

## Pruebas de estanqueidad en envases de tereftalato de polietileno basado en máquina de soporte vectorial

*Leak tests on polyethylene terephthalate containers  
based on support vector machine*

Luis Francisco Niño Sierra<sup>1</sup>    Darío Amaya Hurtado<sup>1</sup>    Oscar Fernando Avilés<sup>1</sup>

Recibido 10 de julio de 2014, aceptado 12 de enero de 2015

*Received: July 10, 2014    Accepted: January 12, 2015*

### RESUMEN

En este trabajo se presentan los resultados de la implementación de un algoritmo basado en Máquina de Soporte Vectorial (SVM por sus siglas en inglés, *Support Vector Machine*), sobre la plataforma de National Instruments Compaq Rio, abreviado por el fabricante cRio, para la detección de fugas en envases de tereftalato de polietileno (PET por sus siglas en inglés, *polyethylene terephthalate*). Para cumplir con esto se realiza la adquisición y procesamiento de señales de presión diferencial para luego mediante un modelo de decisión determinar si el envase presenta fuga o no. Existen muchas herramientas de procesamiento digital de señales, sin embargo en aplicaciones de clasificación las SVM han demostrado ser una muy buena alternativa. Las SVM requieren de un proceso de entrenamiento, que se realiza a partir de señales de las que se conoce la clase a la que pertenecen. Luego de su entrenamiento se realiza un programa mediante el cual se realiza la clasificación de una nueva botella. El algoritmo final es implementado sobre el sistema Compac Rio de National Instruments, para aplicación en línea

Palabras clave: Máquina de soporte vectorial, procesamiento, clasificación, prueba de estanqueidad, tereftalato de polietileno.

### ABSTRACT

*In this paper we present the results of implementing an algorithm based on Support Vector Machine (SVM) on the National Instruments platform Compaq Rio, abbreviated cRio for the detection of Polyethylene terephthalate (PET) containers leakage. To accomplish this, the acquisition and processing the differential pressure signal is performed and then using a decision model to determine whether the package leaks or not. There are many tools for digital signal processing, however in the SVM classification applications have proven to be a very good alternative. The SVM requires a training process, performed based on signals which are known class to which they belong. After this training, a classification is performed in a new bottle. The final algorithm is implemented on the cRio National Instruments system for online application.*

*Keywords: Support vector machine, prosecution, classification, tightness, polyethylene terephthalate.*

### INTRODUCCIÓN

La industria colombiana es cada día más competitiva, los tratados de libre comercio firmados en los últimos años hacen que este sector esté en camino

de implementar procesos de producción de alta tecnología, con los que la productividad, calidad y economía estén a nivel de las mejores compañías del mundo. El sector de los plásticos y en particular el de la fabricación de envases no se escapa de esta nueva

---

<sup>1</sup> Programa de Ingeniería en Mecatrónica. Universidad Militar Nueva Granada. Cr 11 N° 101-80. Bogotá, D.C., Colombia.  
E-mail: luis.nino@unimilitar.edu.co; dario.amaya@unimilitar.edu.co; oscar.aviles@unimilitar.edu.co

tendencia mundial. En Colombia hay gran cantidad de empresas dedicadas a la transformación de materia prima en productos de consumo masivo, como es el caso de los envases PET. Una de las pruebas de calidad que debe garantizar la fabricación de este tipo de envases, en especial cuando el uso final es como contenedor de sustancias líquidas como agua, aceite, gaseosas y demás bebidas, es la de no tener ningún tipo de fuga. De ahí la importancia para este sector de contar con sistemas que realizan la verificación de la estanqueidad del producto final.

El sistema desarrollado se puede dividir en dos etapas: La primera consiste en la adquisición de señales, en este caso, señales de presión diferencial medida entre el interior del envase y un punto de referencia a presión conocida y constante. Esta medición se realiza durante tres segundos, a una frecuencia de muestreo de 100 Hz obteniendo un vector de 300 datos para cada envase. La segunda etapa consiste en el procesamiento digital de las señales adquiridas, etapa en la que mediante la SVM y un proceso de votación se determina si existe o no una fuga en el envase.

Dos tipos de sistemas de detección pueden encontrarse, el primero *off line* en el que se toma una muestra aleatoria de envases que son probados en el laboratorio. El segundo tipo, *on line*, consiste en un sistema de detección que se incluye en la línea de producción y por tal razón prueba todos los envases fabricados. Este tipo de sistemas debe tener la capacidad de trabajar al mismo ritmo que el sistema o máquina que fabrica los envases, por lo tanto requiere herramientas de procesamiento rápidas, como es el caso de algoritmos sobre plataformas de trabajo en tiempo real. Otra característica de los sistemas de detección *on line* es la de permitir su ensamble o acople con la máquina que produce los envases. El sistema desarrollado y presentado en este trabajo corresponde a esta clase.

Cuando se trata de envases de PET la señal a procesar se obtiene en la mayoría de los casos, presurizando con aire el envase y midiendo el decaimiento de la presión en un breve lapso de tiempo, sin embargo los fabricantes no publican las técnicas de procesamiento empleadas, esto debido a la ventaja competitiva que esto representa en especial cuando se trata del sector comercial. En otras áreas, como es el caso

del transporte del petróleo, transporte de agua o el proceso de la refinación del petróleo, se encuentran muchas técnicas de procesamiento digital aplicadas a la detección de fugas. Algunas de ellas son: redes neuronales [3-4], wavelets [4-5], teoría de control difuso [6-8], entropía [1], Lyapunov [11], autocorrelación [1-2], teoría de caos [9], predictor ARX [10], interferometría óptica [12-13], entre otras.

## MÁQUINA DE SOPORTE VECTORIAL

Una máquina de soporte vectorial (SVM) es una herramienta de clasificación de datos, que permite encontrar fronteras lineales a partir de no lineales.

Esto se logra a partir de una transformación del espacio al que pertenecen los datos a un espacio de dimensión mayor, por ejemplo si los datos pertenecen a  $\mathfrak{R}^2$  entonces la SVM los mapea en un espacio de dimensión 3 o  $\mathfrak{R}^3$ , en este nuevo espacio las fronteras de separación pueden ser vistas como rectas o superficies de decisión, linealmente separables [14]. Es decir, la SVM realiza la siguiente transformación, ecuación (1):

$$\mathfrak{R}^m \rightarrow \mathfrak{R}^n \quad || n \geq m \quad (1)$$

En donde  $m$ , es la dimensión del espacio de los datos y  $n$  es la dimensión del espacio de los datos mapeados por la SVM. Suponiendo un conjunto  $X$  de  $N$  datos, correspondiente a las muestras tomadas para ser clasificadas en una de dos clases, en donde cada dato  $x \in \mathfrak{R}^m$ , con  $m=2$ , es decir, los elementos de  $x$  son vectores de la forma  $x=(t,f(t))$ . Siendo  $y$  el conjunto de las dos posibles clases a las cuales pueden pertenecer los datos, se puede definir que  $y \in (1, -1)$ , en donde el valor 1 representa una clase y el valor  $-1$  representa la otra, el conjunto de todos los valores de  $y$  se conoce como vector de etiquetas, es decir,  $y=(y_1, y_2 \dots y_n)$ .

Se puede definir un nuevo conjunto de la forma  $(x_1, y_1), (x_2, y_2), (x_3, y_3) \dots (x_n, y_n)$ , en donde los vectores  $x$  corresponden a los datos a clasificar, mientras que los valores de  $y$  son escalares que definen la clase a la que pertenece el vector  $x$  correspondiente. Estos datos definen un hiperplano de la forma que se indica en la ecuación (2) [14]:

$$\{x : f(x) = x^T \beta + \beta_0 = 0\} \quad (2)$$

en donde,  $\beta$  es un vector unitario y  $\beta_0$  es un desplazamiento o *Bias*. De esta manera  $f(x)$  induce una regla de decisión, ecuación (3).

$$G(x) = \text{sign}(f(x)) = \text{sign}(x^T \beta + \beta_0) \quad (3)$$

La distancia  $M$ , de  $x$  al hiperplano de separación se representa en la Figura 1 y está dada por la ecuación (4):

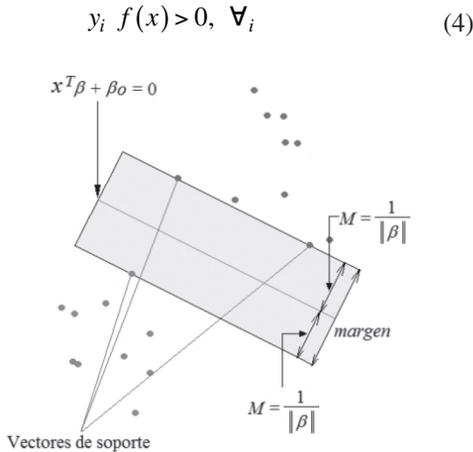


Figura 1. Representación del hiperplano de separación.

Si se trata del caso linealmente separable, el problema se transforma en maximizar el margen  $M$  entre los puntos y las clases  $+1$  y  $-1$ . Es decir, se convierte en un problema de optimización, el cual puede ser expresado por la ecuación (5):

$$\max_{\beta, \beta_0, \|\beta\|=1} M \quad (5)$$

Sujeto a la siguiente restricción:

$$y_i(x^T \beta + \beta_0) \geq M, i = 1, 2, 3 \dots n \quad (6)$$

El cual puede transformarse en un problema más sencillo por el principio de dualidad [15], de la forma, ecuación (7):

$$\min_{\beta, \beta_0} \|\beta\| \quad (7)$$

Sujeto a:

$$y_i(x^T \beta + \beta_0) \geq 1, i = 1, 2, 3 \dots n \quad (8)$$

Todos los puntos que cumplen con la ecuación (7), sujetos a la restricción  $y_i(x^T \beta + \beta_0) = 1$  son llamados vectores de soporte, ver Figura 1. Este es el modo más usual de escribir el criterio de separación de datos basado en vectores de soporte [14]. El entrenamiento de una SVM, consiste en obtener tales vectores de soporte y  $\beta_0$ , a partir de datos de los que se conoce la clase a la que pertenecen.

Cuando se trata de un caso no lineal, este puede verse como un caso en que las clases se traslapan [14]. Una manera de analizar este problema es maximizando aún más  $M$ , pero permitiendo que algunos elementos queden clasificados erróneamente, es decir, que el proceso de clasificación permita algún margen de error. Para esto, se define una nueva variable de holgura  $\xi_i$  que representa la distancia de traslape de cada uno de los datos, y  $\xi = (\xi_1, \xi_2 \dots \xi_n)$  es el conjunto de todas las distancias de traslape. En este caso la restricción presentada en la ecuación (6) puede presentarse de dos formas, como se muestra en las ecuaciones (9) y (10).

$$y_i(x^T \beta + \beta_0) \geq M - \xi_i \quad (9)$$

$$\begin{aligned} y_i(x^T \beta + \beta_0) &\geq M(1 - \xi_i); \forall_i, \xi_i \\ &\geq 0, \sum_{i=1}^N \xi_i \\ &\leq \text{constante} \end{aligned} \quad (10)$$

Las dos formas conllevan a soluciones diferentes. En (9), las clases se traslapan una distancia  $\xi_i$  en el margen, sin embargo su solución resulta en un problema de optimización no convexo. Mientras que en (10) se representa la distancia de superposición relativa, que cambia con el ancho del margen  $M$ , su solución resulta en un problema de optimización convexo [16], razón por la cual es la que generalmente se usa. En este caso,  $\xi$  es la cantidad proporcional por la cual la predicción está en el lado equivocado del margen y  $\sum \xi_i$  es la cantidad proporcional total por la cual la predicción está en el lado equivocado del margen. En (11) se muestra el problema de optimización mencionado

$$\min \|\beta\| \quad (11)$$

Sujeto a

$$\xi_i \geq 0, \sum \xi_i \leq \text{constante}, y_i(x^T \beta + \beta_0) \geq (1 - \xi_i), \forall_i$$

Esta forma más usual del clasificador de soporte vectorial para caso no lineal.

### Solución del problema de optimización mediante multiplicadores de Lagrange

El problema de optimización presentado en (11) puede entenderse como una solución de programación cuadrática mediante el uso de multiplicadores de Lagrange [15], y puede reescribirse de forma equivalente de la siguiente forma, ecuación (12):

$$\min_{\beta, \beta_0} \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^N \xi_i \quad (12)$$

Sujeto a

$$\xi_i \geq 0, y_i(x^T \beta + \beta_0) \geq (1 - \xi_i), \forall_i$$

En donde se introduce un parámetro C, que es el costo y que reemplaza la constante presentada en la ecuación (10). Introduciendo los multiplicadores lagrangianos positivos  $\alpha_i$ , con  $i = 1, 2, \dots, N$ , se tiene la formulación de lagrangiano primal, como se indica en la ecuación (13).

$$\begin{aligned} L_P = & \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^N \xi_i \\ & - \sum_{i=1}^N \alpha_i [y_i(x^T \beta + \beta_0) \\ & - (1 - \xi_i)] - \sum_{i=1}^N u_i \xi_i \end{aligned} \quad (13)$$

Se requiere, entonces, minimizar  $L_P$  respecto de  $\beta, \beta_0$  y además que todas las derivadas de  $L_P$  respecto de  $\alpha_i$  sean cero. Lo anterior es un problema cuadrático convexo, lo que significa que se puede resolver el problema dual de maximizar  $L_P$  sujeto a que el gradiente de  $L_P$  respecto de  $\alpha_i$  sea cero y también sujeto a que  $\alpha_i \geq 0$ . Esta definición es conocida como el Wolfe Dual [16], que tiene la propiedad de que el máximo de  $L_P$  se presenta en

los mismos valores de  $\beta, \beta_0$  y  $\alpha_i$  que el mínimo, ambos sujetos a sus correspondientes restricciones.

Que el gradiente de  $L_P$  mencionado sea cero, conlleva a:

$$\beta = \sum_{i=1}^N \alpha_i y_i x_i \quad (14)$$

$$0 = \sum_{i=1}^N \alpha_i y_i \quad (15)$$

Sustituyendo estas condiciones en (13) se obtiene el lagrangiano dual ( $L_D$ ) para el caso lineal (16).

$$L_D = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i^T \cdot x_j) \quad (16)$$

Tanto  $L_P$  como  $L_D$  parten de la misma función objetivo, ecuación (11), tienen diferentes restricciones y la solución puede obtenerse ya sea minimizando  $L_P$  o maximizando  $L_D$ .

Es de notar que hay un multiplicador de Lagrange por cada punto de entrenamiento. Los vectores de soporte serán los puntos para los cuales se obtiene la solución  $\alpha_i \geq 0$ . Los vectores de soporte serán fundamentales al momento de la implementación de la SVM, debido a que se encuentran lo más cerca posible al umbral de decisión. Si los vectores de soporte son removidos la solución cambia, mientras que si otros puntos no son incluidos en el entrenamiento, la solución es la misma.

Para el caso no lineal, la solución viene de funciones Kernel, las cuales son funciones  $K : \mathfrak{R}^m \times \mathfrak{R}^n \rightarrow \mathfrak{R}$  tal que:

$$K : (x_i, y_i) \rightarrow \phi(x_i) \cdot \phi(x_j) \quad (17)$$

y que representan el producto escalar en el espacio de las características de cualquier dimensión [17-18]. Las funciones  $K$  deben cumplir con el teorema de Mercer, para ser usadas como productos punto y en consecuencia como funciones Kernel.

### Aplicación de la SVM

De la ecuación (14) se puede obtener la función solución de la forma mostrada en (18).

$$h(x) = h(x)^T \beta + \beta_0 = \sum_{i=1}^N \alpha_i y_i K(x_i, x_j) + \beta_0 \quad (18)$$

El proceso de entrenamiento consiste, entonces, en encontrar los vectores de soporte, los valores de los multiplicadores de Lagrange  $\alpha_i$  y los valores de  $\beta_0$  o *bias*. Sin embargo, debido a las restricciones de ecuación (16), en especial  $\alpha_i > 0$ , la función en ecuación (18) no requiere evaluarse en todos los puntos del espacio de características, basta con evaluarse en los vectores de soporte y aplicar la función de decisión ecuación (3) para clasificar un nuevo dato, como se presenta en la ecuación (19).

$$\begin{aligned} \hat{G}(x) &= \text{sign}(x^T \hat{\beta} + \hat{\beta}_0) \\ &= \text{sign}\left(\sum_{i=1}^{N_{VS}} \alpha_i y_i K(x_i, x_j) + \beta_0\right) \end{aligned} \quad (19)$$

Teniendo en cuenta lo anterior, el entrenamiento puede llevarse a cabo con herramientas informáticas como Matlab, y luego ponerse en funcionamiento con datos nuevos que requieran ser clasificados.

### Adquisición de señales

La adquisición de datos se realiza mediante la tarjeta NI6009 y el sensor de presión diferencial MPX2200P, a una frecuencia de muestreo de 100 Hz. El proceso para la adquisición es el siguiente: Primero se inyecta aire dentro de la botella durante un tiempo de un segundo, luego se sella la botella y se mide, durante dos segundos, la presión diferencial entre un punto de referencia y el interior del envase. Adicionalmente, se deja un segundo de inactividad al inicio, con el fin de tener un tiempo de sincronismo con la máquina que produce las botellas. En total se utilizaron cuatro botellas de la siguiente manera:

- Botella 1 en perfecto estado
- Botella 2 con una fuga menor a 1 mm de diámetro (0,97 mm aprox.)
- Botella 3 con fuga menor a 0,5 mm de diámetro (0,46 mm aprox.)
- Botella 4 con fuga menor a 0,2 mm de diámetro (0,15 mm aprox.)

En la Figura 2 se muestran las señales obtenidas. Con el número 1 se indican las señales que corresponden a botellas con fuga de 1mm aproximadamente, las

señales indicadas con el número 2 corresponden a botellas con fuga menor a 0,5 mm, por último con el número 3 se indican señales tomadas a envases con fuga menor a 0,2 mm y envases en perfecto estado, es decir, se traslapan o superponen las señales de estos dos casos.

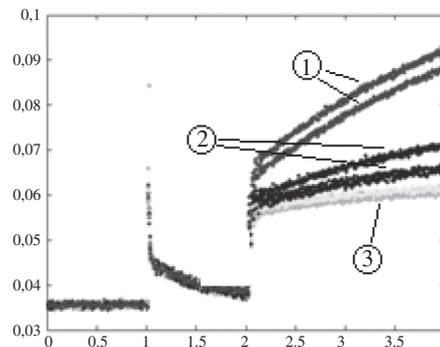


Figura 2. Adquisición de señales de presión diferencial.

### Entrenamiento de la SVM

Para el entrenamiento de la SVM se tomaron los datos de la señal durante los dos últimos segundos, que corresponden a un tiempo de observación una vez sellada la botella. Se eliminaron 20 ms de señal en que presentan oscilaciones justo al inicio de este periodo, debido al cierre de la válvula de sellado. El proceso de entrenamiento se llevó a cabo en Matlab®, se logró la convergencia del proceso con las señales de la botella sin defectos y la de fuga menor a 0,5 mm. El resultado del entrenamiento se muestra en la Figura 3.

El entrenamiento se llevó a cabo con Kernel lineal, ver Figura 3(a) y con Kernel gaussiano, como se presenta en la Figura 3(b). Sin embargo, la implementación en la plataforma de tiempo real se realizó con los resultados obtenidos del entrenamiento con Kernel lineal. Los vectores de soporte y demás datos necesarios para la implementación, luego del proceso de entrenamiento son los siguientes:

$$VS = \begin{Bmatrix} (-1,5471, -1,3226), \\ (-0,5669, -0,6640), \\ (-1, 6432, -0,8522) \end{Bmatrix}$$

$$\alpha = [8,5748 \quad 1,5682 \quad -10,1429]^T$$

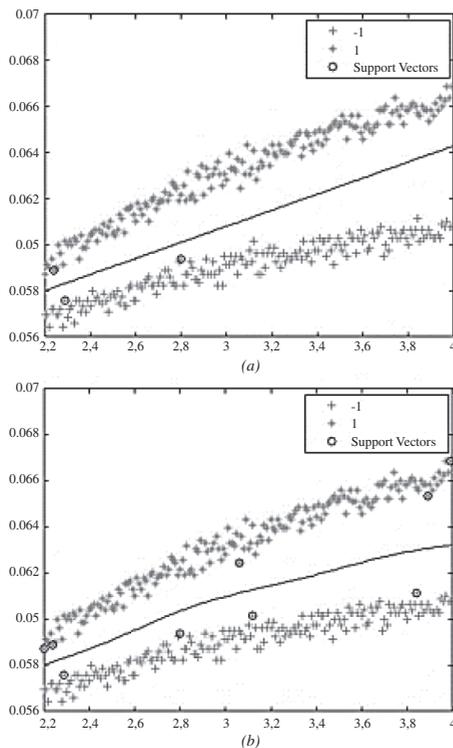


Figura 3. Resultado de entrenamiento de la SVM.  
(a) Kernel Lineal, (b) Kernel Gaussiano.

$$\beta_0 = [-0,0586]$$

$$f_{Esc} = [1,9219 \quad 369,6113] \text{ (Factor de escala)}$$

$$f_{Des} = [-3,0950 \quad -0,0612] \text{ (Factor de desplazamiento)}$$

Con estos valores se realizaron pruebas *off line*, para una botella con fuga menor a 0,5 mm los resultados se muestran en la Figura 4.

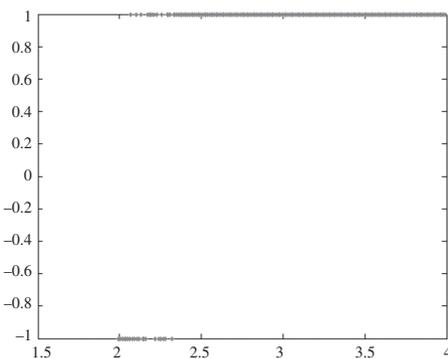


Figura 4. Resultado de la prueba *off line* para envases con fuga menor a 0,5 mm.

Se observa que algunas de las muestras son clasificadas erróneamente, esto se presenta durante los primeros 20 ms, ya que fueron eliminados para lograr la convergencia de la SVM.

Para tomar la decisión sobre el estado de la botella se lleva a cabo un proceso de “votación”, aprovechando que el resultado de la evaluación de un dato es 1 o -1, por medio de la sumatoria de los resultados de todos los datos se puede establecer una competencia entre las dos clases. Se estableció que una botella que tenga 150 “votos” a favor se considera en buen estado, teniendo en cuenta que el total de datos a evaluar es de 200, significa que 50 “votos” fueron en contra de esta clase. Con lo anterior se obtiene que el umbral de “votos” para la clase “botellas buenas” es de 100. Sin embargo, para el caso de botellas defectuosas se define el umbral en 0, lo que significa que si se tiene igual cantidad de votos a favor que en contra se determina que la botella tiene fuga.

El algoritmo de la SVM con Kernel lineal fue implementado en la plataforma cRio de *National Instruments*, se realizaron pruebas a envases diferentes a los usados en el entrenamiento, logrando la correcta clasificación en casos de fugas superiores a 0,5 mm

## CONCLUSIONES

La máquina de soporte vectorial como herramienta de procesamiento para la determinación de las botellas con fugas resulta ser una buena elección, en especial porque la evaluación final se realiza por medio de un proceso de votación a favor o en contra de una de las dos clases establecidas. Es decir, todos los datos obtenidos de la señal de presión aportan información acerca del estado de la botella. Por el contrario, cuando se emplea un valor umbral de la señal de presión para tomar la decisión, solamente un dato, la presión medida después de un tiempo de espera, determina si la botella es buena o no. En este caso el comportamiento durante el tiempo de espera no aporta información alguna.

El proceso de entrenamiento de la SVM es rápido y requiere básicamente de las señales de dos botellas, una en perfecto estado y otra que presente la fuga más pequeña que se desea detectar. La implementación de una SVM con Kernel lineal puede llevarse a

cabo sobre sistemas de control de tipo industrial como PLC, su complejidad depende de la cantidad de vectores de soporte requeridos para diferenciar las clases.

Cuando se trata de una SVM con Kernel no lineal, su implementación en tiempo real puede convertirse en un problema muy complejo, que requiere de sistemas con herramientas de procesamiento digital de señales avanzado, haciéndolo costoso y en algunos casos, como el de esta aplicación, no viable económicamente.

En el caso de la detección de microfugas la SVM debe acompañarse de un sensor de presión diferencial de un rango menor, como se mencionó en la sección anterior, sin embargo el sector industrial cuenta con transmisores de presión diferencial de rango programable, los que evitan el tener que cambiar el sensor para cada aplicación.

La frecuencia de muestreo de 100 Hz permite la utilización de sistemas de procesamiento convencionales como PLC de gama media optimizando aún más los recursos, con lo que es posible pensar en su implementación en procesos industriales.

El proceso de puesta en funcionamiento de este sistema en una producción de envases en línea, puede llevarse a cabo rápidamente sin necesidad de grandes modificaciones en la infraestructura de la empresa, además, teniendo en cuenta que la SVM debe entrenarse con señales obtenidas in situ, se convierte en una solución a la medida, garantizando su aplicación en la mayoría de los casos. Esto es muy importante teniendo en cuenta el carácter industrial de este problema

#### AGRADECIMIENTOS

Los autores agradecen el apoyo al personal del Laboratorio de Automatización y Control de la Universidad Militar Nueva Granada y a los integrantes de los grupos de investigación GAV y DAVINCI.

#### REFERENCIAS

[1] Y. Jin, W. Yumei and L. Ping. "Approximate Entropy-based Leak Detection Using Artificial Neural Network in Water Distribution

Pipelines". 2010 11th Int. Conf. Control, Automation, Robotics and Vision Singapore. December 7-10, 2010.

[2] G. Bhattacharyya and R. Johnson. "Statistical Concepts and Methods". John Wiley & Sons. 1977.

[3] S. Haykin. "Neural Networks: A Comprehensive Foundation". Prentice Hall. 1998.

[4] A. Avelino, J. de Paiva, R. da Silva, G. de Araujo, F. de Azevedo, F de O Quintaes, A. Maitelli, A Neto and A. Salazar. "Real time leak detection system applied to oil pipelines using sonic technology and neural networks". Industrial Electronics. 2009. IECON '09. 35th Annual Conference of IEEE, pp. 2109-2114. November 3-5, 2009.

[5] W. Ming. "Application of Wavelet to Detect Pipeline Leak Point". Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06) 2006.

[6] X. Li and L. Shu. "Fuzzy Theory Based Support Vector Machine Classifier". Fuzzy Systems and Knowledge Discovery. 2008. FSKD '08. Fifth International Conference on. Vol. 1, pp. 600-604. October 18-20, 2008.

[7] L. Yunhu, C. Mian, J. Yan and H. Hui. "Identification of leak zone pre-drilling based on fuzzy control". Power Engineering and Automation Conference (PEAM). IEEE. Vol. 3, pp. 353-356. September 8-9, 2011.

[8] X. Tongtai. "Lost Circulation Prevention And Plugging Technology In Drilling [M]". Beijing: Petroleum Industry, pp. 35-50. 1997.

[9] T. Tani, T. Nagasako, Y. Fujimoto, T. Iokibe and T. Yamaguchi. "Chaos Information Criteria to Detect High-pressure Gas Leak in Petroleum Refining Plant". SICE-ICASE International Joint Conference 2006. Bexco, Busan, Korea. October 18-21, 2006.

[10] C. Vaz, J. de Medeiros and O. de Queiroz Fernandes. "ARX Modeling Approach to Leak Detection and Diagnosis". Journal Of Loss Prevention In The Process Industries. Vol. 23, pp. 462-475. 2010.

[11] K. Abhulimen and A. Susu, "Liquid Pipeline Leak Detection System: Model Development and Numerical Simulation". Chemical Engineering Journal. Vol. 97, pp. 47-67. 2004.

- [12] A. Jourdain, P. de Moor, S. Pamidighantam and H. Tilmans. “Investigation of the Hermeticity of BCB-sealed Cavities for Housing (RF-) MEMS Devices”. In Proc. 15th IEEE Int. Conf. Micro Electro Mechanical Systems, pp. 677-680. Las Vegas, NV, USA. January, 2002.
- [13] Y. Tao and A. Malshe. “Theoretical Investigation On Hermeticity Testing Of MEMS Packages Based On MIL-STD-883E”. Microelectron. Rel. Vol. 45, Issue 3-4, pp. 559-566. 2005.
- [14] T. Hastie, R. Tibshirani and J. Friedman. “The Elements of Statistical Learning Data Mining, Inference, and Prediction”. 2nd edition. Springer. 2011.
- [15] N. Cristianini and J. Shawe-Taylor. “An Introduction To Support Vector Machine And Other Kernel-based Learning Methods”. Cambridge University Press. New York, USA. 2000
- [16] R. Fletcher. “Practical Methods of Optimization”. 2nd edition. Wiley. New York, USA. 2000.
- [17] V. Vapnik. “Statistical Learning Theory”. Wiley. New York, USA. 1988.
- [18] V Kecman. “Learning and Soft Computing”. MIT Press. London, England. 2001.