

Development of a portable community video surveillance system

S. Fakhra A. G, A. Fauzan K, M. Saad H, R. Affendi H, K. H. Fen

Faculty of Electrical & Electronic Engineering Technology, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia

Article Info

Article history:

Received Aug 11, 2018

Revised Nov 20, 2018

Accepted Dec 11, 2018

Keywords:

Image processing

OpenCV

Raspberry pi

Telegram bot

Video surveillance

ABSTRACT

In 2016, a crime rate has been evidently increasing particularly in Kuala Lumpur areas, including reports on house break-ins, car thefts, motorcycle thefts and robbery. One way of deterring such cases is by installing CCTV monitoring system in premises such as houses or shops, but this usually requires expensive equipment and installation fees. In this paper a cheaper alternative of a portable community video surveillance system running on Raspberry Pi 3 utilizing OpenCV is presented. The system will detect motion based on image subtraction algorithm and immediately inform users when intruders are detected by sending a live video feed to a Telegram group chat, as well as sound the buzzer alarm on the Raspberry Pi. Additionally, any Telegram group members can request images and recorded videos from the system at any time by sending a get request in Telegram which will be handled by Telegram Bot. This system uses the Pi NoIR camera module as the image acquisition device equipped with a 36 LED infrared illuminator for night vision capability. In addition to the Python language, OpenCV, a computer vision simulation from Intel is also used for image processing tasks. The performance analysis of the completed system is also presented computational complexity while offering improved flexibility. The performance time is also presented, where the whole process is run with a noticeable 3 seconds delay in getting the final output.

*Copyright © 2019 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

Shamsul Fakhra Abd Gani,
Faculty of Electrical & Electronic Engineering Technology,
Universiti Teknikal Malaysia Melaka (UTeM),
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
Email: shamsulfakhra@utem.edu.my

1. INTRODUCTION

Royal Malaysia Police (PDRM) suggested that installation of surveillance system would deter potential burglaries as well as facilitate police investigation in obtaining evidence and facial identification of suspects in the event that it did happen. However, basic surveillance system is still priced relatively expensive in Malaysia, approximately RM1400 for a basic monitoring system and at least RM1800 with added live mobile viewing. Thus, a low-cost video surveillance system is proposed to increase the affordability and widen the user base for such system. This project is to study and implement a low-cost video surveillance system based on Raspberry Pi 3 by using image-based motion detection algorithm as well as Telegram as the mobile application.

This project uses Raspberry Pi 3 Model B, released in 2016 and costing a mere RM170 for the single board computer. As depicted in Table 1, the processor has had major upgrade from the previous version, from 32-bit ARMv8 to 64-bit Quad Core Broadcom, effectively increasing the processor speed from 900MHz on the Pi 2 to up to 1.2GHz on the Pi 3. In addition to that, it is also now equipped with built-in Wi-Fi chip and Bluetooth Low Energy (BLE) chip, which facilitates our surveillance system to connect to Wi-Fi without needing any additional dongle [1].

Table 1. Raspberry Pi 3 Specification Details

specification	details
SoC	Broadcom BCM2837
CPU	4× ARM Cortex-A53, 1.2GHz
GPU	Broadcom VideoCore IV
RAM	1GB LPDDR2 (900 MHz)
Networking	10/100 Ethernet, 2.4GHz 802.11n wireless
Bluetooth	Bluetooth 4.1 Classic, Bluetooth Low Energy
Storage	External microSD
GPIO	40-pin header, populated
Ports	HDMI, 3.5mm analogue audio-video jack, 4× USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)

An 8 megapixels Raspberry Pi camera is attached to the on-board Raspberry Pi camera connector, and this creates an image capture system with embedded computing that can extract information from images without the need for an external processing unit. Considering the requirements of image processing compared to the Raspberry Pi's processing module and its peripherals, it is decided that the system is capable on executing the tasks specified.

For the remote interface, Telegram Bot will be used as it enables machine and user to interface seamlessly. The Telegram app can be installed on intended user's mobile phone, then it is ready to go. Telegram Bot has artificial intelligence features like replying a message with images via Image Bot. The bot is also designed for programming purposes and can communicate with other Bots. According to Telegram, the messages and requests sent to Bots are connected with an intermediary server and are linked via HTTPS interface which comes with Telegram API [2].

OpenCV, an open source computer vision library developed by visual interaction group from Intel is used as digital image processing library. It allows Raspberry Pi to handle image processing algorithms directly from inside the Pi environment [3].

Earlier work by Patoliya, Mehta and Patel propose a system consisting of night vision wireless camera to transmit videos is proposed [4]. It uses Arduino as the controller which is connected to servo motor for camera movement and Bluetooth module for transmitting images to Android phones. However, the Bluetooth signal is found to be not stable when compared to Wi-Fi. Dong, Yang, Wang and Xu proposed another surveillance system for elderly with fall and inactivity detection. The project uses Raspberry Pi as the controller utilizing Gaussian Mixture Model for background subtraction method which will send email or SMS to inform family members should an incident occurs [5]. Rashid, Abir, Shourove and Muntaha proposed a home security system which identify visitors by image processing. The project uses Raspberry Pi as the controller with OpenCV library for background subtraction method and also Pocketsphinx, a speech recognition software that runs natively on Raspberry Pi for voice recognition [6].

2. PROPOSED DESIGN

2.1. Block diagram

The block diagram in Figure 1 shows the overall system components including the Raspberry Pi 3 as the central platform for the system. An 8 megapixels night vision pi camera with infrared illuminators act as the image acquisition device to the system. When the system detects possible intrusion, the buzzer will emit alert sound and additionally an alert message will be sent from the Raspberry Pi 3 to the group members in Telegram app. Additionally, any Telegram group members can request images and recorded videos from the system at any time by sending a get request in Telegram which will be handled by Telegram Bot.

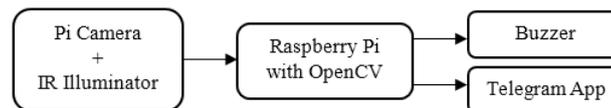


Figure 1. System block diagram

2.2. Motion detection algorithm

Ansari, Sedky, Sharma and Tyagi suggest that motion detection algorithm can be set to detect physical movement of objects in a given range by determining object movement on successive and progressive images. The algorithm compares only the grey frames after converting colour images from RGB

pattern to greyscale [7], [8]. Jyothi and Vardhan suggest that motion detection algorithm applies principles of differencing frame where the algorithm compares pixels on how they change location after each frame [9]. A study conducted by Islam, Nahiyah and Kiber also agreed on the concept that considering frames of video as images, motion can be detected by performing absolute image difference [10], [11]. There is a more reliable method based on background subtraction and frame difference, a fusion to more precisely detect motion. Liu, Fan and Wang designed a technique based on background subtraction and three-frame difference to detect motions underwater by producing a different image to determine pixels from non-moving objects [12]. The function of three frame difference is less sensitive to background noise, and the authors proved that the two combined methods work well in their experiment. This project will also apply background subtraction with frame difference for the motion detection.

The authors agreed that separation process between foreground and background is a key factor in computer vision applications, for example in surveillance use, motion capture, action recognizing and video compressing. However, the acceptable separation process faces many challenges. Factors that resulted in failure in motion detection includes sudden changes in illumination like cloud movement, motion change like waving branches of trees, camera shake, and also background geometry change like shadows and parked cars. Therefore, image pre-processing is crucial to reduce error in motion detection. Small changes in pixel will easily lead to failure in motion detection. For example, noise always happen in the background and it changes the value of pixels. Further explained by the authors, noise results in difference in pixels with the correct pixel value in an image. Consider Equation (1) for an image f with scene s and noise n :

$$f(i, j) = s(i, j) + n(i, j) \quad (1)$$

The equation explained that any acquired image can be affected by scene and noise. If there is no change in intensity, no change is observed. But if one pixel is brighter, and has higher pixel values than its neighbours, this will result in noise. The authors stated that image pre-processing should be done before actual image processing to suppress distortions.

3. IMPLEMENTATION

3.1. Hardware design

The hardware will be run on a Raspberry Pi 3 as the central computer with Raspbian Jessie OS installed and additional peripherals of an 8 megapixels camera to capture image, and a buzzer as an audible alarm. The system is intended to be portable so it is powered by connecting it to a mobile powerbank. Figure 2 shows the prototype of the video surveillance system in development mode, while Figure 3 shows the prototype in normal running mode, with the LCD monitor, keyboard and mouse removed. Figure 4 shows the prototype running in pitch black darkness with only the IR LEDs visible.



Figure 2. System in development mode



Figure 3. System in normal mode



Figure 4. System in pitch black

3.2. Software design

In addition to the Python programming language that is natively used in Raspberry Pi environment, this system will also use OpenCV, a computer vision simulation from Intel to help with image pre-processing that is designed with criterias of resource optimization, low power consumption and improved speed.

4. EXPERIMENTAL RESULTS AND DISCUSSION

4.1. Night vision output

Figure 5(a) shows the image capture from the system during day time. The colour of the image can be seen to be slightly pink in color because of the discarded IR filter. Figure 5(b) shows the image capture during night time without night vision turned on where the image is in total pitch black, while Figure 5(c) shows the image capture when the IR illuminator is turned on where we can see some objects in the image albeit the dark lighting.

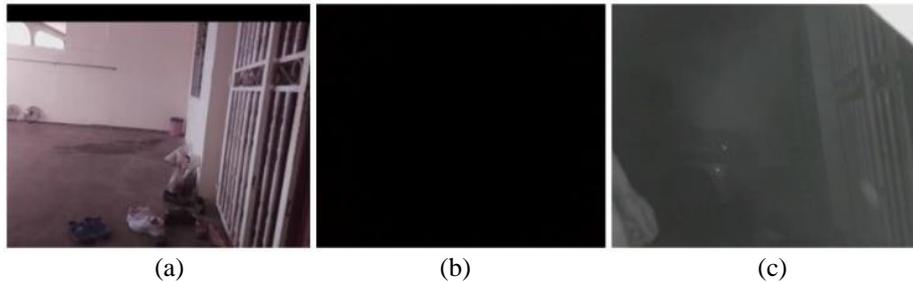


Figure 5. Sample captured image

Figure 6(a) shows the image capture of from the system, and since the during the next few frame captures there are no changes in the video, the image difference in Figure 6(b) and Figure 6(c) results in a black image, which is essentially no motion detected.

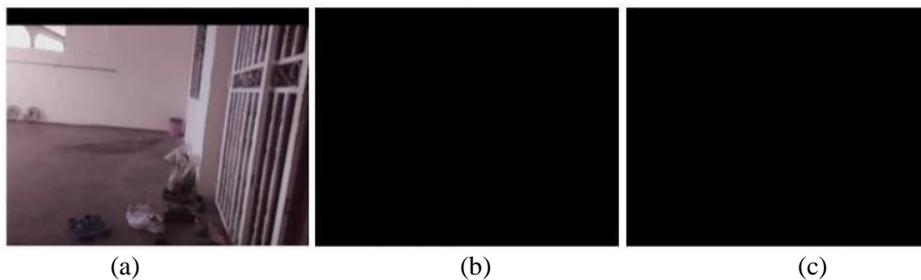


Figure 6. Sample captured image

Figure 7(a) shows what happens when an object moves into the picture during daytime which is originally as illustrated in Figure 6(a). Figure 7(a) is the 'Frame Delta' which is the pre-threshold image, which in Figure 7(b) is shown having grey color frame differences. Figure 7(c) is the 'Thresh' which is the thresholded image, showing white blob against black background, which is classified as motion detected.

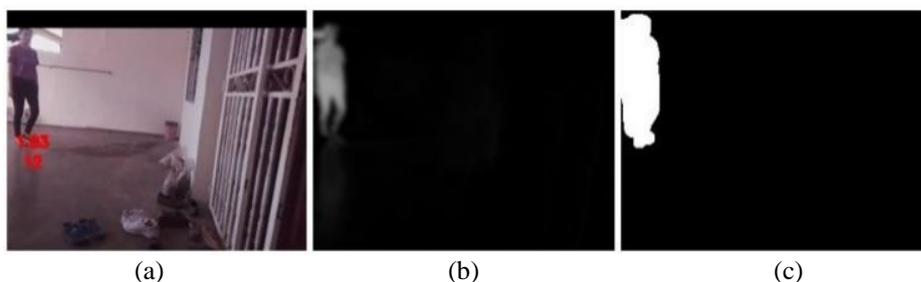


Figure 7. Sample captured image

Figure 8 recreates the experiment done in Figure 7, but the image capture is done during night time and in total darkness. An image is captured with frame difference than the previous capture as shown in Figure 8(a), pre-threshold as in Figure 8(b), and the result is a detected motion of white blob in Figure 8(c).

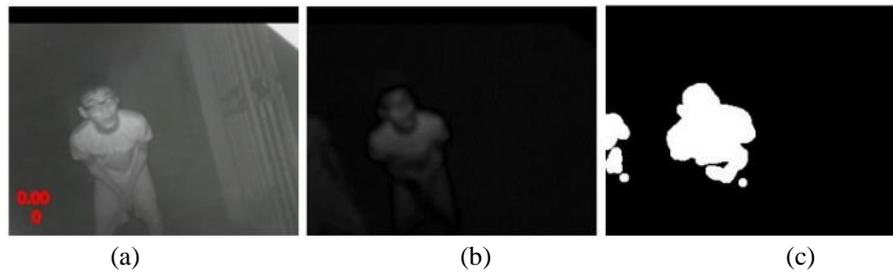


Figure 8. Sample captured image

4.2. Interaction between hardware and software

Once motion detects possible intruder, an image capture is automatically sent to a predefined Telegram group chat together with a 3 seconds video of the incident. Figure 9(a) shows the group chat containing family members as well as the Raspberry Pi as a user. Figure 9(b) shows the alert message notification has arrived into the Telegram group, and Figure 9(c) shows the report from Raspberry Pi with captured image as well as a 3 seconds video that triggers the alarm which can be viewed and downloaded by anyone in that particular Telegram group.



Figure 9. Telegram notification from Raspberry Pi

4.3. Reaction time from Pi to Telegram versus distance

Based on the experiment, it is found that reaction time to transmit picture and video from Raspberry Pi to Telegram is not affected by distance as detailed in Table 2 (day time) and Table 3 (night time). The reaction time is only affected by the speed of the available wifi connection. Within the set distance of 8 meters, the motion detection system works very well during day time. Results in Table 3 shows that night surveillance is not available for distances over 4 meters due to the infrared illuminator lighting which is only effective within 5 meters ideally. Thus, objects moving between 4 meters to 8 meters during night time can be seen from recorded video, but the motion will not be detected.

Table 2. Reaction Time Versus Distance (Day Time)

Distance (m)	Reaction Time (s)
1	2
2	1
3	2
4	1
5	2
6	2
7	2
8	1

Table 3. Reaction Time Versus Distance (Night Time)

Distance (m)	Reaction Time (s)
1	2
2	2
3	1
4	2
5	-
6	-
7	-
8	-

Figure 10 illustrates the reaction time in seconds versus distance in meters for day time and night time analysis of the system respectively. It is observed that given a stable wifi signal is available for the Raspberry Pi to piggyback on, any motion detection alert notification will be delivered to Telegram app in user's mobile phone within 1 to 2 seconds.

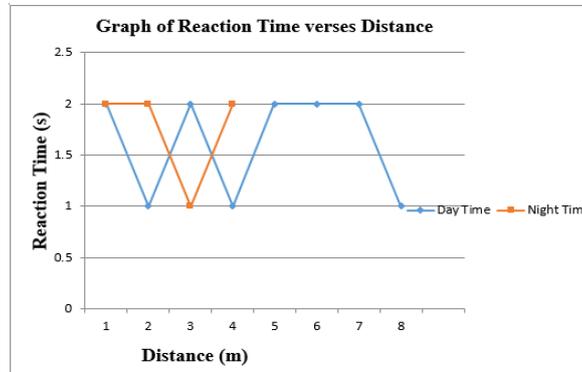


Figure 10. Reaction time versus distance analysis

4.4. Accuracy test, sensitivity test and specificity test

Table 4 and Table 5 shows the accuracy test results for day time and night time surveillance tested directly for 10 consecutive days. According to Table 4, day time surveillance scored 95.42%, which is more accurate than at night time with score of 89.82%. The project shows high accuracy, and the system is able to differentiate intrusion and normal cases correctly.

Table 4. Test for Day Time Surveillance

Trials (Day)	True Positive	True Negative	False Positive	False Negative	Accuracy Test	Sensitivity Test	Specificity Test
1 st	5	4	1	0	90%	100%	80%
2 nd	5	5	1	1	83.33%	83.33%	83.33%
3 rd	5	4	1	0	90%	100%	80%
4 th	5	4	0	0	100%	100%	100%
5 th	5	5	1	0	90.91%	100%	83.33%
6 th	5	5	0	0	100%	100%	100%
7 th	5	5	0	0	100%	100%	100%
8 th	5	5	0	0	100%	100%	100%
9 th	5	5	0	0	100%	100%	100%
10 th	5	5	0	0	100%	100%	100%
1 st	4	4	2	1	72.73%	100%	66.66%
2 nd	4	4	1	1	80%	80%	80%
3 rd	4	5	2	0	81.81%	100%	71.43%
4 th	4	5	1	1	81.81%	80%	83.33%
5 th	5	5	1	1	90.91%	83.33%	83.33%
6 th	5	5	0	0	100%	100%	100%
7 th	5	5	0	0	100%	100%	100%
8 th	5	5	0	0	100%	100%	100%
9 th	5	5	1	0	90.91%	100%	83.33%
10 th	5	5	0	0	100%	100%	100%

Based on sensitivity test, day time surveillance scored 98.33%, which is more sensitive to motion than in night time with 92.33%. For specificity test, day time surveillance scored 92.67%, which is more specific than in night time with 86.81%. The system scores high percentage and is able to determine normal cases correctly.

Figure 11 summarizes the test results of measuring accuracy, sensitivity, and specificity during day time, while Figure 12 summarizes the same test but done during night time. It is found that night time surveillance performs poorer than during day time mainly because of the insufficient night vision lighting provided by the infrared illuminator causing the frame difference to be too small and neglected in greyscale images.

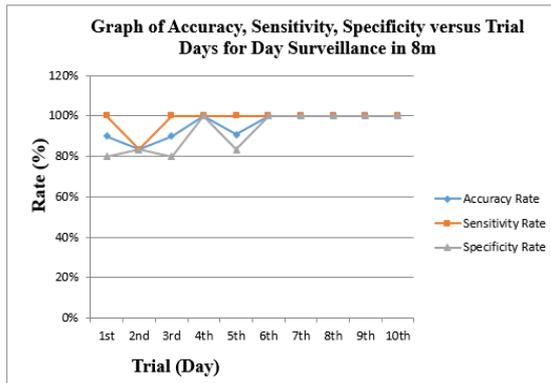


Figure 11. Accuracy, sensitivity, specificity versus days (day time)

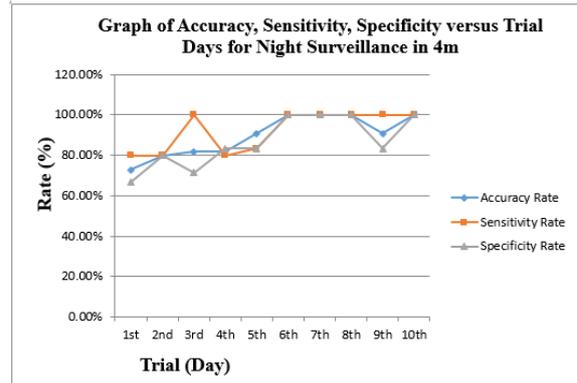


Figure 12. Accuracy, sensitivity, specificity versus days (night time)

5. CONCLUSION

The development of a portable community video surveillance system using Raspberry Pi 3 is demonstrated in this article, implementing OpenCV as the core image processing software. At the correct setup the system performs wonderfully, especially since the whole system costs within a budget of RM500 compared to surveillance systems sold in the market which sells for around RM1800. The notification system also employs an open source chat app, Telegram, which is free and can be easily installed by the user.

ACKNOWLEDGEMENTS

The authors would like to thank Universiti Teknikal Malaysia Melaka (UTeM) for providing the support needed to complete the work herein via grant number: RAGS/1/2015/ICT01/FTK/03/B00121 and PJP/2016/PKA/FTK-CACT/S01512.

REFERENCES

- [1] Hossain N., Kabir M. T., Rahman T. R., Hossen M. S. and Salauddin F., "A Real-Time Surveillance Mini-Rover Based on OpenCV-Python-JAVA Using Raspberry Pi 2," *Proceedings - 5th IEEE International Conference on Control System, Computing and Engineering*, pp. 476–481, Nov 2016.
- [2] Job J., Naresh V. and Chandrasekaran, K., "A Modified Secure Version of the Telegram Protocol (MTProto)," *IEEE International Conference on Electronics, Computing and Communication Technologies, CONECCCT 2015*, pp. 1–6, 2016.
- [3] W. F. Abaya, J. Basa M. Sy, A. C. Abad and E. P. Dadios, "Low Cost Smart Security Camera with Night Vision Capability Using Raspberry Pi and OpenCV," *International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, Palawan, pp. 1-6, 2014.
- [4] Patoliya J, Mehta H. and Patel H., "Arduino Controlled War Field Spy Robot Using Night Vision Wireless Camera and Android Application," *NUiCONE 2015 - 5th Nirma University International Conference on Engineering*, pp. 3–7, 2016.
- [5] Dong Q, Yang Y, Wang H and Xu J. H., "Fall Alarm and Inactivity Detection System Design and Implementation on Raspberry Pi," *International Conference on Advanced Communication Technology, ICACT*, pp. 382–386, 2015.
- [6] Rashid M. T., Abir I. K., Shourove N. S., Muntaha R. and Rhaman M. K., "Intelligent Intrusion Prevention System for Households Based on System-On-Chip Computer," *Canadian Conference on Electrical and Computer Engineering*, Oct 2016.
- [7] A. N. Ansari, M. Sedky, N. Sharma and A. Tyagi, "An Internet of things approach for motion detection using Raspberry Pi," *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*, Harbin, pp. 131-134, 2015.
- [8] Hamzah R. A, Abd Gani S. F, Kadmin A. F, Hamid M. S, Salam S. and T. Wook T. M. F, "Disparity Map Estimation Uses Block Matching Algorithm and Bilateral Filter," *International Conference on Information Technology Systems and Innovation (ICITSI)*, pp. 151–154, 2017.
- [9] S. N. Jyothi and K. V. Vardhan, "Design and Implementation of Real Time Security Surveillance System Using IoT," *International Conference on Communication and Electronics Systems (ICCES)*, Coimbatore, pp. 1-5, 2016.
- [10] M. Z. Islam, K. M. T. Nahiyah and M. A. Kiber, "A Motion Detection Algorithm for Video-Polysomnography to Diagnose Sleep Disorder," *18th International Conference on Computer and Information Technology (ICCIT)*, Dhaka, pp. 272-275, 2015.

- [11] Hamzah R. A, Hamid M. S, Kadmin A. F. and Abd Gani S. F, "Improvement of Stereo Corresponding Algorithm Based on Sum of Absolute Differences and Edge Preserving Filter," *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 222–225, 2017.
- [12] Wang Z, Liu S. and Fan Q., "Cloud-based Platform for Embedded Wireless Video Surveillance System," *Proceedings - International Conference on Computational and Information Sciences*, pp. 1335–1338, 2013.

BIOGRAPHIES OF AUTHORS



Shamsul Fakhhar bin Abd Gani graduated from Universiti Malaysia Perlis (UniMAP) in Bachelor of Engineering (Computer Engineering) with honours in 2006 and later received his Master's degree in Internet & Web Computing in 2015 from Royal Melbourne Institute of Technology (RMIT) Australia. He started his career as an R&D electronic engineer specializing in software design for meter cluster development in Siemens VDO Automotive Penang (later known as Continental Automotive Malaysia). Before leaving Continental, Shamsul plays an active role in Mitsubishi Fuso projects as a software developer and also software project manager. Shamsul is now a lecturer in electronic and computer engineering technology department of FTKEE UTm.



Ahmad Fauzan bin Kadmin CEng. P.Tech currently attached with UTm as a researcher, Ahmad Fauzan bin Kadmin CEng. P.Tech. has over 14 years of experience in electronic & computer engineering field with technical expertise in R&D engineering, computer vision & medical electronics. He graduated with a Bachelor Degree in Electronic Engineering from Universiti Sains Malaysia (USM) and Master Degree in Computer & Communication Engineering from Universiti Kebangsaan Malaysia (UKM). Previously, he worked with Megasteel Sdn. Bhd., Samsung SDI (M) Sdn. Bhd. and Agensi Angkasa Negara. He published several technical and engineering paperworks in image processing and medical electronics.



Mohd Saad born in 1981, graduated from Multimedia University (MMU) where he received his B. Eng majoring in Computer in 2003. He then continues working with Telekom Malaysia Berhad and later joined Siemens VDO Automotive Penang (later known as Continental Automotive Malaysia) as a software engineer. He developed firmware in meter cluster development for both local and international customer in the area of Controller Area Network and ECU diagnostic during his tenure in the company. In 2014 he received his M. Eng majoring in Computer and Communication from Universiti Kebangsaan Malaysia (UKM). Currently he is a lecturer in Universiti Teknikal Malaysia Melaka teaching programming, embedded system and microcontroller subjects.



Rostam Affendi Hamzah graduated from Universiti Teknologi Malaysia (UTM) where he received his B. Eng majoring in Electronic Engineering in 2000. In 2010 he received his M. Sc. majoring in Electronic System Design Engineering from Universiti Sains Malaysia (USM). In 2017, he received PhD majoring in Electronic Imaging from Universiti Sains Malaysia (USM). Currently he is a lecturer in Universiti Teknikal Malaysia Melaka teaching digital electronics and digital image processing.



Kong Hui Fen is an undergraduate student from Universiti Teknologi Malaysia Melaka in Bachelor of Engineering Technology (Computer Engineering). She is currently finishing her last semester internship programme in Intel Programmable Solution Group Penang.