JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 23, 1623-1637 (2007)

Short Paper

Adaptive Mandarin Morse Code Recognition using Fuzzy Support Vector Machines and a Variable Degree Variable Step Size Least-Mean-Square Algorithm

CHENG-HONG YANG[1], HSIU-CHEN HUANG[2], LI-YEH CHUANG[3] AND CHENG-HUEI YANG[4]
[1]*Department of Electronic Engineering*
National Kaohsiung University of Applied Sciences
Kaohsiung, 807 Taiwan
[2]*Department of Physical Medicine and Rehabilitation*
Chiayi Christian Hospital
Chiayi, 539 Taiwan
[3]*Department of Chemical Engineering*
I-Shou University
Kaohsiung, 840 Taiwan
[4]*Department of Electronic Communication Engineering*
National Kaohsiung Marine University
Kaohsiung, 811 Taiwan

Morse code has been shown to be valuable in assistive technology, augmentative and alternative communication, rehabilitation, and education. In this paper, Morse code is selected as a communication adaptive device for persons with various neuromuscular diseases such as amyotrophic lateral sclerosis, multiple sclerosis, and muscular dystrophy. Maintaining a stable typing rate presents a major obstacle to being effective as a communication tool. Therefore, an adaptive automatic recognition method with a high recognition rate is needed. The proposed system uses both fuzzy support vector machines and the variable-degree variable-step-size least-mean-square algorithm. Statistical analyses demonstrated that the proposed method elicited a higher recognition rate than other algorithms in the literature.

Keywords: Morse code, adaptive tool, least-mean-square algorithm, support vector machines, fuzzy theory

1. INTRODUCTION

Recent advances in computer science and related technologies allow adaptive tools, combined with computer software and hardware, to play a more important role in the lives of persons with disabilities. Consequently, many computer assisted key-in systems have been developed for the disabled, *e.g.* the head mouse, mini-keyboard, king-keyboard, trackball, joystick, alternative keyboard, key-guard, and touch screen [1, 2]. Many researchers have focused on a reduced set of switches for these input devices with an

---

Received August 10, 2005; revised June 5 & August 10 & September 21, 2006; accepted September 27, 2006.
Communicated by Pau-Choo Chung.

efficiency rate approaching one keystroke per selected Mandarin character. To help persons whose hand coordination and dexterity are impaired by disabilities such as amyotrophic lateral sclerosis, multiple sclerosis, and muscular dystrophy, a substitute keyboard is needed. Morse code has been shown to be an excellent candidate for such a communication adaptive device [3-8].

A stable typing rate is strictly required for Morse code to be used effectively as a communication tool. However, this restriction is a major hindrance. Therefore, a suitable adaptive automatic recognition method is needed. We used an advanced recognition method, a combination of fuzzy support vector machines (FSVMs) [9, 10] and the variable-degree variable-step-size least-mean-square (VDVSLMS) algorithm [11], to increase the prediction performance. The theory of support vector machines (SVMs), based on the theoretical learning theory [12], is a new and promising technique for data classification and regression [13, 14]. SVMs have been successfully applied to a number of problems, such as handwritten character and digit recognition, face recognition, text categorization and object detection in machine vision [15, 16]. It has been demonstrated that SVMs have a higher performance than traditional learning machines [9, 10]. Statistical analyses indicate that the proposed method elicited a higher recognition rate than other algorithms described in the literature.

2. METHOD

Morse code is a system of asynchronous data bits composed of binary encoded circuit opposites (long-short) used for transmission and reception of alphanumeric information, which allows each character to be translated into a predefined sequence of dots and dashes (the elements of Morse code). A dot is represented as a period ".", while a dash is represented as a hyphen or minus sign "-". Each tone element, dot or dash, is transmitted by sending a signal for a standard length of time. Per definition, the tone ratio for dot to dash and the silent ratio for dot-dash space to character-space must be 1:3 in Morse code [6]. However, according to many users' practical experience, automatic recognition of Morse code is challenging, since maintaining a stable typing rate is difficult to achieve.

According to a report about computer skills by the Mandarin Computer Foundation of the Republic of China (R.O.C.), the Mandarin phonetic symbols and the Chanjei input method are the two most popular methods of entering Mandarin words into a computer. Morse code can be selected as an adaptive input tool to operate Mandarin language computers. One of the authors of this study (Yang, 2003) edited a new Mandarin phonetic and Chanjei Morse code (see Figs. 1 and 2) so that the new Morse code is simpler, faster, and easier to remember when compared to alternative codes found in the literature [17, 18].

In 1996, Luo and Shih [19] proposed a system that could recognize varying typing speeds using an adaptive technique, the least-mean-square (LMS) algorithm. Their system could adjust its characteristics to successfully recognize a message under unstable typing conditions, but had limits on the typing speed variation. To satisfy the limitation of 0.7 to 2.0 times the present typing speed, a user with disabilities had to be well trained in order for the system to successfully recognize his or her Morse code message. However, complying with this limitation proved difficult for beginners or persons with serious

Fig. 1. Morse code for Mandarin phonetic symbols [17].

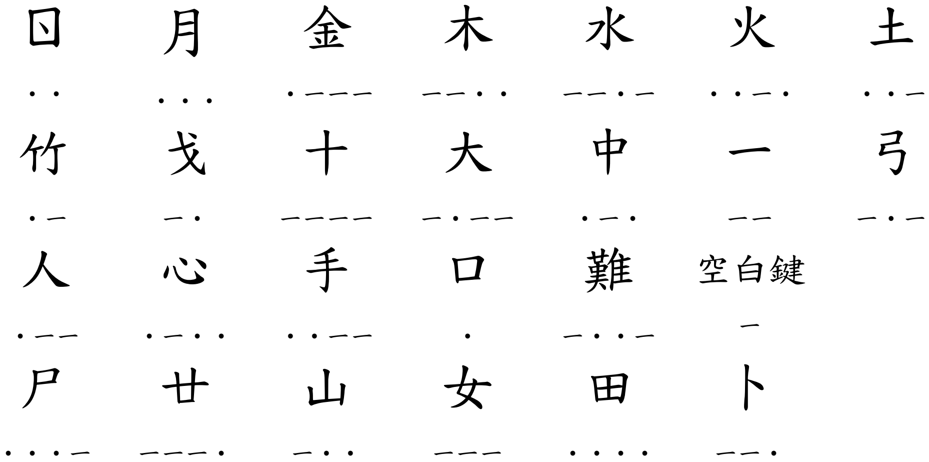

Fig. 2. Morse code for Mandarin Chanjei symbols [18].

disabilities. Subsequently, Shih and Luo [20] proposed an improved method that combined the least-mean-square algorithm with a character-by-character matching technique to over come this limitation.

In this paper, the proposed method is divided into five stages: tone recognition, training process, space recognition, adaptive processing, and character recognition. A block diagram of the Morse code recognition process is shown in Fig. 3. Initially, the input data stream is sent individually to either the tone recognition or the space recognition,

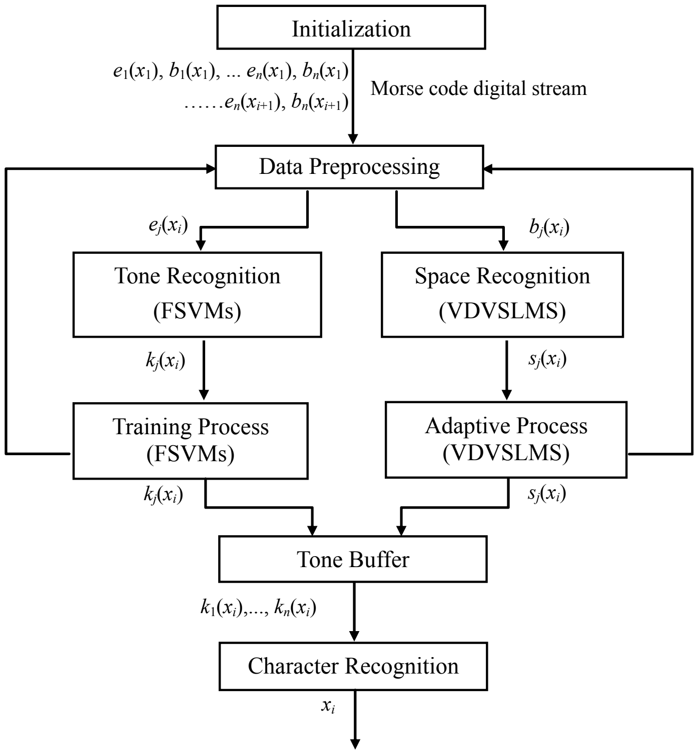

Fig. 3. Block diagram of the Morse code recognition process.

depending on the length of the switch-down time (tone element) or switch-up time (space element). In the tone recognition stage, the tone element value is recognized as either a dot or a dash, and then sent to the training process, which recalculates the decision function. Simultaneously, in the tone buffer section, the recognized tone element (dot or dash) and each successive tone element are saved in a dot-dash buffer and a tone element buffer. In the space recognition stage, the space element value is recognized as being either a dot-dash space (the space between elements of one character) or a character space (the space between characters), which is then fed into the adaptive processing stage. After a character space is obtained, the value(s) in the tone buffer is (are) sent to the character recognition stage, which identifies the appropriate Mandarin character [6].

A Morse code character, x_i, is represented as follows:

$$e_1(x_i),\ b_1(x_i),\ \ldots,\ e_j(x_i),\ b_j(x_i),\ \ldots,\ e_n(x_i),\ b_n(x_i),\ 1 \le j \le n \tag{1}$$

where

$e_j(x_i)$: when a key is pressed down, it is presented as ‘dot’ or ‘dash’, depending upon the duration of the j^{th} Morse code element of the input character x_i.

$b_j(x_i)$: when a key is held up, it is presented as one of three spaces: the space between elements of one character, the space between characters, or the space between words, which is the duration of the j^{th} space of the input character x_i.

$k_j(x_i)$: a dot or dash which is recognized from $e_j(x_i)$.
$s_j(x_i)$: a dot-dash space or character space which is recognized from $b_j(x_i)$.
n: the total number of Morse code elements in character x_i.

2.1 Tone Recognition

The tone recognition process is used for identifying the tone elements as either a dot or dash. Initially, the element value must be in scale to the data-preprocessing phase. The data is treated by scaling to obtain an input value within a range of [– 1, + 1]. This can be described by the following operation:

$$x = lower + [(upper - lower)(\frac{tone\_element - tone\_\min}{tone\_\max - tone\_\min})] \tag{2}$$

where *lower* = – 1, *upper* = + 1, *tone_element* is an input tone value, *tone*_min is the minimal dot-dash value of input points, and *tone*_max is the maximal dot-dash value of input points. In the tone recognition stage, the tone element value is first recognized as either a dot or a dash. It is then sent to the training process stage, which is used to recalculate the decision function.

In this stage, the scaled tone value (x), can be sent into the decision function

$$f(x) = sign(\sum_{i \in I_{s.v.}} \bar{\alpha}_i y_i K(x_i,\, x_j) + \bar{b}) \tag{3}$$

to determine the recognized value as being either a dash ($f(x) \geq 0$) or a dot ($f(x) < 0$), where $\bar{\alpha}_i$ is an optimal solution, $y_i \in \{-1, +1\}$ and $\bar{b}$ is the bias. Since the initial decision function is absent at the start, a training process should be performed to calculate the decision function of the FSVM. An initial training data set has to be defined first. To determine the initial training data set, the relationship of the first twelve sequential values of tone elements is computed. If a value is twice as large as its neighbor value, the value is treated as long (*tone*_L); the smaller ones represent short values (*tone*_S). The *tone*_L and *tone*_S values are taken as candidates for the training data set. Once the *tone*_L and *tone*_S values are obtained, the average of them can be calculated, and assigned as L_base and S_base separately. The equations for calculating an initial standard *tone*_base is shown below:

*tone*_base = (L_base + S_base)/2. (5)

After the *tone*_base has been obtained, the first twelve values of tone elements can be identified as a dash. Each one is labeled $y_i = +1$; otherwise it is labeled $y_i = -1$. Based on the training data set, the training process is performed to calculate the initial decision function for the tone recognition.

After the initialization phase, the new tone value of the input stream will be put into the decision function $f(x)$ to determine the recognized value as being either a dash ($f(x) \geq 0$) or a dot ($f(x) < 0$). After the tone recognition, the resulting value can be labeled and sent into the training data set. Then, the training process is performed to recalculate the decision function and wait for the next value.

2.2 Training Process

The Vapnik-Chervonenkis (VC) theory [12] provides a general measure for a model set complexity regarding statistical learning. Based on the VC theory, SVMs can be designed for classification problems; however, they are sensitive to corrupted data, such as noise or less meaningful data far away from their own classes, which affect the generalization capability of SVMs. For this reason, Huang and Liu [10] proposed a concept that fuzzifies the data set for training in so-called fuzzy support vector machines. A detailed description of FSVMs is given below.

2.3 Fuzzy Support Vector Machines (FSVMs)

Suppose we are given a set S of labeled training points with associated fuzzy membership

$$(y_1, x_1, m_1), \ldots, (y_l, x_l, m_l). \tag{6}$$

Each training point $x_i \in \Re^N$ is given a label $y_i \in \{-1, 1\}$ and a fuzzy membership $\varepsilon \leq m_i \leq 1$ with $i = 1, \ldots, l$, and sufficiently small $\varepsilon > 0$. In most cases, the search for a suitable hyperplane in an input space is too restrictive to be of practical use, especially if S is not linearly separable. A solution to this situation is mapping the input space into a higher dimension feature space and searching for the optimal hyperplane in this feature space. Let $z_i = \psi(x_i)$ denote the corresponding feature space vector with a mapping ψ from $\Re^N$ to a feature space Z. For instance, the training data x_i is mapped into a (possibly infinite) vector in a higher dimensional space.

Hence Eq. (7) below represents a problem in an infinite dimensional space which is not trivial to solve. Currently the main procedure is to solve a dual formulation of Eq. (7). It needs a closed form of $K(x_i, x_j) \equiv \psi(x_i)^T \psi(x_j) = z_i \cdot z_j$ which is usually called the kernel function. Some popular kernels are, for example, the RBF kernel: $e^{\frac{-\|x_i - x_j\|^2}{2\delta^2}}$ and the polynomial kernel: $(x_i^T x_j + \delta)^d$ where δ and d are kernel parameters. Since the fuzzy membership m_i is the attitude of the corresponding point x_i toward one class and the non-negative variable ξ_i is a measure of error in the SVMs, the term $m_i\xi_i$ is a measure of error with different weights. The optimal hyperplane problem, called primal problem, is then regarded as the solution to

$$\text{minimize } \frac{1}{2} w^T \cdot w + C \sum_{i=1}^{l} m_i \xi_i, \tag{7}$$

$$\text{subject to } y_i(w^T \cdot z_i + b) \geq 1 - \xi_i,\ i = 1, \ldots, l, \tag{8}$$

$$\xi_i \geq 0,\ i = 1, \ldots, l, \tag{9}$$

where C is a constant. It should be noted that a smaller m_i reduces the effect of the parameter ξ_i in Eq. (7) in such a way that the corresponding point x_i is treated as less important.

To solve this optimization problem we construct the Lagrangian

$$L(w, b, \xi, \alpha, \beta) = \frac{1}{2}w^T \cdot w + C\sum_{i=1}^{l} m_i\xi_i - \sum_{i=1}^{l}\alpha_i(y_i(w^T \cdot z_i + b) - 1 + \xi_i) - \sum_{i=1}^{l}\beta_i\xi. \tag{10}$$

Then the saddle point of $L(w, b, \xi, \alpha, \beta)$ can be identified. The parameters must satisfy the following conditions:

$$\frac{\partial L(w, b, \xi, \alpha, \beta)}{\partial w} = w - \sum_{i=1}^{l}\alpha_i y_i z_i = 0, \tag{11}$$

$$\frac{\partial L(w, b, \xi, \alpha, \beta)}{\partial b} = -\sum_{i=1}^{l}\alpha_i y_i = 0, \tag{12}$$

$$\frac{\partial L(w, b, \xi, \alpha, \beta)}{\partial \xi_i} = m_i C - \alpha_i - \beta_i = 0. \tag{13}$$

Applying these conditions in the Lagrangian Eq. (10), Eq. (7) can be transformed into the dual problem, and becomes

$$\text{maximize } W(\alpha) = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(x_i, x_j), \tag{14}$$

$$\text{subject to } \sum_{i=1}^{l} y_i\alpha_i = 0, \ \ 0 \le \alpha_i \le m_i C, \ i = 1, \ldots, l, \tag{15}$$

and find the solution $(\overline{w}, \overline{b}, \overline{\xi}, \overline{\alpha})$ for Karuch-Kuhn-Tucker (KKT) conditions [19], which are defined as

$$\overline{w} - \sum_{i \in I_{s.v.}} \overline{\alpha}_i y_i z_i = 0, \tag{16}$$

$$-\sum_{i \in I_{s.v.}} \overline{\alpha}_i y_i = 0, \tag{17}$$

$$y_i(\overline{w} \cdot z_i + \overline{b}) - 1 + \overline{\xi}_i \ge 0, \ i = 1, \ldots, l, \tag{18}$$

$$(m_i C - \overline{\alpha}_i)\overline{\xi}_i = 0, \ i = 1, \ldots, l, \tag{19}$$

$$\overline{\alpha}_i(y_i(\overline{w} \cdot z_i + \overline{b}) - 1 + \overline{\xi}_i) = 0, \ i = 1, \ldots, l, \tag{20}$$

where $\overline{w}, \overline{b}, \overline{\xi}, \overline{\alpha}$ are optimal solution and $I_{s.v.} = \{j \mid x_j \text{ is a support vector}\}$.

The last one of the above mentioned KKT complementarily condition is the primal form. Since $\overline{w} = \sum_{i \in I_{s.v.}} \overline{\alpha}_i y_i z_i$, $\overline{b}$ can easily be solved from the KKT complementarily condition by choosing any i for which $\overline{\alpha}_i > 0$. The point x_i with the corresponding $\overline{\alpha}_i > 0$ is called a support vector. There are two types of support vectors. The one corresponding with $0 < \overline{\alpha}_i < m_i C$ lies on the margin of the hyperplane. The one corresponding with $\overline{\alpha}_i = m_i C$ is upper-bounded to control the effect of misclassified points.

After the dual form is solved, the decision function is written as:

$$f(x) = sign(\sum_{i \in I_{s.v.}} \bar{\alpha}_i y_i K(x_i \cdot x) + \bar{b}). \tag{21}$$

An important difference between SVMs and FSVMs is that points with the same value of $\bar{\alpha}_i$ may indicate a different type of support vector in FSVM due to the factor m_i [10].

2.4 Fuzzy Memberships

Sequential learning and inference methods are important in many applications involving real-time signal processing [9]. Morse code recognition can be regarded as a sequential learning problem. For example, we would like to have a learning machine in which points from the recent past are given more weight than points far back in the past. For this purpose, we can select the fuzzy membership as a function of the time at which the point was generated.

Suppose we are given a sequence of training points

$$(y_1, x_1, m_1, t_1), \ldots, (y_l, x_l, m_l, t_l) \tag{22}$$

where $t_1 \leqq \ldots t_i \leqq t_l$ is the point in time where the data arrives in the system. Let fuzzy membership m_i be a function of time t_i

$$m_i = f(t_i) \tag{23}$$

such that $m_1 = \varepsilon \leq \ldots \leq m_l = 1$.

Different time-dependent functions could be chosen here, *i.e.* a linear function, quadric function or a sigmoid function. We used the linearly time-dependent function Eq. (24) shown below to determine fuzzy membership:

$$m_i = f(t_i) = at_i + b \tag{24}$$

where $a = 0.1$, $b = \varepsilon$.

Kernel-Adatron (KA) algorithms without bias [16] are used to simplify the FSVMs training procedures, as follows.

Step 1: Initialize $\alpha_i^0 = 0$.
Step 2: For $i = 1, \ldots, l$, execute steps 3, 4 below.
Step 3: For labeled point (x_i, y_i) calculate:

$$z_i = \sum_{j=1}^{l} \alpha_j y_j K(x_i, x_j). \tag{25}$$

Step 4: Calculate $\delta\alpha_i^t = \eta(1 - z_i y_i)$:

4.1 If $(\alpha_i^t + \delta\alpha_i^t) \leq 0$ then $\alpha_i^t = 0$

4.2 If $m_i C > (\alpha_i^t + \delta\alpha_i^t) > 0$ then $\alpha_i^t = (\alpha_i^t + \delta\alpha_i^t)$

4.3 If $(\alpha_i^t + \delta\alpha_i^t) \geq m_i C$ then $\alpha_i^t = m_i C$

Step 5: If a maximum number of iterations are exceeded or the margin λ is approximately 1 then stop, otherwise return to step 2 for the next t. The margin λ is given below.

$$\lambda = \frac{1}{2}[\min_{\{i|y_i=+1\}}(z_i) - \max_{\{i|y_i=-1\}}(z_i)] \tag{26}$$

where η is the learning rate of the training procedures. η has a value of 0.1 in this study.

2.5 Space Recognition

The space recognition stage is used to identify the space between characters and to isolate an individual Morse code element of a character. Initial_S is the initial silent_base value. The procedure for this character detection operation is shown below.

Step 1: initiate $j = 1$.
Step 2: if $b_j(x_i) <$ silence_base , then go to step 3, otherwise go to step 4.
Step 3: $b_j(x_i)$ is a dot-dash space. Let $j = j + 1$ and go to step 2.
Step 4: $b_j(x_i)$ is a character space. Then a sequence of tone durations between the character spaces is obtained. Go to step 1.

Due to the absence of the initial space length value initial_S, the first character x_i cannot be isolated immediately. Thus, the initial space length initial_S is obtained by taking the first nine values of silent elements as reference values; then, all of the values taken are sorted in descending order. After sorting, the relationship among each value is compared. If a value is twice larger than any other value, this value is represented as long (*silent*_L), if the value is smaller it is represented as short (*silent*_ S). Once this relationship has been determined, the average of these nine values can be calculated and is assigned as the initial space length initial_S. Whenever an initial_S value is obtained, the data stream is separated into a character space set and a dot-dash space set. After the Morse code elements of a character are isolated from a data stream, the elements can be recognized in the character recognition stage.

2.6 Adaptive Process

Use a variable-degree, variable-step size LMS algorithm (VDVSLMS) instead of a simple LMS algorithm, the convergence speed of the algorithm rises, allowing the weight to reach an optimal solution in a smaller number of iterations. This in turn reduces the computational cost of the whole process [11].

The VDVSLMS used in the system serves to cleverly change the *silent*_base in order to predict an unstable typing speed. This change utilizes the current data to compute a new weight vector using the weight update recursion of the standard LMS with step size μ [11]. $X(n)$ is the input vector of the nth most recent input data, $\varepsilon(n)$ being the error. The VDVSLMS equation generated by the weight update recursion for the algorithm is

$$W(n+1) = W(n) - \alpha_2(n)\hat{\nabla}(n) \tag{27}$$

where

$$\alpha_2(n) = 2\mu(1 - \mu X^T(n)X(n)) \tag{28}$$

where the subscript on $\alpha_2(n)$ is used to indicate the degree, and

$$\hat{\nabla}(n) = -\ 2\varepsilon(n)X(n) \tag{29}$$

is an estimate of the gradient.

$$\varepsilon(n) = d(n) - X^T(n)W(n) \tag{30}$$

where $d(n)$ is the scalar of the desired signal. μ is the step-size parameter that controls the speed of convergence, as well as the steady-state and/or tracking behavior of the adaptive filter. The step size μ has a value of 0.02 in our system. An algorithm with nine input signals $X(n)$ was used in this experiment. After the updated weight recursion $W(n + 1)$ is calculated, it can be multiplied with the input vector $X^T(n)$ and the result used as a new adjusted *silent*_base.

2.7 Character Recognition

Once a character space value has arrived in the *tone*_buffer, it is a signal for the tone buffer elements to be sent to character recognition. If the recognized character set can be directly matched to a code set from the Morse code table, then it is immediately translated from the Morse code table. Otherwise, it has to be translated by the following minimum distance calculation. First, each tone element value in an unknown tone element stream is divided by the FSVM's decision function of the previous tone element set. Then, the distances between each tone value and the code elements in each character of the Morse code table are calculated. The character with the minimum Euclidean distance to the tone value is chosen as the value for the unknown character. The procedure for determining the shortest Euclidean distance is the following. First, each tone element, $e_j(x_i)$, is divided by the decision function for $j = 1 \sim n$. Then, the roots of the sum of the square distances between the new tone element, $e_j(x_i)$, and the character in the Morse code table are calculated. The character in the Morse code table that has the shortest Euclidean distance is recognized as the unknown character [6].

3. EXPERIMENTAL RESULTS AND DISCUSSIONS

Morse code is now being harnessed for use in rehabilitation applications of augmentative-alternative communication and assistive technology, facilitating mobility, environmental control and adapted worksite access. To use microprocessor devices for many individuals with motor and/or sensory disabilities, Morse code is selected as an input system through switches external to the computer, using newly-developed adapted-access software programs, hardware peripherals, and learning methods. People with limited movement or sensory capabilities have been shown to successfully operate computers and other devices via adapted switching mechanisms and Morse code emulation of keyboard input functions. Research and clinical experience are indicating that the fast rate of entry and low level of physical exertion inherent in a Morse code input system could make it a viable and competitive method of microprocessor control for persons with disabilities [23].

The recognition system is implemented using the Microsoft Visual C++ 6.0 Windows versions. To investigate the efficiency of the proposed method, testing data (Dis-n) were collected from two participants with disabilities (P1 and P2). The first test participant (P1) was a 14-year-old boy, who has been diagnosed with cerebral palsy. His voluntary movements were accessible but an initial delay was exhibited before the movement was initiated. An IQ (Intelligent Quotient) test showed his intelligence to be normal. Participant 2 (P2) was a 14-year-old female adolescent, diagnosed with cerebral palsy, athetoid type. She exhibited involuntary and uncontrollable movements of her four limbs. The involuntary motion increases when she is excited. Her intelligence is normal, but disarticulation is noted, resulting in difficulties of verbal communication. Each participant typed 100 phonetic and Chanjei characters in 15 test samples, which are denoted Dis-01 to Dis-15 in Tables 1 and 2, respectively. The 100 characters were randomly chosen in the form of short paragraphs, which can be considered representative of average texts. Some characters of the text passages might occur more than once, since in any normal text passage certain characters appear more frequently than others.

Since the Luo and Shih system [19] is restricted to a variation of typing speeds ranging from 0.67-2.0 times the typing speed of the preceding character, it wasn't included for comparison in this study. Instead the proposed method was compared against Shih and Luo's improved system [20] and a system developed by Yang [21, 22]. Tables 1 and 2 illustrated that the proposed method consistently produced the highest number of matches for phonetic and Chanjei Morse code recognition respectively. The average number of matches for SL, Yang, and the proposed method were 70.87, 85.56, and 88.60 for Mandarin phonetic Morse code recognition. The average number of matches for SL, Yang, and the proposed method were 69.37, 84.73, and 88.07 for Mandarin Chanjei Morse code recognition.

The reason for the proposed method obtaining such a high recognition rate is the following. FSVMs have a good generalization capability for classification problems, and the membership functions can adjust weights associated with the time relationship between points of the recent past and points farther back in the past. In other words, the proposed method can adapt to a user's typing rate. Furthermore, according to the statistical theory, FSVMs can determine the optimization margin for character recognition.

A Morse code time series is generally an unstable one, unstable in speed and/or in rate that can generate two kinds of errors: a space recognition error and a tone recognition error. Maintaining precise intervals is a difficult task for any persons. Generally speaking, a person's typing rate is constant over a short period of time. That is, the person's present typing rate is similar to the typing rate of the immediately preceding several words. Because each person has his or her individual typing speed, the dot and dash values cannot be set to a default value. Therefore, in the proposed method, the S_base, L_base, and tone_base were constantly recalculated to predict the next space and tone length as each tone element was entered.

Table 1. Number of correct Mandarin phonetic character matches (out of 100) using three adaptive methods for two test participants with disabilities.

Problems	P1			P2			P1&P2 (average)		
	SL	Yang	NEW	SL	Yang	NEW	SL	Yang	NEW
Dis01	70	93	93	79	95	96	74.5	94.0	94.5
Dis02	66	78	84	81	90	92	73.5	84.0	88.0
Dis03	75	87	92	81	94	95	78.0	90.5	93.5
Dis04	70	89	91	73	93	93	71.5	91.0	92.0
Dis05	71	88	91	67	74	76	69.0	81.0	83.5
Dis06	71	89	91	75	95	95	73.0	92.0	93.0
Dis07	73	83	89	74	88	90	73.5	85.5	89.5
Dis08	70	90	92	81	94	94	75.5	92.0	93.0
Dis09	79	90	92	78	91	94	78.5	90.5	93.0
Dis10	72	86	91	77	90	93	74.5	85.0	92.0
Dis11	77	77	92	70	94	94	73.5	85.5	93.0
Dis12	73	79	91	82	95	95	77.5	87.0	93.0
Dis13	66	84	88	75	94	94	70.5	89.5	91.0
Dis14	62	88	91	32	44	49	47.0	66.0	70.0
Dis15	68	90	92	32	45	48	50.0	67.5	70.0
Average	70.87	86.07	90.67	70.47	85.07	86.53	70.67	85.56	88.60
Standard error	4.37	4.86	2.23	16.19	17.29	16.16	9.39	8.38	8.03

Legends: SL: Shih and Luo's method [19], Yang: Yang's method [21], NEW: the proposed method.

Table 2. Number of correct Mandarin Chanjei character matches (out of 100) using three adaptive methods for two test participants with disabilities.

Problems	P1			P2			P1&P2 (average)		
	SL	Yang	NEW	SL	Yang	NEW	SL	Yang	NEW
Dis01	75	86	89	75	89	91	75.0	87.5	90.0
Dis02	68	87	89	68	89	91	68.0	88.0	90.0
Dis03	64	73	78	73	79	85	68.5	76.0	81.5
Dis04	81	87	89	73	89	92	77.0	88.0	90.5
Dis05	73	90	92	72	89	92	72.5	89.5	92.0
Dis06	70	83	88	61	76	82	65.5	79.5	85.0
Dis07	65	86	90	70	82	86	67.5	84.0	88.0
Dis08	74	86	89	72	91	91	73.0	88.5	90.0
Dis09	68	86	91	69	84	88	68.5	85.0	89.5
Dis10	59	85	87	65	83	88	62.0	84.0	87.5
Dis11	67	82	85	74	90	91	70.5	86.0	88.0
Dis12	66	79	85	59	74	78	62.5	76.5	81.5
Dis13	67	88	90	70	89	91	68.5	88.5	90.5
Dis14	67	83	87	70	89	92	68.5	86.0	89.5
Dis15	73	83	87	73	85	88	73.0	84.0	87.5
Average	69.13	84.27	87.73	69.60	85.20	88.4	69.37	84.73	88.07
Standard error	5.36	4.13	3.35	4.67	5.40	4.14	4.24	4.28	3.15

Legends: SL: Shih and Luo's method [20], Yang: Yang's method [22], NEW: the proposed method.

The three recognition methods were compared by applying the number of matches, in two statistical tests, the Friedman test and the multiple comparison approach [24]. The Friedman test was used to test whether the total numbers of matches of the different recognition methods were equal. The multiple comparison approach was used to determine which method had significantly different median total matches if the Friedman test was rejected.

3.1 Friedman Test

The Friedman test is a nonparametric counterpart of the parametric two-way analysis of variance test and was used to compare the total matches of the recognition methods when the distribution of the underlying population was not specified. The hypothesis being tested was that all the methods had equal median total matches, and the alternative hypothesis was that all methods did not have equal median total matches. The null hypothesis is rejected at the α significance level if the value of the test statistic exceeds the $1 - \alpha$ quantile of the F-distribution with $k - 1$ and $(n - 1)(k - 1)$ degrees of freedom. The calculated value of $T_f = 91.70$ is greater than the critical value of $F_{.05}$ (2, 28) = 2.02. We rejected the null hypothesis that all the methods had the same median total matches at a significance level of $\alpha = 0.05$.

3.2 Multiple Comparison Approach

Following the multiple comparison test given in Conover [24], the total matches of the three methods were ordered in an array, and a rank was assigned to each corresponding to its order. For both Mandarin phonetic and Chanjei Morse code recognition, the rank sums of NEW, Yang, and SL are 45.0, 30.0, and 15.0 for the test problems. If the rank sums of any two methods are greater than 2.02 units apart (with $\alpha = .05$), they may be regarded as having unequal medians. Therefore, it can be concluded that the proposed method here is statistically superior to both the Yang and SL's methods.

4. CONCLUSIONS

Morse code has been shown to be a valuable tool in assistive technology, augmentative and alternative communication, and rehabilitation for people with various neuromuscular diseases, such as amyotrophic lateral sclerosis, multiple sclerosis, and muscular dystrophy [6]. A major hindrance is a stable typing rate, which is strictly required for the recognition of Morse code characters. In this paper, a combination of FSVMs and a VDVSLMS algorithm were applied to the Morse code recognition problem. Statistical analyses demonstrated that the proposed method elicited a recognition rate of as high 88.60 and 88.07 for Mandarin phonetic character and Chanjei character respectively, an improvement over alternative methods from the literature. A higher recognition rate is very important for persons with disabilities, since they tend to feel tired and weary, and thereby make comparatively more errors in their operation than most people without disabilities.

REFERENCES

1. R. Bower, *et al.* (eds.), *The Trace Resourcebook-Assistive Technology for Communication*, *Contro*l, *and Computer Access*, Trace Research & Development Center, Universities of Wisconsin-Madison, Waisman Center, 1998.
2. S. P. Levine, J. D. Gauger, L. D. Bwera, and K. J. Khan, "A comparison of mouth stick and Morse code text inputs," *AAC Augmentative and Alternative Communication*, Vol. 2, 51, 1996, pp. 45-51.
3. L. N. Goble and H. A. Colle, "High-speed Morse code training," in *Proceedings of the IEEE National Aerospace and Electronics Conference*, 1985, pp. 944-951.
4. R. Trace and D. Center, *Two Switch Auto-Repeat Morse Code*, Waisman Center, University of Wisconsin-Madison, 1984.
5. D. A. Shannon, W. S. Satewen, J. Miller, and B. S. Cohen, "Morse-code controlled computer aid for the non-vocal quadriplegic," *Medical Instrumentation*, Vol. 15, 1981, pp. 341-343.
6. C. H. Yang, L. Y. Chuang, and C. H. Luo, "Morse code application for wireless environmental control system for severely disabled individuals," *IEEE Transactions on Neural Systems & Rehabilitation Engineering*, Vol. 11, 2003, pp. 463-469.
7. C. H. Yang, "An interactive Morse code emulation management system," *Computer and Mathematics with Applications*, Vol. 46, 2003, pp. 479-492.
8. C. H. Yang, L. Y. Chuang, and C. H. Luo, "Internet access for disabled persons using Morse code," *International Journal of Computers and Application*s, Vol. 26, 2004, pp. 10-16.
9. C. F. Lin and S. D. Wang, "Fuzzy support vector machines," *IEEE Transactions on Neural Networks*, Vol. 13, 2002, pp. 464-471.
10. H. P. Huang and Y. H. Liu, "Fuzzy support vector machine for pattern recognition and data mining," *International Journal of Fuzzy Systems*, Vol. 4, 2002, pp. 826-835.
11. M. A. Khasawneh and K. A. Mayyas, "A newly derived variable degree variable steps size LMS algorithm," *International Journal of Electronics*, Vol. 79, 1995, pp. 255-264.
12. V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer Verlag, 1995.
13. A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. N. Vapnik, "A support vector clustering method," in *Proceedings of the International Conference on Pattern Recognition*, Vol. 2, 2000, pp. 728-732.
14. C. Cortes and V. N. Vapnik, "Support vector network," *Machine Learning*, Vol. 20, 3, 1995, pp. 273-297.
15. C. Campbell, N. Cristianini, and J. Shawe-Taylor. "Dynamically adapting kernels in support vector machines," *Advances in Neural Information Processing Systems*, Vol. 11. 1999, pp. 204-210.
16. C. Campbell, N. Cristianini, and T. T. Frieß, "The Kernel-Adatron: a fast and simple learning procedure for support vector machines," in *Proceedings of the 15th International Conference on Machine Learning*, 1998, pp. 188-196.
17. C. H. Yang, "A newly developed Chinese phonetic Morse code for the people with physical impairments," *Journal of Biomedical Engineering-Applications*, *Basis*, *and Communications*, Vol. 10, 1998, pp. 262-269.
18. C. H. Yang, "Developing a new Chinese Chang Jei Morse code for the disabled per-

sons," *Chinese Journal of Medical and Biological Engineering*, Vol. 18, 1998, pp. 189-194.

19. C. H. Luo and C. H. Shih, "Adaptive Morse-coded single-switch communication system for the disabled," *International Journal of Medical Informatics,* Vol. 41, 1996, pp. 99-106.
20. C. H. Shih and C. H. Luo, "A Morse-Coded recognition system with LMS and matching algorithms for persons with disabilities," *International Journal of Medical Informatics*, Vol. 44, 1997, pp. 193-202.
21. C. H. Yang, "A Chinese phonetic Morse code recognition system for person with physical disability," *Journal of Biomedical Engineering-Applications*, *Basis*, *and Communications*, Vol. 11, 1999, pp. 19-25.
22. C. H. Yang, "A Chinese Chang Jei Morse code recognition system," *Chinese Journal of Medical and Biological Engineering*, Vol. 19, 1999, pp. 203-208.
23. T. W. King, *Assistive Technology*: *Essential Human Factors*, Allyn and Bacon, Boston, 2000.
24. W. J. Conver, *Practical Nonparametric Statistics*, 2nd ed., Wiley & Sons Inc., New York, 1986.

Cheng-Hong Yang (楊正宏) is a full professor of the Electronic Engineering Department of National Kaohsiung University of Applied Sciences. He obtained M.S. degree and Ph.D. degree in Computer Engineering from North Dakota State Univ., in 1988 and 1992, respectively. His main areas of research are evolution- nary computation, bioinformatics, and assistive tool implementation.

Hsiu-Chen Huang (黃秀珍) is the chief of Physical Medicine and Rehabilitation department in Chiayi Christian Hospital. She received M.D. degree from National Yang-Ming Medical College in 1991. Her current research interests are assistive devices/technologies in rehabilitation medicine.

Li-Yeh Chuang (莊麗月) received her M.S. degree from the University of North Carolina at Greensboro, in 1988, and Ph.D. degree from North Dakota State University, in 1994. She is a professor in the department of Chemical Engineering, I-Shou University. Her current research interests are biochemistry, rehabilitation engineering, bioinformatics, and multimedia systems.

Cheng-Huei Yang (楊正輝) received his M.S. degree from the Northeastern University in 1985 and Ph.D. degree in Electrical Engineering from National Cheng Kung University in 2001. Currently, he is an associate professor in the Department of Electronic Communication Engineering, National Kaohsiung Marine University. His current research interests are biomedical instrumentation, mobile communication, and image processing.