ⓒ 2007 KSII TIIS

RFID Tag Detection on a Water Content Using a Back-propagation Learning Machine

Minho Jo[1], *Member, KSII*, Chang-Gyoon Lim[2], *Member, KSII* and Emory W. Zimmers[3], *Non-member*

[1]School of Information and Communication Engineering, Sungkyunkwan University, 300 Cheoncheon-Dong, Jangan-Gu, Suwon, Gyunggi-Do 440-746, South Korea [e-mail: minhojo@skku.edu]

[2]Department of Computer Engineering, College of Engineering Science, Chonnam National University, 96-1 San, Dunduck-Dong, Yeosoo, Chonnam 550-749, South Korea [e-mail: cglim@jnu.ac.kr]

[3]Department of Industrial and Systems Engineering, Lehigh University, 27 Memorial Drive West, Bethlehem, PA 18015, USA [e-mail: ewz0@lehigh.edu]

*Corresponding author: Minho Jo

Received September 27, 2007; revised November 15, 2007; accepted November 22, 2007; published December 25, 2007

Abstract

RFID tag is detected by an RFID antenna and information is read from the tag detected, by an RFID reader. RFID tag detection by an RFID reader is very important at the deployment stage. Tag detection is influenced by factors such as tag direction on a target object, speed of a conveyer moving the object, and the contents of an object. The water content of the object absorbs radio waves at high frequencies, typically approximately 900 MHz, resulting in unstable tag signal power. Currently, finding the best conditions for factors influencing the tag detection requires very time consuming work at deployment. Thus, a quick and simple RFID tag detection scheme is needed to improve the current time consuming trial-and-error experimental method. This paper proposes a back-propagation learning-based RFID tag detection prediction scheme, which is intelligent and has the advantages of ease of use and time/cost savings. The results of simulation with the proposed scheme demonstrate a high prediction accuracy for tag detection on a water content, which is comparable with the current method in terms of time/cost savings.

***Keywords*:** RFID, Water Object, Tag Detection, Wireless Communications, Back-propagation Learning

1. Introduction

RFID (Radio Frequency IDentification) is based on radio communication for tagging and identifying an object. Using a special antenna device called an RFID reader, RFID technology enables objects to be labeled and tracked as they move from place to place. A typical RFID system consists of tags, readers, middleware, application program, and server [1]. The application program typically handles a specific task such as tracking the inventory in a warehouse, checking vehicles/human beings, or reordering the items removed from the shelf in a retail store, based on the inventory data. It also takes an appropriate action based on the data extracted from the tag of the target item; for example retail products, pallets, cartons, shipments, animals, human beings, or vehicles. The middleware is a bridge interfacing the hardware components from the lower layer with the higher application program layer. In some literature the combination of the application program and middleware is called middleware. An RFID system is based on wireless communications between an RFID reader and a tag. An RFID tag is a small radio frequency chip coupled with a microprocessor, which can communicate by wireless with the RFID reader (sometimes called antenna). The RFID reader is a powered RF device communicating with the tags on the wireless side and one or more computers on the other side of the wired infrastructure.

In supply chain management and factory automation, for example, it is possible to track the vehicles (or conveyer) loaded with goods by attaching a tag to them. Information about the vehicle and goods, such as the vehicle number, the contents of goods and time-stamped departure and location data, can be recorded on the tag, and then read from it at a specific time. Wireless radio communications between the RFID tag and RFID antenna are very sensitive to factors such as the reader type, tag position, and direction of the tag, material of the object, angle of the antenna, and speed of the object [2][3][4]. The angle of the antenna, speed of the tag (or object), and position of the tag are the most important factors influencing successful RFID tag detection (sometimes simply called “RFID tag detection” or “tag detection” in this paper). Successful tag detection (also known as “recognition”) enables the RFID reader to receive a unique ID and other data from the detected tag, and then this information is transmitted to a DB server.

Successful communications between the reader and the tag are called “RFID tag detection” by the reader. Recognizing the tag using the reader requires tracking successfully the tag object to which the tag is attached. The RFID tag detection is directly related to the tag signal strength detected by the RFID reader. The tag signal strength is influenced by the RFID tag detection factors mentioned previously. The more difficult problem with the deployment of the RFID system in practice, is the failure in reading the tags, i.e., in tag detection. The solution to the tag detection failure problem to date has been based on the hardware of the RFID system [5], while very little has been achieved analytically, using software for maximizing the readability. The performance of RFID tag recognition (detection) with respect to the different contents of water bottles, paper aluminum foil, and tin cans was studied by Park et. al [6]. They analyzed tag recognition rates of boxes on a pallet by antenna type and tag position and determined the best influence factor conditions for tag detection.

Finding the appropriate position, direction, speed of RFID tag, which is called the best influence factor condition search of the RFID tag detection, in addition to the angle of the RFID antenna, is not easy but requires tedious trial-and-error steps. This work is intended to find the best conditions for the influence factors enabling the maximum tag detection rate (or called “detectability”). To date, this has been achieved by the trial-and-error experimental method. Thus, we are motivated to introduce an intelligent RFID tag detection method which saves time and is simple.

To date, little research on the best influence factor condition search has been conducted. This is a crucial issue for the RFID system at deployment. However, as more RFID systems have been successfully implemented in practice, the problem has become much more important. In this paper, firstly, the effect of variations in the influence factors is systematically analyzed through actual and trial-and-error experimentation. Then, we also propose a new scheme for the best influence factor condition search. The proposed scheme is an intelligent back-propagation learning-based RFID tag detectability prediction method. The end result is that we compare the tag detection results by the current method and proposed method, respectively. The key work of the proposed method is in predicting the tag detectability without the trial-and-error steps. We first train the back-propagation learning-based tag detectability prediction model using the historical data obtained by the trial-and-error experimental method. Thus, the back-propagation learning model predicts the RFID detection. Thus, the predicted tag detectability results enable us to obtain the best influence factor conditions without the time consuming trial-and-error experimental method. The accuracy of tag detectability prediction for the proposed methods is verified by comparing the real data with the predicted data. The accuracy of RFID detection prediction is high. The performance results demonstrate that the tag detectability prediction accuracy is as great as 92%.

The rest of the paper is organized as follows: The background, including the structure of RFID systems and test-bed environments applied to this study, is introduced in Section 2. The proposed schemes, such as the back-propagation learning-based tag detectability prediction models, are discussed in Section 3. Section 4 presents the performance evaluation, with results obtained from both the experimental scheme and the proposed intelligent back-propagation learning-based tag detectability prediction scheme. We conclude the paper with remarks on future work in Section 5.

2. The Background for RFID Systems

2.1 Structure for RFID Systems

In a typical RFID system, passive tags are attached to objects such as goods, vehicles, humans, animals, and shipments, while a vertical/circular polarization antenna is connected to the RFID reader. The RFID reader and tag can communicate with each other via wireless radio, using a number of different frequencies. Currently, most RFID systems use an unlicensed spectrum. The common frequencies used are low frequency (125 KHz), high frequency (13.56 MHz), ultra high frequency (860-960 MHz), and microwave frequency (2.4 GHz). The typical RFID readers are able to read (or detect) the tags of only a single frequency but multi-mode readers which are capable of reading the tags of different frequencies are becoming cheaper and more popular [7].

The RFID system used in the experiment is an Alien ARL-9800. The technical specifications of reader, antenna, and tag are the following:

- Reader: 910 MHz~914 MHz frequencies. AC/DC power converter and 45 Watts maximum (120 or 240 VAC). 50 hopping channels and 500 KHz channel spacing.
- Antenna: multi-static topology, circular polarization and reverse polarity TNC.
- Tag: ISO 18000-6 Type C Gen 2 Class 1 Tag.

2.2 The Testbed

The factors influencing communications between the RFID tag and RFID reader include: (1) contents of object; (2) type, position, and direction of tag; (3) speed of tag; (4) angle of antenna; (5) power, type, gain, frequency range and number of antenna; (6) work environment of the RFID system, etc. The position and direction of the tag, the speed of the tag, and the angle of the antenna can be controlled with an application program, such as the proposed intelligent SVM tag detectability prediction model, whereas other factors are hardware-controlled. We consider the position and direction of tag and the speed of tag as influence factors, because other factors are fixed by the hardware.

We carry out the experiment in a testbed to obtain the real historical tag recognition data by time consuming trial-and-error steps, called an experimental scheme. We discover that a tag can be detected or not (a binary decision) under specified conditions for influence factors in the experiment. This experiment is conducted on the RFID tags attached to a carton box containing water content, on a moving conveyer, as illustrated in **Fig. 1**.

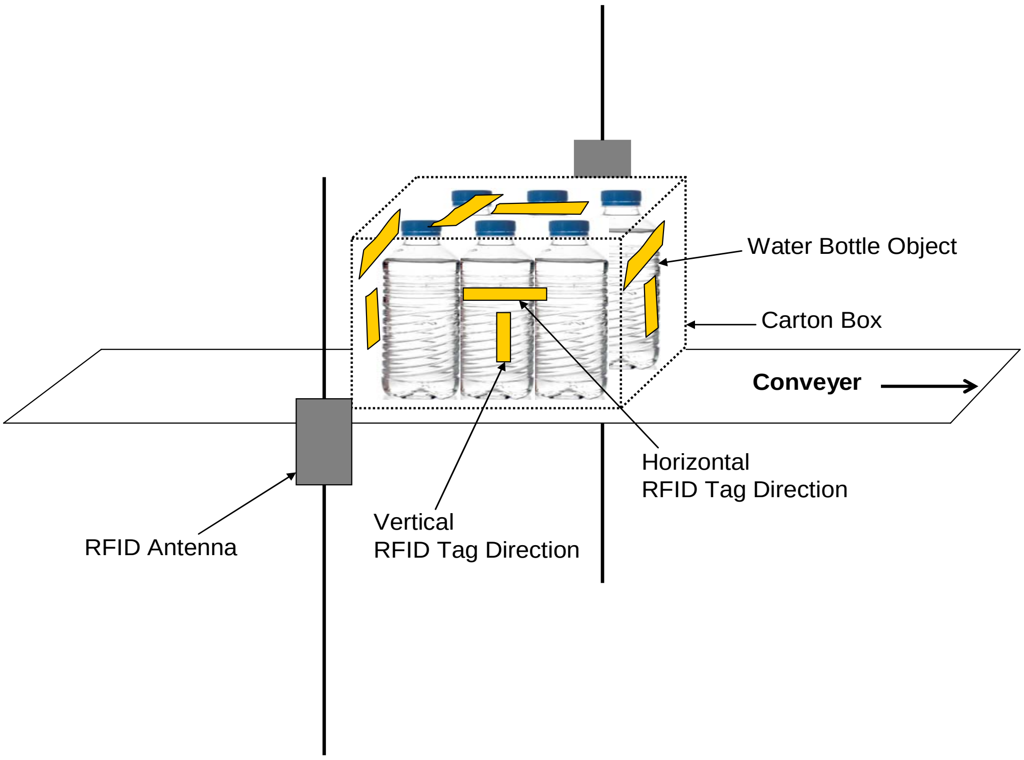


Fig. 1. A testbed environment to obtain trial-and-error experimental RFID tag detection data.

A tag is attached to 5 positions on the water bottle box: left, right, top, front and back side. A carton box which is 30 *cm* wide, 20 *cm* deep and 32 *cm* high, contains 6 2-liter pet bottles of water. There are two antennas, each a meter located away from both sides of the conveyer as shown in **Fig. 1**.

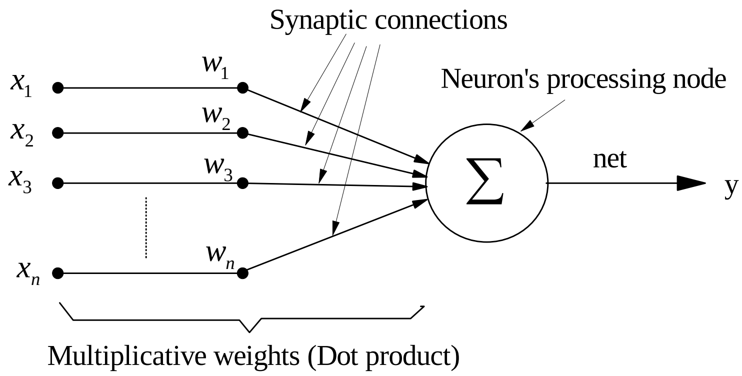
We maintain 15 dB power attenuation to obtain the experimental RFID tag detection data under a more critical condition for sensing. The tag is attached on each side of the box in two different directions, horizontal and vertical. The experimental results demonstrate different recognition success rates with respect to the tag direction. The tag recognition test is applied for different speeds of the conveyer, 1.647 m/sec, 2.219 m/sec, 3.359 m/sec and 3.636 m/sec. A total of 20 test iterations for each speed are executed at different tag positions. A total of 800 examples are obtained as historical experimental data. The experimental results of tag detection for four different speeds of the conveyer and in two different tag directions are provided in Section 4.

3. Proposed Schemes

3.1 Backpropagation-learing machine

In this paper, a back-propagation learning-based scheme is applied, to predict the RFID tag. A back-propagation learning-based scheme is a model of artificial neural networks. Artificial neural networks are composed of highly parallel building blocks that are interconnected to form highly complex systems, by mimicking the fault-tolerance and capability to learn of a biological neural system [8]. These models have layered architectures with uniform simple nonlinear processing elements (neurons), which receive input from other elements. Every neural model consists of processing elements having synaptic input connections (weights) and a single output. Several learning algorithms enable it to self-adapt by changing the parameters.

General neural network model: A general neural model is shown in **Fig. 2**. The model shows a set of weights and the neuron's processing unit, which computes the function f of the weighted sum of its inputs.



$$\text{net} = \sum_{i=1}^{n} w_i x_i$$

$$y = f(\text{net}) = \frac{1}{1 + e^{-\lambda(\text{net})}}$$

Fig. 2. General neural model

The weight matrix W is usually initialized with random values. The net is a weighted sum called the net input into a unit. The function is the unit activation function, which yields the unit's output for its net input. An input vector is chosen at random from training examples, that are a series of cases with a range of input and output values. The network memorizes the inputs by altering its weight matrix. Thus, the training consists of using the training examples to tune the appropriate weight values between layers. Training examples are presented at the input layer. The vector is then propagated throughout the network. Selected learning algorithms are applied to determine weight adaptations. The network can be trained at each point or after each epoch. An epoch represents one complete iteration through the training data set. This learning procedure should be repeated until an acceptable error rate is achieved or until a certain number of iterations are completed using training examples. Neural networks have been applied to many fields, including automatic control, pattern recognition, adaptive signal processing, function approximation, etc [9][10][11][12].

Supervised back-propagation learning: Neural networks attempt to solve problems through learning from training examples (supervised learning) and have been applied in several areas such as pattern recognition and function approximation. Once the neural network model is trained, the trained model can predict or classify patterns. In this paper, we use the back-propagation (BP) algorithm [13] with the gradient descent method to change the weights, as a way of detecting the tag on a water content. Although this method has the drawbacks of a local minimum and a long training time, it is commonly applied to the pattern recognition area. RFID tag detection is classified as pattern recognition. Thus we use the BP model for our prediction of RFID tag detectability.

The back-propagation training algorithm is one of most successful proven pattern recognition techniques to train networks. It is evolved from Madaline structures by including nonlinear differentiable activation functions for the output layer of each neuron, and it uses multiple adaptable layers. Back-propagation is different from the Madaline in that the weights of all layers are adjustable. It can be demonstrated that a back-propagation network is able to approximate as accurately as desired, any mapping between any vector of dimension *m* and any other vector of dimension *n* [14].

The neurons used in our back-propagation network will be McCulloch-Pitts neurons modeled by a sigmoid activation function. The fixed asymptotic values are generally +1 and 0, or +1 and -1. Another example of a bipolar sigmoid activation function can be represented as

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

where $\lambda>0$ in Eq. (1), which is proportional to the neuron gain determining the gradient of the continuous function *f(net)* near *net*=0, and where net is a scalar product of the weight and input vector for the neuron. As $\lambda \rightarrow \infty$, the limit of the continuous function is the *sgn*(*net*) function, the bipolar function. **Fig. 3** shows the bipolar continuous sigmoid activation function graph of Eq. 1 for λ=1.

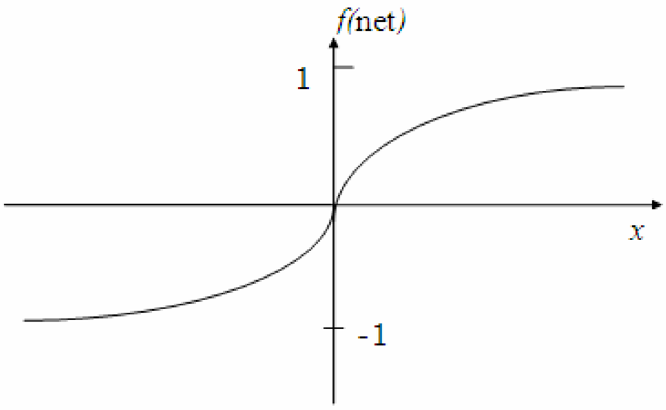


Fig. 3. Bipolar continuous activation function

Generally, the basic form of the sigmoid activation functions is continuous and monotonically increasing as shown in the figure. Back-propagation has a hierarchical network architecture, which consists of an input layer, an output layer, and at least one hidden layer between the input and output layer. The number of neurons in the input layer and the output layer is fixed by the system before training. The number of neurons in the hidden layer can be determined by the network designer for better performance. There are no interconnections between neurons in the same layer in Back-propagation. However, each neuron in an input layer provides its output to each and every neuron in the first hidden layer, and each neuron in the last hidden layer passes its output to each and every neuron in the output layer. Each neuron in the input layer has a single input and a single output, and only passes the value of its input to its output. Initially, weight matrices between the layers are initialized randomly.

Back-propagation networks use a gradient search algorithm to minimize an objective function which is equal to the mean square errors between the desired and the actual output. The networks take the input samples from the input layer, propagate them to the output layer, and generate the network output, as illustrated in **Fig. 4**. Present error signal terms are propagated backward for tuning each layer's weights.

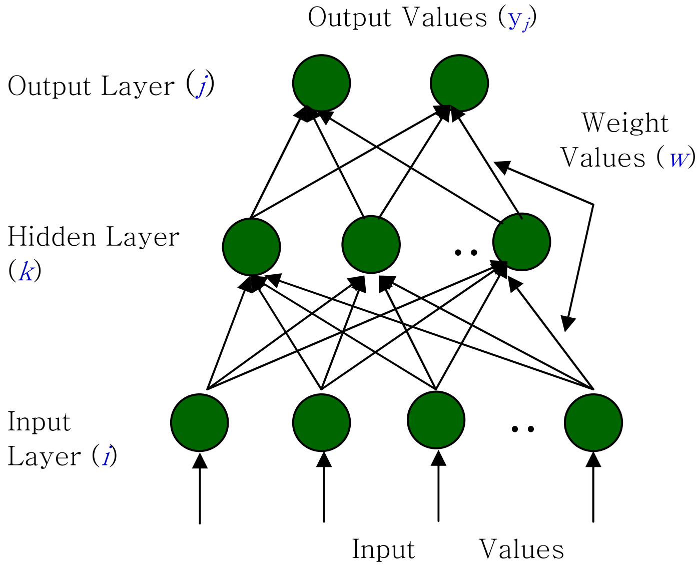


Fig. 4. The general structure of back-propagation

The threshold value for a decision in the verification phase should be determined during the training stage.

Fig. 4 shows the structure of back-propagation with one input layer, one hidden layer, and one output layer. The learning algorithm of back-propagation can be considered to be a generalization of the LMS algorithm. It uses a gradient search scheme to minimize an objective function equal to the square of the difference between the desired and the actual network output. The networks are trained by initially picking small random weight values between -1 and 1, selecting internal thresholds and then presenting all training data repeatedly. Weights are either revised instantaneously after each training vector is presented or only after all training data vectors have been presented. This process is repeated until weights converge and the objective function is reduced to an acceptable value.

We construct a cost function E (the total error of the system) that measures how the network has learned. The vector *x* represents an input pattern into the network, and the vector *d* the corresponding target. We change the weights of the system in proportion to the derivative of the error with respect to the weights. The output functions are differentiable. The rule for changing the weights for input/output pair (*x*, *d*) is given by the gradient descent method. We minimize the quadratic error function using the following iteration process

$$w_{ji} = w_{ji} - \eta \frac{\partial E_k}{\partial w_{ji}} \tag{2}$$

where $\eta > 0$ is the learning constant. A generalized delta learning rule with a uni-polar activation function and P training pairs is applied in this paper by applying the following algorithm:

Table 1. Generalized delta learning algorithm.

1. Select $\eta > 0$ and $E_{max} > 0$.

2. Initialize the weights V and W with small random values.
Set r = 1 and E (Error) = 0.

3. Compute the output with $x = x^r$ and the desired output $d = d^r$.

$$y = \frac{1}{1 + \exp\left(-W^T z\right)}$$

, where z_k is the output vector of the hidden layer.

$$z_k = \frac{1}{1 + \exp\left(-V_k^T x\right)}$$

4. Update the weight matrix for the output units

$$W = W + \eta \delta z$$

, where $\delta = (d - y)y(1 - y)$.

5. Update the weight matrix for the hidden units

$$V_k = V_k + \eta \delta W_k z_k (1 - z_k) x, \quad k = 1,2,3\ldots K.$$

6. Compute the cumulative cycle error by adding the present error to E

$$E = E + \frac{1}{2}(d - y)^2$$

7. If r < R then r = r + 1 and continue the training by returning to step 3
Else go to step 8.

8. Check the stop criterion.
If $E < E_{max}$ then STOP
Else If $E > E_{max}$ then go to step 3 by setting E = 0 and initializing a new training cycle

3.2 The proposed model

We build 2 nodes of the input layer, one node which inputs the position of the tag and the other node which inputs the speed of the conveyor. The output layer is a single node which indicates whether the tag has been detected or not, i.e., a binary decision. The tag has a total of 10 different positions, as already shown in **Fig 1**. The output node will predict whether the tag is detected or not, based on classification of the input pattern. The accuracy of the training model will be verified by comparing the output result of prediction to the real experimental result. The number of nodes in the hidden layer will be determined by the trial and error method. A total of 800 training examples which were obtained from the trial-and-error experimental scheme, are used to train the back-propagation learning model for predicting the RFID tag detectability. A total of 180 test data are used for verifying the prediction of the tag at existing speeds. A total of 120 test data are used for the case of prediction at a new speed, 2.795 m/sec. These will be discussed in the next section.

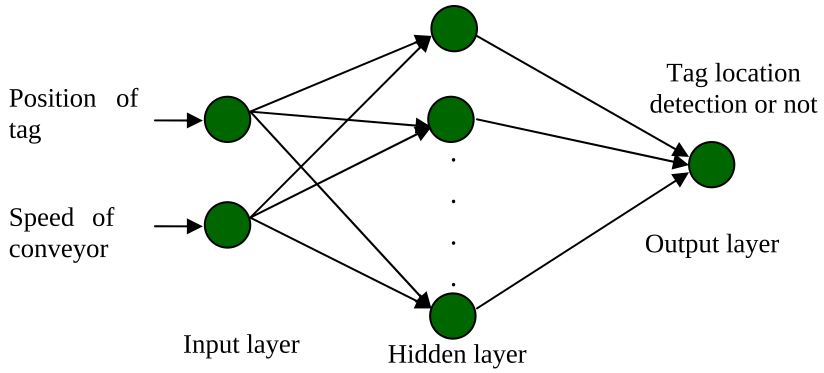


Fig. 5. Proposed back-propagation learning-based prediction model.

4. Performance Evaluation

4.1 Trial-and-Error Experimental Approach

We classify the tags attached on the box carried on the conveyer, into the horizontal and vertical directions, based on the position of the tag. As shown in **Fig. 1**, each side of the box has two tags, the horizontal tag and the vertical tag. There are 6 positions of the tag. In **Fig. 6**, the success rate of tag detection in the case of the horizontal tag direction is significantly higher than for the vertical. Note that the horizontal direction is the same as the direction of movement of the conveyer belt. Also, note that there is little difference in tag recognition rate with respect to speed of the conveyer. Because the RFID tag signal is not sensitive at low speeds, such as 1.647 m/sec, 2.219 m/sec, 3.359 m/sec and 3.636 m/sec, the speeds here do not influence the tag recognition performance.

4.2 Intelligent Learning-based Prediction Approach

The back-propagation learning-based model is trained by the historical tag detectability data obtained by the trial-and-error experimental scheme. Then, the accuracy of the RFID tag detectability prediction model is verified based on comparison between the result of the prediction scheme and the trial-and-error experimental method. When the prediction capability of the model is verified, we don’t consider the horizontal or vertical direction of the tag to be classified. However, because our purpose here is to find the best prediction model, we consider model variables, such as the number of hidden layers, and input variables, such as new speed and existing speed. Accuracy is verified for new speed and existing speed by using different numbers of hidden layers. We consider the number of hidden layers as a back-propagation modeling decision parameter. The new speed is 2.791

m/sec, a new influence factor condition which has never been used for training the back-propagation learning-based model, whereas the existing speeds, 1.647 m/sec, 2.219 m/sec, 3.359 m/sec and 3.636 m/sec, are the influence factor conditions used for training. However, the test data samples of the existing speed have never used for training the model. Thus, we can see the accuracy of the RFID tag detectability prediction for both the new speed and existing speed. Because the new speed condition has never been considered as an influence factor condition, it is better suited to verify the prediction capability than the existing speed conditions.

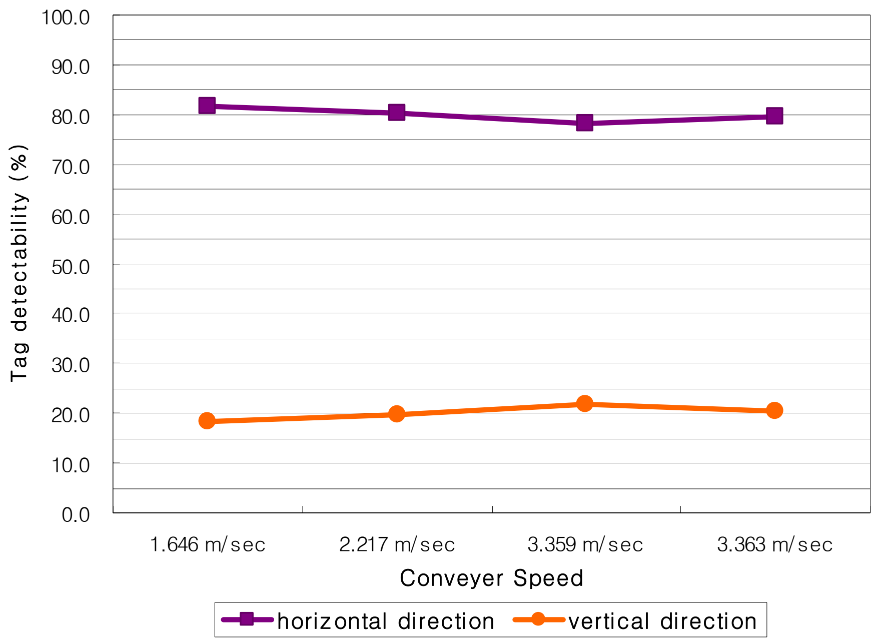


Fig. 6. Tag detectability of two different tag directions.

We repeat the search procedure for each hidden layer, to achieve the best prediction accuracy for the new speed and existing speed, respectively. 20 iterations for each hidden layer are made with the prediction model, to determine the best prediction accuracy. For an example of the existing speed, the results of 9 hidden layers are provided in **Fig. 7**. The maximum prediction accuracy is 96.32%. The minimum is 85.3%. The result of 20 iterations yields an average of 92%.

For the new speed, the iterated search procedure for each hidden layer is executed in the same way. An example of 20-iterations with 3 hidden layers is given in **Fig. 8**. The maximum prediction accuracy is 83.3%. The minimum is 70.0%. The result of 20 iterations yields an average of 79.5%. The iterated search procedure is executed for different hidden layers. The number of hidden layers used for testing is between 2 and 10. The average prediction accuracy of each hidden layer is shown in **Fig. 9**.

As illustrated in **Fig. 9**, for the new speed the prediction accuracy is less for 4 hidden layers, after reaching a maximum of 79.5% for 3 hidden layers. For the existing speed, the prediction accuracy of 92% is highest for 9 hidden layers. The prediction accuracy for the new speed is less than for the existing speed. This is because the new speed of 2.795 m/sec has never been trained, i.e., the new speed is predicted based purely on training of the existing speed.

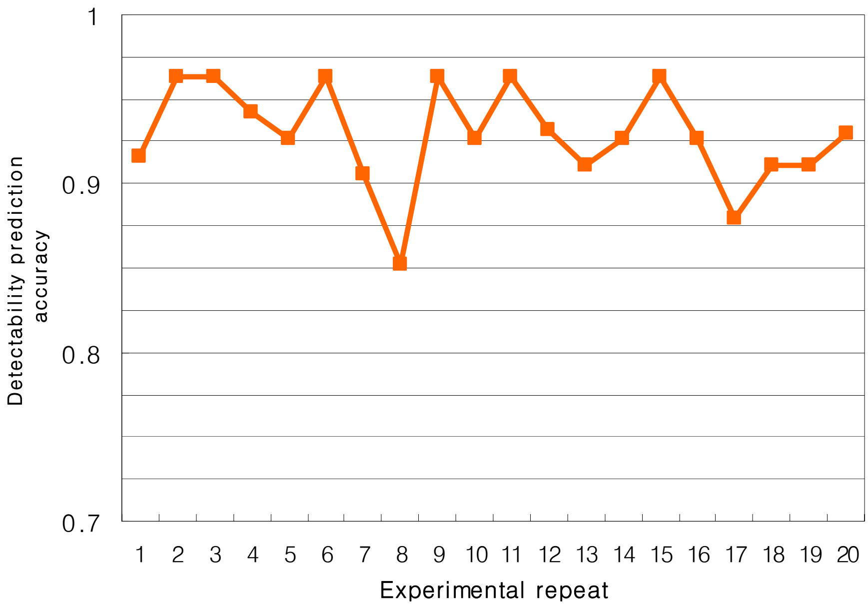


Fig. 7. Accuracy of RFID tag detectability prediction for the existing speed by 9 hidden layers of the model with 20 trials.

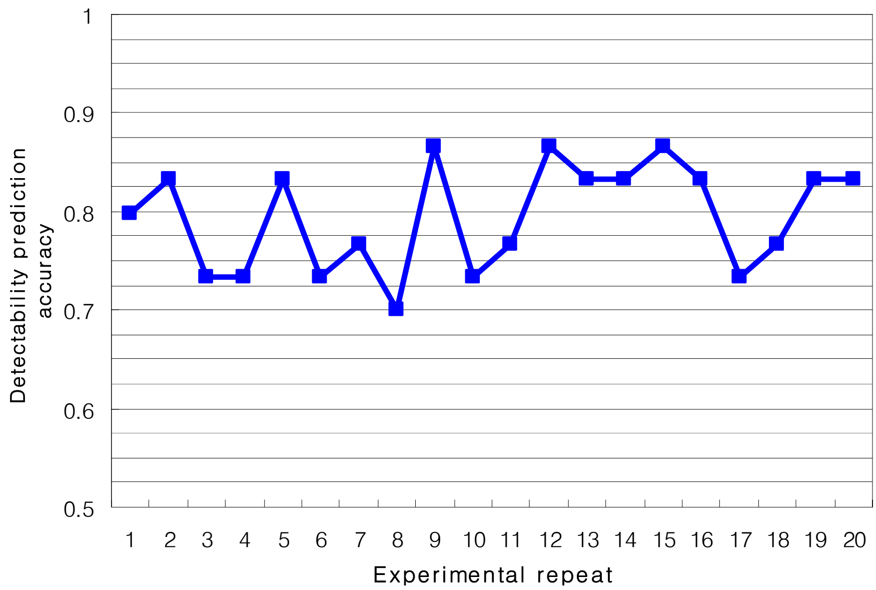


Fig. 8. Accuracy of RFID tag detectability prediction for the new speed by 3 hidden layers of the model with 20 trials.

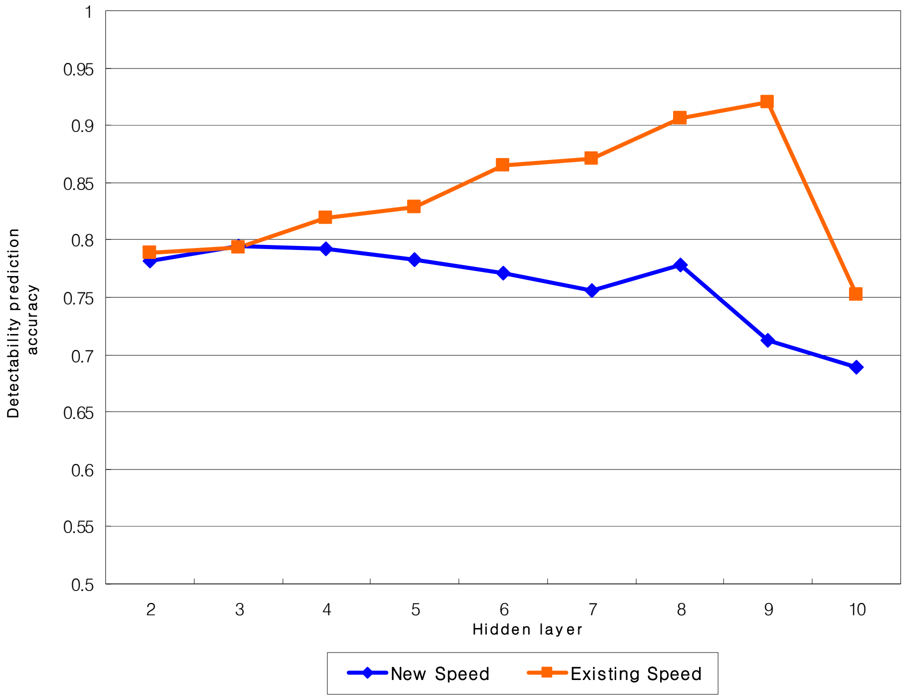


Fig. 9. Accuracy of RFID tag detectability prediction for the new speed and existing speed by various hidden layers of the model.

5. Conclusion

Although finding the best influence factor conditions enabling excellent RFID tag detection is very important at deployment, the work to date has been done by the time consuming trial-and-error experimental method. We propose an intelligent prediction scheme, a back-propagation learning-based model for RFID tag detection. The proposed method enables us to find the best influence factor conditions by predicting the RFID tag detectability. The proposed method yields time and cost-savings. The prediction capability of the new scheme is verified by comparing the results of the experimental method and the proposed intelligent method. In order to verify the pure prediction capability, an untrained influence factor condition of 2.75 m/sec as a new speed is introduced. The results of prediction accuracy are comparable to the trial-and-error method. This research is conducted for the water content on the conveyer. We plan to do research on different types of object and additional influence factors such as the number and height of antennas in the future.

References

[1] Rajit Gadh, “The state of RFID: heading toward a wireless internet of artifacts,” *COMPUTERWORLD*, August 11, 2004.

[2] J.L.M. Flores, S.S. Srikant, B. Sareen and A. Vagga, “Performance of RFID tags in near and far field,” *Personal Wireless Communications 2005, IEEE International Conference on* (*ICPWC 2005*), pp. 353-357, Jan. 2005.

[3] Bing Jiang, K.P. Fishkin, S. Roy and M. Philipose, “Unobtrusive long-range detection of passive

RFID tag motion," *Instrumentation and Measurement, IEEE Transactions on*, Vol. 55, Issue 1, pp. 187-196, Feb. 2006.

[4] Techsolutions: "Application note: RFID solutions for pallet tracking," http://www.techsolutions.co.za/PDF/Application%20Note% 20Pallet%20TrackingV13.pdf

[5] "RFID passport shield failure demonstration," http://www.flexilis.com/download/RFIDPassportShieldFailureDemonstration.pdf, June, 2006.

[6] Jeong-Hyun Park and Jong-Heung Park, "Postal RFID application and model and performance," *ETRI Journal, Vol. 28, Number 3,* pp. 405-408, June 2006

[7] Kevin Curran, Harkin Derek, Peter Mee and Tierney Ian, "An RFID enabled wide area gaming architecture," *Proceeding of the 4th MiNEVA Workshop in Sintra,* Lisboa, Portugal, pp. 43-57, June, 2006.

[8] D. W, Arti Patterson, Artificial neural networks, Prentice Hall, Aug. 1996.

[9] Chen Tang, Wenjing Lu, Song Chen, Zhen Zhang, Botao Li, Wenping Wang, Lin Han, "Denoising by coupled partial differential equations and extracting phase by backpropagation neural networks for electronic speckle pattern interferometry," *Applied Optics,* Vol. 46, Issue 30, pp. 7475-7484, October 2007.

[10] Po–Hung Chen and Hung-Chung Chen, "Application of Back-Propagation Neural Network to Power Transformer Insulation Diagnosis," *LNCS, Vol. 4993*, July 2007.

[11] Chengwei Li, Tao Song and Yonghong Xu, "The research and simulation in the liquid-level based on CMAC-PID controller," *Integration Technology, IEEE International Conference 2007 on*, pp. 348 – 351, 20-24 March 2007.

[12] Ayman H. Nasr, Mahmoud H. Ahmed, "Detecting Oil Spills in the Offshore Nile Delta Coast using Image Processing of ERS SAR Data," *WSEAS Transactions on Signal Processing,* Vol. 3, Issue 1, pp. 102-107, January 2007.

[13] D. E. Rumehart and J. L. McCelland, Parallel Distributed Processing, Vol. I, Cambridge, MA: *MIT Press*, 1986.

[14] B. Irie and S. Miyake, "Capabilities of three-layered perceptions," *Proceedings IEEE International Conference on Neural Networks*, 1988.

Minho Jo received B.S. degree in Industrial Engineering, Chosun Univ., South Korea and his Ph.D. in Computer Networks in Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA in 1994. He worked as a staff researcher with Samsung Electronics, South Korea and was professor of School of Ubiquitous Computing and Systems, Sejong Cyber University, Seoul, South Korea. He is now a Research Professor in School of Information and Communication Engineering, Sungkyunkwan Univ., Suwon, South Korea. He is Chair of the Steering Committee of KSII Transactions on Internet and Information Systems, General Chair of International Ubiquitous Conference, and Co-chair of International Conference on Ubiquitous Convergence Technology. He is Chairman of IEEE/ACM WiMax/WiBro Services and QoS Management Symposium, IWCMC 2008. He is Technical Program Committee of IEEE ICC2008 and IEEE GLOBECOM2008. He is Associate Editor of Journal of Security and Communication Networks published by Wiley and Sons, Referee of IEEE Transactions on Vehicular Technology, and Referee of IEEE Transactions on Mobile Computing. His interesting area lies in wireless ad-hoc sensor networks, RFID, security in communication networks, machine intelligence in communications, and ubiquitous computing.

Chang-Gyoon Lim received B.S and M.S. degrees in computer science, Chosun University, Korea in 1990 and his Ph. D. degree in computer engineering, Wayne State University, USA in 1997. He is currently Associate Professor of Computer Engineering at Chonnam National University. Simultaneously, he acts as a committee for several public and private regional institutions. He also leads several research projects of medium and small-sized business consortium and u-Payment as well as initiatives for innovation and quality in university teaching. The author of several articles on machine learning using soft computing, pattern recognition, and education technology, his current interests are ubiquitous and embedded software.

Emory W. Zimmers is the Director of the National Science Foundation's Center for Engineering Logistics and Distribution at Lehigh University (CELDi). He has been Professor of Department of Industrial and Systems Engineering, Lehigh Univ., Bethlehem, PA, USA since 1973. He received his Ph. D in Industrial and Systems Engineering from Lehigh University in 1973, BS and MS degrees from Lehigh University. His research interests include computer control of industrial systems, agility applications in manufacturing, logistics and distribution systems, new techniques in multi-media distance learning, entrepreneurship, and leadership development. He is co-author of a best selling textbook in the world recording 18th reprint since 1984, CAD/CAM. His research is coordinated with the Enterprise Systems Center at Lehigh University and often involves collaboration among education, industry and government partners to implement new findings. Courses which Dr. Zimmers teaches include computer integrated manufacturing, computer graphics, manufacturing problems, and experimental industrial engineering. Dr. Zimmers also oversees the senior projects.