Foot Gait Time Series Estimation Based on Support Vector Machine

Jeevan K. Pant and Sridhar Krishnan[1]

***Abstract*— A new algorithm for the estimation of stride interval time series from foot gait signals is proposed. The algorithm is based on the detection of beginning of heel strikes in the signal by using the support vector machine. Morphological operations are used to enhance the accuracy of detection. By taking backward differences of the detected beginning of heel strikes, stride interval time series is estimated. Simulation results are presented which shows that the proposed algorithm yields fairly accurate estimation of stride interval time series where estimation error for mean and standard deviation of the time series is of the order of 10^{-4}.**

I. INTRODUCTION

Wearable and portable medical sensors can be used for remote monitoring of human beings over a long period of time [1][2]. Such extended period monitoring can produce large-size signals from a large number of people, resulting in massive amount of data. An analysis of such data would require huge computational effort which would be beyond the capability of available computing resources. Time series can be used to represent signal features, such as RR intervals in electrocardiogram signals and stride interval (SI) (and also stride frequency) in gait signals, by significantly reducing the signal dimension [3][4][5]. Such a time series can be estimated by detecting the indices corresponding to a suitable pattern in the signal. Support vector machine (SVM) and neural networks are two popular supervised machine learning techniques used for pattern recognition [6][7][8][9][10].

We present a new algorithm for the estimation of SI time series (SI-TS) of foot gait signals. The algorithm is based on estimating the beginning of heel strike (BoHS) instances on the signal. Several foot-gait signals and their indices corresponding to the BoHS instances are assumed to be available for training the SVM. The trained SVM is used for classifying the components of a new signal in two classes based on their proximity to the BoHS instances. Resulting class labels are represented as a vector of binary valued components. Morphological operations are applied to the resulting vector to mitigate the effect of classification error. BoHS instances are determined from the resulting vector, and then SI-TS is computed by taking backward difference of the BoHS instances sorted in the ascending order.

II. BACKGROUND

A. Support vector machines (SVMs)

Let $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$ be m vectors each representing a point in a N dimensional space. Suppose these m points appear as two clusters which are separable by a linear hyperplane. The vectors associated with a cluster can be labelled with -1 and that associated with the next cluster can be labelled with $+1$. If the label of the cluster associated with the ith vector is denoted as $y_i \in \{-1, 1\}$, then the hyperplane separating the two clusters can be obtained as

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i^* y_i \boldsymbol{x}_i^T \boldsymbol{x} + b \quad (1)$$

where α_i^* is the ith component of the vector $\boldsymbol{\alpha}$ obtained by solving the optimization problem [10] [11]

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}}{\text{maximize}} & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j \\ \text{subject to:} & \sum_{i=1}^{m} \alpha_i y_i = 0, \\ & 0 \le \alpha_i \le C \text{ for } i = 1, \ldots, N \end{array} . \quad (2)$$

Typically, only a few components of $\boldsymbol{\alpha}$ are nonzero, and a nonzero α_i implies that the corresponding vector $\boldsymbol{x}_i$ is a *support vector*. Parameter b can be determined as

$$b = \sum_{j=1}^{m} \alpha_j y_j \boldsymbol{x}_j^T \boldsymbol{x}_i - y_i \quad (3)$$

where $\boldsymbol{x}_i$ is any one support vector and y_i is the corresponding label [12].

The product $\boldsymbol{x}_i^T \boldsymbol{x}$ in (1) can be replaced by the *radial basis function* kernel given by

$$k(\boldsymbol{x}_i, \boldsymbol{x}) = \exp\left(-\frac{||\boldsymbol{x}_i - \boldsymbol{x}||^2}{2\sigma^2}\right), \quad (4)$$

where σ is a parameter. This technique is called as a *kernel trick* which performs an implicit mapping of vectors $\boldsymbol{x}$ and $\boldsymbol{x}_i$ to a higher dimensional space for better classification performance. As a result of the kernel trick, function $f(\boldsymbol{x})$ in (1) becomes

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i^* y_i k(\boldsymbol{x}_i, \boldsymbol{x}) + b. \quad (5)$$

The label of the cluster associated to a new vector $\boldsymbol{x}_k$ can be determined as $\texttt{sign}[f(\boldsymbol{x}_k)]$.

III. TIME SERIES ESTIMATION

A. Foot Gait Signal

A foot-gait signal obtained by measuring pressure under the sole of a walking person is shown in Fig. 1(upper panel). The signal resembles a sequence of rectangular pulses where each pulse corresponds to a SI. A SI in Fig. 1(upper panel) can be divided into four regions R_1, R_2, R_3, and R_4 (see Fig. 1(lower panel)) which correspond to (i) heel strike, i.e.,

[1]J. K. Pant and S. Krishnan are with Department of Electrical and Computer Engineering, Ryerson University, 350 Victoria Street, Toronto, ON M5B 2K3, CANADA `jpant@rnet.ryerson.ca`, `krishnan@ee.ryerson.ca`

when heel touches the floor, (ii) mid stance, i.e., when foot remains flat on the floor directly supporting the weight of the body, (iii) toe off, i.e., when the foot begins to leave the floor, and (iv) swing, i.e., when foot remains in the air, respectively. SI at the BoHS can be measured as the time elapsed from the BoHS to the next BoHS. A vector whose successive components are SIs at successive BoHSs is called as the SI time series (SI-TS) vector.

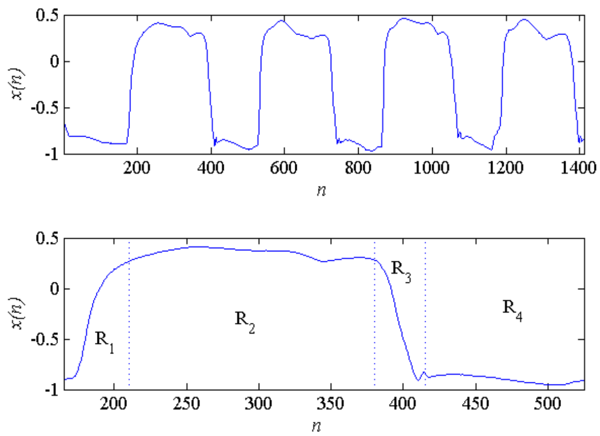

Fig. 1. A foot gait signal (upper panel) and four regions, R_1, R_2, R_3 and R_4 of a pulse (lower panel).

B. Estimation of SI-TS

1) Construction of training data: Consider a vector $\boldsymbol{x}$ of length N representing a foot gait signal and a vector $\boldsymbol{u}$ of length T whose components are the indices of $\boldsymbol{x}$ corresponding to its BoHSs, called as the BoHS indices. Let d be a positive integer much smaller than the typical number of samples in SIs in signal $\boldsymbol{x}$. Total $N-2d$ sub-vectors of $\boldsymbol{x}$ each of length $2d+1$ can be constructed as

$$\tilde{\boldsymbol{x}}_i = [x_i \ x_{i+1} \ \cdots \ x_{i+2d}]^T$$

for $i = 1, 2, \ldots, N-2d$.

Let c be a positive integer much smaller than d and u_i be the ith component of $\boldsymbol{u}$. We assume that the BoHS of $\boldsymbol{x}$ at its index u_i, called as the u_ith BoHS of $\boldsymbol{x}$, can be characterized in terms of total $2c+1$ sub-vectors, namely, $\{\tilde{\boldsymbol{x}}_{u_i-c}, \tilde{\boldsymbol{x}}_{u_i-c+1}, \ldots, \tilde{\boldsymbol{x}}_{u_i+c}\}$. With $c \ll d$, it can be fairly assumed that the components at the middle of these sub-vectors have greater influence on the characterization of the BoHS relative to the peripheral components. Dependency of the u_ith BoHS on the middle components of the sub-vectors $\{\tilde{\boldsymbol{x}}_i, i = u_i - c, \ldots, u_i + c\}$ can be enhanced relative to the peripheral components, and also the dimension of the sub-vectors $\tilde{\boldsymbol{x}}_i$ can be reduced, by using the procedure described below.

First, let N_1 and N_2 be two small positive integers so that $N_1 \ll d$, $N_2 \ll d$, and $2d+1$ is divisible by $N_1 \times N_2$. Divide vector $\tilde{\boldsymbol{x}}_i$ into total $(2d+1)/N_1$ blocks each of length N_1, compute average value of the components for each block, and construct a vector $\hat{\boldsymbol{x}}_i$ of length $(2d+1)/N_1$ by setting the computed average values as its components. Next, compute the number of peripheral components of $\hat{\boldsymbol{x}}_i$, denoted as t_{n_1}, using

$$t_{n_1} = \text{round}\left[(0.5-t)(2d+1)/N_1N_2\right] N_2, \tag{6}$$

where $0 < t < 0.5$, and construct three sub-vectors $\hat{\boldsymbol{x}}_i^1$, $\hat{\boldsymbol{x}}_i^2$, $\hat{\boldsymbol{x}}_i^3$ consisting of the first t_{n_1} components, last t_{n_1} components, and rest of the components, respectively, of $\hat{\boldsymbol{x}}_i$. After that, divide the sub-vectors $\hat{\boldsymbol{x}}_i^1$ and $\hat{\boldsymbol{x}}_i^2$ each into total t_{n_1}/N_2 blocks each of length N_2, compute average value of the components for each block, and construct smaller sub-vectors $\hat{\boldsymbol{x}}_i^{1s}$ and $\hat{\boldsymbol{x}}_i^{2s}$ using the computed average values as their components. Finally, construct vector $\breve{\boldsymbol{x}}_i$ as

$$\breve{\boldsymbol{x}}_i = \left[\left(\hat{\boldsymbol{x}}_i^{1s}\right)^T \left(\hat{\boldsymbol{x}}_i^{3}\right)^T \left(\hat{\boldsymbol{x}}_i^{2s}\right)^T\right]^T. \tag{7}$$

Vector $\breve{\boldsymbol{x}}_i$ can be used as a feature vector of $\tilde{\boldsymbol{x}}_i$, and it can be labelled as either $+1$ or -1 based on whether the $(d+1)$th component of $\tilde{\boldsymbol{x}}_i$ corresponds to the index range $u_i - c, u_i - c + 1, \ldots, u_i + c$ of $\boldsymbol{x}$ or not. In other words, a label y_i for the ith feature vector $\breve{\boldsymbol{x}}_i$ can be set as

$$y_i = \begin{cases} +1 & \text{if } \begin{cases} u_j - c \le i + d \le u_j + c \\ \text{for any } j \text{ over } 1 \le j \le T \end{cases} \\ -1 & \text{otherwise} \end{cases}.$$

for $i = 1, 2, \ldots, N-2d$.

Using $\{\breve{\boldsymbol{x}}_i\}$ and $\{y_i\}$, a training matrix $\mathbf{T}_r$ and corresponding group vector $\boldsymbol{g}_r$ can be constructed as

$$\mathbf{T}_r = [\breve{\boldsymbol{x}}_1 \ \breve{\boldsymbol{x}}_2 \ \cdots \ \breve{\boldsymbol{x}}_{N-2d}]^T \tag{8a}$$

$$\boldsymbol{g}_r = [y_1 \ y_2 \ \cdots \ y_{N-2d}]^T. \tag{8b}$$

When more than one, say, total N_{tr}, training signals $\{\boldsymbol{x}_i\}$ and corresponding BoHS vectors $\{\boldsymbol{u}_i\}$ are available, then the training matrix and group vector can be constructed as

$$\mathbf{T}_r = \left[\left(\mathbf{T}_r^1\right)^T \ \left(\mathbf{T}_r^2\right)^T \ \cdots \ \left(\mathbf{T}_r^{N_{tr}}\right)^T\right]^T \tag{9a}$$

$$\boldsymbol{g}_r = \left[\left(\boldsymbol{g}_r^1\right)^T \ \left(\boldsymbol{g}_r^2\right)^T \ \cdots \ \left(\boldsymbol{g}_r^{N_{tr}}\right)^T\right]^T \tag{9b}$$

where matrix $\mathbf{T}_r^i$ and vector $\boldsymbol{g}_r^i$ are obtained as (8) using the ith training signal $\boldsymbol{x}_i$ and the ith BoHS vector $\boldsymbol{u}_i$.

In practice, the size of training data data $\{\mathbf{T}_r, \boldsymbol{g}_r\}$ can be reduced by removing rows from $\mathbf{T}_r$ and $\boldsymbol{g}_r$ so that the number of $+1$ and the number of -1 in $\boldsymbol{g}_r$ remain approximately equal.

2) Estimation of SI-TS vector using SVM: Suppose we want to find SI-TS for a signal $\boldsymbol{x}_k$ of length N. A test matrix $\mathbf{T}_s$ for SVM can be constructed from $\boldsymbol{x}_k$ by using the procedure used to construct $\mathbf{T}_r$ in (8a). Estimation of SI-TS for $\boldsymbol{x}_k$ involves performing binary classification to determine the labels, -1 or 1, associated with the rows of $\mathbf{T}_s$. SVM is a binary classification technique where an appropriate training algorithm, such as that available from [13], [14], or `svmtrain` function in the statistics toolbox of MATLAB, can be applied on the training data $\{\mathbf{T}_r, \boldsymbol{g}_r\}$. Moreover, the feature vectors $\{\breve{\boldsymbol{x}}_i\}$ constructed using (7) from a gait foot signal can be expected to be non-separable. The kernel trick used in SVM has been found to be effective for the classification of such non-separable vectors.

Function `svmtrain` yields various SVM parameters including support vectors, α^*, and b. These parameters can be supplied to the SVM classification function in the statistical toolbox called as `svmclassify` which is based on the realization of equations (3), (4), and (5). This function can be expected to yield an estimated label, 1 or -1, for each row of $\mathbf{T}_s$. Let $\boldsymbol{g}_s$ be a vector whose components are the estimated labels for the $N-2d$ rows of $\mathbf{T}_s$. A SI-TS vector $\hat{\boldsymbol{u}}$ can be estimated from $\boldsymbol{g}_s$ using the procedure described below. Components of $\boldsymbol{g}_s$ corresponding to the indices of $\boldsymbol{x}_k$ at the BoHSs and their neighbourhood would be expected to be of value 1. The value for the rest of the components of $\boldsymbol{g}_s$ can be expected to be -1. For ease of exposition, we threshold the ith component of $\boldsymbol{g}_s$ as

$$g_{si} = \text{maximum}\{g_{si}, 0\} \tag{10}$$

for $i = 1, 2, \ldots, 2N-d$. The resulting vector $\boldsymbol{g}_s$ can be expected to contain a rectangular pulse with unit height centred at each index which corresponds to a BoHS index of $\boldsymbol{x}$. However, possible classification error by `svmclassify` may split the rectangular pulses into two or more pulses of smaller widths. Also, undesired rectangular pulses of small widths may appear nearby the locations of the desired rectangular pulses. These problems can be overcome by applying two variants of the morphological operations described in [15], which are detailed below.

First, apply a dilation operation on $\boldsymbol{g}_s$ where the ith component of the improved vector $\tilde{\boldsymbol{g}}_s$ is obtained as

$$\tilde{g}_{s1} = g_{s1}, \quad \tilde{g}_{s(2N-d)} = g_{s(2N-d)}, \tag{11a}$$

and

$$\tilde{g}_{si} = \text{maximum}\ \{g_{s(i-1)}, g_{si}, g_{s(i+1)}\} \tag{11b}$$

for $1 < i < 2N-d$. Then, set $\boldsymbol{g}_s = \tilde{\boldsymbol{g}}_s$ and repeat the operation in (11). Continue this process for, say, N_{dl} times, and finally, set $\boldsymbol{g}_s = \tilde{\boldsymbol{g}}_s$.

Next, apply an erosion operation on $\boldsymbol{g}_s$ obtained after dilation where the first and the last components of the improved vector $\tilde{\boldsymbol{g}}_s$ are set as in (11a). The other components are obtained as

$$\tilde{g}_{si} = \text{maximum}\ \{\tilde{g}^1_{si}, \tilde{g}^2_{si}\} \tag{12}$$

for $1 < i < 2N-d$, where

$$\begin{aligned} \tilde{g}^1_{si} &= \text{minimum}\ \{g_{s(i-1)}, g_{si}, g_{s(i+1)}\} \\ \tilde{g}^2_{si} &= \text{minimum}\ \{1-\tilde{g}_{s(i-1)}, g_{si}, 1-g_{s(i+1)}\} \end{aligned}.$$

Then, set $\boldsymbol{g}_s = \tilde{\boldsymbol{g}}_s$ and repeat the above operation. Continue this process for, say, N_{en} times, and finally, set $\boldsymbol{g}_s = \tilde{\boldsymbol{g}}_s$. With a sufficiently large value of N_{en}, this operation would shrink each rectangular pulse in $\boldsymbol{g}_s$ to a discrete-time unit impulse. Hence, the resulting $\boldsymbol{g}_s$ can be expected to be sparse vector with nonzero components at the indices corresponding to the BoHS indices of $\boldsymbol{x}_k$.

Finally, SI-TS vector for $\boldsymbol{x}_k$ can be computed as follows. First, construct a vector $\tilde{\boldsymbol{u}}$ whose components are the indices of the nonzero components of $\boldsymbol{g}_s$ in the increasing order of their values. If $\tilde{\boldsymbol{u}}$ has $N_{ts}+1$ components, then construct

$$\hat{\boldsymbol{u}} = [\hat{u}_1\ \hat{u}_2\ \cdots\ \hat{u}_{N_{ts}}]^T \tag{13}$$

where $\hat{u}_i = (\tilde{u}_{i+1} - \tilde{u}_i)/f_s$ for $i = 1, 2, \ldots, N_{ts}$ and f_s is the frequency used to acquire the signal $\boldsymbol{x}_k$.

A flow chart of the procedure described above is shown in Fig. 2.

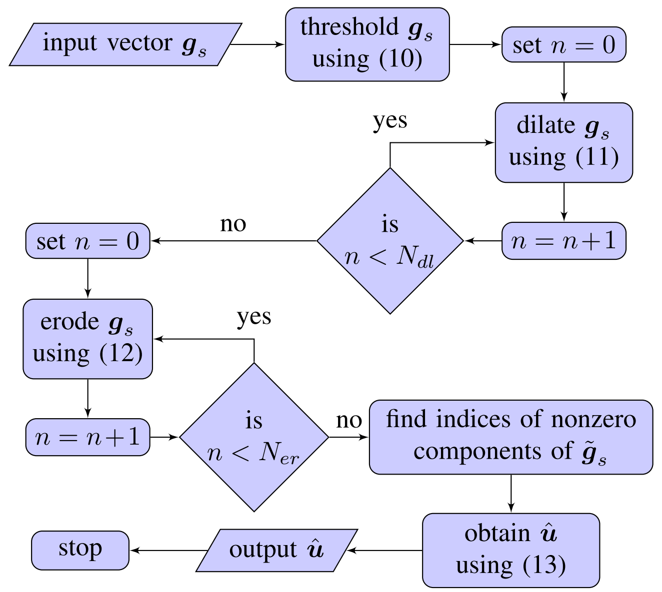

Fig. 2. Flow chart of the procedure for the estimation of SI-TS vector $\hat{\boldsymbol{u}}$ from vector $\boldsymbol{g}_s$

The procedure for the estimation of SI-TS vector $\hat{\boldsymbol{u}}$ from signal $\boldsymbol{x}_k$ can be summarized as an algorithm shown in Table I. In Step 1 of the algorithm, $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{u}_i\}$ denote training signals and corresponding BoHS indices, respectively. Parameter t in Step 1 is used for the evaluation of equation in (6).

TABLE I
SI-TS Estimation Algorithm

Step 1 Input $\boldsymbol{x}_k$, $\{\boldsymbol{x}_i\}$, $\{\boldsymbol{u}_i\}$, d, c, N_1, N_2, N_{dl}, N_{en}, t, f_s.
Step 2 Construct $\mathbf{T}_r$ and $\boldsymbol{g}_r$ from $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{u}_i\}$ using (9).
Step 3 Using $\mathbf{X}$ and $\boldsymbol{y}$ in `svmtrain`, obtain SVM parameters.
Step 4 Construct a testing matrix $\mathbf{T}_s$ from $\boldsymbol{x}_k$ using (8a).
Step 5 Obtain vector $\boldsymbol{g}_s$ from $\mathbf{T}_s$ using `svmclassify`.
Step 6 Obtain $\hat{\boldsymbol{u}}$ using (10)-(13).
Step 7 Output $\hat{\boldsymbol{u}}$ and stop.

IV. Simulation Results

Sixteen records of foot gait signals obtained from healthy control subjects and corresponding time series data were downloaded from the Physionet website [16]. As reported in [16], each signal was originally acquired with the sampling rate of $f_s = 300$ Hz over 600 seconds. In the first simulation, the algorithm in Table I was run with $d = 200$, $c = 40$, $N_1 = 5$, $N_2 = 5$, $N_{dl} = 40$, $N_{en} = 360$, $t = 0.06$, and $f_s = 300$. Seven records were randomly chosen from the total sixteen records, and left foot signals from the seven records were selected as $\{\boldsymbol{x}_i\}$. From the corresponding seven time series files, elapsed time (column 1) were selected as seven vectors $\{\boldsymbol{u}_i\}$. The resulting $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{u}_i\}$ were used as training data in Step 1 of the algorithm. A record was randomly chosen from among the sixteen records and its left foot signal was used as $\boldsymbol{x}_k$ in Step 1. The algorithm was run several times. A typical original SI-TS signal (column 2 of the time series file) and the SI-TS signal estimated from $\boldsymbol{x}_k$ are shown in Fig. 3. As can be seen, the estimated SI-TS signal is very similar to the original SI-TS signal.

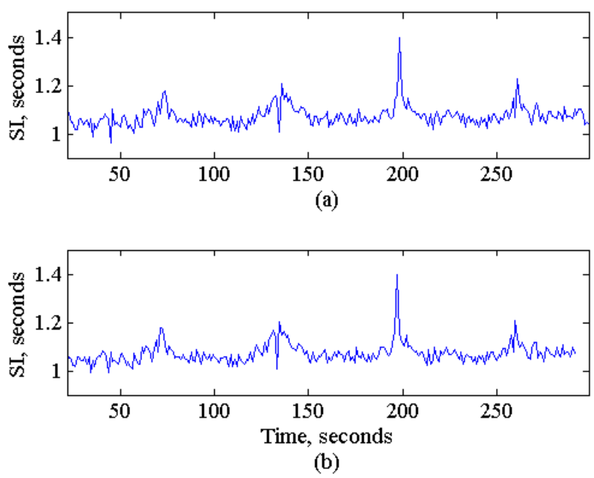

Fig. 3. (a) Original left foot SI-TS (`Control1`) and (b) estimated left foot SI-TS (`Control1`). SVM trained with record numbers 9, 13, 1, 5, 15, 8 and 7.

In the second simulation, the left foot gait signal from the first record, i.e., `Control1`, was set to $\boldsymbol{x}_k$ and the algorithm in Table I was run as in the first simulation. This process was repeated for fifteen times by setting the left foot gait signals from the remaining fifteen records, namely, `Control2`, `Control3`, ..., `Control16`, to $\boldsymbol{x}_k$. The values of mean and standard deviation of the estimated SI-TS signals in a typical case are shown in the second and fourth columns of Table II. Absolute error between the values of mean and standard deviation for the original SI-TS and estimated SI-TS signals are shown in the third and fifth columns, respectively. As can be seen, the error for the mean and standard deviation is of the order of 10^{-4}.

V. CONCLUSIONS

A new algorithm for the estimation of SI-TS vector from foot gait signals is proposed. The algorithm is based on the recognition of the beginning of heel strike in the signal by using SVM and morphological operations. As demonstrated using simulation results, the proposed algorithm yields estimated SI-TS signals which are fairly accurate in terms of mean and standard deviation.

ACKNOWLEDGMENT

The authors would like to thank the Canada Research Chairs program for supporting this research.

TABLE II

MEAN AND STANDARD DEVIATION FOR ESTIMATED SI-TS

Record	Mean		Standard deviation	
	Value	Error $\times 10^{-4}$	Value $\times 10^{-3}$	Error $\times 10^{-4}$
`Control1`	1.0723	2	1	1.80
`Control2`	1.1566	9	3	1.13
`Control3`	1.0907	4	4	3.31
`Control4`	1.0409	2	3	0.34
`Control5`	1.1084	4	2	6.52
`Control6`	1.0297	7	7	1.47
`Control7`	1.0681	4	4	0.26
`Control8`	1.0648	10	10	0.35
`Control9`	1.0124	1.2	12	0.21
`Control10`	1.0025	1.3	13	1.45
`Control11`	1.0353	6	6	2.58
`Control12`	1.1433	12	11	0.30
`Control13`	1.1082	4	4	0.46
`Control14`	1.1160	15	19	3.61
`Control15`	1.4001	5	15	7.77
`Control16`	1.1120	0	1	2.25

REFERENCES

[1] B. Mariani, S. Rochat, C. J. Bula, and K. Aminian, "Heel and toe clearance estimation for gait analysis using wireless inertial sensors," *IEEE Trans. Biomed. Engg*, vol. 59, no. 11, pp. 3162-3168, Nov. 2012.

[2] C. B. Redd and S. J. M. Bamberg, "A wireless sensory feedback device for real-time gait feedback and training," *IEEE Trans. Mechantronics*, vol. 17, no. 3, pp. 425-433, June 2012.

[3] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Engg*, vol. BME-32, no. 3, pp. 230-236, Mar. 1985.

[4] V. P. Stokes, A. Thorstensson, and H. Lanshammar, "From stride period to stride frequency," *Gait Posture*, vol. 7, no. 1, pp. 35-38, 1998.

[5] T. Chau and K. Parker, "On the robustness of stride frequency estimation," *IEEE Trans. Biomed. Engg*, vol. 51, no. 2, pp. 294-303, Feb. 2004.

[6] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273-279, 1995.

[7] C. M. Bishop, *Neural networks for pattern recognition,* Clarendon press. Oxford, 1995.

[8] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121-167, 1998.

[9] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: a review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4-37, Jan. 2000.

[10] A. Shilton, M. Palaniswami, D. Ralph, and A. C. Tsoi, "Incremental training of support vector machines," *IEEE Trans. Neural Networks*, vol. 16, no. 1, pp. 114-131, Jan. 2005.

[11] K.-R. M´uller, S. Mika, G. R´atsch, K. Tsuda, and B. Sch´olkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 181-201, Mar. 2001.

[12] M. Welling, "Classnotes in machine learning," `http://www.ics.uci.edu/~welling/classnotes/papers_class/SVM.pdf`, [Online, Accessed Mar. 2014].

[13] LIBSVM - A library for support vector machines `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`, [Online, Accessed Feb. 2014].

[14] SVM Matlab `http://cns.bu.edu/~gsc/CN710/pmwiki.php?n=Main.SVMMatlab`, [Online, Accessed Feb. 2014].

[15] C.-H. H. Chu and E. J. Delp, "Impulsive noise suppression and background normalization of electrocardiogram signals using morphological operators," *IEEE Trans. Biomed. Engg*, vol. 36, no. 2, pp. 262-273, Feb. 1989.

[16] Physionet: gait dynamics in neuro-degenerative disease data base `http://www.physionet.org/physiobank/database/gaitndd/`, [Online, Accessed Feb. 2014].