2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

A Locally Adaptive Boundary Evolution Algorithm for Novelty Detection Using Level Set Methods

Xuemei Ding, Yuhua Li, Ammar Belatreche, and Liam Maguire

***Abstract*—This paper proposes a new locally adaptive boundary evolution algorithm for level set methods (LSM)-based novelty detection. The proposed approach consists of level set function construction, boundary evolution, and evolution termination. It utilises the exterior data points lying outside the decision boundary to effect the segments of the boundary that need to be locally evolved in order to make the boundary better fit the data distribution, so it can evolve boundary locally without requiring knowing explicitly the decision boundary. The experimental results demonstrate that the proposed approach can effectively detect novel events as compared to the reported LSM-based novelty detection method with global boundary evolution scheme and four representative novelty detection methods when there is an exacting error requirement on normal events.**

I. Introduction

NOVELTY detection refers to identifying unforeseen anomalies that deviate from normal behaviours. It can be described by a decision boundary around an expected fraction of the given normal data. Any unseen data point lying outside the boundary is considered abnormal.

Novelty detection is especially important for safety-critical environments such as jet engines [1] where anomalies are rare and knowledge about novel conditions is unavailable. A number of commonly used novelty detection techniques have been reported such as those based on support vector machines (SVM), nearest neighbours (NN), clustering, and statistics. Recently, Ding et al. [2] proposed a LSM-based novelty detection approach in order to address some of the challenges experienced by the commonly used techniques. The authors' approach uses some inherent features such as: 1) It can construct a nonlinear decision boundary directly in the input space using an implicit level set function (LSF); 2) The boundary can be evolved (shrunk/expanded) in order to tightly enclose the given data; 3) It can smooth and manage the boundary shape, e.g. boundary merging and splitting; and 4) It does not need any assumptions on the given data distribution, hence a nonparametric approach. However, the approach has the following limitations: 1) The data points on the boundary and their associated normal vectors are required in order to identify the location of a given data point relative to the boundary, this involves an isosurface reconstruction which is still a challenging and ongoing research topic for high dimensional data; and 2) The boundary evolution is global, i.e. all segments of the boundary have the same evolution direction and the same speed at each evolution iteration. As a result some segments of the boundary closely fit the nearby data points, whereas other segments may not.

To remedy the above two limitations, this paper extends this work and proposes a novel algorithm that can identify the location of a given data point relative to the boundary using the sign of the LSF at that point rather than involving the isosurface reconstruction and can locally evolve the boundary according to the distribution of the exterior points lying outside the boundary in order to make the boundary shape better fit the given data distributions especially when there is an exacting error requirement on normal data.

The remainder of the paper is organised as follows: Section II briefly reviews the recent related works. Section III presents the proposed locally adaptive LSM-based novelty detection approach. Section IV evaluates the proposed approach. Section V concludes the paper and discusses the potential extensions.

II. Related Work

There have been a number of reviews of novelty detection methods from differing theoretical perspectives, e.g. statistics and neural networks, data mining, machine learning, and from different application domains such as intrusion detection for cyber-security, fraud detection for credit cards, image processing, health care, fault detection in safety-critical systems etc. [3,4,5,6,7]. Before presenting the proposed approach, we provide a brief review of the commonly used unsupervised novelty detection techniques that use normal data only, much like the learning process used in this paper.

One technique is based on SVM which utilises a 'kernel trick' of mapping a nonlinear problem in the input space into a linear problem in a feature space. For example, one-class SVM (OCSVM) proposed by Schölkopf et al. constructs a hyper-plane to separate the normal data from the origin with maximum margin in a feature space [8]. Some researchers explored pre-processing methods before applying the OCSVM in order to improve the learning performance for big datasets, e.g. a data selection strategy in [9] , a dimension reduction tool in [10]. Additionally, Tax and Duin [11] proposed a support vector data description (SVDD) by finding a minimum volume hyper-sphere surrounding the normal data points in a feature space. Some improvements on SVDD have been developed to address such issues like over-fitting, computational time and memory requirements, e.g. [12,13]. SVM-based techniques have been applied in many domains, e.g. OCSVM applied to identifying

Xuemei Ding is with the Fujian Normal University, Fuzhou, China, 350108 , and University of Ulster, Londonderry, UK, BT48 7JL (corresponding author to provide phone/fax: +86 (0)591 22868468; e-mail: xuemeid@fjnu.edu.cn; ding-x@email.ulster.ac.uk).

Yuhua Li, Ammar Belatreche, and Liam Maguire are with the University of Ulster, Londonderry, UK, BT48 7JL (e-mail: {y.li, a.belatreche, lp.maguire}@ulster.ac.uk).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

deterioration in emergency department patients [14], and SVDD applied to loudness scaling in hearing aid devices [15]. The weakness of the techniques is that, after the inverse mapping from the feature space, the decision boundary may not enclose the data points tightly in the input space [16].

The second technique is based on NN in which normal data are assumed to be lying near to their neighbours, whereas potential anomalies are far away from their neighbours. It is a very simple and effective method and it does not make any assumptions on the data distribution. However, it needs an appropriate distance threshold to measure a potential novelty and also it needs to store all the training data in order to further measure the dissimilarity with new unseen data. Recent work on NN-based novelty detection centres around: devising a more effective detection strategy [17], and reducing the size of stored data required while maintaining detection accuracy[18].

The third technique is based on clustering, which refers to the gathering of training data into some clusters. If a new unseen data point belongs to any of the clusters, then it is considered normal, otherwise it is considered abnormal. As this technique treats a complex type of data by simply plugging in a clustering algorithm, the performance of the technique is determined by the chosen method for capturing the clustered structure of the given data. Therefore, the distance or density measurement used by the clustering algorithm needs to be optimised and remains a challenge for this type of technique. Recent related works were reported such as [19,20].

The fourth technique models the data distributions of a given class using their statistical properties. The established model is then used to estimate whether the probability of test data belongs to such distributions or not. Gaussian mixture (GM) probabilistic distribution is often used in this category for some applications such as acoustic surveillance [21]. Additionally, Filippone and Sanguinetti [22] proposed an information theoretic approach for novelty detection. Benezeth et al. [23] estimated a statistical model from a normal behaviour video sequence, the abnormal behaviour can then be detected using low-level features whenever the observed pattern is unlikely under the normal activity model following a likelihood ratio test. In this category, parametric approaches assume a particular distribution of data and estimate parameters of the distribution from the normal data. However, these assumptions are not often true in real world applications and they need extensive a priori knowledge of the problem. While nonparametric approaches are more flexible and they do not require any assumptions, the performance is yet decided by the selection of the free parameters (e.g. smoothing bandwidth).

Ding et al. [2] recently applied LSM to novelty detection in order to address some of the above challenges. Their level set boundary description (LSBD) method aims to construct a decision boundary that tightly encloses an expected fraction of the given normal data. The inherent strengths are that the boundary can be constructed directly in the input space and can be evolved to fit the data distribution; and the method does not need any assumptions on the given data. However, the weaknesses are that its implementation involves an isosurface reconstruction, and the boundary is evolved in terms of a global scheme rather than a locally adaptive scheme. To address those weaknesses, a new locally adaptive boundary evolution algorithm without the burden of isosurfacing is developed for novelty detection using LSM.

III. LSM-Based Novelty Detection With Local Boundary Evolution

The proposed approach consists of three components: LSF construction, boundary evolution, and evolution termination.

A. LSF Construction

LSM employ an implicit function, LSF, to represent a complicated boundary of co-dimension one with the zero level set (ZLS) of the function. The boundary can then be evolved using time-dependent partial differential equations that govern the dynamics of the boundary motion. Therefore, the first step of the approach is to construct a LSF. The Laplacian of kernel density estimations (KDE) on the given data is employed as [24] such that the KDE is

$$KDE(\vec{x}) = \frac{1}{Nh^d}\sum_{i=1}^{N} K\left(\frac{\vec{x}-\vec{x_i}}{\sigma}\right) \tag{1}$$

where $\vec{x} = \{\vec{x_1}, \vec{x_2}, \dots, \vec{x_i}, \dots, \vec{x_N}\}$ is a dataset with N points in d-dimension, $\vec{x_i}$ is the i^{th} data vector $[x_{i1}, x_{i2}, \dots, x_{id}]'$, K represents the kernel function, σ is the kernel bandwidth. The Gaussian kernel is commonly used because of its smooth density estimation and only σ needs to be optimised. So

$$K\left(\frac{\vec{x}-\vec{x_i}}{\sigma}\right) = \frac{1}{(2\pi)^{d/2}}\exp\left(-\frac{\|\vec{x}-\vec{x_i}\|^2}{2\sigma^2}\right) \tag{2}$$

The Laplacian operation, which is an isotropic operation of the second spatial derivative, is then employed to obtain the ZLS of the implicit LSF (φ) as the initial boundary. Hence for the Gaussian kernel,

$$\varphi = \sum_{i=1}^{N} \frac{\|\vec{x}-\vec{x_i}\|^2 - \sigma^2}{N\sigma^{4+d}(2\pi)^{d/2}} \exp\left(-\frac{\|\vec{x}-\vec{x_i}\|^2}{2\sigma^2}\right) \tag{3}$$

Fig. 1 illustrates a LSF constructed using (3) on a 2-D Banana shaped dataset. The blue dots mark the given data points. The meshed surface is the LSF built on a grid in the given space occupied by the given data. The black boundary

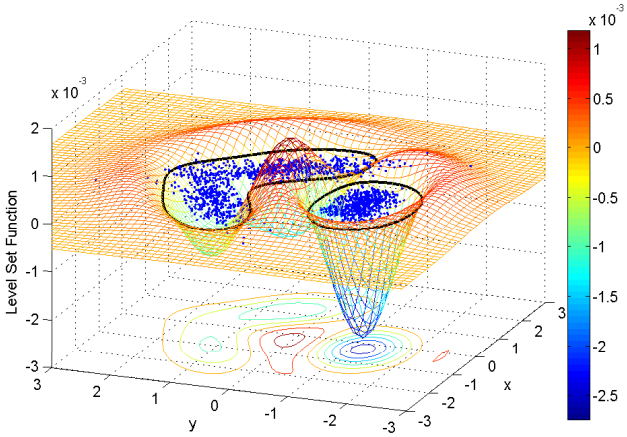

Fig. 1. The LSF on a 2-D Banana shaped dataset.

(two closed curves) is composed of the ZLS of φ. From the projected contours in the x-y plane and the right colour bar, φ satisfies: $\varphi = 0$ defines the boundary; $\varphi < 0$ at the points lying inside the boundary (i.e. interior points); and $\varphi > 0$ at the points outside the boundary (i.e. exterior points). Therefore, for any given data point, its location relative to the boundary can be identified according to the sign of the value of φ at that point. This identification method is employed in this paper. It differs from the approach in [2] where the boundary points and the corresponding normal vectors were required.

B. Boundary Evolution

The boundary evolution is represented with the following level set equation (LSE) [25]

$$\frac{\partial \varphi}{\partial t} + a(\vec{x}, t)|\nabla \varphi(\vec{x}, t)| = 0 \tag{4}$$

where $\partial\varphi/\partial t$ denotes the partial derivative of φ with respect to the time t. $|\nabla\varphi(\vec{x}, t)|$ is the magnitude of the gradient ∇. The evolution speed term a is a function of the space $\vec{x}$ and the time t, and it can be of either sign. The surface normal vectors point outwards, hence, $a > 0$ indicates the boundary expands in the outward normal direction; $a < 0$ indicates the boundary shrinks in the opposite direction; $a = 0$ makes (4) be a trivial $\partial\varphi/\partial t = 0$, i.e. φ is constant for all time. The bigger the absolute value of a the faster the boundary evolution and vice versa.

The numerical solution of (4) can be obtained by a Hamilton-Jacobi equation. It is computed on a uniform Cartesian grid. Considering a uniform grid ($i_1\Delta$, $i_2\Delta$, …, $i_d\Delta$) for d dimensional problems, where i_1, i_2, …,i_d are integers, Δ is the spatial subinterval (spatial step size). The time variable t is discretized to $k\Delta t$, where k is an integer and Δt is the time interval (the temporal step size). Denoting (i_1, i_2, …,i_d) by i, the data sample of $\varphi(\vec{x}, t)$ at the grid point ($i_1\Delta$, $i_2\Delta$, …, $i_d\Delta$) at time $k\Delta t$ is denoted by φ_i^k. Let n_j=(0, …, 0, 1, 0, …0) is the d elements vector whose j^{th} element is 1, while the other elements are 0. Consequently, the forward difference operator in the j^{th} spatial dimension, D_j^+, is defined by

$$D_j^+ \varphi_i^k = \frac{\varphi_{i+n_j}^k - \varphi_i^k}{\Delta} \tag{5}$$

and the backward difference operator in the j^{th} spatial dimension, D_j^-, is defined by

$$D_j^- \varphi_i^k = \frac{\varphi_i^k - \varphi_{i-n_j}^k}{\Delta} \tag{6}$$

For LSE (4), a first order accurate upwind scheme is given by

$$\frac{\varphi_i^{k+1} - \varphi_i^k}{\Delta t} = \max\left(a_i^k, 0\right)\nabla^+ + \min\left(a_i^k, 0\right)\nabla^- \tag{7}$$

where

$$\begin{aligned} \nabla^+ &= [\max(D_1^-, 0) + \min(D_1^+, 0) + \cdots + \max(D_d^-, 0) \\ &\quad + \min(D_d^+, 0)]^{1/2} \\ \nabla^- &= [\max(D_1^+, 0) + \min(D_1^-, 0) + \cdots + \max(D_d^+, 0) \\ &\quad + \min(D_d^-, 0)]^{1/2} \end{aligned} \tag{8}$$

More detailed discussion of the Hamilton-Jacobi equation and numerical discretization for motion in the normal direction are provided by Osher and Fedkiw [25]. According to the above analyses on LSE, the boundary evolution direction and speed depend on the evolution term a. Ding et al. [2] set a to a scalar at each evolution iteration, i.e. the boundary is globally expanded or shrunk at a speed a no matter which boundary segment it is. By contrast, we propose a locally adaptive boundary evolution scheme where the evolution direction and speed vary from different boundary segments at each evolution iteration.

C. Exterior Points-Based Local Evolution Speed Adaptation

This section graphically illustrates the proposed exterior points-based locally adaptive boundary description (EP-LABD) algorithm.

The 'banana' dataset including 1462 data points, shown in Fig. 1, is used again to illustrate the EP-LABD with a 2-D plot as shown in Fig. 2. The exterior points can be singled out as the sign of φ at those points is positive, and they are marked with small red circles. The resulting 45 exterior points means the current false negative rate (FNr, the fraction of normal data wrongly considered abnormal) is 0.0308. Suppose the expected FNr (denoted by λ) is set to 0.01, i.e. 1% of the normal data is tolerated to be left outside the boundary, therefore, λ_1=0.0308 and $\lambda_1 > \lambda$ indicate that the boundary needs to be expanded in order to enclose more data points.

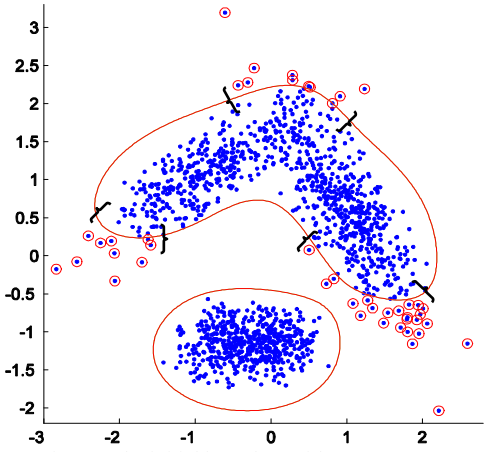

Fig. 2. The initial boundary with FNr=0.0308.

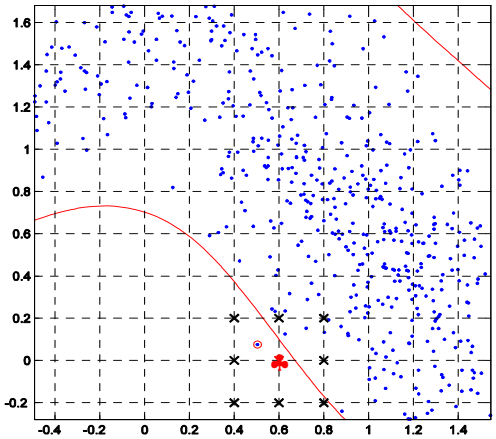

Fig. 3. An illustration for the EP-LABD algorithm.

Observing Fig. 2, it is not appropriate to expand the whole boundary. The boundary needs to be locally evolved, i.e. the segments, delimited by the three pairs of curly braces, that have nearby exterior points need to be expanded, while the remaining segments may be shrunk. As the LSF is maintained on a grid, the local boundary evolution can then be indirectly realised via local evolving the grid areas that cover those nearby exterior points. A zoomed part of Fig. 2 is shown in Fig. 3 where the grid interval is set to 0.2 for clarity but maybe much smaller in practice. The curve is a section of the boundary. The circled point is an exterior point, and its nearest grid point is marked with a red '♣' which is covered by an area defined by eight black '×'. Without loss of generality, a k-range nearest neighbours description is defined to cover an exterior data point using (2×k+1)2 grid points, e.g. k = 1 in Fig. 3.

Thus, the signs of evolution speed term a at those k-range nearest grid points are set to be positive (which means an expanding evolution), while the signs of a at the remaining grid points are negative (which means a shrinking evolution). Consequently, the expansion at those nine grid points will result in the expansion of the nearest segment of the boundary towards that circled exterior point. While the remaining segments of the boundary are also shrunk because of the shrinking evolution at the remaining grid points. A small *offset* can attach on the initial a setting for a quick evolution, e.g. when different exterior points have the same nearby gird point, a positive *offset* can make the corresponding boundary segment quickly expand, approaching to those exterior points. Also, a negative *offset* causes segment shrinking quickly. Therefore, the EP-LABD can locally evolve a boundary segment via identifying its nearby exterior data points without knowing all the boundary points.

D.Evolution Termination Condition

The boundary evolution termination is determined by λ, e.g., for a given application if 99% of the normal data points are expected to be classified as normal, then λ is set to 0.01. If the FNr on training data in current step is λ_i, and $\lambda_i < \lambda$, then the boundary will be shrunk in the next step. Otherwise, if $\lambda_i > \lambda$, the boundary will be expanded in the next step. Hence, the ideal termination condition will be $\lambda_i = \lambda$. However, it may never be met in practice due to the finite data size and computational error. For this reason, the evolution process can be terminated when λ_i is close to λ. Therefore, a practical termination condition will be $\lambda_i \in [\lambda-\varepsilon, \lambda+\varepsilon]$, where ε is a small positive scalar.

In practice, the absolute value of a is initialised to 0.25 and ε=0.001 respectively. Suppose the current boundary is shrinking with a=-0.25, and if $\lambda_i < \lambda-\varepsilon$ and $\lambda_{i+1} > \lambda+\varepsilon$, then a will be adjusted to 0.125 (i.e. $a = a/(-2)$) which means the boundary will expand at a slower speed a=0.125 in the next evolution step in order to make λ_{i+2} fall into the interval $[\lambda-\varepsilon, \lambda+\varepsilon]$. If the final adjusted a does not terminate the evolution, then ε will be slightly increased in order to enlarge the termination interval, e.g., $\varepsilon=\varepsilon\times1.1$. Similarly, assuming that the current boundary is expanding with a=0.25, and if $\lambda_i > \lambda+\varepsilon$ and $\lambda_{i+1} < \lambda-\varepsilon$, then the surface will shrink at a slower speed in the next step (still applying $a=a/(-2)$), and again the ε value may also change in order to enlarge the interval range if the termination condition is not met. Additionally, when $\lambda_{i+1} = \lambda_i$, the boundary will evolve with a quicker speed $a=a\times2$ in the next evolution step. Therefore the values of ε and a are dynamically changed based on the data distribution during the boundary evolution stage.

E. The Algorithm Description

The proposed 'Algorithm 1: EP-LABD' includes the training and testing processes. Three modules are used in the training process. Module 1 constructs a LSF according to Section III (A). Module 2 calculates the current FNr using the sign of LSF at each data point. The decision boundary is locally evolved in Module 3 where it includes an evolution speed term initialisation and a loop for evolution. Within the loop, the evolution direction and speed for different segments of the boundary are first defined by invoking an 'Algorithm 2: Speed-Adaptation' that consists of the following four steps: 1) Finding the exterior points (*otps*) according to the sign of the LSF at those points being positive; 2) Finding the nearest grid points to *otps*, i.e. *grid_otps*. Each point of *grid_otps* is the centre of an area covering the corresponding exterior point; 3) Identifying the areas centred by *grid_otps*. It finds the k-range nearest neighbour grid points to each point of *grid_otps*, i.e. *k_grid_otps*; and 4) Local adjustment for the direction and the speed of a. For those grid points of *k_grid_otps*, the corresponding sign of a is set to be positive, and the speed might be attached with a positive increment (*offset*) in order to quickly expand such a segment that has different nearby grid points (i.e. the boundary segments evolve quicker at a dense area than at a sparse area of the nearby exterior points). For the remaining grid points except *k_grid_otps*, the sign of a is set to be negative, and the evolution speed might also be attached with a negative increment (-*offset*) in order to make the remaining segments of the boundary quickly shrink towards those interior points.

As a is locally adaptive depending on the specific grid points that further depend on the exterior data points, this local boundary adaptation is fully data-driven, which is a distinct feature of the proposed approach. After the adjustment of a, the decision boundary is then evolved using the adjusted a by applying the LSE (4). As the evolution direction and speed may be different from different grid points, the proposed EP-LABD applies both (either) expanding and (or) shrinking locally in each evolution iteration according to the nearest exterior points to the evolved segments. Thirdly, due to the local evolution, the evolved boundary may be very rough. Therefore, the boundary needs to be smoothed to avoid noise-sensitivity. The convolution operation $G_\sigma * \varphi$ is used as in [26], where G_σ is a Gaussian kernel with a standard deviation σ. Finally, Module 2 is applied again to the training data using the smoothed φ in order to calculate the current FNr and check if the evolution termination is satisfied. The inherent computation of the testing process uses the sign of the LSF at the test data to calculate the FNr and FPr (the fraction of abnormal data incorrectly considered as normal) for performance evaluation.

Algorithm 1: EP-LABD

Training Process

Input: given normal data *training*
N_{tr} – number of training points
Module 1: LSF construction
Initialise a grid g
for each $\vec{x_i} \in$ *training*
φ_0 = apply Equation (3)
end for
$\varphi = \varphi_0$
Module 2: Current FNr (λ_i) computation
$s(1) = 0$
for each $\vec{x_i} \in$ *training*
if $\varphi(\vec{x_i}) > 0$ **then** $s(1) = s(1) + 1$
end for
$\lambda_i := s(1)/N_{tr}$
Module 3: Locally adaptive boundary evolution
Initialise *aValue*
while $(\lambda_i \notin [\lambda - \varepsilon, \lambda + \varepsilon])$
a = Speed-Adaptation (φ, *training*, *aValue*, g)
φ = evolve current φ applying LSE (4) with a
$\varphi = G_\sigma * \varphi$
λ_i = apply Module 2 to *training* using φ
end while
Output: φ

Testing Process

Input: φ, *test* data including *normal* and *abnormal* samples
N_{normal} – number of normal points in test data
$N_{abnormal}$ – number of abnormal points in test data
$s(1) = 0$ and $s(2) = 0$
for each $\vec{x_i} \in$ *test.normal*
if $\varphi(\vec{x_i}) > 0$ **then** $s(1) = s(1) + 1$
end for
$FNr = s(1)/N_{normal}$
for each $\vec{x_i} \in$ *test.abnormal*
if $\varphi(\vec{x_i}) \leq 0$ **then** $s(2) = s(2) + 1$
end for
$FPr = s(2)/N_{abnormal}$
Output: *FNr*, *FPr*

Algorithm 2: Speed-Adaptation

Input: φ, *training*, *aValue*, g
a = *aValue*
Step 1: Find the exterior points, *otps*
$otps = \{\vec{x_i} | \varphi(\vec{x_i}) > 0\}$
Step 2: Find the nearest grid points to *otps*
for each $\vec{x_i} \in$ *otps*
$grid\_otps = \{\vec{y_i} \in g | \min(\|\vec{x_i} - \vec{y_i}\|)\}$
end for
Step 3: Find k-range nearest grid points to *grid_otps*
for each $\vec{y_i} \in grid\_otps$
$k\_grid\_otps = \{\vec{y_i} \mp j \times unit\_grid, j = 0,1, \ldots, k\}$
end for
Step 4: Locally adjust evolution direction and speed
for each $\vec{y_i} \in g$
if $\vec{y_i} \in k\_grid\_otps$ **then** $a = a(\vec{y_i}) + offset$
else $a = -a(\vec{y_i}) - offset$
end for
Output: a

Applying EP-LABD to the 'banana' dataset, one randomly selected intermediate status (λ_j=0.0226) and the final status (λ_i=0.0109, as λ=0.01) of the evolved boundary are shown in Fig. 4 (a) and (b), where the dotted and solid boundaries illustrate the initial and the evolved boundaries, respectively. Clearly, relative to the initial boundary, some segments of the evolved boundary were expanded and some were shrunk. Using the final evolved boundary, the results on test data (1462 unseen normal and 2376 abnormal points) are also shown in Fig. 5, with (a) FNr=0.0150 because of 22 exterior points, and (b) FPr=0.3864, i.e. 61.36% of the abnormal data (marked with '+') successfully detected.

IV. Experiment Setup

Preliminary experiments have been carried out to evaluate the proposed EP-LABD approach comparing with the LSBD proposed in [2] and four representative novelty detection methods including OCSVM, k-NN, k-means, and GM. One synthetic dataset (2-D 'banana') as in [2] and two real datasets (3-D 'vowel' and 10-D 'abalone') are derived from the UCI machine learning repository. The datasets detailed in Table I are randomly partitioned into 10 folds for cross validation and parameter tuning such as the bandwidth within OCSVM, the number of nearest neighbours with k-NN, the number of clusters within k-means, the number of Gaussian components within GM, and the kernel width within LSBD and EP-LABD.

From the illustration in Fig. 4, locally adaptive boundary evolution is needed when there is an exacting error requirement on normal data. We hence set five lowering values of λ as 0.05, 0.04, 0.03, 0.02, and 0.01 to investigate the ROC curves of all the six detectors on the selected datasets. The experimental results are averaged over 10 folds for each parameter tuning. Therefore, the combination of these settings requires the training of about 18,000 novelty detectors (i.e. 6 novelty detection methods, 3 datasets, 10 partitions of each dataset, about 20 grid search values for each model parameter, and 5 training thresholds). The resulting ROC curves are shown in Fig. 6 from which the following observations can be made:

Results of the LSBD method in [2] are available only on 2-D 'banana' and 3-D 'vowel' datasets, and are not available on the higher dimensional 'abalone' dataset as an effective isosurface reconstruction for data with higher than three dimensions is currently not available. The EP-LABD approach clearly produced a higher ROC curve than the other five methods on each dataset. It indicates that, for a given TPr (1-FNr), the EP-LABD obtained a lower FPr than the LSBD and other four representative methods. Therefore, the proposed locally adaptive boundary evolution scheme obtained a better performance in terms of the ROC metric as compared to the global boundary evolution scheme and four representative novelty detection methods when a very high classification accuracy is required for normal events.

TABLE I
THE SELECTED DATASETS

Datasets			Each Fold	
Name	*Points*	*Dim*	*Training(normal)*	*Test (normal + novelty)*
banana	5300	2	1462	3838 (1462+2376)
vowel	871	3	246	625 (246+379)
abalone	4177	10	1266	2911 (141+2770)

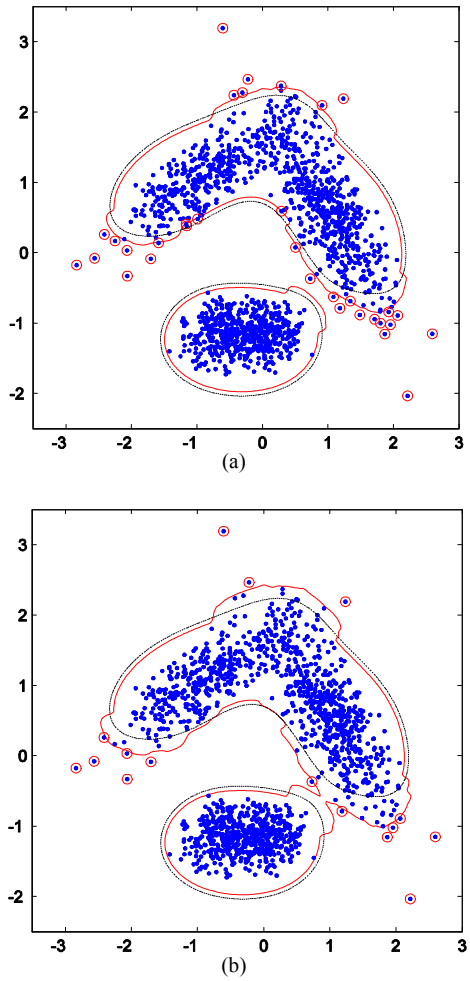

Fig. 4. Two statuses of the boundary evolved by EP-LABD where the FNr is 0.0226 in (a) and 0.0109 in (b).

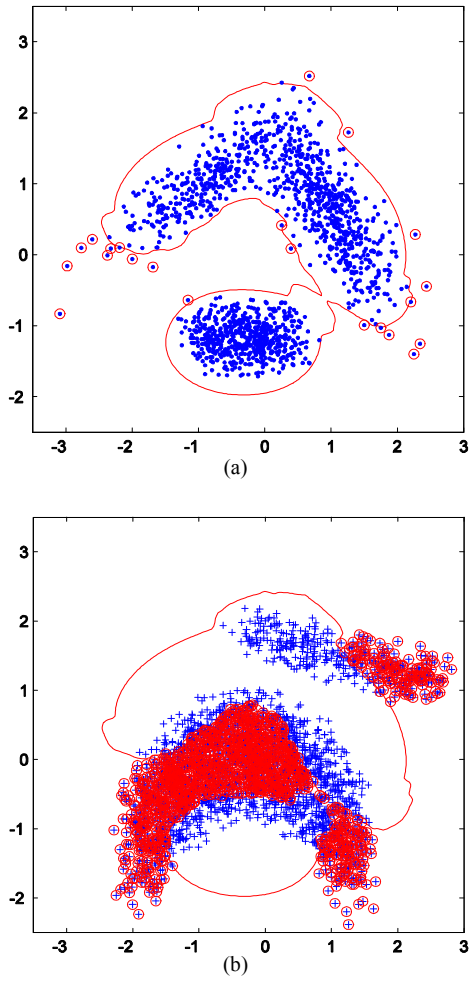

Fig. 5. The testing results where (a) shows FNr=0.0150 and (b) shows FPr=0.3864.

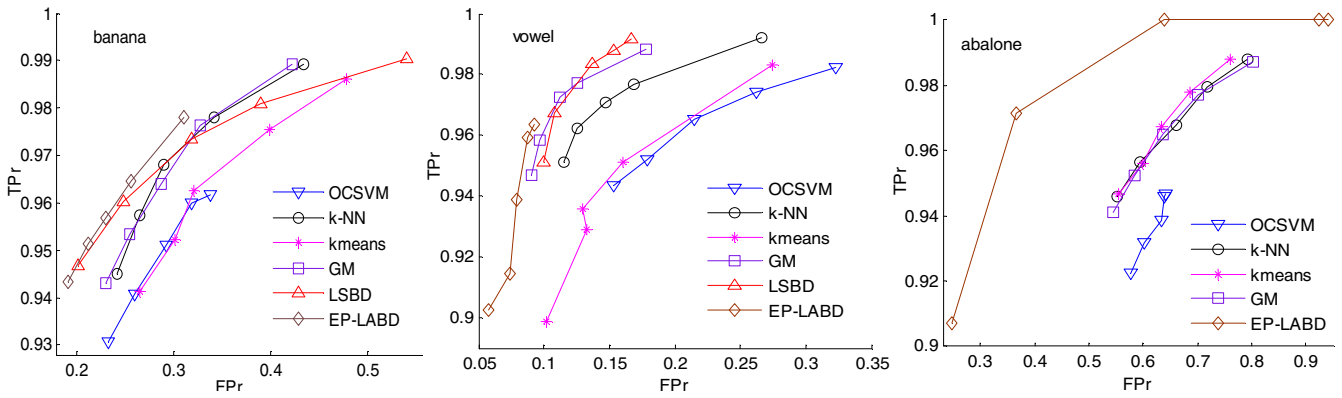

Fig. 6. ROC curves of six novelty detectors with the three datasets.

V. Conclusion

This paper proposed a locally adaptive boundary evolution algorithm for LSM-based novelty detection. The algorithm, EP-LABD, utilised the exterior points to indirectly locally evolve the nearby segments of the boundary. This results in an adaptive boundary that better fits the data distribution than the reported LSBD-based globally evolved boundary. The EP-LABD is practically applicable to any dimensional data, while the LSBD suffers from the requirement of isosurface reconstruction which is currently impracticable for data with higher than three dimensions. The experimental results demonstrated that the EP-LABD approach obtained higher novelty detection accuracy than the LSBD and four representative methods under an exacting FNr requirement. Future work will focus on the evaluation on more datasets.

REFERENCES

[1] P. Hayton, B. Schölkopf, L. Tarassenko, and P. Anuzis, "Support vector novelty detection applied to jet engine vibration spectra," *Advances in Neural Information Processing Systems*, vol. 13, pp. 946-952, 2001.

[2] Xuemei Ding, Yuhua Li, Ammar Belatreche, and Liam Maguire, "Constructing minimum volume surfaces using level set methods for novelty detection," in *International Joint Conference on Neural Networks*, Brisbane, Australia, 2012, pp. 3158-3163.

[3] M. Markou and S. Singh, "Novelty detection: a review--part 1&2: statistical approaches & neural network based approaches," *Signal Process*, vol. 83, no. 12, pp. 2481-2521, 2003.

[4] V. J. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85-126, 2004.

[5] A. Patcha and J. M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Computer Networks*, vol. 51, no. 12, pp. 3448-3470, 2007.

[6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1-58, 2009.

[7] P. Gogoi, D. K. Bhattacharyya, B. Borah, and J. K. Kalita, "A survey of outlier detection methods in network anomaly identification," *The Computer Journal*, vol. 54, no. 4, pp. 570-588, 2011.

[8] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection," *Advances in Neural Information Processing Systems*, vol. 12, no. 3, pp. 582-588, 2000.

[9] Y. Li, "Selecting training points for one-class support vector machines," *Pattern Recognition Letters*, vol. 32, no. 11, pp. 1517-1522, 2011.

[10] H. Lian, "On feature selection with principal component analysis for one-class SVM," *Pattern Recognition Letters*, vol. 33, no. 9, pp. 1027-1031, 2012.

[11] D. M. J. Tax and R. P. W. Duin, "Support vector domain description," *Pattern Recoginition Letters*, vol. 20, no. 11-13, pp. 1191-1199, 1999.

[12] S. M. Guo, L. C. Chan, and J. S. H. Tsai, "A boundary method for outlier detection based on support vector domain description," *Pattern Recognition*, vol. 42, no. 1, pp. 77-83, 2009.

[13] X. Peng and D Xu, "Efficient support vector data descriptions for novelty detection," *Neural Computing and Applications*, vol. 21, no. 8, pp. 2023-2032, 2012.

[14] D. A. Clifton et al., "Novelty detection for identifying deterioration in emergency department patients," *Leture Notes in Computer Science*, vol. 6936, pp. 220-227, 2011.

[15] M. Mariam, W. Delb, B. Schick, and D. J. Strauss, "Feasibility of an objective electrophysiological loudness scaling: a kernel-based novelty detection approach," *Artificial Intelligence in Medicine*, vol. 55, no. 3, pp. 185-195, July 2012.

[16] G. Pölzlbauer, T. Lidy, and A. Rauber, "Decision manifolds-a supervised learning algorithm based on self-oragnization," *IEEE Transactions on Neural Networks*, vol. 19, no. 9, pp. 1518-1530, 2008.

[17] G. G. Cabral, A. L. I. Oliveira, and C. B. G. Cahu, "Combining nearest neighbor data description and structural risk minimization for one-class classification," *Neural Computing & Applications*, vol. 18, no. 2, pp. 175-183, 2009.

[18] Z. Tang, J. Yang, and B. Yang, "A new outlier detection algorithm based on manifold learning," in *Chinese Control and Decidion Conference*, 2010, pp. 452-457.

[19] M. M. Masud, J. Gao, L. Khan, J. Han, and B. Thuraisingham, "Classification and novel class detection in concept-drifting data streams under time constraints," *IEEE Transaction on Knowledge and Data Engineering*, vol. 23, no. 6, pp. 859-874, 2011.

[20] M. Moshtaghi et al., "Clustering ellipses for anomaly detection," *Pattern Recognition*, vol. 44, no. 1, pp. 55-69, 2011.

[21] S. Ntalampiras, I. Potamitis, and N. Fakotakis, "Probabilistic novelty detection for acoustic suiveillance under real-world conditions," *IEEE Transaction on Multimedia*, vol. 13, no. 4, pp. 713-719, 2011.

[22] M. Filippone and G. Sanguinetti, "Information theoretic novelty detection," *Pattern Recognition*, vol. 43, no. 3, pp. 805-814, 2010.

[23] Yannick Benezeth, Pierre-Marc Jodoin, and Venkatesh Saligrama, "Abnormality detection using low-level co-occurring events," *Pattern Recognition Letters*, vol. 32, no. 3, pp. 423-431, 2011.

[24] Andy M. Yip, Chris Ding, and Tony F. Chan, "Dynamic cluster formation using level set methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 877-889, June 2006.

[25] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. New York: Springer, 2003.

[26] Chunming Li, Chenyang Xu, Changfeng Gui, and Martin D. Fox, "Distance regularized level set evolution and its application to image segmentation," *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3243-3254, December 2010.