Development of Scanner Machine for the Blinds and Visually Impaired

Shaharil Bin Mad Saad[1*], Jason Sia Teng Hong[1]

Faculty of Mechanical Engineering Universiti Teknologi Malaysia 81310 UTM Johor Bahru Johor, Malaysia

[*]Corresponding Author

DOI: https://doi.org/10.47772/IJRISS.2026.1026EDU0211

Received: 15 April 2026; Accepted: 20 April 2026; Published: 06 May 2026

ABSTRACT

Technological aid in scanner for blind people and visually impaired is still complicated and considered costly. They lost privacy when ask someone to read their documents for them. Therefore, a scanner machine that is cheap, easy to maintain and privacy secured should be developed. This report describes the development of scanner machine which can perform image capture, optical character recognition (OCR), and text to speech audio output. The objectives of this project include to develop a scanner machine, to develop an algorithm for text recognition based on OCR method, to measure the accuracy of text recognition and to measure the average working time of scanner machine. The scope of this project limited the size of scanning materials, limited the language used for text recognition (English), and the output only audio form. The scanner machine prototype is composed of Raspberry Pi 3B+, camera, dc motor, servo motor, speaker/headphone, and switches. The coding or algorithm is running by the Raspberry Pi 3B+ to receive signal from the switches, perform certain operations and send the output in audio form. this scanner machine able to deal with rotated scanning material in range of -90º to 75º with accuracy greater than 80%. When different font size of scanning materials is used, the scanner machine performs well for font size greater than 13 with accuracy above 85%. Besides, the scanner had overall running time of "book" mode at around 16.58 seconds and overall running time of "normal mode" at around 13.39 seconds.

Keywords: Scanner machine, Privacy, Optical Character Recognition, Text to Speech, Raspberry Pi

INTRODUCTION

There has been a steady growth of advancement in developing assistive technologies for scanner. However, technological aid in scanner for blind people is still complicated and considered costly or often gets overlooked while it is mostly unreachable for many of them in this world especially in teaching and learning factor. It is estimated that 9.03% out of 14,289 of Malaysian have a vision impaired and blindness. In Malaysia, the 1.2% of people are blind, 1.0% of people are suffering from severe visual impairment, and 5.9% of people are suffering from moderate visual impairment [1].

A visually impaired student cannot use sight to gain and learn information. Thus, other sense like touch and hearing is the only way for them to learn. Hence, learning, teaching, and reading becomes a challenge for the blinds or visually impaired student and the teacher who teaching them. When blind or visually impaired student deal with document, books, and letters without braille code, they are not able to know any information from them without a user-friendly scanner for them. Besides, when blinds or visually impaired received letter or other personal document, they are incapable to read the information on it. Hence, they need to ask help from other people who are able to read. Hence, they lose their privacy which should be a humans' right. This project aims to tackle these problems to create a better future and learning method, and privacy protection for every blind or visually impaired.

The problem statements of this project are expensive scanners for blinds and visually impaired (400+ USD to 5000+ USD), long duration of scanner fixing, and loss of privacy by blinds and visually impaired. The objectives of this project include develop a scanner machine, develop an algorithm for text recognition based on OCR method, measuring accuracy of text recognition from scanning materials, and measuring average running time of the scanner machine.

METHODOLOGY

The project is started from the literature review in order to identify the problem definition. Not only the problem definition, the required features of the scanner machine will be identified from the literature review which included the buttons used, scanning functions, image processing features, OCR libraries, and text to speech methods. After identifying the feature, the scanner machine hardware is developed by identifying the components used and other features. The algorithm of scanning machine is developed also. After fabricating the scanner machine hardware and developed the scanning algorithm, all the components are assembled and integrated with the coding. The performance testing will be carried out in the term of functionality of the scanner machine and also the scanning algorithm. The redesign of the algorithm will be carried out if any unexpected result occurred during the testing. The discussion and conclusion will be done only if the result is accepted.

Overall System Description

The Raspberry Pi 3 Model B+ 1 GB SDRAM is used as the main controller to control the scanner machine for the blind and visually impaired. Raspberry Pi is the "Brain" of the scanner machine and it will read the signal from GPIO pin that connected to the switches, perform required operation and send the signal to the camera, interpret image send back by camera, and also output audio signal to the speaker. In this project, different type of switches is used to ease the user to differentiate the operation switch. The coding installed in the Raspberry Pi 3 B+ will control the camera, dc motor, and servo motor according to the mode that control by one of the switches. The scanner machine will be operated only if the main switch is turn on. The camera will take photo of scanning material and send it to Raspberry Pi for further processing. The camera also needs to have high resolution of results. The Raspberry Pi will then perform image processing, optical character recognition performed by pytesseract library and transform text to speech using pyttsx3 library regardless modes. DC motor and servo motor will only operate during book mode for page flipping purpose. The speaker will be activated automatically at to inform the user current progress of scanning and also give the final output of scanning results.

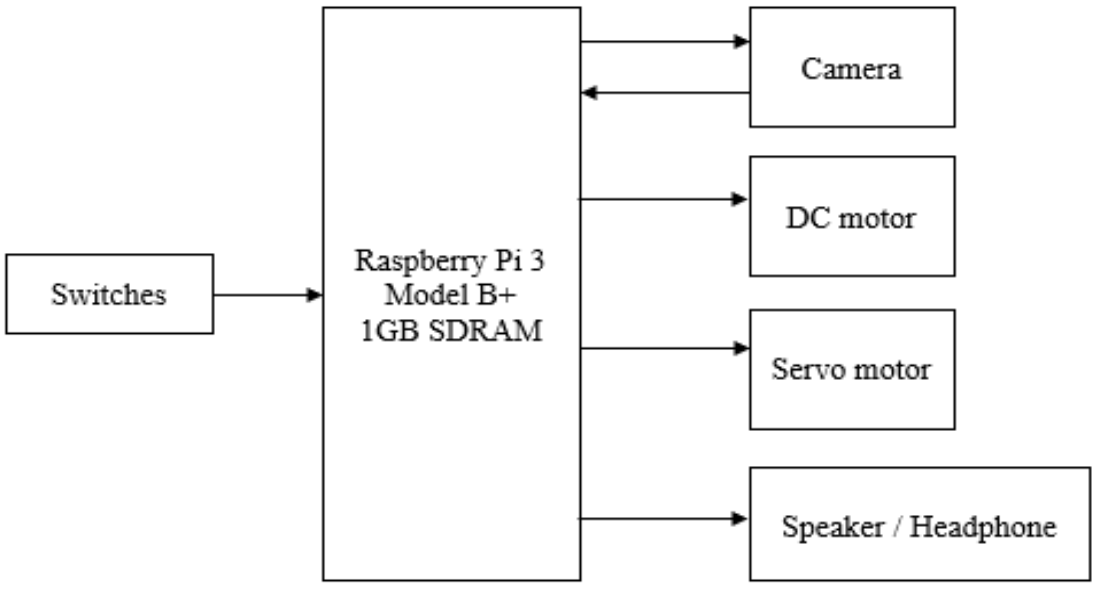

Figure 1: The block diagram of scanner machine.

Instrumentations, Tools, Materials

The instrumentation, tools and materials used in this project included Raspberry Pi Model 3B+, power adapter, plug, Monitor, HDMI wire, SD card, SD card reader, laptop , camera , speaker / headphone, DC motor, servo motor, aluminum composite board, PVC pipe, PVC pipe 90 degrees elbow, 90 degrees angle bracket, wood bar, wood stick, hydraulic press machine, drilling machine, cordless hand drill, rivet gun, rivet, jigsaw machine, hand grinder, bolts, nuts, painting aerosol can, glue, wires, and relay.

Scanner Machine Algorithm

First, the algorithm will import required library and modules for functions of scanner machine. Then the camera is setup for scanning purpose. The algorithm will initialize and setup text to speech engine for audio output too. GPIO pin for switch signal receiving and control dc motor and servo motor will be setup before the scanner is ready to perform scanning purpose. When the algorithm enters the main while loop it will check the activation of toggle switch if the switch is activated set main active to true. When main active is true, the activation of push on and off button is checked to differentiate the modes. When push on and off button is activated, the mode is "book" mode. In this mode, when push button is pushed for the first time, camera will take an image of scanning area. Then the image is sent back to Raspberry Pi and the image will be process to clear and sharp image. The unwanted part of the image is cropped away and the image is cut half in middle into left part and right part. Next the algorithm will perform optical character recognition on the final images. After process the images, error checking and convert to audio speech. The algorithm will back to main loop until the power supply is off. If the push button is pushed again, the dc motor and servo motor is activated to flip the page to the next page, follow by same scanning procedure as shown above for "book" mode. When push on and off button is not activated, the mode is "normal" mode. When push button is pushed, camera will capture image. Raspberry Pi will preprocess the image. The color of image is inverted for contour finding. Then the biggest contour is determined. Next, perspective transform is applied to the biggest contour (which is the image of scanning material). After that, adaptive thresholding is applied on the transformed image. The algorithm will then perform optical character recognition on the final image, error checking and convert to audio speech. Finally, the algorithm will back to main loop until the power supply is off.

Libraries Selected for Algorithm and One Significant Advantage

The first library selected is RPI.GPIO where it simplifies GPIO access on the Raspberry Pi, providing a Pythonic interface to interact with the GPIO pins without the need for complex low-level programming. The second library is time library which allow pausing of execution for a specified duration enabling to introduce delays or control the timing of certain operations. The third library used is subprocess library. The subprocess library allows you to execute external commands and scripts from your Python program, giving you the ability to interact with the command-line interface. The fourth library used is Numpy library. NumPy supports broadcasting, which enables efficient element-wise operations between arrays of different shapes. The fifth library used is OpenCV library which provides a comprehensive set of functions and algorithms for image processing and computer vision tasks [2]. It covers a wide range of functionality, from basic image manipulation to advanced feature extraction and machine learning integration [3]. This reduces the need for using multiple libraries and simplifies the development process. The sixth library used is Pytesseract library. This library provides a simple and straightforward API to work with, allowing you to extract text from images with just a few lines of code [4]. The last library used is pyttsx3 library. This library provides a simple and straightforward API for performing text-to-speech conversion [5]. It requires minimal setup and configuration, making it easy to integrate TTS capabilities into this project.

Design of Hardware

The scanner must have a platform that can fit open A4 books. Besides, the platform must have some design for user to navigate and know the scanning area of the scanner machine. Besides, the platform needs to have design for camera holding and also design area for dc motor and servo motor allocation. The scanner machine also need port for power supply, switch location, easy handling, and also space for wire arrangement and storage.

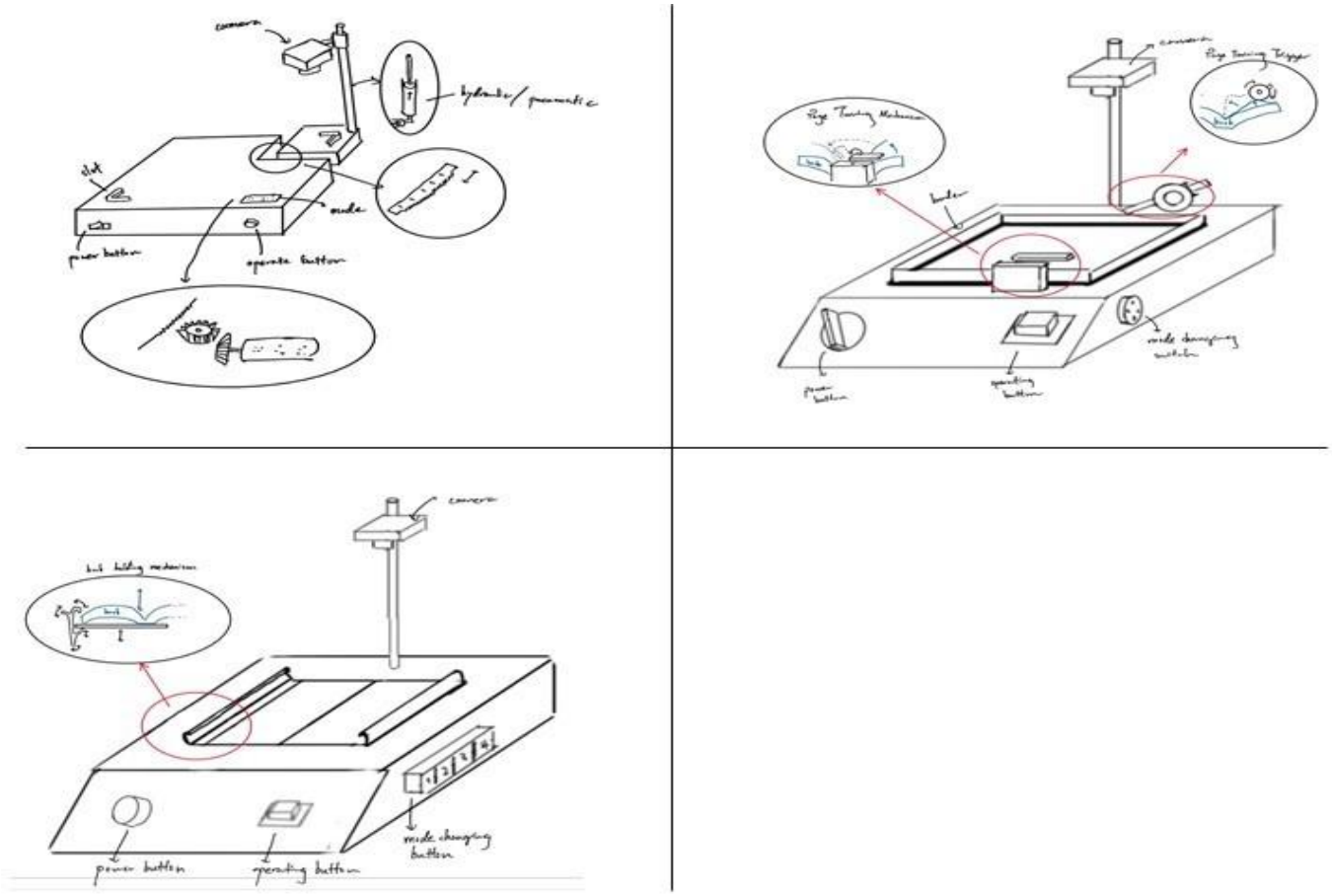

Figure 2: (a) Conceptual design 1. (b) Conceptual design 2. (c) Conceptual design 3.

Table 1: Scoring Matrix of conceptual designs

Selection Criteria	Weight	Concept 1		Concept 2		Concept 3	
		Rating	Weighted Score	Rating	Weighted Score	Rating	Weighted Score
Ease of use for blinds and visually impaired	15%	3	0.45	4	0.6	3	0.45
Size	2%	2	0.04	4	0.08	4	0.08
Weight	2%	2	0.04	3	0.06	4	0.08
Material	3%	4	0.12	4	0.12	4	0.12
Ease of manufacturing	5%	2	0.15	4	0.2	3	0.15
Manufacturing Cost	10%	2	0.1	4	0.4	2	0.2
Maintenance Cost	10%	2	0.1	3	0.3	3	0.3
Environment resistance	5%	3	0.15	3	0.15	3	0.15
Ease of Maintenance	10%	2	0.2	3	0.3	2	0.2
Ease of Assembly	5%	2	0.1	4	0.2	2	0.1
Ease of Disassembly	5%	2	0.1	3	0.15	3	0.15
Product Life Span	10%	3	0.3	3	0.3	3	0.3
portablility	5%	3	0.15	3	0.15	4	0.2
mechanism	10%	4	0.4	4	0.4	4	0.4
Aesthetic/Appearance	1%	4	0.04	4	0.04	4	0.04
Quantity of components	2%	3	0.06	3	0.06	3	0.06
	Total Score	2.5		3.51		2.98	
	Rank	3		1		2	
	Continue?	No		Yes		No	

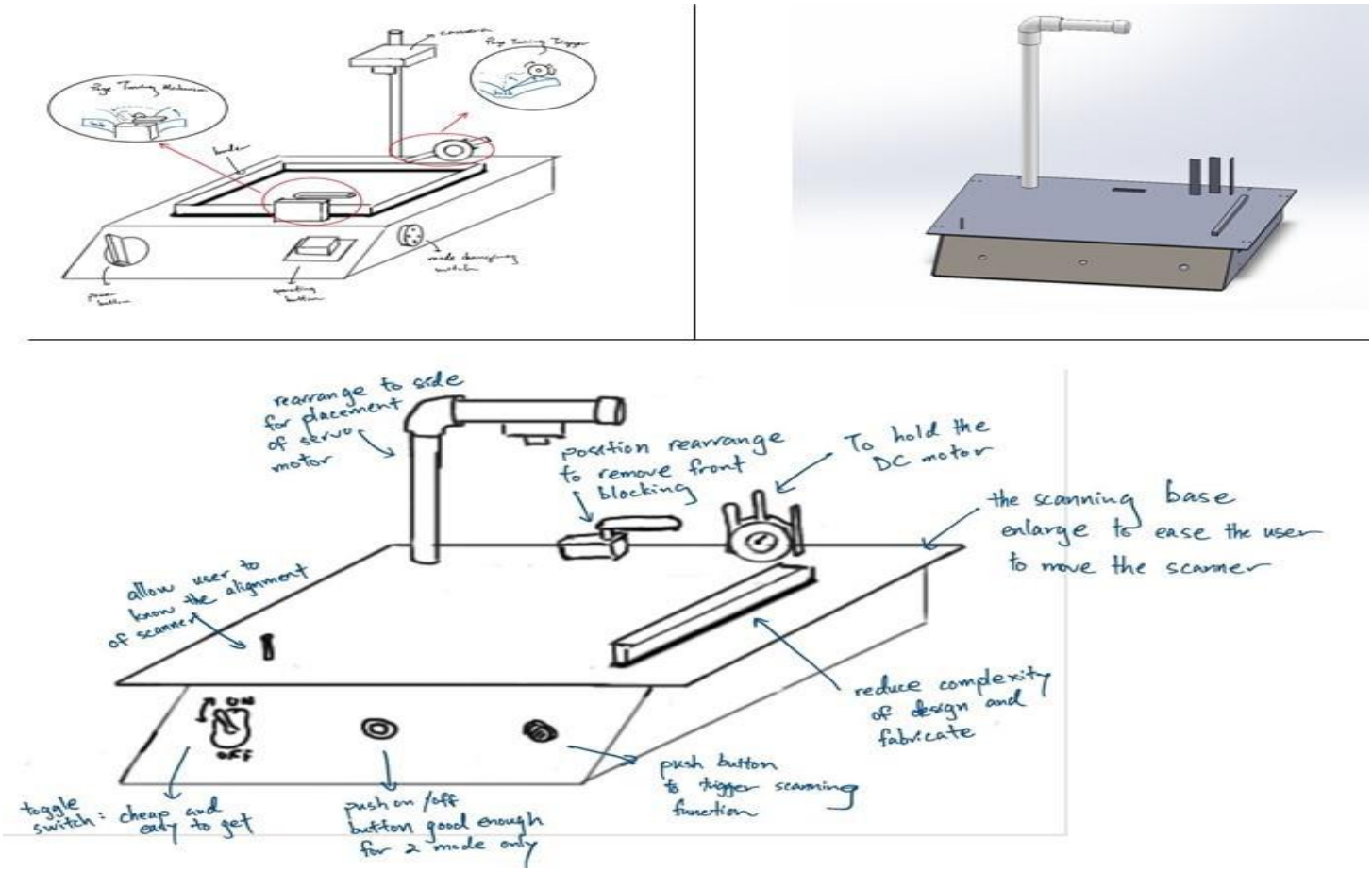

Figure 3: (a) Final conceptual design. (b) Virtual prototype of scanner machine. (c) Final conceptual design after modification.

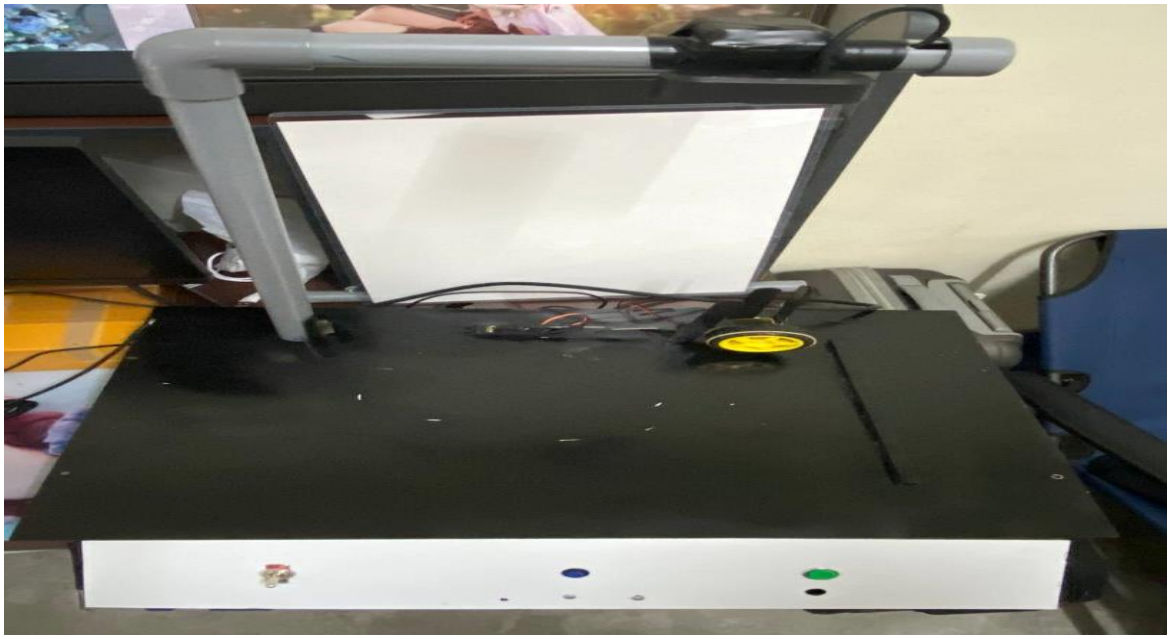

Figure 4: Prototype of scanner machine.

RESULT AND DISCUSSION

The results of the project will be discussed in term of the features of the scanner machine and the functionality test. The features of the scanner machine are the outcomes of the algorithm. The functionality test will prove the switches, algorithm, and page turning mechanism are able to function included with accuracy test.

Image Reading Test

The OCR engine (pytesseract) able to read image format of TIFF (Tagged Image File Format), BMP (Bitmap Image File), JPEG (Joint Photographic Experts Groups), JPEG, and PNG (Portable Network Graphics) except GIF (Graphics Interchange Format). Besides, the OCR method of scanner machine is able to read different font formats such as bold, italic, underline, and text shadow.

Functionality Test of Common Image Pre-processing Feature

The common image pre-processing features that are used in the image include invert the color of the image by using "bitwise_not" function. Next is to change the colour from BGR (Blue, Green, Red) color space to grayscale and the process is called binarization. After that is remove noise from the image which is perform by "cv2.threshold" function, "cv2.morphologyEx" function, and "cv2.medianBlur" function. Then the content in the image is undergo either erosion operation to thinning the font of image or undergo dilation to thickening the font in the image. Besides, the algorithm can also remove unwanted border of the image or adding border to the image. Lastly, the font in the image can be deskew through a series of process by "np. arctan2" function which count the skew angle, "cv2.getRotationMatrix2D" function to get the center point and desired angle of rotation, and "cv2.warpAffine" function to perform the rotation transformation of image. The experiment on image deskew show that the working range of the feature is between -90 º to 75 º.

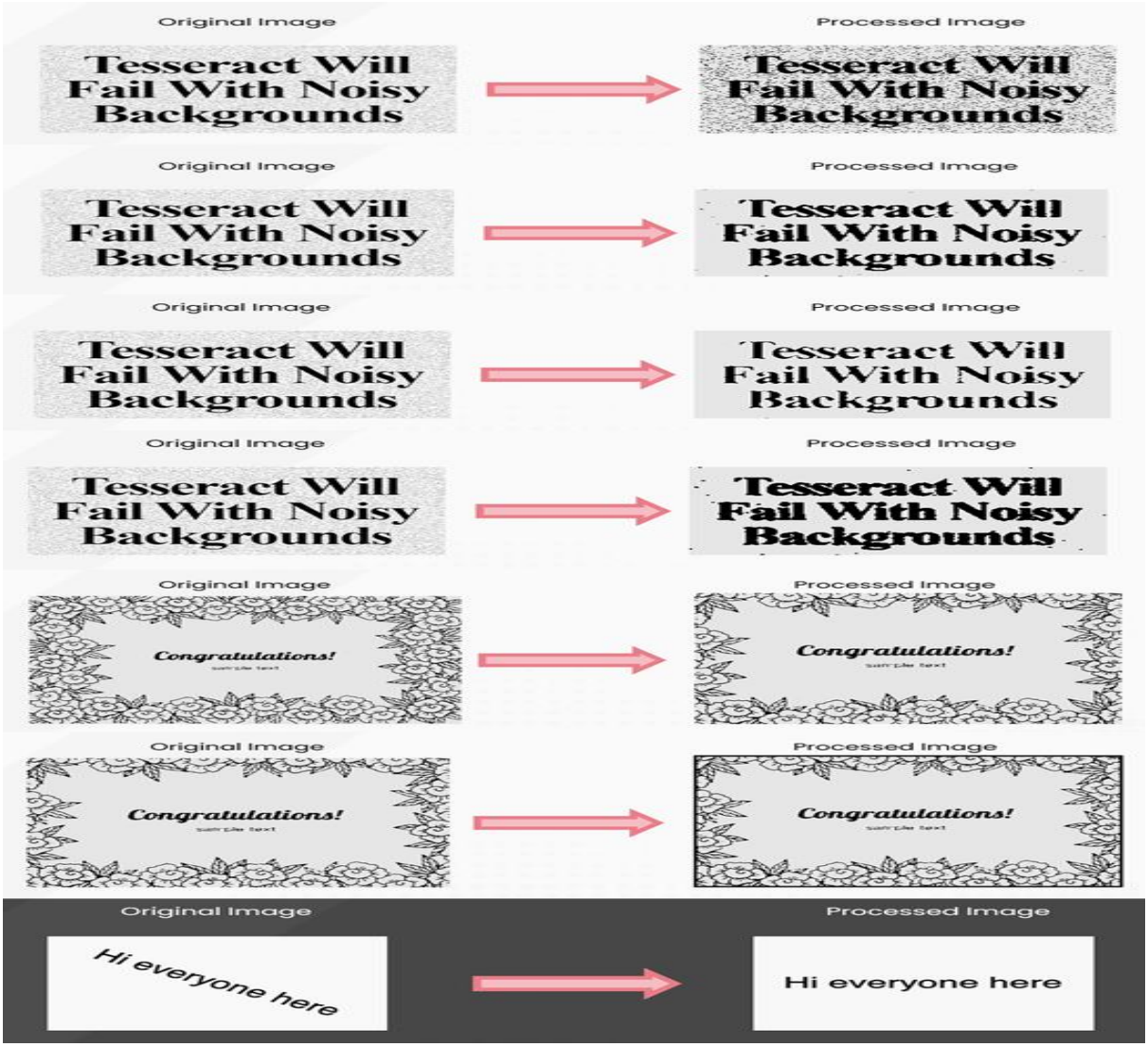

Figure 5: (a) Result of color inverted function. (b)Result of binarization. (c) Result of noise removal. (d) Result of thinner font. (e) Result of thicker font. (f) Result of remove border. (g) Result of adding border. (h) Result of image deskewing.

Functionality Test of Special Image Pre-processing Feature for Book Mode

Since the book material is usually big. Hence, the book is limit in the border we prepared. Hence, some image processes for book mode are easier than normal mode as unwanted part can be cropped away easily. However, the book is limit to A4 size book as the default parameters set for cropping each side of the image is pre-set and need to change the coding if want to change other size of book or need to improve the cropping mechanism to become more intelligent. In order to output the content of book separately according pages. The cropped image should divide into half from middle so that we can produce OCR and TTS output for the left page and right page in sequence.

Figure 6: (a) Original image. (b) Result of image cropping. (c) Result of left page. (d) Result of right page.

Functionality Test of Special Image Pre-processing Feature for Normal Mode

The color of image is inverted and the image is undergoing a series of common image processing. Then the contour is determined and drawn. Next the biggest contour which is our scanning material is point out. Then, a perspective transformation is being calculated. Pts1 is a NumPy array containing the points of the biggest contour, converted to the float32 data type. Pts2 represents the destination points for the transformation, defining a rectangle in the output image. The cv2.getPerspectiveTransform function calculates the perspective transformation matrix based on the source (pts1) and destination (pts2) points. After that, the perspective transformation is applied to the input image using the transformation matrix (matrix) and the image finished this image processing.

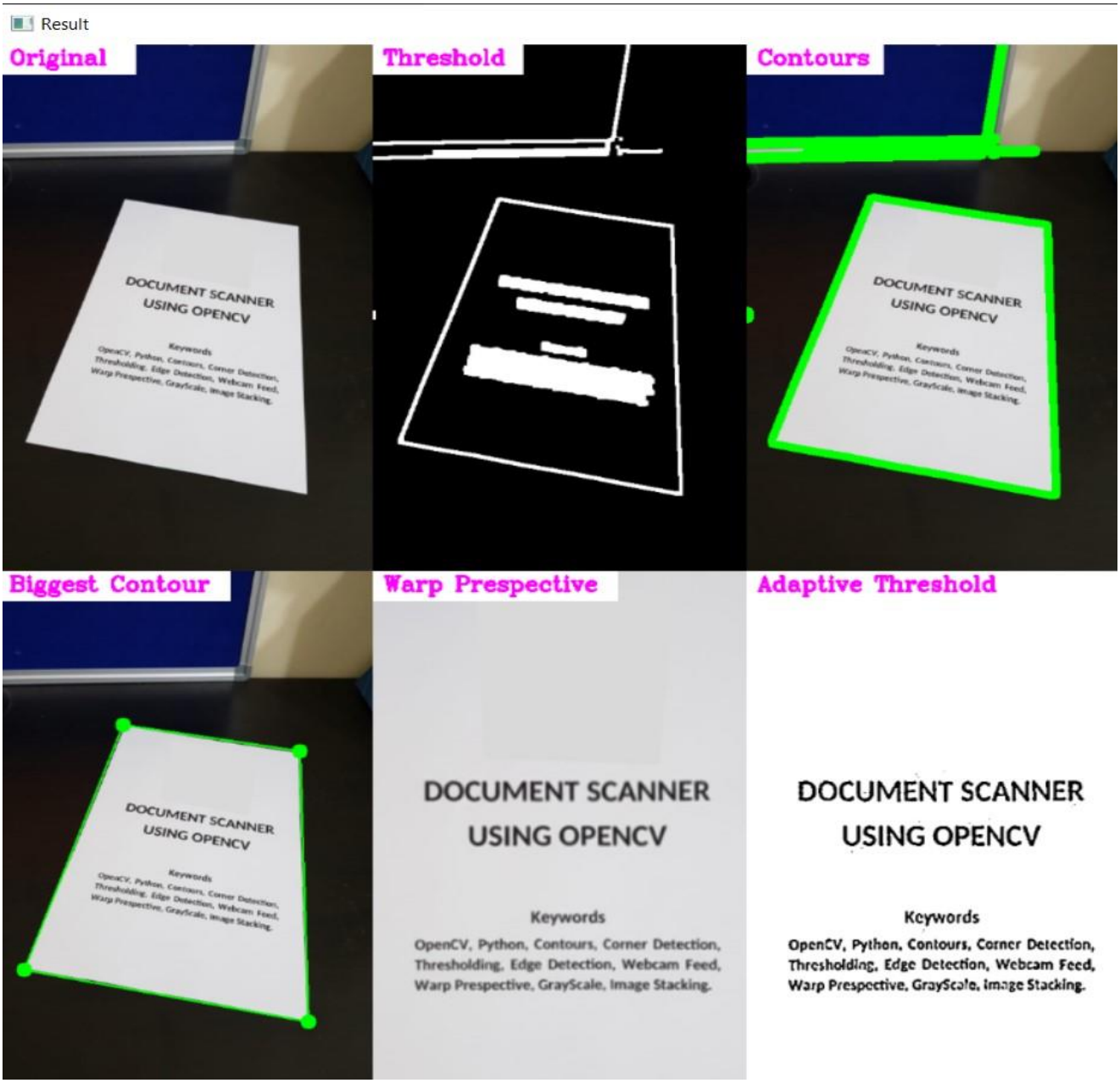

Figure 7: Results of special image processing feature for normal mode at different states.

Camera Functionality Test

The "fswebcam" library is used to access to the camera. Different parameters of functions for "fswebcam" library had tested such as skip, and frame captured. When there is no any skip of frame when capturing image, the camera is not fully activated yet, hence the image taken is black. When comparing the result of 1 skip of frame and 2 skip of frame, the result with 2 skipped frame is better and clearer. While when the capturing frames

different, the most satisfied number of frames is 3. 3 frames captured image is clearer and does not alike 4 and 5 frames captured images that have weird color tone.

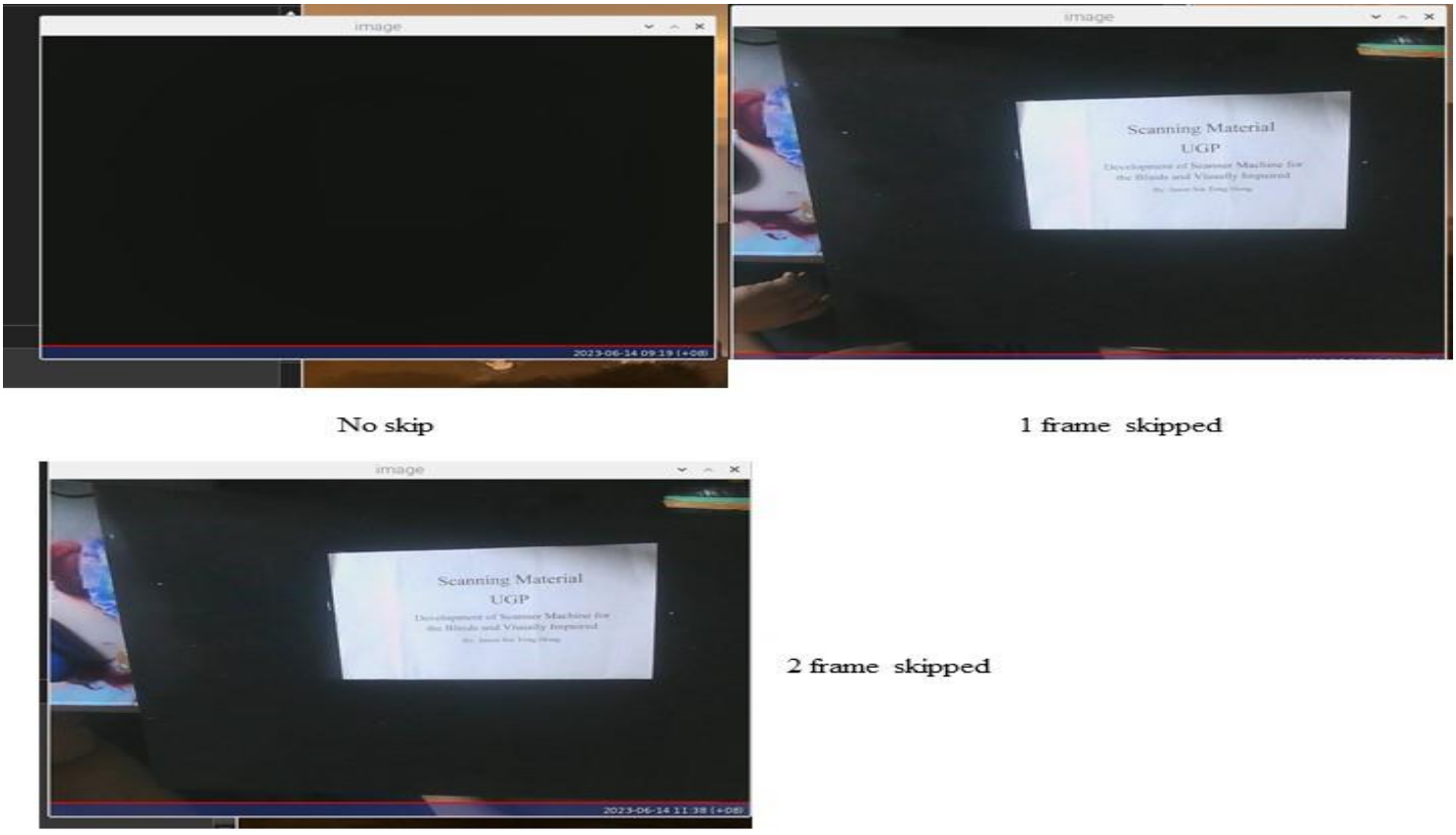

Figure 8: Image captured at different number of skipped frames.

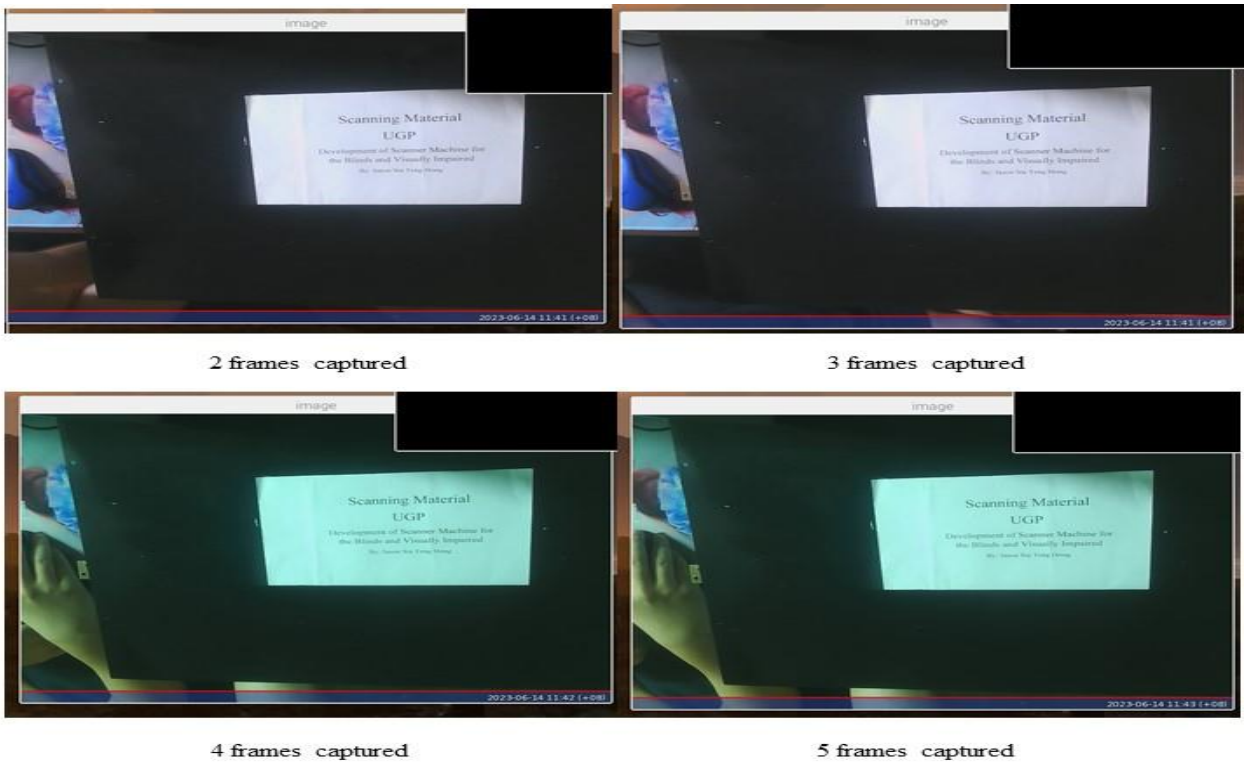

Figure 9: Results of different capturing frames of image.

OCR Error Handling

The errors set in my algorithm are 2. One is the OCR library did not detect any text. The algorithm will inform the error to the user and advice user to put scanning material on the scanning area (in audio form). Then the algorithm will raise exception to stop the for loop. The next error is the text detect is not English. The algorithm will inform the error to the user and advice user to change the material with English text or put scanning material on the scanning area (in audio form). Then the algorithm will also raise exception to stop the for loop.

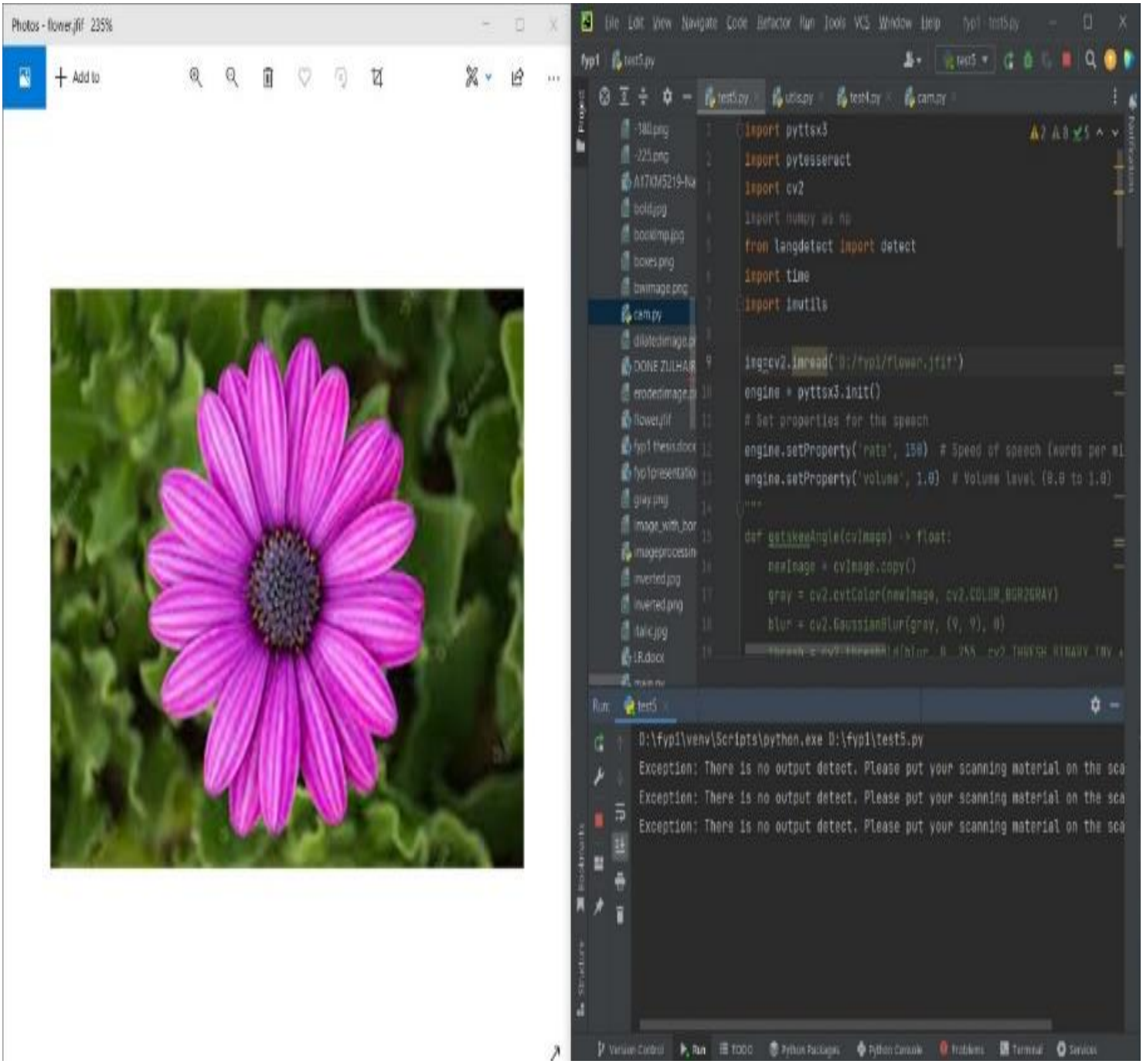

Figure 10: Exception raised when no text in the image.

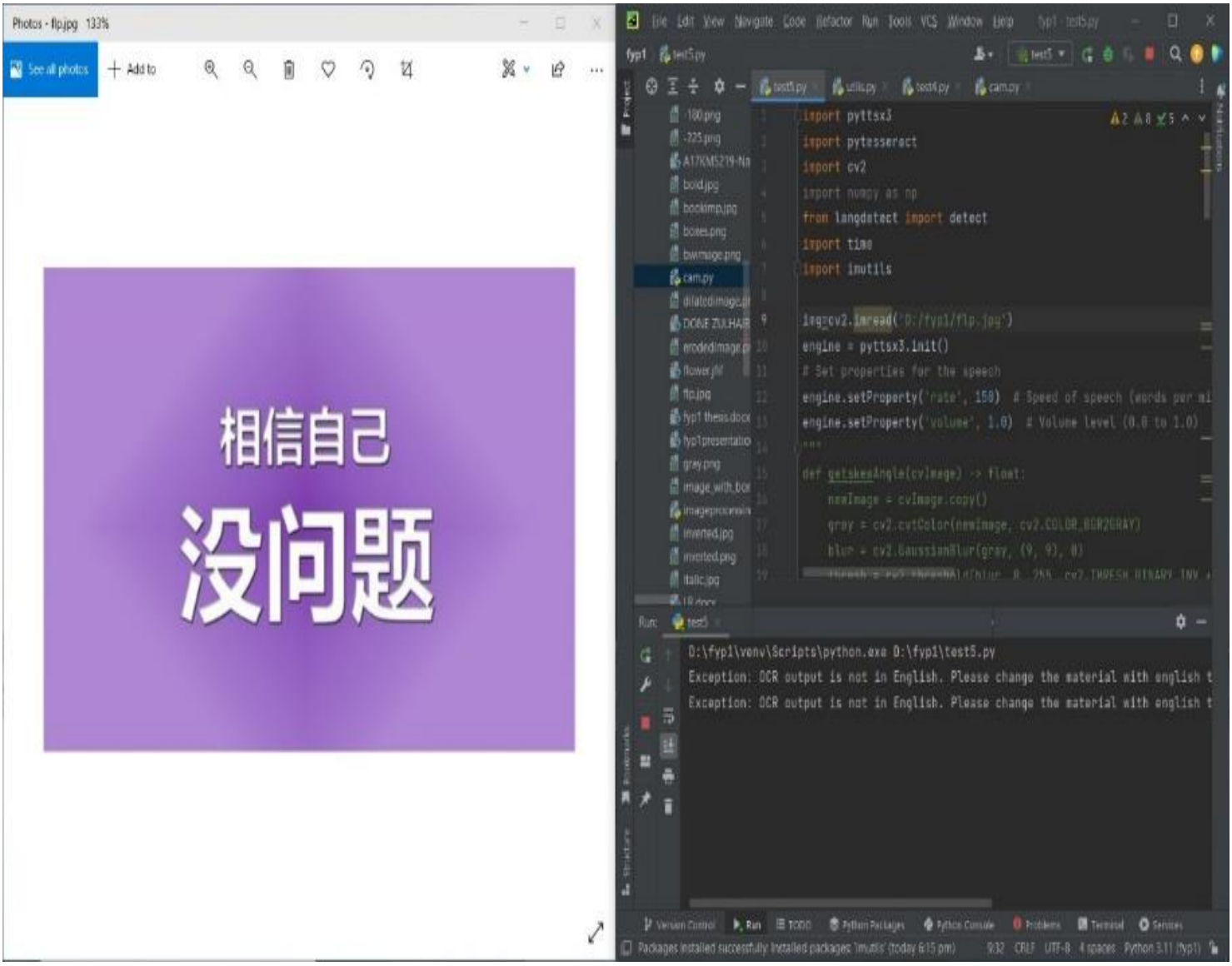

Figure 11: Exception raise when no English text in the image.

TTS Special Feature

At different modes, I had included some features to output the audio of content information and content clearly. For the book mode, the pages and paragraphs will be mentioned. After finish output content of left page, the algorithm will also inform user before changing to next page. While for the normal mode, it will be a simplified version of the previous mode where only the sequence of paragraph be informed.

Accuracy of OCR when Scanner Camera and Smartphone Camera Used on Capturing Image of Different Font Size

Based on the experiment took, the accuracy of OCR of the scanner machine will be satisfied when font size is greater or equal to 14 where the accuracy will exceed 80%. For the accuracy of image take by smartphone, it had better performance when the font size is small as the image take by the smartphone is better in quality (smartphone has better camera with higher resolution and self-stabilizes). The performance of it worse than image taken by scanner camera after font size increase above 14 due to the image processing parameters and features is customize for the scanner camera. Hence, in order to improve the result output, better camera should be used as the scanner camera.

Table 2: Accuracy of OCR when Images with Different Font Size Captured by Scanner Camera and Smartphone Camera

Font Size	Accuracy of OCR (scanner camera)	Accuracy of OCR (smartphone camera)
11	5.7%	21.8%
12	9.7%	65.8%
14	89.4%	89.4%
15	87.3%	70.0%
16	96.7%	89.3%

Performance of Page Turning Mechanism

In book mode, the page turning mechanism will trigger every time when the push button is pushed except the first time when entering book mode (triggered by pushed push on off button). The page turning mechanism is simple. At first, the dc motor will operate and roll up edge of a page from the book. Then, the servo will turn 180 degrees to turn the page to another side.

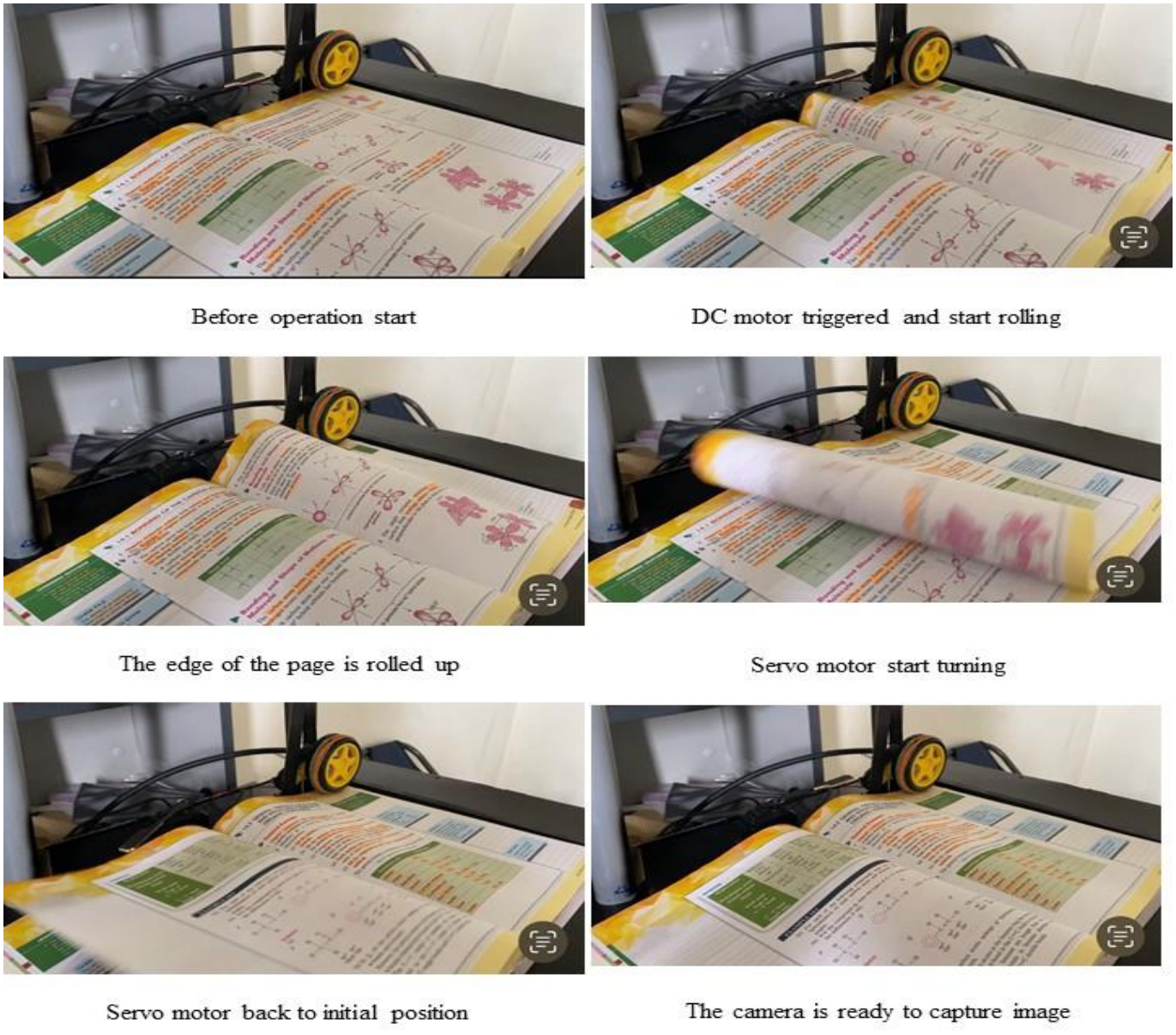

Figure 12: Overall procedure of page turning mechanism

Average Running Time of Different Mode and Crash Checking

The average setup time 5.01 seconds. Since there are two modes and the time taken of audio output is depend on the length of text. Hence, to get the running time of scanner machine at different mode, I set a pause right before the output generated for each mode. While we calculating the average running times, we also perform crash checking to ensure the scanner machine working well without any program crashed. The average running time for book mode is 11.57 seconds which is higher than normal mode as it has extra steps to process the image into 2 images. While normal mode had an average running time of 8.38 seconds. For the crashing test the possibility of crash occur is 20% which is quite high. Improvement or smoothing of algorithm need to be carried out for better performance.

Table 3: Average setup time and running time for different modes.

No.	Setup Time (s)	Time taken for book mode (s)	Time taken for normal mode (s)
1	5.76	11.33	11.31
2	5.07	12.05	7.55
3	5.14	11.02	7.79
4	5.02	11.90	7.76
5	4.07	11.53	7.50
Average	**5.01**	**11.57**	**8.38**

CONCLUSION

Based on the functionality test, a conclusion can be made that the objective of this project is achieved. A scanner machine for blinds and visually impaired (user) has been developed successfully with a completed hardware and software. This scanner machine had lots of features which can perform different image processing methods according mode selected, inform user about which errors raised during the process, give different solution message to user according to the errors and give different output according mode selected. Other than that, this scanner machine able to deal with rotated scanning material in range of -90º to 75º with accuracy greater than 80%. When different font size of scanning materials is used, the scanner machine performs well for font size greater than 13 with accuracy above 85%. Besides, the scanner had an average setup time of 5.01 seconds, average running time ("book" mode) of 11.57 seconds and average running time ("normal" mode) of 8.38 seconds. Hence, we have a conclusion where overall running time of "book" mode is around 16.58 seconds and overall running time of "normal mode" is around 13.39 seconds. Not only that, the algorithm crashing test also provide satisfied result where the possibility of algorithm crashed is less than 20% which is 13.33%. However, this scanner can be still improved by modified scanner design, include auto adjust camera holder, improve to better page turning mechanism, include high resolution camera, include PCB design, improve image processing for noise removal and sharpening image, and improve error handling algorithm.

ACKNOWLEDGMENTS

A special gratitude I give to my final year project supervisor, Ts. Dr. Shaharil Bin Mad Saad, whose contribution in stimulating suggestions and encouragement, helped me to coordinate my project especially in writing this report. I am extremely thankful to my family, friends, classmate for giving their fully support and suggestion in completing my project.

REFERENCES

1. F. L. M. Chew et al., "Estimates of visual impairment and its causes from the National Eye Survey in Malaysia (NESII)," PLoS One, vol. 13, no. 6, Jun. 2018, doi: 10.1371/JOURNAL.PONE.0198799.
2. Suarez, G. B. G. O. D. (2013). Learning image processing with OpenCV.
3. Xie, G., & Lu, W. (2013). Image edge detection based on opencv. International Journal of Electronics and Electrical Engineering, 1(2), 104-106.
4. Yadav, A. V., Verma, S. S., & Singh, D. D. (2021). Virtual Assistant for blind people. International Journal, 6(5).

5. Van Rossum, G., & Drake Jr, F. L. (1995). Python tutorial (Vol. 620). Amsterdam, The Netherlands: Centrum voor Wiskunde en Informatica.
6. Thongbai, N., & Nakpong, N. (2020, June). Reading Aid Machine for Elderly and Visually Impaired Using Single-Board Computer. In 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (pp. 543- 546). IEEE.
7. Kasthuri, N., Kumar, N., & Madhavan, R. (2021, November). Finger Vision for Visually Impaired. In 2021 Innovations in Power and Advanced Computing Technologies (i-PACT) (pp. 1-6)
8. A. (n.d.). Atiz - Book Scanners, Digitization & Scanning Solutions, digital archiving and preservation of cultural documents for libraries, archives and museums. Atiz Innovation. https://www.atiz.com/
9. MaxiAids I i-Reader 2 Desktop Scanner. (n.d.).
10. https :/ /www.maxiaids.com/i-reader-desktop-scanner
11. Pacific Vision - Malaysia. (n.d.).
12. https://my. l lowvision.com/index.php?route=product/product