International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.7 (2016), pp.321-330
http://dx.doi.org/10.14257/ijmue.2016.11.7.32

Vision Based Natural Smart Phone Interface using Face Detection for Optimal UX

Young Jae Lee[1]

[1]*Dept. of Smartmedia Jeonju University, 303 Cheonjam-ro, Wansan-gu, Jeonju, 56069, Korea*
leeyj@jj.ac.kr

Abstract

User interface is the key technology in high-value smart phone applications services. This paper suggests a user interface algorithm and tests its validity through experiments. User's face is detected in the input image on the smartphone camera sensor. Based on the sensing information about the detected face area, analysis is carried out to infer the locations for the eyes and the mouth. With the resulting information, the nose is detected. To identify the movements of each area, image-processing methods are applied to the eyes, the nose and the mouth. An algorithm is proposed to use them as interfaces. To test the effectiveness of the proposed algorithm, experiments are conducted on interfaces made with the proposed algorithm in the 3D augmented reality game environment with four virtual objects augmented. The test results show that the proposed algorithm is applicable for possible interfaces for various interactions with virtual objects in the augmented reality environment. This means that by realizing specialized interfaces on various smart phone services, exclusive conveniences are available satisfying critical and independent needs of the user. The proposed algorithm can also be used in user interfaces for optimal UX in AR Game where vision-processing methods are involved.

Keywords*: User interface, face detection, interactions, camera sensor, UX*

1. Introduction

The smart phone is the keyword in IT. Accordingly, at the center of the issue is how to provide natural GUI (graphical user interface) for optimal UX, and to develop the killer content AR. The competitive edge lies in user friendliness deriving from human intuition, intuitive user interfaces allowing individualized UX combining multiple sensor information, and AR contents for optimal UX. For this, the role of the interfaces is getting more and more important: each sensor needs the capability to catch the intentions of the smart phone user, which should be interpreted properly for natural interactions and appropriate response [1-14].

There are two kinds of leading interface technologies for smartphone-'touch-recognition interface' and 'vision-recognition interface'. In 'touch-recognition interface', the user touches the display to input words or to select icons, or to control the device on the screen. Since the user is directly controlling the screen, the experience is intuitive and learning is quick, thus making the decision-making process the fastest. In addition, the device can be equipped with minimal physical buttons replacing conventional key pads or hardware switches. So it can provide the most intuitive interfaces of all interfaces currently being used. The advantages of the touch-recognition interface technology are easy input and access as well as convenience of use. According to manufacturing methods or operating principles, it is classified into 'reduced pressure', 'capacitance', 'optical' and 'ultrasound'. Depending on the way it is used, it can also be divided into 'single touch interface' and 'multi-touch gesture interface' [1-4].

Vision-based interfaces can provide important information since 80% of the

ISSN: 1975-0080 IJMUE

information people get from the outside world comes from visual sources. Other sensors can only recognize one or two information at most, whereas the vision sensor-camera-can get multiple information at one time including color, shape, shade, distance, surroundings and objects, or their movements. Vision-based interface technologies have the greatest potential for development among various applications since a smartphone always comes with a camera device. Currently, most camera interfaces are usually focused on motion recognition of user's hands, torso, or the whole body, image search function using feature-detection of objects, and user friendliness of applications through face detection and recognition. Camera interfaces have shown high achievements on desktops. But in smartphones, with the limited hardware capabilities, the developments have been lagging behind, especially in the area of contents development and research in natural experience interface or augmented reality.

This study proposes a new algorithm, based on which user's face is detected in input images on smartphone camera, the information of which is then used to analyze and infer the locations of eyes and mouth. Next, the location of the nose is identified. The identified information on the eyes, the mouth, and the nose is visually processed to see about the existence of each part's movements to come up with interfaces. To test the validity of the algorithm, 3D AR game environment has been created with 4 virtual objects augmented. Experiments have been performed to test the interfaces based on the proposed algorithm using the detection information of the face, the eyes, the mouth, and the nose along with their movements. To realize natural interactions, each of the 4 three-dimensional objects responding to the eyes, the nose, and the mouth was designed in such a way that their movement frames are different in number, so that their movements can be optimally expressed. In Eclipse environment, Android ver 4.1 and openGL ES were used. Galaxy Tab was used for the experiments.

2. Augment Reality Interface

'Augmented Reality' has been derived from 'Virtual Reality'. AR technology combines computer-processed virtual information onto the real world information. Virtual objects are integrated with real-time image or sound information, or related information is combined to provide augmented information services. AR is different from virtual reality in that it is based on the real world to provide virtual information. So we can provide virtual information at low cost and also make interactions more real-like. The AR realized in mobile environment goes beyond the concept of integrating real-life image information with three-dimensional virtual objects. As various recognition technologies are available on the mobile device, the definition should be changed into providing AR information services through AR interfaces and service mesh-ups [2-3].

AR technology can provide optimal environment for interfaces capable of natural interactions since various forms of augmented information are available through physical and software interface designs. Especially, augmented information made from processed content-based visual factors in real space can serve the purpose of important AR interfaces. Audio or tactile information can also be used as additional resources if processed properly for natural interface [4-5].

3. Detection of Face

'Detection of face' means, "in a given image, we just determine whether there is a human face. When there is a face, we try to decide its location and size". [6] For face detection, a lot of approaches have been developed including 'feature invariant approach', 'template matching method', 'appearance-based method', and 'knowledge-based method'.[4-12] In 'feature-based methods', the face is detected in regard to sizes and shapes of characteristic features of the face, its color, texture, shape, interconnectivity or

combined information.

These methods ensure quick processing time and easy recognition of a face. It has drawbacks: surrounding area or other objects with similar color to the face may be mistakenly identified; color or texture information may be lost on the face depending on light intensity; the face gradient may hinder precise capturing of face features. In other words, these methods have the danger of being overly sensitive to noise factors including lightning, posture, or complex settings.

In 'template-matching methods', standard templates for all faces concerned are made and then compared with the input image to detect the face. These methods are less subject to lightning or surroundings since there is no need for face features to exist in easily detectable forms. So detection is possible even in complex settings. However, they are sensitive to other factors including change in face size depending on distance, rotating angles of the face depending on directions, or gradient. Also, since it is hard to create standard templates encompassing all kinds of faces, making templates takes too much effort. In 'appearance-based methods', face detection is carried out by using the learned model based on learned image collection. These methods include eigenface generated by Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Neural Network (NN), and Support Vector Machine (SVM). These methods are used to detect the face area in a complex image. Learned unique vectors are created by using conventional face area and non-face area based on learned data collection, which is then used to identify the face. High success rates for recognition are ensured since limitations existing in other detecting methods can be overcome with learning. However, it takes a lot of time to learn the database with PCA, NN, or SVM. Also, if the database changes, relearning is required. Since the methods involve large amount of calculations in real-time, real-time detection might not be possible. In 'knowledge-based methods', researcher's knowledge base comprises the basis for face detection including the fact that human face contains eyebrows, eyes, a nose, and a mouth, and that each component keeps a certain distance from one another and positions follow certain rules. These methods apply easily to an image where a face faces front since you can detect the face easily by using the rules for locations and sizes of face components. But it gets harder to detect the face when there are variations in an image in terms of face gradient, angles of the face facing front, facial expressions, and *etc.* This study aims for real-time face detection using input images on smartphone. Therefore, we need simple face detection method which allows quick detection and easy realization. So the conventional methods are not applicable. Especially, to use the information of the movements in eyes, nose and mouth for interface purposes, there is a need for a new algorithm.

4. Proposed Algorithm

When an image is input on the smartphone camera, the process of face detection gets started. Conventional algorithms are not applicable since it requires real-time face detection. Instead, we use the face detection method provided by Android API which ensures fast and high-quality detection and complex realization. With the applied API, we can see the bitmap size of the image, and identify the location and number of faces which exist within the input image by using 'Face Detector' and 'Face Class'. Also available are CONFIDENCE _THRESHOLD constant number information (set at 0.4) and each oiler value for the face on x, y, and z axis. By using this information, we can get the locations for the face and the eyes.

The calculation required is relatively simple and reliable, thus being applicable where quick and reliable conditions are required as in AR environment in which real-time interactions take place [4,9-11] Face is detected first. Based on the information, the conventional method (referring to relational locations, directions, and sizes) [8] and the new method(inferring the nose by analyzing the locations of the eyes and the mouth) are

selectively applied to get the information on the locations of the eyes, the nose, and the mouth. The movement information is also derived from image-processing of information on detected locations and directions. In the smartphone camera used for image output, byte format in preview is in YUV420sp color image, which is transformed into gray image. Image differentials for time are used for t1 frame image and t2 frame image to get the movement information. After that, locations of the eyes, the nose and the mouth are analyzed along with changed data volume to identify the presence of movements in the nose and the mouth, and eye blinking. All of this is integrated to be used as interfaces.

To use the information as interfaces in AR, we build AR 3D game with 4 virtual objects augmented. These virtual objects are divided into enemy and ally characters and attack the game player. The game player defends or attacks by using his/her own eyes, nose and mouth as interfaces. When the virtual objects attack the face of the game player, he/she can attack or destroy the enemy I by moving the eye interface. Or, by moving the eye and nose interfaces, he/she can attack the enemies I and II simultaneously. By using three interfaces(eye, nose, mouth), the ally can attack and destroy three enemy characters. If the interfaces are not used in time, enemy attack may result in explosion.

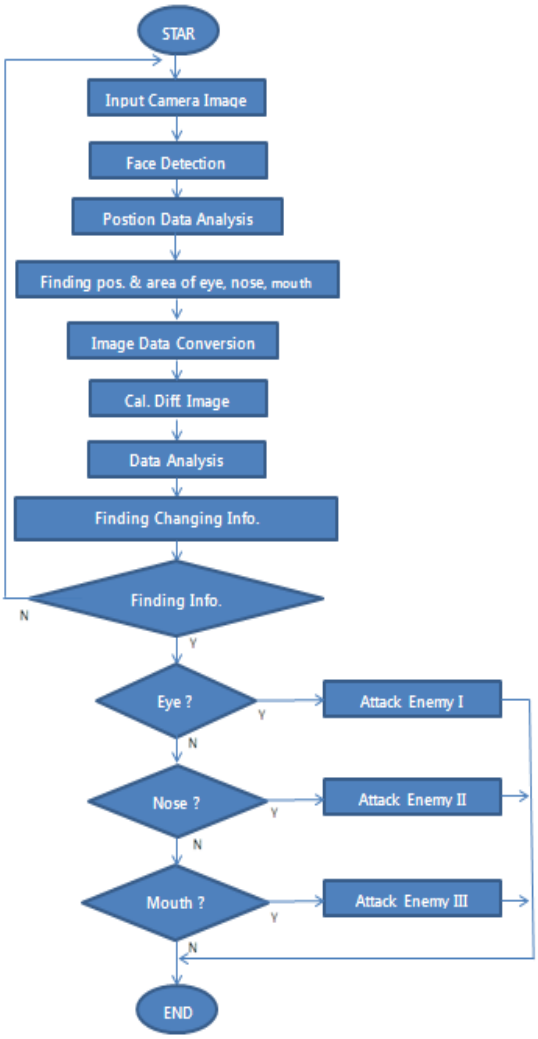
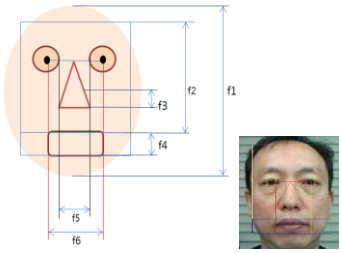


Figure 1. Flow Chart of the Proposed Algorithm

5. Face Interface



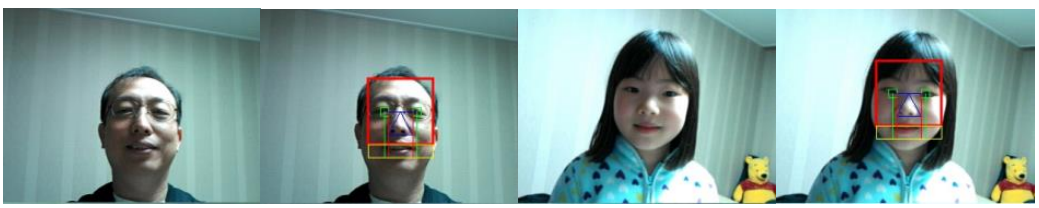
(a) (b)

Figure 2. (a) The Proposed Facial Feature Extraction (b) The Detection Result of Eyes, Nose, and Mouth

Figure (a) shows how we infer the locations of the eyes, the nose, and the mouth by analyzing the face detection information in the input image. f1 in Figure(a) is the total size of the face, and f2, the detected face size, with horizontal and vertical parts showing the detected area. The size is divided by constant number N1, and the resulting size value f2/N1 is used to infer the location of the mouth f4(the vertical size of the mouth).[8] The location of the eyes can be inferred referring to the locations of the face and its size. The inferred location and the size of the eyes are divided by the constant N2 to get f2/N2, which is the size of the eyes. By using the information on eye location [8-11], f5 is determined according to the distance between the eyes. f5 is the size of the area below the nose. Nose is determined by drawing a triangle from the center of both eyes. f3 is where the nostrils are, and shows the location where nose movements are identifiable after the nose is detected. To get it, you divide by N3 the centers of the eyes and the lower part of f2, the detected face area. f6 shows the distance(horizontal) between the centers of both eyes. The size can be determined by considering either the mouth area located outside the eyes or the one inside the eyes. If the area outside the eyes is considered, the size can be bigger than the original mouth. But in detecting the moving mouth, the size can be adjusted depending on each environment. N1, N2, N3 can be determined through experiments by using the face data. In the experiments, N1 was set to 5.0, N2 to 6.0(experiment 1), and N3 to 3.0. Figure (b) shows the resultant image of the eyes, nose, and mouth detected by using the proposed methods.(N2=6.0) It displays that face detection, locations for eyes, mouth, and nose are accurately captured.

6. Experiments

6.1. Experiment 1



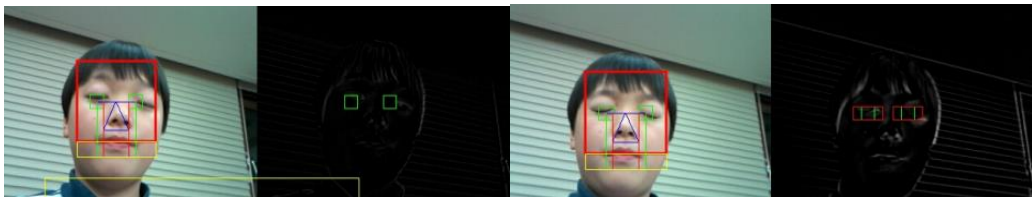
(a) (b) (c) (d)

Figure 3. The Results of Facial Feature Extraction

In experiment 1, face is detected in the image input on smartphone camera. Then the information is used to detect eyes and mouth. And the nose is then detected based on the

proposed algorithm. Figure(a) is the input image. Figure(b) is the resulting detected image of the face, the eyes, the mouth, and the nose. Figure(b) confirms that the nose location was well placed with the proposed algorithm based on the detection information of the face, the eyes and the mouth. In experiment 1, Figure(c) shows the input image of the face. Figure(d) shows the resulting image of detected face, mouth, nose, and eyes. From the input images (a) and (c) and the resulting detected images (c) and (d), it is proved that the proposed algorithm is effective in detecting the location of the nose by using the detection information of the face, the eyes, and the mouth.

6.2. Experiment 2



(a) (b) (c) (d)

Figure 4. The Result of Eye Blinking on Face Detection

In experiment 2, the proposed algorithm is used to detect the face, the eyes, the mouth, and the nose in the input image on smartphone camera. Blinking of the eyes has also been detected. Figure (b) and (d) are resulting gray images, which have been changed gray from the color images on camera. The differential image value has been calculated. The value of Threshold was set to judge the presence of differences. Figure(a) is the image input on t1 frame camera. Figure(b) is the image where eye location information has been detected in gray image. Figure(c) is the color image of t2 frame. Figure(d) is the image where the eye blinking has been judged to exist by calculating differential value. The calculation shows blinking happened and a large square is drawn in red around the eyes. Through experiment 2, we confirmed detecting eye blinking is possible with the proposed algorithm.

6. 3. Experiment 3

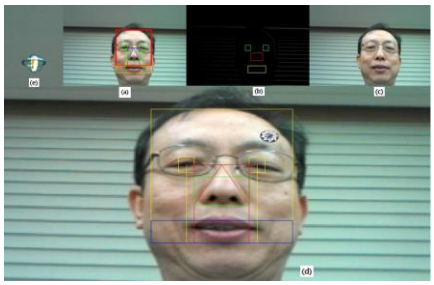


Figure 5

Figure 5. The Facial Feature Extraction and Augmented Objects on Face Detection

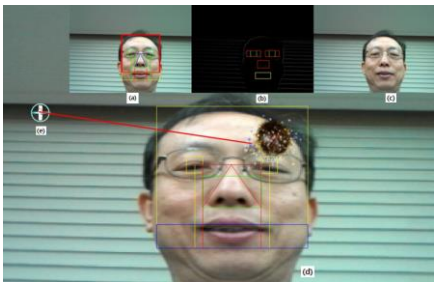


Figure 6

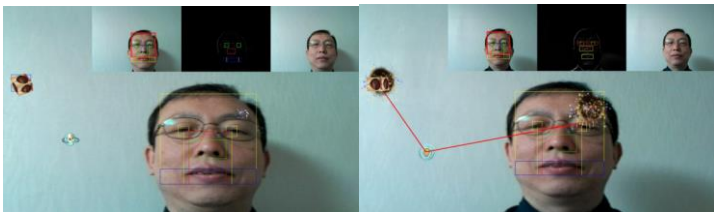
Figure 6. The Eyes Blinking Interface and Interaction in AR Space

In experiment 3, the blinking of the eyes is detected as well as the face, the eyes, the nose, and the mouth in input image on smartphone. All the detection information is used as interfaces for interactions. For this purpose, the AR game environment has been constructed with two virtual objects augmented on the camera input image. The enemy character moves toward the face area to attack the face of the game player. Upon sensing it, the game player blinks the eyes(the interface) to attack the enemy character. Here, the ally character senses the blinking of the eyes of the game player and laser-attacks the enemy character and explodes it.

Figure(c) in Figure 5, is the input image. Figure(a) is the image of face, eyes, nose, and mouth detected. Figure(b) is the image of blinking detected in gray image. Figure(d)(virtual object I) and Figure(e) (virtual object II) are images augmented with virtual flying object. Figure(d) is the case where augmentation has been added to the face of the game player and where attack is needed through blinking of the eyes. The 3D virtual objects created for the purpose have been divided for natural augmentation. The virtual object I was assigned to 7 frame, and the virtual object II, to 10 frame. They have been realized in such a way that real-time attack and defense are possible following the process of chasing objects, area designation, and assumption of directions. Also, for the purpose, the already detected face, eyes, nose, and mouth act as interfaces for interaction.

Figure 6 is the resulting image by using the proposed algorithm. Through identification of the enemy location and the blinking of the eyes, virtual object II is attacking the enemy, virtual object I. Figure(b) in Figure 4 shows the resulting image of the blinking checked for presence by checking variations in information around the eyes in the input image. Figure(d) and (e) are the images where blinking has been detected and the enemy character is attacked and destroyed. The explosion image has been naturally realized by using 10 frame. Every time there is a movement in the position of the enemy virtual object aiming to attack the face area, vibrating sensor is activated to alert the game player of the danger.

6.4. Experiment 4

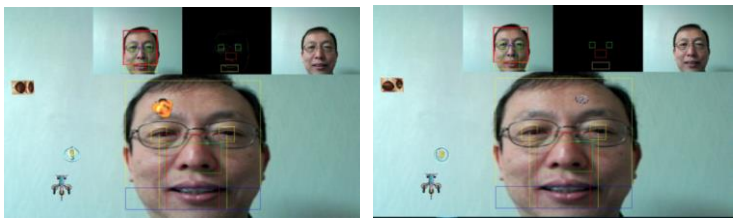


(a) (b)

Figure 7. The Eyes and Nose Interface and Interactions with Augmented Objects in AR Space

Experiment 4 is to test the interfaces. Additional virtual object is augmented, and the attack and defense are realized. The object on the left is the enemy character, three-dimensional virtual object IV, created by openGL ES. In Experiment 4, virtual object I(Figure a) is attacking the face area. So the game player moves the interfaces, the eyes and the nose in this case, to respond to the attack. Here, the proposed algorithm is applied to analyze and infer the movements of each frame for the eyes and the nose, to attack and destroy the enemy character I and III by using the virtual ally character II(Figure b). The Experiment 4 confirms the validity of the proposed algorithm as interface for interaction under AR environment by analyzing and inferring the varied information of images of the detected eyes and nose for each time frame.

6.5. Experiment 5

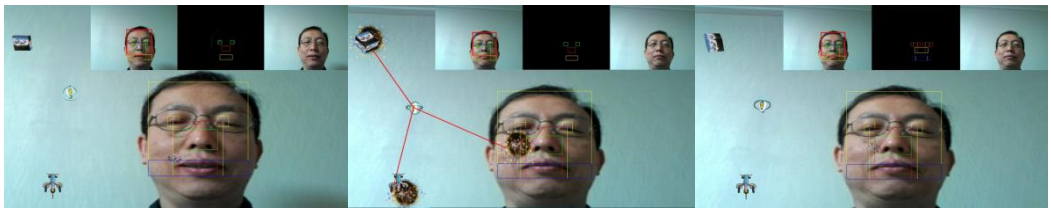


(a) (b)

Figure 8. The Facial Feature Extraction Using No Interface and Enemy Attack

Experiment 5 is to test the interfaces for interactions between the game player and the virtual objects in the environment where four virtual objects are augmented. The virtual object I is attacking the game player as shown in Figure(a). But the game player does not use the interfaces and fails to respond. As a result, in Figure(b), the next frame, the game player's face is attacked and explosion results.

6.6. Experiment 6



(a) (b) (c)

Figure 9. The Eyes, Mouth, and Nose Interface and Interactions with Augmented Objects in AR Space

Experiment 6 is to test the interfaces for movement information of the three interfaces(eyes, nose, and mouth) and for interactions with the virtual objects. In Figure(a), four virtual objects are augmented and the virtual object I moves toward the face to attack the game player. Figure(b) shows that the gamer player is using the interfaces of eyes, nose, and mouth to attack the virtual object I, III, and IV. Figure(c) is the resulting image of the explosion after all three virtual objects have been attacked. With the proposed algorithm, the movements of three interface objects have been identified and attacked. Through Experiment 6, we could confirm that by using three interfaces-eyes, nose, and mouth-we could successfully realize the attacks on all three virtual objects in AR environment.

7. Conclusion

In this study, based on the information gotten from face detection, the locations of the eyes, the nose, and the mouth are identified to be used as interfaces in the proposed algorithm. The validity of the algorithm has been confirmed through experiments. In the proposed algorithm, the color image was changed into gray and the movement information of has been extracted by using different image per time for each image. After that, the location and variation data have been analyzed to know whether there were movements of nose or mouth or eyes blinking. By using the information on eyes, nose and mouth of people, we realize interfaces for natural and intuitive interactions. For the purpose, we built the AR environment where four virtual objects have been augmented, and interfaces have been experimented for various interactions. Through the experiments, we could confirm the efficiency and efficacy of the proposed algorithm in an objective way. With the proposed algorithm, we could effectively detect the eyes, the nose, and the mouth, the interfaces of which successfully operated. To build the AR environment, the experiments have been designed for the four virtual objects to naturally interact in optimal frame. When enemy character attacks, to add fun factor, vibration sensors were triggered to send the alert signal to the game player. Further research is needed about multi-modal user interface which integrates various sensors smartphones are equipped with, and also about AR.

References

[1] D. M. Kim and C. W. Lee, “Trends in Interface Technology for Smartphone Users”, Information Science, **(2010)**, pp 15-26.
[2] J. Jeon, “Standardization Trend in Mobile Augmented Reality”, TTA Journal, 01 - 02 /, vol. 139, **(2012)**, pp. 81-86.
[3] J. H. Jeon and S. Y. Lee, “Standardization for Mobile Augmented Reality Technology”, Analysis of the Trends in Electronic Communications Book, vol. 2, no. 26, **(2011)**, pp. 61-74.
[4] T. Furness and Y. J. Lee*, “Interaction Control Based on Vision for AR Interface of Smart Phone”, International Journal of Smart Home, vol. 7, no. 4, **(2013)**, pp. 349-360.
[5] J. Chun, “Vision-based Motion Control for the Immersive Interaction with a Mobile Augmented Reality Object”, Korean Society for Internet Information, vol. 12, no. 3, **(2011)**, pp. 119-129.
[6] http://www.dmi.re.kr/board/config/list_view.jsp?n_idx=10&bd_idx=45999&view=nex&search_type=&search_data =&currentPage=10
[7] http://www.tta.or.kr/data/weekly_view.jsp?news_id=2271
[8] H. M. Lee and U. Uh, “User Interface Technology and Prospects Based on AR for Organic Interaction”, Information Science, Featured Article, **(2011)**, pp. 15-19.
[9] http://blog.naver.com/PostView.nhn?blogId=budlbaram&logNo=50106543009
[10] http://developer.android.com/reference/android/media/FaceDetector.html
[11] http://www.developer.com/ws/android/programming/face-detection-with-android-apis.html
[12] M. D. Marsico, C. Galdi, M. Nappi and D. Riccio, “FIRME: Face and Iris Recognition for Mobile Engagement”, Journal Image and Vision Computing Date Available online, **(2014)**.
[13] Chen and N. Liu, “Smart Parking by Mobile Crowdsensing”, International Journal of Smart Home, vol. 10, no. 2, **(2016)**, pp. 219-234.
[14] W. M. Liu and J. Li, “Application of Android Mobile Platform in Remote Medical Monitoring System”, International Journal of Smart Home, vol. 9, no. 4, **(2015)**, pp. 163-174.

Author

Young Jae Lee, received the B.S. degree in Electronic Engineering from Chungnam National University in 1984, the M.S. degree in Electronic Engineering from Yonsei University in 1994, and the Ph. D. in Electronic Engineering from Kyung Hee University in 2000. He is presently a Professor in the College of Culture Convrgence at Jeonju University. His research interests include smart media, computer vision, AR (Augmented reality), and computer games.