

# Difference Detection Between Two Images for Image Monitoring

Long Feng<sup>1</sup> and Peihua Qiu<sup>2</sup>

<sup>1</sup>*School of Mathematics and Statistics, Northeast Normal University, China*

<sup>2</sup>*Department of Biostatistics, University of Florida, USA*

## Abstract

In manufacturing industries, images are commonly used for quality control purposes. In such applications, if the quality of the products is good, then their images should be all similar to the image of a good-quality product. Therefore, comparison of images is a fundamental task in image-based quality control. This problem, however, is complicated in the sense that 1) observed images often contain noise, and 2) the related images need to be geometrically matched up first because images of different products could be geometrically mismatched due to the fact that the relative positions between a camera and different products are often not exactly the same. The first issue requires a statistical method that can remove noise, and the second issue is related the so-called image registration problem in the image processing literature. In this paper, we propose effective methods for detecting difference between two images of products, and our proposed methods can accommodate both noise and geometric mismatch mentioned above. Theoretical results and numerical examples show that they can work effectively in applications.

**Keywords:** Consistency; Continuity region; Edge detection; Hypothesis tests; Image registration; Rigid-body transformation.

# 1 Introduction

Quality control is a fundamental task for manufacturing industries to guarantee the quality of manufactured products (Qiu 2014). In modern manufacturing industries, images become a more and more popular tool for quality inspection because of the low cost and convenience in data acquisition (Kumar 2008, Yan et al. 2016). This paper aims to address a key problem about difference detection between two images for image-based quality control.

In manufacturing industries, images have been widely used for quality control purposes, including stress and strain analysis of products (Patterson and Wang 1991), anomaly detection of rolling processes (Jin et al. 2004), inspection of composite material fabrication (Sohn et al. 2004), quality control in liquid crystal display manufacturing (Jiang et al. 2005), structural health monitoring (Balageas et al. 2010), and so forth. In all these applications, one fundamental problem is to compare images of different products. This problem is challenging because of the following major reasons. First, images often have edges and other complicated structures, and the related image intensity surfaces would have jumps and other singularities (Gonzalez and Woods 1992, Qiu 2005). So, conventional methods for estimating smooth functions are usually inapplicable (Qiu 2007). Second, observed images would contain noise and other contaminations. When removing the noise and recovering the signal, edges and other important image structures need to be preserved, which makes the related image-denoising task especially challenging (e.g., Geman and Geman 1984, Gijbels et al. 2006, Mukherjee and Qiu 2011, Qiu 1998, Saint-Marc et al. 1991, Tomasi and Manduchi 1998). Third, images of different products are most likely geometrically mismatched because the relative positions between the camera and different products are hardly the same. To make the comparison between two images meaningful and reliable, the images should be geometrically matched up first, which is the so-called image registration problem in the image processing literature (Modersitzki 2009, Qiu and Xing 2013a, Zitova and Flusser 2003). Otherwise, the difference between two same but mismatched images could be large. For instance, consider a black-white image with black in the left half and white in the right half and another image obtained by moving the black-white image to the right by certain pixels. The difference between these two images could be relatively large, although the two images

are actually the same, except a geometric mismatch.

In the literature, there is a limited discussion about image comparison, mainly by researchers in computer vision and graphics (e.g., Davis et al. 1997). A major existing tool for comparing two images is to use a quantitative similarity or dissimilarity measure (Freire et al. 2002). Commonly used measures include the mean squared difference between two images, Pearson’s correlation coefficient of the observed image intensities of the two images, entropy of the difference between the two images, and so forth (Qiu and Nguyen 2008). However, these measures alone cannot tell whether two related images are significantly different or not, especially in cases when image misalignment is relevant. They do not take into account the complicated image structure, such as edges, either. Because observed images almost always contain noise and other contamination (e.g., spatial blur) in the image acquisition process (Gonzalez and Woods 1992, Qiu 2005), image comparison is a statistical problem. This problem has not been well discussed in the statistical literature yet. This paper tries to fill this gap.

In this paper, we propose several testing procedures for comparing two images. The novelty of our proposed methods is reflected in the fact that they can simultaneously accommodate all the issues mentioned above related to the edge structure of the images, the noise in the observed images, and the image registration. Theoretical results and numerical examples show that they work well in practice. The image registration issue is discussed in Section 2. Our proposed testing procedures based on proper image registration are described in Section 3. Some practical guidelines on parameter selection are given in Section 4. Some simulation results are presented in Section 5. A real-data example is discussed in Section 6. Several remarks conclude the article in Section 7. Some theoretical results about the consistency of the related parameter estimators are given in the appendix.

## 2 Rigid-Body Image Registration

As discussed in Section 1, images of two products obtained from a production line are often geometrically mismatched because the relative positions between the camera and the

products can hardly be exactly the same. So, to compare a pair of two images automatically by a computer, the two images need to be geometrically matched up first. This is the *image registration (IR)* problem in the literature (Qiu and Xing 2013a). As a convention, one image in the pair is called *reference image*, and the other one is called *moved image*. Then the IR problem can be described as follows. Assume that the two observed images follow the models

$$\begin{aligned} Z_R(x_i, y_j) &= R(x_i, y_j) + \varepsilon_R(x_i, y_j), \\ Z_M(x_i, y_j) &= M(x_i, y_j) + \varepsilon_M(x_i, y_j), \quad i, j = 1, \dots, n, \end{aligned} \quad (1)$$

where  $\{(x_i, y_j)\}$  are equally-spaced pixels,  $M(x_i, y_j)$  and  $R(x_i, y_j)$  are the true moved and reference image intensity functions,  $Z_R(x_i, y_j)$  and  $Z_M(x_i, y_j)$  are their observed versions, and  $\{\varepsilon_M(x_i, y_j)\}$  and  $\{\varepsilon_R(x_i, y_j)\}$  are i.i.d. random errors with mean 0 and unknown variances  $\sigma_R^2$  and  $\sigma_M^2$ . In (1), we assume that  $\sigma_R^2 = \sigma_M^2 = \sigma^2$  and the numbers of rows and columns of pixels are the same for convenience of presentation. All methods proposed in the paper can actually work well in cases when they are different. For the two true image intensity functions, it is assumed that they have the following relationship when the related products come from an “in-control” process:

$$M(T_1(x, y), T_2(x, y)) = R(x, y), \quad (x, y) \in \Omega, \quad (2)$$

where  $\Omega = [0, 1] \times [0, 1]$  is the design space of the reference image  $R$ , and  $\mathbf{T}(x, y) = (T_1(x, y), T_2(x, y))$  is an unknown geometric transformation. In manufacturing applications where the geometric difference between  $M(x, y)$  and  $R(x, y)$  is mainly due to the position move between the products and the camera, it is reasonable to assume that  $\mathbf{T}(x, y)$  is a *rigid-body transformation* defined by

$$\begin{aligned} T_1(x, y) &= x \cos(\phi) + y \sin(\phi) + \Delta x, \\ T_2(x, y) &= -x \sin(\phi) + y \cos(\phi) + \Delta y, \end{aligned} \quad (3)$$

where  $\phi$  is a rotation parameter, and  $\Delta x$  and  $\Delta y$  are the translation parameters in the  $x$ - and  $y$ -axes, respectively. By a rigid-body transformation, the Euclidean distance between any two points in an image will not change after the transformation. Then, the major goal of the IR problem is to estimate the parameters  $\boldsymbol{\theta} = (\phi, \Delta x, \Delta y)^T$  from the two observed images.

To this end, there could be two possible approaches. One is to estimate the parameters based on the continuity parts of the two image intensity functions (i.e., the parts of the images where the image intensity functions are continuous), and the other is based on the detected edges of the two images. These two approaches are discussed in the two subsections below.

In most industrial applications, images of different products are taken in a same or similar lighting environment or at similar positions relative to the camera. So, equation (2) is (roughly) appropriate for describing the geometric difference between the reference and moved images. However, in certain applications, it is possible that the lighting condition of different products is different. In such cases, the overall intensity levels of different images could be different. To accommodate such difference, we can standardize the overall intensity levels of different images before image registration or testing. More specifically, for any observed image  $\{Z(x_i, y_j)\}$ , its standardized image is defined as  $\{Z^*(x_i, y_j) = [Z(x_i, y_j) - \bar{Z}]/(Zmax - Zmin)\}$ , where  $\bar{Z}$ ,  $Zmax$  and  $Zmin$  are the mean, maximum and minimum of  $\{Z(x_i, y_j)\}$ , respectively. Also, in most manufacturing applications, the geometric difference among different product images would be small because the relative position between a product and the camera is often pre-determined and the geometric difference among different images is mainly due to small position moves of the products in the production process. However, if a big geometric difference among images is possible, then a pre-processing to roughly align the images might be helpful, before using our proposed methods discussed below. One possible pre-processing approach is to minimize  $\sum_{i,j=1}^n [Z_M(\mathbf{T}(x_i, y_j)) - Z_R(x_i, y_j)]^2$  with respect to  $\boldsymbol{\theta}$ , where  $\mathbf{T}(x, y)$  has the expression (3).

## 2.1 Intensity-based image registration

For a given point  $(x, y) \in \Omega$ , assume that  $M$  has the first-order partial derivatives at  $(x, y)$  and  $\mathbf{T}(x, y) - (x, y)$  is small, then by the Taylor's expansion, we have

$$M(T_1(x, y), T_2(x, y)) = M(x, y) + M'_x(x, y)(T_1(x, y) - x) + M'_y(x, y)(T_2(x, y) - y) + o(\|\mathbf{T}(x, y) - (x, y)\|),$$

where  $M'_x(x, y)$ ,  $M'_y(x, y)$  are the partial derivatives of  $M$  at  $(x, y)$ , and  $\|\cdot\|$  is the Euclidean norm. Combining the above equation with (2), we have

$$\begin{aligned}
R(x, y) &\approx M(x, y) + M'_x(x, y)(T_1(x, y) - x) + M'_y(x, y)(T_2(x, y) - y) \\
&= M(x, y) - M'_x(x, y)x - M'_y(x, y)y + \left( M'_x(x, y)x + M'_y(x, y)y \right) \cos(\phi) \\
&\quad + \left( M'_x(x, y)y - M'_y(x, y)x \right) \sin(\phi) + M'_x(x, y)\Delta x + M'_y(x, y)\Delta y \\
&\approx M(x, y) + \left( M'_x(x, y)y - M'_y(x, y)x \right) \phi + M'_x(x, y)\Delta x + M'_y(x, y)\Delta y.
\end{aligned}$$

In the last “ $\approx$ ”, we have used the results that  $\cos(\phi) \approx 1$  and  $\sin(\phi) \approx \phi$  when  $\phi$  is small. To estimate parameters  $\boldsymbol{\theta} = (\phi, \Delta x, \Delta y)^T$ , it is then natural to consider the following sum of squares of the approximation errors, after  $R(x, y)$  and  $M(x, y)$  are replaced by their observed versions:

$$\begin{aligned}
Q(\phi, \Delta x, \Delta y) &= \sum_{(x_i, y_j) \in \bar{D}_M} \left[ Z_M(x_i, y_j) - Z_R(x_i, y_j) + \left( M'_x(x_i, y_j)y_j - M'_y(x_i, y_j)x_i \right) \phi \right. \\
&\quad \left. + M'_x(x_i, y_j)\Delta x + M'_y(x_i, y_j)\Delta y \right]^2, \tag{4}
\end{aligned}$$

where  $\bar{D}_M$  denotes the set of all pixels at which  $M(x, y)$  is continuous. Let  $\mathbf{Y}$  be a long vector, consisting of  $\{Z_M(x_i, y_j) - Z_R(x_i, y_j), (x_i, y_j) \in \bar{D}_M\}$ , and  $\mathbf{X}$  be a matrix, consisting of rows  $\{(M'_x(x_i, y_j)y_j - M'_y(x_i, y_j)x_i, M'_x(x_i, y_j), M'_y(x_i, y_j)), (x_i, y_j) \in \bar{D}_M\}$ , arranged in the same order as that for  $\mathbf{Y}$ . Then, by minimizing  $Q(\phi, \Delta x, \Delta y)$  in (4), we get

$$\hat{\boldsymbol{\theta}}_C = \begin{pmatrix} \hat{\phi} \\ \hat{\Delta x} \\ \hat{\Delta y} \end{pmatrix} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}. \tag{5}$$

To use the estimators in (5), we need to (i) estimate  $M'_x(x, y)$  and  $M'_y(x, y)$  for a given continuity point  $(x, y)$ , and (ii) obtain an estimator of  $\bar{D}_M$ . To estimate  $M'_x(x, y)$  and  $M'_y(x, y)$ , we suggest using the local linear kernel estimators defined by

$$\begin{aligned}
\widehat{M}'_x(x, y) &= \frac{\sum_{(x_i, y_j) \in \bar{D}_M} (x_i - x) Z_M(x_i, y_j) K_h(x_i - x, y_j - y)}{\sum_{(x_i, y_j) \in \bar{D}_M} (x_i - x)^2 K_h(x_i - x, y_j - y)}, \\
\widehat{M}'_y(x, y) &= \frac{\sum_{(x_i, y_j) \in \bar{D}_M} (y_j - y) Z_M(x_i, y_j) K_h(x_i - x, y_j - y)}{\sum_{(x_i, y_j) \in \bar{D}_M} (y_j - y)^2 K_h(x_i - x, y_j - y)},
\end{aligned} \tag{6}$$

where  $K_h(x, y) = K(x/h, y/h)$ ,  $K$  is a bivariate density kernel function with unit circular support, and  $h > 0$  is a bandwidth parameter. To estimate  $\bar{D}_M$ , we adopt the edge detection

method proposed recently by Kang and Qiu (2014) for detecting all pixels at which  $M(x, y)$  has jumps (i.e., edge pixels). By that method, at a given pixel we consider a circular neighborhood. Then, the neighborhood is divided into two halves along a direction perpendicular to the estimated gradient direction of the image intensity surface at the given pixel. The difference between two weighted averages of the image intensities in the two halves of the neighborhood is then used as the edge detection criterion. The given pixel is detected as an edge pixel if this criterion is larger than a predefined threshold value. Then, the estimator of  $\bar{D}_M$  is defined to be the set of all pixels after the detected edge pixels are excluded.

From (5), it can be seen that the estimator  $\hat{\boldsymbol{\theta}}_C$  is not well defined if the matrix  $\mathbf{X}^T \mathbf{X}$  is singular (i.e.,  $|\mathbf{X}^T \mathbf{X}| = 0$ ). For simplicity of notation, let  $M'_x(x_i, y_j) = a_{ij}$  and  $M'_y(x_i, y_j) = b_{ij}$ . Then

$$\mathbf{X}^T \mathbf{X} = \begin{pmatrix} \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)^2, & \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)a_{ij}, & \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)b_{ij} \\ \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)a_{ij}, & \sum_{(x_i, y_j) \in \bar{D}_M} a_{ij}^2, & \sum_{(x_i, y_j) \in \bar{D}_M} a_{ij}b_{ij} \\ \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)b_{ij}, & \sum_{(x_i, y_j) \in \bar{D}_M} a_{ij}b_{ij}, & \sum_{(x_i, y_j) \in \bar{D}_M} b_{ij}^2 \end{pmatrix}.$$

If  $|\mathbf{X}^T \mathbf{X}| = 0$ , then one of the following equations must hold:

$$\begin{aligned} \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)^2 \sum_{(x_i, y_j) \in \bar{D}_M} a_{ij}^2 &= \left( \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)a_{ij} \right)^2, \\ \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)^2 \sum_{(x_i, y_j) \in \bar{D}_M} b_{ij}^2 &= \left( \sum_{(x_i, y_j) \in \bar{D}_M} (a_{ij}y_j - b_{ij}x_i)b_{ij} \right)^2, \\ \sum_{(x_i, y_j) \in \bar{D}_M} a_{ij}^2 \sum_{(x_i, y_j) \in \bar{D}_M} b_{ij}^2 &= \left( \sum_{(x_i, y_j) \in \bar{D}_M} a_{ij}b_{ij} \right)^2. \end{aligned}$$

By the Cauchy inequality, the above three equations are equivalent to: for any  $(x_i, y_j) \in \bar{D}_M$ ,

$$a_{ij}y_j - b_{ij}x_i = \kappa_1 a_{ij},$$

$$a_{ij}y_j - b_{ij}x_i = \kappa_2 b_{ij},$$

$$a_{ij} = \kappa_3 b_{ij},$$

where  $\kappa$ 's are constants. By this result and the fact that the area of the edge curves is 0,

$M(x, y)$  must have one of the following forms in the entire design space:

$$\begin{aligned} M(x, y) &= \psi(x^2 + (y - \kappa_1)^2), \\ M(x, y) &= \psi((x - \kappa_2)^2 + y^2), \\ M(x, y) &= \psi(\kappa_3 x + y), \end{aligned} \tag{7}$$

where  $\psi$  is a univariate function. In such cases,  $M(x, y)$  is actually a degenerate function in the design space. If one of the first two equations is valid (i.e.,  $M(x, y)$  is circularly symmetric), then  $\phi$  in  $\theta$  cannot be estimated. If the last equation is valid (i.e.,  $M(x, y)$  is a univariate function of  $\kappa_3 x + y$ ), then one or both  $\Delta x$  and  $\Delta y$  cannot be estimated. In such cases, we can just calculate the estimable parameters, set the unestimable parameters to be zero, and make the image registration accordingly. In the appendix, we show that  $\widehat{\theta}_C$  is statistically consistent under some regularity conditions.

## 2.2 Edge-based image registration

To estimate the parameters  $\theta = (\phi, \Delta x, \Delta y)^T$  in the rigid-body transformation (3), we can also use the detected edge points of the two observed images  $Z_R(x, y)$  and  $Z_M(x, y)$ . The sets of these detected edge points are denoted as  $D_R$  and  $D_M$ , respectively. To this end, we first need to build a 1-1 correspondence, called *feature matching* in the image processing literature (cf., Qiu and Xing 2013b), between the detected edge points in  $D_R$  and  $D_M$ . In this paper, we use the mean squared difference (MSD) metric for this purpose. Without loss of generality, assume that  $|D_R| \leq |D_M|$ , where  $|D_R|$  denotes the number of pixels in  $D_R$ . Otherwise, we can switch the positions of  $D_R$  and  $D_M$ . For any detected edge point  $(x, y) \in D_R$ , its matched edge point in  $D_M$ , denoted as  $(x^*, y^*)$ , is defined by

$$(x^*, y^*) = \arg \min_{(x', y') \in D_M, \|(x', y') - (x, y)\| \leq r_n} \frac{1}{\tilde{N}} \sum_{s^2 + t^2 \leq d_n^2} \left( Z_R(x + s, y + t) - Z_M(x' + s, y' + t) \right)^2,$$

where  $d_n$  and  $r_n$  are two radius parameters and  $\tilde{N}$  is the total number of pixels in the circular neighborhood  $O(x, y; d_n)$ .

From (3), when  $\phi$  is small, we have

$$T_1(x, y) \approx x + y\phi + \Delta x,$$

$$T_2(x, y) \approx -x\phi + y + \Delta y.$$

Thus, to estimate  $\boldsymbol{\theta} = (\phi, \Delta x, \Delta y)^T$ , it is natural to consider the minimizer of the following objective function

$$Q_E(\phi, \Delta x, \Delta y) = \sum_{(x_i, y_j) \in D_R} (x_i^* - x_i - y_j\phi - \Delta x)^2 + (y_j^* - y_j + x_i\phi - \Delta y)^2. \quad (8)$$

The resulting estimators are then

$$\hat{\boldsymbol{\theta}}_E = \begin{pmatrix} \hat{\phi} \\ \hat{\Delta x} \\ \hat{\Delta y} \end{pmatrix} = \mathbf{A}^{-1}\mathbf{B}, \quad (9)$$

where

$$\mathbf{A} = \begin{pmatrix} \sum_{(x_i, y_j) \in D_R} (x_i^2 + y_j^2), & \sum_{(x_i, y_j) \in D_R} y_j, & -\sum_{(x_i, y_j) \in D_R} x_i \\ \sum_{(x_i, y_j) \in D_R} y_j, & |D_R|, & 0 \\ -\sum_{(x_i, y_j) \in D_R} x_i, & 0, & |D_R| \end{pmatrix},$$

$$\mathbf{B} = \begin{pmatrix} \sum_{(x_i, y_j) \in D_R} (x_i^* - x_i)y_j - \sum_{(x_i, y_j) \in D_R} (y_j^* - y_j)x_i \\ \sum_{(x_i, y_j) \in D_R} (x_i^* - x_i) \\ \sum_{(x_i, y_j) \in D_R} (y_j^* - y_j) \end{pmatrix}.$$

It is not difficult to check that  $\mathbf{A}^{-1}$  always exists when  $|D_R| > 0$ . For the estimator  $\hat{\boldsymbol{\theta}}_E$ , we show in the appendix that it is statistically consistent under some regularity conditions.

### 3 Hypothesis Tests for Detecting Difference Between Two Images

After the observed reference image  $Z_R(x, y)$  and the observed moved image  $Z_M(x, y)$  are properly registered, we can test whether the two geometrically matched images are the same or not. More specifically, we are interested in testing the hypotheses

$H_0$  : There is a rigid-body transformation  $\mathbf{T}(x, y)$  such that  $M(\mathbf{T}(x, y)) \equiv R(x, y)$

versus

$H_1$  : No rigid-body transformation exists such that  $H_0$  is true.

However, images have complicated structures. Roughly speaking, a general image consists of edges and continuity regions that the image intensity function has jumps and is continuous, respectively. Because the edge pixels and the pixels in the continuity regions are very different in nature, they can be considered separately in the above hypothesis testing problem. In the three subsections below, we will discuss three hypothesis testing procedures constructed based on the continuity regions of the images, their detected edges, and the combination of the continuity regions and detected edges, respectively.

### 3.1 Test based on the continuity regions of the images

To test whether  $M(\mathbf{T}(x, y))$  and  $R(x, y)$  are the same, we can first investigate whether their continuity regions are the same. Remember that the sets of detected edges in the two observed reference and moved images are  $D_R$  and  $D_M$ , respectively. Also, the edges are detected by the local smoothing method in Kang and Qiu (2014), and the detected edges would be distributed mainly in a local band of the true edges. So, instead of deducting  $D_R \cup D_M$  from the design space  $\Omega$ , we consider

$$G = \Omega \setminus \left( D_R(h_G) \cup D_M(h_G) \right),$$

where  $D_R(h_G)$  denotes the set of pixels whose Euclidean distance to  $D_R$  is less than or equal to  $h_G$ , and  $h_G > 0$  is a bandwidth. Then, it is natural to consider the statistic

$$U' = \sum_{(x_i, y_j) \in G} \left( Z_R(x_i, y_j) - Z_M(\hat{\mathbf{T}}(x_i, y_j)) \right)^2,$$

If  $H_0$  is true and  $\hat{\mathbf{T}}(x, y)$  is a perfect estimator of  $\mathbf{T}(x, y)$ , then the value of this statistic would be small. Otherwise, its value would be large. Therefore,  $U'$  can be used for testing  $H_0$  and  $H_1$ . Because  $U'$  has a quadratic form, its asymptotic distribution would be close to a  $\chi^2$  distribution, which is often skewed. Thus, it is natural to consider its following standardized

version:

$$U = \frac{1}{\sqrt{2|G|}} \sum_{(x_i, y_j) \in G} \left( (Z_R(x_i, y_j) - Z_M(\hat{\mathbf{T}}(x_i, y_j)))^2 / (2\hat{\sigma}^2) - 1.0 \right),$$

where  $|G|$  denotes the number of pixels in  $G$  and  $\hat{\sigma}^2$  is the variance estimator in Kang and Qiu (2014). Because its true null distribution is unknown, its critical value will be determined by a numerical approach described in Section 3.3 below. In the previous section, we proposed two different estimators of  $\mathbf{T}(x, y)$ , using  $\hat{\boldsymbol{\theta}}_C$  and  $\hat{\boldsymbol{\theta}}_E$ , respectively. The corresponding test statistics are denoted as  $U_{CC}$  and  $U_{EC}$ , where the first subscript identifies the way for image registration and the second subscript denotes the fact that this testing procedure is based on the continuity regions of the images.

### 3.2 Test based on the detected edges of the images

We can also use the detected edge pixels of the observed reference and moved images for testing whether the two geometrically matched images are the same. To this purpose, for any pixel  $(x_i, y_j) \in D_R$ , let the matched pixel in  $D_M$  be  $(\hat{x}_i, \hat{y}_j) = \hat{\mathbf{T}}(x_i, y_j)$ , as discussed in Section 2.2. Then, we suggest the following test statistic:

$$U_E = \frac{1}{\sqrt{2|D_R|}} \sum_{(x_i, y_j) \in D_R} \left\{ (Z_R(x_i, y_j) - Z_M(\hat{x}_i, \hat{y}_j))^2 / (2\hat{\sigma}^2) - 1 \right\} \times I \left( (Z_R(x_i, y_j) - Z_M(\hat{x}_i, \hat{y}_j))^2 < \gamma^2 \hat{\sigma}^2 \right),$$

where  $\gamma > 0$  is a constant. In the above expression, the indicator  $I((Z_R(x_i, y_j) - Z_M(\hat{x}_i, \hat{y}_j))^2 < \gamma^2 \hat{\sigma}^2)$  is used to exclude cases when  $(x_i, y_j)$  is on one side of an edge curve in  $Z_R(x, y)$  but  $(\hat{x}_i, \hat{y}_j)$  is on another side of the same edge curve in  $Z_M(x, y)$ , which can happen because the detected edge pixels are usually scattered around the true edge curves.

### 3.3 Determination of the critical values of the tests

In the test statistics  $U_{CC}$ ,  $U_{EC}$  and  $U_E$  discussed above, besides the random errors in the observed images, the estimated transformation  $\hat{\mathbf{T}}(x, y)$ , the estimated variance  $\hat{\sigma}^2$ , and the detected edges are all involved. So, their actual distributions would be quite complicated,

although their asymptotic distributions might all be standard normal. In this part, we propose the following bootstrap procedure to calculate their critical values.

1. We apply the local piecewisely liner kernel smoothing procedure by Qiu (2004) to the two observed images, and obtain edge-preserved estimators of  $R(x, y)$  and  $M(x, y)$ , denoted as  $\widehat{R}(x, y)$  and  $\widehat{M}(x, y)$ , respectively. The corresponding residuals are denoted as  $\widehat{\varepsilon}_R(x, y)$  and  $\widehat{\varepsilon}_M(x, y)$ .
2. Generated new reference and moved images by

$$\begin{aligned} Z_R^*(x_i, y_j) &= \widehat{R}(x_i, y_j) + \widehat{\varepsilon}_R^*(x_i, y_j), \text{ for } i, j = 1, 2, \dots, n, \\ Z_M^*(x_i, y_j) &= \widehat{M}(x_i, y_j) + \widehat{\varepsilon}_M^*(x_i, y_j), \end{aligned}$$

where  $\{\widehat{\varepsilon}_R^*(x_i, y_j)\}$  and  $\{\widehat{\varepsilon}_M^*(x_i, y_j)\}$  are bootstrap samples obtained from  $\{\widehat{\varepsilon}_R(x_i, y_j), i, j = 1, 2, \dots, n\}$  and  $\{\widehat{\varepsilon}_M(x_i, y_j), i, j = 1, 2, \dots, n\}$ , respectively. Because the residuals around the true edges could be large, in the above resampling procedure, we suggest replacing  $\widehat{\varepsilon}_R(x_i, y_j)$  by

$$\widetilde{\varepsilon}_R(x_i, y_j) = \begin{cases} \widehat{\varepsilon}_R(x_i, y_j), & \text{if } |\widehat{\varepsilon}_R(x_i, y_j)| < \widetilde{\gamma}\widehat{\sigma}, \\ \text{a random number from } N(0, \widehat{\sigma}^2), & \text{otherwise,} \end{cases}$$

where  $\widetilde{\gamma}$  is a constant chosen in the interval  $[3, 5]$ . The same modification is also made for  $\widehat{\varepsilon}_M(x_i, y_j)$ .

3. Calculate the values of the three test statistics using the two generated images  $Z_R^*(x, y)$  and  $Z_M^*(x, y)$ .
4. Repeat Steps 2 and 3 by  $B$  times, and the bootstrap critical values are the empirical  $(1 - \alpha)$ -th quantiles of the  $B$  sets of values of the three test statistics.

### 3.4 Combination tests

In the above three parts, we have proposed three testing procedures: the ones using  $U_{CC}$  and  $U_{EC}$  are based on the continuity regions of the two images  $R(x, y)$  and  $M(x, y)$ , and the one using  $U_E$  is based on the detected edges. If  $H_0$  is true, then both the continuity regions and

the edges of the two images should be the same. So, it is natural to consider the following two combination tests:

$$U_{E,CC} : \text{reject } H_0 \text{ if } (U_E > z_{\alpha/2}^E) \text{ or } (U_{CC} > z_{\alpha/2}^{CC}),$$

$$U_{E,EC} : \text{reject } H_0 \text{ if } (U_E > z_{\alpha/2}^E) \text{ or } (U_{EC} > z_{\alpha/2}^{EC}),$$

where  $z_{\alpha/2}^E$ ,  $z_{\alpha/2}^{EC}$  and  $z_{\alpha/2}^{CC}$  are the  $(1 - \alpha/2)$ -th quantiles of  $U_E, U_{EC}, U_{CC}$ , respectively, under  $H_0$ .

## 4 Practical Guidelines on Parameter Selection

In our proposed methods described above, there are a number of parameters involved. They should be chosen properly in advance to have a good performance of the proposed methods. All these parameters are the so-called smoothing parameters (Qiu 2005, Chapter 2). Generally speaking, their values should be chosen relatively large when the noise level is high, and relatively small when the noise level is low. However, it is difficult (or even impossible) to provide formulas for determining their values in all different cases, because besides the noise level their values also depend on the complexity of the related image structures, including the edge structure, shape and magnitude of the image intensity surfaces, and so forth. With the proposed methods, we have performed many simulation studies, including those presented in the next section. Based on our numerical experience, we provide the following practical guidelines for proper selection of the parameters. These guidelines provide ranges for the related parameters, and we find that our proposed methods perform reasonably well in all cases that we have considered when their parameters are chosen in the recommended ranges.

**On choosing  $h$  in (6):** The bandwidth value  $h$  in (6) can be determined properly by a bootstrap procedure. However, because it is only used for estimating  $M'_x(x, y)$  and  $M'_y(x, y)$  that are used in the middle of estimating  $\hat{\theta}$ , selection of its value is less important and we found that any number in  $[0.1, 0.3]$  would produce similarly good results.

**On parameter selection in edge detection:** The parameters used in the edge detection

procedure by Kang and Qiu (2014) that is mentioned in Sections 2 and 3 are chosen in the same way as suggested in Kang and Qiu (2014).

**On choosing  $r_n$  and  $d_n$  in the MSD procedure:** In the MSD procedure described in Section 2.2, there are two parameters  $r_n$  and  $d_n$ . The size of  $r_n$  is related directly to  $\delta_n$ : a large  $r_n$  should be used if  $\delta_n$  is large. In many different cases, we find that it is good enough to choose  $r_n \in [0.05, 0.25]$ . For  $d_n$ , we can choose it to be  $\min\{r_n, s_n\}$ , where  $s_n \in [0.01, 0.1]$ . In the simulation studies presented in the next section, we choose  $r_n = 0.25h$  and  $d_n = 0.25h$ .

**On choosing  $h^*$  used in  $D_R(h^*)$ :** Overall,  $h_*$  should be chosen small. Based on our numerical experience, we suggest choosing  $h_* \in [0.01, 0.05]$ . In the simulation studies in the next section, we choose  $h_* = 0.25h$ .

**On choosing  $\gamma$  used in  $U_E$ :** The constant  $\gamma$  is used for avoiding pixels located on different sides of an edge curve being included simultaneously in the definition of  $U_E$ . We find that results are reasonably good when  $\gamma \in [3, 5]$ .

**On choosing  $\tilde{\gamma}$  in the bootstrap procedure in Section 3.3:** Similar to  $\gamma$ ,  $\tilde{\gamma}$  is used mainly for excluding some unusually large residuals caused by edges. We find that results are reasonably good when  $\tilde{\gamma} \in [3, 5]$ .

## 5 Simulation

In this section, we present some numerical examples about the proposed methods for testing difference between two images. Four reference images with the following image intensity

functions are considered:

$$\begin{aligned}
f_1(x, y) &= \begin{cases} -4(x - 0.5)^2 - 4y^2 + 1, & \text{if } x^2 + y > 0.5, \\ -4(x - 0.5)^2 - 4y^2, & \text{otherwise;} \end{cases} \\
f_2(x, y) &= \begin{cases} -4(x - 0.5)^2 - 4(y - 0.5)^2 + 0.8, & \text{if } x < 0.5, 0.5x + y < 0.75 \\ -4(x - 0.5)^2 - 4(y - 0.5)^2, & \text{otherwise;} \end{cases} \\
f_3(x, y) &= \begin{cases} -4(x - 0.5)^2 - 4(y - 0.5)^2 + 1, & \text{if } x < 0.5, 0.5x + y < 0.75 \\ -4(x - 0.5)^2 - 4(y - 0.5)^2, & \text{otherwise.} \end{cases} \\
f_4(x, y) &= \begin{cases} f_3(x, y) + 1, & \text{if } 0.6 < x < 0.8, 0.6 < y < 0.8 \\ f_3(x, y), & \text{otherwise.} \end{cases}
\end{aligned}$$

The four true reference images are shown in Figure 1. From the above expressions and the images in Figure 1, we can see that  $f_1(x, y)$  is degenerate in the sense of (7). So, it is expected that  $\widehat{\theta}_C$  does not perform well.  $f_2(x, y)$  and  $f_3(x, y)$  are both non-degenerate, and  $f_3(x, y)$  has larger jumps at the edge curve than  $f_2(x, y)$ .  $f_4(x, y)$  has three separate continuity regions while the other three reference images have two continuity regions each. The moved images are generated from the reference images, using  $\theta = (\alpha, \Delta x, \Delta y)^T = (0.01, 0.015, 0.015)^T$ . The image registration procedures described in Section 2 are then applied, using the Epanechnikov kernel function  $K(x, y) = \frac{144}{121}(1 - x^2)(1 - y^2)I((x, y) \in [-0.5, 0.5] \times [-0.5, 0.5])$  and  $h = 0.16$  in (6). In the local kernel smoothing literature, the Epanechnikov kernel function is often used because of its good theoretical properties. The noise levels  $\sigma_R^2$  and  $\sigma_M^2$  are both fixed at 0.02. Two sample sizes  $n = 150$  and 200 are considered. Based on 1,000 replicated simulations, the results are shown in Table 1.

From Table 1, we can see that (i)  $\widehat{\theta}_C$  indeed does not perform well in cases with  $f_1(x, y)$ , (ii) results in cases with  $f_3(x, y)$  are better than those with  $f_2(x, y)$ , especially for  $\widehat{\theta}_E$ , (iii) results in cases with  $f_4(x, y)$  are better than those with  $f_3(x, y)$ , and (iv) results when  $n = 200$  are overall better than those when  $n = 150$ . All these results are intuitively reasonable. For  $f_3(x, y)$ , an observed reference image and an observed moved image when  $n = 200$  are shown in Figure 2(a)-(b). The recovered reference image  $Z_M(\widehat{\mathbf{T}}(x, y))$  using  $\widehat{\theta}_E$  and the one using  $\widehat{\theta}_C$  are shown in Figure 2(c)-(d), respectively. The difference image between the ones in Figure 2(a) and Figure 2(c) is shown in Figure 2(e), and the difference image between the ones in Figure 2(a) and Figure 2(d) is shown in Figure 2(f). We can see that the two difference images contain mainly noise, except the places around the edges due to the fact that the

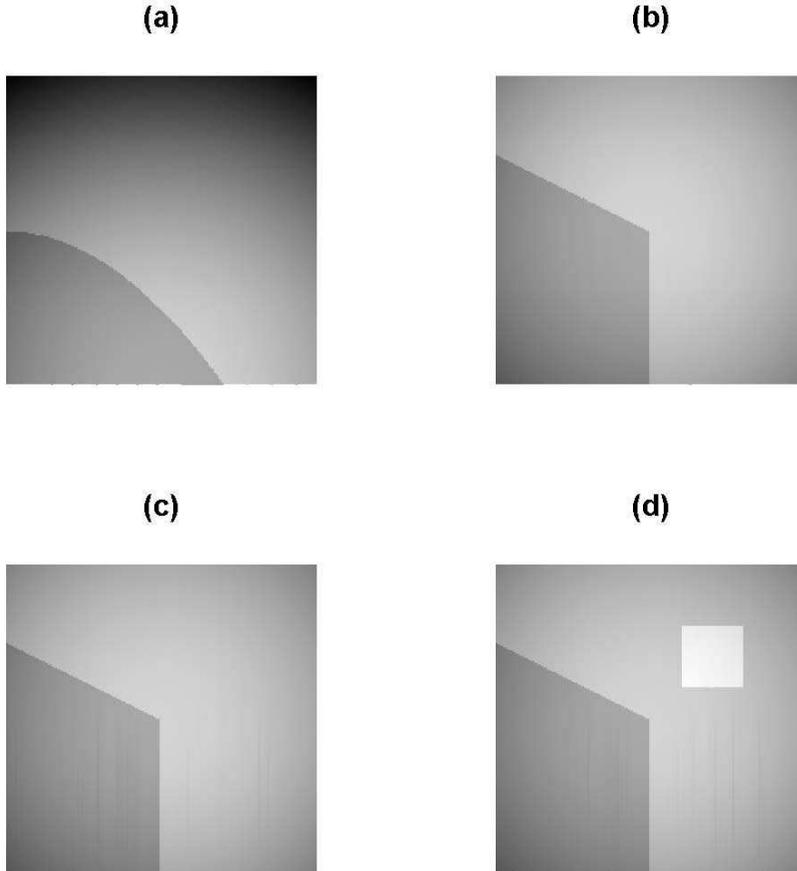


Figure 1: (a)-(c) Four true reference images  $f_1(x, y)$ ,  $f_2(x, y)$ ,  $f_3(x, y)$  and  $f_4(x, y)$ .

estimated geometric transformations are not exactly the true transformation. The results for  $f_4(x, y)$  are similar to those of  $f_3(x, y)$ . Both  $\hat{\theta}_C$  and  $\hat{\theta}_E$  performs well in this case.

Next, we investigate the numerical performance of the testing procedures discussed in Section 3. In addition, we also consider the following two alternative methods: i) hypothesis tests using two entire images without image registration (denoted as NAIVE), and ii) hypothesis tests using two entire images after the images are registered by the popular method FAST original suggested in Rosten and Drummond (2005) (denoted as FAST). The FAST algorithm tries to register the two images using properly detected features (e.g., corners and edges). The critical values of these two tests are both chosen to be 1.96, corresponding to the type-I error probability of 0.05. Next, let us focus on  $f_3(x, y)$  and consider the following

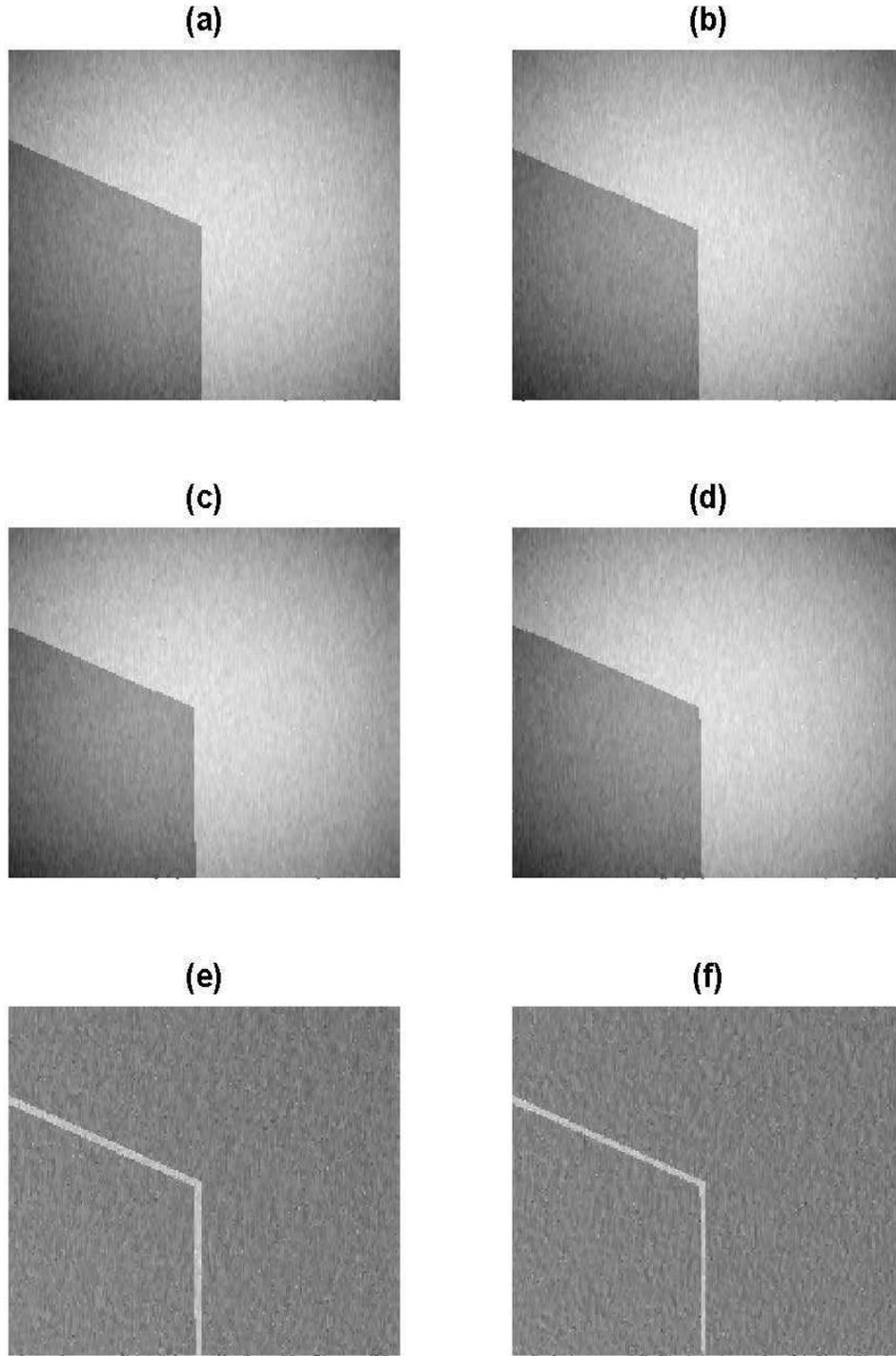


Figure 2: Image registration results in the example with  $f_3(x, y)$  when  $n = 200$  and  $\sigma^2 = 0.02$ . (a) Observed referenced image; (b) observed moved image; (c) recovered reference image  $Z_M(\hat{\mathbf{T}}(x, y))$  using  $\hat{\theta}_E$ ; (d) recovered reference image  $Z_M(\hat{\mathbf{T}}(x, y))$  using  $\hat{\theta}_C$ ; (e) difference between (a) and (c); (f) difference between (a) and (d).

Table 1: Estimated parameters in the geometric transformation and their standard errors (in parentheses).

			$\phi$	$\Delta x$	$\Delta y$
$n = 150$	$f_1(x, y)$	$\widehat{\theta}_E$	0.015(0.005)	0.013(0.001)	0.016(0.002)
		$\widehat{\theta}_C$	-0.009(0.002)	0.013(0.001)	0.008(0.001)
	$f_2(x, y)$	$\widehat{\theta}_E$	0.015(0.006)	0.013(0.001)	0.015(0.001)
		$\widehat{\theta}_C$	0.011(0.003)	0.012(0.001)	0.019(0.001)
	$f_3(x, y)$	$\widehat{\theta}_E$	0.015(0.006)	0.014(0.002)	0.015(0.001)
		$\widehat{\theta}_C$	0.010(0.002)	0.012(0.001)	0.018(0.001)
	$f_4(x, y)$	$\widehat{\theta}_E$	0.011(0.005)	0.014(0.002)	0.015(0.001)
		$\widehat{\theta}_C$	0.010(0.002)	0.013(0.001)	0.015(0.001)
$n = 200$	$f_1(x, y)$	$\widehat{\theta}_E$	0.011(0.004)	0.015(0.002)	0.014(0.002)
		$\widehat{\theta}_C$	-0.011(0.002)	0.011(0.001)	0.005(0.001)
	$f_2(x, y)$	$\widehat{\theta}_E$	0.010(0.005)	0.014(0.002)	0.015(0.002)
		$\widehat{\theta}_C$	0.015(0.002)	0.012(0.001)	0.015(0.002)
	$f_3(x, y)$	$\widehat{\theta}_E$	0.010(0.004)	0.015(0.002)	0.015(0.001)
		$\widehat{\theta}_C$	0.011(0.002)	0.014(0.001)	0.014(0.001)
	$f_4(x, y)$	$\widehat{\theta}_E$	0.009(0.005)	0.015(0.002)	0.015(0.001)
		$\widehat{\theta}_C$	0.011(0.002)	0.014(0.001)	0.015(0.001)

three alternative images:

$$g_1(x, y) = \begin{cases} f_3(x, y) + 0.5, & \text{if } |x - 0.8| + |y - 0.2| < 0.05, \\ f_3(x, y), & \text{otherwise;} \end{cases}$$

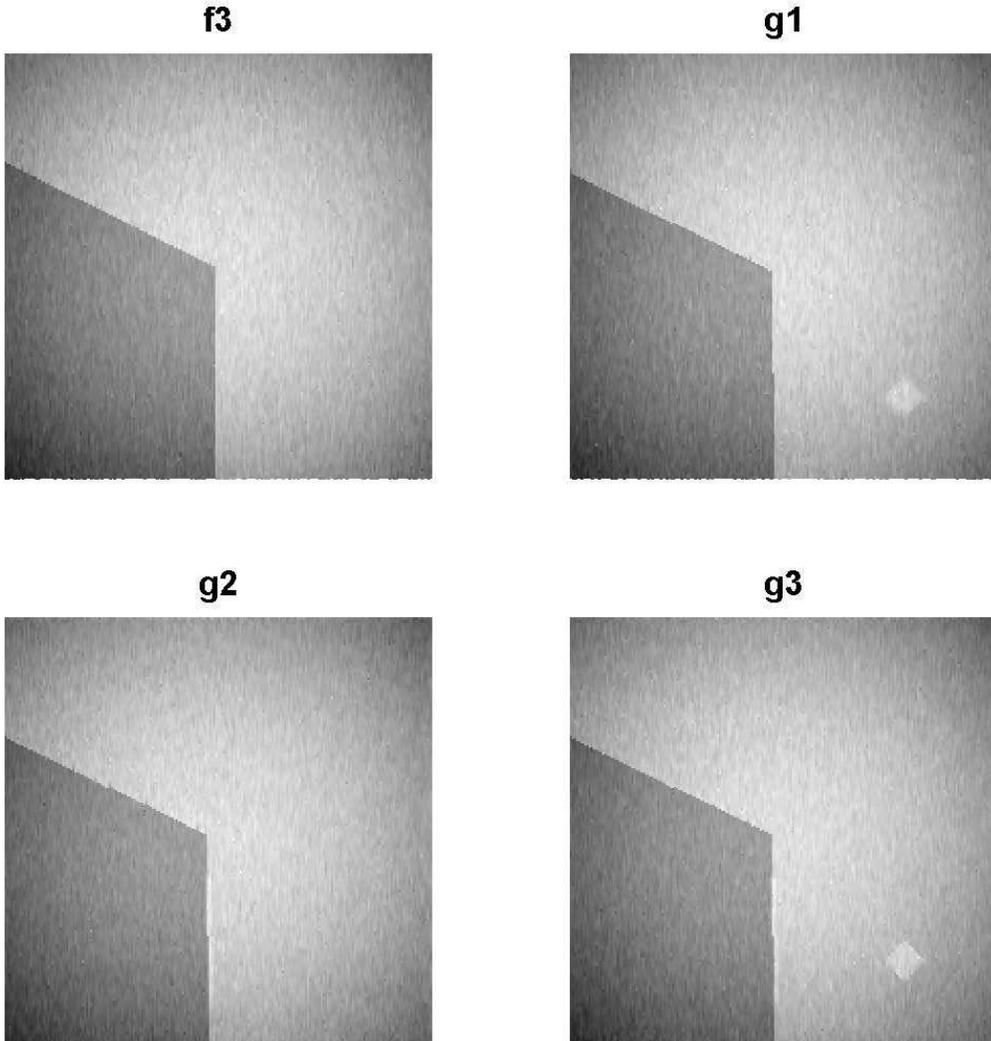
$$g_2(x, y) = \begin{cases} f_3(x, y) + 0.3, & \text{if } |x - 0.5| < 0.006, 0.1 < y < 0.4, \\ f_3(x, y), & \text{otherwise;} \end{cases}$$

$$g_3(x, y) = \begin{cases} g_1(x, y) + 0.3, & \text{if } |x - 0.5| < 0.006, 0.1 < y < 0.4, \\ g_1(x, y), & \text{otherwise.} \end{cases}$$

The four images  $f_3(x, y)$ ,  $g_1(x, y)$ ,  $g_2(x, y)$ , and  $g_3(x, y)$  are shown in Figure 3. From the images in the figure, it can be seen that  $g_1(x, y)$  is different from  $f_3(x, y)$  only in a small diamond,  $g_2(x, y)$  is different from  $f_3(x, y)$  around the edge, and  $g_3(x, y)$  is different from  $f_3(x, y)$  at both places. We consider the following four scenarios: the first scenario is that the reference image is  $f_3(x, y)$  and the moved image is also  $f_3(x, y)$ , and the remaining three scenarios are that the reference image is  $f_3(x, y)$  and the moved image is one of  $g_1(x, y)$ ,  $g_2(x, y)$ , and  $g_3(x, y)$ . So, in scenario (i),  $H_0$  is true, and  $H_0$  is false in the other three

scenarios. For the tests  $U_E$ ,  $U_{EC}$ ,  $U_{CC}$ ,  $U_{E,EC}$ , and  $U_{E,CC}$ , their 0.95 critical values are determined by the bootstrap procedure discussed in Section 3.3 with  $B = 400$  and  $\tilde{\gamma} = 3.5$ . In  $U_E$ ,  $\gamma$  is chosen 3.5 as well. In all scenarios, the true rigid-body transformation has the parameters  $\boldsymbol{\theta} = (\alpha, \Delta x, \Delta y)^T = (0.01, 0.015, 0.015)^T$ . The results based on 1000 replicated simulations are shown in Table 2. From the table, we can have the following conclusions about the five tests  $U_E$ ,  $U_{EC}$ ,  $U_{CC}$ ,  $U_{E,EC}$ , and  $U_{E,CC}$ . (i) The empirical sizes of all these tests are quite close to the nominal size of 0.05. (ii) The edge-based test  $U_E$  is not powerful when the alternative image is  $g_1(x, y)$ , because the difference between  $g_1(x, y)$  and  $f_3(x, y)$  is in a continuity region (i.e., a small diamond). On the other hand, the two continuity-region-based tests  $U_{EC}$  and  $U_{CC}$  are quite powerful in this case. (iii)  $U_{EC}$  and  $U_{CC}$  are not powerful when the alternative image is  $g_2(x, y)$ , because the difference between  $g_2(x, y)$  and  $f_3(x, y)$  is on an edge line only. On the other hand,  $U_E$  is quite powerful in this case. (iv) When the alternative image is  $g_3(x, y)$  which is different from  $f_3(x, y)$  in a continuity region and around an edge line, the two combination tests  $U_{E,EC}$  and  $U_{E,CC}$  are more powerful than the other tests. (iv) Powers of the tests are generally improved when  $n$  increases from 150 to 200. By the way, to obtain each number in Table 2 for the five tests  $U_E$ ,  $U_{EC}$ ,  $U_{CC}$ ,  $U_{E,EC}$ , and  $U_{E,CC}$ , it spends about 400 seconds CPU time when  $n = 150$  and about 1600 seconds CPU time when  $n = 200$ , on a personal computer of Intel i7=4700 with 2.40 GHz.

Regarding the two alternative methods NAIVE and FAST, we can see that NAIVE always rejects the null hypothesis even when it is true. That is because it did not register the two images properly beforehand, and thus the geometric difference between the two images is detected by NAIVE all the time. Therefore, this method should be avoided in practice if image registration is substantial to a specific application problem. The method FAST performs a little bit better, but its actual size is still much larger than the nominal size 0.05.



1'

Figure 3: Four images  $f_3(x, y)$ ,  $g_1(x, y)$ ,  $g_2(x, y)$ , and  $g_3(x, y)$  used in evaluating the numerical performance of the testing procedures.

Table 2: Empirical sizes and powers of the five testing procedures discussed in Section 3 in cases when the reference image is  $f_3(x, y)$  and the alternative image is  $g_1(x, y)$ ,  $g_2(x, y)$ , or  $g_3(x, y)$ .

		$U_E$	$U_{EC}$	$U_{CC}$	$U_{E,EC}$	$U_{E,CC}$	NAIVE	FAST
$n = 150$	$H_0$	0.058	0.057	0.043	0.052	0.051	1.00	0.83
	$g_1$	0.061	0.780	0.560	0.740	0.440	1.00	1.00
	$g_2$	0.600	0.061	0.024	0.540	0.540	1.00	1.00
	$g_3$	0.580	0.690	0.350	0.820	0.670	1.00	1.00
$n = 200$	$H_0$	0.042	0.054	0.063	0.040	0.041	1.00	0.85
	$g_1$	0.034	0.980	0.920	0.920	0.870	1.00	1.00
	$g_2$	1.000	0.083	0.071	1.000	1.000	1.00	1.00
	$g_3$	1.000	0.890	0.770	1.000	1.000	1.00	1.00

Next, we consider the 3-region image  $f_4(x, y)$  and the following three alternative images:

$$\begin{aligned}
 h_1(x, y) &= \begin{cases} f_4(x, y) + 0.5, & \text{if } |x - 0.8| + |y - 0.2| < 0.05, \\ f_4(x, y), & \text{otherwise;} \end{cases} \\
 h_2(x, y) &= \begin{cases} f_4(x, y) + 0.3, & \text{if } |x - 0.5| < 0.006, 0.1 < y < 0.4, \\ f_4(x, y), & \text{otherwise;} \end{cases} \\
 h_3(x, y) &= \begin{cases} h_1(x, y) + 0.3, & \text{if } |x - 0.5| < 0.006, 0.1 < y < 0.4, \\ h_1(x, y), & \text{otherwise.} \end{cases}
 \end{aligned}$$

The other settings are the same as those in the previous example. The images of  $f_4(x, y)$ ,  $h_1(x, y)$ ,  $h_2(x, y)$  and  $h_3(x, y)$  are shown in Figure 4. The simulation results about the 7 testing procedures are presented in Table 3. It can be seen that similar conclusions to those from Table 2 can be made from this table.

Our proposed testing procedures are for cases when image observations are independent. In some applications, image observations could be spatially correlated. In such cases, our proposed testing procedures described in Section 3 could still be used, except that a block bootstrap procedure (Hall et al. 1995, Lahiri 1999) should replace the regular bootstrap procedure in the algorithm discussed in Section 3.3 to accommodate the spatial correlation. By a block bootstrap procedure, all pixels in an image are divided into  $M \times M$  blocks, and then we randomly select blocks of the estimated errors  $\{\widehat{\varepsilon}_R(x_i, y_j), i, j = 1, 2, \dots, n\}$  and  $\{\widehat{\varepsilon}_M(x_i, y_j), i, j = 1, 2, \dots, n\}$  with replacement when defining bootstrap samples in Step 2 of

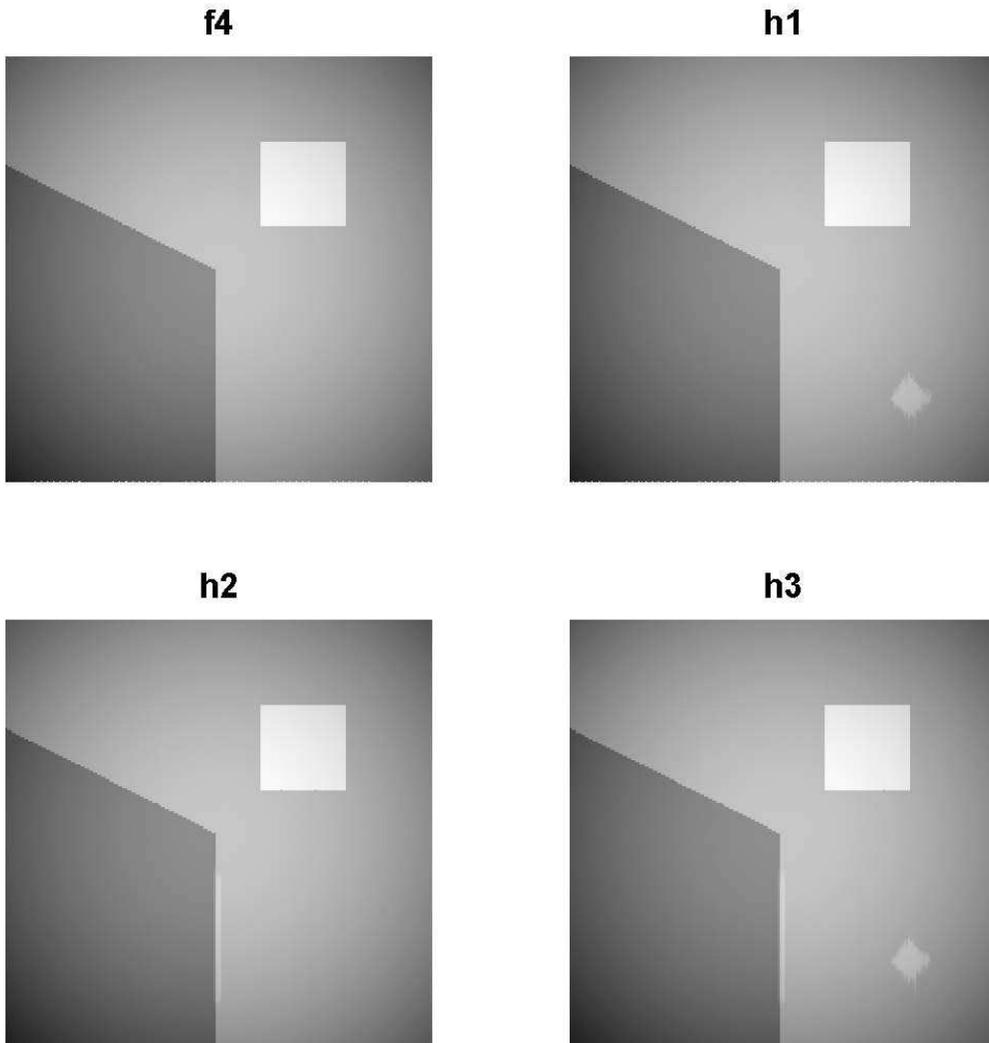


Figure 4: Four images  $f_4(x, y)$ ,  $h_1(x, y)$ ,  $h_2(x, y)$ , and  $h_3(x, y)$  used in evaluating the numerical performance of the testing procedures.

Table 3: Empirical sizes and powers of the five testing procedures discussed in Section 3 in cases when the reference image is  $f_4(x, y)$  and the alternative image is  $h_1(x, y)$ ,  $h_2(x, y)$ , or  $h_3(x, y)$ .

		$U_E$	$U_{EC}$	$U_{CC}$	$U_{E,EC}$	$U_{E,CC}$	NAIVE	FAST
$n = 150$	$H_0$	0.063	0.067	0.047	0.062	0.049	1.00	1.00
	$h_1$	0.071	0.810	0.430	0.730	0.410	1.00	1.00
	$h_2$	0.510	0.110	0.041	0.520	0.440	1.00	1.00
	$h_3$	0.490	0.740	0.500	0.750	0.600	1.00	1.00
$n = 200$	$H_0$	0.052	0.044	0.043	0.045	0.041	1.00	1.00
	$h_1$	0.064	0.950	0.990	0.900	0.870	1.00	1.00
	$h_2$	0.930	0.103	0.051	0.950	1.000	1.00	1.00
	$h_3$	0.930	0.960	0.890	1.000	1.000	1.00	1.00

the algorithm in Section 3.3 for calculating critical values of the tests. Also, for the tests  $U_{CC}$  and  $U_{EC}$  that compare the two images in continuity regions, the modification of the estimated errors (i.e.,  $\tilde{\varepsilon}_R(x_i, y_j)$ ) should be avoided to maintain spatial correlation in the estimated errors. In the next example, we consider the cases in Table 2, except that image observations are correlated here. More specifically, we assume that  $\varepsilon_R(x_i, y_j) = \varepsilon_1(x_i) + \varepsilon_2(y_j)$ , where  $\{\varepsilon_1(x_i)\}$  and  $\{\varepsilon_2(y_j)\}$  are independent, and both follow the AR(1) time series model with mean 0, correlation parameter  $\rho$ , and the white noise variance  $\sigma^2$ . When  $n = 150$ ,  $M = 15$ ,  $\sigma^2 = 0.02$ , and  $\rho = 0.05, 0.1$  or  $0.2$ , the calculated sizes and powers of the five suggested tests are presented in Table 4. From the table, we can see that when the spatial correlation is relatively weak (i.e.,  $\rho = 0.05$ ), the sizes of the five tests are all below or close to the nominal level 0.05, and the edge-based tests tests  $U_E$ ,  $U_{E,EC}$  and  $U_{E,CC}$  are quite powerful in cases when the alternative images are  $g_2(x, y)$  and  $g_3(x, y)$  which are quite different from the reference image  $f_3(x, y)$  at a major edge line. The tests based on continuity regions (i.e.,  $U_{EC}$  and  $U_{CC}$ ) do not have much power in detecting the difference between the reference image and the alternative images because such a difference is really small in this example. When the spatial correlation gets larger, it can be seen that the sizes of the tests also get larger, and some of them are larger than the nominal level 0.05. Therefore, it requires much future research to properly address cases when the spatial correlation is strong.

Table 4: Empirical sizes and powers of the five testing procedures in cases of Table 2 when  $n = 150$  and the image observations are correlated.

		$U_E$	$U_{EC}$	$U_{CC}$	$U_{E,EC}$	$U_{E,CC}$
$\rho = 0.05$	$H_0$	0.066	0.007	0.004	0.052	0.049
	$g_1$	0.032	0.044	0.071	0.032	0.052
	$g_2$	0.682	0.034	0.010	0.602	0.602
	$g_3$	0.530	0.040	0.047	0.410	0.409
$\rho = 0.1$	$H_0$	0.110	0.012	0.013	0.065	0.058
	$g_1$	0.034	0.110	0.092	0.073	0.057
	$g_2$	0.739	0.023	0.011	0.614	0.613
	$g_3$	0.490	0.072	0.053	0.367	0.356
$\rho = 0.2$	$H_0$	0.110	0.012	0.073	0.112	0.063
	$g_1$	0.036	0.350	0.257	0.212	0.164
	$g_2$	0.716	0.091	0.031	0.591	0.568
	$g_3$	0.440	0.302	0.191	0.502	0.442

## 6 Real Data Application

In a rolling process, a metal bar passes through one or more pairs of rolls to achieve certain thickness. The surface images of a rolling bar can be used for monitoring its quality, and they are often recorded by a high-speed camera. Each image has  $128 \times 512$  pixels, and a sequence of images can be recorded at consecutive times during the rolling process. In this section, we first consider two images taken at two consecutive times in a rolling process, that are shown in Figure 5(a)-(b). The first one is used as a reference image and the second one is used as a moved image. Then, we apply the image registration and hypothesis testing procedures described in the previous sections to this dataset, using the same parameter values as those in the simulation examples in Section 5. The estimated parameters in the rigid-body transformation are  $\hat{\theta}_E = (0.003, -0.001, 0.005)^T$  and  $\hat{\theta}_C = (0.008, -0.014, 0.000)^T$ . So, although the estimated parameters in the rigid-body transformation are different by the edge-based and intensity-based methods, the overall trend (e.g., the signs of the estimated parameters) is the same. The three test statistics are calculated to be  $U_E = 6.95$ ,  $U_{EC} = 12.31$ , and  $U_{CC} = 19.36$ , and their corresponding  $p$ -values are 0.335, 0.116, and 0.059.

Therefore, we can not reject the null hypothesis at the 0.05 significant level in this case. Consequently, we can not reject the null hypothesis by the two combination tests  $U_{E,EC}$  and  $U_{E,CC}$  either. The recovered reference images  $Z_M(\widehat{\mathbf{T}}(x, y))$  by  $\widehat{\boldsymbol{\theta}}_E$  and  $\widehat{\boldsymbol{\theta}}_C$  are shown in Figure 5(c)-(d), and the difference image between the ones in Figure 5(a) and Figure 5(c) and the difference image between the ones in Figure 5(a) and Figure 5(d) are shown in Figure 5(e)-(f). It can be seen that the difference images do not contain much non-random patterns, indicating that the observed data do not contain convincing evidence against  $H_0$ .

Next, we keep the reference image shown in Figure 5(a) and use the image taken at the 18th subsequent time point as the moved image. The estimated parameters are  $\widehat{\boldsymbol{\theta}}_E = (-0.008, -0.001, 0.001)^T$  and  $\widehat{\boldsymbol{\theta}}_C = (-0.003, -0.007, -0.004)^T$ . The three test statistics are calculated to be  $U_E = 4.78$ ,  $U_{EC} = 175.34$ , and  $U_{CC} = 114.95$ , and the corresponding  $p$ -values are 0.522, 0.000, and 0.000. So, we can not reject the null hypothesis by the edge-based test  $U_E$ . However, the difference between the two images in the continuity regions are large. So, the null hypothesis is rejected by the tests  $U_{EC}$  and  $U_{CC}$ , and it is rejected by the combination tests  $U_{E,EC}$  and  $U_{E,CC}$  too. In this case, we can conclude that the two images are significantly different even after they are geometrically matched up, and the difference is mainly in the continuity parts. The results corresponding to those in Figure 5 are shown in Figure 6. From the two difference images, it can be seen that they do contain some non-random patterns.

## 7 Concluding Remarks

In the previous sections, we have described some testing procedures for detecting difference between two images, after the images are geometrically matched up properly. Numerical results show that they perform reasonably well in practice. This research is mainly for the ultimate goal of sequential monitoring of images. To achieve that goal, we need to generalize the methods proposed in this paper to cases when multiple images are present in Phase-I image monitoring and to cases when images are sequentially obtained in Phase-II online monitoring. These research topics are left for our future research.

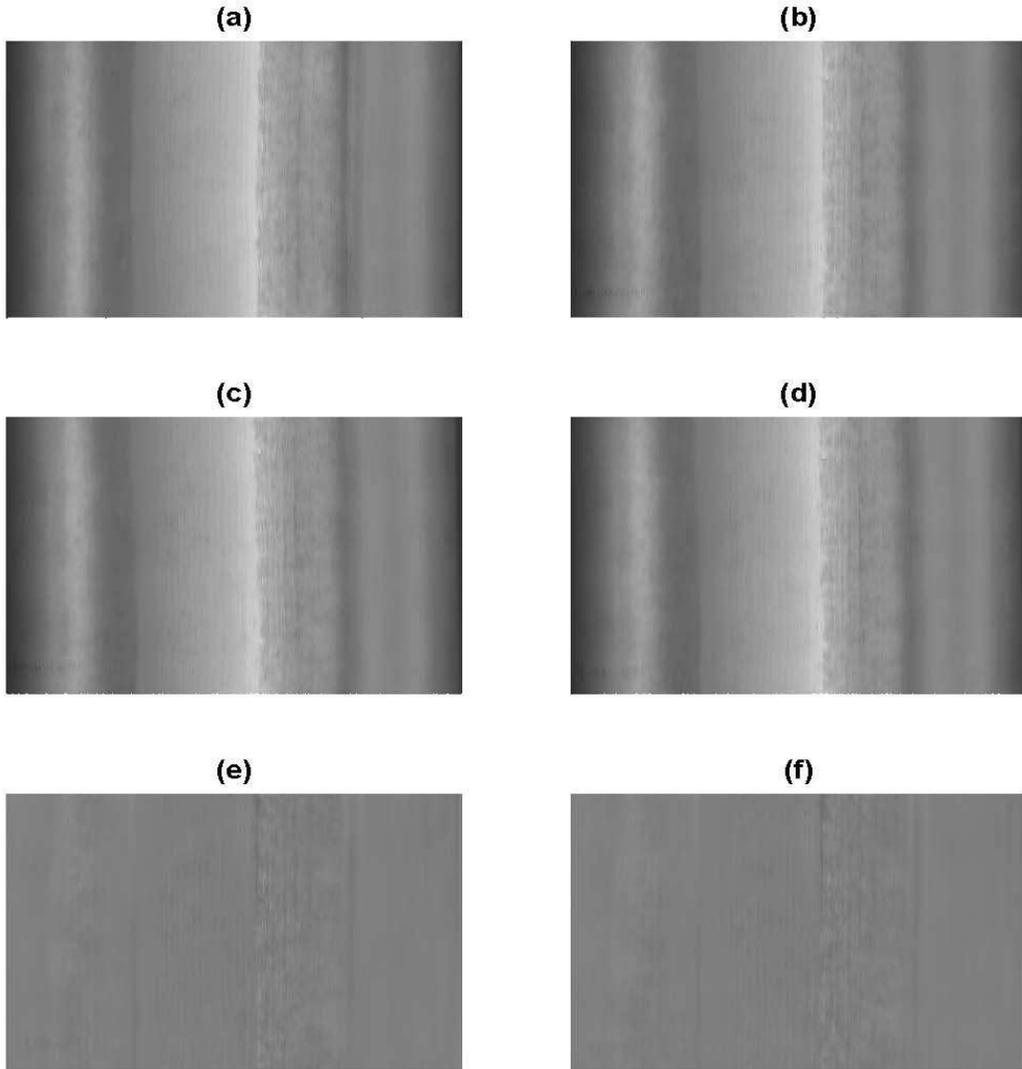


Figure 5: (a)-(b) Reference and moved images in a rolling process obtained at two consecutive times; (c)-(d) recovered reference images  $Z_M(\hat{\mathbf{T}}(x, y))$  by  $\hat{\boldsymbol{\theta}}_E$  and  $\hat{\boldsymbol{\theta}}_C$ , respectively; (e) difference (a) and (c); (f) difference (a) and (d).

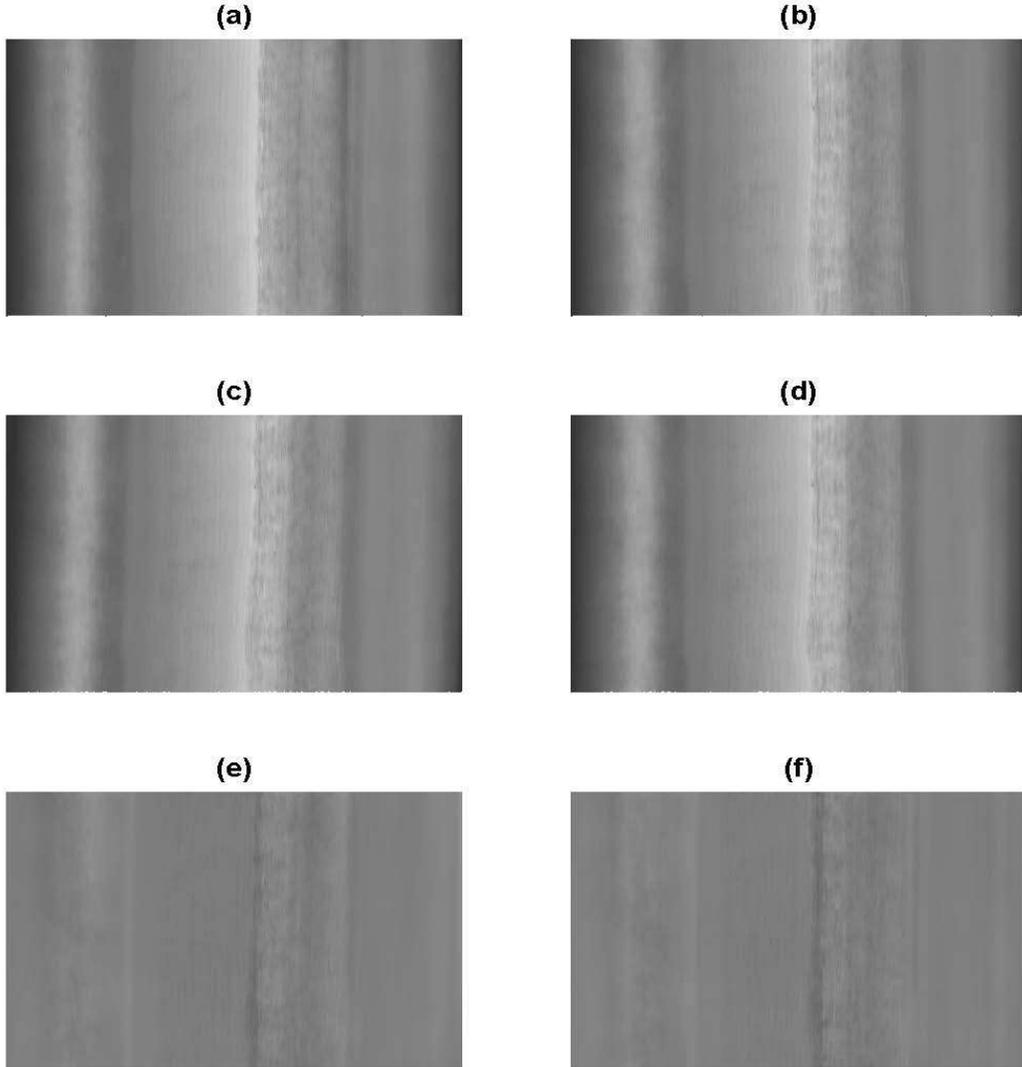


Figure 6: (a)-(b) Reference and moved images in a rolling process obtained at two times that are quite far away; (c)-(d) recovered reference images  $Z_M(\hat{\mathbf{T}}(x, y))$  by  $\hat{\theta}_E$  and  $\hat{\theta}_C$ , respectively; (e) difference (a) and (c); (f) difference (a) and (d).

# Appendices

In the appendices, we provide some theoretical results about the parameter estimators  $\widehat{\theta}_C$  and  $\widehat{\theta}_E$  discussed in Section 2.

## A Conditions used in Theorems 1 and 2

- (C1)  $E(|\varepsilon_R(x, y)|^6) < \infty$  and  $E(|\varepsilon_M(x, y)|^6) < \infty$ ;
- (C2)  $K$  is a Lipschitz-1 continuous circularly symmetric two-dimensional density kernel function with support  $\{(x, y) : x^2 + y^2 \leq 1\}$ ;
- (C3) The true regression function  $R(x, y)$  has piecewise continuous third-order derivatives in each closed subset of  $[0, 1] \times [0, 1]$ , the first-order derivatives of  $R(x, y)$  are continuous on each boundary curve of the pieces, and the first-order derivatives of  $R(x, y)$  have uniformly bounded one-sided limits in the individual pieces;
- (C4)  $\delta_n \rightarrow 0$  as  $n \rightarrow \infty$ ;
- (C5)  $h \rightarrow 0$  and  $nh^3 \rightarrow \infty$ ;
- (C6) The number of non-edge pixels  $|\bar{D}_R|$  satisfies the condition that  $|\bar{D}_R|\delta_n \rightarrow \infty$  and the image intensity function  $M(x, y)$  is not degenerate in the sense of (7) in the design space;
- (C7) The bandwidth  $h^*$  used in edge detection satisfies the conditions that  $h^* \rightarrow 0$ ,  $\log n/(nh^{*4}) \rightarrow 0$  and  $h^*/\delta_n \rightarrow 0$ ; the significant level  $\alpha^*$  used in the edge detection procedure satisfies the conditions that  $\alpha^* \rightarrow 0$ ,  $Z_{1-\alpha^*}/(nh^*) \rightarrow 0$  and  $nh^{*3}/Z_{1-\alpha^*} \rightarrow 0$  where  $Z_{1-\alpha^*}$  is the  $(1 - \alpha^*)$ -th quantile of the standard normal distribution;
- (C8) The image intensity function  $R(x, y)$  has uniformly bounded one-sided slopes on all edge curves;
- (C9) Let  $\Lambda_n(x_0, y_0)$  be a subregion of  $\{(s, t) : s^2 + t^2 \leq d_n^2\}$  in which  $|R(x_0 + s, y_0 + t) - M(T_1(x_0, y_0) + s, T_2(x_0, y_0) + t)|$  is at least  $\tau_n$  where  $(x_0, y_0)$  is an edge pixel in the

reference image. Let  $q_n = \inf_{(x_0, y_0) \in \tilde{D}_R} \frac{|\Lambda_n(x_0, y_0)|}{\pi n^2 d_n^2}$ , where  $|\Lambda_n(x_0, y_0)|$  is the area of  $\Lambda_n(x_0, y_0)$  and  $\tilde{D}_R$  is the set of true edge points in the reference image. We assume that  $q_n \tau_n^2 / d_n^4 \rightarrow \infty$ ,  $q_n \tau_n^2 / h^{*2} \rightarrow \infty$ ,  $q_n \tau_n^2 / r_n \rightarrow \infty$  and  $nd_n q_n \tau_n^2 / \log n \rightarrow \infty$ .

The conditions (C1)-(C3) are common in image registration. To use the Taylor's expansion, we need the condition (C4) for the consistency of  $\boldsymbol{\theta}$ . Intuitively, the condition (C4) says that when the image resolution is higher (i.e.,  $n$  is larger), the geometric difference between the reference and moved images is smaller. This assumption might be reasonable because higher image resolution often implies better imaging technique and consequently the geometric misalignment would be smaller among different images. This assumption is commonly used in the image registration literature (cf., Qiu and Xing 2013a). Condition (C5) is for the consistency of the partial derivative estimators, such as  $\widehat{M}'_x(x_i, y_j)$ . Condition (C6) is for excluding cases when image registration by (5) is not well defined. Condition (C7) is for the consistency of the detected edge points, which is the same as that in Theorem 3.1 in Kang and Qiu (2014). Conditions (C8) and (C9) are for the consistency of the feature matching procedure, and (C9) is similar to those in Theorem 3 in Qiu and Xing (2013a).

## B Consistency of $\widehat{\boldsymbol{\theta}}_C$

**Theorem 1** *Under the conditions (C1)-(C7) given in Appendix A, we have  $\widehat{\boldsymbol{\theta}}_C = \boldsymbol{\theta} + o_p(\delta_n)$ , where  $\delta_n = \sqrt{\phi^2 + \Delta x^2 + \Delta y^2}$ .*

Theorem 1 shows that  $\widehat{\boldsymbol{\theta}}_C$  is a consistent estimator of  $\boldsymbol{\theta}$  under some regularity conditions. Although  $\delta_n$  is assumed to be smaller and smaller when  $n$  gets larger, Theorem 1 says that the distance between  $\widehat{\boldsymbol{\theta}}_C$  and  $\boldsymbol{\theta}$  is much smaller than  $\delta_n$  (i.e.,  $(\widehat{\boldsymbol{\theta}}_C - \boldsymbol{\theta})/\delta_n = o_p(1)$ ).

**Proof.** From Fan and Gijbels (1996), we have

$$\begin{aligned}\widehat{M}'_x(x_i, y_j) &= M'_x(x_i, y_j) + O_p(h^2 + (nh^3)^{-1/2}), \\ \widehat{M}'_y(x_i, y_j) &= M'_y(x_i, y_j) + O_p(h^2 + (nh^3)^{-1/2}).\end{aligned}\tag{10}$$

By the Taylor expansion,

$$\begin{aligned}
& M(x_i, y_j) - R(x_i, y_j) \\
&= \left( M'_x(x_i, y_j)y_j - M'_y(x_i, y_j)x_i \right) \phi + M'_x(x_i, y_j)\Delta x + M'_y(x_i, y_j)\Delta y \\
&\quad + \left( M'_x(x_i, y_j)x_i + M'_y(x_i, y_j)y_j \right) (1 - \cos(\phi)) \\
&\quad + \left( M'_x(x_i, y_j)y_j - M'_y(x_i, y_j)x_i \right) (\sin(\phi) - \phi) + O(\delta_n^2) \\
&= \left( M'_x(x_i, y_j)y_j - M'_y(x_i, y_j)x_i \right) \phi + M'_x(x_i, y_j)\Delta x + M'_y(x_i, y_j)\Delta y + O(\phi^2) + O(\delta_n^2) \\
&= \left( \widehat{M}'_x(x_i, y_j)y_j - \widehat{M}'_y(x_i, y_j)x_i \right) \phi + \widehat{M}'_x(x_i, y_j)\Delta x + \widehat{M}'_y(x_i, y_j)\Delta y \\
&\quad - \left( (\widehat{M}'_x(x_i, y_j) - M'_x(x_i, y_j))y_j - (\widehat{M}'_y(x_i, y_j) - M'_y(x_i, y_j))x_i \right) \phi \\
&\quad - \left( \widehat{M}'_x(x_i, y_j) - M'_x(x_i, y_j) \right) \Delta x - \left( \widehat{M}'_y(x_i, y_j) - M'_y(x_i, y_j) \right) \Delta y + O(\phi^2) + O(\delta_n^2) \\
&= \left( \widehat{M}'_x(x_i, y_j)y_j - \widehat{M}'_y(x_i, y_j)x_i \right) \phi + \widehat{M}'_x(x_i, y_j)\Delta x + \widehat{M}'_y(x_i, y_j)\Delta y \\
&\quad + O_p(\delta_n(h^2 + (nh^3)^{-1/2})) + O(\delta_n^2).
\end{aligned}$$

Define  $\boldsymbol{\varepsilon}$  be a long vector, consisting of  $\{\varepsilon_M(x_i, y_j) - \varepsilon_R(x_i, y_j)\}$ . Then,

$$\widehat{\boldsymbol{\theta}}_C = (\widehat{\mathbf{X}}^T \widehat{\mathbf{X}})^{-1} \widehat{\mathbf{X}} \mathbf{Y} = \boldsymbol{\theta} + (\widehat{\mathbf{X}}^T \widehat{\mathbf{X}})^{-1} \widehat{\mathbf{X}} \boldsymbol{\varepsilon} + O_p(\delta_n(h^2 + (nh^3)^{-1/2})) + O_p(\delta_n^2).$$

Next, we will show that  $(\widehat{\mathbf{X}}^T \widehat{\mathbf{X}})^{-1} \widehat{\mathbf{X}} \boldsymbol{\varepsilon} = O_p(|\bar{D}_M|^{-1/2})$ . By (10), we have

$$|\bar{D}_M|^{-1} \widehat{\mathbf{X}}^T \widehat{\mathbf{X}} = |\bar{D}_M|^{-1} \mathbf{X}^T \mathbf{X} + O_p(h^2 + (nh^3)^{-1/2}) = \boldsymbol{\Sigma} + O(|\bar{D}_M|^{-1}) + O_p(h^2 + (nh^3)^{-1/2}).$$

And  $|\bar{D}_M|^{-1} \widehat{\mathbf{X}} \boldsymbol{\varepsilon} = |\bar{D}_M|^{-1} \mathbf{X} \boldsymbol{\varepsilon} + |\bar{D}_M|^{-1} (\widehat{\mathbf{X}} - \mathbf{X}) \boldsymbol{\varepsilon}$ . Obviously,  $|\bar{D}_M|^{-1} \mathbf{X} \boldsymbol{\varepsilon} = O_p(|\bar{D}_M|^{-1/2})$  and  $|\bar{D}_M|^{-1} (\widehat{\mathbf{X}} - \mathbf{X}) \boldsymbol{\varepsilon} = o_p(|\bar{D}_M|^{-1/2})$ . So  $(\widehat{\mathbf{X}}^T \widehat{\mathbf{X}})^{-1} \widehat{\mathbf{X}} \boldsymbol{\varepsilon} = O_p(|\bar{D}_M|^{-1/2})$ , and  $\widehat{\boldsymbol{\theta}}_C = \boldsymbol{\theta}_C + O_p(\delta_n(h^2 + (nh^3)^{-1/2})) + O_p(\delta_n^2) + O_p(|\bar{D}_M|^{-1/2}) = \boldsymbol{\theta} + o_p(\delta_n)$ .

## C Consistency of $\widehat{\boldsymbol{\theta}}_E$

**Theorem 2** *Under the conditions (C1)-(C4) and (C6)-(C9) given in Appendix A, we have  $\widehat{\boldsymbol{\theta}}_E = \boldsymbol{\theta} + o_p(\delta_n)$ .*

**Proof.** Let  $(x, y)$  be a detected edge point in the reference image, and  $\tilde{D}_R$  be the set of the

true edge points in the reference image. Define

$$(x_0, y_0) = \arg \min_{(a,b) \in \tilde{D}_R} d_H((x, y), (a, b)),$$

where  $d_H((x, y), (a, b))$  is the Euclidean distance between two points  $(x, y)$  and  $(a, b)$ . Define  $R'_{x+}(x_0, y_0), R'_{y+}(x_0, y_0), R'_{x-}(x_0, y_0), R'_{y-}(x_0, y_0)$  to be the first-order one-sided partial derivatives at  $(x_0, y_0)$ . According to Theorem 3.1 in Kang and Qiu (2014), we have  $d_H((x, y), (x_0, y_0)) = O(h^*), a.s.$ , where  $h^*$  is the bandwidth used in edge detection.

Define  $(x'_0, y'_0)$  to the detected edge point in  $D_M$  who is located on the same side of the true edge curve as  $(x, y)$  and whose Euclidean distance to  $\mathbf{T}(x_0, y_0)$  is the smallest among all points in  $D_M$ . Then, we also have  $d_H((x'_0, y'_0), (T_1(x_0, y_0), T_2(x_0, y_0))) = O(h^*), a.s.$ . It is obvious that

$$\begin{aligned} & \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( Z_R(x+s, y+t) - Z_M(x'+s, y'+t) \right)^2 \\ &= \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x+s, y+t) - M(x'+s, y'+t) + \varepsilon_R(x+s, y+t) - \varepsilon_M(x'+s, y'+t) \right)^2 \\ &= \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x+s, y+t) - M(x'+s, y'+t) \right)^2 \\ & \quad + \frac{2}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x+s, y+t) - M(x'+s, y'+t) \right) \left( \varepsilon_R(x+s, y+t) - \varepsilon_M(x'+s, y'+t) \right) \\ & \quad + \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( \varepsilon_R(x+s, y+t) - \varepsilon_M(x'+s, y'+t) \right)^2. \end{aligned}$$

According to Proposition 2 in Qiu (2009), we have

$$\begin{aligned} & \frac{2}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x+s, y+t) - M(x'+s, y'+t) \right) \left( \varepsilon_R(x+s, y+t) - \varepsilon_M(x'+s, y'+t) \right) \\ & \quad = O(\log(\tilde{N})\tilde{N}^{-1/2}), a.s., \\ & \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( \varepsilon_R(x+s, y+t) - \varepsilon_M(x'+s, y'+t) \right)^2 = 2\sigma^2 + O(\log(\tilde{N})\tilde{N}^{-1/2}), a.s. \end{aligned}$$

Because  $\tilde{N} = O(n^2 d_n^2)$ , we have

$$\begin{aligned} & \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( Z_R(x+s, y+t) - Z_M(x'+s, y'+t) \right)^2 \\ &= \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x+s, y+t) - M(x'+s, y'+t) \right)^2 + 2\sigma^2 + O(n^{-1} d_n^{-1} \log(n)), a.s. \end{aligned}$$

If  $R'_{x_+}(x_0, y_0)s + R'_{y_+}(x_0, y_0)t \geq 0$ , by the Taylor's expansion, we have

$$\begin{aligned} R(x+s, y+t) &= R(x_0, y_0) + R'_{x_+}(x_0, y_0)(x-x_0+s) + R'_{y_+}(x_0, y_0)(y-y_0+t) + O(d_n^2 + h^{*2}) \\ &= R(x_0, y_0) + R'_{x_+}(x_0, y_0)s + R'_{y_+}(x_0, y_0)t + O(d_n^2 + h^*). \end{aligned}$$

$$\begin{aligned} M(x'_0 + s, y'_0 + t) &= M(T_1, T_2) + M'_{x_+}(T_1, T_2)(x'_0 - T_1 + s) + M'_{y_+}(T_1, T_2)(y'_0 - T_2 + t) + O(d_n^2 + h^{*2}) \\ &= M(T_1, T_2) + M'_{x_+}(T_1, T_2)s + M'_{y_+}(T_1, T_2)t + O(d_n^2 + h^*). \end{aligned}$$

Thus,  $R(x+s, y+t) - M(x'_0 + s, y'_0 + t) = O(d_n^2 + h^*)$  on the left subregion  $\Omega^+ = \{(s, t) \mid R'_{x_+}(x_0, y_0)s + R'_{y_+}(x_0, y_0)t \geq 0\}$ . Similarly, we also have  $R(x+s, y+t) - M(x'_0 + s, y'_0 + t) = O(d_n^2 + h^*)$  on the right subregion  $\Omega^- = \{(s, t) \mid R'_{x_+}(x_0, y_0)s + R'_{y_+}(x_0, y_0)t < 0\}$ . So,

$$\begin{aligned} &\frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( Z_R(x+s, y+t) - Z_M(x'_0 + s, y'_0 + t) \right)^2 \\ &= 2\sigma^2 + O(d_n^4 + h^{*2}) + O\left(\frac{\log n}{nd_n}\right), a.s. \end{aligned}$$

On the other hand, when  $(x', y') \in O(x, y, r_n) \setminus O(x'_0, y'_0, h^*)$ ,

$$\begin{aligned} &\frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x+s, y+t) - M(x'+s, y'+t) \right)^2 \\ &= \frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( R(x_0+s, y_0+t) - M(T_1(x_0, y_0) + s, T_2(x_0, y_0) + t) + O(r_n) \right)^2 \\ &\geq \frac{1}{\tilde{N}} \sum_{(s,t) \in \Lambda_n} \left( R(x_0+s, y_0+t) - M(T_1(x_0, y_0) + s, T_2(x_0, y_0) + t) \right)^2 + O(r_n) \\ &\geq q_n \tau_n^2 + O(r_n), \end{aligned}$$

where  $\Lambda_n$  is a subregion of  $\{(s, t) : s^2 + t^2 \leq d_n^2\}$  in which  $|R(x_0+s, y_0+t) - M(T_1(x_0, y_0) + s, T_2(x_0, y_0) + t)|$  is at least  $\tau_n$ , and  $q_n = \frac{|\Lambda_n|}{\pi n^2 d_n^2}$  where  $|\Lambda_n|$  is the area of  $\Lambda$ . Thus,

$$\begin{aligned} &\frac{1}{\tilde{N}} \sum_{s^2+t^2 \leq d_n^2} \left( Z_R(x+s, y+t) - Z_M(x'+s, y'+t) \right)^2 \\ &= 2\sigma^2 + q_n \tau_n^2 + O(r_n^2) + O\left(\frac{\log n}{nd_n}\right), a.s. \end{aligned}$$

Therefore, by the conditions  $q_n \tau_n^2 / d_n^4 \rightarrow \infty$ ,  $q_n \tau_n^2 / h^{*2} \rightarrow \infty$ ,  $q_n \tau_n^2 / r_n^2 \rightarrow \infty$ , and  $nd_n q_n \tau_n^2 / \log n \rightarrow \infty$ , we have

$$\inf_{(x', y') \in O(x, y, r_n) \setminus O(x'_0, y'_0, h^*)} MSD((x, y), (x', y')) > MSD((x, y), (x'_0, y'_0)), \quad a.s..$$

So,  $d_E((x^*, y^*), \mathbf{T}(x_0, y_0)) = O(h^*), a.s.$ , which implies that

$$\begin{aligned} x^* &= x_0 \cos(\phi) + y_0 \sin(\phi) + \Delta x + O(h^*) = x_0 + y_0 \phi + \Delta x + O(\delta_n^2) + O(h^*) \\ &= x + y \phi + \Delta x + O(\delta_n^2) + O(h^*), a.s. \\ y^* &= -x_0 \sin(\phi) + y_0 \cos(\phi) + \Delta y + O(h^*) = -x_0 \phi + y_0 + \Delta y + O(\delta_n^2) + O(h^*) \\ &= -x \phi + y + \Delta y + O(\delta_n^2) + O(h^*), a.s.. \end{aligned}$$

Thus,

$$\begin{aligned} &\sum_{(x_i, y_j) \in D_R} (x_i^* - x_i) y_j - \sum_{(x_i, y_j) \in D_R} (y_j^* - y_j) x_i \\ &= \sum_{(x_i, y_j) \in D_R} (x_i^2 + y_j^2) \phi + \sum_{(x_i, y_j) \in D_R} y_j \Delta x - \sum_{(x_i, y_j) \in D_R} x_i \Delta y + O(|D_R|(\delta_n^2 + h^*)), a.s., \\ &\sum_{(x_i, y_j) \in D_R} (x_i^* - x_i) = \sum_{(x_i, y_j) \in D_R} y_j \phi + |D_R| \Delta x + O(|D_R|(\delta_n^2 + h^*)), a.s., \\ &\sum_{(x_i, y_j) \in D_R} (y_j^* - y_j) = - \sum_{(x_i, y_j) \in D_R} x_i \phi + |D_R| \Delta y + O(|D_R|(\delta_n^2 + h^*)), a.s.. \end{aligned}$$

By the above results, the vector  $\mathbf{B}$  in (9) can be written as

$$\mathbf{B} = \mathbf{A}\boldsymbol{\theta} + O(|D_R|(\delta_n^2 + h^*)), a.s..$$

So,  $\widehat{\boldsymbol{\theta}}_E = \boldsymbol{\theta} + O(h^*) + O(\delta_n^2), a.s..$  By the conditions that  $h^*/\delta_n \rightarrow 0$  and  $\delta_n \rightarrow 0$ , we have  $\widehat{\boldsymbol{\theta}}_E = \boldsymbol{\theta} + o(\delta_n), a.s..$  The proof is then finished.

## References

- Balageas, D., Fritzen, C.P., and Güemes, A. (2010), *Structural Health Monitoring*, volume 90. Wiley Online Library.
- Davis, M.H., Khotanzad, A., Flamig, D.P., and Harms, S.E. (1997), “A physics-based coordinate transformation for 3-D image matching,” *IEEE Transactions on Medical Imaging*, **16**, 317–328.
- Freire, L., Roche, A., and Mangin, J.-F. (2002), “What is the best similarity measure for motion correction in fMRI times series?” *IEEE Transactions on Medical Imaging*, **21**, 470–484.

- Geman, S., and Geman, D. (1984), “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 721–741.
- Gijbels, I., Lambert, A., and Qiu, P. (2006), “Edge-preserving image denoising and estimation of discontinuous surfaces,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**, 1075–1087.
- Gonzalez, R.C., and Woods, R.E. (1992), *Digital Image Processing*, Addison-Wesley Publishing Company, Inc.
- Hall, P., Horowitz, J.L., and Jing, B.Y. (1995), “On blocking rules for the bootstrap with dependent data,” *Biometrika*, **82**, 561–574.
- Jiang, B., Wang, C.C., and Liu, H.C. (2005). “Liquid crystal display surface uniformity defect inspection using analysis of variance and exponentially weighted moving average techniques,” *International Journal of Production Research*, **43**, 67–80.
- Jin, N., Zhou, S., and Chang, T.S. (2004), “Identification of impacting factors of surface defects in hot rolling processes using multi-level regression analysis,” *Transactions of the North American Manufacturing Research Institute of SME*, **32**, 557–564.
- Kang, Y., and Qiu, P. (2014), “Jump detection in blurred regression surfaces,” *Technometrics*, **56**, 539–550.
- Kumar, A. (2008), “Computer-vision-based fabric defect detection: a survey,” *IEEE Transactions on Industrial Electronics*, **55**, 348–363.
- Lahiri, S.N. (1999), “Theoretical comparisons of block bootstrap methods,” *The Annals of Statistics*, **27**, 386–404.
- Modersitzki, J. (2009), *Fair: Flexible Algorithms for Image Registration*, SIAM, Philadelphia.
- Mukherjee, P.S., and Qiu, P. (2011), “3-D image denoising by local smoothing and non-parametric regression,” *Technometrics*, **53**, 196–208.

- Patterson, E., and Wang, Z. (1991), “Towards full field automated photoelastic analysis of complex components,” *Strain*, **27**, 49–53.
- Qiu, P. (1998), “Discontinuous regression surfaces fitting,” *The Annals of Statistics*, **26**, 2218–2245.
- Qiu, P. (2004), “The local piecewisely linear kernel smoothing procedure for fitting jump regression surfaces,” *Technometrics*, **46**, 87–98.
- Qiu, P. (2005), *Image Processing and Jump Regression Analysis*, New York: John Wiley & Sons.
- Qiu, P. (2007), “Jump surface estimation, edge detection, and image restoration,” *Journal of the American Statistical Association*, **102**, 745–756.
- Qiu, P. (2009), “Jump-preserving surface reconstruction from noisy data,” *Annals of the Institute of Statistical Mathematics*, **61**, 715–751.
- Qiu, P. (2014), *Introduction to Statistical Process Control*, Boca Raton, FL: Chapman & Hall/CRC.
- Qiu, P., and Nguyen, T. (2008), “On image registration in magnetic resonance imaging,” *IEEE Proceedings of the 2008 International Conference on BioMedical Engineering and Informatics*, 753–757.
- Qiu, P., and Xing, C. (2013a), “On nonparametric image registration,” *Technometrics*, **55**, 174–188.
- Qiu, P., and Xing, C. (2013b), “Feature based image registration using non-degenerate pixels,” *Signal Processing*, **93**, 706–720.
- Rosten, E., and Drummond, T. (2005), “Fusing points and lines for high performance tracking,” *IEEE International Conference on Computer Vision*, **2**, 15081511.
- Saint-Marc, P., Chen, J., and Medioni, G. (1991), “Adaptive smoothing: a general tool for early vision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**, 514–529.

- Sohn, H., Park, G., Wait, J. R., Limback, N. P., and Farrar, C. R. (2004), “Wavelet-based active sensing for delamination detection in composite structures,” *Smart Materials and Structures*, **13**, 153.
- Tomasi, C., and Manduchi, R. (1998), “Bilateral filtering for gray and color images,” *Proceedings of the 1998 IEEE International Conference on Computer Vision*, 839–846, Bombay, India.
- Yan, H., Paynabar, K., and Shi, J. (2016), “Anomaly detection in images with smooth background via smooth-sparse decomposition,” *Technometrics*, in press.
- Zitova, B., and Flusser, J. (2003), “Image registration methods: a survey,” *Image and Vision Computing*, **21**, 977–1000.