Journal of Computer Science 1 (3): 346-350, 2005
ISSN1549-3636

An Intelligent Multilingual Mouse Gesture Recognition System

[1]Nidal F. Shilbayeh and [2]Mahmoud Z. Iskandarani
[1]Faculty of Science and Information Technology, Al-Zaytoonah University
P.O. Box 911597, Post Code 11191, Amman, Jordan
[2]Faculty of Computer Science and Information Technology, Applied Science University
P.O. Box 41, Post Code 11931, Amman, Jordan

Abstract: A comprehensive mouse gesture system is designed and tested successfully. The system is based on UNIPEN algorithm in terms of mouse movements and applies its geometrical principles such as angles and transposition steps. The system incorporates Neural Networks as its learning and recognition engine. The designed algorithm is not only capable of translating discrete gesture moves, but also continuous sentences and complete paragraphs. Hopfield Network is also used for initial learning to add a feature of language independence to the system.

Key words: Mouse Gesture, Neural Network, Hopfield Neural Network, Pattern Recognition, Algorithm, Dynamic Sampling

INTRODUCTION

A person with disability has the same right to voice his or her opinions, to move about freely and to perform the tasks that can be performed by able-bodied individuals. These rights are often constrained by the individual's inability to move or to communicate due to their disability. Advances in technology however, have led to the development of machines, or assertive devices, which alleviate some of these restrictions.

Gestural interfaces to computers are gaining prominence in the research field of human-computer interaction owing to their ability to be tailored to a particular user's capabilities and their intuitiveness [1]. Developing a gesture recognizer is only part of the solution for human inabilities. It needs to be supported by the identification of appropriate clinical conditions under which gestural input would provide improved computer access.

A primary goal of gesture recognition is to provide a system which can identify specific human gestures and use them to convey information or for device control. From a biological and sociological perspective, gestures are loosely defined, thus, researchers are free to visualize and classify gestures as they see fit [1].

People frequently use gestures to communicate. Gestures are used for everything from pointing at a person to conveying information about space and temporal characteristics and do not simply embellish spoken language, but is part of the language generation process. Biologists define a gesture notion as to embrace all kinds of instances where an individual engages in movements whose communicative intent is paramount, manifest and openly acknowledged [2].

Gestures are interpreted to control computer memory and displays or to control actuated mechanisms. Human-computer interaction concentrates on the computer input-output interface, hence, the design of gesture language identification system. In such systems, gestures are created by a static hand or body pose, or by a physical motion in two or three dimensions and can be translated by computer into either symbolic commands or trajectory motion commands. Gestures may also be interpreted as letters of an alphabet or words of a language [3].

In computer interpretation of human generated gestures, certain concerns appear when designing a recognition system for such actions namely:

I. How consistent are people in their use of gestures?
II. What are the most common gestures used in a given domain and how easily are they recalled?
III. Do gestures contain identifiable spatial components, which correspond to the functional components of command scope and target?
IV. What kind of variability exists in a gesture and is the deviations predictable or random [4].

In this study the above questions would be answered through our design of comprehensive neural network based mouse gesture system. This system captures and intelligently recognizes mouse movements.

Background: Handwriting (Mouse movements) recognition systems are similar to gesture recognition systems, because all of these systems perform recognition of something that moves, leaving a "trajectory" in space and time [4, 5]. Real time "dynamic" recognition systems identify handwriting as a user writes. These systems have the advantage of capturing the dynamic information of writing, including the number of strokes, the ordering of strokes and the direction and velocity profile o each stroke.

Most of available systems capture writing as a sequence of coordinate points, taking into account character blending and merging and the problem of characters that have close similarity to each other. This is solved via pre-processing of the characters prior to recognition, hence, performing a shape recognition process. The post-processing recognition process can be achieved using zones that define eight directions of travel, where characters are recognized as connected zones using a lookup table for matching and classification. Other techniques of handwriting recognition, such as dot search above letters technique, signal processing and Fourier coefficient based technique and basic components of pen strokes technique do exist but suffer low recognition rates and time delay of more than a second.

Based on the use of gestures by humans, some requirements for handwriting recognition system design suggest themselves such as:

I. To choose gestures that fit a useful environment.
II. To create a system which can recognize non-perfect human created gestures.
III. To create a system which can use both static and dynamic information components of gestures.
IV. To perform gesture recognition with image data presented at field rate or as fast as possible.
V. To recognize the gesture as quickly as possible, even before the full gesture is completed.
VI. To use a recognition method which requires a small amount of computational time and memory.
VII. To create an expandable system which can recognize additional types of gestures.

System Design: In designing our novel mouse gesture system with neural networks recognition engine, we tried to satisfy two main sets of requirements:

1. The User Requirements

I. Simple to use-user friendly
II. Flexible enough to cope with day-to-day variations in the condition of the user.
III. Aesthetically acceptable-the user must not feel uncomfortable when using it.

2. The Technical Requirements

I. Have a high degree of input recognition.
II. In the event of an error, non-recognition is implemented instead of miss-recognition, especially in critical conditions.
III. Recognizes inputs quickly.
IV. Require minimal training-using Artificial Neural Networks.
V. Able to be incorporated into or communicate with existing programs.

To implement the above mentioned points the following algorithm is designed:

I. Main Process

If (the neural engine is previously trained)
    Execute the Recognition Process;
else
    Execute The Learning Process using Hopfield neural network;

II. The Learning Process

If (there is a stored pattern)
{ Load the stored pattern;
Display the learned pattern(output pattern);
}
While (the stored pattern does not produce a good recognition OR there isn't a stored pattern)
Do
{ Input training pattern;
Calculate the weight of the input pattern;
Display the learned pattern(output pattern);
}

III. The Recognition Process

Enter the password using mouse gesture movement;
If (not a correct password)
    Try again;
else
{ Enter your choice; // C means character recognition, T means text recognition
Switch (choice) {
Case 'C': Enter type of recognition;
    Execute The Character Recognition Process;
Case 'T': Enter type of recognition;
    Execute Text Recognition Process;

IV. The Character Recognition Process

Draw the character;
If (the drawn character is recognized)
    Display a typed version of the character;
else
    Display a blank screen;

V. Text Recognition Process

While (not end of input) **do**
{ execute the character recognition process;
minimize the recognized character into the required writing font size;
put it in the proper place according to the desired format;
}

System Implementation: Recognizing patterns can be accomplished using many techniques [4, 5]. One of the most popular and known one is Hopfield neural network. However, a technique can be tailored to a particular set of characters or gestures such as the one shown in Fig.1. The two patterns in the figure correspond to the character Z in the English alphabet; hence it is up to the programmer to train his system to recognize either one or both. To accomplish the idea of

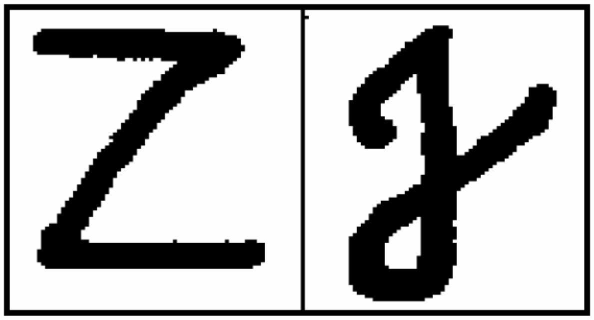

Fig. 1: Gestural Representations of Character Z

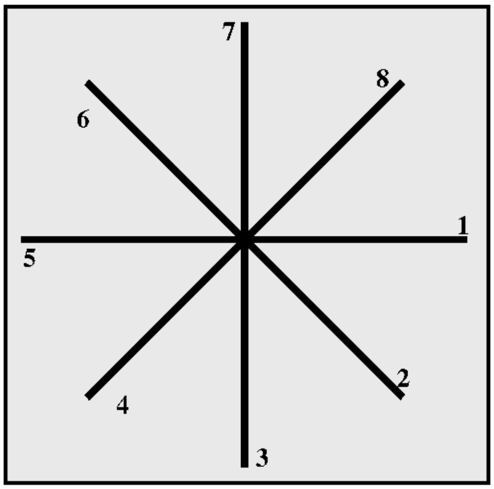

Fig. 2: The Direction Model

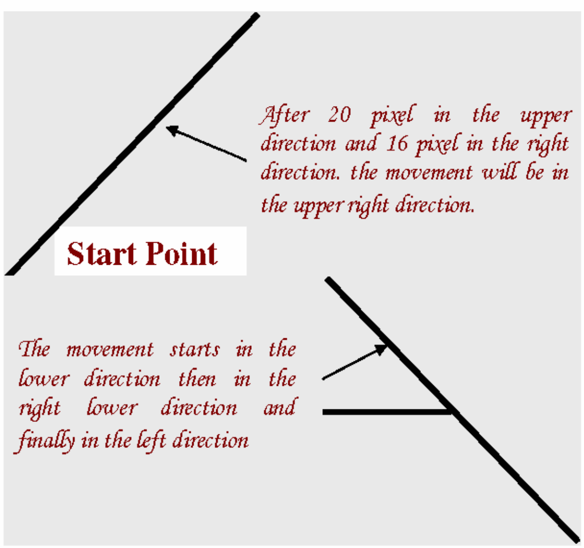

Fig. 3: Basic Movements to Recognize Letter A

gesture recognition a directional dynamic technique based on vector movements is designed and implemented. Figure 2 shows the eight main directions adopted in the used model.

Using the directional model, the movements of the mouse are stored in an array, the following illustrates how does the letter A gets its representation using the discussed technique and algorithm

Array A = {7, 8, 7, 1, 3, 2, 3, 5, 0, 0, 0, 0, 0, 0,0},

Movements

7: movement in the upper direction,
8: movement in the upper right direction,
7: movement in the upper direction,
1: movement in the right direction,
3: movement in the lower direction,
2: movement in the lower right direction,
3: movement in the lower direction,
5: movement in the left direction,

Note that movement in direction 1is necessary to indicate the 16-pixel movement along the x-axis.

The contents of the array are the numbers from 1 to 8. If the movement accrued between any of the stored vectors, no change in the previous direction will happen until the deviation between two successive points is greater than 20 pixels in the vertical direction and 16 pixels in the horizontal direction.

To enable the system to recognize a set of mouse movements, [6, 7] (English characters) the movements are stored in a database file with the corresponding standard (typed) characters as shown in Fig. 3.

Since the mouse movement is stored in a database file and the system can be trained to recognize these movements then different set of mouse movements can be stored in different database files, which will make our system universal to any set of gestures. Figure 4 and 5 shows English characters and Arabic characters as examples used to test the multilingual ability of our system. Figure 6 shows an example of how our algorithm works by illustrating how the character A is sampled and saved in an array in the database file. The algorithm was further developed to recognize not only discrete English characters but also complete words, sentences and even whole paragraph as shown in Fig. 7.

The above is achieved by implementing a sub algorithm which produces a writing screen once the program is run. This part of the program place each recognized English character in an array then the contents of the array will be displayed on the screen once the user start to move the mouse to construct the next character.

In case the user constructed the wrong character a facility for deletion of that mouse gesture movement is built into the software using either the backspace or space on the computer keyboard the system is also capable of recognizing mouse movement that represent the universally use numerals as shown in Fig. 8, [8, 9].

The reliability of our designed and implemented mouse gesture system enabled us incorporate a mouse gesture movement as a password to run our algorithm (in this case the character Z) as shown in Fig. 9.

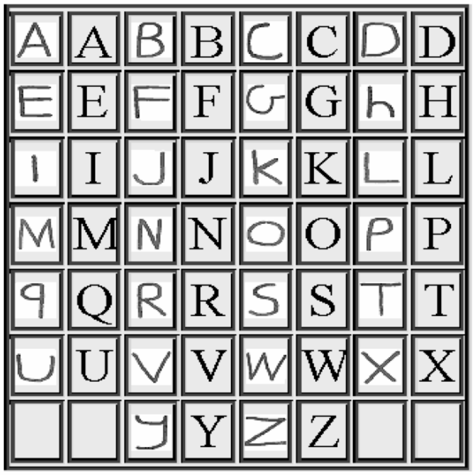

Fig. 4: English Character Set (Mouse Generated and Equivalent)

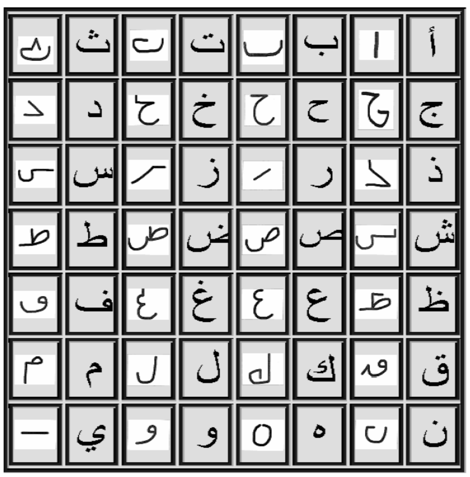

Fig. 5: Arabic Character Set (Mouse Generated and Equivalent)

Fig.6: Character A Representation

Once the correct password is constructed using the mouse the writing area shown above becomes active which will enable the user to start inputting characters, numerals, or sentences [10, 11].

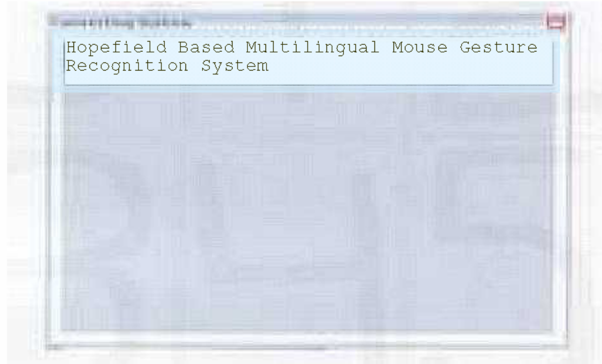

Fig.7: Program Main Menu

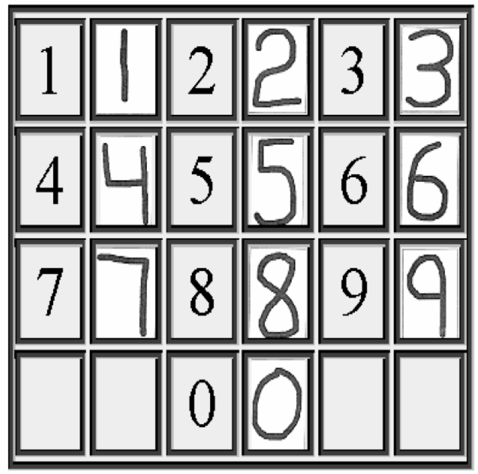

Fig. 8: Arabic Numerals Character Set (Mouse Generated and Equivalent)

Fig. 9: The Correct Password Representation

DISCUSSION AND CONCLUSION

It is clear that the collective computational properties of the used network and algorithm produced good results. Gesture movements are retained as stable entities and can be correctly identified using the trained Hopfield model. Ambiguities are resolved based on statistical assumptions. Some capacity of generalization is present. These properties follow from the designed recognition model and algorithm, which is phase-space dependent and is not strongly associated with precision of movements. This robustness indicates that our algorithm as it proved to be can be generally applied to any type of gestural movements with minimum level of modification [12, 13].

REFERENCES

1. Thomas Baudel and Michel Beaudouin-Lafon, 1993. CHARADE: Remote control of objects using free-hand gestures. Communications of the ACM, 36: 28-35.
2. Wren, C., A. Azarbayejani, T. Darrel and A. Pentland P_nder, 1995. Real-time tracking of the human body, in pho-tonics east. SPIE Proceedings, Vol. 2615, Bellingham, WA.
3. Zhang, H. and D. Zhong, 1995. A scheme for visual feature based image retrieval. Proc. SPIE Storage and Retrieval for Image and Video Database.
4. Charles J. Cohen, Lynn Conway and Dan Koditschek, 1996. Dynamical system representation, generation and recognition of basic oscillatory motion gestures. 2nd International Conference on Automatic Face- and Gesture-Recognition, Killington, Vermont.
5. Manjunath, B. and W. Ma, 1996. Texture features for browsing and retrieval of image data. IEEE T-PAMI.
6. Pentland A., R. Picard and S. Sclaro, 1996. Photobook: Content-based Manipulation of Image Database. Intl. J. Computer Vision.
7. Popescu, M. and P. Gader, 1998. Image content retrieval from image database using feature integration by choquet integral. SPIE Storage and Retrieval for Image and Video Database, VII.
8. Imagawa, K., S. Lu and S. Igi, 1998. Color-based hands tracking system for sign language recognition. Proc. of IEEE FG'98, pp: 462-467.
9. Nigam, K., A. Mccallum, S. Thrun and T. Mitchell, 1999. Text classification from labeled and unlabeled documents using EM. Machine Learning.
10. Mitchell, T., 1999. The role of unlabeled data in supervised learning. Proc. Sixth Intl. Colloquium on Cognitive Science.
11. Rui, Y., T. Huang and S. Chang, 1999. Image retrieval: Current techniques, promising directions and open issues. J. Visual Comm. Image Representation, 10: 1-23.
12. Ying Wu and Thomas S. Huang, 2000. Color tracking by transductive learning. Proc. of IEEE Conference on CVPR'2000, 1: 133-138, Hilton Head Island SC.
13. Ying Wu, Qi Tian and Thomas S. Huang, 2000. Discriminant-EM Algorithm with Application to Image Retrieval. Proc. of IEEE Conference on CVPR'2000, 1: 222-227, Hilton Head Island SC.