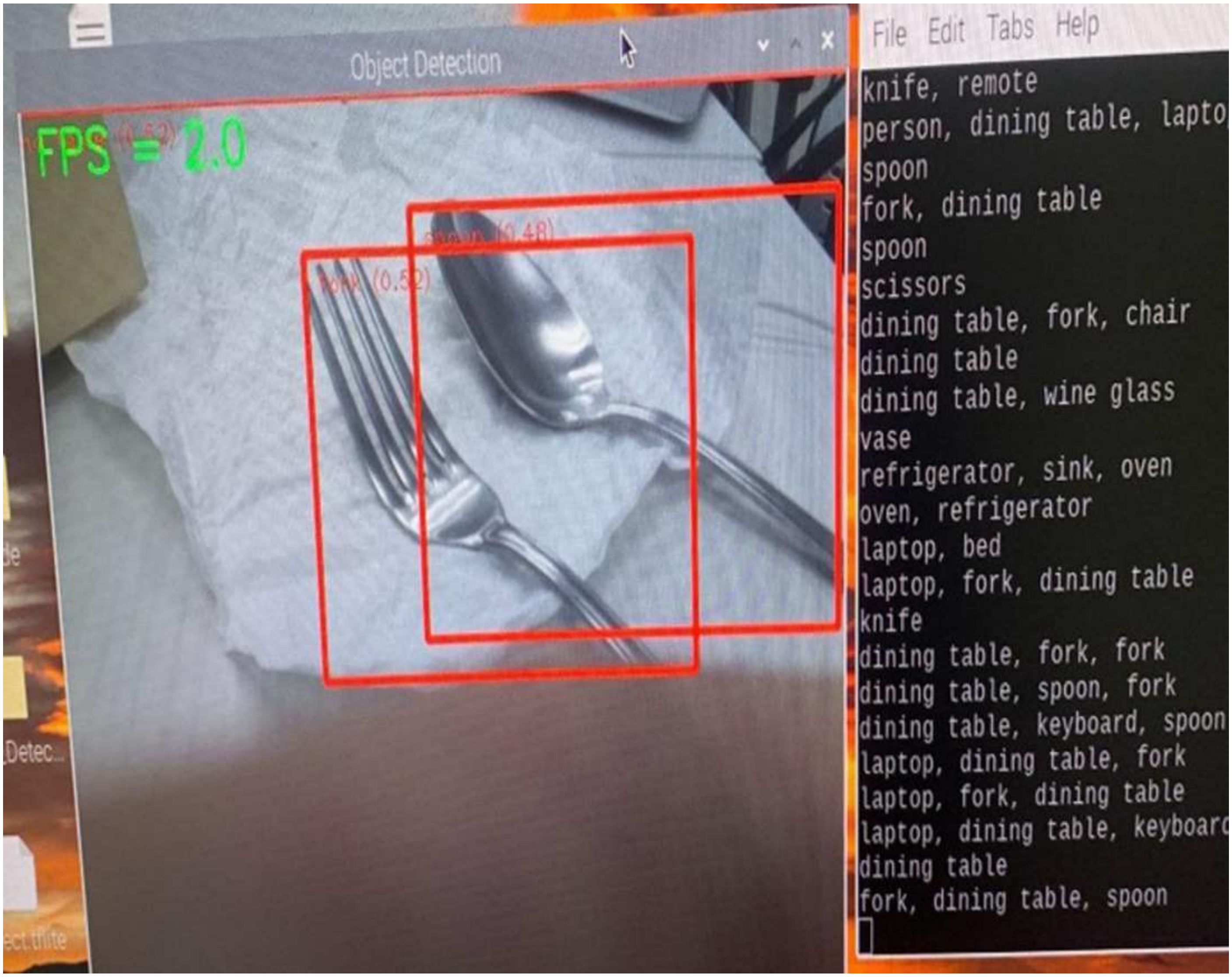


Figure 17

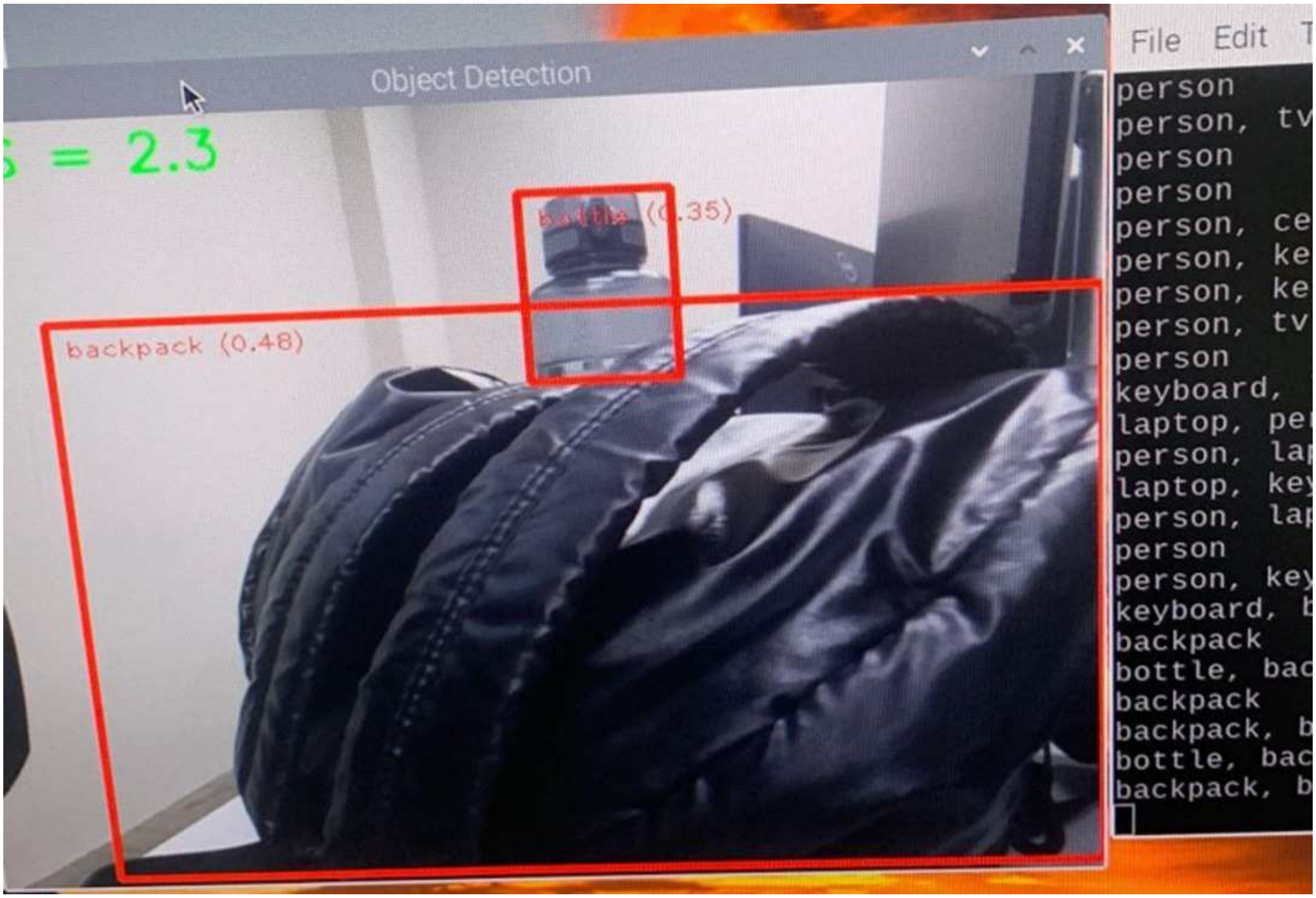
Object detection on fork & spoon



```
knife, remote
person, dining table, lapto
spoon
fork, dining table
spoon
scissors
dining table, fork, chair
dining table
dining table, wine glass
vase
refrigerator, sink, oven
oven, refrigerator
laptop, bed
laptop, fork, dining table
knife
dining table, fork, fork
dining table, spoon, fork
dining table, keyboard, spoon
laptop, dining table, fork
laptop, fork, dining table
laptop, dining table, keyboar
dining table
fork, dining table, spoon
```

Figure 18

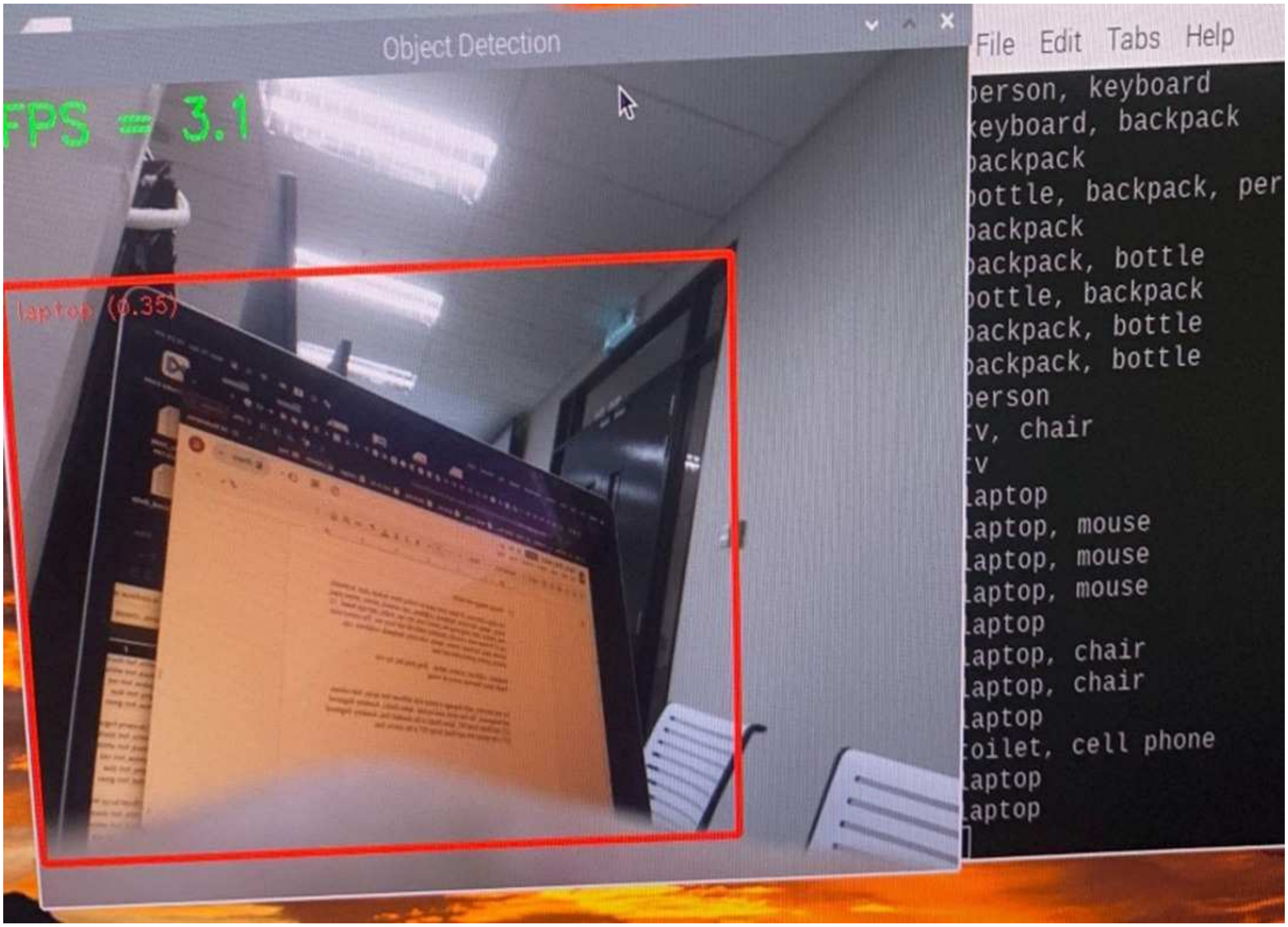
Object detection on backpack & bottle



```
person
person, tv
person
person
person, ce
person, ke
person, ke
person, tv
person
keyboard,
laptop, pe
person, la
laptop, key
person, la
person
person, key
keyboard, b
backpack
bottle, bac
backpack
backpack, b
bottle, bac
backpack, b
```

Figure 19

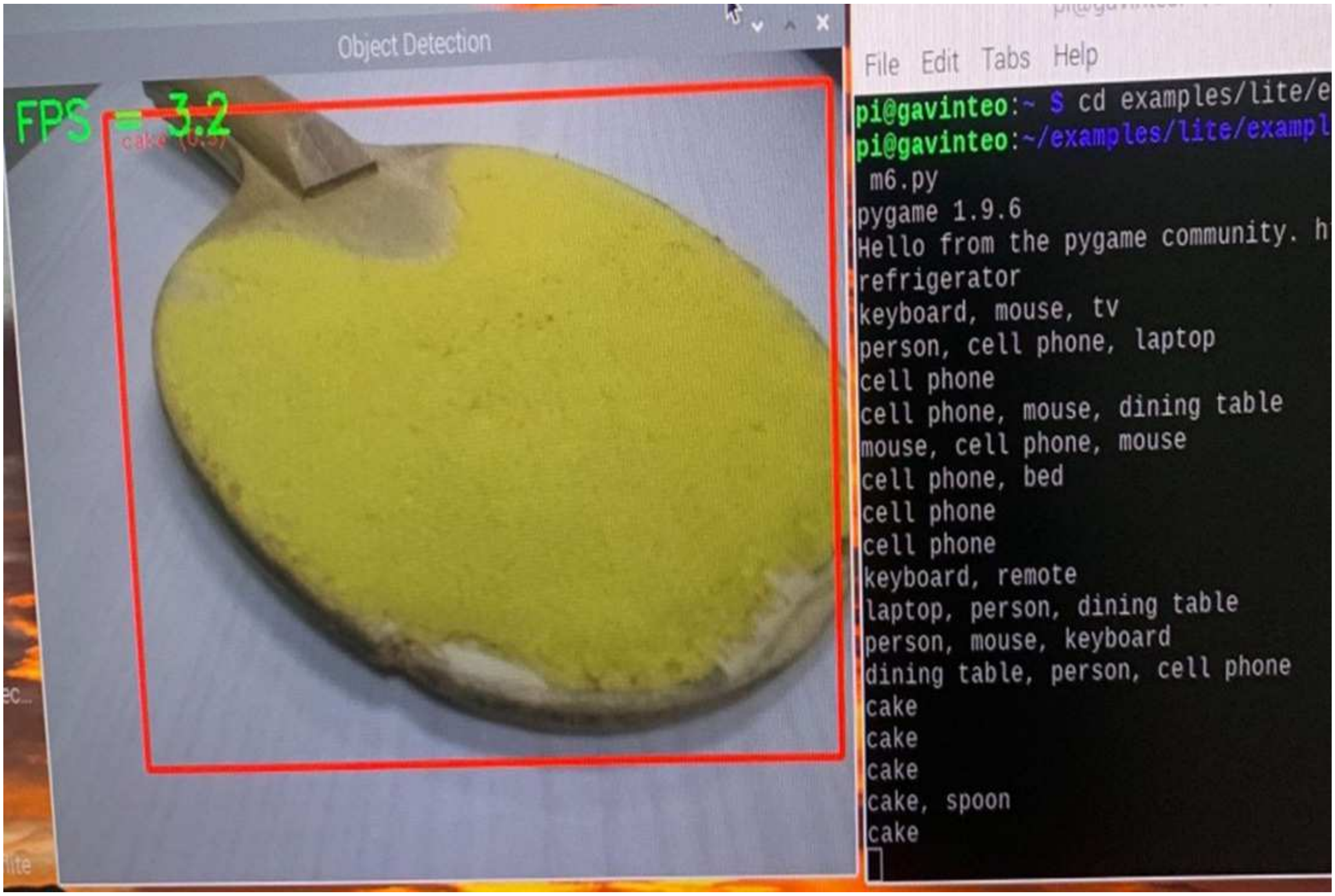
Object detection on laptop



```
erson, keyboard
keyboard, backpack
ackpack
ottle, backpack, per
ackpack
ackpack, bottle
ottle, backpack
ackpack, bottle
ackpack, bottle
erson
v, chair
v
aptop
aptop, mouse
aptop, mouse
aptop, mouse
aptop
aptop, chair
aptop, chair
aptop
oilet, cell phone
aptop
aptop
```

Figure 20

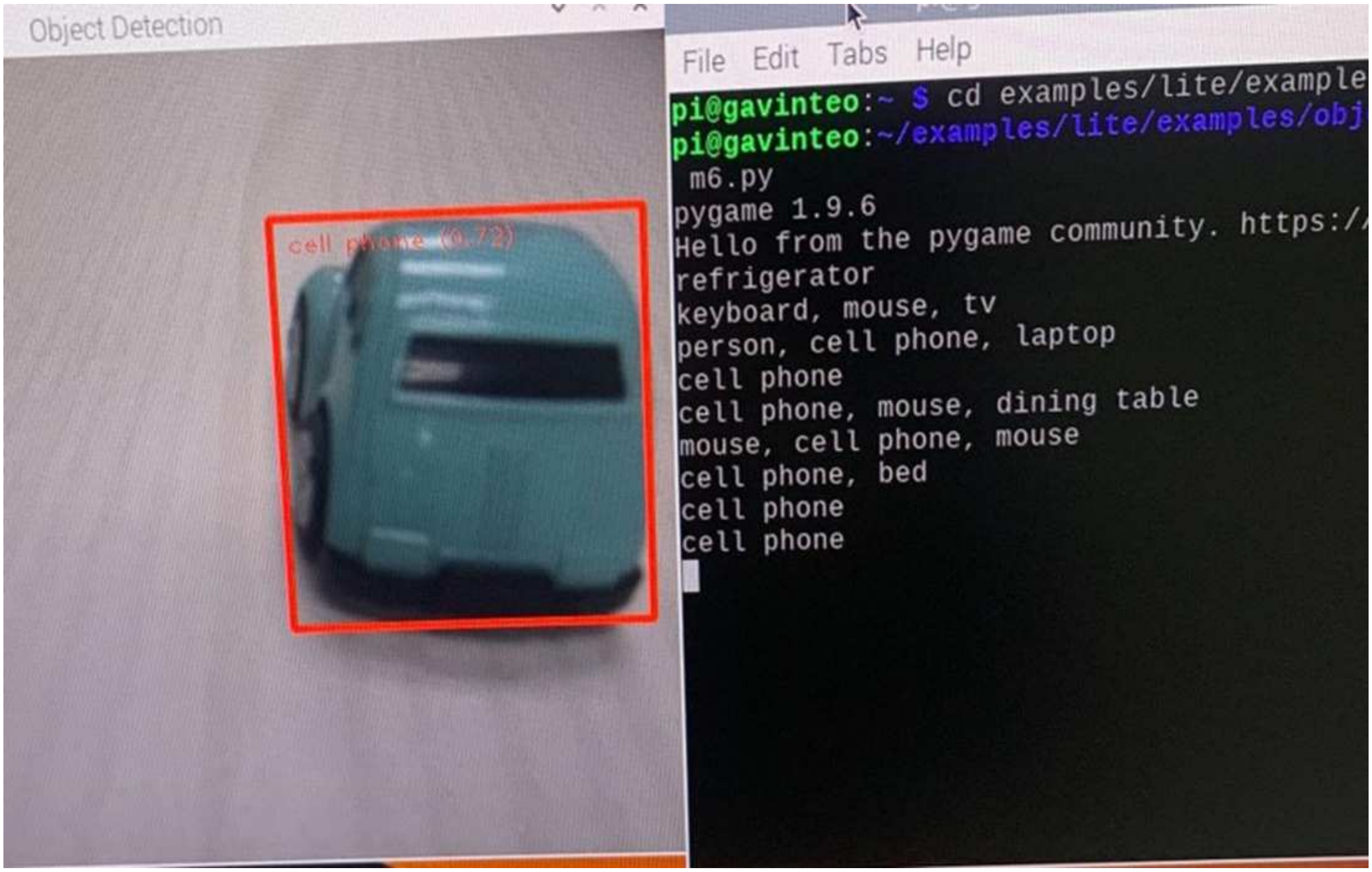
Object detection on ping pong bat (Unsuccessful)



```
pi@gavinteo:~ $ cd examples/lite/e
pi@gavinteo:~/examples/lite/exampl
 m6.py
pygame 1.9.6
Hello from the pygame community. h
refrigerator
keyboard, mouse, tv
person, cell phone, laptop
cell phone
cell phone, mouse, dining table
mouse, cell phone, mouse
cell phone, bed
cell phone
cell phone
keyboard, remote
laptop, person, dining table
person, mouse, keyboard
dining table, person, cell phone
cake
cake
cake
cake, spoon
cake
```

Figure 21

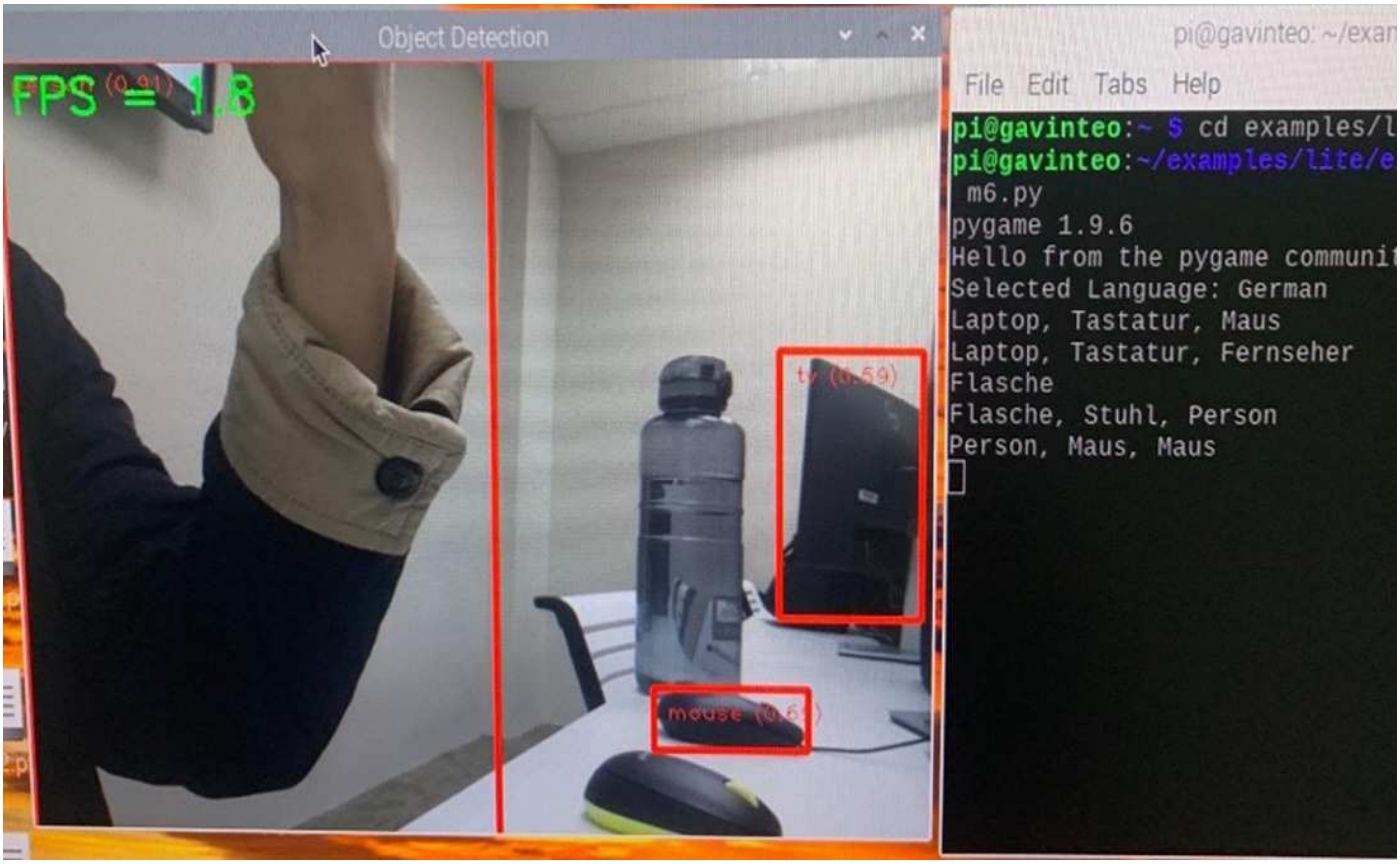
Object detection on toy car (Unsuccessful)



```
pi@gavinteo:~ $ cd examples/lite/example
pi@gavinteo:~/examples/lite/examples/obj
 m6.py
pygame 1.9.6
Hello from the pygame community. https:/
refrigerator
keyboard, mouse, tv
person, cell phone, laptop
cell phone
cell phone, mouse, dining table
mouse, cell phone, mouse
cell phone, bed
cell phone
cell phone
```

Figure 22

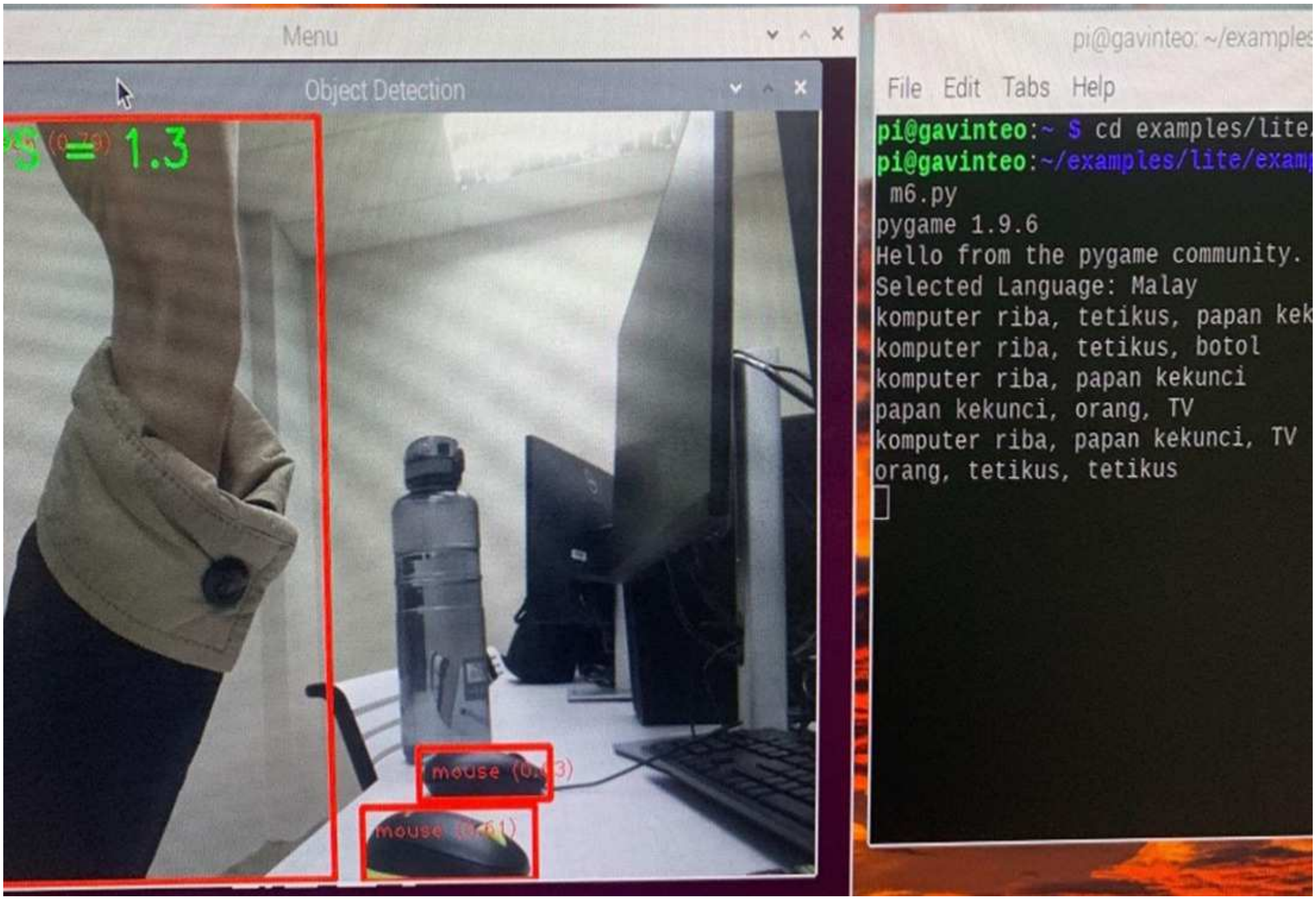
Object detection on mouse, bottle, laptop with German translation



```
pi@gavinteo:~ $ cd examples/l
pi@gavinteo:~/examples/lite/e
 m6.py
pygame 1.9.6
Hello from the pygame communit
Selected Language: German
Laptop, Tastatur, Maus
Laptop, Tastatur, Fernseher
Flasche
Flasche, Stuhl, Person
Person, Maus, Maus
```

Figure 23

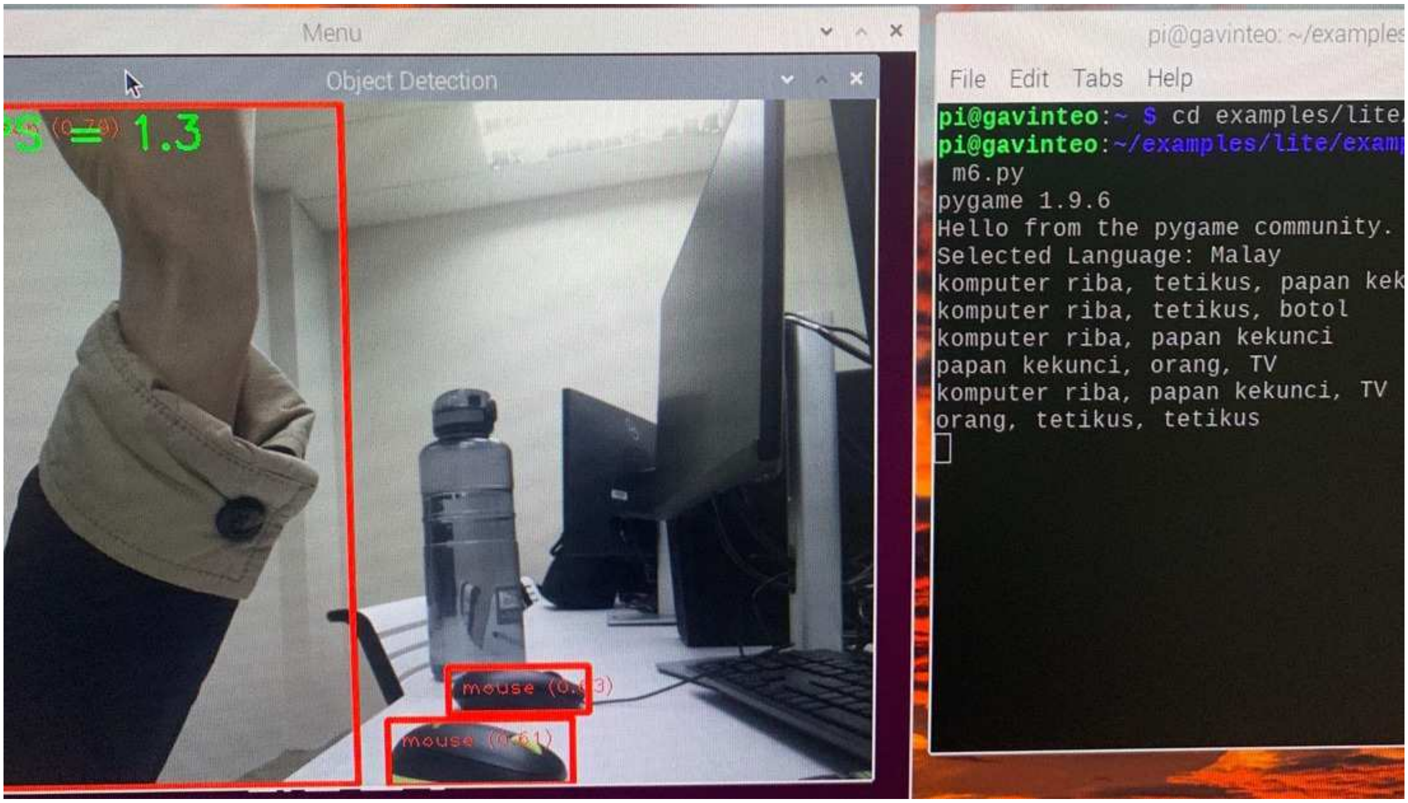
Object detection on mouse, bottle, laptop with French translation



```
pi@gavinteo:~ $ cd examples/lite
pi@gavinteo:~/examples/lite/examp
 m6.py
pygame 1.9.6
Hello from the pygame community.
Selected Language: Malay
komputer riba, tetikus, papan kek
komputer riba, tetikus, botol
komputer riba, papan kekunci
papan kekunci, orang, TV
komputer riba, papan kekunci, TV
orang, tetikus, tetikus
```

Figure 24

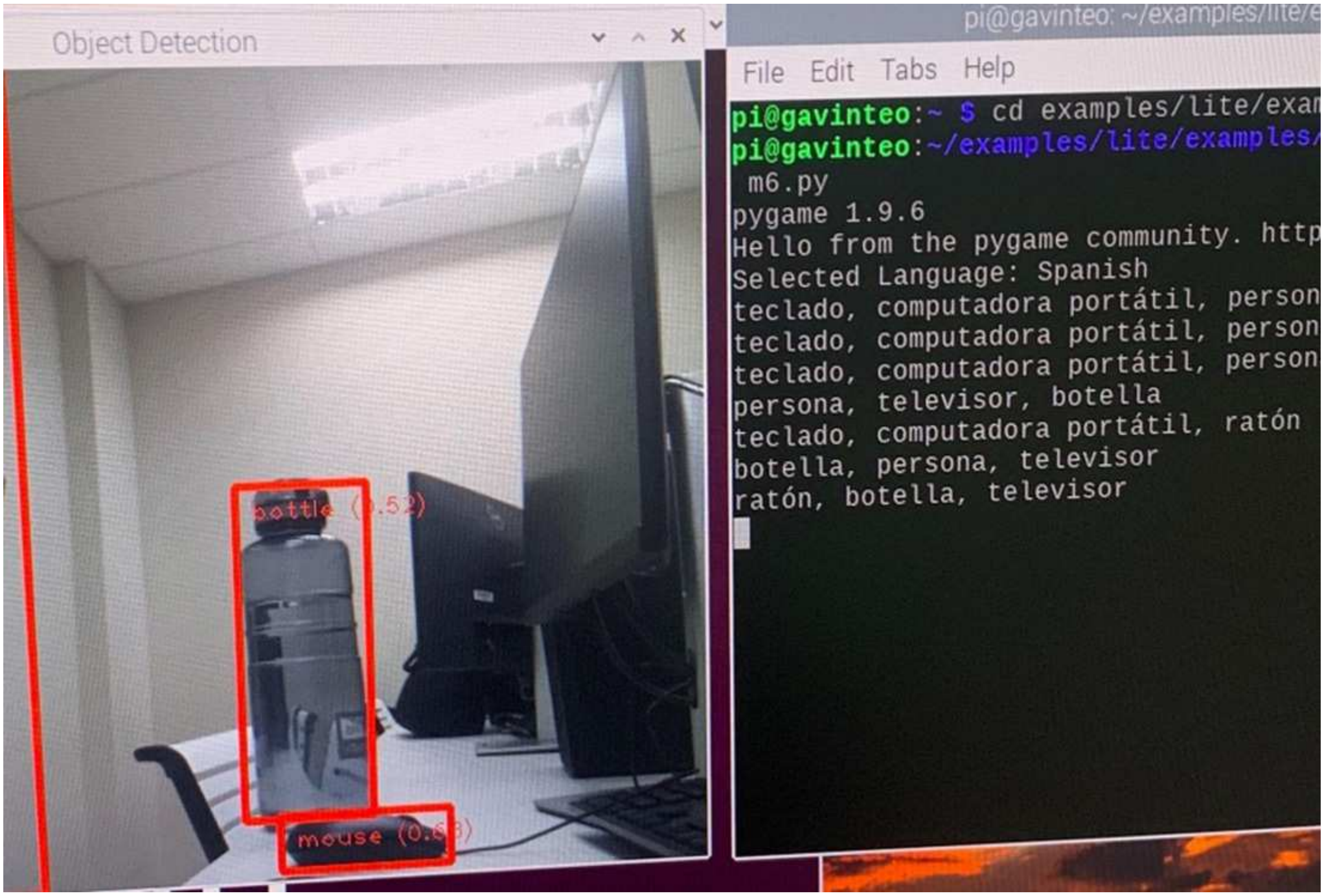
Object detection on mouse, bottle, laptop with Malay translation



```
pi@gavinteo:~ $ cd examples/lite
pi@gavinteo:~/examples/lite/examp
 m6.py
pygame 1.9.6
Hello from the pygame community.
Selected Language: Malay
komputer riba, tetikus, papan kek
komputer riba, tetikus, botol
komputer riba, papan kekunci
papan kekunci, orang, TV
komputer riba, papan kekunci, TV
orang, tetikus, tetikus
```

Figure 25

Object detection on mouse, bottle, laptop with Spanish translation



```
pi@gavinteo:~ $ cd examples/lite/exam
pi@gavinteo:~/examples/lite/examples/
 m6.py
pygame 1.9.6
Hello from the pygame community. http
Selected Language: Spanish
teclado, computadora portátil, person
teclado, computadora portátil, person
teclado, computadora portátil, person
persona, televisor, botella
teclado, computadora portátil, ratón
botella, persona, televisor
ratón, botella, televisor
```

Figure 26

Object detection on car at outdoor environment

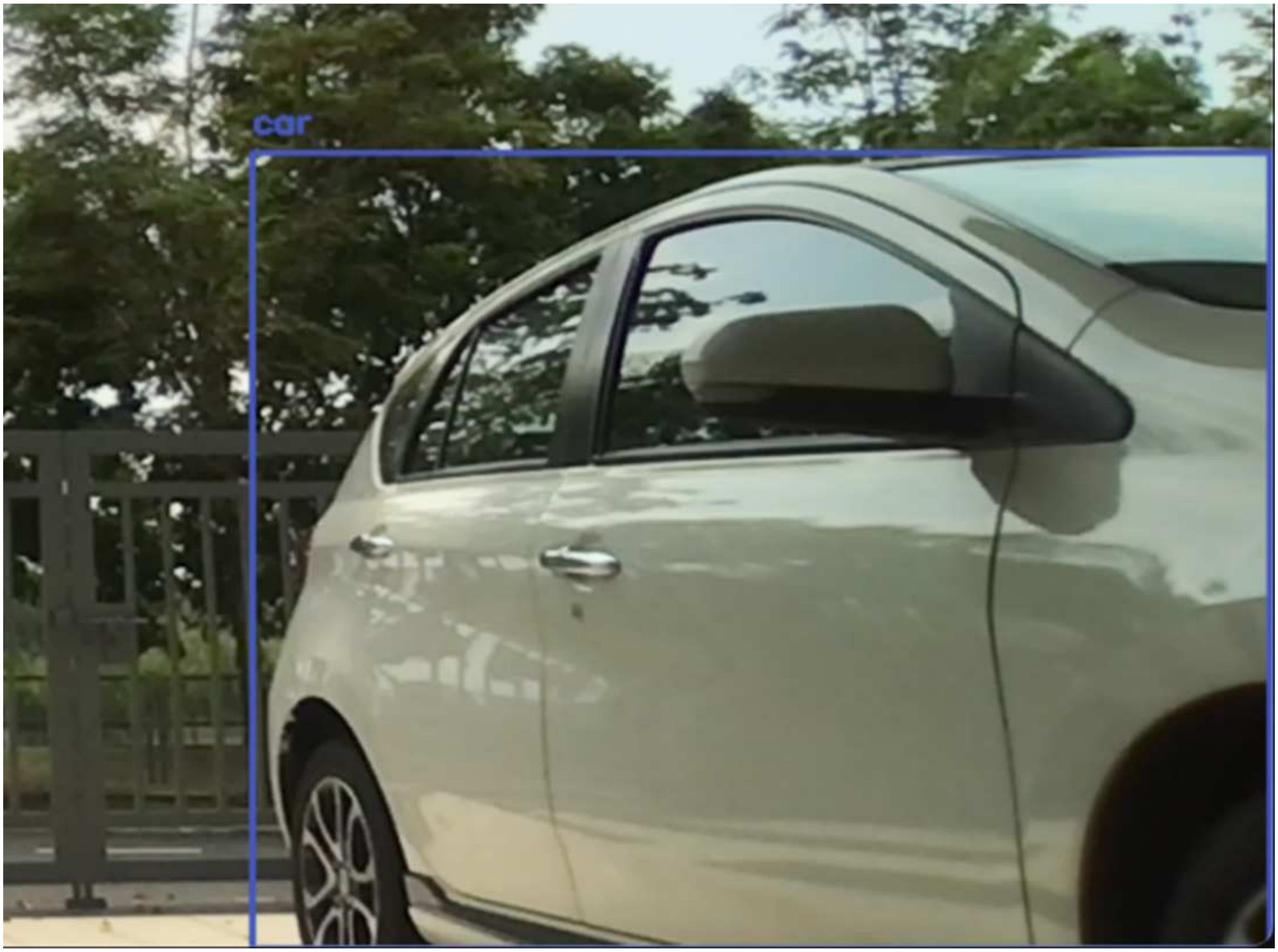


Figure 27

Object detection on plants at outdoor environment

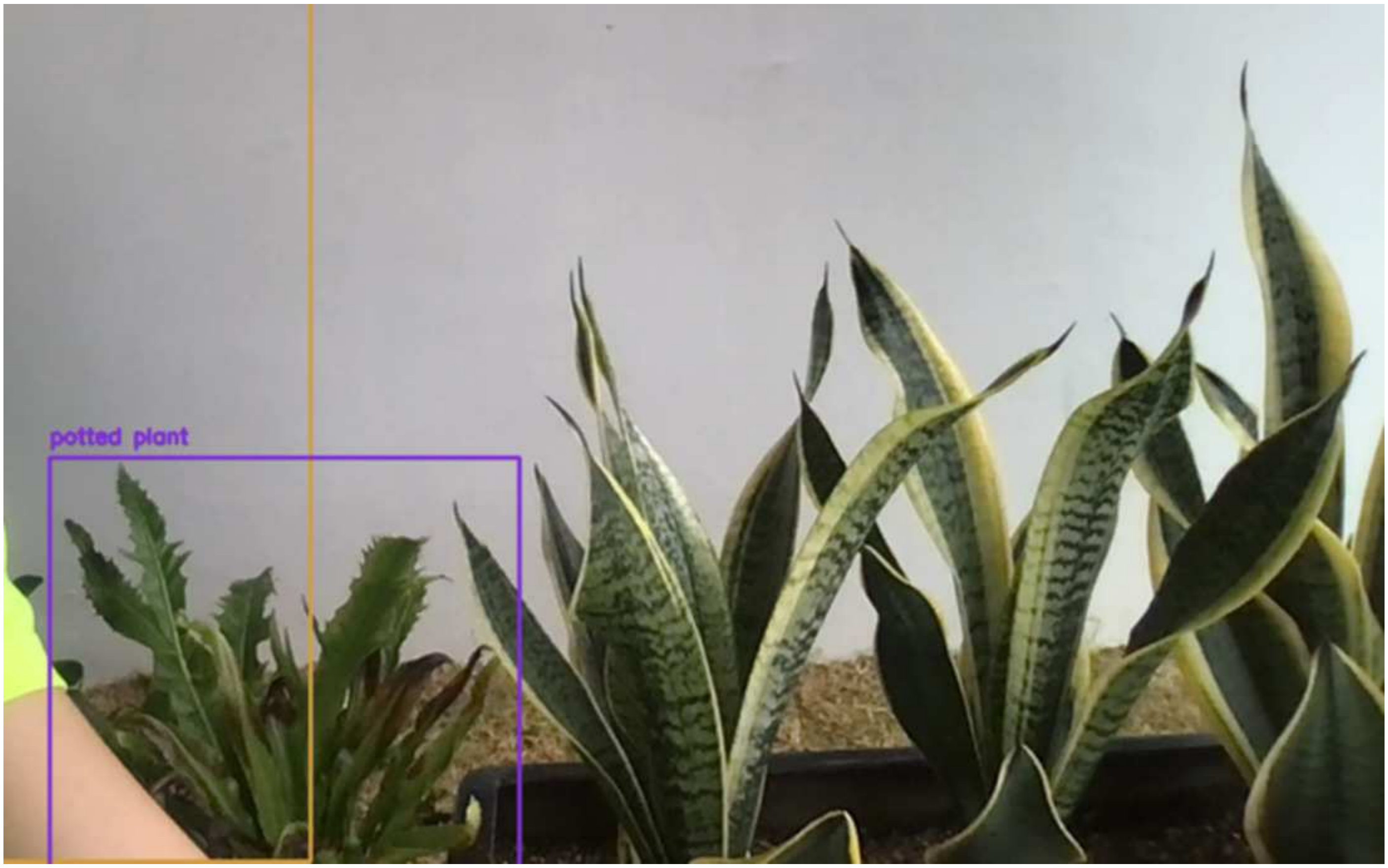


Figure 28

Object detection on chairs at outdoor environment

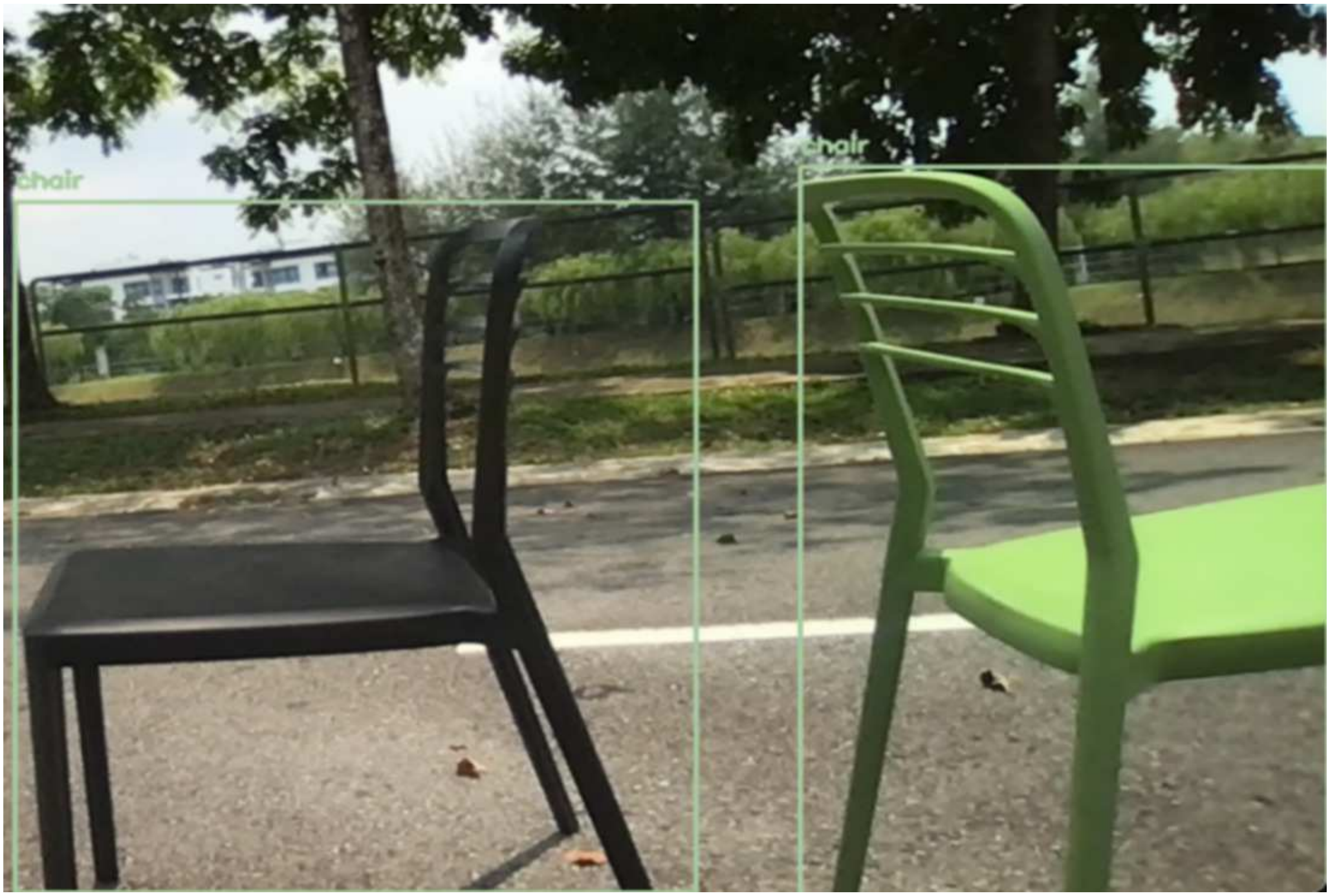


Figure 29

Object detection on cars at outdoor environment

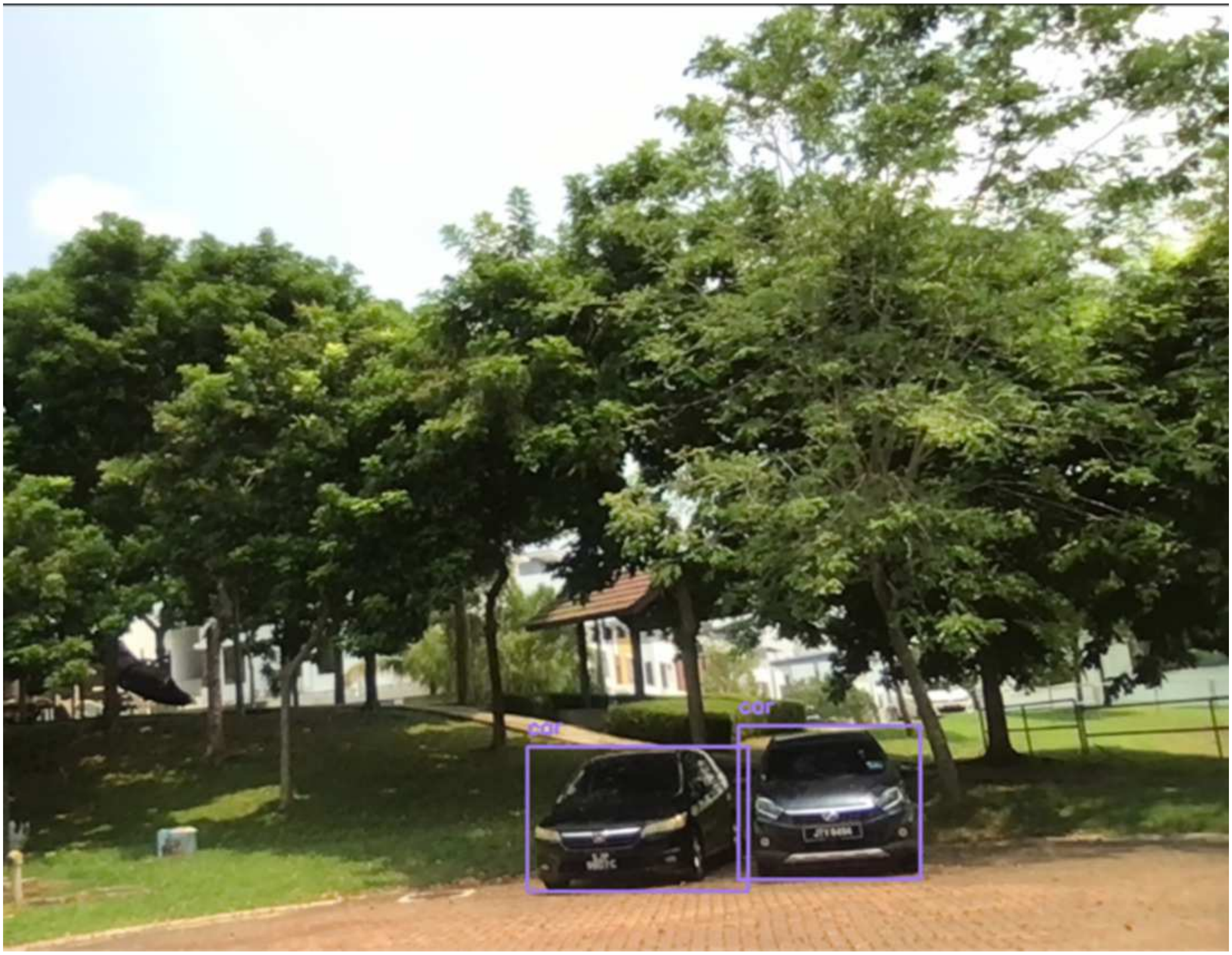


Figure 30

Object detection on rubbish bin at outdoor environment (Unsuccessful)

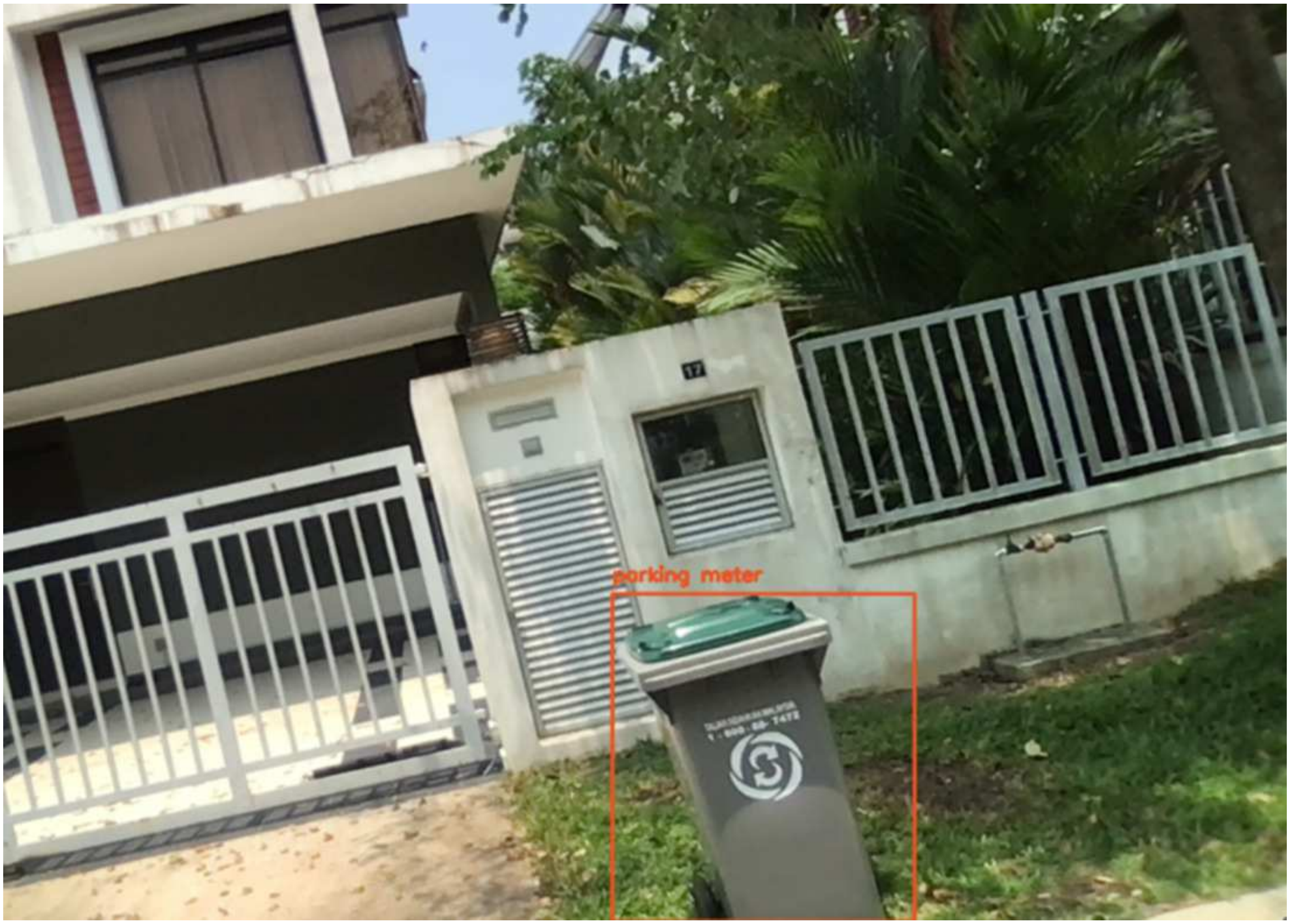


Figure 31

Example text for Aptos (body) (Top), Academy Engraved LET (Middle), Brush Script MT (Bottom)

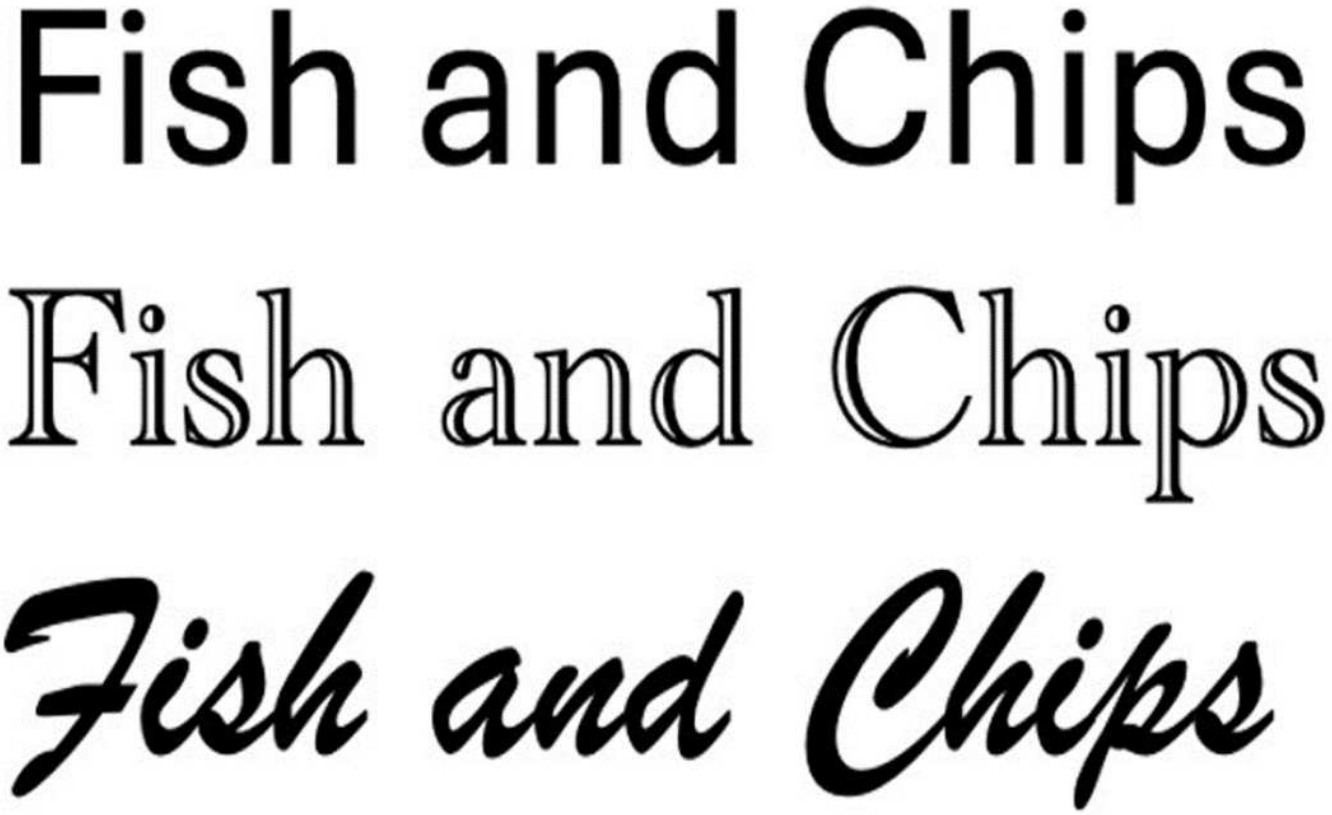
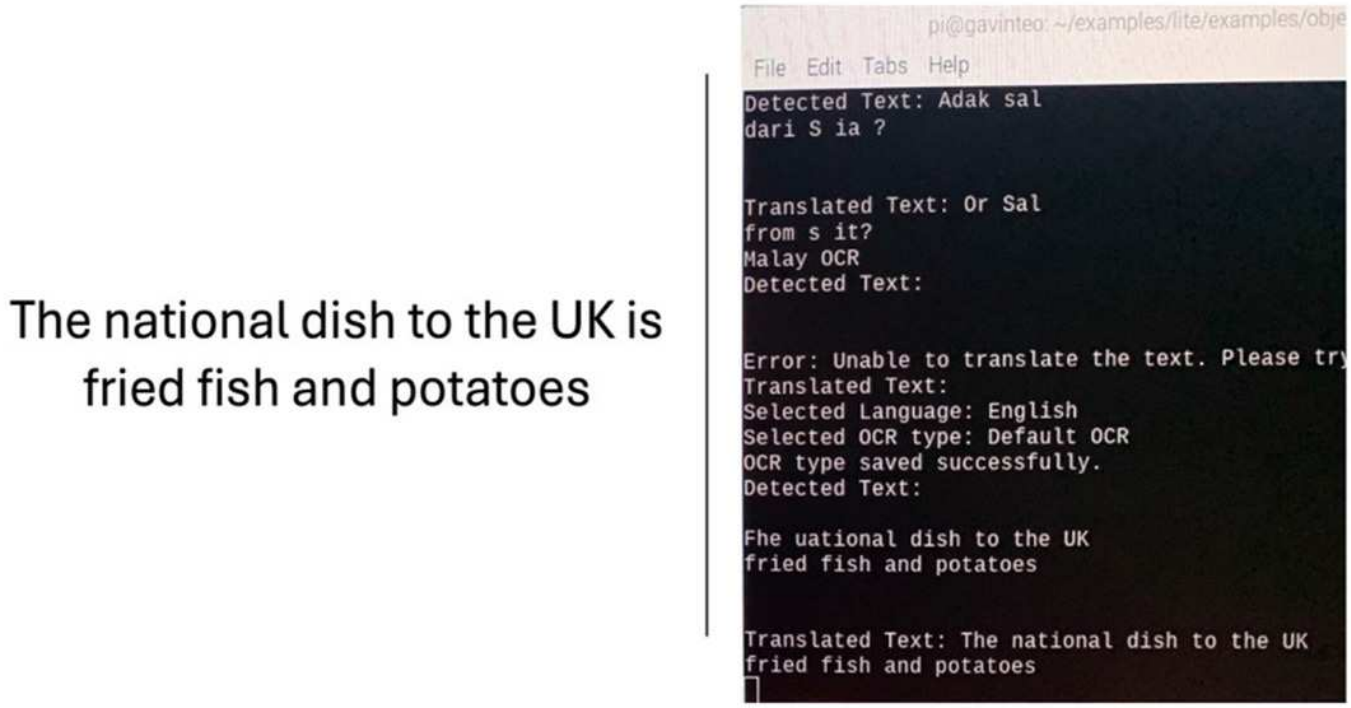


Figure 32

Translation from English to English (With white background, black font)

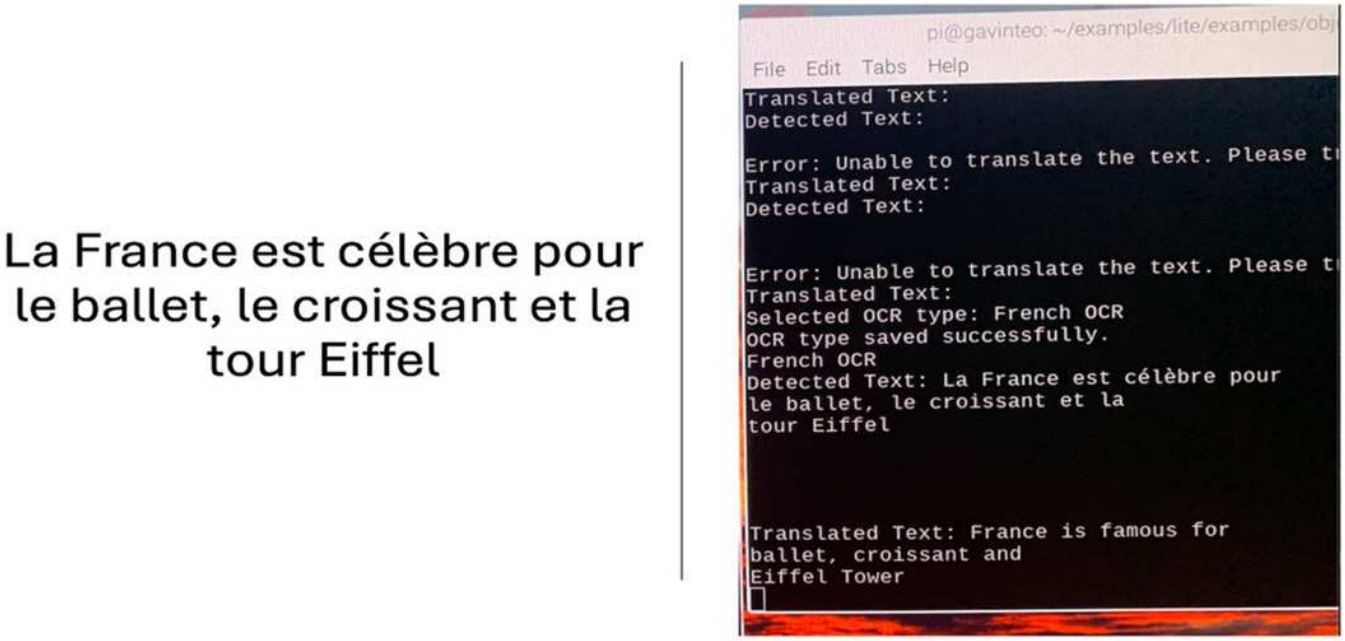


The national dish to the UK is fried fish and potatoes

```
Detected Text: Adak sal
dari S ia ?


Translated Text: Or Sal
from s it?
Malay OCR
Detected Text:


Error: Unable to translate the text. Please try
Translated Text:
Selected Language: English
Selected OCR type: Default OCR
OCR type saved successfully.
Detected Text:

Fhe uational dish to the UK
fried fish and potatoes


Translated Text: The national dish to the UK
fried fish and potatoes
```

Figure 33

Translation from French to English (With white background, black font

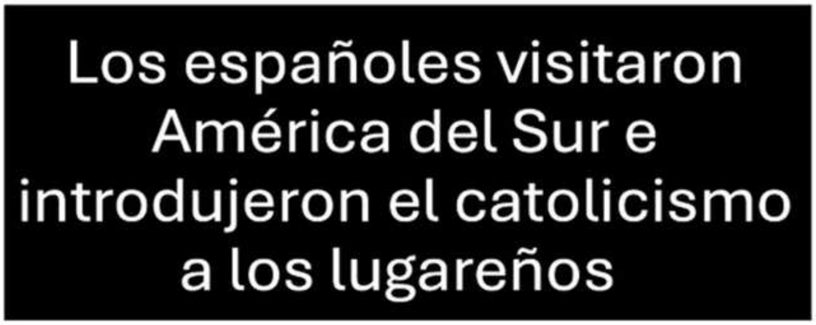


La France est célèbre pour
le ballet, le croissant et la
tour Eiffel

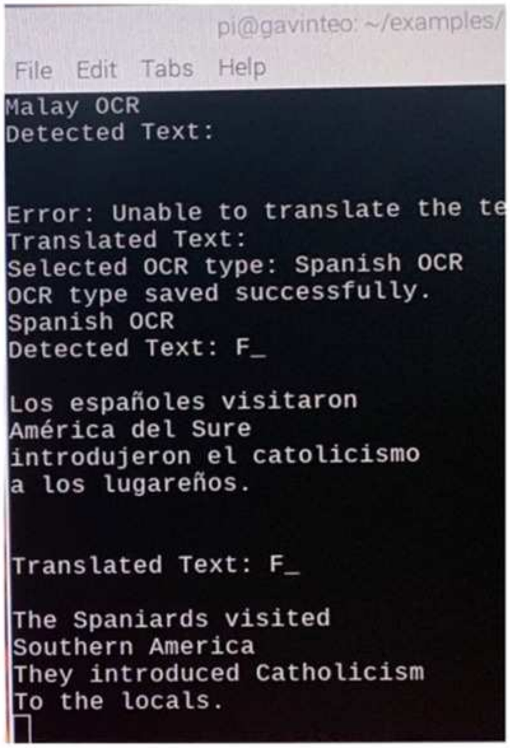
```
Translated Text:
Detected Text:

Error: Unable to translate the text. Please t
Translated Text:
Detected Text:


Error: Unable to translate the text. Please t
Translated Text:
Selected OCR type: French OCR
OCR type saved successfully.
French OCR
Detected Text: La France est célèbre pour
le ballet, le croissant et la
tour Eiffel



Translated Text: France is famous for
ballet, croissant and
Eiffel Tower
```

Figure 34

Translation from Spanish to English (With black background, white font)



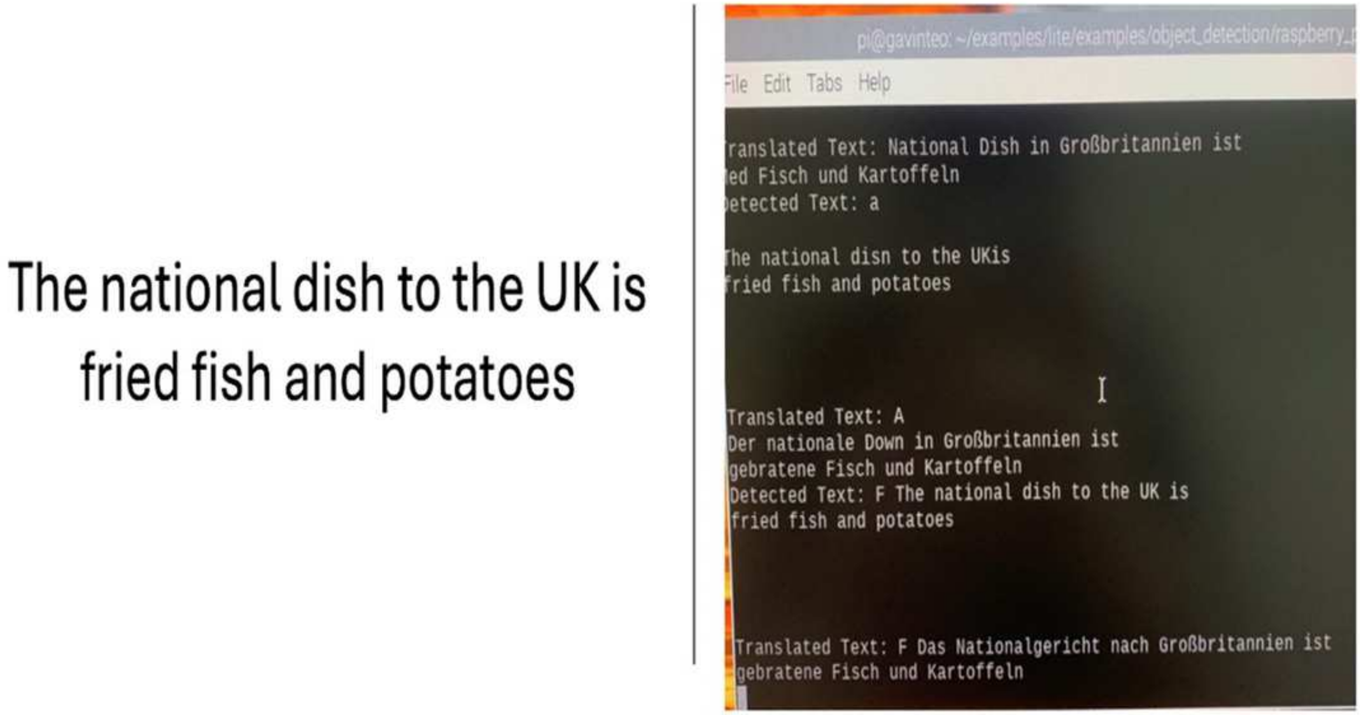
Los españoles visitaron
América del Sur e
introdujeron el catolicismo
a los lugareños



```
Malay OCR
Detected Text:


Error: Unable to translate the te
Translated Text:
Selected OCR type: Spanish OCR
OCR type saved successfully.
Spanish OCR
Detected Text: F_

Los españoles visitaron
América del Sure
introdujeron el catolicismo
a los lugareños.


Translated Text: F_

The Spaniards visited
Southern America
They introduced Catholicism
To the locals.
```

Figure 35

Translation from English to German (With white background, black font)



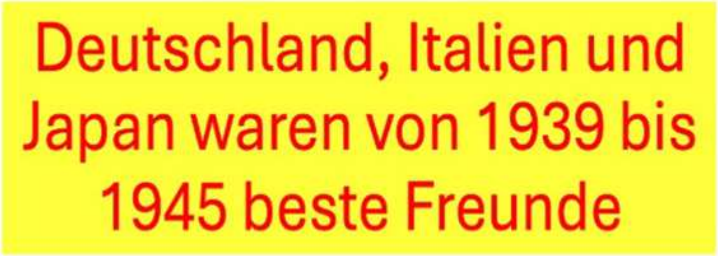
The national dish to the UK is
fried fish and potatoes

pi@gavinteo: ~/examples/lite/examples/object_detection/raspberry_p
File Edit Tabs Help

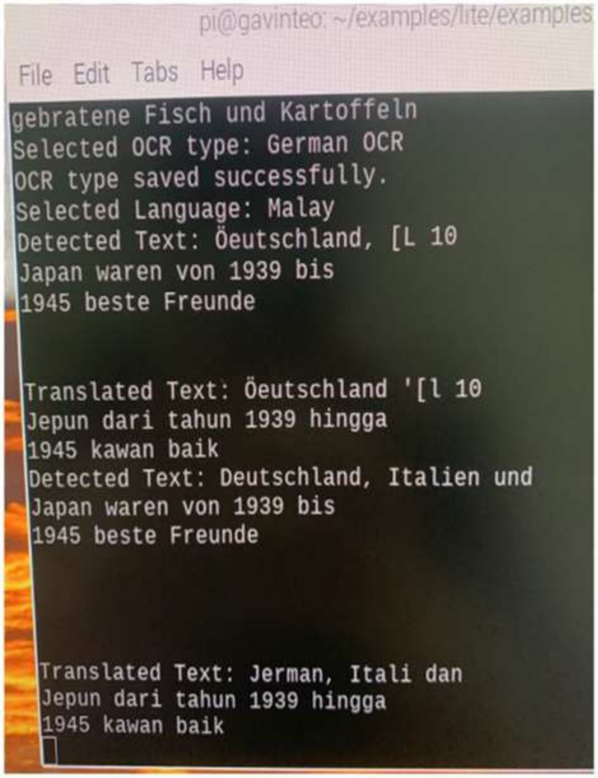
```
ranslated Text: National Dish in Großbritannien ist
ed Fisch und Kartoffeln
etected Text: a

he national disn to the UKis
ried fish and potatoes



Translated Text: A
Der nationale Down in Großbritannien ist
gebratene Fisch und Kartoffeln
Detected Text: F The national dish to the UK is
fried fish and potatoes



Translated Text: F Das Nationalgericht nach Großbritannien ist
gebratene Fisch und Kartoffeln
```

Figure 36

Translation from German to Malay (With yellow background, red font)



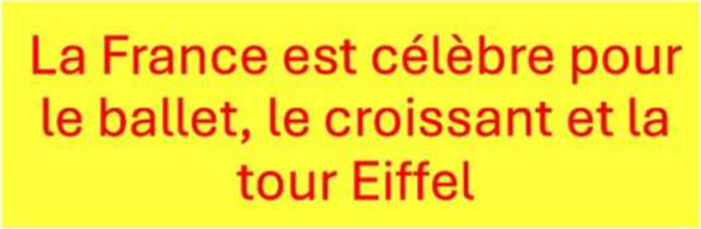
Deutschland, Italien und
Japan waren von 1939 bis
1945 beste Freunde



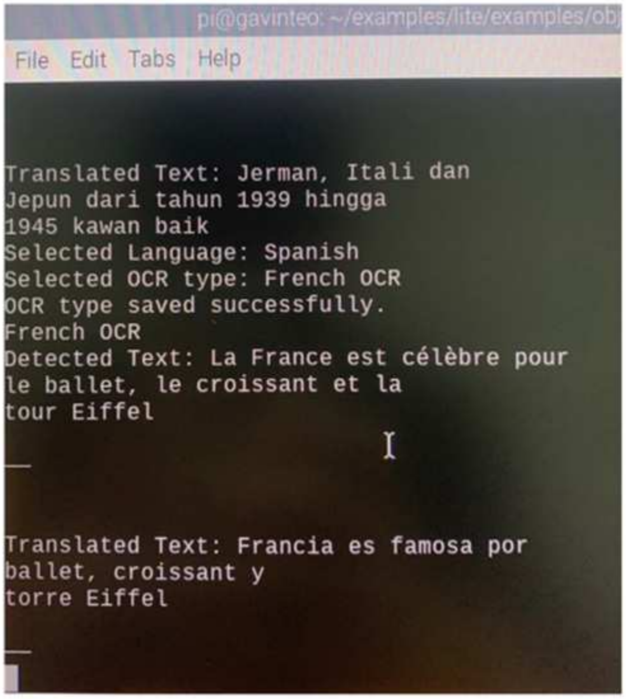
```
gebratene Fisch und Kartoffeln
Selected OCR type: German OCR
OCR type saved successfully.
Selected Language: Malay
Detected Text: Öeutschland, [L 10
Japan waren von 1939 bis
1945 beste Freunde


Translated Text: Öeutschland '[l 10
Jepun dari tahun 1939 hingga
1945 kawan baik
Detected Text: Deutschland, Italien und
Japan waren von 1939 bis
1945 beste Freunde



Translated Text: Jerman, Itali dan
Jepun dari tahun 1939 hingga
1945 kawan baik
```

Figure 37

Translation from French to Spanish (With yellow background, red font)



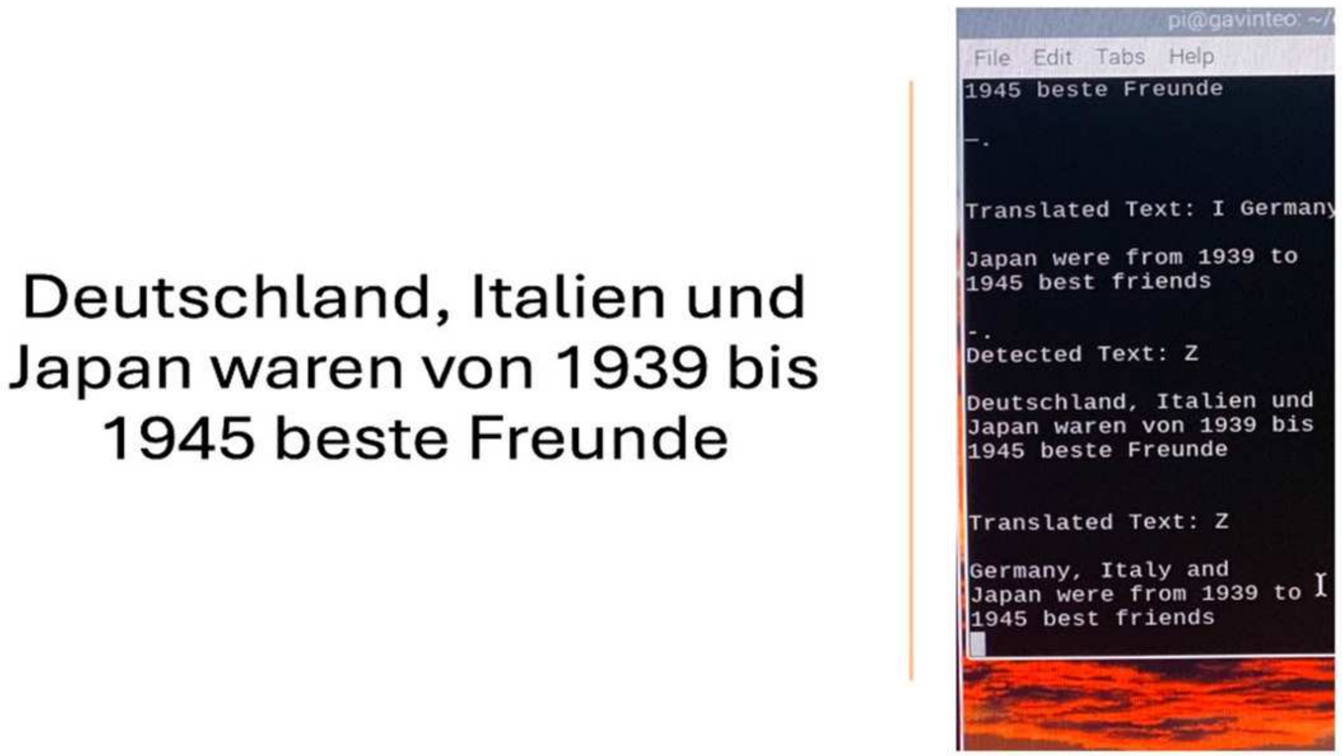
La France est célèbre pour
le ballet, le croissant et la
tour Eiffel



```
Translated Text: Jerman, Itali dan
Jepun dari tahun 1939 hingga
1945 kawan baik
Selected Language: Spanish
Selected OCR type: French OCR
OCR type saved successfully.
French OCR
Detected Text: La France est célèbre pour
le ballet, le croissant et la
tour Eiffel

__


Translated Text: Francia es famosa por
ballet, croissant y
torre Eiffel

__
```

Figure 38

Translation from German to English (With white background, black font)

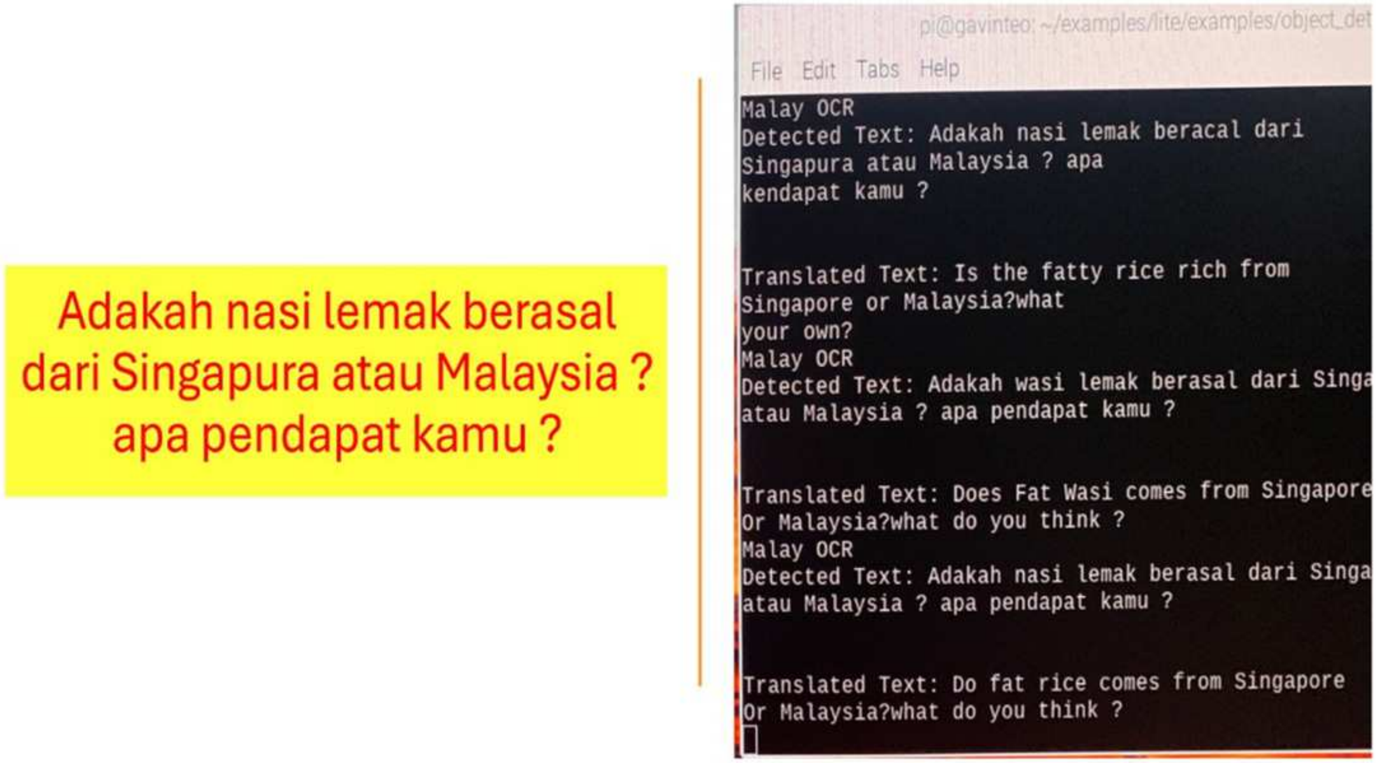


Deutschland, Italien und
Japan waren von 1939 bis
1945 beste Freunde

```
1945 beste Freunde

-.


Translated Text: I Germany

Japan were from 1939 to
1945 best friends

-.
Detected Text: Z

Deutschland, Italien und
Japan waren von 1939 bis
1945 beste Freunde


Translated Text: Z

Germany, Italy and
Japan were from 1939 to
1945 best friends
```

Figure 39

Translation from Malay to English (With yellow background, red font)

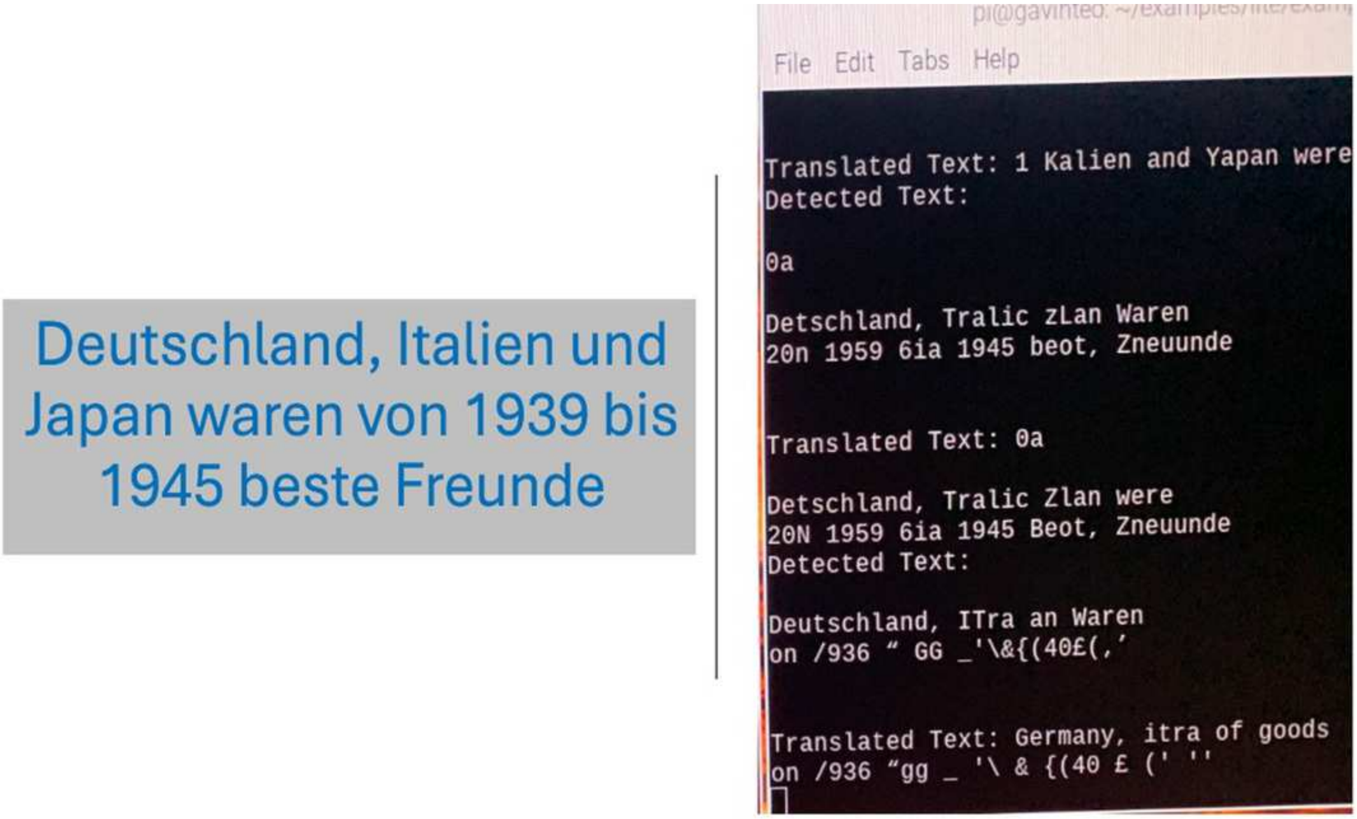


Adakah nasi lemak berasal
dari Singapura atau Malaysia ?
apa pendapat kamu ?

```
Malay OCR
Detected Text: Adakah nasi lemak beracal dari
Singapura atau Malaysia ? apa
kendapat kamu ?


Translated Text: Is the fatty rice rich from
Singapore or Malaysia?what
your own?
Malay OCR
Detected Text: Adakah wasi lemak berasal dari Singa
atau Malaysia ? apa pendapat kamu ?


Translated Text: Does Fat Wasi comes from Singapore
Or Malaysia?what do you think ?
Malay OCR
Detected Text: Adakah nasi lemak berasal dari Singa
atau Malaysia ? apa pendapat kamu ?


Translated Text: Do fat rice comes from Singapore
Or Malaysia?what do you think ?
```

Figure 40

Translation from German to English (Unsuccessful) (With grey background, blue font)

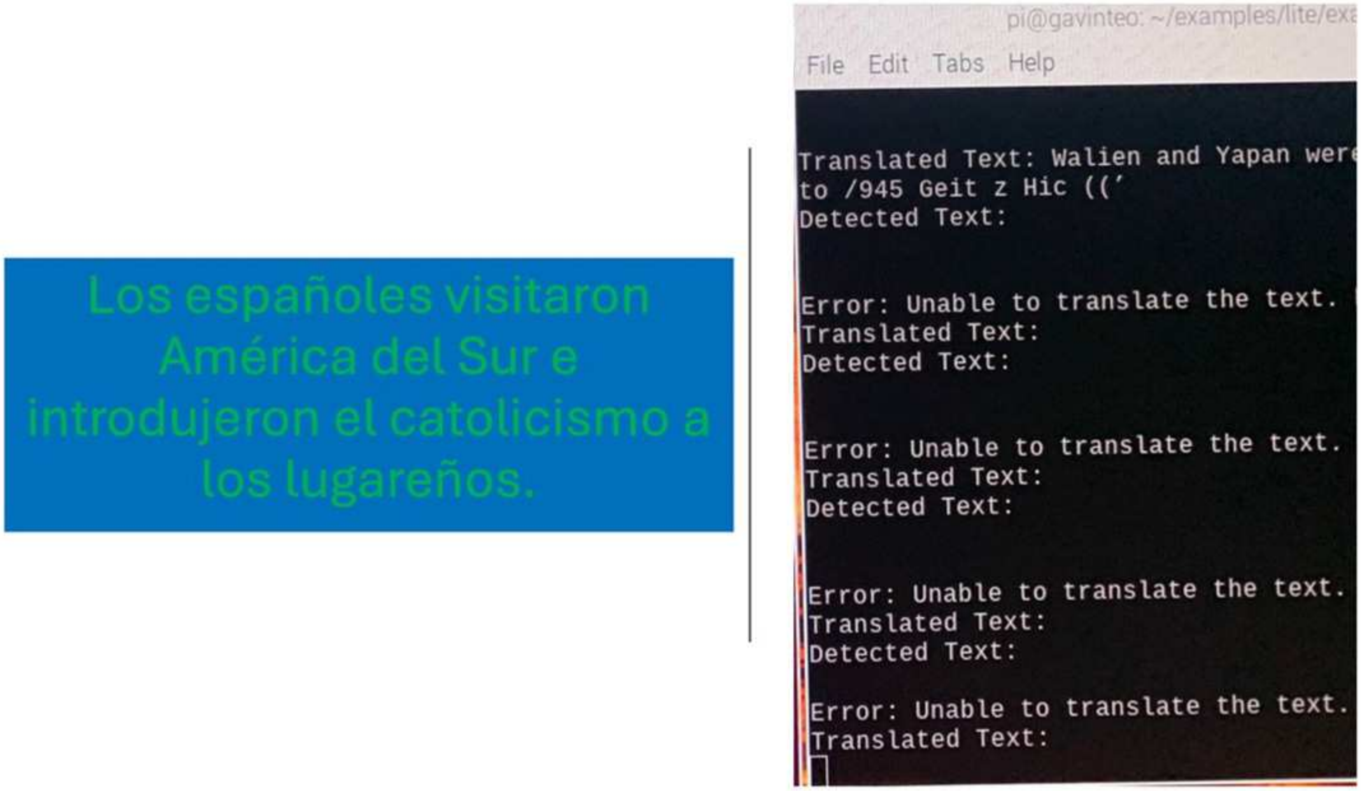


Deutschland, Italien und Japan waren von 1939 bis 1945 beste Freunde

```
Translated Text: 1 Kalien and Yapan were
Detected Text:

0a

Detschland, Tralic zLan Waren
20n 1959 6ia 1945 beot, Zneuunde


Translated Text: 0a

Detschland, Tralic Zlan were
20N 1959 6ia 1945 Beot, Zneuunde
Detected Text:

Deutschland, ITra an Waren
on /936 “ GG _'\&{(40£(,’


Translated Text: Germany, itra of goods
on /936 “gg _ '\ & {(40 £ (' ''
```

Figure 41

Translation from Spanish to English (Unsuccessful) (With blue background, grey font)

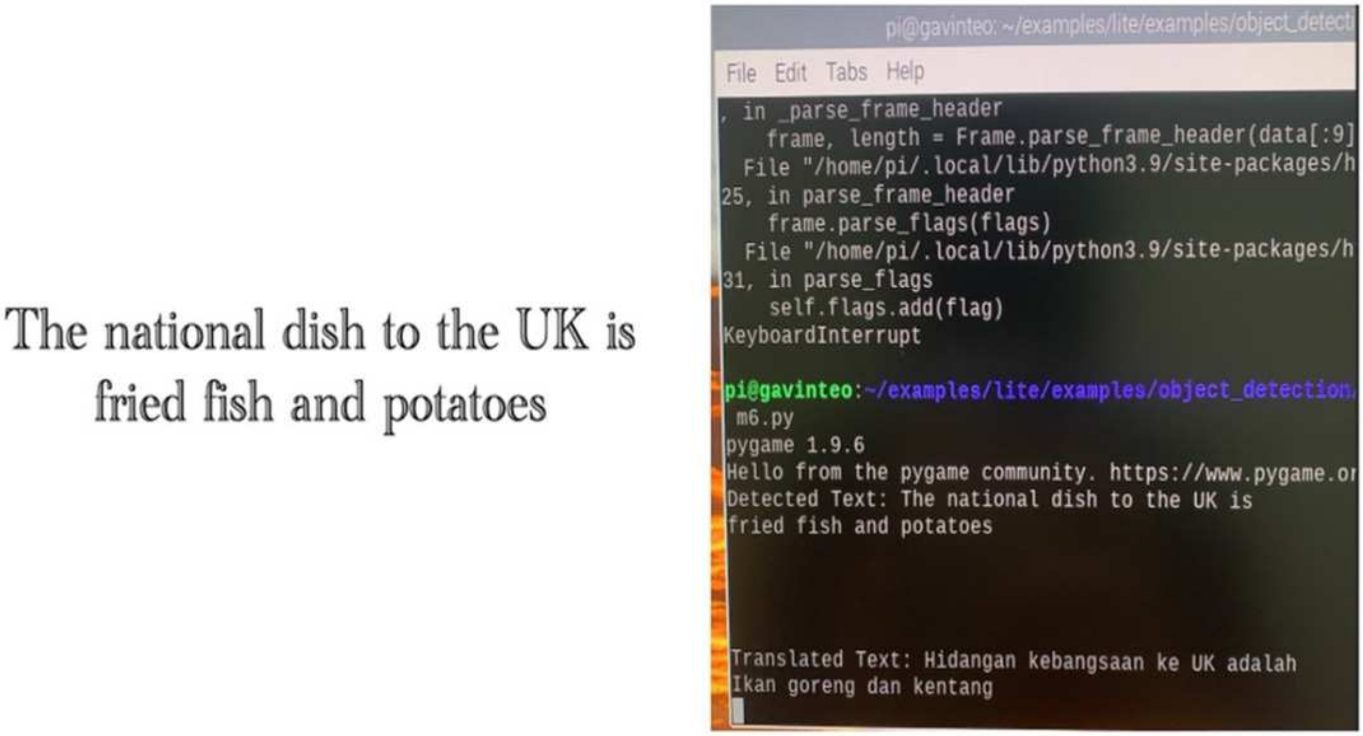


Los españoles visitaron América del Sur e introdujeron el catolicismo a los lugareños.

```
Translated Text: Walien and Yapan wer
to /945 Geit z Hic ((’
Detected Text:


Error: Unable to translate the text.
Translated Text:
Detected Text:


Error: Unable to translate the text.
Translated Text:
Detected Text:


Error: Unable to translate the text.
Translated Text:
Detected Text:

Error: Unable to translate the text.
Translated Text:
```

Figure 42

Translation from English to Malay (With white background, black font)

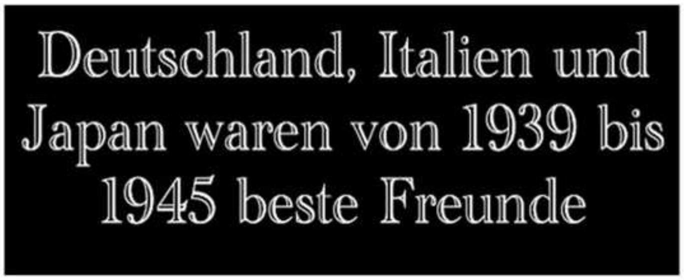


The national dish to the UK is
fried fish and potatoes

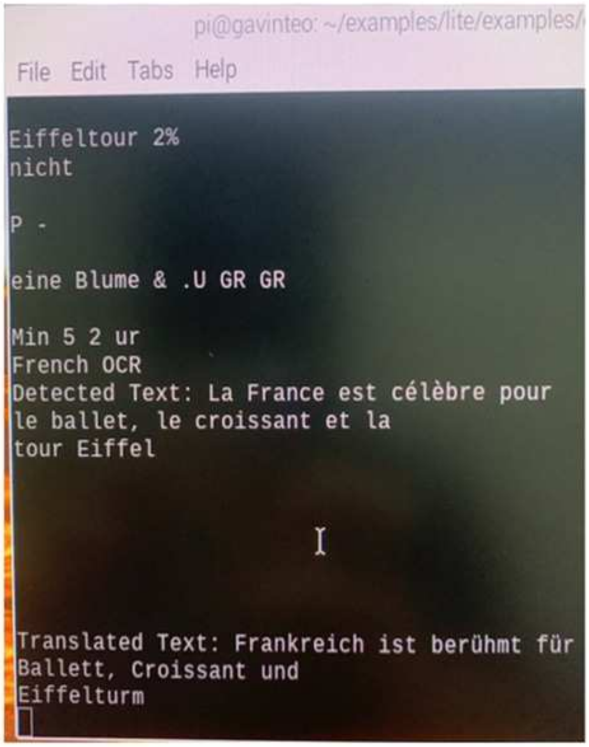
```
, in _parse_frame_header
    frame, length = Frame.parse_frame_header(data[:9]
  File "/home/pi/.local/lib/python3.9/site-packages/h
25, in parse_frame_header
    frame.parse_flags(flags)
  File "/home/pi/.local/lib/python3.9/site-packages/h
31, in parse_flags
    self.flags.add(flag)
KeyboardInterrupt

pi@gavinteo:~/examples/lite/examples/object_detection
 m6.py
pygame 1.9.6
Hello from the pygame community. https://www.pygame.or
Detected Text: The national dish to the UK is
fried fish and potatoes



Translated Text: Hidangan kebangsaan ke UK adalah
Ikan goreng dan kentang
```

Figure 43

Translation from French to German (With black background, white font)



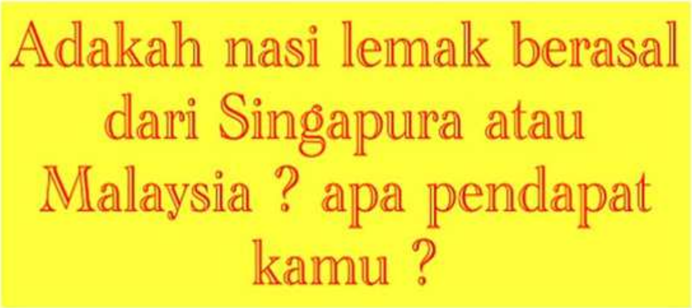
Deutschland, Italien und Japan waren von 1939 bis 1945 beste Freunde



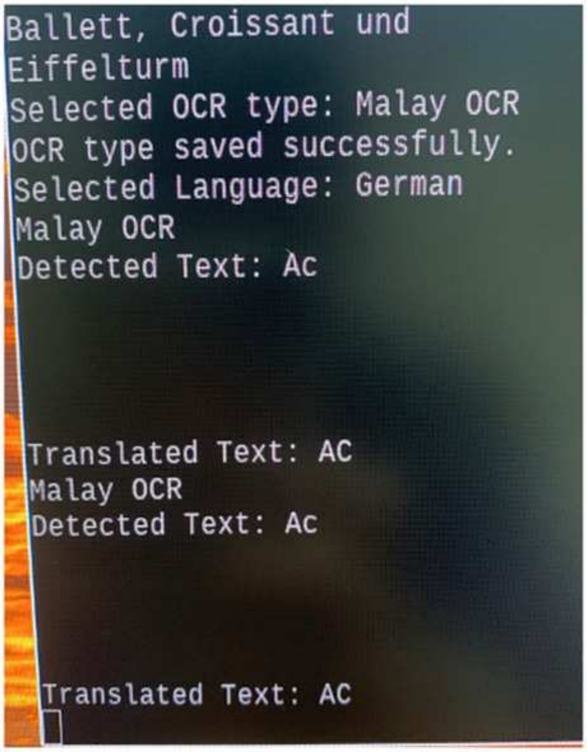
```
Eiffeltour 2%
nicht

P -

eine Blume & .U GR GR

Min 5 2 ur
French OCR
Detected Text: La France est célèbre pour
le ballet, le croissant et la
tour Eiffel



Translated Text: Frankreich ist berühmt für
Ballett, Croissant und
Eiffelturm
```

Figure 44

Translation from Malay to German (Unsuccessful) (With yellow background, red font)



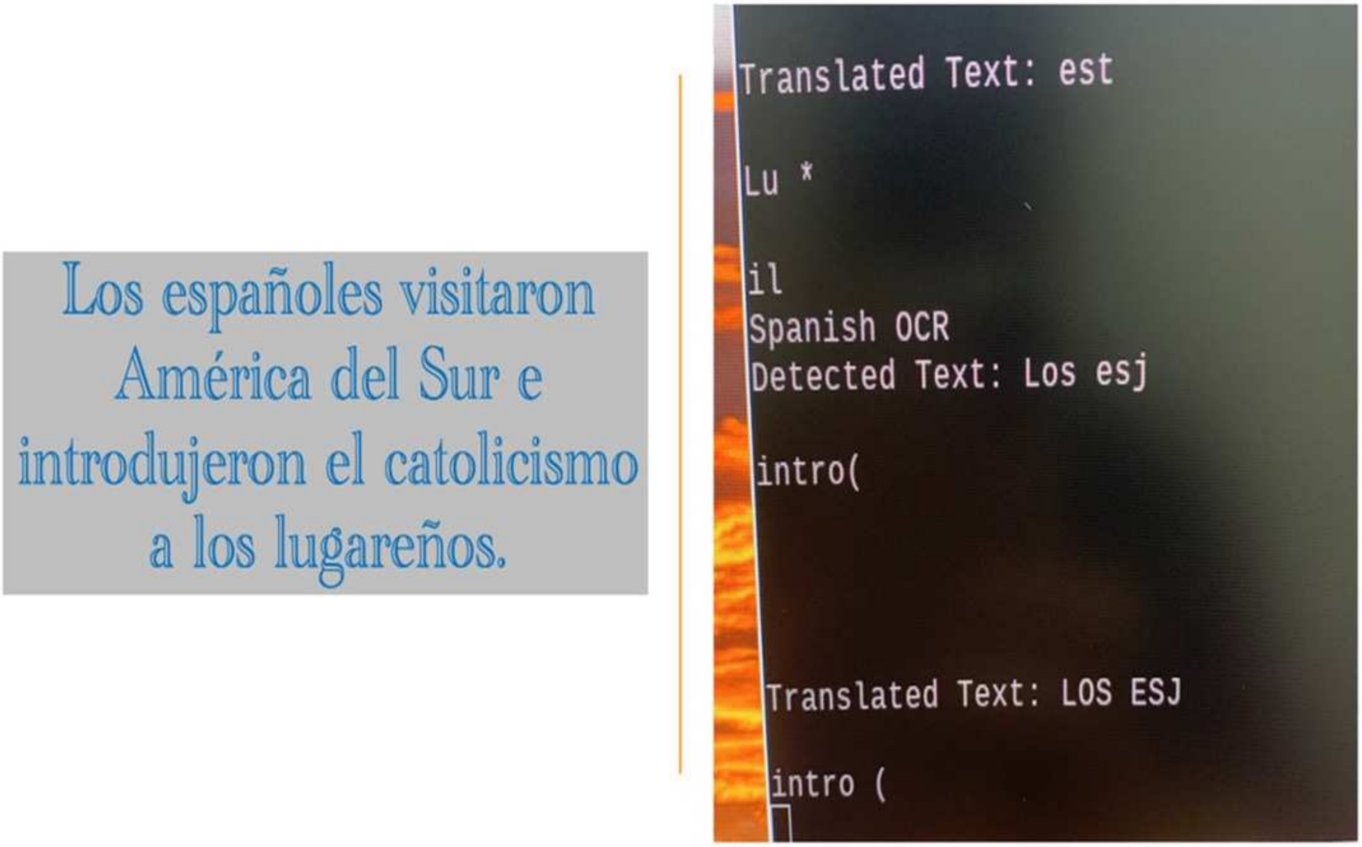
Adakah nasi lemak berasal dari Singapura atau Malaysia ? apa pendapat kamu ?



```
Ballett, Croissant und
Eiffelturm
Selected OCR type: Malay OCR
OCR type saved successfully.
Selected Language: German
Malay OCR
Detected Text: Ac



Translated Text: AC
Malay OCR
Detected Text: Ac



Translated Text: AC
```

Figure 45

Translation from Spanish to French (Unsuccessful) (With grey background, blue font)

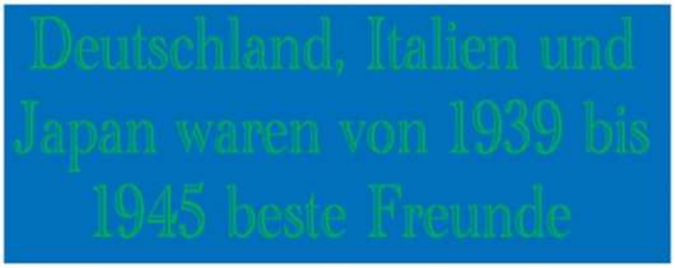


Los españoles visitaron
América del Sur e
introdujeron el catolicismo
a los lugareños.

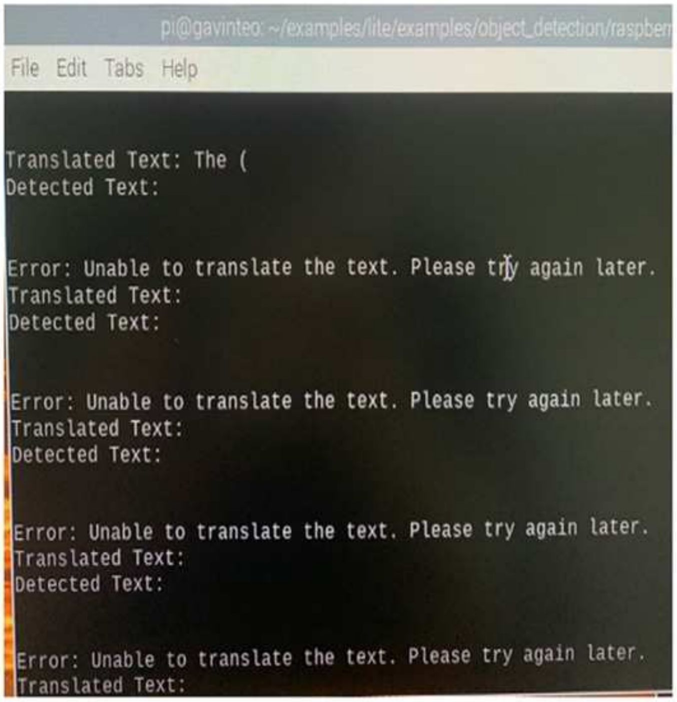
```
Translated Text: est

Lu *

il
Spanish OCR
Detected Text: Los esj

intro(



Translated Text: LOS ESJ

intro (
```

Figure 46

Translation from German to Malay (Unsuccessful) (With blue background, green font)



Deutschland, Italien und
Japan waren von 1939 bis
1945 beste Freunde

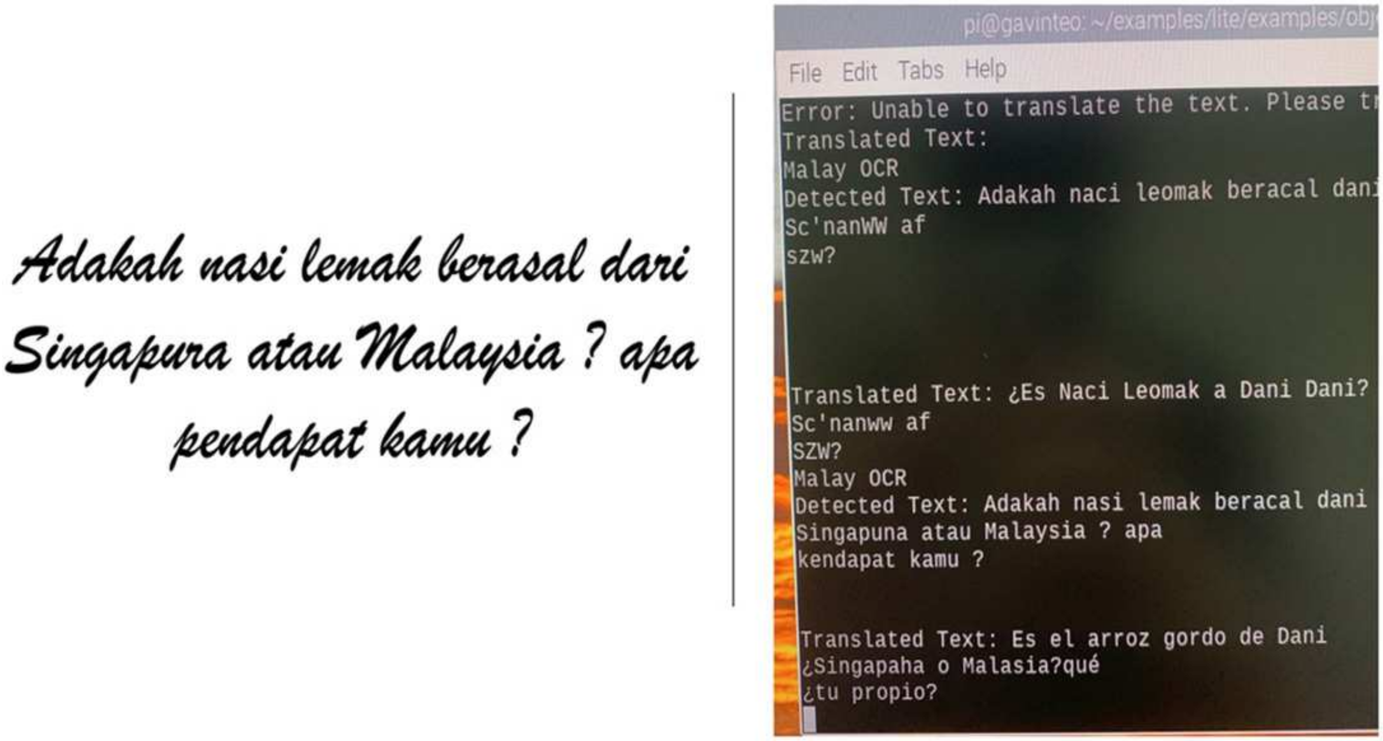


pi@gavinteo: ~/examples/lite/examples/object_detection/raspber
File Edit Tabs Help

```
Translated Text: The (
Detected Text:


Error: Unable to translate the text. Please try again later.
Translated Text:
Detected Text:


Error: Unable to translate the text. Please try again later.
Translated Text:
Detected Text:


Error: Unable to translate the text. Please try again later.
Translated Text:
Detected Text:


Error: Unable to translate the text. Please try again later.
Translated Text:
```

Figure 47

Translation from Malay to Spanish (With white background, black font)

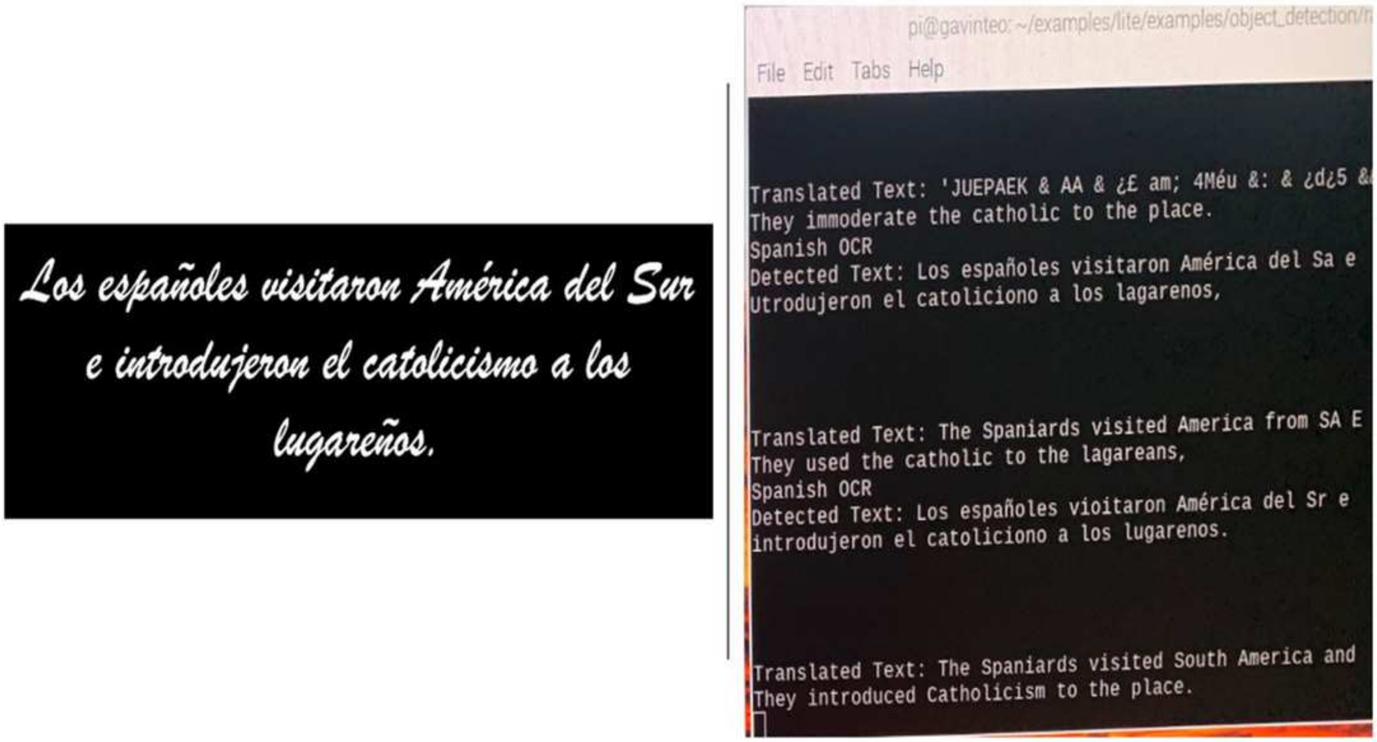


Adakah nasi lemak berasal dari
Singapura atau Malaysia ? apa
pendapat kamu ?

```
Error: Unable to translate the text. Please tr
Translated Text:
Malay OCR
Detected Text: Adakah naci leomak beracal dani
Sc'nanWW af
szw?



 Translated Text: ¿Es Naci Leomak a Dani Dani?
 Sc'nanww af
 SZW?
 Malay OCR
 Detected Text: Adakah nasi lemak beracal dani
 Singapuna atau Malaysia ? apa
 kendapat kamu ?


 Translated Text: Es el arroz gordo de Dani
 ¿Singapaha o Malasia?qué
 ¿tu propio?
```

Figure 48

Translation from Spanish to English (With black background, white font)

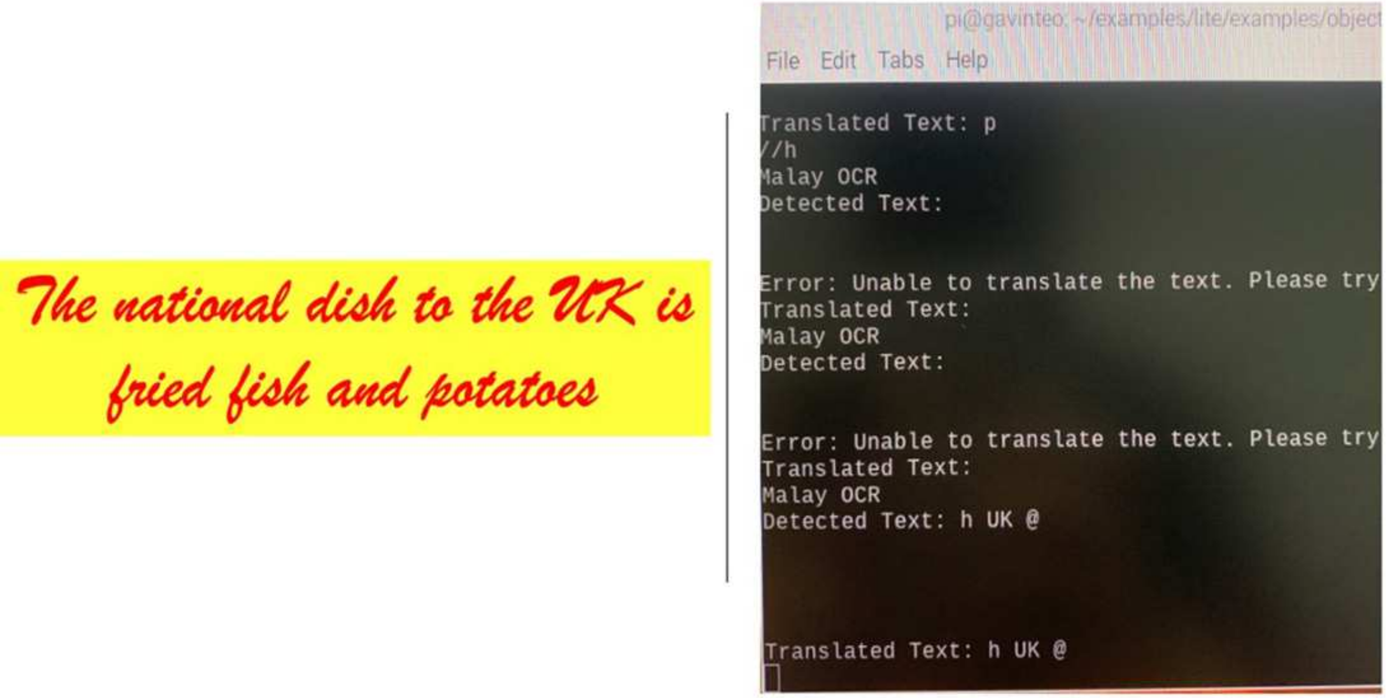


Los españoles visitaron América del Sur
e introdujeron el catolicismo a los
lugareños.

```
Translated Text: 'JUEPAEK & AA & ¿£ am; 4Méu &: & ¿d¿5 &
They immoderate the catholic to the place.
Spanish OCR
Detected Text: Los españoles visitaron América del Sa e
Utrodujeron el catoliciono a los lagarenos,



Translated Text: The Spaniards visited America from SA E
They used the catholic to the lagareans,
Spanish OCR
Detected Text: Los españoles vioitaron América del Sr e
introdujeron el catoliciono a los lugarenos.



Translated Text: The Spaniards visited South America and
They introduced Catholicism to the place.
```

Figure 49

Translation from English to Malay (Unsuccessful) (With yellow background, red font)

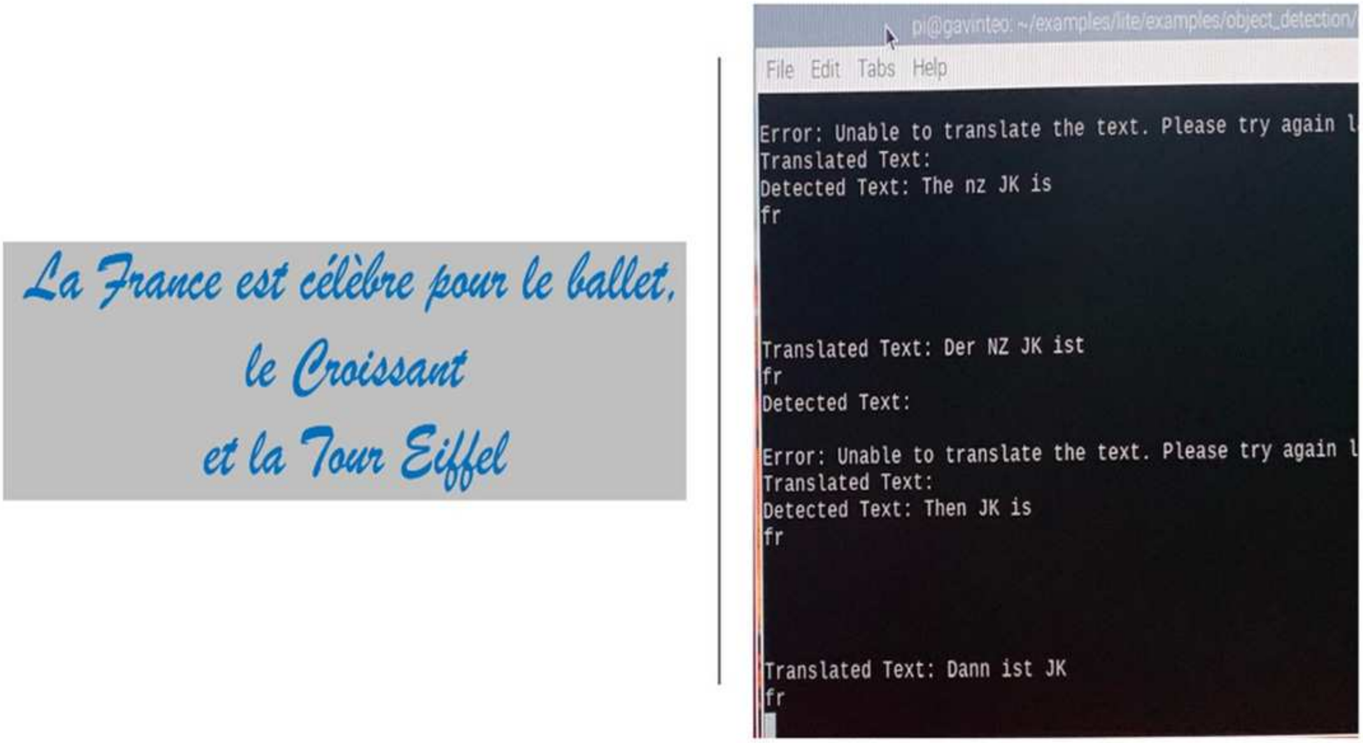


The national dish to the UK is fried fish and potatoes

```
Translated Text: p
//h
Malay OCR
Detected Text:


Error: Unable to translate the text. Please try
Translated Text:
Malay OCR
Detected Text:


Error: Unable to translate the text. Please try
Translated Text:
Malay OCR
Detected Text: h UK @



Translated Text: h UK @
```

Figure 50

Translation from French to German (Unsuccessful) (With grey background, blue font)

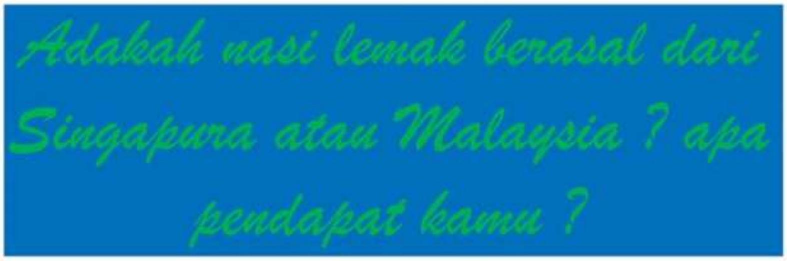


La France est célèbre pour le ballet,
le Croissant
et la Tour Eiffel

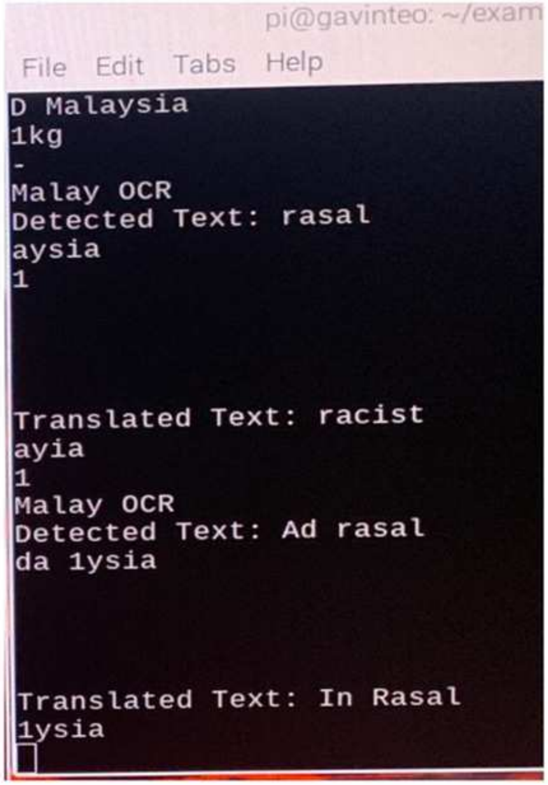
```
Error: Unable to translate the text. Please try again l
Translated Text:
Detected Text: The nz JK is
fr



Translated Text: Der NZ JK ist
fr
Detected Text:

Error: Unable to translate the text. Please try again l
Translated Text:
Detected Text: Then JK is
fr



Translated Text: Dann ist JK
fr
```

Figure 51

Translation from Malay to German (Unsuccessful) (With blue background, green font)



Adakah nasi lemak berasal dari Singapura atau Malaysia ? apa pendapat kamu ?



```
D Malaysia
1kg
-
Malay OCR
Detected Text: rasal
aysia
1



Translated Text: racist
ayia
1
Malay OCR
Detected Text: Ad rasal
da 1ysia



Translated Text: In Rasal
1ysia
```