

Machine Learning Techniques for the Analysis of Magnetic Flux Leakage Images in Pipeline Inspection

Ahmad Khodayari-Rostamabad¹, James P. Reilly¹, Natalia K. Nikolova¹, James R. Hare² and Sabir Pasha²

Abstract—The magnetic flux leakage (MFL) technique is commonly used for non-destructive testing of oil and gas pipelines. This testing involves the detection of defects and anomalies in the pipe wall, and the evaluation of the severity of these defects. The difficulty with the MFL method is the extent and complexity of the analysis of the MFL images. In this paper we show how modern machine learning techniques can be used to considerable advantage in this respect. We apply the methods of support vector regression, kernelization techniques, principal component analysis, partial least squares, and methods for reducing the dimensionality of the feature space. We demonstrate the adequacy of the performance of these methods using real MFL data collected from pipelines, with regard to the performance of both the detection of defects, and the accuracy in the estimation of the severity of the defects. We also show how low-dimensional latent variable structures can be effective for visualizing the clustering behaviour of the classifier.

Index Terms—magnetic flux leakage, pipeline inspection, non-destructive testing, regression, support vector machines, partial least squares, kernelization.

I. INTRODUCTION

THIS paper addresses the problem of automated non-destructive testing of installed oil and natural gas pipelines using inline *magnetic flux leakage* (MFL) inspection techniques. The MFL technique provides a high-resolution image of the interior of the pipe wall from which defects and other anomalies can be detected and duly reported.

Due to the adverse effect of environmental damage that could result from a pipeline leak or catastrophic failure, pipelines must be routinely evaluated for integrity. The logistics and cost of shutting down a pipeline for inspection is prohibitive, so inspection devices, referred to as *pipeline inspection gauges* (PIGs) in the trade, are designed for autonomous operation in the pipeline, and are propelled along the pipe by normal transport flow. PIGs are also referred to as *in-line-inspection* (ILI) tools.

The ILI tool magnetizes the pipe wall as it travels down the pipe. Hall effect or coil sensors measure the localized magnetic flux leakage intensity along the pipe wall. Defects in the pipe wall cause irregularities in the magnetic field, that are detected by an array of these sensors, placed at regular intervals around the circumference of the inside of the pipe wall. Thus, MFL testing is based on detecting the magnetic field that leaks from a pipe's wall in the vicinity of surface and subsurface flaws. See [1]–[4] for reviews of history, problems and concerns.

The stored magnetic image data is then analyzed off-line. Of special interest to pipeline operators is the extent and location of various defects that can adversely affect the integrity of the pipeline and its operation. Some of these defects include corrosion¹, deformations, fatigue, hairline cracks, dents, buckles, de-laminations, and faulty welds. The results are used to determine repair and replacement priorities for a pipeline operator. Typically, the image analysis portion of the inspection procedure is done exclusively by human operators referred to as *non-destructive testing* (NDT) technicians. As a result of the manual nature of the inspection process, it is inherently slow and error-prone. Thus there is a strong motivation in the industry to automate the inspection process using methods such as machine learning as we propose.

Once defects have been identified, an equally important problem is the assessment of the size or severity of the defect (sizing). Estimated defect depths are used to determine the safety of the pipe, and to calculate accurate *maximum allowable operating pressure* (MAOP) of the oil/gas flowing through the line. In this paper, we propose machine learning techniques for defect detection and sizing using real MFL images recovered from actual pipeline inspections.

There have been several previous works in defect detection and sizing using the MFL technique. Machine learning has been used previously in this context in [5] which evaluates the use of multilayer Perceptrons (MLP) for pattern recognition of MFL signals in weld joints of pipelines. Inverse modelling approaches are

¹ Electrical and Computer Engineering Department, McMaster University, 1280 Main St. West, Hamilton, ON, Canada, L8S 4K1

² Intratech Inline Inspection Services Ltd., 3995 Sladeview Cres., Unit 7, Mississauga, ON, Canada, L5L 5Y1

Manuscript received Oct. 14, 2008; revised Feb. 25, 2009. The corresponding author: J. P. Reilly (e-mail: reillyj@mcmaster.ca)

¹Corrosion turns steel into non-ferromagnetic iron oxide that locally reduces permeability.

used in [6] for defect-shape construction. In [7], an iterative inversion method is proposed, using a multi-resolution wavelet transform and a radial basis function neural network (RBFN), for the prediction of 3-D defect geometry from MFL measurements.

The inversion procedure in [8] employs the ‘space mapping’ (SM) methodology. Space mapping shifts the optimization burden from a computationally expensive accurate or fine model like a finite-element method (FEM) simulation, to a less accurate or coarse but fast model, like analytical formulas.

The works of [9], [10] present a modified wavelet transform domain adaptive finite impulse response (FIR) filtering algorithm for removing seamless pipe noise (SPN) in the MFL data. Papers [11], [12] give wavelet based approaches to this problem for both de-noising and classification. In [13], an adaptive method for channel equalization (to compensate the mismatch between sensors) in MFL inspection is presented using the finite impulse response filter. Reference [14] presents a model based probability of detection (POD) evaluation technique for MFL inspection of natural gas transmission pipelines.

Comparison of the MFL method to ultrasonic methods is given in [18]. See also [19]–[21] for some other related topics.

The work of [15] presented a fast direct method that provides estimation of the crack parameters (orientation, length, and depth) in rectangular surface-breaking cracks, based on measurements of one tangential component of the magnetic field. The difficulty with this and other defect depth estimation methods is that the technique is limited to regular rectangular cracks. Real defects occurring naturally in pipelines virtually never exhibit this simple form of geometry.

A traditional approach for crack depth estimation is to use ‘calibration curves or surfaces’. For example, [15] constructed a calibration surface that shows MFL signal strength versus two parameters of crack—depth and crack-length. In [16], three two-dimensional curves are drawn to show MFL signal sensitivity, in which each curve shows the effect of a specific test parameter on the magnitude of the MFL signal. In a different approach, [17] presented an analytic method for depth estimation in rectangular slots. In general, calibration curves/surfaces are 2D (or 3D) plots showing the effect of one measured parameter (or at most 2 parameters/features) on the magnitude of the measured MFL signal, or on the defect depth. However, the problem in real life is that cracks (and metal-losses) exhibit irregular and complex geometries and therefore cannot be characterized with just a few simple parameters. Therefore methods which assume a specific geometry or require representation in terms of a few parameters will not perform well with real defects.

In this paper, we extend the above works to develop high performance machine learning methods for both

defect detection and sizing of MFL images. We discuss the use of the support vector machine, kernelized principal component analysis (PCA) and kernelized partial least squares (PLS) methods for this purpose, along with methods for feature selection and dimensionality reduction. We show that these methods, if properly executed, can give very satisfactory performance for both defect detection and depth estimation for real defects with irregular (or arbitrary) geometric configurations.

The organization of this paper is as follows. In Sect. II, we review the magnetic flux leakage imaging technique. In Sect. III, we overview the basic principles of machine learning, and in Sect. IV, we summarize the theory of the machine learning algorithms used in this work. In Sect. VI, we give experimental results for real MFL data, and Sect. VII gives the conclusions.

II. THE MAGNETIC FLUX LEAKAGE IMAGING METHOD

A simplified generic ILI tool is depicted in Fig. 1. Permanent magnets magnetize the pipe to saturation or near saturation flux density, typically in the axial direction. Flux excitation in the circumferential direction is also possible. As shown, the magnetic leakage fields from the pipe wall are detected using uniformly-spaced Hall² or coil sensors. The sensor is placed at the center of the poles in the magnetic circuit to ensure that it is located in the most uniform part of the field. Systems with standard resolution measure the leakage flux in one direction of excitation, whereas high-resolution MFL systems employ a large number of such sensors that can record anomalies and changes in the magnetic field in one or two directions with sufficient density to recognize even very small pipeline metal defects. The signals from the sensors are sampled as the ILI moves through the pipe. These samples are then recorded on-board the ILI. Once the ILI is retrieved at the end of the inspection run, the stored signals are then processed and displayed section-by-section on a computer monitor, to render a magnetically-derived 2D or 3D image of the pipe wall.

The magnetic flux from powerful magnetizers is coupled to the pipe wall through steel brushes (or steel protectors). The magnetic sensors are placed in close proximity with the inside surface of the pipe wall for optimum sensitivity to flux variations. The ILI tool body, magnets, steel couplers, and the pipe wall then create a magnetic circuit. A volumetric wall loss in the pipeline wall acts as a region of high magnetic reluctance causing an increased magnetic leakage field. For example, if the wall’s thickness is reduced by the presence of a defect (as depicted in the figure), a higher fraction of the magnetic flux leaks from the wall into the space inside

²An example of Hall effect sensor which was used in our experiments is built as a solid-state sensor using quad Hall sensing elements. The detection area of the sensor is less than 2mm by 3mm. The output voltage varies in linear proportion to the strength of magnetic field.

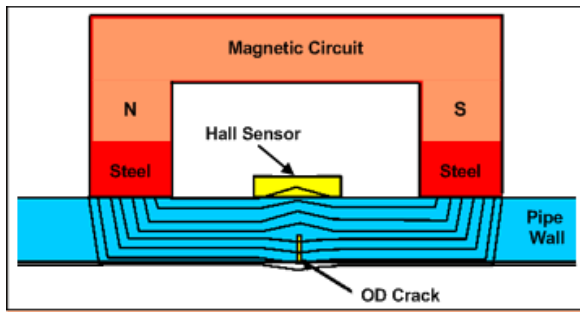


Fig. 1. Simplified schematic of the magnetic circuit of a typical ILI tool, showing the scattering of the magnetic field due to an external crack. OD stands for ‘outer diameter’.

and outside the pipe, allowing the defect to be detected by the presence of a corresponding increase in leakage magnetic flux density.

In general, the information that is stored on-board the ILI tool includes measurements of distance, tool speed, temperature, pressure, along with the 2D or 3D samples of the magnetic field through the Hall effect sensors. Other information such as wall-thickness, diameter, properties of the fluid or gas being transported by the pipe, geometry of the pipe, material of the pipe, properties of magnetizers and magnetic sensor arrays are also available to the NDT technician.

The functionality of the ILI tools is divided up into separate modules that are inter-connected by articulating joints. This configuration allows the tool to negotiate elbows and sharp turns that may be encountered in the pipeline. Fig. 2 shows two examples of these devices, with each module being identified as 1) the tow section, 2) the main sensor array, and 3) the electronics body. Other modules (not shown) include battery vessels, odometer modules, sensor arrays for interior/exterior discrimination, and calipers to detect physical damage or protruding artifacts in the pipeline.

ILI tools come in different sizes to fit various pipeline diameters. Outer diameters of pipes used in our experiments are 8.625, 10.75 and 12.75 inches (referred to as 8, 10, and 12 in. pipes in the paper), with corresponding wall thicknesses equal to 4.77mm, 5.16mm, and 5.6mm, respectively.

A sample of a real MFL image of a metal loss defect in an 8 in. pipe is shown in Fig. 3, and its smoothed and de-noised version is shown in Fig. 4. The deep blue regions represent a girth weld in the pipe. Fig. 5 shows the corresponding contour plot. The de-noising and smoothing method used in our experiments is discussed in Sect. VI. The depth of the metal-loss as determined by independent measurement is approximately 60% of wall-thickness, where the wall-thickness is 4.77mm. A single recognizable “bump” corresponding to the defect is apparent in the image. The physical shape of the outer surface of this metal-loss defect, as recorded off-line by an NDT technician is shown in Fig. 6.

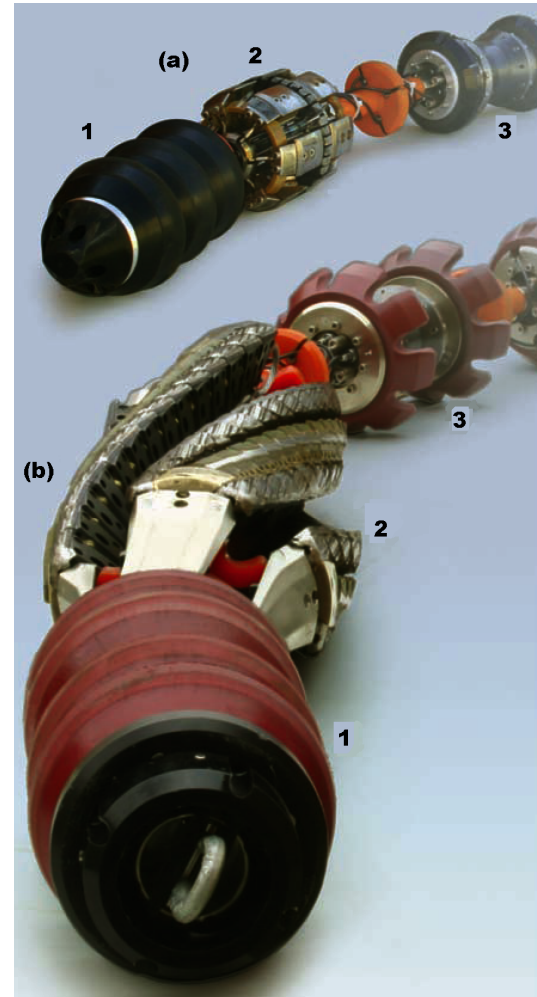


Fig. 2. Images of frontal parts of two ‘in-line inspection tools’ or PIGs by Intratech. Printed with permission from Intratech Inline Inspection Services Ltd., Mississauga, ON, Canada.

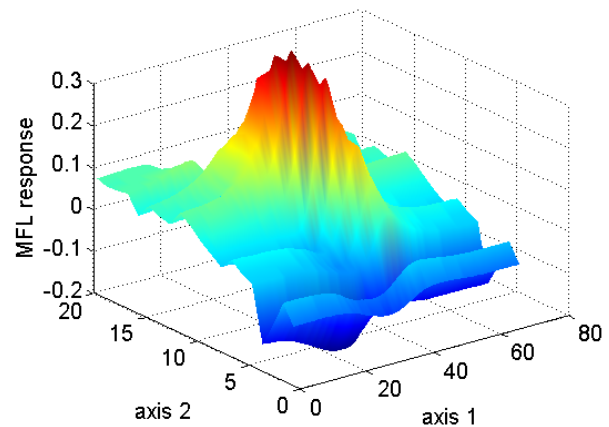


Fig. 3. A sample real MFL image data for a metal-loss defect in a pipe wall. Printed with permission from Intratech Inline Inspection Services Ltd., Mississauga, ON, Canada. The Z-axis has been scaled.

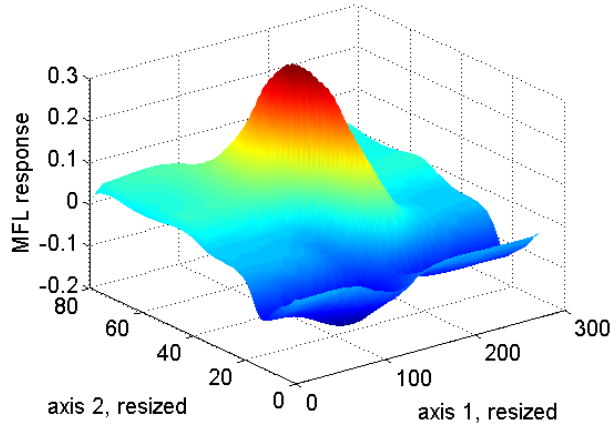


Fig. 4. De-noised and smoothed version of the sample real MFL image data shown in Fig. 3. The Z-axis has been scaled.

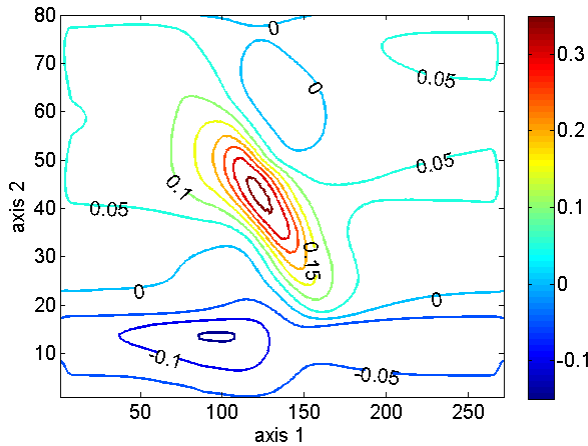


Fig. 5. Contour plot of de-noised MFL image data of Fig. 4.

As may be seen, the MFL image shown in Fig. 3 is not well represented by a simple geometric configuration, such as a rectangle. Other real defects may be even more complex. For example, this may happen when the metal-defect is close to or inside a ‘long-seam weld’ or a ‘girth weld’. This justifies our assertion that methods which fit simple geometric models, or characterize defects using a simple low-dimensional parameter estimation methods will not perform well with real measurements. Therefore the benefit of using machine learning in MFL data analysis becomes apparent when we work with real MFL



Fig. 6. Physical shape of the outer surface of the metal-loss defect corresponding Fig. 3-5. The image is rotated and scaled relative to the images of Figs. 3-5.

data. In such cases, the machine learning method will automatically find an estimate of the detection/estimation model. However, a drawback of machine learning methods is that they require an adequate and often large quantity of training samples to perform properly. The generation of training data is discussed in Sect. VI.

III. A SUMMARY OF THE MACHINE LEARNING PROCESS

There are at least two inter-related aspects of the machine learning process that are relevant to the topic at hand. The first is the pattern classification problem where we map an input object into a particular element of a set of classes. In this paper, we concentrate on the binary detection problem of classifying anomalous image segments into one of two classes: the first class is the one which consists of *injurious* or *non-benign defects* such as various crack-like anomalies and metal-losses in girth welds, long-seam welds, or in the pipe wall itself, which if left untreated, could lead to pipeline rupture. The second class consists of *non-injurious* or *benign* objects such as noise events, safe and non-harmful pipeline deformations, manufacturing irregularities, etc. In effect, the classification process “learns” a mathematical model which describes the observations.

The second aspect is the selection and preparation of *features*. Features are quantities extracted from a measured data sample which is to be classified. The set of features \mathbf{x} associated with the sample defines a point in a Euclidean space. Features are selected so that objects of the same class cluster into a specific region in the feature space. The classifier then determines which region of the space a particular test point belongs to and assigns an appropriate class to the measurement.

The classification process may be viewed as a mapping $f(\mathbf{x}_i) : \mathbf{x}_i \in \mathbb{R}^{N_r} \mapsto \mathbb{N}$ from a point \mathbf{x}_i in an input feature space into a discrete output space which corresponds to the respective classes of the input.

The data model is assumed to be of the form

$$y_i = f(\mathbf{x}_i; \Theta) + n_i, \quad i = 1, \dots, M_t \quad (1)$$

where the n_i are additive random variables, modeling measurement noise, etc., whose probability distribution is unknown. The discrete-valued variable y_i corresponds to the class of the i th data sample, and Θ represents the parameters of the model function f which are to be determined. The function f accepts the input feature vector \mathbf{x} from an object and outputs a discrete value corresponding to the class of the object.

There are many methods of determining the function f . These include the support vector machine, Bayesian classifiers, regularized least squares, artificial neural network models and many more, e.g., [22]–[24]. All these methods are based on the general principle of minimizing the expected discrepancy between the target y and $f(\mathbf{x}; \Theta)$ for a given input \mathbf{x} , by some loss

functional $\mathcal{L}(y, f(\mathbf{x}; \Theta))$, in addition to minimizing the model/function complexity.

The function f is determined with one of the above methods using a *training process*. In the present context, we are given M_t segments of an MFL image (e.g. corresponding to defects, welds, etc.). The N_r reduced features are extracted from each image segment and assembled into a vector $\mathbf{x}_i \in \mathbb{R}^{N_r}$. The associated class, image label or target value (assumed known) corresponding to the observed image segment is denoted by the variable $y_i \in \mathbb{N}$. The set $\mathcal{D} = \{\mathbf{x}_i, y_i, \quad i = 1, \dots, M_t\}$ is referred to as the *training set*. For classification, f establishes decision boundaries in the feature space which effectively separate patterns or clusters corresponding to the different classes. As described later, the training data used for the experiments in this paper are generated from real inspection runs and have been prepared by NDT technicians.

An alternative approach to modeling the underlying observations is the regression process where the function $f(\cdot)$ is continuous; i.e., $f(\cdot) \in \mathbb{R}$. In this case, f gives an estimate of parameter(s) of interest, and is determined to give as close a fit as possible between the values y_i and the corresponding function value $f(\mathbf{x}_i)$, $i = 1, \dots, M_t$. In this paper, regression may be used, e.g. to determine the size of a defect, particularly depth, from an MFL image segment.

Even though classification and regression are often seen as two separate entities in the literature, in fact regression can be readily modified to perform classification tasks. This is accomplished in the regression case simply by fitting f to a discrete-valued function, rather than a continuously-valued one as suggested previously. Because the algorithms used for regression typically gave better performance than classification algorithms in our application, most of the classification results in this paper are obtained using regression methods.

The first step in a machine learning problem is to transform the observed data into a feature space. The extraction and selection of features is a critical issue in any machine learning application. Candidate features include an assortment of standard image processing features like statistical attributes, coefficients of various transforms e.g., Fourier, wavelet, discrete cosine transform, etc. The number N_c of candidate features can become very large. For optimum performance of the classification/regression algorithm, it is necessary to prune the candidate set containing N_c features into a subset containing only N_r features, where usually $N_r \ll N_c$. This is referred to as the *feature reduction* or *dimensionality reduction* problem and is discussed in Sect. V.

If N_r is relatively large compared to M_t , then numerical stability and over-training issues can arise, with the result that the estimated function f has large variance. If N_r is too small relative to N_c , then the selected features may not adequately describe the clustering behaviour of

the observations. Both of these scenarios lead to poor performance.

IV. TECHNIQUES FOR CLASSIFICATION AND REGRESSION

In this section, we give a summary of various classification and regression methods that were found to give good performance in the pipeline inspection problem under consideration. We first give a brief discussion on kernelization techniques, which is a convenient method for introducing nonlinearity into the formulation of the function f . We then discuss various relevant classification and regression schemes.

A. Kernelization

Many techniques (such as the basic support vector machine below) result in linear decision boundaries and require only inner-product operations on the input data. It is possible to produce nonlinear decision boundaries in the feature space by transforming a feature vector $\mathbf{x} \in \mathbb{R}^{N_r}$ into another space \mathcal{U} by defining a new vector $\mathbf{U} = \phi(\mathbf{x})$ for some nonlinear function $\phi(\cdot)$. However, in the case at hand, a more efficient method of introducing nonlinearity is to compute inner products of the form $\phi(\mathbf{u})^T \phi(\mathbf{v})$, where \mathbf{u} and \mathbf{v} are input feature vectors in \mathbb{R}^{N_r} using kernel representations [22], [25], [26], as follows

$$\phi(\mathbf{u})^T \phi(\mathbf{v}) = \mathcal{K}(\mathbf{u}, \mathbf{v}). \quad (2)$$

This procedure allows us to compute the value of the inner product in \mathcal{U} without explicitly carrying out the transformation $\phi(\cdot)$. Examples of commonly used kernels are the d th-order polynomial, the Gaussian and the sigmoid, given respectively as

$$\mathcal{K}_1(\mathbf{u}, \mathbf{v}) = (\gamma \mathbf{u}^T \mathbf{v} + 1)^d = \left(\gamma \sum_{i=1}^{N_r} u_i v_i + 1 \right)^d \quad (3)$$

$$\mathcal{K}_2(\mathbf{u}, \mathbf{v}) = \exp \left(\frac{-\|\mathbf{u} - \mathbf{v}\|^2}{2\sigma^2} \right) \quad (4)$$

$$\mathcal{K}_3(\mathbf{u}, \mathbf{v}) = \tanh(\kappa(\mathbf{u}^T \mathbf{v} + \theta)) \quad (5)$$

where $\gamma, d, \sigma, \kappa$ and θ are design parameters.

B. Regularized Least-Squares Regression

In the specific case of least-squares regression, the loss functional assumes the form $\mathcal{L}(y_i, f(\mathbf{x}_i; \Theta)) = \|y_i - f(\mathbf{x}_i; \Theta)\|_2^2$. In any machine learning problem, there is always a tradeoff between model complexity on the one hand (which can lead to over-fitting) and model accuracy, which is essential for prediction, on the other. Regularization theory is a means of regulating this tradeoff. The goal is to minimize the following regularized risk functional

$$\mathcal{R}(f) = \frac{1}{M_t} \sum_{i=1}^{M_t} \mathcal{L}(y_i, f(\mathbf{x}_i; \Theta)) + \beta \|f\|_{\mathcal{K}}^2 \quad (6)$$

where the second term, $\|f\|_{\mathcal{K}}^2$, is a stabilizer to minimize the model complexity of the function f , and the positive constant $\beta > 0$ is called the regularization parameter, which represents the relative importance of the model complexity with respect to the performance measure, (or model accuracy). The function $\|f\|_{\mathcal{K}}^2$ is a norm in a ‘reproducing kernel Hilbert space’ (RKHS) \mathcal{H} defined by the positive definite function \mathcal{K} [27]. Instead of $\|f\|_{\mathcal{K}}^2$, other stabilizer functions like $\|D(f)\|^2$ can be used in (6), where D is a linear differential operator [23]. This makes the function smooth, thereby satisfying the property of continuity.

Under some general conditions, the function $f(\cdot)$ in (6) may be constructed using a kernelization process as follows:

$$f(\mathbf{x}; \Theta) = \sum_{i=1}^{M_t} v_i \mathcal{K}(\mathbf{x}_i, \mathbf{x}) \quad (7)$$

where the coefficients v_i are to be determined. In the regularized least-squares (RLS) regression method, we use the Gaussian kernel, defined as follows

$$\mathcal{K}(\mathbf{x}_i, \mathbf{x}) = \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x} - \mathbf{x}_i\|^2\right). \quad (8)$$

The solution \mathbf{v}_* minimizing (6) where $f(\cdot)$ assumes the form of (7) and $\|f\|_{\mathcal{K}}^2 = \mathbf{v}^T \mathbf{K} \mathbf{v}$, satisfies

$$(\mathbf{K} + \beta \mathbf{I}) \mathbf{v} = \mathbf{y} \quad (9)$$

where \mathbf{I} denotes the $M_t \times M_t$ identity matrix, $\mathbf{y} = [y_1, y_2, \dots, y_{M_t}]^T$, $\mathbf{v} = [v_1, v_2, \dots, v_{M_t}]^T$, and

$$\mathbf{K} = \begin{bmatrix} \mathcal{K}(\mathbf{x}_1, \mathbf{x}_1) & \dots & \mathcal{K}(\mathbf{x}_1, \mathbf{x}_{M_t}) \\ \vdots & \ddots & \vdots \\ \mathcal{K}(\mathbf{x}_{M_t}, \mathbf{x}_1) & \dots & \mathcal{K}(\mathbf{x}_{M_t}, \mathbf{x}_{M_t}) \end{bmatrix}. \quad (10)$$

We now summarize how to estimate the depth (or label) of a metal defect from a given segment of pipeline MFL image data using the regularized least squares procedure. First, the feature vector $\mathbf{x} \in \mathbb{R}^{N_r}$ is extracted from the image segment under test. The matrix \mathbf{K} is calculated from all the feature vectors in the training data, and the coefficient vector \mathbf{v} is then obtained from (9) for some specified values of β and σ . The desired estimate of the metal defect depth (or label), yielded by f , is obtained by substituting \mathbf{x} and the resulting \mathbf{v} into (7).

C. Support Vector Machine for Regression

The original ‘support vector machine’ (SVM) was proposed by Vapnik [22] for use as a classifier, in which the classification function is found by solving a regularization problem: minimizing classification errors in addition to maximizing the separation margin in feature space between mapped data sets of two classes. It was also extended to the regression case, e.g., [26]. The support vector machine regression (SVR) method uses

the so-called ϵ -insensitive loss function (ϵ -ILF), defined as follows

$$\mathcal{L}_\epsilon(y, f(\mathbf{x}; \Theta)) = \begin{cases} 0, & \text{if } |y - f(\mathbf{x}; \Theta)| \leq \epsilon \\ |y - f(\mathbf{x}; \Theta)| - \epsilon, & \text{otherwise} \end{cases}$$

where $\epsilon > 0$.

The regression function f in nonlinear SVR has the following form

$$f(\mathbf{x}) = \sum_{i=1}^{M_t} (\alpha_i - \alpha_i^*) \mathcal{K}(\mathbf{x}_i, \mathbf{x}) + b \quad (11)$$

where the parameters $\{\{\alpha_i, \alpha_i^*\}_{i=1}^{M_t}, b\}$ are found by using the training set and solving the SVR convex optimization problem. Any of the kernels defined by eqs. (3)–(5) may be used in (11).

D. Kernel PCA

In conventional principal component analysis (PCA), we form a centered sample covariance matrix $\mathbf{R}_x = \frac{1}{M_t} \sum_{j=1}^{M_t} (\mathbf{x}_j - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}})^T$ from M_t vector observations of a random process $\mathbf{x}_j \in \mathbb{R}^M$, where $\bar{\mathbf{x}}$ is the average vector among all data samples. We form the eigendecomposition of \mathbf{R}_x to yield the eigenvalues λ_i and corresponding eigenvectors \mathbf{v}_i , ordered so that $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M$. In cases where the data set $\{\mathbf{x}_j\}$ are highly correlated, the observations concentrate on a subspace \mathbb{S} of dimension S , an orthonormal basis for which is $[\mathbf{v}_1, \dots, \mathbf{v}_S]$, where typically $S \ll M$. The *principal components* of a vector \mathbf{x} are the projections of \mathbf{x} into \mathbb{S} . In effect, the vector \mathbf{x} is transformed into an S -dimensional vector. It is straightforward to show that these eigenvectors are the directions along which \mathbf{x} has maximum variance. This implies that the variance in the corresponding orthogonal complement subspace is minimum. Thus, PCA is a means of representing a random process with reduced dimensionality (S variables instead of M) in such a way that the mean-squared error between the reduced representation and the original \mathbf{x} is minimized.

The lower dimensionality representation can also assist with visualization of the clustering behaviour of the feature space, as we show later.

Kernel PCA can introduce a nonlinear mapping into the feature subspace. Kernel PCA first maps the data into some feature space \mathcal{U} via a (usually nonlinear) function ϕ , and then performs linear PCA on the mapped data. Since PCA is a linear algebra construct, it involves only inner product evaluations, and thus it is possible to perform kernel PCA without explicitly evaluating the transformation ϕ . The details of this process, and specifically the evaluation of the nonlinear principal components, are well described in [25] and consequently are omitted here.

E. Partial Least Squares Regression

Partial least squares (PLS) is an extension of PCA, and is useful in the pipeline inspection problem for reduced dimensional regression. We arrange the feature vectors into a matrix $\mathbf{X} \in \mathbb{R}^{M_t \times N_r}$, where each row is a set of features extracted from one image segment. We also similarly arrange the y_i into a vector $\mathbf{y} \in \mathbb{R}^{M_t}$.

The goal in PLS is to predict³ $\mathbf{Y} \in \mathbb{R}^{M_t \times N_o}$ using components from \mathbf{X} . The dependent variables y are referred to as *response variables*, and the input variables \mathbf{x} are the *predictors*. It is assumed that $N_r > N_o$. The prediction is done by extracting a set of orthogonal factors called *latent variables* (not directly observed or measured) which have the best predictive power of \mathbf{Y} . PLS regression (PLSR) is particularly useful when the columns of \mathbf{X} and \mathbf{Y} are highly correlated, and \mathbf{X} and \mathbf{Y} exhibit strong cross-correlations. The objective criterion for constructing components in PLS is to sequentially maximize the covariance between the response variable and a linear combination of the predictors [30], [31].

PLS regression searches for a set of components that perform a simultaneous decomposition of \mathbf{X} and \mathbf{Y} with the constraint that these components explain as much of the covariance between \mathbf{X} and \mathbf{Y} as possible. It is this step which generalizes PCA. PLS is followed by a regression step where the decomposition of \mathbf{X} is used to predict \mathbf{Y} .

PLS can also be kernelized. Kernel-based PLS is an extension of PLS and employs kernels to model nonlinear data relations, [30], [32], [33].

V. FEATURE SELECTION AND DIMENSIONALITY REDUCTION

As we mentioned in Sect. III, feature selection plays an important role in the machine learning process. Our goal is to extract the N_r “most relevant” features from the set of N_c candidate features. In this section, we discuss two automated methods for accomplishing this purpose. One of these uses mutual information, whereas the other exploits the inherent properties of the PCA and PLS techniques for dimensionality reduction.

A. Feature Selection Using Maximum Mutual Information

The mutual information of two discrete random variables U and V whose samples are u_i and v_j respectively, is defined as follows

$$\mathcal{M}(U; V) = \sum_{v_j \in V} \sum_{u_i \in U} p(u_i, v_j) \log \frac{p(u_i, v_j)}{p(u_i)p(v_j)} \quad (12)$$

where $p(u, v)$ is the joint probability distribution function of U and V , and $p(u)$ and $p(v)$ are the marginal

³In the general PLS formulation, the response variables \mathbf{Y} can have multiple columns and thus is represented as an $M_t \times N_o$ matrix rather than a vector, as it is in this application.

probability distribution functions of U and V , respectively.

With the *maximal relevance criterion*, the selected features are required, individually, to have the largest mutual information with the target class y , reflecting the largest dependency on y . In the pipeline inspection context, this idea can be explained as follows. We first form an augmented feature matrix $\tilde{\mathbf{X}} \in \mathbb{R}^{M_t \times N_c}$ from all the observed data. The i th row contains all N_c candidate features extracted from the i th image segment, whereas the n th column \mathbf{x}_n contains the values of the n th feature over all M_t image segments. We then evaluate each $\mathcal{M}(\mathbf{x}_n; \mathbf{y})$, $n = 1, \dots, N_c$ in (12) where \mathbf{y} is the vector of all the y_i . The reduced feature matrix $\mathbf{X} \in \mathbb{R}^{M_t \times N_r}$ has columns corresponding to the features with the largest N_r such values of \mathcal{M} . Various computational aspects involved with evaluating the mutual information in (12) are discussed in [28].

B. Minimizing Redundancy

The difficulty with the maximal relevance criterion is that it is likely the features selected according to maximal-relevance criterion could have rich redundancy, i.e., the dependency among these features could be significant. When two features highly depend on each other, the respective class-discriminative power would be relatively invariant if one of them were removed. Therefore, a more desirable choice is to select features that are both relevant (i.e., have high mutual information with the target class) but also have minimum mutual redundancy.

One suboptimal implementation of the above idea using an ‘incremental search’ method is explained next. This method is used in our experiments. Suppose that the set of N_r best features to select is denoted by \mathcal{A} , and the set of all N_c available features is denoted by \mathcal{X} . The first member of \mathcal{A} is the feature with maximum mutual information with the target value \mathbf{y} . Then, suppose we already have \mathcal{A}_{m-1} , the feature set with $m-1$ best features. The task is to select the m th feature from the remaining set $\tilde{\mathcal{A}} = \{\mathcal{X} - \mathcal{A}_{m-1}\}$. This can be done by solving the following optimization problem which implements a tradeoff between having maximum relevance and minimum redundancy [29]

$$\max_{\mathbf{x}_j \in \tilde{\mathcal{A}}} \left\{ \mathcal{M}(\mathbf{x}_j; \mathbf{y}) - \eta \frac{1}{m-1} \sum_{\mathbf{x}_i \in \mathcal{A}_{m-1}} \mathcal{M}(\mathbf{x}_j; \mathbf{x}_i) \right\} \quad (13)$$

where $\eta > 0$ is the regularization or trade-off parameter. The best value for the regularization parameter can be found using a simple grid search, for example.

C. Low-dimensional Representation

Kernel PCA (as well as PLS and its kernel variant), inherently reduce the dimensionality of the feature space

by exploiting the hidden correlations that exist in the feature set. The representation of the data in the dimensionally reduced latent variable space given by these methods is most representative of the original data and can be used as features for training the machine learning algorithms.

The low-dimensional representation (2D or 3D) is also useful by itself, as it gives a view of the geometric properties of the data samples, and can help an MFL inspection operator to visually recognize metal defects.

It is often difficult to choose the number of latent variables to best represent the feature space. In this respect, the mutual information criterion discussed previously is effective for selecting the most relevant components.

VI. EXPERIMENTAL RESULTS

In this section, we examine the performance of the machine learning algorithms we have discussed in the analysis of real MFL images. We look at the two primary issues related to this problem: 1) the binary classification of image segments into (i) injurious or metal loss defects, and (ii) non-injurious, safe or non-harmful objects, and 2) estimation of the severity of defects. In all experiments, we use real MFL data derived from actual pipeline inspection runs. With real data, it is sometimes difficult to guarantee that we have correct labeling for all our training data. For our classification experiments, some target values y_i corresponding to each \mathbf{x}_i may not indicate the true class in some training samples. The reason is that the metal defects are not easily accessible and verifiable; secondly, the labeling of data is done through manual inspection by an NDT expert. Thus, the reference training data are subject to some technical errors and it is likely that we have, for example, some small or complex metal-loss samples mislabeled as non-injurious, and vice-versa. So, in our particular numerical analysis, we do not expect to get 100% performance.

Pipelines used in the experiments are of the ‘electric resistance welded’ (ERW) long-seam type. The pipe material is standard carbon steel which is used throughout North America for oil and natural gas pipelines, including X42 and X52 pipes as specified in ‘API specifications 5L’, or the corresponding 290 and 359 grades as specified in ‘CSA Z245.1’.

Hall effect sensors are used to measure the circumferential or transverse (B_y) component of magnetic field, where the excitation flux is in the same (y) direction. The output sensitivity of the sensors is 2.5 mV per Gauss. Sensor spacing is 3.32mm in y direction. Since the sampling rate is time-based and the ILI tool varies in speed due to the variation of transport flow, the sample spacing in the x direction can vary, but is generally in the range of 2mm to 3mm. Due to its compressibility, the flow speed of natural gas varies more than that of crude oil or other liquids.

We now briefly describe the processing steps involved in the following experiments. Initially, the received MFL

image is segmented to isolate regions which may contain an object of interest such as some form of metal-defect or injurious event, or some form of benign event. Then we used a two-stage de-noising and smoothing process: First, the level of MFL measurement noise (also referred to as ‘seamless pipe noise’ or SPN) is reduced, and the MFL image smoothed, by using a pixel-wise 2D adaptive Wiener filtering method based on statistics estimated from a local neighborhood of each pixel [34]. Then to further remove the noise, we used a 2D discrete stationary wavelet transform (with Daubechies wavelets) with soft-thresholding to de-noise the result [35].

Once the image segments have been de-noised, we use one of the feature reduction methods of Sect. V to automatically pick the N_r most discriminating features. However, the selected set is not unique as some features are statistically dependent. The designer can choose any of these dependent features with only a slight change in performance.

As described above, the initial list of simple candidate features can be large, since at the beginning we do not know which properties of the MFL image are relevant in solving the problem. In the first and second experiments that are explained in this section, a total of $N_c = 81$ simple candidate features are extracted. In the third and fourth experiments, the number of simple candidate features of the MFL images are $N_c = 406$, in which $N_r = 24$, or $N_r = 36$ is used.

The initial set of N_c candidate features consists of quantities commonly used in image classification problems, as explained in [24], [34]. Space does not permit the listing of all specific feature items used for our experiments. However, features of interest include such quantities as statistical properties of average background image information, magnitude, size and scale of the metal defect, orientation, maximum to average ratio of the segment, statistical measurements of the segment, such as means, variances, covariances and coefficients of 2D autocorrelation, aspect ratios, absolute peak values, major principal components of the 2D discrete cosine transform (DCT) of the MFL image segment, various parameters such as skewness relating to the histogram of the segment, various statistical moments (central), normalized integrated optical density, coefficients of the wavelet transform, etc.

In the experiments of this section, we consider a wide range of possible defects that are commonly encountered in practical MFL inspection. These include external corrosion pits, linear crack-like indications for which many are associated with the ERW long seam, and various types of local metal-losses and crack-like defects. Since the defects we wish to process are both internal and external to the pipe wall, the training data sets we employ in the experiments must also include both internal and external defect samples.

The following procedures were followed for the implementation of the classification and regression models.

The Platt's *sequential minimal optimization* method is used (see [26]) to solve the quadratic programming convex optimization problem associated with the SVR. For the RLS method, the regression coefficient vector \mathbf{v} in (9) is determined using a straightforward linear system solver. For each of the regression methods considered in this paper, the associated design parameter values are determined using a grid search and a cross-validation procedure [24] using the training data for a reference⁴.

A. Detection of Injurious Defects

Here, we show results for a binary detection case. In this set of experiments, suspicious image segments are classified either into a *injurious* class (including metal loss defects and crack-like anomalies), and a *non-injurious* object class. Then, given that a particular image segment is determined to be a 'injurious' defect, the second step is to estimate the depth of the defect, so that the urgency of repair can be established. This step is discussed in the next subsection.

In the following experiments, we wish to determine the percentage accuracy of our machine learning inspection procedures for classifying defects into their respective classes as described above. Thus, the class of all image segments used in these experiments must be determined in advance. This is done by an NDT technician who examined all the measured MFL image samples considered in this experiment on a computer screen, and based on their experience, manually assigned a classification label (i.e., injurious or non-injurious) to each image segment. Some of these data samples are then randomly selected as training data for the classifier, while the remaining samples are tested by the classifier. This process may be repeated many times, each time drawing a different sample of training samples. The accuracy of our method is then established by comparing the decision made by the classifier to the corresponding value determined by the analyst, for each test case. The actual number of image segments available depend on the experiment and is given later.

In the detection or classification problem, the two classes are

- 1) The injurious class, including metal loss on a girth weld (labelled as ML-GW), metal loss on a 'long seam weld' (ML-LSW), metal loss on the plain pipe-wall (ML-PW), manufacturing metal-loss samples on a long seam weld (MML-LSW), and a crack-like anomaly on a long seam weld (CL-LSW). The MML-LSW label denotes metal-losses associated with the manufacturing process

or any pipeline installation material loss. The corresponding target value is assigned to be $y = 1$.

- 2) The non-injurious or benign class including measurement noise and benign pipeline manufacturing anomalies (such as dents) and MFL image irregularities that may be confused with metal-loss, but are not harmful to pipeline operation. In general, non-injurious class anomalies do not require replacement or repair of the pipe, and are considered as safe. The corresponding target value is $y = 2$.

In the first experiment, MFL data from an 8-inch pipeline is used. We have a total of 1529 image segments, including 656 injurious, and 873 non-injurious samples. The data in the injurious class include 197 ML-PW, 132 ML-LSW, 58 ML-GW, 116 MML-LSW and 153 CL-LSW samples. 90% of the available data samples are randomly selected to be used for training, while the remaining 10% are used for testing.

A comparison amongst the different identification techniques is shown in Table I. The parameters for each method were estimated by optimizing the performance using a grid search and cross-validation procedure using the training data. The linear discriminant analysis (LDA) method shown in the table is a well-known classification technique used as a reference in our comparison [24]. In a manner similar to [5], we also used a multilayer Perceptron network (MLPN) (with 18 hidden neurons, and employing a conjugate gradient method for training). We also evaluate performance using a radial-basis function network (RBFN), [23], [24]. As can be seen, RLS, SVR and kernel PLSR methods with a Gaussian kernel are all powerful methods and have approximately similar performance in this experiment, but are all better than the LDA, MLPN [5] and RBFN methods. For the SVR method, the identification results with $N_r = 36$ corresponds to the following number of misclassifications in each class: 19 misclassification cases in the injurious class (including 12 ML-PW errors, 3 ML-GW, and 4 CL-LSW), and 13 misclassification cases in the non-injurious or benign class. The result by the kernel PLSR method corresponds to using 9 major latent vectors. By experimenting with various scenarios (not discussed in this paper) we also noticed that the SVR method, compared to others, can provide good detection performance even with relatively small number of training samples.

Performance was also evaluated for the $N_r = 24$ case; however, as can be seen in the lower part of the Table I, the results for the SVR and kernel PLSR methods did not vary significantly from $N_r = 36$ case. This is in contrast to the LDA method which showed a significant drop in performance for the $N_r = 24$ case.

Fig. 7 shows regression results using the kernel PLSR method (with a Gaussian kernel). Here, the data are sorted so that the first 656 samples correspond to class 1

⁴The optimum values of the parameters of the regression function depend on the particular experiment design and on the particular MFL image pre-processing, normalization and feature extraction stages used. For example, the values depend on the number of features, number of training data samples, scale and range value of each numerical feature, etc.

TABLE I

8-INCH PIPELINE: COMPARISON OF PERFORMANCE AMONG DIFFERENT INJURIOUS METAL DEFECT IDENTIFICATION METHODS. A TARGET VALUE $y = 1$ CORRESPONDS TO AN INJURIOUS DATA SAMPLE, WHILE A TARGET VALUE OF $y = 2$ CORRESPONDS TO A NON-INJURIOUS CLASS.

method	sensitivity, $p(\hat{y} = 1 y = 1)$	specificity, $p(\hat{y} = 2 y = 2)$	% average performance
LDA, $N_r = 36$	0.9	0.94	92%
Multilayer Perceptron Network, $N_r = 36$	0.931	0.976	95.37%
Radial-Basis Function Network, $N_r = 36$	0.942	0.951	94.64%
RLS, $N_r = 36$	0.976	0.978	97.69%
SVR with Gaussian kernel, $N_r = 36$	0.971	0.985	97.81%
Kernel PLSR with Gaussian kernel, $N_r = 36$	0.951	0.99	97.05%
LDA, $N_r = 24$	0.878	0.915	89.64%
SVR with Gaussian kernel, $N_r = 24$	0.97	0.979	97.44%
Kernel PLSR with Gaussian kernel, $N_r = 24$	0.959	0.977	96.79%

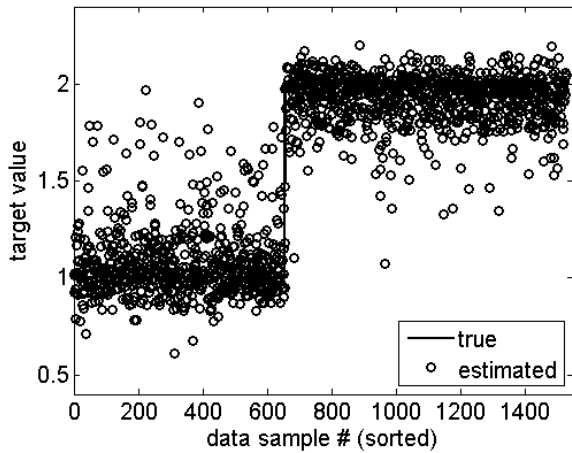


Fig. 7. 8-inch pipeline: Estimated (circles) and true (solid-line) target values using the Kernel PLS regression method with a Gaussian kernel. The data samples are sorted. $y = 1$ denotes injurious class and $y = 2$ denotes non-injurious class. $N_r = 36$.

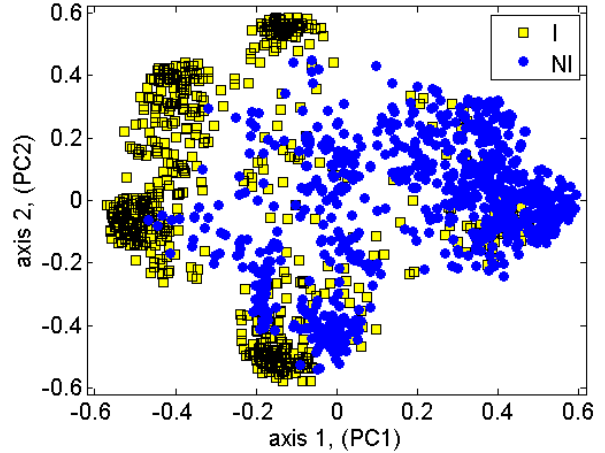


Fig. 8. 8-inch pipeline: Scatter plot of the projection of MFL data samples into the first and second major latent variables using the KPCA method with a Gaussian kernel. 'I' denotes injurious (metal-loss) samples (including ML-GW and ML-PW classes), and 'NI' denotes non-injurious samples.

defects (i.e., injurious class), and the remaining 873 are non-injurious anomalies. In this case, the function f is a step function, with its discontinuity between values 656 and 657. The figure shows how the kernel PLSR method performs in fitting the data. A classification error results when a data sample with index in $[1, 656]$ is above the value 1.5, and is below 1.5 when its index is in the $[657, 1529]$ range. From Table I, we see that this method correctly classifies the MFL samples 97% of the time.

We now show how dimensionality reduction of the feature space can aid in the visualization of the classifier. This can be readily accomplished using the kernelized PCA procedure. In this case, a Gaussian kernel was used. Fig. 8 shows the corresponding two-dimensional scatter plot for the same data set. Even though this two-dimensional representation is not sufficient for the most separable view of the two classes, it allows visualization on a sheet of paper. It is clear that this low-dimensional representation indeed shows reasonable geometric separation and clustering of subjects into injurious (shown by yellow rectangles with black edges) and non-injurious

(shown by blue circles) classes. There are of course, a few overlapping points and these would lead to classification errors, but performance would be improved by an increase of dimensionality. Fig. 8 corresponds to the projection of the data samples onto the first and second major latent variables, which are selected through the maximum mutual information method.

We now show a second example of how dimensionality reduction can assist in visualization. Here we consider only the injurious group of data. We used kernel PCA (KPCA), with a Gaussian kernel, to show how different types of metal-losses cluster geometrically in a two-dimensional space.

The input into the KPCA algorithm were feature vectors of dimension $N_r = 36$. Fig. 9 shows the relative magnitude of the first several major principal components (eigenvalues) of the MFL data kernel matrix. It is clear that the first two or three components dominate. This suggests that only a few latent variables with the highest PC magnitude could be enough to represent the data with small error. Fig. 10 shows the projection of

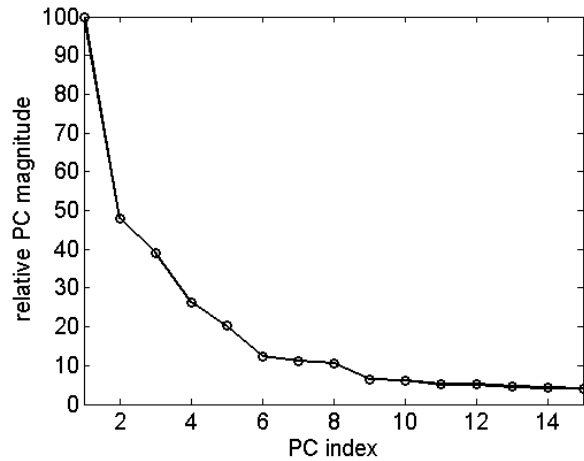


Fig. 9. 8-inch pipeline: Plot of the relative magnitude of a few major principal components (PCs or eigenvalues) of the MFL data kernel matrix using the Kernel PCA method with a Gaussian kernel.

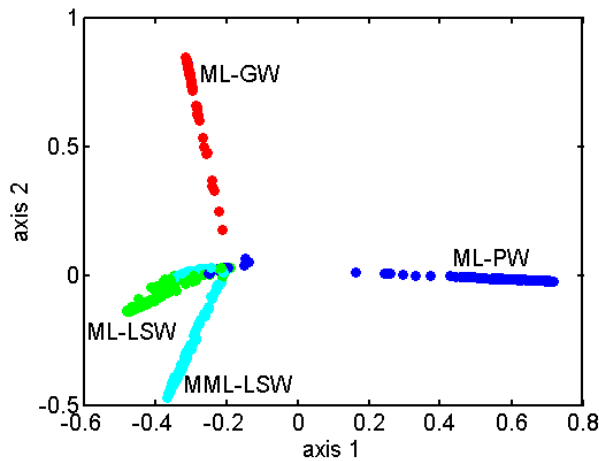


Fig. 10. 8-inch pipeline: Scatter plot of the projection of MFL data samples into the first and third major latent variables using the KPCA method with a Gaussian kernel. For clarification, green dots correspond to ML-LSW, and cyan dots correspond to MML-LSW ('manufacturing metal-loss samples on long-seam weld').

the feature vectors onto the first and third major latent variables. These particular latent variables were again chosen using the maximum mutual information criterion. As can be seen, there are some overlapping samples in the figure, because the dimensionality of two is too low to adequately cluster the data. Nevertheless, only two dimensions were able to capture the data clustering and show the behaviour of the classifier.

As an example of the usefulness of this approach, a new test MFL data sample can be projected into this 2D space and the user/operator can then infer or visualize approximately the class of test sample based on how closely it is to the clusters for each class.

In the second experiment, inspection data for a 10-inch pipeline is used for performance evaluation. The performance numbers are shown in Table II. Here, there

are a total of 1919 samples, 909 of which correspond to injurious events and 1010 are non-injurious or benign anomalies. The data in the injurious class include 403 ML-PW, 120 ML-LSW, 109 ML-GW, 137 MML-LSW and 140 CL-LSW samples. The methodology used in this case is similar to that used in the first experiment. The injurious samples mis-classified with the SVR method comprise 11 ML-PW errors, 2 ML-LSW errors, 10 ML-GW errors, 1 MML-LSW, and 1 CL-LSW error. As can be seen, the kernel PLSR method outperforms linear PLS regression method.

B. Metal-Loss/Crack Depth Estimation

In this case, we are given an image segment which corresponds to a injurious (including non-benign metal loss and crack-like defects), and we wish to assess its severity, or depth, relative to the pipe wall thickness. The MFL image data set used in the following experiments is distinct from that used in previous subsection, and corresponds to several 8, 10 and 12 in. pipe.

We use two approaches for this problem. In the first approach, we quantize the metal-loss/crack depth into two levels:

$y = 1$, 'less-severe loss': metal-loss/crack depth is less than a threshold δ .

$y = 2$, 'severe loss': metal-loss/crack depth is greater than or equal to the threshold δ .

In the second approach, we treat the depth as a continuous variable and use regression methods to estimate the depth. The defect depth refers to the maximum depth of the defect.

The dimensions of the metal defect anomalies used in this subsection are measured by a NDT technician at an actual dig-site, using various techniques.

In the following two experiments, a total of 58 samples of various metal loss and crack-like defects are used as our training/test data. Defect depths vary in the range of 10% to 61% of wall-thickness, where they reasonably uniformly cover a full range of values in the above depth range. Most of the crack-like defects are adjacent to or on the 'long-seam weld'. The data set contains both internal and external defects. The data set comprises 33 metal losses, 4 metal losses on a 'long seam weld' (LSW), 20 cracks and crack-like events on an LSW, and 1 crack-like event adjacent LSW.

In our third experiment, we show results from the quantized depth estimation approach. In this case, we use $\delta = 30\%$ of pipeline wall-thickness as our severity threshold. Performance results are shown in Table III. There are 29 data samples in each class. It is seen that the SVR method with a polynomial kernel (of degree 7) performs better than the other methods. The nearest neighbor method shown in the table is a well-known classification technique used as a reference in our comparisons [24].

TABLE II

MFL DATA FOR A 10-INCH PIPELINE: COMPARISON OF PERFORMANCE AMONG DIFFERENT METAL-LOSS IDENTIFICATION METHODS. $y = 1$ CORRESPONDS TO AN INJURIOUS METAL DEFECT, AND $y = 2$ CORRESPONDS TO A NON-INJURIOUS OR BENIGN CLASS.

method	sensitivity, $p(\hat{y} = 1 y = 1)$	specificity, $p(\hat{y} = 2 y = 2)$	% average performance
LDA, $N_r = 36$	0.879	0.863	87.12%
RLS, $N_r = 36$	0.968	0.966	96.72%
SVR with Gaussian kernel, $N_r = 36$	0.973	0.969	97.09%
Kernel PLSR with Gaussian kernel, $N_r = 36$	0.946	0.977	96.17%
PLS regression (PLSR), $N_r = 36$	0.88	0.862	87.12%

TABLE III

COMPARISON OF PERFORMANCE AMONG DIFFERENT METAL-LOSS SEVERITY IDENTIFICATION METHODS. $y = 1$ CORRESPONDS TO 'LESS-SEVERE LOSS' OR SHALLOW LOSS-DEPTH, AND $y = 2$ CORRESPONDS TO 'SEVERE LOSS' OR DEEP LOSS-DEPTH. $N_r = 36$. THE SEVERITY THRESHOLD IS $\delta = 30\%$ OF PIPELINE WALL-THICKNESS.

method	sensitivity, $p(\hat{y} = 1 y = 1)$	specificity, $p(\hat{y} = 2 y = 2)$	% average performance
Nearest Neighbor	0.621	0.828	72.41%
RLS	0.931	0.897	91.38%
SVR with linear kernel	0.931	0.931	93.1%
SVR with Gaussian kernel	0.966	0.931	94.83%
SVR with polynomial kernel	1	0.966	98.28%
Kernel PLSR with Gaussian kernel	0.966	0.897	93.1%

The fourth experiment shows results for the continuous-valued depth estimation approach. The data set is the same as that used in the third experiment. Fig. 11 shows the metal-defect depth estimation results using the SVR method with a polynomial kernel of degree 9. With reference to Eqs. (3) and ε -insensitive loss function of SVR, the parameters are $\epsilon = 0.05$, $\gamma = 0.001$, $d = 9$. The dashed line corresponds to the output of the regression function f using the SVR method, whereas the solid line (with sample points marked by circles) shows the true relative depth of the corresponding data samples. With this figure, the exact true distribution and range of defect depths can be seen. The mean-squared error for depth-estimation in this experiment is 7.73% (of pipeline wall-thickness). Given the simple set of features used, this figure is of sufficient accuracy, especially considering the complexity in the shape and structure of various actual metal defects used in this case, to determine repair schedules for the pipeline. In comparison, in a separate experiment (not shown), conducted using the radial basis function network (RFBN), the rms error was determined to be 9.94%.

An example, further to that of Fig. 4, of a defect that was included in the above experiment, is shown in Fig. 12. This shows a two-dimensional plot of the de-noised MFL signal amplitude of an actual metal-loss on an LSW for an 8-inch pipe. In contrast, Fig. 4 shows an MFL response of an actual metal loss on plain pipeline wall. In this case, the running speed of the ILI tool was 0.778 m/s. The rectangular area sampled by the Hall effect sensors before denoising was 58×115 sample points. The depth estimated using the SVR method discussed above is 45.7%, which corresponds to an error of 3.3% relative to the value obtained by field

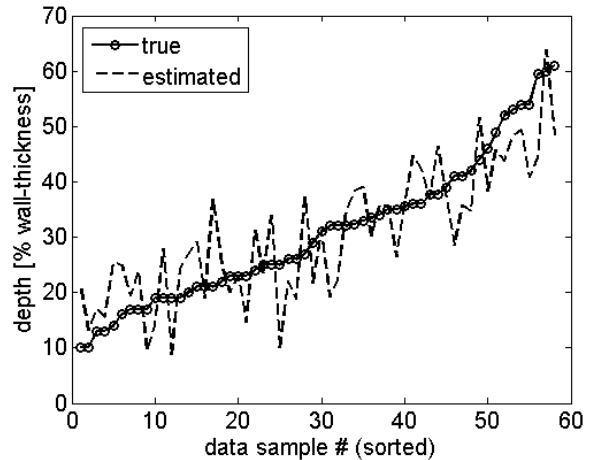


Fig. 11. Metal-loss/crack-depth estimation: Sorted estimated (dotted-line) and true (solid-line) depth values using SVR with a polynomial kernel. $N_r = 24$.

measurement. The signal amplitude and axis values in the figure have been scaled. It may be seen from this figure that the defect shape is very irregular. The fact that the proposed machine learning technique is capable of producing an accurate result shows the effectiveness of these methods over previous traditional methods which assume a very limited class of defect geometries, e.g. [15].

VII. CONCLUSION

In this paper, a complete machine learning procedure for the inspection of MFL images from pipelines is presented. This includes feature extraction, feature selection, pattern-recognition and regression methodologies

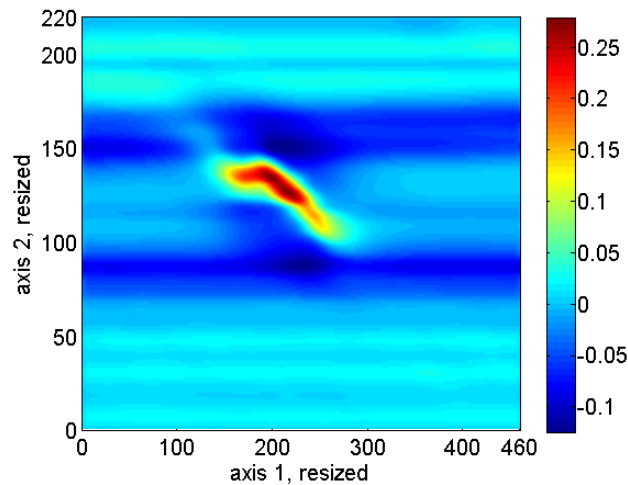


Fig. 12. 2D view of an example MFL image (after de-noising) showing a metal-loss on LSW.

to detect major metal-defects in a pipeline. We also include two techniques for determining the depth and severity of the defect. In many experiments using real MFL data, we observed that by using a subset of the most discriminating features that are both dependent on the target values and also satisfy a minimum redundancy criterion, kernelized techniques such as RLS, SVR and PLS provided a reasonable performance in metal-loss detection, and in metal-loss severity estimation. The average detection performance in recognizing major injurious metal defects versus non-injurious or benign anomalies was over 96%. The root-mean square error in defect depth estimation was less than 8%. Furthermore, the kernel PCA method provided a low-dimensional representation of the MFL data and was presented as an effective visualization tool.

ACKNOWLEDGMENT

This work was supported by a collaborative research development (CRD) grant of the Natural Sciences and Engineering Council of Canada (NSERC) and Intratech Inline Inspection Services Ltd., under Grant CRDPJ 349624-06. The authors would like to thank Intratech Inline Inspection Services Ltd., and specifically its president, Ron Thompson, for supporting this research and providing the required MFL data. We also would like to thank Nazanin Samavati and Reza K. Amineh for helpful discussions during the early stages of the research.

REFERENCES

- [1] V. E. Loskutov, A. F. Matvienko, B. V. Patramanskii, and V. E. Shcherbinin, "The magnetic method for in-tube nondestructive testing of gas and oil pipelines: The past and the present," *Russian Journal of Nondestructive Testing*, vol. 42, no. 8, pp. 493–504, Aug. 2006.
- [2] R. C. Ireland and C. R. Torres, "Finite element modelling of a circumferential magnetiser," *Sensors and Actuators A (Physical)*, vol. 129, no. 1-2, pp. 197–202, May 2006.
- [3] S. O'Connor, L. Clapham, and P. Wild, "Magnetic flux leakage inspection of tailor-welded blanks," *Measurement Science and Technology*, vol. 13, no. 2, pp. 157–162, Feb. 2002.
- [4] W. Lord, *Nondestructive Testing Monographs and Tracts, Volume 3 – Electromagnetic Methods of Nondestructive Testing*, New York: Gordon and Breach Science Publishers, 1985.
- [5] A. A. Carvalho, J. M. A. Rebello, L. V. S. Sagrilo, C. S. Camerini, and I. V. J. Miranda, "MFL signals and artificial neural networks applied to detection and classification of pipe weld defects," *NDT&E International*, vol. 39, no. 8, pp. 661–667, Dec. 2006.
- [6] K. C. Hari, M. Nabi, and S. V. Kulkarni, "Improved FEM model for defect-shape construction from MFL signal by using genetic algorithm," *IET Science, Measurement and Technology*, vol. 1, no. 4, pp. 196–200, July 2007.
- [7] A. Joshi, L. Udpa, S. Udpa, and A. Tamburrino, "Adaptive wavelets for characterizing magnetic flux leakage signals from pipeline inspection," *IEEE Transactions on Magnetics*, vol. 42, no. 10, pp. 3168–3170, Oct. 2006.
- [8] R. K. Amineh, S. Koziel, N. K. Nikolova, J. W. Bandler, and J. P. Reilly, "A space mapping methodology for defect characterization from magnetic flux leakage measurements," *IEEE Trans. Magnetics*, vol. 44, no. 8, pp. 2058–2065, Aug. 2008.
- [9] W. Han and P. Que, "A modified wavelet transform domain adaptive FIR filtering algorithm for removing the SPN in the MFL data," *Measurement*, vol. 39, no. 7, pp. 621–627, Aug. 2006.
- [10] M. Afzal and S. Udpa, "Advanced signal processing of magnetic flux leakage data obtained from seamless gas pipeline," *NDT&E International*, vol. 35, no. 7, pp. 449–457, Oct. 2002.
- [11] J. Tao, Q. Peiwen, C. Liang, and L. Liang, "Research on a recognition algorithm for offshore-pipeline defects during magnetic-flux inspection," *Russian Journal of Nondestructive Testing*, vol. 41, no. 4, pp. 231–238, Apr. 2005.
- [12] S. Mukhopadhyay and G. Srivastava, "Characterisation of metal loss defects from magnetic flux leakage signals with discrete wavelet transform," *NDT&E International*, vol. 33, pp. 57–65, 2000.
- [13] Y. Zhang, Z. Ye, and X. Xu, "An adaptive method for channel equalization in MFL inspection," *NDT&E International*, vol. 40, no. 2, pp. 127–139, Mar. 2007.
- [14] Z. Zeng, L. Xuan, Y. Sun, L. Udpa, and S. Udpa, "Probability of detection model for gas transmission pipeline inspection," *Research in Nondestructive Evaluation*, vol. 15, no. 3, pp. 99–110, July/Sep. 2004.
- [15] R. K. Amineh, N. K. Nikolova, J. P. Reilly, and J. R. Hare, "Characterization of surface breaking cracks using one tangential component of magnetic leakage field," *IEEE Trans. Magnetics*, vol. 44, no. 4, pp. 516–524, Apr. 2008.
- [16] S. Mandayam, L. Udpa, S. S. Udpa, and W. Lord, "Invariance transformations for magnetic flux leakage signals," *IEEE Trans. Magnetics*, vol. 32, no. 3, pp. 1577–1580, May 1996.
- [17] J. Philip, C. B. Rao, T. Jayakumar, and B. Raj, "A new optical technique for detection of defects in ferromagnetic materials and components," *NDT&E International*, vol. 33, no. 5, pp. 289–295, Jul. 2000.
- [18] H. Goedecke, "Ultrasonic or MFL inspection: Which technology is better for you?" *Pipeline and Gas Journal*, vol. 230, no. 10, pp. 34–41, Oct. 2003.
- [19] C. Mandache and L. Clapham, "A model for magnetic flux leakage signal predictions," *Journal of Physics D: Applied Physics*, vol. 36, no. 20, pp. 2427–2431, Oct. 2003.
- [20] J. B. Nestleroth and R. J. Davis, "Application of eddy currents induced by permanent magnets for pipeline inspection," *NDT&E International*, vol. 40, no. 1, pp. 77–84, Jan. 2007.
- [21] R. W. Tucker, S. W. Kercel, and V. K. Varma, "Characterization of gas pipeline flaws using wavelet analysis," in *Proceedings SPIE, Sixth International Conference on Quality Control by Artificial Vision*, 2003, pp. 485–493.
- [22] V. N. Vapnik, *Statistical Learning Theory*. John Wiley and Sons, 1998.
- [23] S. Haykin, *Neural Networks, A Comprehensive Foundation*, 2nd ed. Prentice Hall, 1999.
- [24] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 2nd ed. USA: Elsevier Academic Press, 2003.
- [25] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda and B. Schölkopf,

- "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 181–201, Mar. 2001.
- [26] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, Aug. 2004.
- [27] T. Evgeniou, M. Pontil, and T. Poggio, "Regularization networks and support vector machines," *Advances in Computational Mathematics*, vol. 13, no. 1, pp. 1–50, Apr. 2000.
- [28] N. Kwak and C.-H. Choi, "Input feature selection by mutual information based on Parzen window," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1667–1671, Dec. 2002.
- [29] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
- [30] R. Rosipal and N. Krämer, "Overview and recent advances in partial least squares," in *Subspace, Latent Structure and Feature Selection Techniques*, C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, Eds. Lecture Notes in Computer Science, Springer, 2006, pp. 34–51.
- [31] H. Abdi, "Partial least square regression (PLS regression)," in *Encyclopedia of Measurement and Statistics*, N. J. Salkind, Ed. Thousand Oaks, 2007, pp. 740–744.
- [32] R. Rosipal and L. J. Trejo, "Kernel partial least squares regression in reproducing kernel Hilbert space," *Journal of Machine Learning Research*, vol. 2, no. 2, pp. 97–123, 2002.
- [33] N. Krämer and M. L. Braun, "Kernelizing PLS, degrees of freedom, and efficient model selection," in *Proc. Int. Conf. Machine Learning*, 2007, pp. 441–448.
- [34] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. USA: Prentice Hall, 2007.
- [35] J. C. Pesquet, H. Krim and H. Carfantan, "Time-invariant orthonormal wavelet representations," *IEEE Trans. Signal Processing*, vol. 44, no. 8, pp. 1964–1970, Aug. 1996.