Least Squares Support Vector Machines for Clustering in Wireless Sensor Networks

Jerzy Martyna[1]

Abstract: We introduce a new approach to clustering in wireless sensor networks (WSNs). The clustering problem is viewed as a classification problem. We map the output of the Least Squares Support Vector Machine (LS-SVM) to probability and present a scheme for estimating clusters obtained from it. We show that the application of the LS-SVM gives a good estimate of a cluster formation. Through the use of the mixtures of kernels we have obtained better results than with one kernel. A computer experiment involving the sensor clustering is being carried out.

Keywords: wireless sensor networks, least squares support vector machines

1. Introduction

In a wireless sensor network (WSN) (see Akyildiz, 2002), the sensor nodes sense information and send it to selected sensors called clusterhead sensor nodes. The clusterhead sensor nodes are responsible for collecting data from the environment and sending them to the sink. Given the constrained radio transmission range of the sensor and the need of conserving energy, the clusterhead sensors need to be located as close to the sensors as possible. By means of the clustering of sensor nodes in the sensor field, we prevent large amounts of packet transmission and save energy power.

The main the objective of clustering scheme is to generate energy-efficient clusters for randomly deployed sensor nodes where each cluster is managed by a clusterhead sensor node. A key determinant of the effectiveness of WSNs is their longevity, which is limited by the energy that is stored in each sensor. Therefore, the clustering scheme which uses it for acquisition, processing and communication must be as energy-efficient as possible.

We recall that the clustering has been proposed by D.J. Baker et al. (1981) to form a cluster. This approach allows us to manage the cluster and relay the collected data. In the paper by Heinzelman et al. (2002) a clustering-based approach, called LEACH was proposed. In this approach all clusterhead sensors communicate directly with the base station whereas other sensor nodes forward their data through the clusterheads. More recently, in Tang and Li (2006) a clustering approach has been proposed for QoS supporting and an optimal energy allocation. In the paper by Krishnan et al. (2006) some algorithms which involve the so called

[1] Institute of Computer Science, Jagiellonian University, ul. Nawojki 11, 30-072 Cracow
e-mail: martyna@softlab.ii.uj.edu.pl

local "growth budgets" to neighbours were proposed. In this methodology, an upper bound on the expected time for network decompostion was derived. Nevertheless, none of these algorithms aim at minimizing the energy during the formation of the clusters. The primary goal of these algorithms is to minimize the number of clusters so that a sensor node in any cluster is at most d hops away from a clusterhead.

Based on a statistical learning theory as a new tool for data classification, feature selection and function estimation, the support vector machines (SVM) have been introduced by Cortes and Vapnik (1995) and Vapnik (1998a,1998b). Briefly, the SVM method maps input data into a high dimensional hypersurface where it may become linearly separable. Thus, the SVM is used to minimize the structural task whereas the previous techniques are based on the minimization of the number of misclassified points on the training set. Among others, the SVM were used in many pattern recognition problems (see Mitéran, 2003), feature selection (see Barzilay, 1999), face authentication (see Jonsson, 1999). A modified version of the SVM called least squares SVM (LS-SVM) was proposed by Suykens and Vandewolle (1999) and Suykens et al. (2002). In this approach a set of liner equations instead of a quadratic programming problem is used. Recently, many solutions with the SVM methods are being implemented in low-cost VLSI chips.

The main goal of this paper is to introduce a novel method of clustering sensor nodes in WSNs. The proposed method is based upon an LS-SVM formulation with a mixture of a radial basis function and polynomial kernels. The two-dimensional sensor field is mapped into the sensor energy intensity surface, and then the sensor clustering process is efficiently realized. The mixture of kernels gives a better performance than any single kernel. A number of numerical case studies indicate the usefulness of this approach.

The rest of the paper is organized as follows. In section 2, we describe the clustering problem in a wireless sensor network. Section 3 introduces a support vector machine for problem solving. In section 4, we give some computational results. Section 5 concludes the paper.

2. Clustering in Wireless Sensor Networks

This section describes the clustering problem in wireless sensor networks (WSNs).

A hierarchical approach leads to clustered layers as shown in Fig. 1. All ordinary sensor nodes are grouped into clusters with a clusterhead sensor node that is responsible for data aggregation and sends them to the sink over the backbone formed by clusterhead sensors belonging to higher-level of sensor nodes in the WSN. In a cluster-based hierarchy the data are moving to the sink faster than in the multi-hop model. It is caused by fact that in the cluster-based model only clusterhead sensor nodes perform data aggregation whereas in the multi-hop model each intermediate node realizes data aggregation. However, this model has one drawback: as the distance between clustering levels increases, the energy expenditure is proportional to the square (in the air) of the distance.

We assume that some sensor nodes have more powerful energy resources than the ordinary sensor nodes. We assume that they initiate the clustering process and possible some of them would be clusterhead sensor nodes. The clusterheads should

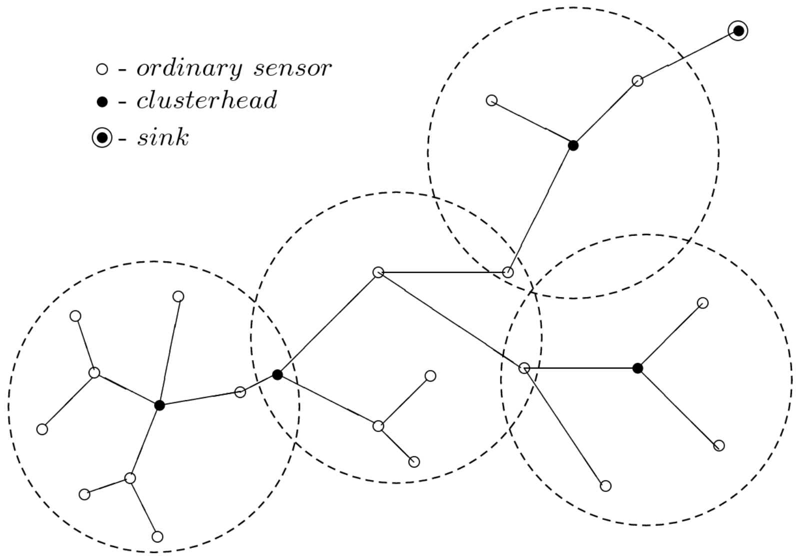

Fig. 1. An example of clustered layers in sensor network

be able to build a backbone network between themselves and the sink such that they can communicate without relying on ordinary sensor nodes. We assume two types of communication. The first one between the clusterhead sensor nodes and the ordinary sensor nodes with a low transmission power. The second one between the clusterheads with a higher transmission power. It is possible to use two different frequency bands or encoding codes in each clusterhead sensor node.

The activity of the sensor network is as follows. During the initialization phase, all the sensor nodes exchange their identifiers with the information about the actual energy power. The sensor nodes decide to which cluster they wish to belong, based on the strength of the signal. If the signal is stronger the sensor node decides to which cluster it should belong. After the clusters are formed, the sensors decide which of them should be a clusterhead. We assume that only the nodes with the highest energy power in the clusters are selected as clusterhead sensor nodes. This procedure is repeated after each long time period.

The proposed clustering scheme needs the following assumptions:

1) The field contains N sensors and only one sink.

2) In the sensor field there are some sensors with a higher energy power than other sensor nodes. These sensors can initiate the clustering process.

3) All the determined clusterhead sensor nodes consume their energy more quickly than other ordinary sensors.

4) For each sensor r is the range of radio transmission.

5) The nodes may fail only in the event of the lack of energy power.

6) Any possible interface between the clusters and the backbone sensor is ommitted.

The consumption of energy in each sensor proceeds in three phases, namely

a) *the sensing phase*, in which each sensor node collects the data from the environment. The energy needed to the data sensing is equal to

$$E_1 = c_1 \cdot b \tag{1}$$

where c_1 is the speed of the data sensing, b is the number of the sensing bits from the environment.

b) *the computation phase*, in which the sensors deplete their energy for coding, decoding, data aggregation, etc. According to (see Sankarasubramaniam, 2003, Eq. 13) the coding/decoding energy depends on the block length m and the number of bits b as follows

$$E_2 = (2m \cdot b + 2b^2) \cdot (E_{add} + E_{mult}) \tag{2}$$

where E_{add} and E_{mult} are the energy needed to carry out the addition and the multiplication in the Galois field GF(2^k) with $k = \lfloor \log_2 m + 1 \rfloor$, m is the length of the channel block.

c) *the communication phase.* The energy consumption in the communication phase is given by

$$E_3(n_1, n_2) = (c_2 \cdot d(n_1, n_2))^{\xi} + c_3 \cdot E(n_1) \cdot E(n_2) + c_4 \cdot \left(\frac{E_b}{N_o}\right)^{(required)} \tag{3}$$

where c_2, c_3, c_4 are positive constants, ξ is the coefficient which depends on the environment. For the air ξ is equal to 2. $d(n_1, n_2)$ is the distance between the two nodes n_1 and n_2. $E(n_1)$ and $E(n_2)$ are the energy of the sensor node sending the data and sensor node receiving it, respectively. $\frac{E_b}{N_o}$ in db is the ratio of energy per bit and the noise energy. This ratio has a close relationship to the SNR (Signal-to-Noise Ratio) or SINR (Signal-to-Inference and Noise Ratio), when interference is treated as noise (see Sklar, 1988).

Looking at the energy consumption by a single sensor, we can summarize all three components, namely

$$E_{cons}(j) = \sum_{i=1}^{3} E_i(j) = E_1(j) + E_2(j) + \sum_{k \in N^{(1)}} E_3(j, k) \tag{4}$$

where $N^{(1)}$ is the set of neighbouring sensors of the first order of the neighbourhood for sensor j. We assume that the set of sensors of the first order of the neighbourhood contains all neighbours which are available by a single hop. On the other hand, all the sensors of the neighbouring set of the first order of the neighbourhood for a given sensor are direct neighbours for him, because, for example, of the limits on the transmission power.

3. Least Squares SVM for Clustering in Wireless Sensor Networks

3.1. An overview of the LS-SVM

Consider a given training set $\{(x_i, y_i)\}_{i=1}^{N}$, with the input data $x_i \in R^n$ and output data $y_i \in R$ with class labels $y_i \in \{-1, +1\}$ and the linear classifier

$$y(x) = sign[w^T x + b] \tag{5}$$

When the data of the two classes are separable we have the original SVM classifier (see Vapnik, 1995, 1998a, 1998b) that satisfies the following conditions:

$$\begin{cases} w^T \phi(x_i) + b \geq +1 & \text{if } y_i = 1 \\ w^T \phi(x_i) + b \leq -1 & \text{if } y_i = -1 \end{cases} \tag{6}$$

These two sets of inequalities can be combined into one single set as follows

$$y_i[w^T \phi(x_i) + b] - 1 \geq 0, \quad i = 1, 2. \ldots, N \tag{7}$$

where ϕ: $R^n \rightarrow R^m$ is the feature mapping the input space to a usually high dimensional feature space. The data points are linearly separable by a hyperplane defined by the pair ($w \in R^m$, $b \in R$). Thus, the classification function is given by

$$f(x) = sign\{w^T \phi(x) + b\} \tag{8}$$

Instead of estimating with the help of the feature map we work with a kernel function in the original space given by

$$K(x_i, x_j) = \phi(x_i)^T \cdot \phi(x_j) \tag{9}$$

In order to allow for the violation of Eq.(7), we introduce slack variables ξ_i such that

$$y_i[w^T \phi(x_i) + b] \geq 1 - \xi_i, \quad \xi_i > 0, \quad i = 1, 2, \ldots, N \tag{10}$$

The following minimization problem is accounted for as follows:

$$\begin{aligned} \min_{w,b,\xi} J(w, b, \xi) \quad &= \quad \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{N} \xi_i \\ \text{subject to} \quad & \quad y_i[w^T \phi(x_i) + b] \geq 1 - \xi_i, \quad \xi_i > 0, \\ & \quad i = 1, 2, \ldots, N, \quad C > 0 \end{aligned} \tag{11}$$

where C is a positive constant parameter used to control the tradeoff between the training error and the margin.

The dual problem of the system (11), obtained as a result of Karush-Kuhn-Tucker (KKT) condition (see Kuhn, 1951), leads to a well-known convex quadratic programming (QP). The solution of the QP problem is slow for large vectors and it is difficult to implement in the on-line adaptive form. Therefore, a modified version of the SVM called the Least Squares SVM (LS-SVM) was proposed by Suykens et al (2002).

In the LS-SVM method, the following minimization problem is formulated

$$\min_{w,b,e} J(w,b,e) = \frac{1}{2}w^T w + \gamma \frac{1}{2}\sum_{k=1}^{W} e_k^2$$
$$\text{subject to} \quad y_k[w^T\phi(x_k)+b] = 1 - e_k,\ k = 1,2,\ldots,N \tag{12}$$

The corresponding Lagrangian for Eq. (11) is given by

$$L(w,b,e;\alpha) = J(w,b,e) - \sum_{k=1}^{N} \alpha_k\{y_k[w^T\phi(x_k)+b] - 1 + e_k\} \tag{13}$$

where the α_k are the Langrange multipliers. The optimality condition leads to the following $(N+1) \times (N+1)$ linear system

$$\begin{bmatrix} 0 & Y^T \\ Y & \Omega^* + \gamma^{-1}I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \vec{1} \end{bmatrix} \tag{14}$$

where

$Z = [\phi(x_1)^T y_1, \ldots, \phi(x_N)^T y_N]$

$Y = [y_1, \ldots, y_N]$

$\vec{1} = [1, \ldots, 1]$

$\alpha = [\alpha_1, \ldots, \alpha_N]$

and $\Omega^* = ZZ^T$. Due to the application of Mercer's condition (see Suykens, 2002) there exists a mapping and an expansion

$$\Omega^*_{kl} = y_k y_l \phi(x_k)^T \phi(x_l) = y_k y_l K(x_k, x_l) \tag{15}$$

Thus, the LS-SVM model for the function estimation is given

$$y(x) = \sum_{k=1}^{N} \alpha_k y_k \cdot K(x, x_k) + b \tag{16}$$

where parameters α_k and b are based on the solution to Eqs. (13) and (14).

3.2. Mixtures of kernels

Each kernel function is characterized by its advantages and disadvantages. For instance, a radial basis function (RBF) kernel $K(x, x_i) = \exp\{- \mid x - x_i \mid^2 / \sigma^2\}$, where σ is the width of the radial basis function, is a typical local kernel in which only the data that are close have an influence on the kernel values. The polynomial kernel (see Vapnik, 1995) such as $K(x, x_i) = [x \cdot x_i + 1]^q$, where q is the kernel parameter which defines the degree of the polynomial to be used, guarantees the influence of all the data points that are far away from each other. Therefore, the mixture of these kernels gives a better performance than any single kernel. As was defined by Smits et al. (2002), an exemplary mixture of the RBF and polynomial kernels is given by

$$K_{mix} = \rho \cdot K_{poly} + (1-\rho)K_{RBF} \tag{17}$$

where ρ is the mixing coefficient treated as a constant scalar.

3.3. The LS-SVM transformed into a clustering problem

We assume that the input data are split up into blocks of $16 \times 16, 64 \times 64, 256 \times 256$, etc. pixels. The input data are described by a coordinate (r, z) and the output data is the energy value. For any data block, the input points are in the form $\{(r_o + dr, z_o + dz) : \mid dr \mid < m, \mid dz \mid < n\}$. All such sets of points can be translated to the same set $\{(dr, dz) : \mid dr \mid < m, \mid dz \mid < n\}$ by subtracting (r_o, z_o) from all the vectors, m and n are the half numbers of the horizontal and vertical pixels of the blocks. The LS-SVM method can be transformed into a clustering problem by the use of the same set of input vectors but different sets of labels.

We can transform the Eq. (14) into a system

$$\begin{bmatrix} 0 & 1^T \\ 1 & \Omega \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ Y \end{bmatrix} \tag{18}$$

where $Y = [y_1; \ldots; y_N]$, $\Omega = K(x_i, y_i) + \gamma^{-1} I$, $1^T = [1_1; \ldots; 1_N]$, $\alpha = [\alpha_1; \ldots; \alpha_N]$. Thus, the Ω is given by

$$\Omega = K(x_i, x_j) + \gamma^{-1} I \tag{19}$$

The solution of Eq. (18) gives the values

$$\begin{cases} b &= \frac{1^T \Omega^{-1} Y}{1^T \Omega^{-1} 1} \\ \alpha &= \Omega^{-1}(Y - b1) \end{cases} \tag{20}$$

By setting $A = \Omega^{-1}$ and $B = \frac{1^T \Omega^{-1}}{1^T \Omega^{-1} 1}$ we obtain

$$\begin{cases} b &= BY \\ \alpha &= A(Y - b1) \end{cases} \tag{21}$$

where A and B are precalculated matrixes that depend only on the input vector (x_k) but not on the vector y_k.

The sensors are usually correlated due to the high probability that the adjacent pixels will contain the sensors in the cluster. We assume that the sensor field is two-dimensional and the image energy distribution of the sensor field on the surface, known as the point spread funcion (PSF), can be approximated by the Gaussian PSF. On the other hand, the center point of the PSF corresponds to the measured sensor position.

In our approach based on the mixtures of kernels, we take into consideration two sets of indexes of the neighbourhood of the sensor image that satisfies the condition of linearly separable patterns. We recall that the linear separability requires that in order to be classified, the patterns must be separated from each other to ensure that decision surfaces should consist of hyperplans.

The LS-SVM with the RBF and polynomial kernels transformed into a sensor clustering problem has the fitted energy intensity surface function over the constant vector space as follows

$$g(r,z) = \sum_{k=1}^{N} \alpha_k \left\{ (\rho[(rr_k + zz_k) + 1]^q + (1-\rho) \cdot e^{\frac{-(|r-r_k|^2 + |z-z_k|^2)}{\sigma^2}} \right\} + b \quad (22)$$

where (r, z) are the coordinates of the pixels. The function $g(r, z)$ gives the corresponding energy intensity value, b and α are obtained as a solution of Eq. (20).

3.4. A multi-class formulation of the LS-SVM transformed into a sensor clustering problem

In comparison with the standard SVM method, the LS-SVM has a lower computational complexity and memory requirements. Nevertheless, in certain situations, such as the classification of several characters, clusters, etc., a multi-class classification is very suitable.

In the multi-class formalism we now use multiple output values y^i with $i = 1, \ldots, n_y$, where n_y defines the number of output values (see Suykens et al., 2002). Thus, in the primal weight space the multi-class classification system possesses the following binary classifiers

$$\left\{ \begin{array}{lcl} y^{(1)}(x) & = & sign[{w^{(1)}}^T \phi^{(1)}(x) + b^{(1)}] \\ y^{(2)}(x) & = & sign[{w^{(2)}}^T \phi^{(2)}(x) + b^{(2)}] \\ \vdots & & \vdots \\ y^{(n_y)}(x) & = & sign[{w^{(n_y)}}^T \phi^{(n_y)}(x) + b^{(n_y)}] \end{array} \right. \quad (23)$$

with mappings on a high dimensional feature space $\phi^{(i)}(.) : R^n \rightarrow R^{n_h}$, $i = 1, 2, \ldots, n_y$, with dimensions $n_{h_1}, n_{h_2}, \ldots, n_{h_{n_y}}$.

By the extension of Eq. (20) to a multi-class problem, we obtain

$$\left\{ \begin{array}{lcl} \overline{b} & = & B\overline{Y} \\ \overline{\alpha} & = & A(\overline{Y} - \overline{b}1) \end{array} \right. \quad (24)$$

where matrix $\overline{Y}$ is given by

$$\overline{Y} = \begin{bmatrix} y_1^{(1)}, \ldots, y_N^{(1)} \\ y_2^{(2)}, \ldots, y_N^{(2)} \\ \ldots \\ y_1^{(n_y)}, \ldots, y_N^{(n_y)} \end{bmatrix} \quad (25)$$

and vector $b_M = [b^{(1)}, \ldots, b^{(n_y)}]$, where N is the number of pixels in one processing block.

4. Simulating experiments and results

We used the LS-SVMLab (see Pelckmans et al., 2003) within a MATLAB to simulate the clustering process for WSNs with a varying sensor density and several kernels with different parameters. To generate the WSN for each simulation experiment, the location of each sensor and energy power is found by generating three random numbers, two of them uniformly distributed in [0, 2 α], where 2α is the length of a side of the square area in which the sensors are distributed, and one of them to define the current energy power uniformly distributed in [0, E_{max}]. In all of these experiments, the communication range of each sensor was assumed to be 1 unit.

```
procedure clustering_of_WSN;
  begin
    while sensor_state(i) = ACTIVE do
      begin
        energy_of_sensor(i) := test_of_residual_energy_of_sensor(i);
        if energy_sensor(i) > minimal_energy_level;
        then add_sensor(i)_to_cluster(j);
      end;
    energy_balance_for_all_clusters;
  end;
```

Fig. 2. Pseudo-code of the clustering procedure for WSN

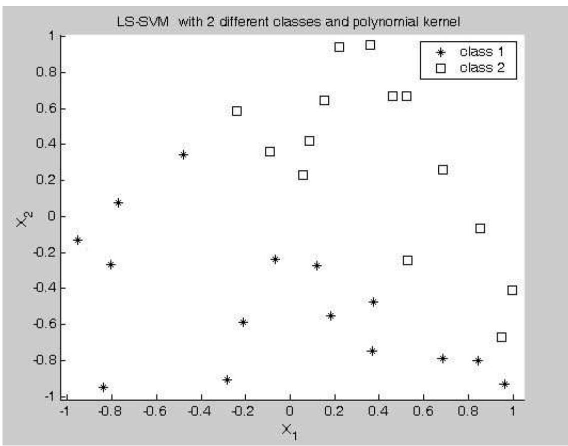

Fig. 3. The clustering process with the polynomial kernel

Our goal is to build clusters for WSN. Each of the sensors must belong to one of them. We propose the following algorithm for this purpose (see Fig. 2). The proposed algorithm can provide the energy balance between all the obtained clusters. As a result, we obtain well energy balanced clusters with all of the sensors from the sensor field.

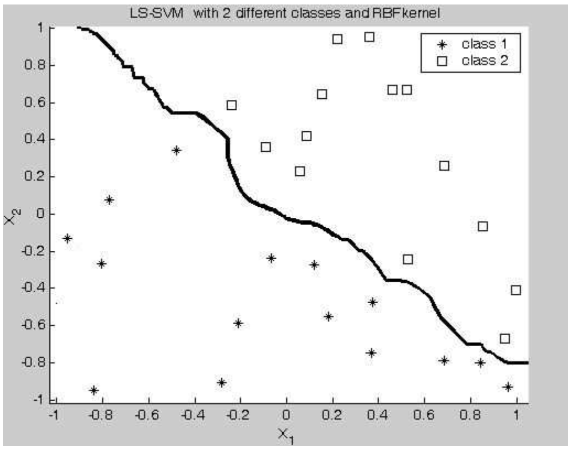

Fig. 4. The clustering process with the RBF kernel

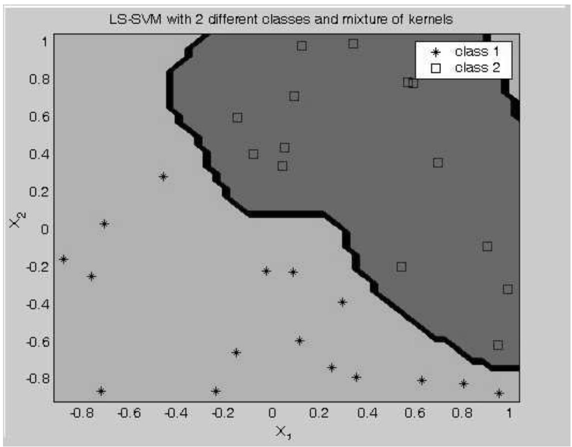

Fig. 5. The clustering process with the mixture of kernels

Figures 3, 4 show the output of these simulations of our algorithm with the polynomial and the RBF kernel, respectively, for a sensor network of 30 sensors that are distributed uniformly in a square of 100 square unit. Figure 5 gives the output of our algorithm with the mixure of both kernels. It can be seen that the clustering process with a mixture of kernels gives better results, especially in the case of the LS-SVM multi-class classifiers used for the clustering problem. It is worth noting that there are many parameters in the proposed method. For example, the different kernels functions may provide to several clusters with different performances.

5. Conclusions

This paper has discussed a method of building clusters in the wireless sensor networks. The applied strategy is possible through the use of information about the energy power of all the sensors in the sensor field, while the communication between the sensors is guided by a set of parameters, such as the data number sent by the sensors, the speed of transmission, etc. The LS-SVM method has been transformed into our clustering problem. With the use of the image energy distribution of the sensor field surface and the mixtures of kernels, we have been able to solve the clustering problem in WSNs.

A number of experiments illustrate the use of the proposed method both with respect to visuability as well as generalization performance. The proposed method has a large potential in practice. Firstly, the LS-SVM method to solve the clustering problem in WSN is very sparse. Secondly, compared to the traditional clustering methods, this method incorporates energy information in the sensors and their localizations in the sensor networks. This information is available in all sensor networks and is crucial in WSNs, including the routing, lifetime estimation, etc.

Since it is the first time the LS-SVM in clustering of WSNs are applied, there still remain some problems, such as the selection of the best mixtures of kernels and the optimization of parameters.

References

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. (2002) Wireless Sensor Networks. *Computer Networks* **38**, 393 – 422.

Baker, D.J. and Ephremides, A. (1981) The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm. *IEEE Trans. on Communications* **29**, 11, 1694 – 1701.

Barzilay, O. and Brailovsky, V.L. (1999) On Domain Knowledge and Feature Selection Using a Support Vector Machine. *Pattern Recognition Letters* **20**, 475 – 484.

Cortes, C. and Vapnik, V.N. (1995) Support Vector Networks. *Machine Learning* **20**, 273 – 297.

Heinzelman, W.B., Chandrakasan, A.P. and Balakrishnan, H. (2002) An Application-Specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Transactions on Wireless Networking* **1**, 4, 660 – 670.

Jonsson, K., Kittler, J. Li, Y.P. and Matas, J. (1999) Support Vector Machines for Face Autentication. In: Pridmore, T. and Elliman, D. (Eds.), *British Machine Vision Conference*, London, 543 – 553.

Kuhn, H. and Tucker, A. (1951) Nonlinear Programming. In: *Proceedings of the 2nd Berkeley Symposium on Mathematica Statistics and Probabilistics*, University of California Press, 481 – 492.

Krishnan, R. and Starobinski, D. (2006) Efficient Clustering Algorithms for Self-organizing Wireless Sensor Networks. *Ad Hoc Networks* **4**, 36 – 59.

Mitéran, J., Bouillant, S. and Bourennane, E. (2003) SVM Approximation for Real-Time Image Segmentation by Using an Improved Hyperrectangles-based Method. *Real-Time Imaging* **9**, 179 – 188.

Pelckmans, K., Suykens, J.A.K., Van Gestel, T., De Brabanter, J., Lukas, L., Hamers, B., De Moor, B., Vandewalle, J. (2003) *LS-SVMLab Toolbox User's Guide, ver. 1.5*, `http://www.esat.kuleuven.ac.be/sista/lssvmlab/`.

Sankarasubramaniam, Y., Akyildiz, I.F. and McLanghlin, S.W. (2003) Energy Efficiency Based Packet Size Optimization in Wireless Sensor Networks. *Proceedings of the 1st International Workshop on Sensor Network Protocols and Applications (SNPA)*, Anchorage.

Sklar, B. (1988) *Digital Communication - Fundamentals and Applications*, Prentice Hall, Englewood Cliffs, NJ.

Smits, G.F. and Jordan, E.M. (2002) Improved SVM Regression Using Mixtures of Kernels, *Proceedings of IJCNN '02 on Neural Networks*, **3**, 2785 – 2790.

Suykens, J.A.K. and Vandewalle, J. (1999) Least Squares Support Vector Machine Classifiers, *Neural Processing Letters* **9**, 3, 293 – 300.

Suykens, J.A.K., van Gestel, T., de Brabanter, J., de Moor, B., Vandewalle, J. (2002) *Least Squares Support Vector Machines*, World Scientific, New Jersey, London, Singapore, Hong Kong.

Tang, Sh. and Li, W. (2006) QoS Supporting and Optimal Energy Allocation for a Cluster Based Wireless Sensor Network. *Computer Communications* **29**, 2569 – 2577.

Vapnik, V.N. (1995) *The Nature of Statistical Learning Theory*, Springer-Verlag, Berlin, Heidelberg, New York.

Vapnik, V.N. (1998) *Statistical Learning Theory*, John Wiley and Sons.

Vapnik, V.N. (1998) The Support Vector Method of Function Estimation. In: [Suykens, J.A.K. and Vandewolle, J.] *Nonlinear Modeling: Advanced Black-box Techniques*, Kluwer Academic Publishers, Boston, 55 – 85.