Contour Shapes and Gesture Recognition by Neural Network

Lee Chin Kho, Sze Song Ngu, Annie Joseph, and Liang Yew Ng

***Abstract*—This paper describes on a real time tracking by using images captured from a closed circuit television (CCTV) before being transmitted to a recognition system for identification of the object's contour shape and gesture. The purposes of this research are to develop a contour shapes and gesture recognition model that can be implemented in an intelligent CCTV target recognition system to discover the possible crime events immediately at the critical areas, while reducing the human power. The crime events that had been focused on were robberies and stealing that commonly happen in shopping malls and ATM machines. Therefore, the contour shape of dangerous weapon and suspected person's gesture had been included in this study. The recognition system was designed using the Image Processing and Neural Network tools of Matrix Laboratory (MATLAB) programming language. The analysis of Sum Square Error and correlation coefficient of the designed network in this study had showed that the recognition system was performing well in recognizing the contour shapes and gesture.**

***Index Terms*—Contour shape, neural network, multilayer perceptron, sum square error (SSE).**

I. Introduction

Nowadays, closed circuit television (CCTV) system becomes commonly used for monitoring and surveillance, especially in commercial areas. To observe wider area, larger amount of camera is required. However, the data of CCTV will not even be processed or looked because it requires intensive labors for monitoring purpose. Therefore, the development of real time tracking systems on the contour shape like dangerous weapons or suspected motions for crime prevention is necessary in order to reduce the crime events that keep increasing nowadays. Some studies on automated surveillance [1], motion detection [2]-[5], and human shape recognition [6]-[10] had been proposed and constructed by other researchers. This study is critical in applying the contour shape recognition system to the human security field.

In this study, the real time tracking system was developed by the pattern recognition program, moving multiple frames into workspace, motion detection and lastly the neural network. Before that, the basic surveillance system was briefly discussed because it was the medium used in this study to capture the images before transmitting to the recognition system to identify the contour shapes of dangerous weapons and suspected person's motions.

The basic surveillance system consisted of four main components, which were cameras, transmission medium, the peripheral and monitor as shown in the Fig. 1.

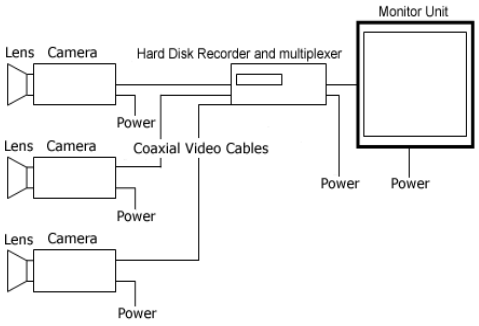

Fig. 1. Basic surveillance system.

The real time tracking pattern recognition program in this paper refered to the automatic surveillance that consisted of specific object detection and motion detection which were used to recognize the dangerous weapons and suspected person's motions. These functions were important to improve the ability of the surveillance software.

II. Design Module

The design module for this study consisted of eight main stages: Motion Detection, Frame Crop to Edge, Frame Resize, Frame Representation in Single Vector, Assemble the Training Data, Define the Network, Train the Network, and Simulate the Network Response to Testing.

The Motion Detection was used to produce a set of frames that consisted of moving objects. These frames were then used to initialize the frame crop to edge procedure. After that, the cropped frames were led to the frame resize process before being converted into single vector. Once the frame became single vector, it would be the training data to initialize the Neural Network, and if it failed to do so, the frame would go back to the initial stage to repeat the image processing stages.

After the image processing stage, the process would proceed to assemble the training data which would then load to the defined network before it could be trained, and simulated the network response to the testing set. If the network was able to recognize the contour shape, the recognition system was successfully established. If not, the neural network stages were repeated with more varieties of training set.

Manuscript received May 27, 2012; revised June 27, 2012.

Lee Chin Kho is with Department of Information Science, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa, 923-12 Japan (e-mail: s1120203@jaist.ac.jp).

Sze Song Ngu and Liang Yew Ng are with the Electronic Engineering Department, Faculty of Engineering, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Malaysia (e-mail: ssngu@feng.unimas.my, ngliangy@feng.unimas.my).

Annie Joseph is with Kobe University, 657-8501 Kobe Shi, Nada-Ku, Rokko dai cho, 1-1, Japan (e-mail:097t805t@stu.kobe-u.ac.jp)

Multilayer Perceptron (MLP) backpropagation neural network was used in this study. This was because MLP backpropagation neural network worked well for pattern matching and this feature was very important in order to create the recognition system. Backpropagation neural network was a feed forward network that used supervised learning to adjust the connection weights [11].

Training the neural network involved processing a set of training data and computing the axis crossover representation for each object. Each frame vector was then given a label of dangerous weapon, not dangerous weapon, suspected person's motion or not suspected person's motion based on what class of object it represented. The general structure of the neural network used to classify the frame vectors was illustrated in Fig. 2.

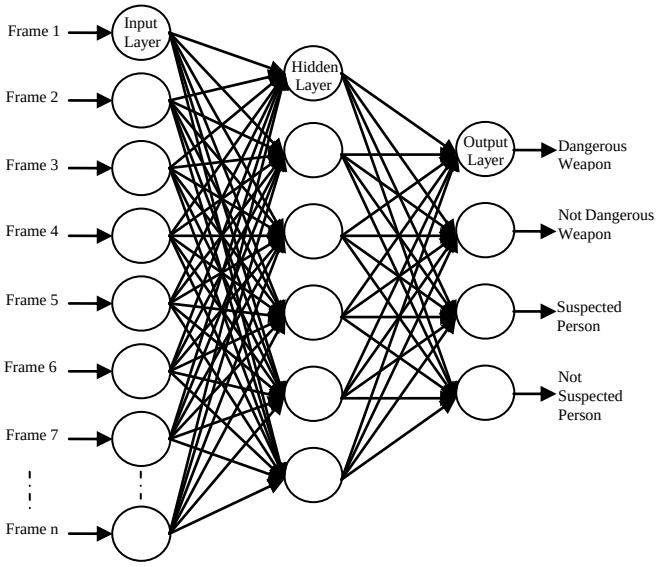

Fig. 2. Feed-forward neural network used to classify the frame crossover vectors consists of a single hidden layer.

Once the network weights and biases had been initialized, the network was ready for training. The network could be trained for function approximation (nonlinear regression), pattern association, or pattern classification. The training process required a set of examples of proper network behavior - network inputs P and target outputs T. The performance function for feed forward networks was Sum Square Error (SSE) - the total squared error between the network outputs and the target outputs T.

III. Results and Discussion

The frames for dangerous weapon recognition system and suspected person motion recognition system were led to the testing set for its network. Thus, this system consisted of 80 frames of testing set, which were 20 frames of dangerous weapon, 20 frames of NOT dangerous weapon, 20 frames of NOT suspected person's motion and 20 frames of suspected person's motion.

The network testing result would be the dangerous weapon, NOT dangerous weapon, NOT suspected person's motion or suspected person's motion. This was because of the four linear output neurons that had been set for the network training of the system. A graph which consisted of the actual result and testing result for the recognition system was plotted and shown in Fig. 3.

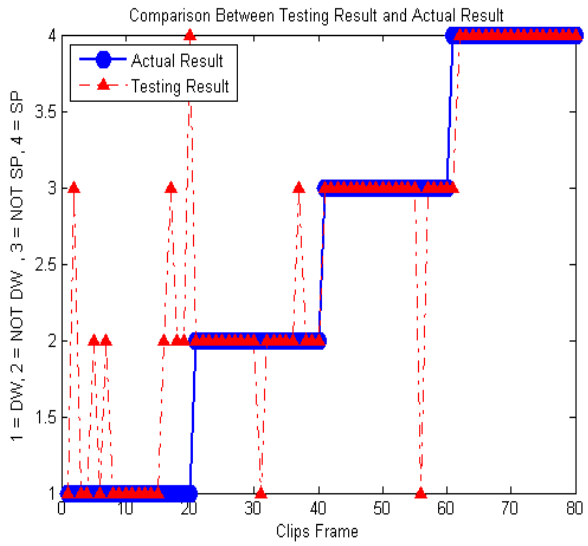

Fig. 3. Comparison between testing and actual result for the recognition system.

Axis-y in Fig. 3 is the linear output neurons where 1 represents dangerous weapon, 2 represents NOT dangerous weapon, 3 represents NOT suspected person's motion and 4 represents suspected person's motion, whereas the axis-x is the frames that lead to the network of the testing set for the system. The blue line with the round nodes represents the actual result for every frame that leads to the network.

There were total 80 frames that had been tested. The first 20 frames had actual results of 1 (dangerous weapon), 21 to 40 frame had actual result of 2 (NOT dangerous weapon), 41 to 60 frame had actual result of 3 (NOT suspected person's motion) while the remaining frames had actual results of 4 (suspected person's motion). The red dashed line with triangle nodes represents the testing result of the network for every frame.

There were some error recognition occurred in the network as shown in Fig. 3. The network recognized fifth, sixteenth, seventeenth, eighteenth and nineteenth frame as 2 (NOT dangerous weapon); second and seventh frame as 3 (NOT suspected person's motion); and twentieth frame as 4 (suspected person's motion), while all frames from 1 to 20 were supposed to be recognized as 1 (dangerous weapon). This was why there were 8 red triangle nodes mismatched with the blue round nodes on line 1 (dangerous weapon) for the first 20 frames.

For the 21 to 40 frame, the network was wrongly recognized for thirty-first and thirty-seventh frame as dangerous weapon and NOT suspected person's motion. For the 41 to 60 frame, the actual result should be NOT suspected person's motion, but the network was wrongly recognized for fifty-sixth frame as dangerous weapon. For the remaining frames with actual result of suspected person's motion, the network was wrongly recognized at sixtieth frame as NOT suspected person's motion. Therefore, total wrong recognition for the network was 12 out of 80 frames.

In order to determine the accuracy of the network, Sum Square Error (SSE) and correlation coefficient (R-value) were used as referred. The SSE was used to measure the network performance function, whereas R-value was the computation between the network response and the target shown in linear regression between the network response and the target.

Fig. 4 illustrates the linear regression for recognition system that corresponds to the testing result. There were eight errors recognition of frame at the first 20 frame or at T = 1, which resulted in the best linear fit for T = 1 around 1.45. On the other hand, there were two errors recognition of frame at the 21 to 40 frame, causing the best linear fit for T = 2 around 2.2. There was one error recognition at the 41 to 60 frame and the best linear fit value for T = 3 was equal to 3. Lastly, there was also one error recognition at the 61 to 80 frame, causing the best linear fit value for T = 4 which was around 3.8.

As shown in Fig. 4, the correlation coefficient for the best linear fit line R-value was 0.852 and from the Figure 3, the sum square error was

$$SSE = (-2)^2 + (-1)^2 + (-1)^2 + (-1)^2 + (-2)^2 + (-1)^2 + (-1)^2 + (-3)^2 + (1)^2 + (-1)^2 + (2)^2 + (1)^2 = 27$$

For each simulation, different values of SSE and R-value were obtained due to the random initial weights for network training [12]. Therefore, in order to get the more accurate value of SSE and R-value for each recognition system, at least ten simulations should be recorded and calculated for the average values.

Table I shows the simulation values of the SSE and R-value for the recognition system. The smallest R-value and the largest SSE value for the recognition system was 0.673 and 63 at 1st simulation. By comparing every couple values of SSE and R-value for each recognition system, it was found that the R-value was inversely proportional to the SSE value.

TABLE I: SIMULATIONS VALUES OF SUM SQUARE ERROR AND CORRELATION COEFFICIENT (R-VALUE) FOR RECOGNITION SYSTEM

Simulation	Recognition Systems	
	SSE	R
1	63	0.673
2	26	0.868
3	47	0.755
4	46	0.762
5	33	0.832
6	58	0.695
7	42	0.793
8	32	0.837
9	27	0.865
10	58	0.698
Average of SSE and R-value	43.2	0.778

When training a network, the number of hidden neurons is critical. If there is too few of hidden neurons, it means that there is not enough available "brain" to learn the problem. Whereas too many, the network "memorizes" instead of "learns" [13]. Therefore, it is important to find out the most suitable number of hidden neuron that can be used in this study.

The different numbers of hidden neuron that had been set for the comparison were 1, 5, 10, 20, 40, 60, 80, 100, 120 and 140.

From Fig. 5, the number of hidden neurons with 1, 120 and 140 had larger value of SSE than the remaining of hidden neurons. This indicated that the system with hidden neuron of 1, 120 and 140 had lower accuracy and they were not suitable to be applied in this system.

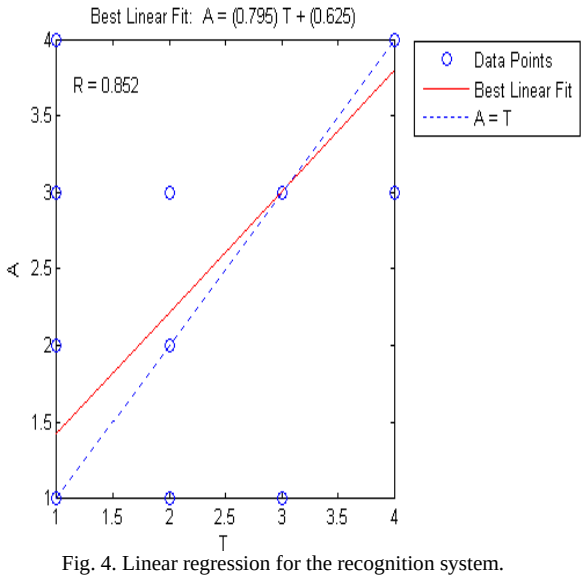

Fig. 4. Linear regression for the recognition system.

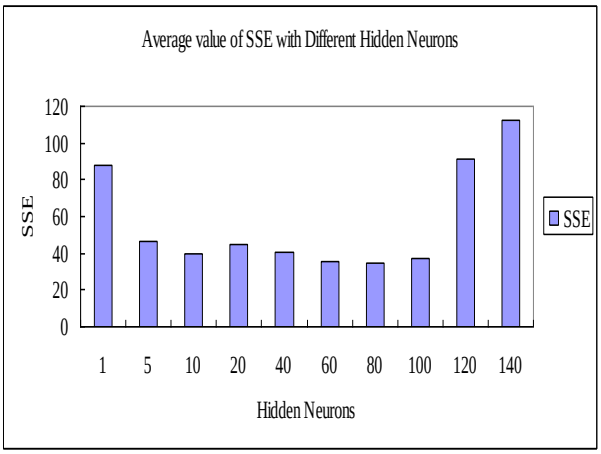

Fig. 5. Average values of Sum Square Error with different hidden neurons for Combine recognition system

Fig. 6 shows that the number of hidden neurons with 1, 120 and 140 had smaller R-value compared to the remaining hidden neurons. It meant that the system with hidden neuron of 1, 120 and 140 had lower accuracy compared to others. When hidden neuron was 1, the network was probably already brain-dead, and would never learn. For the networks with 120 and 140 hidden neurons, the network's predictive powers could only be improved by reducing the number of hidden neurons to the acceptable range. Hidden neurons in the range of 5 to 100 are suitable to be applied in this system. However, the best number of hidden neuron that could be set was 80 because it had the highest average R-value and the lowest average value of Sum Square Error compared to others.

Besides, the performance of the algorithm in this study was very sensitive to the proper setting of the learning rate. If the learning rate was set too high, the algorithm might oscillate and became unstable. If the learning rate was too small, the algorithm would take too long to converge [11]. Therefore, the comparison between different learning rates was done on the system. The average values of 10 simulations for both SSE and R-value with different hidden neurons had been calculated and recorded. The different hidden neurons that had been set for the comparison were 0.1, 0.09, 0.08, 0.07, 0.06, 0.05, 0.04, 0.03, 0.02 and 0.01.

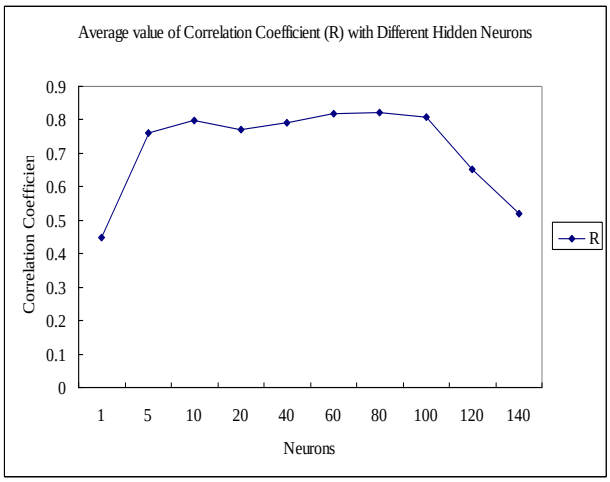

Fig. 6. Average values of correlation coefficient (R) with different hidden neurons for combine recognition system.

Fig. 7 shows the average values of SSE with different learning rate for the recognition system. The learning rate of 0.04, 0.03, 0.02 and 0.01 had smaller value and it meant that the system had higher accuracy compared to others. In other words, the learning rate of 0.1, 0.09, 0.08, 0.07 0.06 and 0.05 were not suitable to be applied in this system.

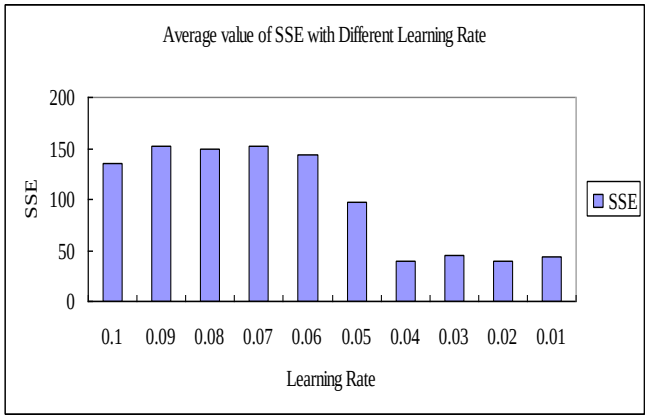

Fig. 7. Average values of sum square error with different learning rate for the recognition system.

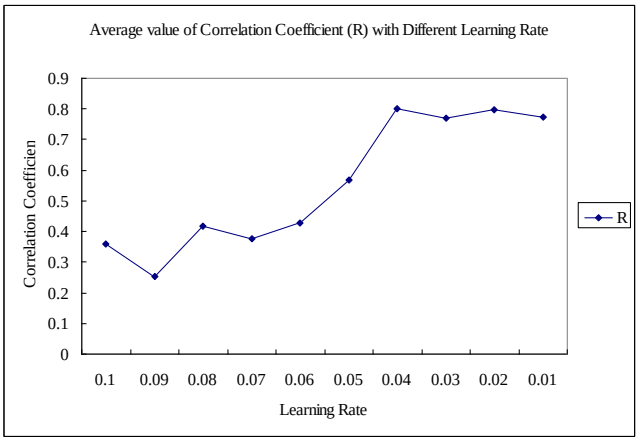

Fig. 8. Average values of correlation coefficient (R) with different learning rate for the recognition system.

From Fig. 8, the learning rate of 0.1, 0.09, 0.08, 0.07 0.06 and 0.05 had smaller R-value compared to the remaining learning rate. These high learning rates would cause the algorithm to be oscillated and become unstable. Thus, the system with learning rate of 0.1, 0.09, 0.08, 0.07, 0.06 and 0.05 were not suggested to be used in this system.

Therefore, the system was accepting the range of learning rate between 0.04 to 0.01. However, the best learning rate for this system was 0.04 because it had the highest average R-value and the lowest average value of Sum Square Error compared to others.

IV. Conclusion

This study was implemented utilizing basic MATLAB programming which was capable of combining image processing and neural network techniques to create a contour shape recognition system. From the results, the system had been proved that it was performing well in recognizing the dangerous weapon and suspected person's motion. By analyzing the values of Sum Square Error and Correlation Coefficient (R-value), the accuracy of the recognition system could be verified.

Most of the major features of the system had been successfully accomplished and all the requirements had been fulfilled, but there were some limitations due to certain constraint occurred. The limitations were that the system would take longer time to operate if the number of training set was too large and there was higher resolution of the frame in the training set.

Acknowledgment

The author would like to thank Universiti Malaysia Sarawak (UNIMAS) for providing the funding to publish and present this paper.

References

[1] R. T. Collins, A. J. Lipton, and T. Kanade, "A system for video surveillance and monitoring," in *Proc. 8th International Topical Meeting on Robotics and Remote Systems*, USA, 1999, pp. 1–15.

[2] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 781– 796, August 2000.

[3] Y. Guo, G. Xu, and S. Tsuji, "Understanding human motion patterns," in *Proc. 12th IAPR International Conference on Pattern Recognition*, Jerusalem, vol. 2 , 1994, pp. 325-330

[4] J. Russell, "Detecting Humans in Video Footage using Multiple Classifiers," Honours dissertation, School of Comp. Sci. and Software Eng., Western Australia Uni., 2004.

[5] L. Wang, W. Hu, and T. Tan. (May 2002). Recent developments in human motion analysis. *The Journal of the Pattern Recognition Society*. [Online]. 36. pp. 585–601. Available: http://vc.cs.nthu.edu.tw/home/paper/codfiles/pcchu/200404211710/recent_developments_in_human_motion_analysis.pdf

[6] K. Tabb, S. George, R. Adams, and N. Davey, "Human shape recognition from snakes using neural networks," in *Proc. 3rd International Conference on Computational Intelligence and Multimedia Applications*, USA, 1999, pp. 292–296.

[7] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, New York, NY: J. Wiley and Sons, 2001.

[8] C. A. Nicolaou, A. L. Egbert, R. C. Lacher, and S. I. Bassett, "Human shape recognition using the method of moments and artificial neural networks," in *Proc.IJCNN'99 International Joint Conference on Neural Networks*, Washington, vol. 5, 1999, pp. 3147–3151.

[9] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, USA, vol. 2, 2000, pp. 142–149.

[10] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no 24, pp. 509-522, April 2002

[11] The MathWorks. Neural Network Toolbox 6.0. (January 2008). [Online]. Available: http://www.mathworks.com/products/neuralnet/.
[12] A. Pavelka and A. Proch´azka, "Algorithms for Initialization of Neural Network Weights," *Sbornik prispevku 11. Konference MATLAB 2004*, vol. 2, 2004, pp. 453-459,
[13] VerDuin, "Solving Manufacturing Problems with Neural Networks," in Article Automation (Cleveland, Ohio: 1987), July 1990, pp. 54-58.

Lee Chin Kho received the B.Eng (Hons) Electronics Engineering from Multimedia University in 2003 and Master of Electrical Engineering from Adelaide University in 2004. Now, she is further her PhD. study in Japan Advance Institute of Science and Technology (JAIST). In 2003, she becomes Process Integration Engineer in 1st Silicon Sdn Bhd for six months. Since 2005, she worked as lecturer in University Malaysia Sarawak. In 2005, she obtain a grant on FRGS from UNIMAS for two years on the research of Signal Penetration Into Building Materials and another two grant from UNIMAS in 2010 and 2011 on the Microstrip Antenna Design and Motion Detection by Neural Network research respectively. She is the member of Board of Engineer in Malaysia (BEM) and graduate member of Institute of Engineering Malaysia (IEM).

Sze Song Ngu received the B.Eng. (Hons) degree in Electronics Engineering from Multimedia University, Cyberjaya (2003) and M.Eng degree in Electrical Engineering from the University of Adelaide (2004). He is working as a lecturer in the Department of Electronic Engineering at University Malaysia Sarawak (UNIMAS), Malaysia. He is currently a PhD. Student with the School of Engineering at the University Glasgoww. His research interests include electrical machines and drive, power electronics, control system and renewable energy.

Annie Joseph received the BEng and MSc degrees in Electrical and Electronic Engineering and Mathematics from Colleague University Tun Hussein Onn in 2005, and University Science Malaysia, in 2006 respectively. She is currently working towards the PhD degree in Electrical and Electronic Engineering at the Kobe University, Japan. Her research interest is online learning, neural network, concept drift, feature extraction and machine learning. She is a member of board of Engineer of Malaysia (BEM). She is also a student member of the IEEE.