Support Vector Machines for Mandarin Phonetic Morse Code Recognition

Cheng-Hong Yang[1], Li-Yeh Chuang[2], Shih-Chung Chen[3], Ching-Hsing Luo[4]

[1]Dept of Electronic Eng., National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan 807
[2]Dept of Chemical Eng., I-Shou University, Kaohsiung, Taiwan 840
[3]Dept of Electrical Eng., Southern Taiwan University of Technology Tainan, Taiwan 701
[4]Dept of Electrical Eng., National Cheng Kung University, Tainan, Taiwan 701

Abstract- *Morse code, with its single switch operation, 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. A major hindrance is difficult to maintain a stable typing rate for persons with severe disabilities. In this paper, a combination of two methods, support vector machines and the variable degree variable step least-mean-square algorithm was used to increase the prediction accuracy for Mandarin Morse code recognition. Experimental findings revealed that the proposed method results in a better recognition rate when compared to alternative methods from the literature.*

Keywords: Mandarin phonetic Morse code, Adaptive signal processing, Least-mean-square algorithm, Support vector machines, Friedman test, Multiple comparison test.

1. Introduction

Many adapted and alternative input methods have been developed to allow users with severe disability to use a computer, including modified direct selections (via mouth stick, head stick, splinted hand, etc.), scanning methods (row-column, linear, circular) and other ways of controlling a sequentially stepping selection cursor in an organized information matrix. Computer input systems, which use Morse code via special software programs, hardware devices, and switches, are invaluable assets in Assistive Technology (AT), Augmentative-Alternative Communication (AAC), rehabilitation, and education. To date, more than 30 manufactures/developers of Morse code input hardware or software for use in AAC and AT have been identified [1].

To aid persons with various neuromuscular diseases, a substitute keyboard is necessary. Morse code, with its single switch operation, has been shown to be valuable in AT, AAC, rehabilitation, and education [2-3]. 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 two methods, support vector machines [4] and the variable degree variable step least-mean-square algorithm [5], was used to increase the prediction accuracy for Mandarin Morse code recognition. The results show a much better recognition rate when compared to previous results in the literature [6-7].

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 with which each character can 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 "-". The tone ratio for dot to dash has to be 1:3 based on the definition of Morse code. That means the duration of a dash is required to be three times that of a dot. Furthermore, the silent ratio for dot-space to character-space also has to be 1:3 [7].

To maintain precise time intervals is a difficult task even for abled persons, not to mention persons with disabilities [7, 8]. In fact, a Morse code time series is generally an unstable one; unstable in speed and/or in rate. In 1996, Luo and Shih [7] proposed a system that could recognize varying typing speeds using an adaptive technique, the least-mean-square algorithm. Their proposed system could adjust its characteristics to successfully recognize a message under unstable typing conditions, but the speed variation of typing had to be limited to a range between 0.67 and two times the present speed. To satisfy this limitation, persons with disabilities had to be well trained; otherwise, the system would not successfully recognize the Morse code message. However, this restriction could not always be complied with by a beginner or by users with serious disabilities. Therefore, the proposed method could not be effectively used. Subsequently, Shih and Luo [8] proposed an improved method that combines the least-mean-square algorithm with a character-by-character matching technique to overcome the limitation. Yang has proposed a statistical method to solve this performance problem, a method consistently providing encouraging results [6].

The proposed method in this paper is divided into five modules: tone recognition, space recognition, learning process, adaptive processing, and character recognition. A block diagram of the Morse code recognition process is shown in Figure 3. Initially, the input data stream is sent individually to either the tone recognition module or the space recognition module, depending on switch-down time (tone element) or switch-up time (space element). In the tone recognition module, the tone element value is first recognized as either a dot or a dash, and then sent to the learning process (support vector machines, SVMs). Simultaneously, 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 tone buffer section. Next, in the space recognition module, 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), and then fed directly into the adaptive processing module. Once a character space is obtained, the value(s) in the tone buffer is (are) sent to the character recognition module, which identifies this character.

Tone Recognition

Initially, each tone_element is treated by normalization to obtain an input value within a range of -1 to 1.

$$\vec{x} = 2.0 * \frac{(tone\_element - 0.5 * (tone\_\max + tone\_\min))}{tone\_\max - tone\_\min}$$

where tone_max and tone_min are the largest and smallest values of the tone element respectively. If a tone_element is larger than the tone_max value, then the tone_max value is substituted by this tone_element value. If a tone_element is smaller than the tone_min value, then the tone_min value is substituted by this tone_element value. The obtained value $\vec{x}$ can be sent into a decision function, f(x), to recognize the value as being either a dash $(f(x) \geq 0)$ or a dot $(f(x) < 0)$. The decision function can be written as:

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

where $\bar{\alpha}_i$ is the solution of the constrained maximization problem, $y_i \in \{-1, +1\}$ and $\bar{b}$ is the bias [9]. The kernel function used is a radial basis function (RBF).

The new tone value of the input stream will be put into the decision function $f(x)$ to determine the value as being either a dash or a dot. Once the decision function is identified, the resulting value can be labeled and sent into the training data set. Then the training process is performed to recalculate the decision function.

Support Vector Machines

Support Vector Machines (SVMs) are based on the theoretical learning theory developed by Vapnik [10, 11]. Support vector machines (SVM) have proven to be highly effective for a number of real world problems, including recognition of handwritten digits, 3-D objects, breast cancer prognosis, and engine-knock detection [12-15]. They demonstrate an impressive resistance to overfitting in classification and their training is performed by maximizing a convex functional, which means that there is a unique solution that can always be found in polynomial time. In the SVMs, original input space is mapped to a high dimensional dot product space called feature space, and in the feature space the optimal hyperplane is determined to maximize the generalization ability. SVMs are designed to automatically trade-off accuracy and complexity by minimizing an upper bound on the generalization error provided by the Vapnik-Chervonenkis (VC) theory [16].

In this study, SVM algorithms are applied to dots or dashes of Morse code recognition. However, training this system is non-trivial and a high cost of computation is required by the use of optimization packages. Kernel-Adatron (KA) algorithms [9], are used to emulate SVM training procedures, which are a based on the Adatron algorithm [17], but adapted by the introduction of kernels so that they can find nonlinear decision boundaries using the high-dimensional feature space. This is a fast and simple learning procedure, which finds a maximal margin hyperplane in a high feature space. Experimental results have shown that the predictive power is equivalent to that of a SVM and the running time can be orders of magnitude faster [9]. The KA procedure is shown below (η = 0.1):

1) Initialize $\alpha_i^0 = 0$.
2) For i=1, …, m execute step 3, 4 below.
3) For a labeled point (x_i, y_i) calculate:

$$z_i = \sum_{j=1}^{m} \alpha_j y_j K(x_i, x_j)$$

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 $(\alpha_i^t + \delta\alpha_i^t) > 0$ then $\alpha_i^t = (\alpha_i^t + \delta\alpha_i^t)$.

5) If a maximum number of iterations is exceeded or the margin λ is approximately 1 then stop, otherwise return to step 2.

$$\lambda = \frac{1}{2}[\min_{\{i|y_i=+1\}}(z_i) - \max_{\{i|y_i=-1\}}(z_i)] \quad (9)$$

Space Recognition

The space recognition module is employed to detect the spaces existing between whole characters, as well as the space between isolated Morse code elements, which comprise a unique character. When data stream of characters composed of Morse code elements is entered, these elements must then be identified as being either dot-dash space (spaces between whole characters) or character space (spaces between isolated elements of a character).

Adaptive Processing

The variable degree variable step least-means-square (VDVSLMS) algorithm used here serves to change the standard 'space' length [5]. The average of space $b_j(x_i)$ (i = 1, n-1) in x_i is the *i*th input data of the algorithm. The VDVSLMS algorithm utilizes the current data to compute a new weight vector using the weight update recursion of the standard LMS algorithm with step size μ. The new weight vector together with the current data are then utilized to update again the desired weight vector using the standard LMS algorithm weight update recursion with step size μ. Each adaptive weight, *W(n)*, is adjusted according to the equation

$$W(n+1) = W(n) - \alpha_2(n)\hat{\nabla}(n) \quad (4)$$

where

$$\alpha_2(n) = 2\mu(1 - \mu X^T(n)X(n)) \quad (5)$$

The subscript on the α (n) is used to indicate the degree, and

$$\hat{\nabla}(n) = -2\varepsilon(n)X(n) \quad (6)$$

is an estimate of the gradient.

$$\varepsilon(n) = d(n) - X^T(n)W(n) \quad (7)$$

where *d(n)* is the scalar 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 [6].

Character Recognition

Once a character space value has arrived in the tone_buffer, the elements in the tone buffer have to be sent to the character recognition process. 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 tone_base 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 the shortest Euclidean distance method is the following. First, each tone element, $e_j(x_i)$, is divided by the tone_base. Then, the roots of the sum of the square distances between the new tone element 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. Results and Discussion

Technologically assistive devices are gradually playing more important roles in the lives of people with disabilities, with one of the more promising considerations being a combination of the functions of computer software and hardware. 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 [1].

We have evaluated the performance of the proposed method on testing data from two test participants with disabilities. Participant 1 (P1), a 14-year-old female adolescent, is a victim of cerebral palsy, athetoid type, with involuntary movements of her four limbs. The involuntary motion is increased when she is excited and efforts for voluntary movement. Participant 2 (P2), a 40-year-old male adult, is a victim of a spinal cord injury with incomplete quadriparesis. His right wrist cannot extend purposefully and each finger cannot individually move, which results in dysfunction of hand movement. The sensation of both upper extremities is well preserved, and his mental and verbal communication skills are as good as before the accident. Each participant typed 100 phonetic characters in 15 test samples respectively, numbered from Dis01 to Dis15 in Table 1.

Table 1 illustrated that the proposed method consistently produced the highest number of matches for phonetic Morse code recognition. The average number of matches for SL, Yang, and the proposed method were 70.67, 85.56, and 88.17 for Mandarin phonetic Morse code recognition. These results can be explained as follows. A Morse code time series is generally an unstable one, unstable in speed and/or in rate. An unstable typing speed or rate can generate two kinds of errors: a space recognition error and a tone recognition error. Maintaining precise intervals is a difficult task. 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. After an initial dot-dash classifier is obtained from the initial training data, the testing data is sent to the training process in order to obtain an adjusted decision function, which in turn increases the system's recognition rate.

Following the multiple comparison test given in Conover [18], the total matches of the three methods were ordered in an array, and a rank was assigned to each corresponding to its order. 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 α = .05), they may be regarded as having unequal medians. Therefore, it can be concluded that the method proposed here is statistically superior to both the Yang and SL's methods.

Table 1. Number of correct Mandarin phonetic character matches (out of 100) using the 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	97	74.5	94.0	95.0
Dis02	66	78	83	81	90	91	73.5	84.0	87.0
Dis03	75	87	91	81	94	95	78.0	90.5	93.0
Dis04	70	89	90	73	93	93	71.5	91.0	91.5
Dis05	71	88	92	67	74	75	69.0	81.0	83.5
Dis06	71	89	90	75	95	96	73.0	92.0	93.0
Dis07	73	83	87	74	88	89	73.5	85.5	88.0
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	92	74.5	85.0	91.5
Dis11	77	77	90	70	94	94	73.5	85.5	92.0
Dis12	73	79	92	82	95	95	77.5	87.0	93.5
Dis13	66	84	87	75	94	94	70.5	89.5	90.5
Dis14	62	88	90	32	44	47	47.0	66.0	68.5
Dis15	68	90	92	32	45	47	50.0	67.5	69.5
Average	70.87	86.07	90.13	70.47	85.07	86.20	70.67	85.56	88.17

Legends:
SL: Shih and Luo's method [14].
Yang: Yang's method [11].
NEW: the proposed method.

4. Conclusions

Morse code has been shown to be an attractive candidate as an adaptive tool for some people with various neuromuscular diseases such as amyotrophic lateral sclerosis, multiple sclerosis, and muscular dystrophy, when using computer technology to communicate with others. Due to the fact that maintaining a stable typing rate is a challenge, an improved method with a higher character recognition rate is desperately needed. In this paper, we experimented with support vector machines and the variable degree variable step least-mean-square algorithm for the adaptive Mandarin phonetic Morse code recognition. Experimental results revealed that the recognition rate of the method proposed here was superior to alternative methods from the literature.

ACKNOWLEDGEMENTS

This research was supported by the National Science Council, R.O.C., under grant NSC 90-2218-E-151-005.

References

[1] T.W. King, Modern Morse Code in Rehabilitation and Education, MA: Allyn and Bacon, 2000.

[2] R. Bower, *et al.* (Eds), The Trace Resourcebook-Assistive Technology for Communication, Control, and Computer Access, Trace Research & Development Center, Universities of Wisconsin-Madison, Waisman Center, 1998.

[3] D.K. Anson, Alternative Computer Access: A Guide to Selection, F.A. Davis, Philadelphia, 1997.

[4] Frieß, T., N. Cristianini, and C. Campbell (1998), "The Kernel-Adatron: a Fast and Simple Learning Procedure for Support Vector Machines," *Proceedings of the Fifteenth International Conference on Machine Learning*, pp. 188-196, 1998.

[5] M.A. Khasawneh and K.A. Mayyas, A Newly Derived Variable Degree Variable Step Size LMS Algorithm, Int. J. Electronics, Vol. 79, No. 3, pp. 255-264, 1995.

[6] Yang, Cheng-Hong (1999), A Chinese Phonetic Morse Code Recognition System for Person with Physical Disability, Journal of Biomedical Engineering-Applications, Basis, and Communications, Vol. 11, No. 1, 19-25.

[7] C.H. Luo and C.H. Shih, Adaptive Morse-coded single-switch communication system for the disabled, International Journal of Bio-Medical Computing, Vol. 41, 1996, pp. 99-106.

[8] C.–H. Shih and C.-H. Luo, A Morse-Coded recognition system with LMS and matching algorithms for persons with disabilities, Int. J. of Medical Informatics 44, pp. 193-202, 1997.

[9] Frieß, T., N. Cristianini, and C. Campbell (1998), "The Kernel-Adatron: a Fast and Simple Learning Procedure for Support Vector Machines," *Proceedings of the Fifteenth International Conference on Machine Learning*, pp. 188-196, 1998.

[10] Boser, B., Guyon, I., Vapnik, V. (1992), A training algorithm for optimal margin classifiers, Fifth Annual Workshop on Computational Learning Theory, ACM Press.

[11] Cortes, C., and Vapnik, V. (1995), Support Vector networks, Machine Learning 20:273-297.

[12] Friess, T., Cristianini, N., and Campell, C., 1998, The kernel-adatron algorithm: A fast and simple learning procedure for support vector machines, In proceedings of the fifteenth International Conference on Machine Learning, ICML, 180-196.

[13] Roobaert, D. 1999, Improving the generalization of support vector machines: An application to 3D object recognition with cluttered background. In proceedings of the Workshop on SV Machines at IJCAI, 29-34.

[14] Rychetsky, R., Ortmann, S., and Glesner, M., 1999, Support vector approaches for enigner knock detection, In international Joint Conference on Neural Networks.

[15] Scholkopf, B., 1997, Support vector Learning, Ph.D. Dissertation, Max-Planck-Institut fur biologische Kybernetik.

[16] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer Verlag, 1995.

[17] Anlauf, J.K., and Biehl, M. (1989), The Adatron-an adaptive perceptron algorithm, Europhysics Letters, 10, 687-692.

[18] W.J. Conover, Practical Nonparametric Statistics, 2nd ed. Wiley & Sons Inc., New York, 1986.