*Sensors* **2012**, *12*, 15292-15307; doi:10.3390/s121115292

OPEN ACCESS

*sensors*

**ISSN 1424-8220**

www.mdpi.com/journal/sensors

*Article*

# A Modular Spectrum Sensing System Based on PSO-SVM

**Zhuoran Cai, Honglin Zhao *, Zhutian Yang and Yun Mo**

School of Electronics and Information Technology, Harbin Institute of Technology, Harbin 150001, China; E-Mails: qingdaogancai@sina.com.cn (Z.C.); deanzty@gmail.com (Z.Y.); moyuntc@126.com (Y.M.)

* Author to whom correspondence should be addressed; E-Mail: hlzhao@hit.edu.cn; Tel.: +86-136-0363-9389.

*Received: 15 August 2012; in revised form: 5 November 2012 / Accepted: 5 November 2012 / Published: 8 November 2012*

**Abstract:** In the cognitive radio system, spectrum sensing for detecting the presence of primary users in a licensed spectrum is a fundamental problem. Energy detection is the most popular spectrum sensing scheme used to differentiate the case where the primary user's signal is present from the case where there is only noise. In fact, the nature of spectrum sensing can be taken as a binary classification problem, and energy detection is a linear classifier. If the signal-to-noise ratio (*SNR*) of the received signal is low, and the number of received signal samples for sensing is small, the binary classification problem is linearly inseparable. In this situation the performance of energy detection will decrease seriously. In this paper, a novel approach for obtaining a nonlinear threshold based on support vector machine with particle swarm optimization (PSO-SVM) to replace the linear threshold used in traditional energy detection is proposed. Simulations demonstrate that the performance of the proposed algorithm is much better than that of traditional energy detection.

**Keywords:** cognitive radio; spectrum sensing; PSO-SVM; detection threshold

## 1. Introduction

Based on the conventional fixed spectrum allocation policy, most available radio spectra have been assigned to registered users, which lead to a serious waste of spectrum utilization. In fact, recent reports from Federal Communications Commission (FCC) have shown that only 30% of the allocated spectrum in US is fully utilized [1]. Cognitive radio, which enables secondary users to utilize the spectrum when primary users are not occupying it, has been proposed as a promising technology to improve spectrum utilization efficiency [2–4], and has three essential components: (1) Spectrum sensing: the secondary users sense the radio spectrum environment within their operating range to detect the frequency bands which are not occupied by primary users; (2) Dynamic spectrum management: cognitive radio networks dynamically select the best available bands for communication; (3) Adaptive communications: a cognitive radio device can configure its transmission parameters (e.g., carrier frequency, transmission power) to opportunistically make best use of the ever changing available spectrum [5].

Spectrum sensing is a fundamental task for cognitive radio. However, there are several factors that make spectrum sensing practically challenging (e.g., low signal-to-noise ratio (*SNR*) of primary users, noise uncertainty, multipath fading). Several sensing methods have been proposed, including likelihood ratio test (LRT) [6–8], energy detection method [9,10], match filtering (MF) method [11], cyclostationary detection method [12,13] and the statistical covariances-based method [14]. Each of them has its own advantages and disadvantages, e.g., LRT is proven to be optimal, but it requires exact channel information and distributions of the primary signal and noise. The MF-based method needs perfect knowledge of the channel responses from primary users to the receiver and accurate synchronization (otherwise, its performance will dramatically be reduced) [15], it may not be possible if the primary users do not cooperate with the secondary users. The cyclostationary detection method requires the cyclic frequencies of the primary users, which may not be realistic for many spectrum reuse applications. Furthermore it needs high computation capabilities. The energy detection method does not require any primary signal information and it is robust to unknown dispersed channels and fading, but if the *SNR* of the received signal is low, the number of received signal samples is small and the power of noise is estimated inaccurately, the energy detection performance will decrease seriously [11]. The covariances-based method also does not require any prior information, but its computation complexity is also high [14].

As mentioned one drawback of the traditional energy detection is that if the *SNR* of received signal is low and the number of the received signal samples is small the corresponding performance may decrease seriously. In order to overcome this drawback a novel method with the purpose of obtaining a nonlinear threshold for energy detection based on PSO-SVM is proposed in this paper. The proposed method focuses on one single point and one antenna scenario, which can be divided into Offline module and Online module. In the Offline module, the proposed system generates two classes of training samples, one for the simulated situation that both signal and noise exist simultaneously while another case is for noise only. The normalized energy of these two classes of training samples is used as the classification feature to train the PSO-SVM. After each training step, a decision function is generated. In the Online module, the decision functions obtained in the Offline module are used as the nonlinear thresholds for energy detection to verify if the primary user is present. The experimental results show that the receiver operating characteristic (ROC) curve of proposed approach is much better than traditional energy detection. The rest of this paper is organized as follows: the PSO-SVM is introduced in Section 2. In Section 3 determination of threshold and theoretical analysis are proposed. Simulation results are given in Section 4. Conclusions are finally drawn in Section 5.

## 2. PSO-SVM

### 2.1. SVM

In this subsection, a brief introduction to SVM, proposed by Vapnik [16], is given. Let $(\mathbf{x}_i, y_i)_{1\le i\le N}$ be a set of *N* training samples, each sample $\mathbf{x}_i \in R^d$ is a vector, *d* being the dimension of the input space, belonging to a class labeled by $y_i \in \{1,-1\}$. It amounts to find weight vector $\mathbf{w}$ and scale *b*, which satisfy:

$$y_i\left(\langle \mathbf{w}\cdot\mathbf{x}_i\rangle + b\right) \ge 1 \tag{1}$$

where $\langle\cdot\rangle$ is inner product. The aim of SVM is to find the hyper-plane which makes the samples with the same label on the same side of the hyper-plane. The quantity $1/\|w\|_2$ is the margin, and optimal separating hyper-plane (OSH) is the separating hyper-plane which maximizes the margin. The larger the margin, the better the generalization is expected [17]. To search the maximum *γ*, quadratic programming is usually used, leading to:

$$\left.\begin{array}{l}\underset{\mathrm{w,b}}{\text{minimize}}\dfrac{1}{2}\|\mathbf{w}\|^2\\ \textit{subject to } y_i\left(\langle \mathbf{w}\cdot\mathbf{x}_i\rangle + b\right)\ge 1, i=1,2,\cdots,N\end{array}\right\} \tag{2}$$

according to Equation (2), a hyper-plane $\langle \mathbf{w}\cdot\mathbf{x}_i\rangle + b = 0$ with the largest margin can be obtained. For equality constraints in Equation (2), they can be modified to unconstrained by integrating positive Lagrange multipliers, leading to:

$$L(\mathbf{w},b,\alpha) = \frac{1}{2}\langle \mathbf{w}\cdot\mathbf{w}\rangle - \sum_{i=1}^{N}\alpha_i\left[y_i\left(\langle \mathbf{w}\cdot\mathbf{x}_i\rangle + b\right) - 1\right] \tag{3}$$

to minimize $L(\mathbf{w},b,\alpha)$ by requiring the gradient of $L(\mathbf{w},b,\alpha)$ respect to $\mathbf{w}$ and *b* vanish, a dual form is given by:

$$\begin{array}{l}\mathbf{w} = \displaystyle\sum_{i=1}^{N}\alpha_i y_i \mathbf{x}_i\\ \displaystyle\sum_{i=1}^{N}\alpha_i y_i = 0\\ \alpha_i \ge 0\end{array} \tag{4}$$

substituting Equation (4) into Equation (3) gives:

$$\begin{aligned}&\inf_{\mathbf{w},b} L(\mathbf{w},b,\alpha)\\ &= \frac{1}{2}\sum_{i=1}^{N} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i\cdot\mathbf{x}_j\rangle - \sum_{i=1}^{l} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i\cdot\mathbf{x}_j\rangle + \sum_{i=1}^{l}\alpha_i\\ &= \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i\cdot\mathbf{x}_j\rangle = w(\alpha)\end{aligned} \tag{5}$$

According to Equation (3) and Equation (5), it is obvious that $w(\alpha)=\inf_{\mathbf{w},b} L(\mathbf{w},b,\alpha)\le L(\mathbf{w},b,\alpha)$, thus the quadratic programming problem in Equation (3) can be converted to:

$$\text{maximize } w(\alpha)=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}y_i y_j \alpha_i \alpha_j \left\langle \mathbf{x}_i\cdot\mathbf{x}_j\right\rangle \tag{6}$$

subject to:

$$\begin{aligned}&\sum_{i=1}^{N}\alpha_i y_i=0\\ &\alpha_i\ge 0\ i=1,\ldots,N\\ &\alpha_i\left[y_i\left(\left\langle\mathbf{w}\cdot\mathbf{x}\right\rangle+b\right)-1\right]=0\end{aligned} \tag{7}$$

the third constraint condition in Equation (7) is the Karush-Kuhn-Tucker (KKT) condition [18]. There is a Lagrange multiplier $a_i$ for each training sample, the training samples for which $a_i > 0$ are called "support vectors", lying on one of the two hyper-planes: $\left\langle\mathbf{w}\cdot\mathbf{x}^{+}\right\rangle+b=+1$; $\left\langle\mathbf{w}\cdot\mathbf{x}^{-}\right\rangle+b=-1$. The training samples will only appear in the form of inner products between vectors.

In the nonlinear case, the approach adapted to noisy data is to make a soft margin. We introduce the slack variables $\xi_i\ge 0, i=1,2,\cdots,N$ so that:

$$y_i\left(\left\langle\mathbf{w}\cdot\mathbf{x}_i\right\rangle+b\right)\ge 1-\xi_i \tag{8}$$

The generalized OSH is the solution of minimizing:

$$\frac{1}{2}\left\langle\mathbf{w}\cdot\mathbf{w}\right\rangle+C\sum_{i=1}^{N}\xi_i \tag{9}$$

subject to Equation (7). The parameter $\sum_{i=1}^{N}\xi_i$ is the upper bound of the number of training errors and $C$ is the penalty parameter to control errors.

In the nonlinear SVM, a kernel function is introduced to map the initial data into a feature space with a high dimension. In the new space, the data should be linearly separable. Then Equation (6) can be converted to:

$$\text{maximize } w(\alpha)=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i=1}^{N}y_i y_j \alpha_i \alpha_j K\left(\mathbf{x}_i,\mathbf{x}_j\right) \tag{10}$$

subject to Equation (7), and $0\le\alpha_i\le C$. $K\left(\mathbf{x}_i,\mathbf{x}_j\right)$ is the kernel function. As one of the most popular kernel functions, the RBF kernel function is considered in this paper, and it takes the following form:

$$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\exp\left\{-g\left\|\mathbf{x}_j-\mathbf{x}_i\right\|^2\right\} \tag{11}$$

where $g$ is kernel parameter, and denoting the width of kernel function. If $g$ is too big, the SVM may outfit the training data, and a too small $g$ may make SVM algorithm not flexible enough for complex function approximation. By solving Equation (10) we can obtain the minimum $\left\langle\mathbf{w}\cdot\mathbf{w}\right\rangle$:

$$\left\langle \mathbf{w}^* \cdot \mathbf{w}^* \right\rangle = \sum_{i=1}^{N} y_i y_j \alpha_i^* \alpha_j^* K\left(\mathbf{x}_i, \mathbf{x}_j\right) \tag{12}$$

where $\mathbf{w}^*, \alpha_i^*$ are the solution of Equation (10). Then the decision function is:

$$f(x) = sign\left(\left\langle \mathbf{w}^* \cdot \mathbf{x} \right\rangle + b\right) \tag{13}$$

where $\mathbf{x}$ is a test sample with an unknown label $y$. Noting that $\mathbf{w}^*$ only depend on the training samples $\left(\mathbf{x}_i, y_i\right)_{1 \le i \le N}$, if we choose a set of training samples $\left(\mathbf{x}_i, y_i\right)_{1 \le i \le N}$, a unique decision function will be obtained by solving Equation (10).

## *2.2. Particle Swarm Optimization*

The parameters *C* and *g* need to be set to solve Equation (10). However, in a practical situation it is difficult to set these two parameters properly. In this subsection, we integrate a PSO (particle swarm optimization) method to adaptively set *C* and *g*.

Particle Swarm Optimization (PSO) is inspired by the social behavior of birds, birds in a swarm preying on food and cooperation with each other to search for the optimal position to obtain the food. In a PSO optimal problem, there are a group of individuals, each individual is called a "particle", which may be a potential solution. Suppose that the solution space of optimization problem is *D* dimensions. The *i-th* particle is $X_i = \left(x_{i1}, x_{i2}, x_{i3}, \cdots, x_{iD}\right)$, the optimal position for itself is $P_i = (p_{i1}, p_{i2}, p_{i3}, \ldots p_{iD})$ and its speed move to this position is $V_i = \left(v_{i1}, v_{i2}, v_{i3}, \cdots, v_{iD}\right)$, the optimal swarm position is $P_g = (p_{g1}, p_{g2}, p_{g3}, \ldots p_{gD})$, iteration to find the optimal position is given by:

$$x_{id}^{t+1} = \varpi \cdot v_{id}^t + c_1 \cdot r_1 \cdot \left(p_{id} - x_{id}^t\right) + c_2 \cdot r_2 \cdot \left(p_{gd} - x_{id}^t\right) \tag{14}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{15}$$

where $x_{id}^t$ and $v_{id}^t$ are the *d–th* location and speed component of *i–th* particle at the *t–th* iteration, $c_1$ and $c_2$ are two positive coefficients, $r_1$ and $r_2$ are the random number selected from 0 to 1, $\varpi$ is the flexible coefficient of $v_{id}^t$. The second part of Equation (12) is called "cognitive" part which is the optimization of the particle itself, and the third part of Equation (12) is the "swarm part", which is the cooperation with other particles. In addition, a boundary for each particle should be set as $\left[-x_{id\max}, x_{id\max}\right]$, and $v_{id\max}$. In the process of the iteration, if the parameter of particle is out of the range, they are replaced by the boundary value. For the optimization of the error penalty parameter $C$ and kernel parameter *g*, each particle is set as $(C, g)$, and a three cross validation is used to estimate the performance of each $(C, g)$, training data is separated into three parts, one part is considered as the validation set and the remaining two parts are for training . Our earlier work has validated this PSO algorithm [19].

## 3. Threshold Determination and Theoretical Analysis

Common notation as summarized in Table 1 is used throughout this section.

**Table 1.** Notation

| $N_s$ | **number of signal samples for spectrum sensing** |
|---|---|
| $s(n)$ | actual received primary signal sample |
| $u(n)$ | actual received noise sample |
| $s_{tr}(n)$ | training primary signal sample in offline module |
| $u_{tr}(n)$ | training noise sample in offline module |
| $\sigma_s^2$ | variance of actual received primary signal |
| $\sigma_u^2$ | variance of actual noise |
| $\sigma_{s_{tr}}^2 = \lambda_{tr}$ | variance of training primary signal in offline module |
| $\sigma_{u_{tr}}^2 = 1$ | variance of training noise in offline module |
| $\lambda$ | *SNR* of actual received primary signal |
| $\lambda_{tr}$ | *SNR* of training primary signal in offline module |
| $T$ | actual decision statistic |
| $T_1$ | actual decision statistic at hypothesis $H_1$ |
| $T_0$ | actual decision statistic at hypothesis $H_0$ |
| $T^{'} = T/\sigma_u^2$ | normalized actual decision statistic |
| $T_1^{'} = T_1/\sigma_u^2$ | normalized actual decision statistic at hypothesis $H_1$ |
| $T_0^{'} = T_0/\sigma_u^2$ | normalized actual decision statistic at hypothesis $H_0$ |
| $T_{1_{tr}}^{N_s,\lambda_{tr}}$ | training decision statistic defined by $N_s$, $\lambda_{tr}$ at hypothesis $H_1$ in offline module |
| $T_{0_{tr}}^{\ N_s}$ | training decision statistic defined by $N_s$ at hypothesis $H_0$ in offline module |

### *3.1. Basic Conception of Energy Detection*

In this subsection we introduce the general model for spectrum sensing, then review the energy detection scheme and analyze the relationship between the probability of false alarm and probability of detection.

Suppose that we are interested in the frequency band with carrier frequency $f_c$, bandwidth $W$ and the received signal is sampled at sampling frequency $f_s$, respectively. When the primary user is active, the discrete received signal at the secondary user is given by [5]:

$$z(n) = s(n) + u(n) \tag{16}$$

where $z(n)$ is under hypothesis $H_1$. When the primary user is inactive, the received signal is given by:

$$z(n) = u(n) \tag{17}$$

and this case is referred to the hypothesis $H_0$. In this paper we only focus on one single point and one antenna spectrum sensing scenario, thus for simplicity we make the following assumptions [5]:

(AS1) The primary signal $s(n)$ is an independent, and identically distributed (IID) random process with mean zero and unknown variance $E\left[\left|s(n)\right|^2\right]=\sigma_s^2$;

(AS2) The noise $u(n)$ is a Gaussian IID random process with mean zero and variance $E\left[\left|u(n)\right|^2\right]=\sigma_u^2$, which can be estimated;

(AS3) The primary signal $s(n)$ is independent of the noise $u(n)$.

The signal-to-noise ratio (*SNR*) of the actual primary user measured at the secondary receiver of interest is $\lambda=\frac{\sigma_s^2}{\sigma_u^2}$, under hypothesis $H_1$. We consider the circularly symmetric complex Gaussian (CSCG) as the noise case. For the primary signal $s(n)$, we consider complex PSK modulated signal.

Two probabilities are of interest for spectrum sensing: probability of detection $P_d$, which defines, at hypothesis $H_1$, the probability of sensing method correctly detecting the presence of primary user; and probability of false alarm $P_f$, which defines, at hypothesis $H_0$, the probability of sensing method claiming the presence of primary user.

Energy detection is one of the most popular spectrum sensing schemes, because of it does not need any prior information about the primary signal and is easy to apply. Let $\tau$ be the available sensing time and $N_s$ be the number of samples, for simplicity we assume $N_s=\tau f_s$. The test statistics are given by [5]:

$$\begin{aligned} T&=\frac{1}{N_s}\sum_{n=1}^{N_s}\left|z(n)\right|^2\\ T_1&=\frac{1}{N_s}\sum_{n=1}^{N_s}\left|s(n)+u(n)\right|^2\\ T_0&=\frac{1}{N_s}\sum_{n=1}^{N_s}\left|u(n)\right|^2 \end{aligned}\tag{18}$$

for generality, decision statistics need to be normalized with the estimated power of noise, then the normalized decision statistics are given by:

$$\begin{aligned} T^{'}&=\frac{T}{\sigma_u^2}\\ T_1^{'}&=\frac{T_1}{\sigma_u^2}\\ T_0^{'}&=\frac{T_0}{\sigma_u^2} \end{aligned}\tag{19}$$

if the number of signal samples $N_s$ is small, the probability of false alarm $P_f$ for a predefined threshold $\varepsilon$ is given by:

$$P_f=\Pr\left(T^{'}>\varepsilon\mid H_0\right)=\Pr\left(T_0^{'}>\varepsilon\right)\tag{20}$$

The far right-hand side of Equation (20) indicates a class of chi-square variable with $2N_s$ degrees of freedom for complex-valued case. From Equation (20) the threshold $\varepsilon$ is related to the $P_f$ as

$\varepsilon = chi2^{-1}\left(P_f, 2N_s\right)$, where $chi2^{-1}(\cdot)$ is inverse of the chi-square cumulative distribution function. For the same threshold ε, the probability of detection $P_d$ is given by:

$$P_d = \Pr\left(T^{'} > \varepsilon \mid H_1\right) = \Pr\left(T_1^{'} > \varepsilon\right) \tag{21}$$

The symbol $T_1^{'}$ indicates a class of non-central chi-square variable with $2N_s$ degrees of freedom and a non-centrality parameter $\lambda$, in our case $\lambda = \dfrac{\sigma_s^2}{\sigma_u^2}$, extensive tables exist for the chi-square distribution, but the non-central chi-square has not been as extensively tabulated.

In this paper, we use approximations proposed by Patnaik [20] to replace the non-central chi-square with a central chi-square having a different number of degrees of freedom and a modified threshold level, If the non-central chi-square variable has $2N_s$ degrees of freedom and non-centrality parameter $\lambda$, define a modified number of degrees of freedom *D* and a threshold divisor *G* given by:

$$\begin{aligned} D &= \left(2N_s + \lambda\right)^2 \big/ \left(2N_s + 2\lambda\right) \\ G &= \left(2N_s + 2\lambda\right) \big/ \left(2N_s + \lambda\right) \end{aligned} \tag{22}$$

then:

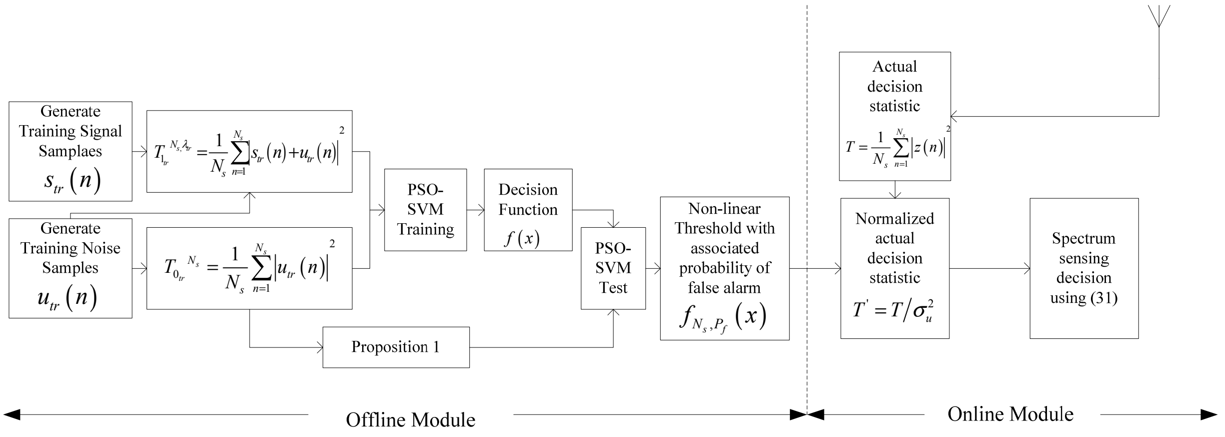
$$P_d = \Pr\left(T_1^{'} > \varepsilon\right) = \Pr\left(T_0^{'} > \varepsilon / G\right) \tag{23}$$

As mentioned above, when the probability of false alarm $P_f$ and the number of samples $N_s$ is set, we can obtain a unique value of threshold ε, it equals to a linear classifier in binary classification problem. But if the *SNR λ* is low while the number of signal samples $N_s$ is small, the corresponding spectrum sensing problem is linearly inseparable, and the traditional energy detection can not classify this linearly inseparable problem efficiently.

### *3.2. Nonlinear Threshold System*

To overcome the drawbacks of traditional energy detection mentioned above, the authors here propose a method in purpose of obtaining nonlinearly threshold based on PSO-SVM. The system process encompasses two distinct modules *i.e.*, Offline and Online, which are clearly illustrated by the Figure 1.

**Figure 1.** System model of the proposed method.


Generate Training Signal Samplaes $s_{tr}(n)$
$T_{1_{tr}}^{N_s,\lambda_{tr}} = \frac{1}{N_s}\sum_{n=1}^{N_s}\left|s_{tr}(n)+u_{tr}(n)\right|^2$
Generate Training Noise Samples $u_{tr}(n)$
$T_{0_{tr}}^{N_s} = \frac{1}{N_s}\sum_{n=1}^{N_s}\left|u_{tr}(n)\right|^2$
PSO-SVM Training
Decision Function $f(x)$
PSO-SVM Test
Proposition 1
Non-linear Threshold with associated probability of false alarm $f_{N_s,P_f}(x)$
Actual decision statistic $T = \frac{1}{N_s}\sum_{n=1}^{N_s}\left|z(n)\right|^2$
Normalized actual decision statistic $T^{'} = T/\sigma_u^2$
Spectrum sensing decision using (31)
Offline Module
Online Module

### 3.2.1. Offline Module

The main function of the Offline Module is to generate the nonlinear thresholds for energy detection.

Firstly, the proposed system generates training signal and training noise under AS1-3. Therefore the variance of training signal is a known value $\sigma_{s_{tr}}^2 = \lambda_{tr}$, the variance of training noise is $\sigma_{u_{tr}}^2 = 1$, and training *SNR* is $\lambda_{tr}$.

Secondly, based on the parameter $\lambda_{tr}$ and number of signal samples $N_s$, two classes of training decision statistics under hypothesis $H_0$ and hypothesis $H_1$ could be obtained, which are given by :

$$T_{1_{tr}}^{N_s,\lambda_{tr}} = \frac{1}{N_s}\sum_{n=1}^{N_s}\left|s_{tr}(n)+u_{tr}(n)\right|^2 \tag{24}$$

$$T_{0_{tr}}^{N_s} = \frac{1}{N_s}\sum_{n=1}^{N_s}\left|u_{tr}(n)\right|^2 \tag{25}$$

where the symbol $T_{1_{tr}}^{N_s,\lambda_{tr}}$ indicates a class of non-central chi-square variables with mean $1+\lambda_{tr}$, $2N_s$ degrees of freedom, and a non-centrality parameter $\lambda_{tr}$. While the symbol $T_{0_{tr}}^{N_s}$ indicates a class of central chi-square variables with mean 1 and $2N_s$ degrees of freedom.

Thirdly, labeling each variable of $T_{1_{tr}}^{N_s,\lambda_{tr}}$ class as "+1", and each variable of $T_{0_{tr}}^{N_s}$ class as "–1", then these two classes of variables are used as training data to train PSO-SVM mentioned in Section 2. Consequently, a separating hyper-plane $\left\langle \mathbf{w}^* \cdot \mathbf{x}\right\rangle + b = 0$ and a decision function $f(x) = sign\left(\left\langle \mathbf{w}^* \cdot \mathbf{x}\right\rangle + b\right)$ could be derived.

In the fourth step, the variables of $T_{0_{tr}}^{N_s}$ class are applied to test this decision function so as to gain the probability (denoted as $P_e^{N_s,\lambda_{tr}}$) that the decision function mistakenly label a $T_{0_{tr}}^{N_s}$ variable as a $T_{1_{tr}}^{N_s,\lambda_{tr}}$ variable.

Proposition1: The probability of a variable mistakenly labeled by decision function is determined by the geometric distance $\gamma$ which is from this variable to the separating hyper-plane.

Proof: The proof is mainly based on Rosenblatt classifier, and detailed proof is given in Appendix A.

The average geometric distance from variables of $T_{0_{tr}}^{N_s}$ class to separating hyper-plane is given by:

$$\gamma_{-1_{tr}} = E\left[\frac{\left|\left\langle \mathbf{w}^* \cdot T_{0_{tr}}^{N_s}\right\rangle + b\right|}{\left\|\mathbf{w}^*\right\|_2}\right] = \frac{\left|\left\langle \mathbf{w}^* \cdot 1\right\rangle + b\right|}{\left\|\mathbf{w}^*\right\|_2} \tag{26}$$

while the average geometric distance from variables of $T_0^{'}$ class to separating hyper-plane is given by:

$$\gamma_{-1} = E\left[\frac{\left|\left\langle \mathbf{w}^* \cdot T_0^{'}\right\rangle + b\right|}{\left\|\mathbf{w}^*\right\|_2}\right] = \frac{\left|\left\langle \mathbf{w}^* \cdot 1\right\rangle + b\right|}{\left\|\mathbf{w}^*\right\|_2} \tag{27}$$

according to Equation (26) and Equation (27), it is obvious that the average geometric distance from variables of $T_{0_{tr}}^{N_s}$ class to the separating hyper-plane is equal to the average geometric distance from variables of $T_0^{'}$ class to the separating hyper-plane, and the probability distribution of $T_{0_{tr}}^{N_s}$ class are same as the probability distribution of $T_0^{'}$ class. Then, based on proposition 1, the probability that a variable of ${T_{0_{tr}}}^{N_s}$ class mistakenly labeled as ${T_{1_{tr}}}^{N_s,\lambda_{tr}}$ class is equal to the probability (the probability of false alarm) that a variable of $T_0^{'}$ class is mistakenly labeled as $T_1^{'}$ class. To conclude, it could be expressed as equation:

$$P_e^{N_s,\lambda_{tr}} = \Pr(f\left({T_{0_{tr}}}^{N_s}\right)=1) = \Pr(f\left(T_0^{'}\right)=1) = P_f \tag{28}$$

Each set of parameter $\lambda_{tr}$ and $N_s$ will be used to generate two classes of variable: ${T_{1_{tr}}}^{N_s,\lambda_{tr}}$ and ${T_{0_{tr}}}^{N_s}$. A decision function marked as $f_{N_s,P_f}\left(x\right)$ could be obtained by training PSO-SVM with these two classes of data. Finally, the decision function $f_{N_s,P_f}\left(x\right)$ is stored as a non-linear threshold.

The process to obtain a non-linear threshold is shown in Table 2, and some of typical training results of $f_{N_s,P_f}\left(x\right)$ are shown in Table 3 and Figure 2.

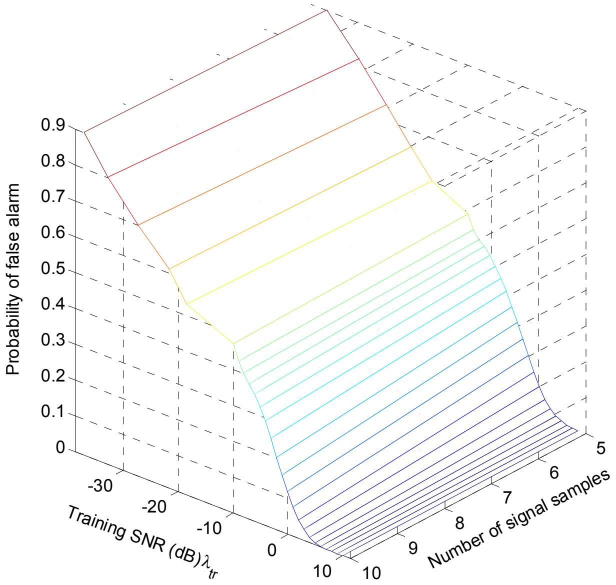
**Table 2.** The process to obtain a non-linear threshold.

| |
|---|
| 1. Generate training signal $s_{tr}\left(n\right)$ and training noise $u_{tr}\left(n\right)$, with $\sigma_{s_{tr}}^2=\lambda_{tr}$, $\sigma_{s_{tr}}^2=1$. |
| 2. Compute two classes of data: ${T_{1_{tr}}}^{N_s,\lambda_{tr}}$, ${T_{0_{tr}}}^{N_s}$ by (24) and (25). |
| 3. Train PSO-SVM with two classes of data: ${T_{1_{tr}}}^{N_s,\lambda_{tr}}$, ${T_{0_{tr}}}^{N_s}$ to obtain a decision function $f\left(x\right)$. |
| 4. Test this $f\left(x\right)$ with the variables of ${T_{0_{tr}}}^{N_s}$ class to obtain $P_e^{N_s,\lambda_{tr}}$, based on Proposition 1 $P_e^{N_s,\lambda_{tr}}=P_f$. |
| 5: Return $f\left(x\right)$ as $f_{N_s,P_f}\left(x\right)$, and store it as non-linear threshold. |

**Table 3.** Typical Training Results.

| $N_s$=5 | | | $N_s$=10 | | |
|---|---|---|---|---|---|
| $\lambda_{tr}$ | $P_f$ | $f_{N_s,P_f}\left(x\right)$ | $\lambda_{tr}$ | $P_f$ | $f_{N_s,P_f}\left(x\right)$ |
| −35.7 dB | 0.9 | $f_{5,0.9}\left(x\right)$ | −37.1 dB | 0.9 | $f_{10,0.9}\left(x\right)$ |
| −29.8 dB | 0.8 | $f_{5,0.8}\left(x\right)$ | −32.8 dB | 0.8 | $f_{10,0.8}\left(x\right)$ |
| −24.9 dB | 0.7 | $f_{5,0.7}\left(x\right)$ | −27.2 dB | 0.7 | $f_{10,0.7}\left(x\right)$ |
| −20.4 dB | 0.6 | $f_{5,0.6}\left(x\right)$ | −21.6 dB | 0.6 | $f_{10,0.6}\left(x\right)$ |
| −16.5 dB | 0.5 | $f_{5,0.5}\left(x\right)$ | −18.4 dB | 0.5 | $f_{10,0.5}\left(x\right)$ |
| −6.5 dB | 0.4 | $f_{5,0.4}\left(x\right)$ | −8.1 dB | 0.4 | $f_{10,0.4}\left(x\right)$ |
| −2.8 dB | 0.3 | $f_{5,0.3}\left(x\right)$ | −4.2 dB | 0.3 | $f_{10,0.3}\left(x\right)$ |
| −0.2 dB | 0.2 | $f_{5,0.2}\left(x\right)$ | −1.7 dB | 0.2 | $f_{10,0.2}\left(x\right)$ |
| 2.3 dB | 0.1 | $f_{5,0.1}\left(x\right)$ | −0.2 dB | 0.1 | $f_{10,0.1}\left(x\right)$ |

**Figure 2.** Training Results of Table 3.



### 3.2.2. Online Module

In the Online module, the proposed system automatically chooses one of the decision functions (according to required number of signal samples $N_s$ and probability of false alarm $P_f$ ) stored in the Offline Module as non-linear threshold to judge whether the actual primary user is present e.g., the required number of signal samples is $N_s = 5$ and probability of false alarm is $P_f = 0.1$, the proposed system would apply decision function $f_{5,0.1}(x)$ as the non-linear threshold.

If a decision function $f_{N_s,P_f}(x)$ is chosen as the nonlinear threshold, Equations (20) and (21) will be converted to:

$$P_f = \Pr(f_{N_s,P_f}(T^{'}) = 1 | H_0) = P_e^{N_s,\lambda_{tr}} \tag{29}$$

$$P_d = \Pr(f_{N_s,P_f}(T^{'}) = 1 | H_1) \tag{30}$$

The result of spectrum sensing is given by:

$$\begin{cases} f_{N_s,P_f}(T^{'}) = 1, \text{ Primary user is presented } (\text{hypothesis } H_1) \\ f_{N_s,P_f}(T^{'}) = -1, \text{ Primary user is not presented } (\text{hypothesis } H_0) \end{cases} \tag{31}$$

### 3.2.3. Comparison with Traditional Energy Detection

Energy detection is the basic sensing method, which was first proposed in [9] and further studied in [5,10]. It does not need any information of the signal to be detected and is robust to unknown dispersive channels. Energy detection compares the normalized average power of the actual received signal plus noise $T_1^{'} = T_1/\sigma_u^2$ variable with the noise power $T_0^{'} = T_0/\sigma_u^2$ to make a decision. To guarantee a reliable detection, the threshold must be set according to the actual noise power $\sigma_u^2$ and the number of samples $N_s$ [9]. The difference between the traditional Energy detection and the proposed system is that the proposed system has a Offline module to obtain decision functions as the non-linear thresholds.

In the Offline module, the system needs a great number of ${T_{1_{tr}}}^{N_s,\lambda_{tr}}$ variables and ${T_{0_{tr}}}^{N_s}$ variables to train PSO-SVM during each training process. Taking the simulation process in this article as an instance, 500 variables from ${T_{1_{tr}}}^{N_s,\lambda_{tr}}$ and the same from ${T_{0_{tr}}}^{N_s}$ are deployed for each training process, of which computational complexity is about $O\left(1000^3\right)$ [16].

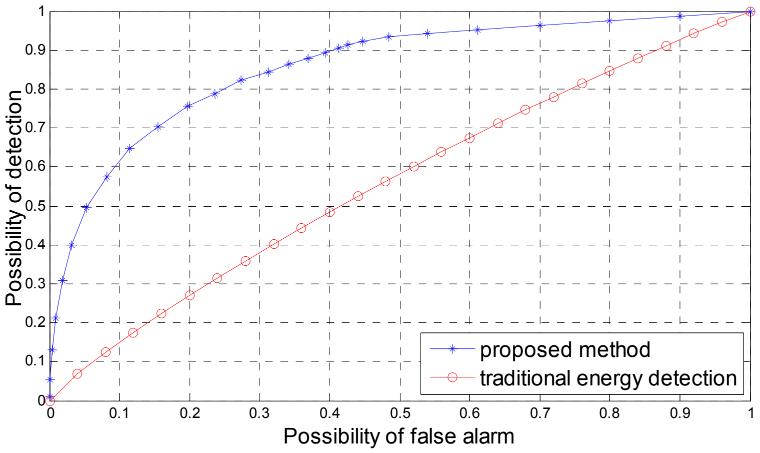
As the price of getting the full list of decision functions, the training times are huge. Therefore the overall computational complexity of the offline module is extremely high. However, in a real spectrum sensing situation, we only take care of the computational complexity in the Online module. The computational complexity of traditional Energy detection needs about $N_s$ multiplications and additions. Hence, the computational complexity of the proposed methods is about $N_s+1$ multiplications and additions, which is competitive with traditional Energy Detection.

## 4. Results and Discussion

In this section, we use the decision functions stored in the Offline module as the nonlinear thresholds to simulate the probability of detection $P_d$ in the Online module.

In Figure 3 we compare the receiver operating characteristic curve of the proposed method and traditional energy detection for actual *SNR* $\lambda=0dB$ and number of signal samples $N_s=5$. If the *SNR* of the actual received signal is low and the number of signal samples is small e.g., $\lambda=0\ dB$ and $N_s=5$, the corresponding spectrum sensing problem based on energy detection is a linearly inseparable binary classification problem. Traditional energy detection with a predefined threshold is a linear classifier, it cannot solve linearly inseparable problem efficiently. But the proposed method is a nonlinear classifier based on PSO-SVM, which can solve linearly inseparable problem efficiently. As shown in Figure 3, the performance of the proposed method is much better than traditional energy detection.

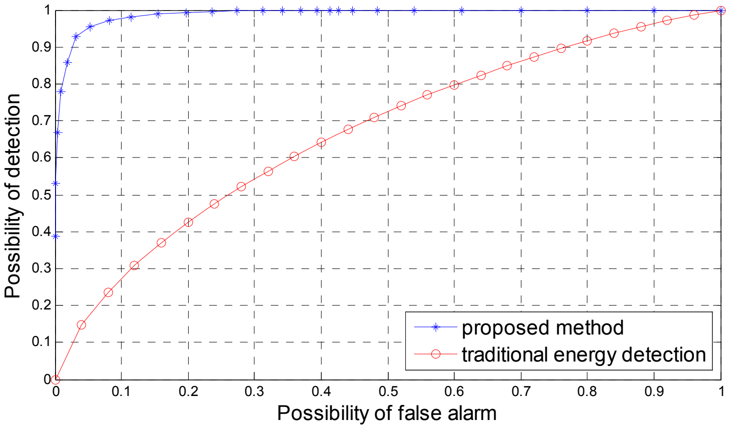
**Figure 3.** Receiver operating characteristic curve of the proposed method and traditional energy detection for number of signal samples $N_s=5$ and actual *SNR* $\lambda=0\ dB$.



In Figure 4 we compare the receiver operating characteristic curve of the proposed method and traditional energy detection in terms of actual *SNR* $\lambda=5\ dB$ and number of signal samples $N_s=5$.

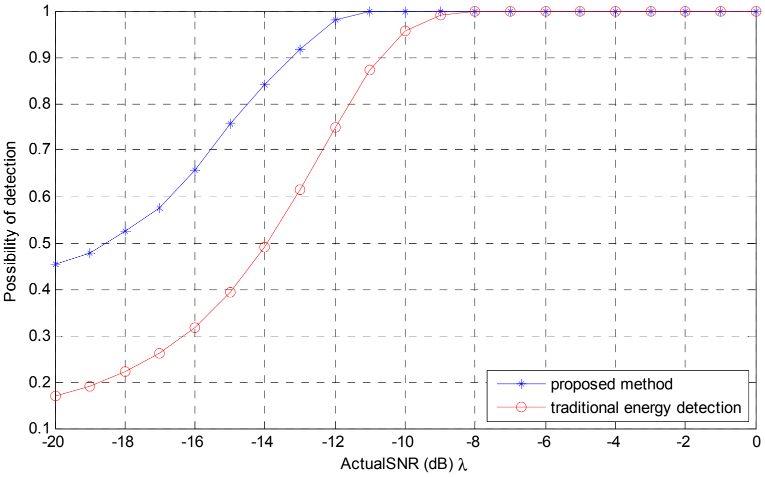
Although the actual *SNR* $\lambda$ increases to 5 *dB* but the number of signal samples $N_s = 5$ is small, which means the corresponding spectrum sensing problem based on energy detection is still linearly inseparable. Therefore, as shown in Figure 4, the performance of proposed method is dramatically better than the traditional energy detection method.

**Figure 4.** Receiver operating characteristic curve of the proposed method and traditional energy detection for $P_d$ for number of signal samples $N_s = 5$ and actual $SNR\ \lambda = 5\ dB$.



In Figure 5 we compare $P_d$ of the proposed method and traditional energy detection for fixed $P_f = 0.1$ and number of signal samples $N_s = 1{,}000$, the corresponding performance of the proposed method is still better than energy detection. This is because although the number of sensing samples is large *i.e.*, 1000, but actual *SNR* $\lambda$ is low, thus the corresponding spectrum sensing problem is linearly inseparable, however the proposed method can classify linearly inseparable problem efficiently. As shown in Figure 5, at $\lambda = -20\ dB$ the proposed method is almost three times better than traditional energy detection. With the $\lambda$ increases the spectrum sensing problem coverage to linearly separable, and the difference of performance between the proposed method and the traditional energy detection also decreases.

**Figure 5.** $P_d$ of the proposed method and traditional energy detection for number of signal samples $N_s = 1{,}000$, $P_f = 0.1$ and actual $SNR\ \lambda = -20\ dB - 0\ dB$.



## 5. Conclusions

In this paper, a novel modular spectrum sensing method for cognitive radio based on PSO-SVM is proposed. It comprises two distinct modules, *i.e.*, Offline and Online. In the Offline module, the decision functions with associated probabilities of false alarm are obtained. In the Online module, the primary user is detected by using the decision functions obtained in the Offline module.

The proposed method actually is independent from the traditional detection method, a nonlinear decision is exploited to replace the linear threshold, which drastically improves the performance of detection without increasing the computational complexity in Online phase. The approach can be used for various signal detection applications without *a priori* knowledge of signals and channels. Simulations have been carried to evaluate the performance of the proposed method. It has been shown that the proposed approach is more effective than the traditional energy detection approach in hostile environments. More specifically, when the received signal samples are lacking and *SNR* is low, the approach proposed in this paper can give a reliable performance, while the traditional energy detection approach is hypodynamic.

In our future research work, we will try to apply the proposed method to enhance more sophisticated detection algorithm which uses predefined linear threshold (e.g., the method proposed in [14]). More specifically, secondary users are located with detected radio map thereby deploying space-time spectrum sensing. And the specific signal pattern from the primary user can be recognized by analyzing the signal detected.

## References

1. *Spectrum Policy Task Force Report*; Technical Report TR 02-155; Federal Communications Commission: Washington, DC, USA, 2002.
2. Mitola, J.; Maguire, G.Q. Cognitive radio: Making software radios more personal. *IEEE Personal Commun*. **1999**, *6*, 13–18,
3. Mitola, J. Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio. Ph.D. Thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, 2000.
4. Haykin, S. Cognitive radio: Brain-Empowered wireless communications. *IEEE J. Sel. Area Commun*. **2005**, *23*, 201–220.
5. Liang, Y.C.; Zeng, Y.H.; Peh, E.C.Y.; Hoang, A.T. Sensing-throughput tradeoff for cognitive radio networks. *IEEE Trans. Wirel. Commun.* **2008**, *7*, 1326–1337.
6. Kay, S. *Fundamentals of Statistical Signal Processing: Detection Theory*, 1st ed.; Prentice-Hall: Englewood Cliffs, NJ, USA, 1998.
7. Quan, Z.; Cui, S.; Sayed, A.H. Optimal linear cooperation for spectrum sensing in cognitive radio networks. *IEEE J. Sel. Top. Signal Process*. **2008**, *2*, 28–40.
8. Taricco, G. Optimization of linear cooperative spectrum sensing for cognitive radio networks. *IEEE J. Sel. Top. Signal Process*. **2011**, *5*, 77–86.
9. Urkowitz, H. Energy detection of unknown deterministic signals. *Proc. IEEE* **1967**, *55*, 523–531.
10. Sonnenschein, A.; Fishman, P.M. Radiometric detection of spread-spectrum signals in noise of uncertainty power. *IEEE Trans. Aerosp. Electron. Syst.* **1992**, *28*, 654–660.

11. Sahai, A.; Cabric, D. Spectrum Sensing: Fundamental Limits and Practical Challenges. In *Proceedings of the IEEE DySpan Conference*, Baltimore, MD, USA, 8–11 November 2005.
12. Gardner, W.A. Exploitation of spectral redundancy in cyclostationary signals. *IEEE Signal Process. Mag.* **1991**, *8*, 14–36.
13. Gardner, W.A.; Brown, W.A.; Chen, C.K. Spectral correlation of modulated signals—Part II: Digital modulation. *IEEE Trans. Commun.* **1987**, *35*, 595–601.
14. Zeng, Y.H.; Liang, Y.C. Spectrum—Sensing algorithms for cognitive radio based on statistical covariances. *IEEE Trans. Veh. Technol.* **2009**, *58*, 1804–1815.
15. Chen, H.S.; Gao, W.; Daut, D.G. Signature Based Spectrum Sensing Algorithms for IEEE 802.22 WRAN. In *Proceedings of the IEEE International Conference on Communications*, Glasgow, UK, 24–28 June 2007; pp. 6487–6492.
16. Vapnik, V.N. *The Nature of Statistical Learning Theory*, 1st ed.; Springer-Verlag: New York, NY, USA, 1995.
17. Chapelle, O.; Haffner, P.; Vapnik, V. Support vector machines for histogram-based image classification. *IEEE Trans. Neural Netw.* **1999**, *10*, 1055–1064.
18. Kuhn, H.; Tucker, A. Nonlinear Programming. In *Proceedings of 2nd Berkeley Symposium on Mathematical Statistics and Probabilistics*, Berkeley, CA, USA, 31 July–12 August 1950; pp. 481–492.
19. Cai, Z.R.; Zhao, H.L.; Jia, M. Spectrum sensing in cognitive radio based on adaptive optimal SVM. *Inform. Technol. J.* **2011**, *10*, 1427–1431.
20. Patnaik, P.B. The noncentral $\chi^2$ and *F* distributions and their applications. *Biometrika* **1949**, *36*, 202–232.

## Appendix: Proof of Proposition 1

If the point $(\mathbf{x}, y)$ is correctly classified by the optimal hyper-plane $y\frac{\left|\langle \mathbf{w}\cdot\mathbf{x}\rangle + b\right|}{\|\mathbf{w}\|_2} \geq \gamma$. In order to facilitate the analysis, we introduce an additional coordinate to expand the sample, the new sample is $\hat{\mathbf{x}} = \left(\mathbf{x}', R\right)'$, $R$ is a constant, $\mathbf{x}'$ is the transpose of $\mathbf{x}$. Similarly we can use bias $b$ to expand the weighted vector *w*, and the expanded *w* is $\hat{\mathbf{w}} = \left(\mathbf{w}', b/R\right)'$, the iteration starts from $\hat{\mathbf{w}} = 0$, if there is a wrong separation, the $\hat{\mathbf{w}}$ is updated accordingly. Suppose $\hat{\mathbf{w}}_{t-1}$ is the weighed vector before the wrong separation:

$$\left\langle \hat{\mathbf{w}}_{t-1}\cdot \hat{\mathbf{x}}_i \right\rangle = y_i\left(\mathbf{w}_{t-1}\cdot \mathbf{x}_i + b_{t-1}\right) \leq 0 \tag{32}$$

where $(\mathbf{x}, y)$ is the data mistakenly separated by $\hat{\mathbf{w}}_{t-1} = \left(\mathbf{w}'_{t-1}, b_{t-1}/R\right)'$ and the $t-th$ update is given by:

$$\hat{\mathbf{w}}_t = \left(\mathbf{w}'_t, b_t/R\right)' = \left(\mathbf{w}'_{t-1}, b_{t-1}/R\right)' + \alpha_i y_i \left(\mathbf{x}'_i, R\right)' = \hat{\mathbf{w}}_{t-1} + \alpha_i y_i \hat{\mathbf{x}}_i \tag{33}$$

where $b_t/R = b_{t-1}/R + \alpha y R$, $\alpha$ is Lagrange multiplier and we can obtain a result:

$$\left\langle \hat{\mathbf{w}}_t \cdot \hat{\mathbf{w}} \right\rangle = \left\langle \hat{\mathbf{w}}_{t-1} \cdot \hat{\mathbf{w}} \right\rangle + \alpha y_i \left\langle \hat{\mathbf{x}}_i \cdot \hat{\mathbf{w}} \right\rangle \geq \left\langle \hat{\mathbf{w}}_{t-1} \cdot \hat{\mathbf{w}} \right\rangle + \alpha\gamma \tag{34}$$

Before the *t-th* mistaken separation there already has $t-1$ wrong separations thus we can obtain:

$$\left\langle \hat{\mathbf{w}}_t \cdot \hat{\mathbf{w}} \right\rangle \geq t\alpha\gamma \tag{35}$$

similarly:

$$\begin{aligned} \left\| \hat{\mathbf{w}}_t \right\|^2 &= \left\| \hat{\mathbf{w}}_{t-1} \right\|^2 + 2\alpha y \left( \hat{\mathbf{w}}_{t-1} \cdot \hat{x} \right) + \alpha^2 \left\| \hat{\mathbf{x}} \right\|^2 \\ &\leq \left\| \hat{\mathbf{w}}_{t-1} \right\|^2 + \alpha^2 \left\| \hat{\mathbf{x}} \right\|^2 \leq \left\| \hat{\mathbf{w}}_{t-1} \right\|^2 + \alpha^2 \left( \left\| \hat{\mathbf{x}} \right\|^2 + R^2 \right) \\ &\leq \left\| \hat{\mathbf{w}}_{t-1} \right\|^2 + 2\alpha^2 R^2 \end{aligned} \tag{36}$$

which means:

$$\left\| \hat{\mathbf{w}}_t \right\|^2 \leq 2t\alpha^2 R^2 \tag{37}$$

and a close form is given by Equation (34) and Equation (35):

$$\left\| \hat{\mathbf{w}} \right\| \sqrt{2t}\alpha R \geq \left\| \hat{\mathbf{w}} \right\| \left\| \hat{\mathbf{w}}_t \right\| \geq \left\langle \hat{\mathbf{w}}_t \cdot \hat{\mathbf{w}} \right\rangle \geq t\alpha\gamma \tag{38}$$

based on Equation (35) and Equation (36):

$$t \propto \frac{1}{\gamma} \tag{39}$$

Proposition 1 is proven.