Volume 3, Issue 11, November 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Optical Character Recognition

Shimpy Goyal
Department of Computer Science and Applications,
M.D. University, Rohtak, Haryana, India

Abstract— Handwriting recognition is the ability of a computer to receive intelligible handwritten input. The image of the written text may be sensed "off line" from a piece of paper by optical scanning(optical character recognition). . Due to the poor performance achieved by these systems at that time, less research on handwriting recognition took place during the eighties. The first research works dealt with the recognition of isolated hand printed characters. These works followed on the works carried out on OCR, i.e. optical character recognition. Later on ,researchers began to study the recognition of on-line (dynamic) cursive handwriting and even more recently those of off-line(static) cursive handwriting. .More recently, the combination or cooperation of several independent recognizers, the use of lexicons or dictionaries and of language models as post-processing have been suggested to improve the overall efficiency of the system. In this paper to solve this problem, we constructed OCR system that saves abstracted characters to DB automatically after extracting only equivalent and necessary characters from a large amount of document by using Kohonen algorithm that is one of Artificial neural network.

Keywords— Optical Character Recognition, Modules, Images , Interface, Requirements and Implementation

1. INTRODUCTION

Vision:

Optical character recognition, usually abbreviated to **OCR**, is the mechanical or electronic conversion of scanned images of handwritten, typewritten or printed text into machine-encoded text. It is widely used as a form of data entry from some sort of original paper data source, whether documents, sales receipts, mail, or any number of printed records [1]. A study of Optical Character Recognition (OCR) techniques employed in automatic mail sorting equipment is presented. Methods and algorithms for image pre-processing, character recognition, and contextual post processing are discussed and compared. The objective of the study is to provide a background in the state-of-the-art of this equipment as the first element in a search for techniques to significantly improve the capabilities of postal address recognition.

Description:

- OCR is the acronym for Optical Character Recognition. This technology allows a machine to automatically recognize characters through an optical mechanism. Human beings recognize any objects in this manner our eyes are the "optical mechanism." But while the brain "sees" the input, the ability to comprehend these signals varies in each person according to many factors. By reviewing these variables, we can understand the challenges faced by the technologist developing an OCR system. First, if we read a page in a language other than our own, we may recognize the various characters, but be unable to recognize words. However, on the same page, we are usually able to interpret numerical statements - the symbols for numbers are universally used. This explains why many OCR systems recognize numbers only, while relatively few understand the full alphanumeric character range.
- Second, there is similarity between many numerical and alphabetical symbol shapes. For example, while examining a string of characters combining letters and numbers, there is very little visible difference between a capital letter "O" and the numeral "0." As humans, we can re-read the sentence or entire paragraph to help us determine the accurate meaning. This procedure, however, is much more difficult for a machine.
- Third, we rely on contrast to help us recognize characters. We may find it very difficult to read text which appears against a very dark background, or is printed over other words or graphics.

2. MODULES:

- Drawing Images: Though not directly related to neural networks, the process by which the user is allowed to draw the characters is an important part of the OCR application. Most of the actual drawing is handled by the process Mouse Motion Event. If the mouse is being drug, then a line will be drawn from the last reported mouse drag position to the current mouse position. It is not enough to simply draw a dot. The mouse moves faster than the program has time to accept all values for. By drawing the line, we will cover any missed pixels as best we can. The line is drawn to the off screen image, and then updated to the users screen.

Down sampling the Image:

- Every time a letter is drawn for either training or recognition, it must be down sampled. In this section we will examine the process by which this down sampling occurs. . All down sampled images will be stored in a gridof 5X7 matrix

Training the Neural Network:

- Learning is the process of selecting a neuron weight matrix that will correctly recognize input patterns. Neural Networks can be used to solve highly nonlinear control problems [2]. A Kohonen neural network learns by constantly evaluating and optimizing a weight matrix. Kohonen[3][4] has developed an algorithm with self-organizing properties for a network of self adaptive elements. To do this a starting weight matrix must be determined. This starting weight matrix is chosen by selecting random numbers. Of course this is a terrible choice for a weight matrix, but it gives a starting point to optimize from. Once the initial random weight matrix is created the training can begin. First the weight matrix is evaluated to determine what its current error level is. This error is determined by how well the trained inputs (the letters that you created) map to the output neurons. The error is calculated by the "evaluate Errors" method of the Kohonen Network class. If the error level is low, say below 10%, the process is complete. The training process begins when the user clicks the "Begin Training" button.

Recognition of characters:

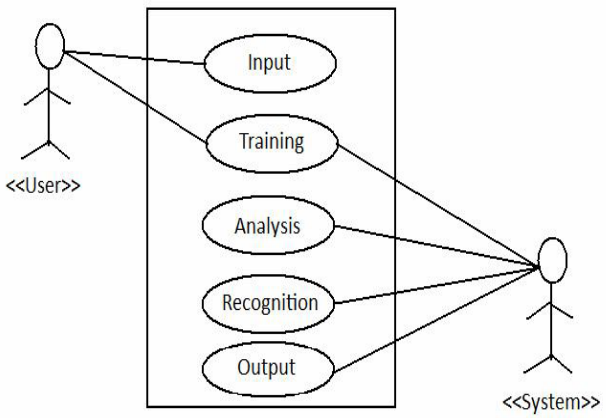
- The Kohonen neural network contains only an input and output layer of neurons. There is no hidden layer in a Kohonen neural network. Detection of text and identification of characters in scene images is a challenging visual recognition problem. As in much of computer vision, the challenges posed by the complexity of these images have been combated with hand-designed features [5], [6], [7] and models that incorporate various pieces of high-level prior knowledge [8], [9]. The input to a Kohonen neural network is given to the neural network using the input neurons. These input neurons are each given the floating point numbers that make up the input pattern to the network. A Kohonen neural network requires that these inputs be normalized to the range between -1 and 1.

3. GENERAL I/O AND DATA FLOW

- **Input Files:**
  Any character drawn by the user .A data file containing the learned characters and their associated property vectors.
- **GUI:**
  Add, and train application for new characters. Character recognition, downsampling, loading, deleting, saving and clearing facility.

- **Output Files:**
  A data file containing the trained ASCII translation of the drawn characters. Updated file of learned characters

- **Support of loading of characters already trained:**
  Alphabets, which are trained to recognize the characters by default, can be loaded directly to start working with the software.
  · Downsampling provisions to see how the image attributes are changed to make the image file of standard size.
  · Pixels mapping to find the most probable match.



Combined Use Case For User Interaction And System Processing

- **Support for drawing characters with personalized strokes:**
  · Users can further train the system for its own style of writing characters.
  · Personalized characters can be further trained and saved to make the character recognition better.
  · Image canvas area can be refreshed in case there of any discrepancies.

User Classes and Characteristics

A. Operating Environment

- Operating System: Microsoft Windows 98/ 2000/ XP Home / XP Professional/ Vista/7

B. Design and Implementation Constraints

- Automatic recognition of hand-written characters has many applications, but designing reliable systems is challenging because of the natural variations in human handwriting. This software doesn't support multiple characters written at a time. It can be trained for only one character at a time for more accurate recognition.
- Characters drew with sharp edges at the canvas boundaries cause conflicts in recognition.Black on white character images is supported.
- Only the single character entry is expected and supported by the software.
- The software will recognize skewed images of up to few degrees. Skewed images of a greater angle cannot be corrected by the system. In that case they need to be separately trained.
- Character Images are not taken separately from the file. They take only the characters drawn on the drawing area of the software.

C. Drawing Images: Though not directly related to neural networks, the process by which the user is allowed to draw the characters is an important part of the OCR application. Most of the actual drawing is handled by the process Mouse Motion Event. If the mouse is being drug, then a line will be drawn from the last reported mouse drag position to the current mouse position. It is not enough to simply draw a dot. The mouse moves faster than the program has time to accept all values for. By drawing the line, we will cover any missed pixels as best we can. The line is drawn to the offscreen image, and then updated to the users screen.

Downsampling the Image: Every time a letter is drawn for either training or recognition, it must be downsampled. In this section we will examine the process by which this downsampling occurs. . All down sampled images will be stored in a grid of 7x5 matrix.

Training the Neural Network: Learning is the process of selecting a neuron weight matrix that will correctly recognize input patterns. Neural Network is a very flexible package [10]. A Kohonen neural network learns by constantly evaluating and optimizing a weight matrix. To do this a starting weight matrix must be determined. This starting weight matrix is chosen by selecting random numbers. Of course this is a terrible choice for a weight matrix, but it gives a starting point to optimize from. Once the initial random weight matrix is created the training can begin. First the weight matrix is evaluated to determine what its current error level is. This error is determined by how well the training inputs (the letters that you created) map to the output neurons. The error is calculated by the "evaluate Errors" method of the Kohonen Network class. If the error level is low, say below 10%, the process is complete. The training process begins when the user clicks the "Begin Training" button.

Recognition of characters: The Kohonen neural network contains only an input and output layer of neurons. There is no hidden layer in a Kohonen neural network. The input to a Kohonen neural network is given to the neural network using the input neurons. These input neurons are each given the floating point numbers that make up the input pattern to the network. A Kohonen neural network requires that these inputs be normalized to the range between -1 and 1. Presenting an input pattern to the network will cause a reaction from the output neurons. In a Kohonen neural network only one of the output neurons actually produces a value. Additionally, this single value is either true or false. When the pattern is presented to the Kohonen neural network, one single output neuron is chosen as the output neuron. Therefore, the output from the Kohonen neural network is usually the index of the neuron that fired.

User Interface: The user interface allows the user to

- Load a file of learned characters (maybe automatic).
- Downsample the image.
- Initiate the OCR process.
- Identify ASCII value of unrecognized character.
- Save the ASCII file.

External Interface Requirements

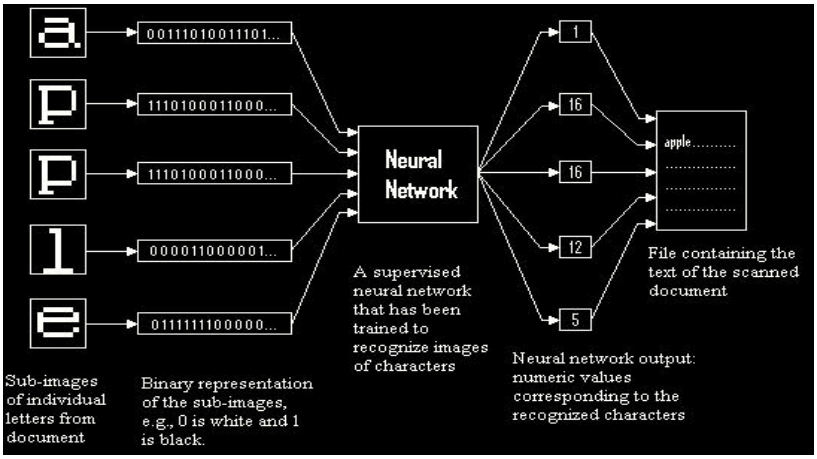
Hardware Requirement:

Intel Pentium III processors and above.
Minimum 128 MB RAM.
Hard-disk of minimum 5 GB capacity.
16-bit Monitor with minimum 640x480 screen resolution and 60 Hz refresh rate.
Mcrosoft compatible keyboard and mouse.

Software Requirement

Operating System: Microsoft Windows 98/ 2000/ XP Home / XP Professional/ Vista
Software Development Kit: j2sdk1.4.2_12
Java Net Beans IDE 5.5

4. SYSTEM ARCHITECTURE:



Block Diagram Of An Ocr

The OCR software breaks the image into sub-images, each containing a single character. The sub-images are then translated from an image format into a binary format, where each 0 and 1 represents an individual pixel of the sub-image. The binary data is then fed into a neural network that has been trained to make the association between the character image data and a numeric value that corresponds to the character. The output from the neural network is then translated into ASCII text and saved as a file.

5. PRODUCT REQUIREMENTS

Speed and Memory Efficiency:

Our focus in writing this software is functionality, rather than speed or memory efficiency. Of course, the reality is that time and space are valuable resources, but they are, in fact, valuable resources that OPTICA: An OCR (at least in its initial implementation) will consume with a voracious appetite. Even so, OPTICA: An OCR will hold only one file of trained character images in its memory. It is assumed that user must have enough memory to hold a page representation for character drawing and downsampled image.

Implementation and Portability:

Software is to be written in Java. The software must be portable to other Windows systems when recompiled, provided the Java Runtime Environment and Java Compilers are present.

6. CONCLUSION:

Automatic recognition of hand-written characters has many applications, but designing reliable systems is challenging because of the natural variations in human handwriting. One way to solve this problem is to use a neural network that learns to identify characters much like a person learns to read. If the training set for such a network is sufficiently large and diverse, the network will generalize to recognize hand-written numerals from unfamiliar sources. Algorithm based on Kohonen Neural Network is presented here, and it is shown to have accuracies of about 90%. The experiments also demonstrated that system complexity can be reduced significantly without degrading performance by considering two layered neural network rather than multiple layered neural network.

REFERENCES:

[1] http://en.wikipedia.org/wiki/Optical_character_recognition
[2] Neural Networks for self learning control systems Derrick H.Nguyen and Bernard Widrow
[3] Kohonen T (1988). Self-Organization and Associative Memory, Springer-Verlag, Heidelberg.
[4] Kohonen T (1991). Self-organizing Maps: Optimization and Approaches, ICANN, Espoo, Finland.
[5] T. E. de Campos, B. R. Babu, and M. Varma, "Character recognition in natural images," in Proceedings of the International Conference on Computer Vision Theory and Applications, Lisbon, Portugal, February 2009
[6] M. Yokobayashi and T. Wakahara, "Binarization and recog nition of degraded characters using a maximum separability axis in color space and gat correlation," in International Conference on Pattern Recognition , vol. 2, 2006, pp. 885– 888.
[7] J. J. Weinman, "Typographical features for scene text recognition," in Proc. IAPR International Conference on Pattern Recognition, Aug. 2010, pp. 3987–3990.
[8] J. Weinman, E. Learned-Miller, and A. R. Hanson, "Scene text recognition using similarity and a lexicon with sparse belief propagation," in Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 10, 2009.
[9] Y. Pan, X. Hou, and C. Liu, "Text localization in natural scene images based on conditional random field," in International Conference on Document Analysis and Recognition, 2009
[10] Neuralnet : training of neural networks Frauke Günther and Stefan Fritsch

