Sensors and Materials, Vol. 34, No. 7 (2022) 2581–2589
MYU Tokyo

S & M 2992

Development of Intelligent Drone Remote Control System Based on Internet of Things

Shyh-Wei Chen,[1] Yu-Chun Lai,[2] Ching-Tsorng Tsai,[3] Chia-Hui Liu,[4*] and Jih-Fu Tu[5]

[1]Department of Computer Science, Tunghai University,
1727, Sec. 4, Taiwan Boulevard, Xitun District, Taichung City 407224, Taiwan
[2]Master's Program of Digital Innovation, Tunghai University,
1727, Sec. 4, Taiwan Boulevard, Xitun District, Taichung City 407224, Taiwan
[3]Department of Computer Science, Tunghai University,
1727, Sec. 4, Taiwan Boulevard, Xitun District, Taichung City 407224, Taiwan
[4]Department of Applied Mathematics, Chinese Culture University,
55, Hwa-Kang Road, Yang-Ming-Shan, Taipei 11114, Taiwan
[5]Department of Electrical Engineering, Lunghwa University of Science and Technology,
300, Sec. 1, Wanshou Road, Guishan District, Taoyuan City 333326, Taiwan

(Received April 15, 2022; accepted June 6, 2022)

Keywords: IoT environment, drone, sensor, gesture recognition, remote control

Drones have been under development for many years. As a result of market development and the large amount of research investment, the development of drones has been very rapid in recent years. The application of drones in the IoT is becoming increasingly extensive. Drones are convenient for performing a wide range of monitoring tasks. They can be mounted with sensors and used in various fields, and tasks that were difficult to achieve in the past can now be completed. However, to successfully complete a task, the accurate control of drones is necessary. In this paper, we propose a new drone control method in which image recognition technology is used to combine real-time images and distance-sensing feedback based on an IoT environment. The drone can also be intuitively controlled by moving the head and making gestures with the hands. The drone can identify obstacles along a route by viewing the surrounding environment via its supersonic sensor and its flight can be precisely controlled. The proposed drone control system based on the IoT environment has the advantages of providing users with intuitive control, a better experience, and low-cost, high-stability image recognition for drone control.

1. Introduction

In technological development, the design of human–robot interaction (HRI)(1) is leaning increasingly toward interaction, and many machine control applications are gradually moving toward somatosensory interaction. HRI is an integral element of human reliance on the remote control of machines. Control applications and the development of drones have also received considerable attention. Drones are often used in spaces inconvenient for humans to access, such as in news aerial photography missions and regional reconnaissance of disaster sites.(2) The traditional method of controlling a moving drone is typically by using a remote controller or a

*Corresponding author: e-mail: ljh34@ulive.pccu.edu.tw
https://doi.org/10.18494/SAM3940

handheld device. The drone can be controlled using the cross-shaped direction buttons of a physical or virtual remote controller to move in four directions: front, rear, left, and right. In addition, the yaw axis of the drone can be manipulated using the rotary knob to achieve actions such as ascending, descending, left yaw, and right yaw motions. The modes of controlling machines by HRI can be mainly divided into two symmetrical modes.[3] (1) Exocentric viewing: This is the main mode in the traditional control of a drone, wherein the user controls the drone from the perspective of the machine itself. This means that the control personnel can directly confirm the position of the machine in the visible range of the naked eye while simultaneously controlling the direction of movement.[4] If there is an obstacle between the user and the machine blocking the user's line of sight, the user cannot control the machine.[5] (2) Egocentric viewing: The control personnel can control the machine via a camera mounted on the machine while viewing the surrounding environment. In this mode, the problems mentioned above can be avoided. The designs for control in this mode should take into account the narrow field of view and situations that may cause orientation and altitude misjudgments.[6,7] If the control personnel have the ability to use real-time images of a remote drone to view the surrounding scene, more precise control as well as good on-the-spot experience will be possible, and more tasks that could not be achieved otherwise can be completed. However, a skilled pilot is required to control the drone to perform these tasks. To improve the control precision of the drone and optimize the user's control experience, in this study, we developed an intelligent and intuitive control system for drones in the egocentric viewing mode. This system can provide a simple gesture controller for a user that can make the piloting much easier and more convenient. Therefore, we can mount numerous sensors on a drone to execute different remote sensing tasks including aerial videography, photography, and surveillance. The system uses Google's MediaPipe application programming interface (API)[8–10] to capture the user's head pose and gesture and perform feature point annotation and signal conversion. These movements are used as controller signals transmitted to the drone. The orientation of the user's head determines the direction in which the drone moves, and finger movements indicate different movement commands. When a specific gesture is made, it is transformed into a signal for controlling the magnitude of acceleration, which in turn determines the speed of the drone. Such a control method combined with the operation mode of egocentric viewing enables the control personnel to perceive the environment around the drone and detect whether there are obstacles on its path to increase the accuracy of flight control. The novelties of the proposed intelligent drone remote control system are that it provides users with intuitive control to optimize the user experience, i.e., how a user interacts with and experiences the system. The proposed system also enables low-cost and highly stable image recognition for controlling drones.

2. Literature Review

2.1 Flight principle of drone

The main difference between a drone and a general aircraft is that a drone has no wing design and its lift is provided by motor thrust. Aircraft are generally affected by lift, thrust, drag, and gravity.(11) Because of the restrictions of the shape design of the drone, the thrust and weight are the main factors affecting the drone.(12) Since the weight is a fixed force, the magnitude of the force can be changed by adjusting the load. Thrust, on the other hand, is proportional to the motor speed. When the motor speed is consistent and sufficient, the drone can rise and take off smoothly. The center of gravity (COG) is the point of balance of an aircraft; when lifting power is applied to the aircraft at the COG, it is in perfect balance. If the COG deviates from the center of the aircraft, it will be unstable and cannot be controlled smoothly. In contrast, a drone mainly moves forward, backward, left, and right, and ascends, descends, and rotates clockwise and counterclockwise through the speed control of the four motors. In a balanced state, take-off and landing can be achieved provided the thrust of the four propeller motors can be reduced or increased at the same time.(13) The control states are classified into the following four types,(14,15) as shown in Fig. 1:

1. Height control: the four propeller motors are made to accelerate or decelerate at the same time, thereby controlling the take-off or landing of the drone.
2. Roll control: the speeds of No. 1 and No. 4 motors are made to be different from those of No. 2 and No. 3 motors, thereby controlling whether the drone rotates clockwise or counterclockwise or flies left or right.
3. Pitch control: No. 1 and No. 2 motors and No. 3 and No. 4 motors are made to rotate at different speeds, causing the drone to move forward or backward.
4. Yaw control: No. 1 and No. 3 motors are made to be faster or slower than No. 2 and No. 4 motors to control the yaw angle of the drone so that the drone rotates clockwise or counterclockwise, respectively.

2.2 Experimental drone model

The drone control system proposed in this paper is applied to an RYZE Tello drone for the verification of somatosensory control,(16) mainly because of its low cost and high penetration rate, which should render this system widely applicable in various fields. RYZE Tello provides an open software development kit (SDK) with a complete API that allows developers to attain a high degree of openness and comprehensive control over drones. A Wi-Fi connection is used between the drone and the control terminal. To implement the connection, the power button of

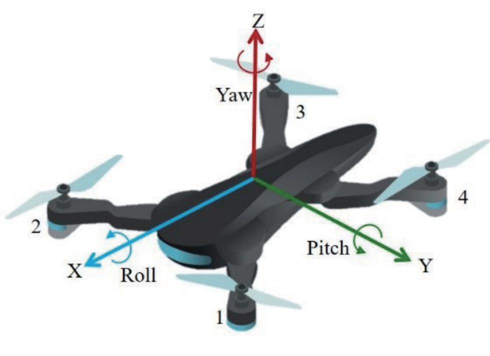


Fig. 1. (Color online) Control states of drone.

the Tello drone is pressed, which immediately generates a set of service set identifiers (SSIDs). At this time, the control terminal must also be equipped with Wi-Fi connection capability. The connection is completed after pairing with the SSID generated by the Tello drone. The RYZE Tello drone weighs about 80 g and measures 98 × 92.5 × 41 mm. It is equipped with an IR sensor, an aneroid barometer, and a Wi-Fi connection.

3. Proposed Drone Control System

3.1 System architecture

The egocentric viewing mode is adopted to design the control system. In this study, the drone is further controlled by movements of the user's head and hands. The system architecture also includes computer vision to capture the movements of the user.(16) Our remote control system uses only a simple webcam to capture the user's head movement and hand gestures without setting up an additional motion detector. Therefore, the proposed drone remote control system can be flexible and make the task of piloting easy. The proposed system can be divided into three modules, as shown in Fig. 2.

1. Input module: The proposed system has a web camera that is set up in front of the user to capture the user's head and hand movements for use as controller signals for impact recognition. In this system, MediaPipe is used to define the marker points of the head pose, and 468 point coordinates in 3D space are returned. Among them, the axis representing the depth has a negligible effect on feature extraction of the system after evaluation. To reduce the complexity of the system, this axis is ignored, and the 3D spatial coordinates are reduced to a 2D plane. The trilinear coordinates are then used to calculate the vector and angle

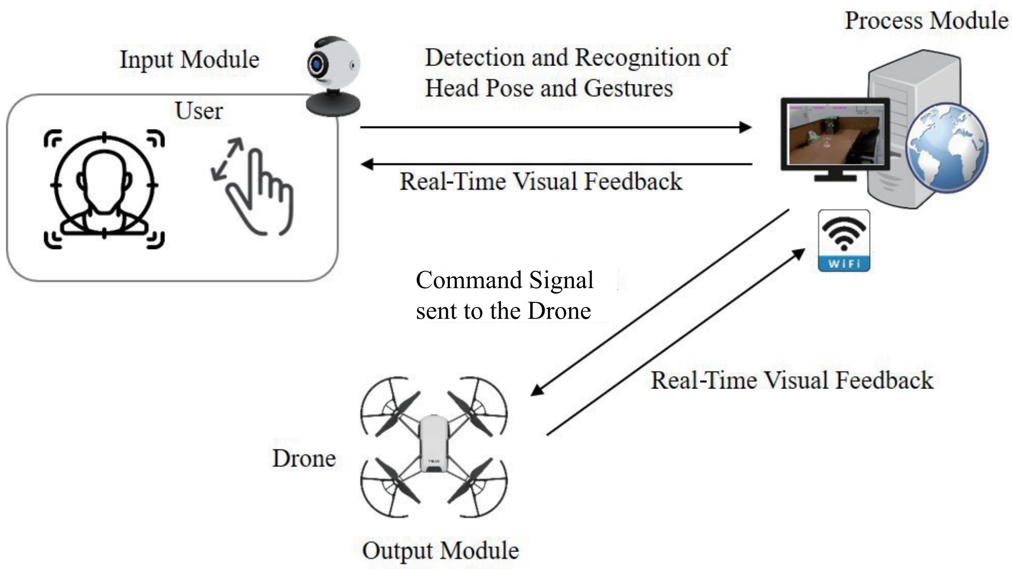


Fig. 2. (Color online) System architecture.

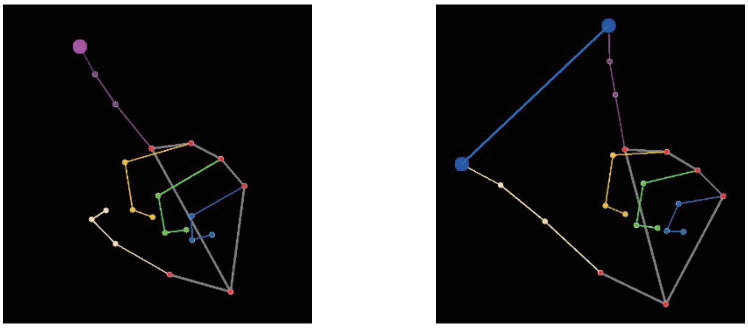
between the marked points, which are extracted as features and used as the basis for judging which gesture corresponds to the direction of the drone. When the direction of the drone corresponds to the user-defined gesture, its spatial coordinates or line segment distance is used as the data returned by the module as necessary.

2. Process module: After this module receives the processed data from the hand and head modules, it integrates the defined gestures with the API in the Tello SDK. These actions during the process module can be categorized as take-off, landing, forward, backward, ascent, descent, left shift, right shift, left turn, right turn, photographing, flight assistance, and other actions. In accordance with the parameter requirements, the gesture is quantified to generate the command signal to control the drone.
3. Output module: This module transmits data to the drone through Wi-Fi and also synchronizes data sent to the drone. After receiving the command signal, the drone performs the corresponding actions and transmits the image in front of it to the user via the installed camera. At the same time, the drone integrates the feedback images with the gestures of the control commands, so that the user can judge whether their control has been correctly conveyed to the drone.

3.2 Head and hand tracking

The hand posture is identified by the input module, and the 21 3D space marker points of the hand are inferred through a single frame. By completing specific gestures and conditions, the application in the human–machine interface further triggers buttons that control the drone's take-off and landing, as well as its forward and backward speeds. To reduce false touches, the control interface should have the ability to recognize whether the hand model represents a defined gesture. In this study, the degree of bending of a single finger was determined by calculating the angle between marked points and used as a form of identification. When the degree of bending is greater than a certain angle, it is determined whether straightening or bending is further performed. When the thumb and the middle, ring, and little fingers are bent at the same time while keeping the index finger straight, the condition of the user's specific gesture is satisfied. At this time, the tip of the index finger is marked and can be used as an indicator to trigger the take-off and landing button. When the middle, ring, and little fingers are bent at the same time, the throttle mode is switched on to control the forward or backward motion. Also, the relative distance between the thumb and the index finger is captured at this time. The relative distance is defined as the ratio to the length of the palm to prevent this distance from being affected by the distance (z) between the hand and the camera. The relative distance in this system ranges from 0 to 21. Therefore, it is calculated after computing linear conversion with the forward and backward speed parameters (0 to 100) of the drone. When the relative length ranges from 0 to 6, it is converted to a backward speed of 0 to 100. The above description is shown in Fig. 3, where Gesture I controls the direction of the drone and Gesture II controls its speed.

Likewise, 468 3D feature models of head poses are inferred from a single frame by MediaPipe. In the proposed control system, the corresponding control of the drone is identified and carried out in accordance with the direction of the swing or the degree of inclination of the



(a) (b)

Fig. 3. (Color online) Display of hand gestures. (a) Gesture I and (b) Gesture II.

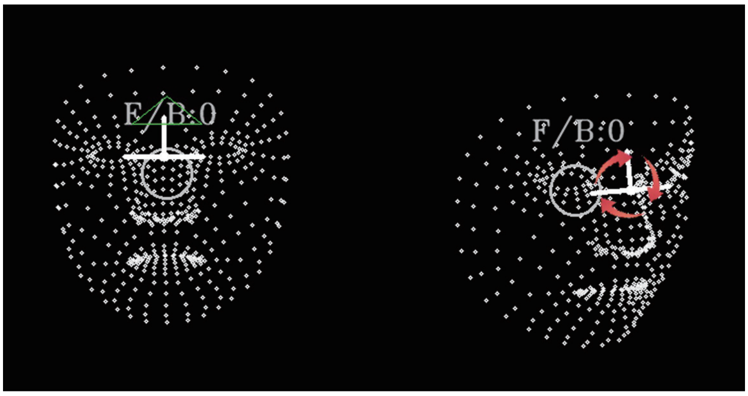


Fig. 4. (Color online) Head pose recognition.

head, as shown in Fig. 4. This form of control includes six directions of movement on the yaw axis, namely, left and right (Y axis), up and down (Z axis), and left and right yaw (rotation around the Z axis). The parameters of the API for transmitting the remote control instructions are specified in the tool. In the head pose recognition to control the drone, to prevent the control recognition of the head from being overly sensitive, a buffer (peripheral white circle) mechanism is added around the reference point (body). Provided the reference point does not exceed this buffer zone, slight movements of the head will not cause any commands to be transmitted to the drone.

4. Experimental Results

4.1 Implementation of system

With the intelligent drone control system produced in this research, real-time images of the drone can be sent back to the user. This allows remote control users to have greater control over the drone's environment, facilitating proper operation. The point on the fuselage located at the center of the interface captured by MediaPipe corresponds to the head point used to inform the control personnel of the state of their own head and quantifies the degree of tilting upward or downward. For example, when the control personnel want the drone to fly up to increase its altitude, the head is tilted upward. The angle by which the head is tilted upward controls the speed of the drone's ascent. This method can be applied to other direction controls. When the head is turned to the right, the drone swings to the right, and when the head is turned to the right so that the angle between the midline of the head and the horizontal is no longer 90°, the drone flies to the right. The greater the angle difference, the higher the speed of the motion. In summary, via the human–machine interface, the movement of the head can control the drone in six main directions: up, down, left, right, left yaw, and right yaw. The forward and backward motions are controlled by right-hand gestures. To increase the accuracy of control and reduce misjudgment, the control commands are designed to be transmitted in accordance with specific right-hand gestures. The distance between the thumb and the index finger is used to quantify the reference speed at which the drone moves forward or backward. This concept is similar to throttle control. The above series of head and hand movements provides eight movement modes of the drone when flying. The image recognition of the head and right hand is shown in Fig. 5.

4.2 Flight mission and corresponding experimental results

To verify the effectiveness of the proposed control method, we designed a test flight mission. In the test, obstacles blocked the user's line of sight, making it necessary to control the drone using real-time images. The flight mission is shown in Fig. 6 and the remote control system of the drone is shown in Fig. 7. To complete the flight mission, the drone must take off smoothly from the starting point, and after passing through a 70-cm-wide door, the drone enters an area that the control personnel cannot see directly. Then, the real-time images obtained from the drone are used by the user to control the flight and direct the drone to pass through a box with a length and width of 40 cm, circle back to the starting point, and land smoothly.

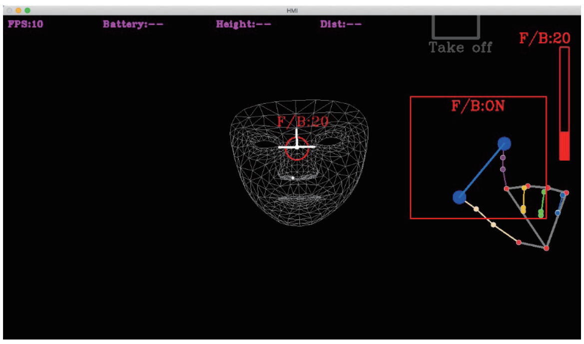


Fig. 5. (Color online) Image recognition of head and right hand.

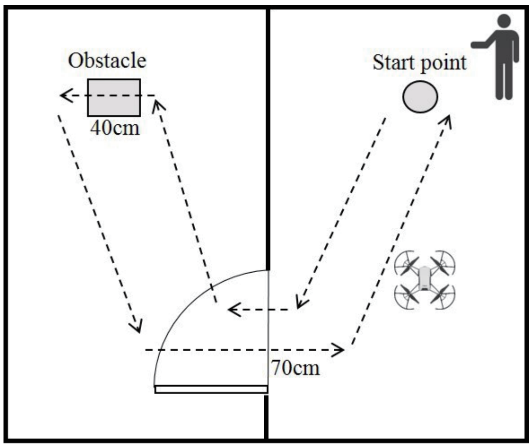


Fig. 6. Path of flight mission.

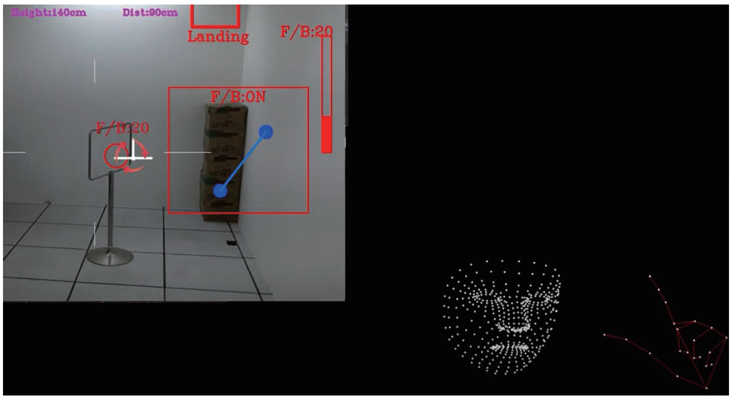


Fig. 7. (Color online) Drone remote control system.

We also set up an experimental group and a control group for further flight tasks. The experimental group used the proposed human–machine interface to perform somatosensory control, whereas the control group used the Tello preset and mobile app to complete the flight task. There were 10 testers, six women and four men, aged between 15 and 35, with no experience in operating drones. The testers were randomly divided into two groups of five people. Each tester was required to complete the flight task three times. It was found from the experimental results of the flight missions that when using the Tello preset and mobile app as the remote controller, the time required to complete the task was longer than that using our proposed remote control system. During the experiment, it was also observed that the user often confused the left and right hands and the cross-shaped buttons when manipulating the drone to fly forward, mainly because the control method is not intuitive. However, participants who used the proposed somatosensory control method could complete the flight mission in a short time, and the required time greatly decreased with increasing number of times of use. This result shows that the system is easier to use than the Tello preset or mobile device and enables skillful control. In addition, since our proposed motion control method relates the magnitude of the speed in each direction to the magnitude of the movement of the head and fingers, the precise control of flight movements is possible. From the experimental results, we concluded that the proposed system of somatosensory drone control can be operated smoothly and easily. Moreover, compared with the traditional method of controlling the drone, the user can complete the same flight mission in a shorter time with higher accuracy.

5. Conclusions

Traditionally, the primary methods of controlling drones have been remote control and the use of a handheld device. By these methods, an obstacle along the route of the drone that blocks the user's vision makes it difficult to control the drone smoothly. However, it would be very convenient to perform tasks in an area outside the direct view of the operator if appropriate sensors could be mounted on the drones. Such drones would be applicable in a wide range of fields, and tasks that were difficult to achieve in the past could be completed. In view of this, we adopted the symmetrical egocentric viewing mode to control a drone by viewing the surrounding environment via a camera mounted on the drone. The intelligent drone remote control system detects the user's head pose and gestures as controller signals. This system uses MediaPipe to capture the user's movement and then annotates feature points and converts signals to commands to control the drone. The following conclusions were drawn from the experimental results. The proposed drone control system can run smoothly and is easy for the user to operate. Compared with the traditional control method, the same flight mission was completed more accurately and in a shorter time when our proposed system was used. We also verified that users of this system can indeed perceive the surrounding environment through the real-time images obtained from the remote drone, enabling more precise control and providing good on-the-spot experience. Many previously unachievable missions can also be achieved using such a system since the control personnel can perceive the environment surrounding the drone and can distinguish obstacles along the travel route, enabling more accurate flight control.

References

1 P. Salvine, M. Nicolescu, and H. Ishiguro: IEEE Rob. Auto. Mag. **18** (2011) 98. https://doi.org/10.5302/J.ICROS.2014.14.9027
2 O. Erat, W. Isop, A., D. Kalkofen, and D. Schmalstieg: IEEE Trans. Visual Comput. Graphics **24** (2018) 1437. https://doi.org/10.1109/TVCG.2018.2794058
3 K. Cho, M. Cho, and J. Jeon: Interact Comput. **29** (2017) 345. https://doi.org/10.1093/iwc/iww027
4 M. Sugimoto, G. Kagotani, H. Nii, N. Shiroma, F. Matsuno, and M. Inami: IEEE Trans. Visual Comput. Graphics **25** (2005) 54. https://doi.org/10.1109/MCG.2005.23
5 C. J. Chen, E. C. Haas, and M. J. Barnes: IEEE Trans. Syst. **37** (2007) 1231. https://doi.org/10.1109/TSMCC.2007.905819
6 C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C. L. Chang, M. G. Yong, J. Lee, W. T. Chang, W. Hua, M. Georg, and M. Grundmann: ArXiv. 1906.08172 (2019). https://doi.org/10.48550/arXiv.1906.08172
7 F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C. L. Chang, and M. Grundmann: ArXiv. 2006.10214 (2020). https://doi.org/10.48550/arXiv.2006.10214
8 A. M. AlNuimi and G. J. Mohammed: Proc. Int. Conf. Contemporary Information Technology and Mathematics (Iraq 2021) 185–190. https://doi.org/10.1109/ICCITM53167.2021.9677878
9 A. Tamboli: Build Your Own IoT Platform (Apress, Berkeley, CA, 2022). https://doi.org/10.1007/978-1-4842-8073-7_1
10 C. S. Sharp, O. Shakernia, and S. S. Sastry: Proc. IEEE Int. Conf. Robotics and Automation (IEEE, Korea 2001) 1720–1727. https://doi.org/10.1109/ROBOT.2001.932859
11 B. Samir: Lausanne, EPFL (2007) https://doi.org/10.5075/epfl-thesis-3727
12 A. Zulu and S. John: Open J. Appl. Sci. **4** (2014) 547. https://doi.org/10.4236/ojapps.2014.414053
13 J. S. Park, S. Y. Oh, and H. L. Choi: Proc. Int. Conf. Control Auto. Syst. (Korea 2021) 271–274. https://doi.org/10.23919/ICCAS52745.2021.9650001
14 K. Subash, M. Srinu, M. Siddhartha, N. Harsha, and P. Akkala: Proc. Int. Conf. Asia Pac. Bus. Innov. Technol. Manag. (India, 2020) 484–490. https://doi.org/10.1109/ICIMIA48430.2020.9074881
15 R. A. Jarvis: IEEE Trans. Pattern Anal. Mach. Intell. **5** (1983) 122. https://doi.org/10.1109/TPAMI.1983.4767365
16 K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov: Commun. ACM **55** (2012) 61. https://doi.org/10.1145/2184319.2184337