9

The Processing of a Proposed Answer Sheet Using File Compression Method and Character Recognition Technique Based on Neutral Network

Dwi Astuti Aprijani
Tengku Eduard A. Sinar
Lintang Patria
Unggul Untan S

Abstract

Universitas Terbuka (Indonesia Open University) is one of the Mega Universities. In 2007, the number of undergraduate and graduate students in active enrolment is more than 350,000 students. Because of the number of its students, UT uses mostly multiple-choice test questions to evaluate the ability of students in understanding of the learning materials. The answers for each question are written on special, computer-readable answer sheets known as scanable forms. However, there are several rules and restrictions in filling the answer sheets. Those rules and restrictions include, but not limited to, the use of good quality computer-readable 2B pencils; neat and proper filling in the circle corresponding to the answer, with adequate darkness; completely erasing old answer without leaving any residues when changing an answer; and leaving the answer sheet clean and tidy without any folds/damages and unnecessary marks.

For most students, those rules and restrictions, apart from filling the answer sheets, impose other pressures as well. These pressures arise from their worries about not using proper quality pencils, unintentionally fill in the wrong answer, and accidentally fold, mark or damage the answer sheet. If any of those factors happens, it will increase the possibility of making mistakes that may damage their chances on passing the tests. Therefore, a technology that can handle all these problems must be developed.

This paper proposes a new answer sheet and its processing. This new answer sheet is easy to fill in. Examinees are not tight up to several rules and restrictions. They only need to write one uppercase letters (of five possible answers, A to E) on the answer sheet.

The proposed answer sheet processing method comprises of scanning the answer sheet and recognizing the hand-written, uppercase letters of each answer. Scanning of the proposed answer sheet is faster than the original answer sheet because its image area is smaller and its data component is fewer. The result of scanning to image files requires large space storage; hence a compression technique is employed in order to reduce storage space. Compression technique that we propose is a preprocessing against the image consisting of the separation between foreground and background, and the saving of the background into available format file, i.e. GIF, PNG, JPEG, and JPEG2000. The character recognition technique uses back propagation neural network.

We expect that the proposed answer sheet and its processing method are able to replace the original answer sheet that are currently being used, and that others can use them for performing selection tests, examinations and evaluations.

Introduction

Universitas Terbuka (Indonesia Open University) is one of the Mega Universities. In 2007, the number of undergraduate and graduate students in active enrolment is 350,000 students. Because of the number of its student, UT uses multiple-choice test questions to evaluate the ability of students in understanding of the learning materials. The answers for each question are written on special, computer-readable answer sheet. However, there are several rules and restrictions in filling the answer sheets. Those rules and restrictions include, but not limited to, the use of good quality computer-readable 2B pencils; neat and proper filling in the circle corresponding to the answer with adequate darkness; completely erasing old answer without leaving any residues when changing an answer; and leaving the answer sheet clean and tidy without any folds/damages and unnecessary marks.

For most students, those rules and restrictions impose other pressures because the successful doing of the test will affect the passing of courses they take. Factors that they might worry include the proper quality of the pencils, unintentionally fill in the wrong answer, and accidentally fold, mark or damage the answer sheet. If any of those things happen, it will increase the possibility of making mistakes that may damage their chances on passing the tests. Therefore a technology that can handle all these problems must be developed.

This paper proposes a new simple answer sheet. This new answer sheet is easy to fill in. Students do not need to use 2B pencils, do not need to fill in of the circle corresponding to the answer with adequate darkness. They only need to write one uppercase letters (of five possible answers, A to E) on the answer sheet. As consequences, the width of the answer sheet can be reduced.

The process of the proposed answer sheet comprises of scanning the answer sheet and recognizing the hand-written, uppercase letters of each answer. Scanning of the proposed answer sheet is faster than the original answer sheet because its image area is smaller and its data component is fewer. The result of scanning to image files requires large space storage; hence a compression technique is employed in order to reduce storage space. Compression technique that we propose is a preprocessing against the image consisting of the separation between foreground and background, and the saving of the background into available format file, i.e. GIF, PNG, JPEG, and JPEG2000. The character recognition technique uses back propagation neural network.

We expect that the proposed answer sheet and its processing method will be able to replace the original answer sheet that are currently being used, and that others can use them for performing selection tests, examinations, and evaluations.

Artificial Neural Network (Ann)

The method to recognize the handwriting is called handwriting recognition (Suen, 1993). Handwriting recognition is the ability of a computer to receive intelligible handwritten input (Wikipedia, the free encyclopedia). The handwriting recognition is one of the most advanced fields in the pattern recognition technology. One of the handwriting recognition techniques is Artificial Neural Network. This method uses the principle of human brain that consists of neuron as input processing to produce output based on the weight (Haykin, 1994).

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well.

ANN is one of artificial representations of the human brain that always try to stimulate learning process on that human brain. Term 'artificial' is used because this neural network is implemented using computer program that is able to handle a large calculation process during the learning process. ANN will do learning to form a reference model that can be used for pattern recognition (Kusumadewi, 2004).

An artificial neural network can be defined as an information processing system that has certain performance characteristics similar to biological neural networks. They have been developed as generalization of mathematical models of human cognition or neural biology based on the assumptions that (Fausett, 1994):

1. Information processing occurs in many simple elements that are called neurons (processing elements).
2. Signals are passed between neurons over connection links.
3. Each connection link has an associated weight, which, in a typical neural network, multiplies the signal being transmitted.
4. Each neuron applies an activation function (usually non-linear) to its net input to determine its output signal.

According to Schalkof (1992), there are 3 entities that form the characteristics of ANN, i.e.:

1. Network topology of neuron units
2. Characteristic of each unit or artificial network
3. Training and testing strategy

The superiority of ANN is its classification capability toward data that is never given before during training session (Han & Kamber 2001).

Compression Technique

In computer science and information theory, data compression or source coding is the process of encoding information using fewer bits (or other information-bearing units) than an unencoded representation would use through use of specific encoding schemes.

Compression is useful because it helps reduce the consumption of expensive resources, such as hard disk space or transmission bandwidth. The design of data compression schemes therefore involves trade-offs among various factors, including the degree of compression, the amount of distortion introduced (if using a lossy compression scheme), and the computational resources required to compress and uncompress the data.

> *Compression is a changing process of large data to source coding to reduce the size of storage and the time of transmission. In general, there are two basic techniques, i.e.: lossless and lossy.*

Lossless compression algorithms usually exploit statistical redundancy in such a way as to represent the sender's data more concisely, but nevertheless perfectly. Lossless compression is possible because most real-world data has *statistical redundancy*. For example, in English text, the letter ‘e’ is much more common than the letter ‘z’, and the probability that the letter ‘q’ will be followed by the letter ‘z’ is very small.

Another kind of compression, called lossy data compression, is possible if some loss of fidelity is acceptable. For example, a person viewing a picture or television video scene might not notice if some of its finest details are removed or not represented perfectly (i.e. may not even notice compression artifacts). Similarly, two clips of audio may be perceived as the same to a listener even though one is missing details found in the other. Lossy data compression algorithms introduce relatively minor differences and represent the picture, video, or audio using fewer bits.

Lossless compression schemes are reversible so that the original data can be reconstructed, while lossy schemes accept some loss of data in order to achieve higher compression.

However, lossless data compression algorithms will always fail to compress some files; indeed, any compression algorithm will necessarily fail to compress any data containing no discernible patterns. Attempts to compress data that has been compressed already will therefore usually result in an expansion, as will attempts to compress encrypted data.

In practice, lossy data compression will also come to a point where compressing again does not work, although an extremely lossy algorithm, which for example always removes the last byte of a file, will always compress a file up to the point where it is empty.

From these two basic techniques, can be established a standard format file for compression such as GIF, PNG, JPEG, and JPEG2000 can be. Each of this formats have its own advantage and disadvantage.

- **GIF** - Graphic Interchange Format. This highly-compressed, lossless format displays 8-bit raster images (less than or equal to 256 colors). It is a good choice for images that contain flat color areas and shapes with well-defined edges, such as type. It is also the choice for images that need a transparent background.
- **JPEG or JPG** - Joint Photographers Expert Group. This format is a better choice than GIF for continuous tone images (continuous gradiations of color or photos) and if your viewers have 24-bit monitors. JPG is a variable-compression image format; it is called "lossy" because the decompressed image does not have the same quality as the original image. The degree of compression affects file size and image quality. JPGs can only be saved in 8-bit per pixel gray-scale mode and 24-bit (the default) per pixel color mode.
- **JPG2000 or JPEG2000** is an initiative to update the standard. It uses state-of-the-art compression techniques based on wavelet technology which can allow an image to be retained without any distortion or loss.
- **PNG** - Portable Network Graphics. This graphics format was designed as the successor to GIF. It features compression, transparency, and progressive loading, like GIF, but it is free of patent and licensing restrictions. PNG supports images with millions of colors and produces background transparency without jagged edges; it is a lossless compression. PNG supports three main image types: true color, grayscale and palette-based (8-bit).

These entire four format files are used to compress scanned file. After that will be conducted statistical analysis to compare the performance of each method based on three factors i.e.: the ratio between the size of compressed file and the size of origin file, the speed of compression, and the quality of the image.

Research Design

The whole process that will be done in this research is shown in Figure 1.

1. Data Collection

The data collection that will be used in this research is the collection of handwriting sample from 500 UT staff. Every staff will write down the uppercase letters from A to Z, and the number from 0 to 9 for 10 times, so the total number of sample is 18.000 characters. The device to write down the uppercase letters and the number can be any kind of writing equipments such as pencil, ballpoint, spidol, etc., with variety of colors.

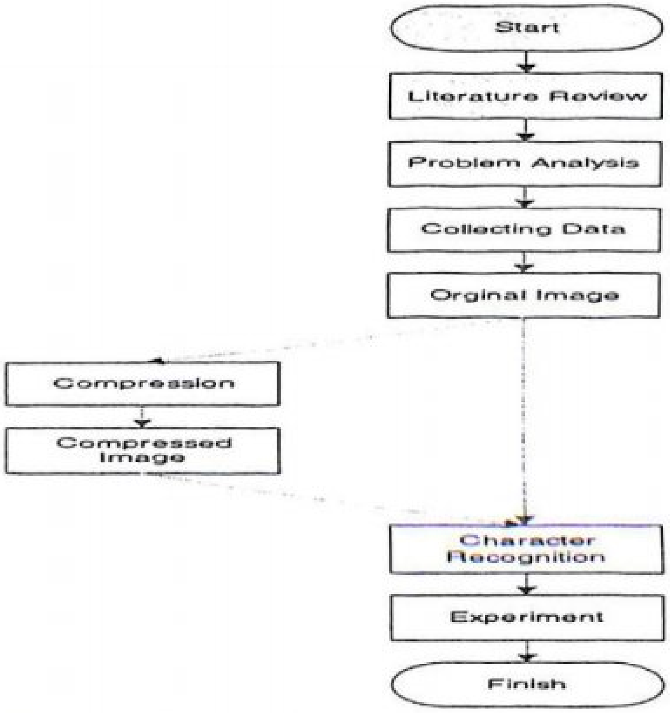

Figure 1. The flowchart of system model development

2. File Compression

The stages to develop system model of file compression in this research are shown in Figure 2. These stages comprise of preprocessing data collection, separation, and conversion to grayscale, compression, and development of prototype.

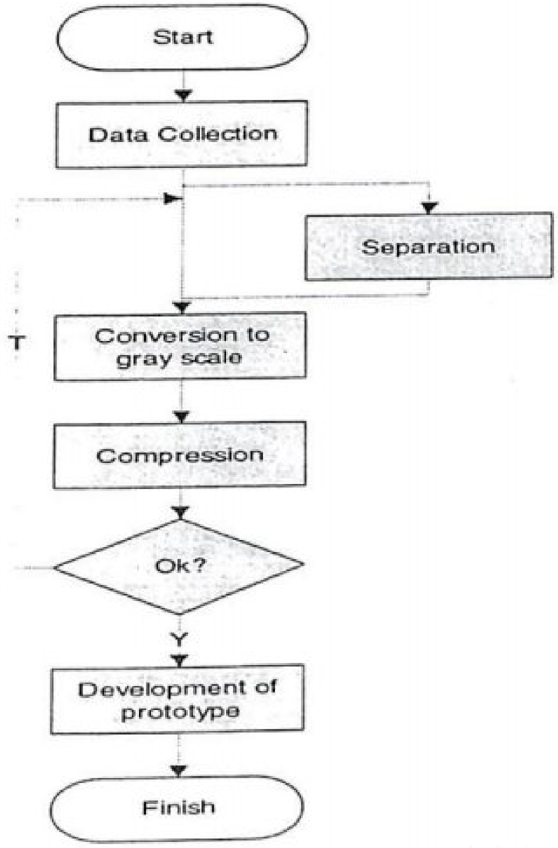

Figure 2 The flowchart of compression system model development

The data collected is a set of 18,000 unique characters in color image, consists of the 26 uppercase letters and the 10 numbers from the handwriting of 500 different people. This image (*foreground*) will be separated from its background and will be converted to grayscale image. This gray scale image then will be stored into the available file format, i.e.: GIF, PNG, JPEG, and JPEG2000. The next step is choosing a file format having the smallest size from these four file formats.

3. Character Recognition

The steps to develop the system model to recognize character are presented in Figure 3.

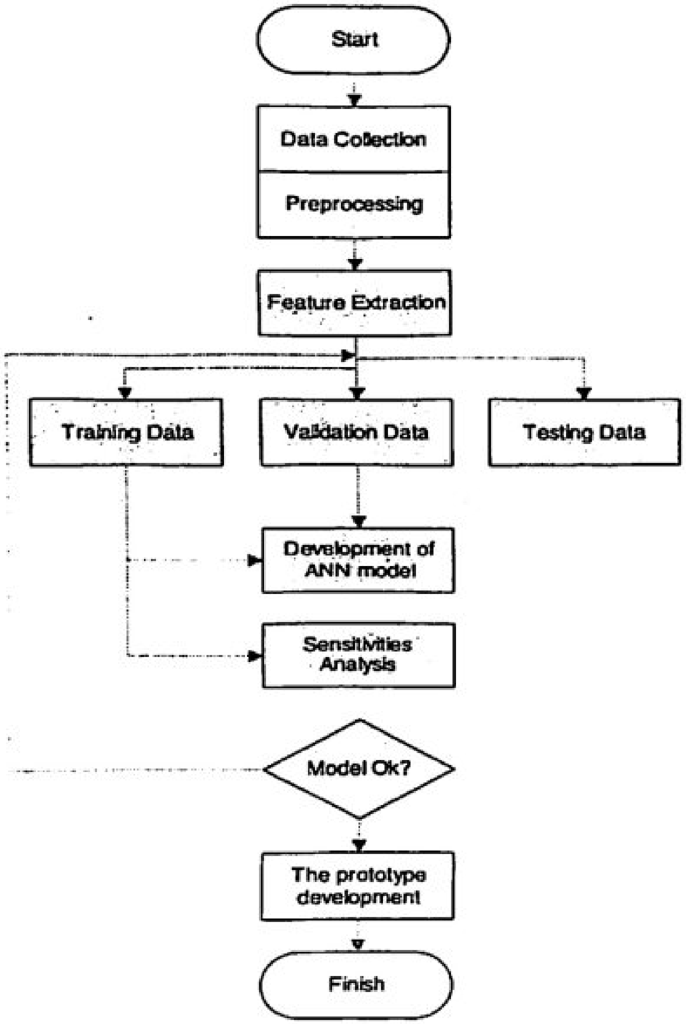

Figure 3 The flowchart of character recognition model system development

The character recognition process is started by preprocessing, feature extraction, and then prototype development.

3.1 Preprocessing

An image of uppercase letters or numbers (then called as sample) that will be identify using ANN have passed through specific steps, so it will be a good input for ANN. A good input for ANN is a set of numerical data. Therefore the main problem is how to convert a digital image into a set of numerical data that is representative and consistent. Although the ANN we are going to use has characteristic, structure, and configuration that are specific enough, there are still some other important parameters of ANN that has to be set to obtain the best result.

Each sample that will be observed and analyzed by the ANN must be well represented in numerical data form. It needs a method which can extract consistently feature data from each sample. Of course, the resulted numerical data must represent characteristics or features of the observed sample, so from a set of data with the same target will be produced a generalization or general featuring toward a sort of target. This data acquisition process must be really accurate considering all characters of each sample *(Nugraha, 2003).*

3.2 Sample Characteristics

Samples that will be observed must be limited by a dimension structure and a simple pixel homogenization so it will make analyzing the process toward this concept easier. Each sample is a digital image which its pixels which are categorized as two represented colors, i.e.: active color (black) and passive color (not black). The dimension of each sample is also limited by the area that is allocable on application program, but it will never reduce the expected flexibility and scalability.

3.3 Data Extraction

In this paper, we selected feature extraction methods, introduced by Frey (1991), to obtain the accurate and consistent data from each sample. Each character image was then scanned, pixel by pixel, to extract 16 numerical attributes. These attributes represent primitive statistical features of the pixel distribution. To achieve compactness, each attribute was then scaled linearly to a range of integer values from 0 to 15. This final set of values was adequate to provide a perfect separation of the 26 classes. That is, no feature vector mapped to more than one class.

The attributes (before scaling to 0-15 range) are:

1. The horizontal position, counting pixels from the left edge of the image, of the center of the smallest rectangular box that can be drawn with all "on" pixels inside the box.
2. The vertical position, counting pixels from the bottom, of the above box.
3. The width, in pixels, of the box.
4. The height, in pixels, of the box.
5. The total number of "on" pixels in the character image.
6. The mean horizontal position of all "on" pixels relative to the center of the box and divided by the width of the box. This feature has a negative value if the image is "leftheavy" as would be the case for the letter L.
7. The mean vertical position of all "on" pixels relative to the center of the box and divided by the height of the box.
8. The mean squared value of the horizontal pixel distances as measured in 6 above. This attribute will have a higher value for images whose pixels are more widely separated in the horizontal direction as would be the case for the letters W or M.

9. The mean squared value of the vertical pixel distances as measured in 7 above.
10. The mean product of the horizontal and vertical distances for each "on" pixel as measured in 6 and 7 above. This attribute has a positive value for diagonal lines that run from bottom left to top right and a negative value for diagonal lines from top left to bottom right.
11. The mean value of the squared horizontal distance tunes the vertical distance for each "on" pixel. This measures the correlation of the horizontal variance with the vertical position.
12. The mean value of the squared vertical distance times the horizontal distance for each "on" pixel. This measures the correlation of the vertical variance with the horizontal position.
13. The mean number of edges (an "on" pixel immediately to the right of either an "off pixel or the image boundary) encountered when making systematic scans from left to right at all vertical positions within the box. This measure distinguishes between letters like "W" or "M" and letters like "T" or "L."
14. The sum of the vertical positions of edges encountered as measured in 13 above. This feature will give a higher value if there are more edges at the top of the box, as in the letter "Y."
15. The mean number of edges (an "on" pixel immediately above either an "off pixel or the image boundary) encountered when making systematic scans of the image from bottom to top over all horizontal positions within the box.
16. The sum of horizontal positions of edges encountered as measured in 15 above.

After the normalization step, this numerical data will become input data of ANN. So the number of area in each sample will be appropriate to the number of ANN input neuron that is going to be used.

System Implementation

1. Computer System

The whole activities that will be conducted in this research are implemented on two server and client computers. The specification of server is Core 2 Duo T7200, 2GB DDR2, 120GB HDD, DVD±RW, 56K Modem, GbE NIC, WiFi, Bluetooth, VGA Intel GMA950 224MB (shared), 13.3" XGA, Win Vista Business. The specification of client is Core 2 Duo T7200, 2GB DDR2, 120GB HDD, DVD±RW, 56K Modem, GbE NIC, WiFi, Bluetooth, VGA ATI Radeon X1400 256MB, 15.4" WXGA, Win XP Pro.

2. Programming Software

Programming Software uses PowerBuilder Enterprise 10.5 for Windows x86 and MATLAB®, completed with Statistics Toolbox, Neural Network Toolbox, Wavelet Toolbox, Image Processing Toolbox, Image Acquisition Toolbox, Excel Link, Database Toolbox.

The Expected Outcome

From the experiment that will be conducted in this research, we expect that:

1. The proposed answer sheet and its processing method are able to replace the original answer sheet that are currently being used, and that others can use, for performing selection tests, examinations, and evaluations.
2. The prototype of the expanded data compression system can be used to compress data in order to reduce the need of space for storage.
3. The prototype of the developed character recognition system can be used to recognize the uppercase handwriting.

References

Duda, Richard O. Hart, Peter E. Stork, David G. 2000. *Pattern Classification.* John Wiley & Son, New York.

Engelbrecht, AP. Cloete, I. Zurada, JM. 1995. *Determining the Significance of Input Parameters Using Sensitivity Analysis.* College of Information Sciences and Technology. http:zSzzSzwww.cs.up.ac.zazSz%7EengelzSzpublicationszSzIWANN95a.pdf/ engelbrecht95determining.pdf [06 June 2006].

Faaborg, Alexander J. 2002. *Using Neural Networks to Create an Adaptive Character Recognition System*, Ithaca NY.

Fausett, L. 1994. *Fundamentals of Neural Network, Architectures, Algorithms, and Applications.* Prentice Hall, Englewood Cliffs, New Jersey.

Frey, Peter W., Slate, David J. 1991. Letter Recognition Using Holland-Style Adaptive Classifiers. Machine Learning, 6, 161-182 (1991).

Han, J. & Kamber, M. 2001. *Data Mining: Concept, Model, Methods, and Algorithm.* Wiley-Interscience, New Jersey.

Haykin, S. 1994. *Neural Networks: A Comprehensive Foundation.* NY: Macmillan.

Howe, D. 1993. *Free On-line Dictionary of Computing*, http://www.foldoc.org/

Kantardzic, M. 2003. *Data Mining: Concept and Techniques.* Morgan Kaufmann Publisher, San Fransisco.

Kusumadewi, S. 2004. *Membangun Jaringan Saraf Tiruan (Menggunakan Matlab dan Excel Link).* Yogyakarta. Graha Ilmu.

Kusumoputro, B., Philipus, E., Widyanto, M. Rahmat. 2000. Pengenalan Huruf Tulisan Tangan Menggunakan Logika Fuzzy dan Jaringan Syaraf Tiruan. *Seminar on Air - PPI Tokyo Institute of Technology 1999-2000 No.1 pages 34-38.*

Larose, D.L. 2005. *Discovering Knowledge in Data: An Introduction to Data Mining.* Wiley-Interscience, New Jersey.

Nugraha, A.P. and Mutiara, A.B. 2003. *Metode Ekstraksi Data untuk Pengenalan Huruf dan Angka Tulisan Tangan dengan Menggunakan Jaringan Syaraf Buatan Propagasi Balik.* Universitas Gunadharma.

Poh, H.-L., Yao, J.T. and Jasic, T. 1998. Neural networks for the analysis and forecasting of advertising and promotion impact. *International Journal of Intelligent Systems in Accounting, Finance and Management, 7(4), pages 253-268.*

Salameh, Walid A. and Otair, Mohammed A. 2004. *Online Handwritten Character Recognition Using an Optical Backpropagation Neural Network.*

Schalkof, RJ. 1992. *Stastical, Structural, and Neural Approaches. Canada. John Wiley & Son, Inc.*

Suen, C.Y., R. Legault, C. Nadal, M. Cheriet, and L. Lam. 1993. Building a New Generation of Handwriting Recognition Systems. *Pattern Recognition Letters*, vol 14, pp. 303-315, Apr.

Yao, J.T. 2003. Sensitivity Analysis for Data Mining. *Proceeding of 22nd International Conference of North American Fuzzy Information Processing Society - NAFIPS. Chicago. Illinois. 24 – 26 Juli 2003. pages 420 – 425.*