IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 30, 2018, accepted June 11, 2018, date of publication July 16, 2018, date of current version August 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2856251

Smart Campus Care and Guiding With Dedicated Video Footprinting Through Internet of Things Technologies

LIEN-WU CHEN[1], (Senior Member, IEEE), TSUNG-PING CHEN[1], DA-EN CHEN[1], JUN-XIAN LIU[1], AND MING-FONG TSAI[2]
[1]Department of Information Engineering and Computer Science, Feng Chia University, Taichung 407, Taiwan
[2]Department of Electronic Engineering, National United University, Miaoli 360, Taiwan

Corresponding author: Lien-Wu Chen (lwuchen@fcu.edu.tw)

This work was supported by the Ministry of Science and Technology, Taiwan, under Grant 106-2221-E-035-019-MY3.

ABSTRACT In this paper, we propose a smart campus care and guiding framework with deep learning-based face recognition, called DeepGuiding, for students through Internet of Things technologies. The DeepGuiding framework can construct the dedicated video trajectory of a campus student, where the recorded video for each student can be automatically classified to achieve efficient footprint review as necessary. In addition, DeepGuiding can provide time-efficient indoor and outdoor guiding in a campus to quickly reach places, meet friends, and find students. To the best of our knowledge, DeepGuiding is the first campus care and guiding system which provides the following features: 1) it achieves the seamless outdoor and indoor navigation between buildings in a campus; 2) it keeps additional construction cost low by utilizing existing surveillance cameras in a campus; and 3) it reduces the total searching time for finding a specific event/target in a campus by alleviating time-consuming labor overhead to review a huge amount of video data. An Android-based prototype using iBeacon indoor localization and global positioning system outdoor positioning with surveillance cameras is implemented to verify the feasibility and superiority of our DeepGuiding framework. The Experimental results show that DeepGuiding outperforms existing face recognition methods and can achieve high recognition accuracy for students not close to surveillance cameras.

INDEX TERMS Face detection, face recognition, indoor positioning, Internet of Things, mobile device.

I. INTRODUCTION

The rapid development of mobile communication and Internet of Things (IoT) technologies has made smart environments possible, such as smart building [1], community [2], city [3], etc. The increasing number of connected mobile devices can be used to collect and distribute environmental information through built-in sensors. In addition, the interconnected IoT objects can communicate and cooperate with connected mobile devices over the Internet, where objects can be simple tag and iBeacon nodes, complex sensors and actuators, and physical items and appliances with Bluetooth, LTE/4G, RFID, Wi-Fi, or ZigBee network interfaces. The global positioning system (GPS) is usually adopted in outdoor location-based services [4]; however, because GPS signals are blocked by building structures, the non-GPS systems using RFID [5], iBeacon [6], or Wi-Fi [7] signals have to be used for indoor object and individual positioning [8], [9]. Mobile devices utilizing indoor positioning have been used to develop innovative applications and systems for individual-based path planning [10], group-based emergency evacuation [11], and microlocation-based geofencing [12].

On the other hand, recent studies for smart campuses of academic, industrial, and medical institutions have been proposed on a basis of IoT technologies [13]–[19]. The applications and services of IoT-enabled smart campuses have many benefits over traditional and digital campuses. For students and faculty, smart campuses can provide an environment to deliver interactive and creative services to the campus community. In addition, campus operating effectiveness can be enhanced for time and cost saving. Furthermore, comprehensive campus surveillance and real-time incident warning can be realized to improve campus safety [13].

Reference [14] used IoT communications to develop a room automation module with digital control in appliance access for energy saving. Reference [15] implemented an energy management system for energy-efficient campus buildings. The energy management system consists of wireless sensors and microcontrollers connecting to data readers and actuators for sensing room temperature and reacting upon corresponding heaters. Reference [16] focused on the analysis and optimization of integrating data from the existing energy and building management systems in a campus-scale IoT infrastructure. Optimizing the control of resources in a multi-building campus can improve energy saving and environmental sustainability by reducing energy use and waste as well as greenhouse gas emissions.

Reference [17] proposed a model for future IoT-based healthcare systems, which consists of wearable sensors, short-range and long-range communications, cloud-based storage, and machine learning. In particular, IoT-based healthcare systems are facing security, privacy, wearability, and low-power operation challenges. In the field of IoT-based healthcare systems, it is critical to develop machine learning for medical data diagnosis and to provide secure encryption with lightweight computing for cloud storage. Reference [18] presented a series of services implemented in the buildings of a smart campus for energy efficiency. The implemented services include indoor localization estimation, building energy consumption prediction, and comfort provisioning and energy saving optimization. The interaction decisions between occupants and automated devices are made by the implemented system to keep comfort conditions while saving energy. Reference [19] designed a hybrid naming scheme to name contents and devices for the IoT-based smart campus environment. In the designed scheme, the name assigned to a specific content is based on content features consisting of IoT application prefix, hierarchical, flat-hash, and attribute components. In addition to addressing and naming, scalability and security can be provided to campus contents and IoT devices. However, the current studies in the literature do not explore the dedicated video footprint of a student who needs special care in a smart campus. In addition, the fastest guiding path to the current location of the special-care student is not provided in an instant manner.

In this work, we design a smart campus care and guiding framework with deep learning based face recognition, called DeepGuiding, for students through IoT technologies, which integrates GPS outdoor localization and iBeacon indoor positioning with surveillance cameras, as shown in Fig. 1. The DeepGuiding system can automatically classify the recorded video clips for each student according to his/her face and thus can efficiently achieve video footprint review for a particular student. In addition, DeepGuiding can provide time-efficient indoor and outdoor guiding in a campus to quickly find a place to be reached, a friend to be met, and a student to be cared. In particular, DeepGuiding alleviates time-consuming labor overhead to find a specific event/target in a huge amount of video data and thus can significantly reduce total searching time in a campus as necessary.

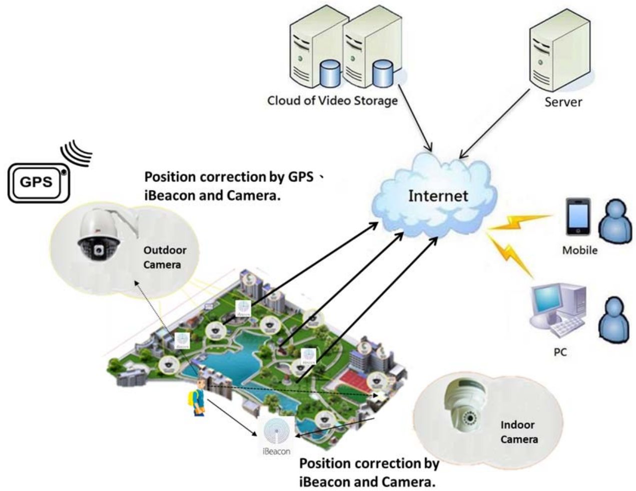


FIGURE 1. System architecture of DeepGuiding.

To the best of our knowledge, DeepGuiding is the first campus care and guiding system which provides the following features: 1) it achieves the seamless outdoor and indoor navigation between buildings in a campus, 2) it keeps additional construction cost low by utilizing existing surveillance cameras in a campus, and 3) it reduces the total searching time for finding a specific event/target in a campus by automatically classifying the recorded video clips for each student. An Android-based prototype is implemented by commercial iBeacon nodes and built-in GPS receivers with surveillance cameras to verify the feasibility and superiority of our DeepGuiding system. Experimental results show that DeepGuiding outperforms existing face recognition methods and can achieve high recognition accuracy for students not close to surveillance cameras.

The rest of this paper is organized as follows. Section II defines our smart campus care and guiding problem. Section III proposes our framework to solve this problem. Section IV shows the system implementation of our framework. Experimental results are discussed in Section V. Finally, Section VI concludes the paper.

II. SYSTEM MODEL

Fig. 1 shows the system architecture of DeepGuiding. On the campus side, face positioning units (FPUs) are installed to record videos and recognize the faces of students in outdoor and indoor areas. The FPU is consisting of a Wi-Fi/4G network interface to access the Internet, a surveillance camera to record students' videos, a database to map image pixels to actual locations, and a microprocessor to recognize face identifiers. In addition, for certain areas that are not monitored by FPUs (due to limited construction budget), iBeacon nodes are deployed to provide position information in those areas without FPUs installed. In particular, the recorded videos of students have to automatically classified and stored in dedicated directories according to the recognized face identifiers in these videos. Thus, teachers and guardians can perform the quick video footprint review of a target student for special care or emergency.

On the student side, a smart handheld unit (SHU) is carried by a campus student to receive his/her current position and guiding information from the DeepGuiding server. The SHU is consisting of a Wi-Fi/4G network interface for Internet access and a microprocessor for video footprint backtracking and guiding path planning in the campus. In addition, teachers and guardians can use a personal computer or a mobile device to find the campus footprint of a student who needs special care. With a proper number of surveillance cameras installed at campus spaces to monitor critical areas, the video footprint of students could be completely recorded and automatically classified. With a sufficient number of iBeacon nodes deployed in the areas not covered by FPUs (i.e., no footprint video), the up-to-date position information can be provided to students, teachers, and guardians for campus guiding and special care. The goal is to achieve seamless and rapid outdoor and indoor navigation for students and teachers in a campus by addressing the following four research issues:

1) Fast Campus Guiding: How do we instantly obtain the up-to-date positions in both outdoor and indoor campus spaces and plan the fastest path from the current location to the destination in a seamless manner?
2) Accurate Face Detection: How do we precisely detect the faces in the live videos of surveillance cameras based on continuous detected frames and detection window sizes?
3) Deep Face Recognition: How do we correctly recognize the identifiers of detected faces not close to surveillance cameras by exploring deep learning based face recognition?
4) Individual Video Classification: How do we automatically classify and store the recorded videos of students in dedicated directories according to the recognized face identifiers in surveillance videos?

III. SMART CAMPUS CARE AND GUIDING

Fig. 2 shows the positioning flowchart of DeepGuiding. Campus students use the DeepGuiding App to login to the DeepGuiding server with their registered user accounts. For DeepGuiding users, there are three manners (with different positioning errors) to obtain the up-to-date positions. The first manner is when GPS satellite signals are available (usually in outdoor spaces), the GPS location is used as the current position of the SHU and transmitted to the DeepGuiding server. The second manner is when iBeacon node signals are available (usually in indoor spaces), the SHU transmits the received iBeacon ID (with the strongest signal strength) to the DeepGuiding server and the actual location associated with the iBeacon ID is replied to the SHU as the current position. The last manner is when a FPU detects and recognizes the face of a DeepGuiding user, the actual location (mapped from the coordinate of the recognized face) is transmitted to the DeepGuiding server and forwarded to the SHU of the DeepGuiding user as the current position. Note that when two or more locations are simultaneously obtained through GPS, iBeacon, and FPU positioning, the priority of FPU locations is the highest (because its positioning error is the lowest), whereas that of GPS locations is the lowest (because its positioning error is the highest) [8].

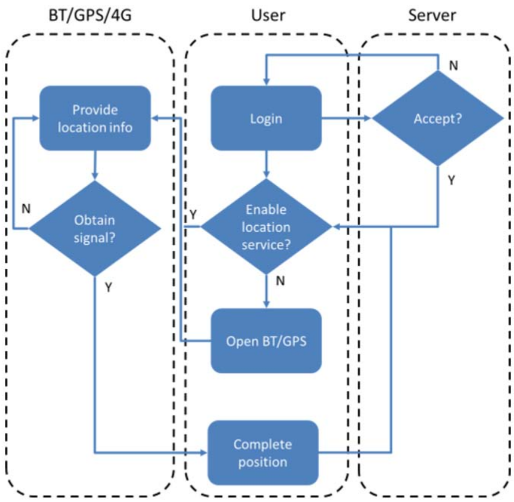


FIGURE 2. Positioning flowchart of DeepGuiding.

The following are the steps of FPU positioning. First, a FPU monitors arriving students in the surveillance area. When one or more students approach the FPU, the videos of these students are recorded by the FPU. Second, the faces of all students in the recorded videos are detected and recognized using the Fisherface-based face recognition [20], as shown in Fig. 3. Third, the center pixel of each face on the video is mapped to the actual location in the surveillance area. Next, the mapped location and associated user account of each student are transmitted to the DeepGuiding

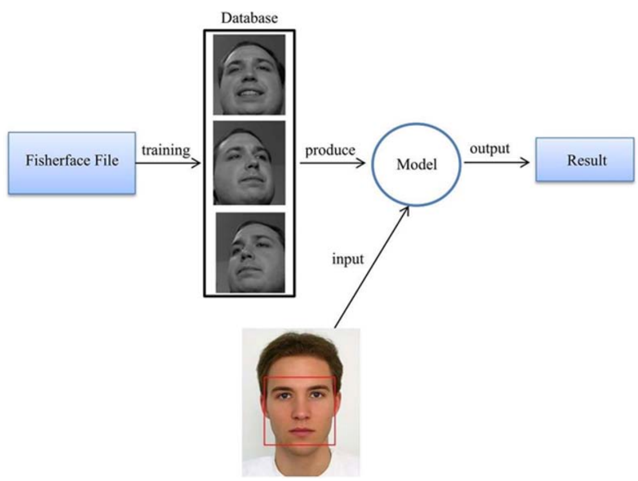


FIGURE 3. Fisherface-based face recognition in DeepGuiding.

server. Finally, the DeepGuiding server sends the location message to the corresponding SHU based on the associated user account, where the location message contains the mapped position of the recognized face. When the corresponding SHU receives the location message, it can use the received up-to-date position to enable location-based services in smart campuses, such as individual-based path planning, group-based emergency evacuation, microlocation-based geofencing, etc.

However, on a basis of our experimental results, the face recognition accuracy is significantly reduced when students are not close to surveillance cameras. To improve the recognition accuracy of students' faces, we explore convolutional neural networks (CNN) to design a deep learning based face recognition approach with similar execution time to the Fisherface-based face recognition. The CNN is a class of deep artificial neural networks (ANN) consisting of an input layer, multiple hidden layers, and an output layer, as shown in Fig. 4. An ANN is a collection of connected artificial neurons (i.e., circles in Fig. 4) similar to biological neurons in a human brain, where each artificial neuron has its activation function, weight, and bias. The input is fed to the input layer and a linear transformation is done on the input by the weights and biases of the artificial neurons, whereas a non-linear transformation is performed by the activation function. The weights and biases of the artificial neurons can be updated through back-propagation based on the error between the actual value and output value, as shown in Fig. 5.

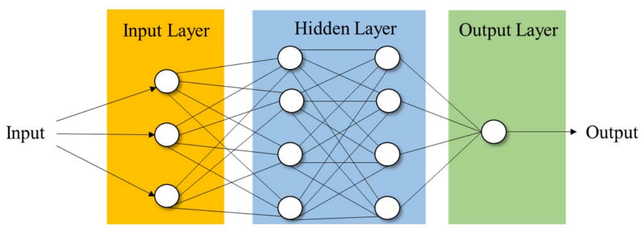


FIGURE 4. Layers of artificial neural networks.

A signal can be transmitted from one artificial neuron to another artificial neuron through the connection (i.e., lines in Fig. 4) between them. The artificial neuron can process the received signal and an activated artificial neuron can signal other connected artificial neurons. The adopted activation function decides whether an artificial neuron is activated or not, where the activation function is the non-linear transformation and the transformed output is the input for the next layer of artificial neurons. Three most commonly used activation functions including Sigmoid, Tanh, and ReLU in an ANN are

$$f(x) = \frac{1}{1+e^{-x}}, \tag{1}$$

$$f(x) = \frac{2}{1+e^{-2x}} - 1, \tag{2}$$

and

$$f(x) = max(0, x), \tag{3}$$

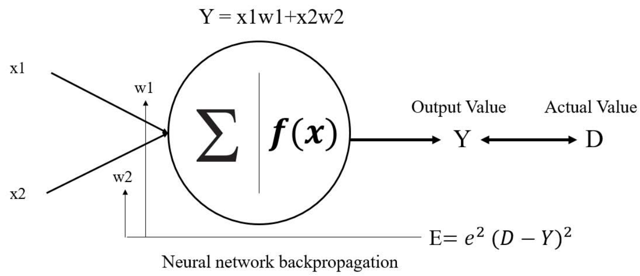


FIGURE 5. Computing of artificial neurons.

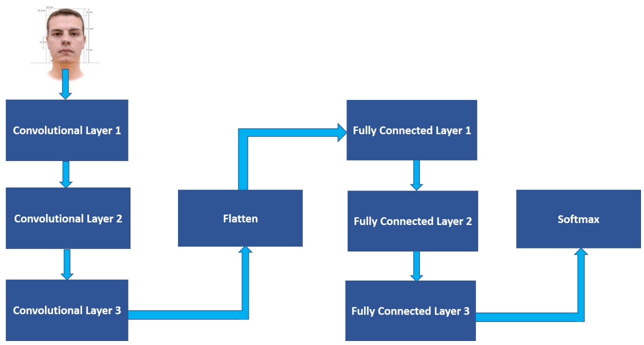


FIGURE 6. Our CNN architecture for face recognition.

respectively. Sigmoid activation function has an S shape ranging from 0 to 1, whereas Tanh activation function is similar to Sigmoid activation function but ranging from -1 to 1. ReLU activation function converts the input to zero if the input is negative, where the zero-input neuron is not activated. By using ReLU activation function, only a few non-zero-input neurons are activated at a time, which can efficiently reduce the computation time of hidden layers.

In particular, the hidden layers of a CNN include convolutional layers, pooling layers, fully connected layers, and normalization layers. Convolutional layers perform a convolution operation to emulate the neuron response as visual stimuli occurs, and the convoluted result is used as the input of the next layer. Pooling layers are used to combine the outputs of a neuron group as the input of a single neuron in the next layer. For instance, the maximum value is adopted by the MaxPooling layer from each group neuron of the previous layer, whereas the average value is adopted by the AvgPooling layer from each group neuron of the previous layer. Through fully connected layers, every artificial neuron in the previous layer can be connected to every artificial neuron in the next layer. Normalization layers are used to highlight the target values and suppress their values that are significantly below the original values.

To achieve high accuracy while keeping execution time low, our CNN architecture for face recognition is designed to consist of three convolutional layers (extracting fine-grained features), one flatten layer (converting 2-dimensional matrix to 1-dimensional output), three fully connected layers (calculating the probability of matching face identifiers), and one normalization layer (using the Softmax function), as shown in Fig. 6. For local feature extraction, convolution filtering,

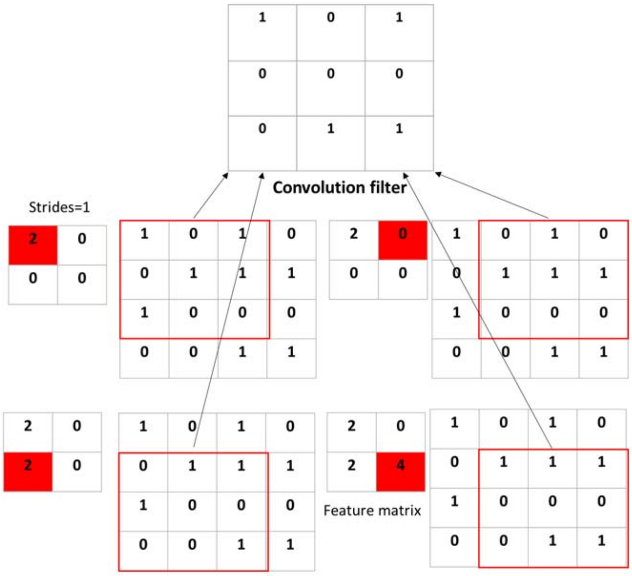


FIGURE 7. Convolution filter.

ReLU activation function, and MaxPooling reduction are used to convert the input face image to its feature matrix. As shown in Fig. 7, for example, suppose that the input is an 4×4 image, the convolution filter is a 3×3 matrix, and the stride length is 1. A 2×2 feature matrix can be obtained by taking the sum of products of the 3×3 filter matrix and each 3×3 sub-image moving with stride 1 in the input image (i.e., top-left, top-right, bottom-left, and bottom-right sub-squares with red lines in Fig. 7). Next, ReLU activation function is used to obtain one of two extreme values, zero for negative input and the original value for non-negative input. Finally, MaxPooling reduction is used to extract the maximum value of surrounding pixels as local features for reducing computation time, as shown in Fig. 8.

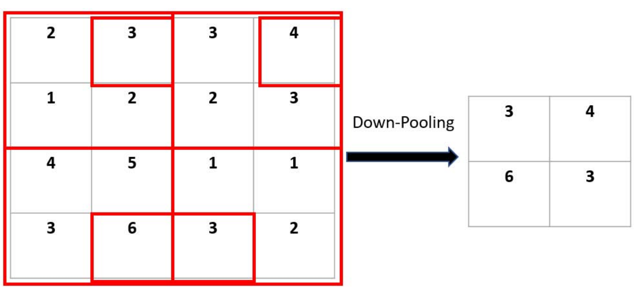


FIGURE 8. MaxPooling reduction.

In the designed CNN model, 3×3, 2×2, and 2×2 filter matrices are used to extract 32, 64, and 128 features in the first, second, and third convolution layers, respectively. In addition, the input and output of the second convolution layer are the output (i.e., the least fine-grained features) of the first convolution layer and the input (i.e., the more fine-grained features) of the third convolution layer, respectively. Suppose that the output (i.e., the most fine-grained features) of the third convolution layer is $n \times n$ matrix. The flatten layer is used to convert 2-dimensional $n \times n$ matrix to 1-dimensional n^2 output values as follow.

$$M_{ij} \rightarrow M_{(i-1)\times n+j}, \tag{4}$$

for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$. Next, the fully-connected layers are used to calculate the probability of the input face image matching to each possible face identifier. In addition, the dropout function is used to randomly close 25% artificial neurons to avoid overfitting. Finally, the Softmax function is used to convert a K-dimensional real-value vector z to a K-dimensional real-value vector $\sigma(z)$ as follow, where K is the number of possible face identifiers, each element value in $\sigma(z)$ is between 0 and 1, and the sum of all element values in $\sigma(z)$ is equal to 1.

$$\sigma(z)_j = \frac{e^{z_j}}{\Sigma_{k=1}^{K} e^{z_k}}, \tag{5}$$

for $j = 1, 2, \ldots, K$. The face identifier with the largest element value in $\sigma(z)$ (which has to be greater than a predefined threshold) is selected as the recognized result of the input face image; otherwise, there is no recognized identifier for the input face image.

IV. SYSTEM IMPLEMENTATION

We have developed a prototype of DeepGuiding [21] using the built-in IEEE 802.11 interfaces of notebook computers and Android mobile devices connecting to a campus IEEE 802.11 access point with Internet access as the network interfaces of FPUs and SHUs, respectively. When campus students register to use the DeepGuiding App for the first time, their faces have to register to the DeepGuiding server associating with user accounts. A few front view and side view face photos have to be taken through the DeepGuiding App and sent to the DeepGuiding server for constructing the database of face recognition and video classification. The SAILS SDK [22] that supports the vector-typed and rotatable map with different render styles is adopted to implement the campus guiding in the Feng Chia University. In addition to iBeacon ID localization, the SAILS SDK can use the signals of Wi-Fi access points for indoor positioning. Furthermore, the SAILS SDK can improve the localization accuracy of Wi-Fi/iBeacon signals by fusing the accelerometer, compass, and gyro inertial sensors of mobile devices. Note that we further design and implement deep learning based face recognition in the prototype.

The FPU is implemented by a notebook computer with a video camera of 1920 × 1080 resolution, which is running on Windows 10 operating system and communicating with the DeepGuiding server through a campus IEEE 802.11 access point. In addition, iBeacon iB07-C2450 nodes (using the Bluetooth 4.0 chip of TI CC2541 [23]) powered by the coin battery of CR2450 are used in the non-FPU-deployed areas. The Eclipse JAVA integrated development environment is used to develop the graphical user interface and integrate with the Open Source Computer Vision (OpenCV [24]) and Google TensorFlow [25] libraries to implement face detection/recognition and video recording, classification, and streaming. The SHU is implemented by an Android mobile device that communicates with the DeepGuiding server through the built-in IEEE 802.11 interface and campus access point.

Fig. 9 and Fig. 10 show the graphical user interfaces of the DeepGuiding App and Website, respectively. The functions of the DeepGuiding App are including My Position, Live Scene, My Video, Student Care, Tracking Platform, and School Guide, as shown in Fig. 9(a). The dedicated video footprint of a student can be quickly reviewed if special events occur to the student, as shown in Fig. 9(b). The guiding path with the shortest distance can be planned between students in different buildings, as shown in Fig. 9(c). The rescue action and path planning can be performed if teachers/guardians find certain emergency event in either live video streaming or video footprint review, as shown in Fig. 9(d). On the other hand, the DeepGuiding Website is developed for teachers/guardians to perform video clip management and video data analysis through personal computers/tablets, as shown in Fig. 10.

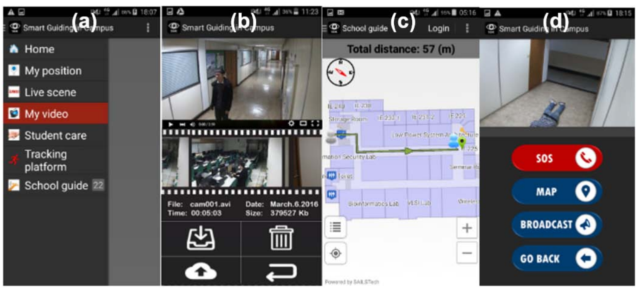


FIGURE 9. Graphical user interfaces of the DeepGuiding App.

FIGURE 10. DeepGuiding Website for teachers/guardians.

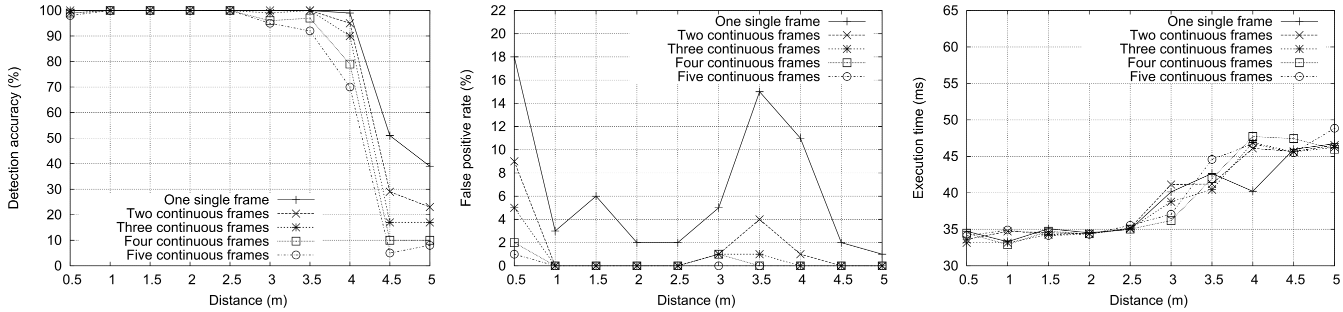
For indoor demonstration, several Android-based mobile devices with different mini dolls (and different human face photos) are used to simulate student moving in a monitored campus. The Android-based mobile device is used to obtain the position information from GPS, iBeacon, and FPU and to display the current position and associated identifier of the mini doll in the DeepGuiding App. The correctness of positioning results and classified videos can be verified by checking the received positions and live video streaming on the mobile device's screen, respectively. In addition, the specific emergency message for help can be sent to the mobile device of a specified teacher/guardian. The guiding path from the location of the specified teacher/guardian to the current position of the student can be immediately displayed on the mobile device's screen for quick rescue.

V. EXPERIMENTATIONS

In this section, to achieve accurate face detection and recognition results for video footprint review in DeepGuiding, we first select a feasible configuration for face detection based on the detection accuracy, false positive rate, and execution time under different numbers of continuous detected frames (i.e., a face is considered to be successfully detected only if the face is continuously detected in a predefined number of successive frames). Next, we find a suitable size of face detection windows used to scan the whole video frame for increasing detection success rates and reducing false positive rates. Finally, we compare face recognition accuracy and execution time using Fisher-based and CNN-based face recognition in bright (i.e., high lightness), dark (i.e., low lightness), and normal (intermediate lightness) indoor environments. The open face database of PICS [26] is adopted in DeepGuiding. Each experimentation is repeated 10 times by realistic trial and the average value is taken.

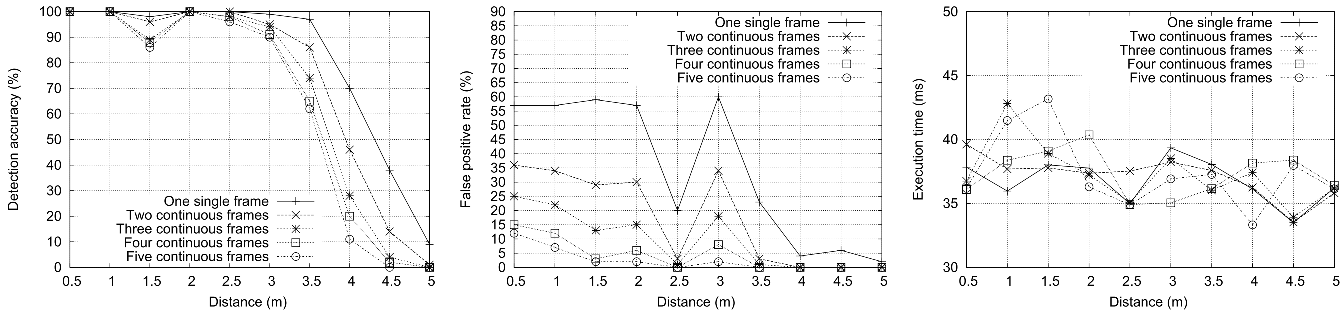
Fig. 11, Fig. 12, and Fig. 13 show performance comparisons of face detection using different numbers of continuous detected frames in three indoor environments with high, low, and intermediate lightness, respectively. We measure the face detection accuracy, false positive rate, and execution time for the distances of 0.5, 1, 1.5, ..., and 5 meters between the individual and surveillance camera through a notebook computer with Windows 10 operating system (Model: ASUS ZenBook UX330, CPU: Intel i7-7500U 2.7GHz, RAM: 8GB). From Fig. 11, Fig. 12, and Fig. 13, it can be observed that the accuracy of face detection significantly decreases (with similar execution time) as the distance between the individual and surveillance camera is greater than 3 meters. More importantly, although using a small number of continuous detected frames can detect more faces not close to the surveillance camera, it causes a high false positive rate. So we have to trade the detection accuracy off against false positive rates. Fig. 14 shows the average detection accuracy, false positive rate, and execution time of face detection in bright, dark, normal environments. It can be seen that using 3 continuous detected frames has very low false positive rates (especially in bright and normal environments) while keeping its detection accuracy and execution time competitive to other numbers of continuous detected frames. Therefore, to balance the detection accuracy and false positive rates of DeepGuiding, we set the number of continuous detected frames as 3 for face detection in the following experimentations.

On the other hand, Fig. 15, Fig. 16, and Fig. 17 show detection accuracy, false positive rate, and execution time comparisons of face detection using different sizes of face detection windows in bright, dark, and normal indoor environments, respectively. We can observe that the detection accuracy is drastically dropping as the detection windows size is larger than 20 (i.e., 20×20 square). In contrast, the false positive rates of face detection are very similar among using



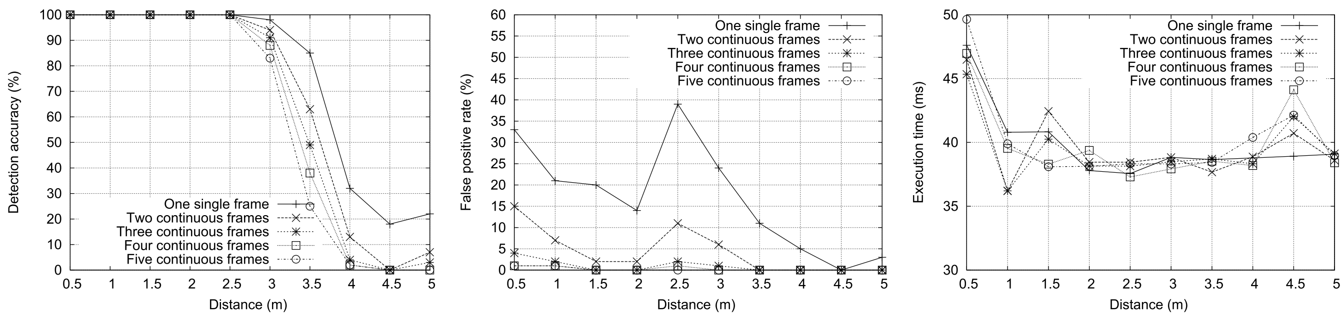
(a) (b) (c)

FIGURE 11. Comparisons of different numbers of continuous detected frames in the bright environment with high lightness. (a) Detection accuracy. (b) False positive rate. (c) Execution time.



(a) (b) (c)

FIGURE 12. Comparisons of different numbers of continuous detected frames in the dark environment with low lightness. (a) Detection accuracy. (b) False positive rate. (c) Execution time.



(a) (b) (c)

FIGURE 13. Comparisons of different numbers of continuous detected frames in the normal environment with intermediate lightness. (a) Detection accuracy. (b) False positive rate. (c) Execution time.

the detection window sizes of 5, 10, . . . , and 30. In addition, the execution time using the face detection window less than or equal to 20 only has a little increasing compared with that using detection window sizes of 25 and 30. As shown in Fig. 18, similar results are obtained for the average performance in three indoor environments. To achieve high detection accuracy and low false positive rates (with similar execution time), the face detection window of 20 is used in our experimentations.

Fig. 19 shows recognition accuracy and execution time comparisons of face recognition using Fisher-based and CNN-based face recognition in bright, dark, and normal environments. It can be observed that our designed CNN-based method has much higher recognition accuracy than the Fisher-based method in all environments with different lightness. In particular, even when the individual is not close to the surveillance camera, the recognition accuracy of our CNN-based method is more than 90%, whereas that of the Fisher-based method is only around 70% in the worst case (i.e., the dark environment with low lightness). On the other hand, the execution time of our CNN-based method is similar to that of the Fisher-based method. In DeepGuiding, the number of hidden layers of the CNN model is decreased to reduce the execution time of face recognition

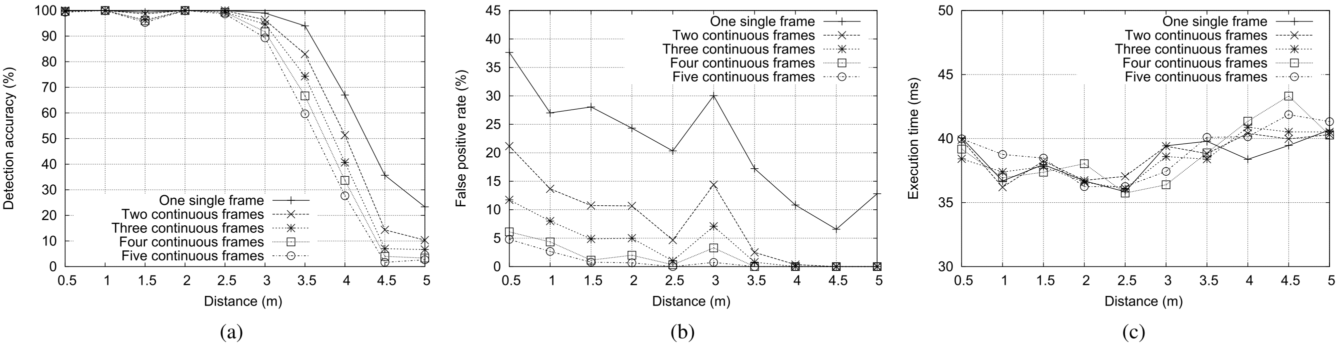


FIGURE 14. Comparisons of different numbers of continuous detected frames averagely in bright, dark, and normal environments. (a) Detection accuracy. (b) False positive rate. (c) Execution time.

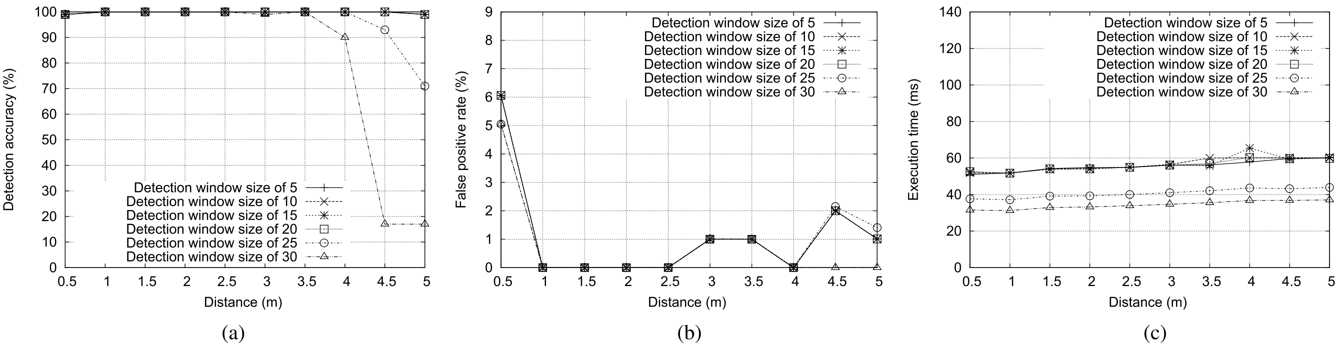


FIGURE 15. Comparisons of different sizes of face detection windows in the bright environment with high lightness. (a) Detection accuracy. (b) False positive rate. (c) Execution time.

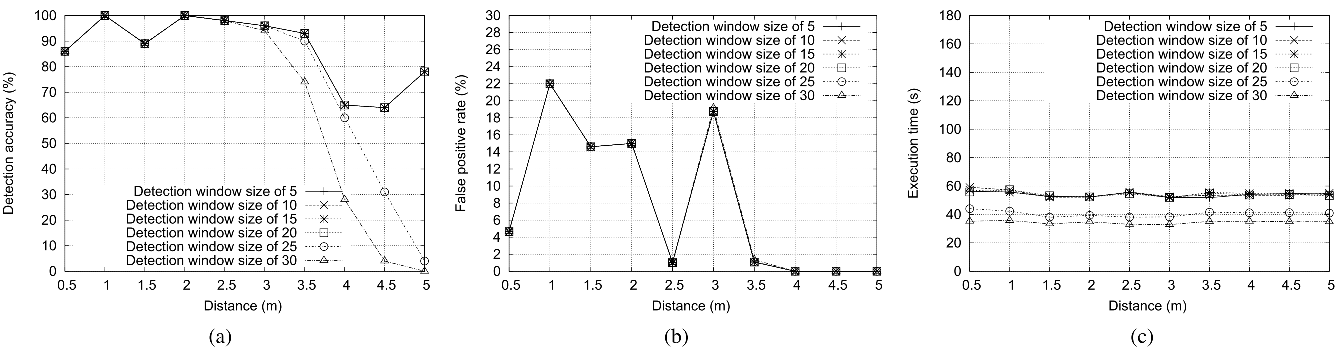


FIGURE 16. Comparisons of different sizes of face detection windows in the dark environment with low lightness. (a) Detection accuracy. (b) False positive rate. (c) Execution time.

(as shown in 19b) while keeping the recognition success rate high (as shown in Fig. Fig. 19a). In particular, through the proper design of the CNN model, the accuracy of face recognition can be significantly improved while slightly increasing execution time (i.e., less than 10 ms).

In particular, for video footprint review using the traditional surveillance system and DeepGuiding, a 30-minute live video is recorded and a specific event of wallet lost from a target student occurs at 20-th minute. Three different users are using the traditional surveillance system and DeepGuiding to find the location of wallet lost for the target student. The video clips without the target student are fast-forwarded with the interval of five seconds in the traditional surveillance system. Table 1 shows the comparisons of footprint review times using the traditional surveillance system and DeepGuiding. It can be observed that DeepGuiding can significantly reduce the footprint review time from eight-and-a-half minutes to

TABLE 1. Comparisons of footprint review times using the traditional surveillance system and DeepGuiding.

Method	User 1	User 2	User 3	Average (min:sec)
Traditional	8:37	9:42	7:18	8:32
DeepGuiding	2:47	2:32	2:07	2:29

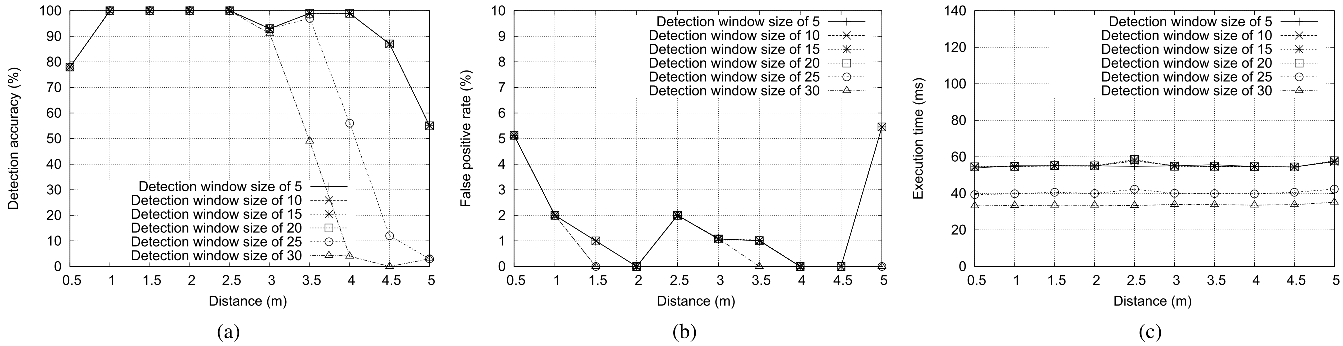


FIGURE 17. **Comparisons of different sizes of face detection windows in the normal environment with intermediate lightness. (a) Detection accuracy. (b) False positive rate. (c) Execution time.**

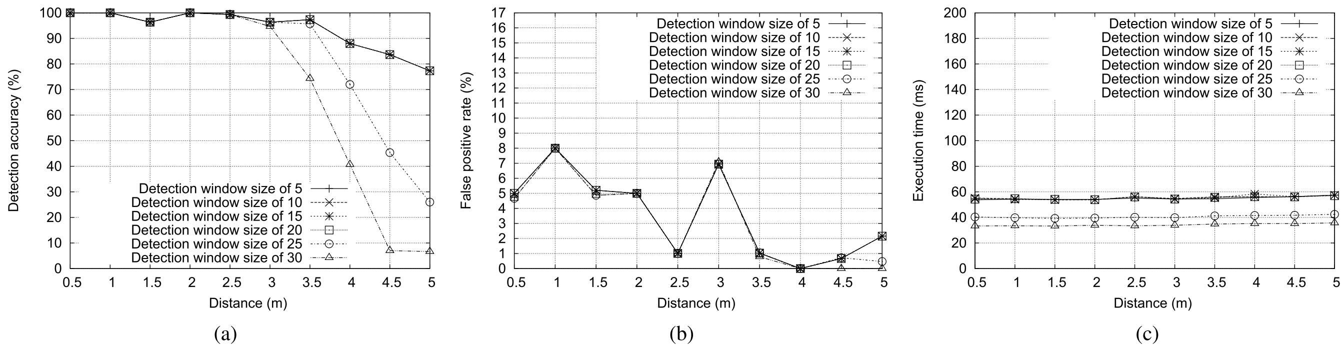


FIGURE 18. **Comparisons of different sizes of face detection windows averagely in bright, dark, and normal environments. (a) Detection accuracy. (b) False positive rate. (c) Execution time.**

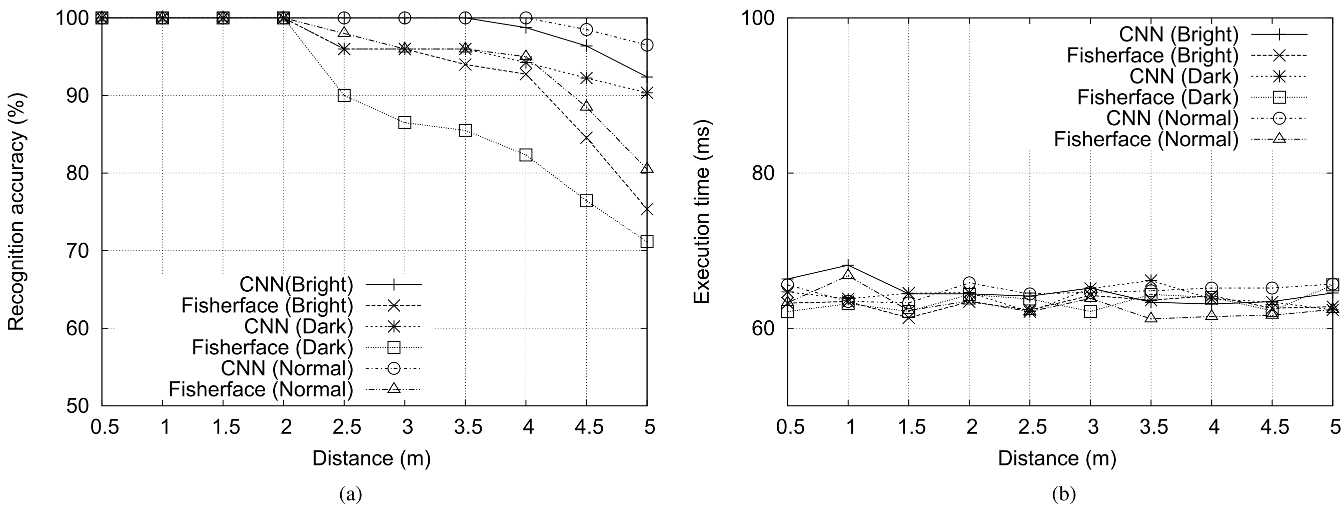


FIGURE 19. **Comparisons of recognition accuracy and execution time using Fisher-based and CNN-based face recognition in bright, dark, and normal environments. (a) Recognition accuracy. (b) Execution time.**

two-and-a-half minutes in average, where the ratio of time saving is about 70%. This is because only the video clips in which the target student appears need to be reviewed in DeepGuiding, whereas all the video clips have to be carefully checked in the traditional surveillance system.

On the other hand, for the hardware construction cost of DeepGuiding, we estimate the total costs with and without using existing surveillance cameras, respectively. The hardware construction cost consists of the surveillance camera, video recorder, cable line, cable connector, and power line, where the cost of each item is based on the online price of amazon.com. Table 2 shows the comparisons of hardware construction costs without and with using existing cameras in the second floor of Information and Electrical Engineering Building, Feng Chia University (as shown in Fig. 20). It can be seen that 24 new cameras (i.e., green and red circles in Fig. 20) have to be installed without using existing surveillance cameras, whereas only 10 new cameras (i.e., red circles in Fig. 20) need to be added with using existing surveillance cameras. The hardware construction costs

TABLE 2. Comparisons of hardware construction costs without and with using existing cameras.

Item Price	Surveillance Camera ($49.99/1pack)	Video Recorder ($499/16ch) ($1099/32ch)	Cable Line ($74.5/100feet)	Cable Connector ($6.99/20pack) ($9.99/40pack)	Power Line ($13.99/40feet) ($22.99/100feet)	Total Cost (U.S. Dallor)
Without Existing Cameras	$24 \times 49.99 = 1199.76$	1099	$74.5 \times 13 = 968.5$	26.97	206.1	3500.33
With Existing Cameras	$10 \times 49.99 = 499.9$	499	$74.5 \times 5 = 372.5$	9.99	105.95	1487.34

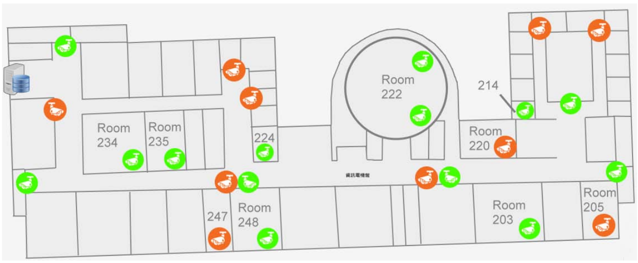


FIGURE 20. Existing surveillance cameras (i.e., 14 green circles) and additional installed cameras (i.e., 10 red circles) for DeepGuidng in the second floor of Information and Electrical Engineering Building, Feng Chia University.

without and with using existing surveillance cameras are 3500 and 1487 U.S. dollars, respectively, where the ratio of cost saving is about 58%.

VI. CONCLUSIONS

In this work, we design, implement, and evaluate a smart campus care and guiding framework with deep learning based face recognition for students through IoT technologies. The proposed framework can efficiently alleviate the labor overhead for reviewing a lot of recorded videos and significantly reduce the searching time for finding a target student who needs to be cared in a campus. In other words, adopting our framework in campus care and guiding can both avoid destination location acquisition consuming time due to unfamiliar environments and prevent teachers/guardians from watching many unrelated video clips of non-target students recorded by a large number of surveillance cameras. Furthermore, experimental results conduced from the prototype of DeepGuiding framework show that our deep learning based face recognition achieves high recognition accuracy while keeping execution time low.

REFERENCES

[1] D. Minoli, K. Sohraby, and B. Occhiogrosso, "IoT considerations, requirements, and architectures for smart buildings—Energy optimization and next-generation building management systems," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 269–283, Feb. 2017.

[2] S. Kubler, J. Robert, A. Hefnawy, K. Främling, C. Cherifi, and A. Bouras, "Open IoT ecosystem for sporting event management," *IEEE Access*, vol. 5, no. 4, pp. 7064–7079, Apr. 2017.

[3] Y. Mehmood, F. Ahmad, I. Yaqoob, A. Adnane, M. Imran, and S. Guizani, "Internet-of-Things-based smart cities: Recent advances and challenges," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 16–24, Sep. 2017.

[4] K. Liu, H. B. Lim, E. Frazzoli, H. Ji, and V. C. S. Lee, "Improving positioning accuracy using GPS pseudorange measurements for cooperative vehicular localization," *IEEE Trans. Veh. Technol.*, vol. 63, no. 6, pp. 2544–2556, Jul. 2014.

[5] S. L. Ting, S. K. Kwok, A. H. C. Tsang, and G. T. S. Ho, "The study on using passive RFID tags for indoor positioning," *Int. J. Eng. Bus. Manage.*, vol. 3, no. 1, pp. 9–15, Feb. 2011.

[6] *iBeacon Insider: Your Guide to iBeacon Technology*. Accessed: Apr. 30, 2018. [Online]. Available: http://www.ibeacon.com

[7] S.-P. Kuo and Y.-C. Tseng, "A scrambling method for fingerprint positioning based on temporal diversity and spatial dependency," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 5, pp. 678–684, May 2008.

[8] L.-W. Chen, C.-R. Chen, and D.-E. Chen, "VIPS: A video-based indoor positioning system with centimeter-grade accuracy for the IoT," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2017, pp. 63–65.

[9] T. Kulshrestha, D. Saxena, R. Niyogi, V. Raychoudhury, and M. Misra, "SmartITS: Smartphone-based identification and tracking using seamless indoor-outdoor localization," *J. Netw. Comput. Appl.*, vol. 98, no. 11, pp. 97–113, Nov. 2017.

[10] L.-W. Chen, J.-H. Cheng, and Y.-C. Tseng, "Optimal path planning with spatial-temporal mobility modeling for individual-based emergency guiding," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 12, pp. 1491–1501, Dec. 2015.

[11] L.-W. Chen and J.-J. Chung, "Mobility-aware and congestion-relieved dedicated path planning for group-based emergency guiding based on Internet of Things technologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2453–2466, Sep. 2017.

[12] F. Zafari, I. Papapanagiotou, and K. Christidis, "Microlocation for Internet-of-Things-equipped smart buildings," *IEEE Internet Things J.*, vol. 3, no. 1, pp. 96–112, Feb. 2016.

[13] A. Abuarqoub *et al.*, "A survey on Internet of Things enabled smart campus applications," in *Proc. Int. Conf. Future Netw. Distrib. Syst. (ICFNDS)*, Jul. 2017, Art. no. 50.

[14] M. Jain, N. Kaushik, and K. Jayavel, "Building automation and energy control using IoT—Smart campus," in *Proc. Int. Conf. Comput. Commun. Technol. (ICCCT)*, Feb. 2017, pp. 353–359.

[15] N. Najem, D. B. Haddou, M. R. Abid, H. Darhmaoui, N. Krami, and O. Zytoune, "Context-aware wireless sensors for IoT-centeric energy-efficient campuses," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, May 2017, pp. 1–6.

[16] O. Bates and A. Friday, "Beyond data in the smart city: Repurposing existing campus IoT," *IEEE Pervasive Comput.*, vol. 16, no. 2, pp. 54–60, Apr. 2017.

[17] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of Things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, no. 11, pp. 26521–26544, Nov. 2017.

[18] M. V. Moreno *et al.*, "Applicability of big data techniques to smart cities deployments," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 800–809, Apr. 2017.

[19] S. Arshad, M. A. Azam, S. H. Ahmed, and J. Loo, "Towards information-centric networking (ICN) naming for Internet of Things (IoT): The case of smart campus," in *Proc. Int. Conf. Future Netw. Distrib. Syst. (ICFNDS)*, Jul. 2017, Art. no. 41.

[20] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

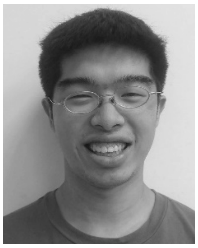
[21] L.-W. Chen, T.-P. Chen, and D.-E. Chen, "iGuiding: A mobile campus care and guidance system based on Internet of Things technologies," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2018, p. 3.

[22] *SAILS SDK*. Accessed: Apr. 30, 2018. [Online]. Available: http://support.sailstech.com/kb

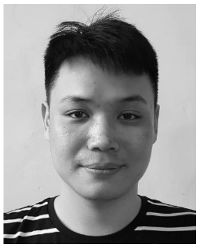
[23] *TI CC2541—SimpleLink Bluetooth Smart and Proprietary Wireless MCU*. Accessed: Apr. 30, 2018. [Online]. Available: http://www.ti.com/product/cc2541

[24] *Open Source Computer Vision*. Accessed: Apr. 30, 2018. [Online]. Available: http://opencv.org
[25] *TensorFlow—An Open-Source Software Library for Machine Intelligence*. Accessed: Apr. 30, 2018. [Online]. Available: https://www.tensorflow.org
[26] *Psychological Image Collection at Stirling*. Accessed: Apr. 30, 2018. [Online]. Available: http://pics.psych.stir.ac.uk/

LIEN-WU CHEN (SM'16) received the Ph.D. degree in computer science and information engineering from National Chiao Tung University in 2008. He joined the Department of Computer Science, National Chiao Tung University, as a Post-Doctoral Research Associate in 2009, and qualified as an Assistant Research Fellow in 2010. From 2012 to 2015, he was an Assistant Professor with the Department of Information Engineering and Computer Science, Feng Chia University, where he has been an Associate Professor since 2015 and the Chief of the Operation and Maintenance Section since 2016. His research interests include wireless communication and mobile computing, especially in the Internet of Vehicles, Internet of Things, and Internet of People. He is currently an Honorary Member of the Phi Tau Phi Scholastic Honor Society, a Lifetime Member of CIEE, IICM, and TIEEE, and a member of the ACM. He received over 20 best/merit paper awards, best/outstanding demo awards, outstanding paper publication awards, and prototyping awards, since 2009, and transferred three technologies to Acer Inc., in 2011. He has been the advisor of over 10 student engineering paper awards, college student research awards, and master thesis awards, since 2014.



TSUNG-PING CHEN is currently pursuing the degree with the Department of Information Engineering and Computer Science, Feng Chia University. His research interests include the Internet of Things and mobile applications and systems. He was a recipient of the Student Engineering Paper Award of the Chinese Institute of Engineers in 2017 and 2018.



DA-EN CHEN received the B.S. and M.S. degrees from the Department of Information Engineering and Computer Science, Feng Chia University, in 2015 and 2017, respectively. His research interests include vehicular ad hoc networks and intelligent transportation systems. He was a recipient of the Best Paper Award and the Best Presentation Award at the 22nd Mobile Computing Workshop in 2017.

JUN-XIAN LIU received the B.S. degree from the Department of Information Engineering and Computer Science, Feng Chia University, in 2014, where he is currently pursuing the Ph.D. degree. His research interests include the Internet of Things and intelligent transportation systems.

MING-FONG TSAI received the Ph.D. degree from the Department of Electrical Engineering, Institute of Computer and Communication Engineering, National Cheng Kung University, Taiwan. He is currently an Assistant Professor with the Department of Electronic Engineering, National United University, Taiwan. His current research interests include error-control coding, multimedia communications, and vehicular communications.