Visual Inspection of Precision Instruments by Least-Squares Outlier Detection

Masafumi Takimoto\textsuperscript{1}, Masakazu Matsugu\textsuperscript{1} and Masashi Sugiyama\textsuperscript{2}

\textsuperscript{1}Canon Inc Corporate R&D Headquarters, Tokyo, Japan
\textsuperscript{2}Tokyo Institute of Technology, Tokyo, Japan

Abstract: Visual inspection is one of the most important processes in precision instrument factories for screening out products of poor quality. Usually this is carried out by human experts who went through long-term training sessions, so there is a strong demand for automating this process in order to reduce production costs. In this paper, we apply a recently-developed outlier detection method called least-squares outlier detection (LSOD) to this task and demonstrate that inferior products can be successfully detected. LSOD can utilize knowledge of inliers for enhancing outlier detection performance, so it suits well to visual inspection in industries. Furthermore, LSOD is equipped with automatic model selection mechanism and, hence, users do not have to grapple with parameter tuning.

Keywords: visual inspection, inlier-based outlier detection, density ratio, importance

1 Introduction

In recent years, various humanoid robots such as ASIMO (HONDA), HRP-2/HRP-3 (Kawada Industries, Inc. / Advanced Industrial Science and Technology / Kawasaki Heavy Industries, Ltd.) and SDR-4X/QRIO (Sony) were developed. With the advancement of information technologies, these robots started to undertake substantial work in the real world such as facility guidance.

On the other hand, industrial robots have been in active use in precision instrument factories, e.g., at assembly lines. Even so, many manufacturing processes are still accomplished by human experts since some tasks are hard to be automated. Visual inspection of products is one of such hard tasks because of a wide variety of complicated defect patterns of products; even human experts need to go through a long-term training process to acquire necessary skills. Since visual inspection by human experts is very costly, there is a strong demand for automating the visual inspection processes.

In this paper, we focus on visual inspection of particular rollers in electrophotography products. The rollers are covered with black gummy materials and the roughness and uniformity of the surface need to be inspected. Our goal is to screen out rollers with poorly-conditioned surfaces.

If non-uniform patterns of the roller’s surface were defined in advance, inferior products would be detected by general pattern recognition technologies. However, this supervised approach is not reliable in the current setup because of the following reasons. It is not possible to define all defect patterns beforehand since abnormal patterns are so diverse and we can observe only a small number of abnormal patterns in practical product lines. Furthermore, the definition of (ab-)normality needs to be updated frequently since the quality of the products is usually improved rapidly. Thus, if we took this supervised approach in visual inspection, we would need to keep the inspection system updated using new abnormal patterns. This would be too expensive in practice.

On the other hand, a number of normal products are available easily. So we would like to utilize such information for improving the detection performance. Here the framework of inlier-based outlier detection [1] comes in handy. In this paper, we apply a recently-developed inlier-based outlier detection method called least-squares outlier detection (LSOD) [3] to this task and demonstrate that inferior products can be successfully detected. LSOD is equipped with automatic model selection mechanism and therefore users do not have to grapple with parameter tuning, which is highly valuable in real-world applications.
In this section, we explain a statistical approach to inlier-based outlier detection. Suppose we have two sets of samples \( \{x_{trn}^j\}_{j=1}^n \) and \( \{x_{ten}^i\}_{i=1}^m \) in a domain \( D \subseteq \mathbb{R}^d \). The training samples \( \{x_{trn}^j\}_{j=1}^n \) are all inliers, while the test samples \( \{x_{ten}^i\}_{i=1}^m \) can contain some outliers.

The goal of inlier-based outlier detection is to identify outliers in the test set based on the training set consisting only of inliers [1]. More formally, we want to assign a suitable inlier score for the test samples \( \{x_{ten}^i\}_{i=1}^m \) such that the smaller the value of the inlier score is, the more plausible the sample is an outlier.

Let us consider a statistical framework of the outlier detection problem: suppose training samples \( \{x_{trn}^j\}_{j=1}^n \) are independent and identically distributed (i.i.d.) following a training data distribution with density \( p_{trn}(x) \) and test samples \( \{x_{ten}^i\}_{i=1}^m \) are i.i.d. following a test data distribution with strictly positive density \( p_{ten}(x) \). Within this statistical framework, test samples with low training data densities are regarded as outliers. However, \( p_{trn}(x) \) is not accessible in practice and density estimation is known to be a hard problem. Therefore, merely using the training data density as an inlier score may not be promising in practice.

In this paper, we propose to use the ratio of training and test data densities, called the importance, as an inlier score:

\[
\tilde{w}(x) = \frac{p_{ten}(x)}{p_{trn}(x)}
\]

If there exists no outlier sample in the test set (i.e., the training and test data densities are equivalent), the value of the importance will be one. The importance value tends to be small in the regions where the training data density is low and the test data density is high. Thus samples with small importance values are plausible to be outliers.

### 3 Unconstrained Least-Squares Importance Fitting (uLSIF)

As formulated above, inlier-based outlier detection is possible if the importance function is available. uLSIF [3] is an importance estimation method which is computationally efficient. In this section, we review the core idea of uLSIF.

In uLSIF, a linear importance model is used.

\[
\tilde{w}(x) = \sum_{l=1}^b \alpha_l \phi_l(x),
\]

where \( \phi_l(x) \) is a non-negative basis function and \( \alpha_l \) is a parameter. The parameters are determined so that the following objective function is minimized:

\[
\frac{1}{2} \int \left( \tilde{w}(x) - \frac{p_{ten}(x)}{p_{trn}(x)} \right)^2 p_{trn}(x) dx
= \frac{1}{2} \int \tilde{w}(x)^2 p_{ten}(x) dx - \int \tilde{w}(x) p_{ten}(x) dx + \frac{1}{2} \int \frac{p_{ten}(x)^2}{p_{trn}(x)} dx,
\]

where the last term in the right-hand side is a constant and therefore can be safely ignored. By the empirical approximation, the following optimization problem is obtained.

\[
\tilde{\alpha} = \arg \min_{\alpha} \left[ \frac{1}{2} \tilde{w}^T \tilde{H} \alpha - \tilde{h}^T \alpha + \frac{\lambda}{2} \alpha^T \alpha \right],
\]

where, for \( \phi(x) = (\phi_1(x), \ldots, \phi_b(x))^T \),

\[
\tilde{H} = \frac{1}{n_{ten}} \sum_{i=1}^{n_{ten}} \phi(x_{ten}^i) \phi(x_{ten}^i) \quad \text{and} \quad \tilde{h} = \frac{1}{n_{ten}} \sum_{i=1}^{n_{ten}} \phi(x_{ten}^i).
\]

\( \lambda \alpha^T \alpha / 2 \) is a regularization term. The solution \( \tilde{\alpha} \) is given analytically by

\[
\tilde{\alpha} = (\tilde{H} + \lambda I_b)^{-1} \tilde{h},
\]

where \( I_b \) is the b-dimensional identity matrix. Since elements of \( \tilde{\alpha} \) could be negative, it is modified as

\[
\hat{\alpha} = \max(0, \tilde{\alpha}),
\]

where \( 0_b \) is b-dimensional vector with all zeros. This is the solution of uLSIF, which can be computed analytically.

Let us consider the leave-one-out cross-validation (LOOCV) score of uLSIF:
\[
\frac{1}{n_w} \sum_{i=1}^{n_w} \left[ \frac{1}{2} \phi(x_i) \mathbf{a}_i^{(t)} \mathbf{a}_i^{(t)\top} - \phi(x_i) \mathbf{a}_i^{(t)} \right],
\]

where \( \mathbf{a}_i^{(t)} \) is a parameter learned without \( x_i^{(t)} \) and \( x_i \).

By using the well-known Woodbury inversion formula, \( \mathbf{a}_i^{(t)} \) can be expressed as

\[
\mathbf{a}_i^{(t)} = \max \left\{ 0, \frac{(n_w - 1)n_y}{n_x(n_y - 1)} \left( a + \frac{\phi(x^{(t)}_i) \mathbf{a}_w}{n_x - \phi(x^{(t)}_i) \mathbf{a}_w} \right) \right\}
\]

where

\[
a = A^{-1} \hat{h}, \quad a_w = A^{-1} \phi(x^{(t)}_i),
\]

\[
a_w = A^{-1} \phi(x^{(t)}_i),
\]

\[
A = \hat{H} + \frac{(n_w - 1) \lambda}{n_x} I_b.
\]

This expression implies that the matrix inverse needs to be computed only once (i.e., \( A^{-1} \)) for computing the LOOCV score. Note that the size of \( A^{-1} \) is \( b \times b \), which is independent of the numbers of training and test samples. Thus LOOCV can be carried out very efficiently without repeating the hold-out loop.

In the experiments, we use 100 Gaussian kernel bases located at a subset of training samples as the basis functions and the Gaussian width as well as the regularization parameter is chosen by LOOCV.

### 4 Experiments

We prepared 2883 roller samples obtained from real factory lines. The training set consists of 180 normal samples, while the test set contains 2626 normal samples and 77 abnormal samples.

Our goal is to detect the abnormal samples in the test set. The surface texture of rollers was captured by a special camera which can take a picture of roughened surface details emphatically. Since the roller has a round shape, the surface images are photographed several times and they are aligned together. Examples of surface images are shown in Figures 1 and 2. As can be seen from the figures, it is very difficult even for humans to detect anomaly from the appearances. The abnormal samples are classified into 13 categories by visual examiners in the factory.

We used a set of Gabor filters (Figure 3) to extract features from the texture of roller samples because Jones and Palmer [2] showed that the real part of complex Gabor functions fit very well with the receptive field weight functions found in simple cells in a cat’s striate cortices.

### Figure 1: Normal surface texture samples

### Figure 2: Abnormal surface texture samples

### Figure 3: set of Gabor filters for feature extraction

We prepared a set of Gabor filters with angle parameter \( \theta \) changed by 0.17 [rad):

\[
G(x, y) = \exp \left\{ -\frac{1}{2} \left( \frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2} \right) \right\} \exp \left\{ -\frac{2\pi R_x}{\lambda} \right\},
\]

\[
\begin{cases}
R_x = x \cos \theta + y \sin \theta, \\
R_y = -x \sin \theta + y \cos \theta.
\end{cases}
\]

\( \sigma_x, \sigma_y \) are the widths of the Gauss window along the horizontal and vertical axes and \( \lambda \) is the wavelength. We call the above set of Gabor filter an octave. We further considered multi-resolution filter sets where the
size (the width or height) of the filter is decreased to 1/2, 1/4, 1/8,…; by this, we obtained multiple octaves of Gabor filter sets. The smaller the size is, the higher the frequency of features. We regard an octave as being corresponding to a human visual projection area of each spatial frequency.

When Euler’s formula $\exp(i\theta) = \cos \theta + i \sin \theta$ is applied to the above expression of the Gabor filter, we have

$$G(x, y) = \exp \left\{ -\frac{1}{2} \left( \frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2} \right) \cos \left( \frac{2\pi R_x}{\lambda} \right) \right\} + i \exp \left\{ -\frac{1}{2} \left( \frac{R_x^2}{\sigma_x^2} + \frac{R_y^2}{\sigma_y^2} \right) \sin \left( \frac{2\pi R_x}{\lambda} \right) \right\}.$$ 

Based on this expression, we compute the absolute value for each position as

$$Z(x, y) = \sqrt{\left( \text{Re}G(x, y) \right)^2 + \left( \text{Im}G(x, y) \right)^2}.$$ 

The Gaussian widths $\sigma_x$ and $\sigma_y$ are determined as a function of $\lambda$ as follows:

$$\sigma_x = S_x \lambda, \quad \sigma_y = S_y \lambda,$$

where $S_x$ and $S_y$ are coefficients.

We computed the convolution with the Gabor filters in the frequency domain, and extracted the maximum values at each pixel over the set of Gabor filters with the same size. However, since the computation time increases exponentially as the size of the Gabor filter increases, we decided to reduce the size (the width and height) of the image to 1/2, 1/4, 1/8, ... until the size reaches the size of the Gabor filter. In our preliminary experiments, we confirmed that this simplification does not deteriorate the outlier detection performance. We obtained a texture image for each hierarchy as shown in Figure 4.

This figure shows that the brightness of regions with anomaly is higher than that of other areas. We extracted three values (the mean, dispersion, and maximum of the brightness) from each texture and formed 27-dimensional feature vectors from all 9 hierarchies.

We visualized the extracted features by multidimensional scaling (MDS) in Figure 5. The red points in the graph denote the normal samples (i.e., accepted rollers), while green points denote failure rollers. This plot shows that our feature extraction scheme can roughly separate normal samples from abnormal ones, although the separation is not perfect.

Then this feature extraction scheme was applied to all 2883 images and we calculated the outlier score by uLSIF. The smaller the score is, the higher the degree of outlyingness is. The ROC plot is provided in Figure 6, showing that the uLSIF result is better than the one-class support vector machine [4], which is a state-of-the-art outlier detection algorithm. The area under the curve (AUC) of one-class SVM was 0.959, while that of uLSIF was 0.978.
6 Conclusions

In this paper, we proposed a new method for visual inspection using direct importance estimation that allows us to avoid solving a substantially more difficult problem of density estimation. The challenges are that abnormal samples are hardly available and the tendency of normal/abnormal patterns changes continuously with a short lapse of time. Our inlier-based outlier detection approach can cope with such changing patterns well and can utilize normal samples which are often abundantly available in industrial applications. The experiments demonstrated that the proposed method is useful. We are currently exploring various possible applications of important estimation methods.

References