2019 IEEE 6th International Conference on Industrial Engineering and Applications

A Real-Time Contrasts Method for Monitoring Image Data

Xiaoxiao Guo, Zhen He, Hui Chen*
Department of management and Economics
Tianjin University
Tianjin, China
e-mail: chenhui2017@tju.edu.cn

***Abstract*—Machine vision inspection integrated with statistical process control is increasingly being used to detect the surface defects in industrial products. The existing methods utilized traditional control charts to monitor the change point or abnormal regions of images. However, there are some challenges in handling high-dimensional and complex image data. This paper proposes to monitor image data based on the real-time contrasts (RTC) method. RTC method converts the monitoring problem to a continuous classification problem by labelling the reference images and the real-time images with different labels. Classification accuracy is used to build statistics for image data monitoring. And a kind of under-sampling method is used to tackle the problem of sample imbalance, which exists in process and has negative effects on the classification accuracy. Performance of the proposed method is compared with that of an alternative method under both single and multiple defects scenarios via simulations. The results show that the proposed method performs better in some situations and can effectively identify the occurrence of shifts in the process.**

Keywords-control charts; change point model; image data; statistical process control; real time contrast

I. INTRODUCTION

In manufacturing processes, quality characteristics like geometry size can be measured and controlled by traditional control charts. However, some sensory quality characteristics like colour and appearance are often inspected by vision. Machine vision inspection is increasingly used in manufacturing processes along with the wide application of computer and sensor technology. Also, it can provide much relevant information in the process of production, such as geometry size and surface defectives. Thus, machine vision system (MVS) is very suitable to inspect the products which quality characteristics are images. These products mainly include liquid crystal display, ceramic tiles, textiles and food products [1].

In recent years, statistical process control (SPC) charts are widely used in processes to detect product faults and keep the process in control. With the application of MVS in processes, some studies have been done on integrating the MVS with SPC charts. Using MVS integrated with SPC in manufacturing process can not only monitor the quality of products, but also make use of numerous quality data and information in product image for diagnostic analysis.

The individual moving range (I-MR) control chart is constructed for each pixel point in the image [2]. However, too many individual control charts would increase time consumption and the risk of false alarm. Reference [3] divide the image of woven fabrics into several windows, and use grey level co-occurrence matrix to extract texture features of every window. And then these feature values are used as a multivariate to calculate Hoteling T^2 statistics to monitor locations of defects. Reference [4] simultaneously consider both temporal and spatial of defects. They propose a spatiotemporal method based on a generalized likelihood ratio control chart. The evaluated performance shows that it has a good estimate of both the change point and location/size of the defects. However, this method assumes that there is only one fault in the image. Reference [5] propose applying multivariate generalized likelihood ratio (MGLR) control chart to monitor multiple faults in image data which had similar performance to [4].

Considering the massive characteristic of image data, we propose a new monitoring method for image data using real-time contrasts (RTC), which is based on classifiers between the reference data and real-time data. Reference [6] first propose RTC method. They construct a control chart with Random Forest and demonstrate that their method has greater efficacy through experiments compared with other traditional methods such as MEWMA. Furthermore, it can be conveniently used when the dataset in phase I is not enough. Another attractive advantage of RTC method is that it performs well in monitoring high-dimensional variable data. Therefore, we consider to extend the method of [6] to image data monitoring. What's more, no matter how many faults (one or multiple) in an image, the method can detect the abnormalities quickly with the same statistic.

In this paper we propose a control chart based on the RTC method to detect defect/defects in industrial image data in the processes considering spatial and temporal aspects simultaneously. We focus on the products of which defects are either characterized by uniformities (e.g. unevenness defects in Liquid Crystal Display) or by specific patterns (e.g. colour defects in ceramic tiles). The image used in the paper is grayscale because it can represent most product surface information and is widely used in industrial environment. By using the method, the fault location and the time when changes occurred can be identified simultaneously.

The remainder of this paper is organized as follows. The real-time contrasts framework for image monitoring is introduced in Section 2. Section 3 evaluates the performance of the proposed method using simulations. Section 4 provides an experiment to apply the proposed method in an industrial environment. Finally, the conclusions and directions of future research are presented.

978-1-7281-0851-3/19/$31.00 ©2019 IEEE

II. Image Monitoring Framework with Real-Time Contrasts

A. Acquisition of Image Data

A digital image can be seen as a function $f(x, y)$, where (x, y) represents the coordinate of pixel point in image spatial. And the value of $f(x, y)$ means the intensity of the pixel point. For an 8-bit grayscale image, which is considered in our paper, the intensity takes value of integer from 0 (black) to 255 (white). More detailed information about image can be obtained from [7].

As mentioned before, our method focuses on two types of industrial images (uniformities and with a specific pattern). To remove the inherit and redundancy information, and to unify these two types of images, we use a simple yet effective method proposed by [4]. First, a standard product image is chosen and is referred to as a nominal image. Then, we subtract it from each captured image and get a resulting image which is independent on the product so that we can use the same method for the two types of images. This operation is similar to image difference in image processing. Due to the image difference, the range of pixel intensities varies from [0, 255] to [-255, 255]. And before this, image registration should be used to ensure that the two images are aligned and improve the accuracy of image difference.

Assume an image with m*n pixels, m represents the rows of pixels, and n represents the columns of pixels. The image can be expressed as $I = \{(x, y): 0 < x \leqslant m, 0 < y \leqslant n, \forall(x, y) \in Z\}$ and Z is the set of pairs of possible integers determined by the camera resolution. The intensity (or intensities) of the corresponding pixel (or pixels) in I will change when any shifts occurred in the process. For a 10*10 image, the possible combinations of the changed pixels will be 2^{100}. It will cause a large computational pressure to monitor each pixel and will increase the probability of false alarm because there is a high correlation between adjacent pixels. At the same time, considering that the pixels of shifts are often clustered, it's reasonable to divide the image into several regions of interests (ROIs) and use the mean of pixels to represent the value of each ROI [4].

We divide the image into some nonoverlapping square-shaped regions with same size. It's important to note that the divided method is different from [4], which increased the size of region step by step. Their method can only monitor one single shift. We try to monitor not only single shift but also multiple defects.

Assuming an image is divided into p regions. Each region contains n_i pixels, and x_{ijt} represents the jth pixel of the ith region in the tth image. Calculate the mean intensity of each region, i.e. $x_{it} = (x_{i1t} + x_{i2t} + \cdots + x_{ijt})/n_{it}$. And x_{it} can represent the image data of the ith region in the tth image.

Assuming an image X_t is divided into p regions, it can be expressed as a vector with p variates, i.e. $X_t = (x_{1t}, x_{2t}, x_{3t}, \dots, x_{it}, \dots, x_{pt})'$

B. Real-Time Contrasts (RTC) Method for Image Data

RTC method converts a monitoring problem to a classification problem based on a supervised learning method [8]. This method defines the reference data as one class and the real-time data as the other class. Then, these two classes are classified by a classifier and the classification accuracy is calculated. At last, a control chart is constructed based on the classification accuracy and used to monitor a process. Research shows that control chart based on RTC method is better than traditional control charts, especially when the dimensions of variates are high (such as one hundred or higher) [6], which is suitable for monitoring image data because in our paper, every image is transformed into a vector with high dimensions.

Assuming an image X is divided into p regions, then, the image can be seen as a vector with p variates.

$$X = (x_1, x_2, x_3, \dots, x_i, \dots, x_p)' \quad (1)$$

where x_i is the mean intensity of the ith region. We assume that x_i follows the normal distribution $N(\mu_{0,i}, \sigma_i^2)$, where the two parameters can be estimated during the phase I monitoring process. In Phase II, if the distribution of image data is normal, all the intensities of ROIs will still follow their primary distributions. If there are faults in the image, the intensities of the corresponding ROIs will change to distributions with unknown parameters $(\mu_{1,i}, \sigma_{1,i}^2)$.

We extend the monitoring system based on RTC method proposed by [6] to image data. As mentioned in Section 1, the method can be applied in both normal distribution and abnormal distribution, while we assume the variates in our paper follow normal distribution for simplicity. Some adjustments were made to fit general production process better. A detailed introduction is illustrated as following.

In statistical process control, an image set S_0 with sample size N_0 from a normal condition is called a reference image set. And an image in the real-time stream is denoted as X_t(t=1,2,3…), which is obtained one by one in the real-time process. To generate a real-time image set, a sliding window is used. The size of sliding window is denoted as N_w, which is fixed in the paper. Therefore, a real-time image set can be represented by $S_t = \{X_{t-N_w+1}, \dots, X_{t-1}, X_t\}$, where $t > N_w$. It contains the most recent N_w images and is updated when a new image is captured. When $t < N_w$, the number of real-time images is fewer than N_w, and $N_w - t$ normal images can be added to generate a real-time image set.

For RTC method, the reference image set is labelled as class 0 and the real-time image set is labelled as class 1. A classifier is trained to divide images into two classes. When there is no fault in the image, it's difficult to distinguish one class from the other. The accuracy of classification will be low. On the contrary, when there is a fault/faults in the real-time process, the accuracy will be high. Consequently, it's reasonable to detect change point of the real-time image set through the estimated classification accuracies.

The RTC monitoring for image data is illustrated in Fig. 1. Every solid circle represents an image observation. One real-time image set is constructed by N_w continuous images. Three real-time image sets in different time points are shown

in Fig. 1. They are $S_w(t-1)$, $S_w(t)$,and $S_w(t+1)$. At every time point, the reference image set S_0 and a real-time image set S_w are classified by a classifier. The classification accuracy is used as statistics to monitor the process.

Let $\hat{p}_k(X_i|t)$ denote the classification accuracy that the image X_i is classified for class k (k=0,1) at time t. Since both S_0 and S_w include more than one images, we compute the mean of these accuracies of S_0 and S_w as the monitoring statistics.

For $X_i \in S_0$,

$$p(S_0, t) = \frac{\sum_{X_i \in S_0} \hat{p}_0(X_i|t)}{N_0} \quad (2)$$

and for $X_i \in S_w$,

$$p(S_w, t) = \frac{\sum_{X_i \in S_w} \hat{p}_1(X_i|t)}{N_w} \quad (3)$$

When there is a shift in a real-time image set, both $p(S_0, t)$ and $p(S_w, t)$ will increase. It can be seen that $p(S_0, t)$ and $p(S_w, t)$ respectively depend on reference image set size N_0 and real-time image set size N_w. N_0 is much bigger than N_w, meaning that there are more observations considered in $p(S_0, t)$. Therefore, $p(S_0, t)$ is more stable to detect defeats than $p(S_w, t)$ [6]. Thus, we only use $p(S_0, t)$ as the monitoring statistic.

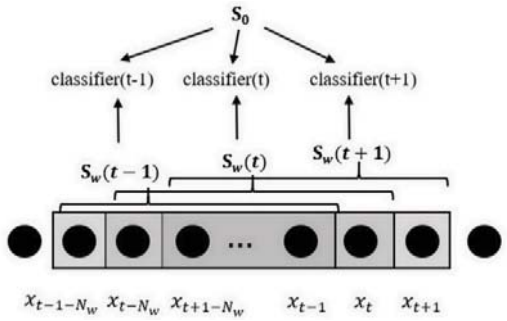


Figure 1. Framework of RTC method for image data.

There are many selections for the classifier in Fig. 1. Reference [6] propose to use random forest as the classifier. Reference [9] utilize support vector machines based on the distance from the decision boundary. And [10] put forward to a method using LDA (Linear Discriminant Analysis), which is also based on the distance. We prefer to choose the random forest as the classifier for two reasons. Firstly, random forest has a wide range of applicability for categorical or continuous data, missing data and different scales data. Secondly, it's convenient to be applied because we don't have to adjust parameters for different data sets.

C. Statistics Based on Random Forest

Random forest has been widely used in classification and regression. It is based on a parallel ensemble of decision trees. Every decision tree is built on the basis of a set of bootstrap samples from the training set. And at every split node, a number of variables are also chosen randomly. Because these two randomly choosing processes, trees in a random forest have a low correlation between each other. And random forest integrates enough number of trees so it has a high accuracy while avoiding over fitting [11].

There are mainly two parameters for random forest. One is the number of chosen variables at each split node, denoted by m. It has been proved that the performance of random forest is not sensitive to the value of m. In most cases, m equals $\sqrt{p}$, where p is the number of variables. The other parameter is the number of trees. Usually it needs to be determined by actual experiment. For most conditions, 500 trees are big enough [10]. In our paper, these two parameters take value of $\sqrt{p}$ and 500 respectively.

The observations which are not used in the construction of a tree are named as out-of-bag (OOB) samples. When observation X_i is an OOB sample, it can be used to estimate the accuracy($\hat{p}_i(x_i)$) belonging to class k=0,1.

$$\hat{p}_i(X_i) = \frac{\sum_{j \in OOB_i} I[\hat{y}(X_i, T_j) = k]}{|OOB_i|} \quad (4)$$

Where $I[\cdot]$ is an indicator function that equals 1 when its argument is true or 0 otherwise. $\hat{y}(X_i, T_j)$ means the predicted class 0 or 1 from tree T_j, ($j = 1,2,3, \dots, n$). OOB_i means the set of trees where observation X_i is an OOB sample. And $|OOB_i|$ means the number of trees in set OOB_i. If $\hat{p}_i(x_i) > 0.5$, the OOB predicted class for observation X_i is 1. If not, the predicted class is 0.

Experiments in [6] showed that we should choose a small value for N_w. Because if the window size N_w is too large, it would include many historical data and the signal of Phase II would be delayed when there is a shift. While N_0 needs to be quite large for enough observations in the reference image set. When N_0 is much larger than N_w, An imbalanced problem will happen. This problem means that the classifier tends to consider more about the majority and ignore the minority. For example, if reference samples N_0 equals 99, and real-time samples N_w equals 1, the classifier will predict all samples as the majority class and in this case the error rate is only 1%. But in fact, the real-time image samples, what we exactly care about, are not classified correctly.

To solve the imbalanced problem, some different methods are proposed, mainly including two categories. One is sampling approach, through over-sampling the minority set or under-sampling the majority set to make two classes have equal samples. The other is by changing algorithm, through modifying the decision threshold to assign different weights to two classes [12]. For example, [13] propose a simple but effective sampling method by selecting some subsets randomly from majority class and using each of them with the minority class to train the classifier. And then combined these learner outputs as the final result.

Assume $N_0 = nN_w$, then we can generate n reference subsets (each one with the size of N_w) by sampling without placement from majority class. Denote the n subsets as $S_0^1, S_0^2, \dots, S_0^n$. Combining each one with the minority subset,

we can get n new datasets (S_0^1, S_w), (S_0^2, S_w), …, (S_0^n, S_w). The output of the ith learner is denoted as $p(S_0^i, t)$.

The final result is denoted as

$$p(S_0, t) = \frac{\sum_1^n p(s_0^i, t)}{n} \quad (5)$$

From (2), (4) and (5), statistics for image monitoring based on RTC method can be calculated. Due to the sliding windows, there are many same images in the neighbouring real-time image sets. Therefore, the statistics have a certain degree of autocorrelations. In this case, the control limit h needs to be determined by computer simulations. The false alarm and detection ability should be balanced in this process. When statistics $p(S_0, t) > h$, a signal will be given and the process distribution has changed. Practitioners can estimate the change point from the control chart. After knowing the faulty image, a region control chart based on all the mean intensities of regions from the faulty image can be plotted. The out of control regions correspond to the fault locations. The details are explained in simulation study.

III. Simulation Study

In this section, we will do some simulations to illustrate performance of the proposed method using nonwoven image. Firstly, we focus on single defect. We compare the performance in this situation with the spatiotemporal method based on GLR proposed by [4]. Then, we pay attention to multiple defects in one image. The performance in this situation is shown with different number of defects.

A nonwoven image shown in Fig. 2 is used to do the simulation study. Firstly, the background is subtracted and then the image is adjusted to the size of 250*250 pixels to reduce the computation load. It is called nominal image, shown in Fig. 3.

Normal images for phase I were generated through adding noise to the nominal image. There are mainly two types of noise adding to an image, including Normal noise and Poisson noise. For details readers can refer to [7]. We prefer to add Poisson noise since Poisson distribution is discrete so it can better represent image pixels intensity. Different faults can be generated by changing the distribution of specific pixels' intensities.

Figure 2. Nonwoven fabric.

To reduce the computational pressure, we ran computer simulations of several situations (with different fault sizes, intensity shift magnitudes and fault locations) with 1000 replications for each one. Because the replication runs are not large enough and it is known that median run length (MRL) is less sensitive to outliers than average run length (ARL) [14]. Meanwhile, in actual production, faults occur more likely during the process instead of the initial start-up [15,16]. Accordingly, we choose steady-state median run length (SSMRL) as the metric to evaluate detection capability of the proposed method. And to compare the performance with that of [4], the in-control SSMRL takes the same value of 150.

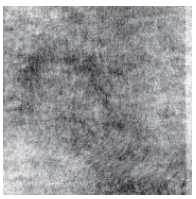


Figure 3. Nominal image for the nonwoven fabric.

Except for SSMRL, we also calculate the relative mean index (RMI) to evaluate the general performance of the control chart for shifts with different magnitudes. It's proposed by [17], and has been used by [18] and [19]. We replace the ARL with SSMRL in the definition of RMI and readers can refer the initial calculation by papers mentioned above. RMI in the paper can be defined by

$$RMI = \frac{1}{N}\sum_{l=1}^{N} \frac{SSMRL_{\delta_l} - SSMRL_{\delta_l,min}}{SSMRL_{\delta_l,min}} \quad (6)$$

where N is the number of different shifts, $SSMRL_{\delta_l,min}$ is the smallest SSMRL among charts for detecting all the shifts, and $SSMRL_{\delta_l}$ is the SSMRL of the control chart for detecting shift δ_l. If RMI is close to 0, it means in general, the control chart has a better performance for different shifts.

For parameters of the simulation experiments, firstly, we divide the image into 625 nonoverlapping square-shaped regions and each region size is 10*10. Therefore, the image is transformed into a vector with 625 variables. 5000 in-control images are generated to estimate the in-control mean vector and covariance matrix. Then, we refer to the parameter values in [6]. The number of reference images is set to 500, and the number of real-time images is 10. That is $N_0 = 500$ and $N_w = 10$, which can achieve the balance of computational pressure and classification effect. For phase I, reference image set is taken from normal images and is labelled as category 0. Real-time image set is also from normal images, but labelled as category 1. Random Forest classification is used to classify these two image sets and statistics can be calculated according to (2), (4) and (5). 1000 experiments are carried out to obtain the control chart limit h in the case that in-control SSMRL is approximately set to 150. For Phase II, reference image set remains unchanged. While real-time image set is taken from real-time image stream, in which first 20 images are normal and a fault is introduced from the 21st image. The two image sets are

classified by random forest classification and the statistics are calculated according to (2), (4) and (5). When statistics exceed the control limit h, the corresponding real-time image sets are out of control.

Next, the monitoring performance of the proposed method is discussed in the case of single defect and multiple defects.

A. Single Shift

For single defect, we use the same defect settings as [4]. Three different defect centres (125,125), (188,206), (158,78) are randomly selected. At each defect centre, rectangular regions with 5 different shift sizes and 10 different intensity shift magnitudes are generated as defects. Therefore, there are total 150 scenarios for single one defect to simulate. The size of defect and magnitude of intensity shift can be seen in Table 1 in detail. 1000 simulation experiments are carried out for each scenario and for each simulation, first 20 images of the image stream are normal, and defective images begin from the 21st.

The results of $SSMRL_1$ in different scenarios including 3 fault centre locations, 5 fault sizes and 10 shift magnitudes are shown in Table 1. As expected, for the same fault size, along with the shift magnitude increasing, the fault is detected more quickly. And for the same shift magnitude, the bigger the fault size is, the earlier the signal is given.

In order to express the monitoring effect of the proposed method on single defect objectively, we compare the performance with image monitoring method based on GLR proposed by [4]. The results are revealed in Table 2. Combined with table 1, we can see that for big faults, the detection capability of the proposed method is a little weaker. But it is still reliable to detect the fault quickly. For small fault, the proposed method has better detection capability than [4], which are expressed in bold in Table 1. What's more, we can see that for all scenarios, especially for small fault sizes, the values of MRI of the proposed method are much smaller than those of [4]. This indicates that the proposed method for monitoring image data based on RTC is relatively better over a range of different shift magnitudes.

TABLE I. $SSMRL_1$ VALUES FROM THE PROPOSED METHOD WITH IN-CONTROL SSMRL=150.5

Fault centre's location	Fault size	Magnitude of intensity shift										RMI
		-10	-5	-3	-2	-1	1	2	3	5	10	
(125,125)	10*10	7	9	**13**	**50.5**	**92**	114.5	**38**	**16**	9	7	**4.09**
	15*15	6	6	8	**9.5**	**33**	**50**	**10**	8	7	6	**1.39**
	20*20	5	6	7	8	**14**	**18**	8	7	6	5.5	**0.69**
	30*30	5	5	6	6	8	8	6	6	5	5	**0.20**
	50*50	4	4	4	5	6	6	5	4	4	4	**0.15**
(188,206)	10*10	7	10	**22.5**	**53.5**	**105.5**	**109**	**45.5**	**20.5**	10	7	**4.58**
	15*15	6	7	8	**9**	**41**	**30.5**	**9**	8	7	6	**1.19**
	20*20	6	6	7	8	**14**	**13**	8	7	6	5	**0.60**
	30*30	5	5	6	6	9	8	6	6	5	5	**0.22**
	50*50	4	4	4	5	6	6	5	4	4	4	**0.15**
(158,78)	10*10	7	9	**16**	**42**	**107**	**100**	**42.5**	**15**	9	7	**4.06**
	15*15	6	6	8	**9**	**30.5**	**35**	**10**	8	7	6	**1.09**
	20*20	6	6	7	8	**12**	**13**	8	7	6	6	**0.32**
	30*30	5	5	5	6	9	9	6	5	5	5	**0.20**
	50*50	4	4	4	4	6	6	5	4	4	4	**0.13**

TABLE II. $SSMRL_1$ VALUES FROM [4] WITH IN-CONTROL SSMRL=148

Fault centre's location	Fault size	Magnitude of intensity shift										RMI
		-10	-5	-3	-2	-1	1	2	3	5	10	
(125,125)	10*10	2	8	41	78	141.5	134.5	81	29	7	2	25.20
	15*15	1	3	6	16	89	92	15	6	3	1	22.20
	20*20	1	1	3	6	34	30	6	3	1	1	7.60
	30*30	1	1	1	2	7	7	2	1	1	1	1.40
	50*50	1	1	1	1	2	2	1	1	1	1	0.20
(188,206)	10*10	3	8	37	88	145	138	87	35	8	3	17.40
	15*15	1	3	6	15	92	90	16	6	3	1	22.30
	20*20	1	1	3	5	27.5	27	5	3	1	1	6.45
	30*30	1	1	1	2	6	6	2	1	1	1	1.20
	50*50	1	1	1	1	2	2	1	1	1	1	0.20
(158,78)	10*10	2	6	23	71	139.5	133	79	29	7	2	23.58
	15*15	1	2	5	11	82	83	13	5	2	1	19.50

20*20	1	1	3	5	21	27	5	3	1	1	5.80
30*30	1	1	1	2	6	6	2	1	1	1	1.20
50*50	1	1	1	1	2	2	1	1	1	1	0.20

TABLE III. $SSMRL_1$ VALUES FOR DETECTING MULTIPLE DEFECTS BASED ON THE PROPOSED METHOD

Fault number	Fault size	Magnitude of intensity shift										RMI
		-10	-5	-3	-2	-1	1	2	3	5	10	
2	10*10	6	8	17	40	94	93	39	17	8	6	4.47
	15*15	5	6	7	9	30	30.5	9	7	6	5	1.29
	20*20	4	5	6	7	12	12	7	6	5	4	0.70
	30*30	4	5	5	6	7	7	6	5	4	4	0.33
	50*50	4	4	4	5	5	5.5	5	4	4	4	0.11
3	10*10	5	7	10	26	85	87	27	10	7	5	4.38
	15*15	4	5	6	8	27	26	8	6	5	4	1.48
	20*20	4	4	5	7	10	10.5	7	5	4	4	0.51
	30*30	4	4	4	5	6	6	5	4	4	4	0.15
	50*50	3	4	4	4	5	5	4	4	4	3	0.33
4	10*10	4	5	6	15	75	72	16	6	5	4	4.20
	15*15	4	4	5	7	20	21	7	5	4	4	1.03
	20*20	4	4	4	6	8	8	6	4	4	4	0.30
	30*30	3	4	4	4	5	5	4	4	4	3	0.33
	50*50	3	3	4	4	4	4	4	4	3	3	0.20

B. Multiple Shifts

In view of the multiple shifts in the image, we set up three different quantities of defects including 2,3 and 4 for simulations. We choose defect centres randomly. At each defect centre, rectangular regions with 5 different shift sizes and 10 different intensity shift magnitudes are generated as defects. The size of defects and magnitude of intensity shifts are the same as those of single defect. 1000 simulation experiments are carried out for each scenario and for each simulation, first 20 images of the image stream are normal, and defective images begin from the 21st.

Table 3 shows the values of $SSMRL_1$ of simulation experiments for multiple defects based on the proposed method in different scenarios.

From the experimental results, we can see that for the same defect size and magnitude, with the increase of the number of defects, the detection speed becomes faster. And for the small magnitude, the change is more obvious. It can also indicate that the proposed method based on RTC can monitor multiple defects in image effectively.

IV. EXPERIMENTAL LABORATORY STUDY

In this section, we provide an example to show how the proposed method to be used in practice. We use the images in [5], which are taken from the Lego car manufacturing lab in Auburn University, USA. This lab simulates the real-life vehicles manufacturing shop. Readers can learn more details about the lab in [5]. The actual monitoring process of image data is divided into three stages involving image acquisition, image processing and statistical process control. The whole process is shown in Fig. 4.

Firstly, in the image acquisition stage, it is important to reduce the changes of external factors such as position and illumination, and we should try to ensure that the image is not affected by any factors except the product itself. Therefore, we need to measure the change of illumination at different locations over time and select the most stable position to collect images. At the same time, it is necessary to set appropriate artificial light and avoid natural light. In the experiment, 100 normal images are collected, 50 of them constitute reference image set, one of them is selected randomly as nominal image, and the remaining 49 are applied for phase I of process control. At the same time, 30 defective images are collected to constitute real-time image stream.

Some image pre-processing steps are applied after image acquisition. First, we convert colour image to greyscale image and resize the grayscale image to 400*200 pixels to ensure the speed and reliability in real-time manufacturing process. Then, image filtering is used to remove image background and uneven illumination. Third, image registration is carried out to make sure the nominal image aligns with other images as far as possible. And at last, these images are subtracted from the nominal image to remove the inherent inter information and highlight defect part. After these pre-processing steps, we get the result images which can be used to monitor the process. Fig. 5 is the main process of image pre-processing for a normal vehicle image.

The processed image is divided into several nonoverlapping regions. Considering the size of image to be detected is 400*200, we set the size of ROI to be 40*20. It results 100 ROIs in one image. The image data can be obtained by calculating the mean intensity of each region.

Statistical process control is divided into two phases. In phase I, we need to determine the size of sliding window and the probability of false alarm to obtain the control limit. Since there are only 49 real-time images and the size of the real-time sliding window is set to 10, 40 real-time image sets can be generated in the image stream. These real-time image sets and reference image set are used to classify, calculate the classification accuracy according to formula 4, and estimate the distribution of statistics. Based on the estimated distribution, simulation experiments are carried out to obtain the control limit. Through 1000 simulation experiments, the upper control limit $h = 0.5287$ when $SSMRL_0 = 149$.

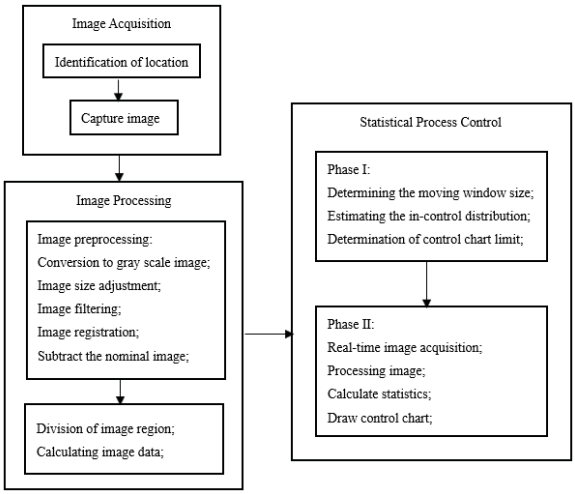
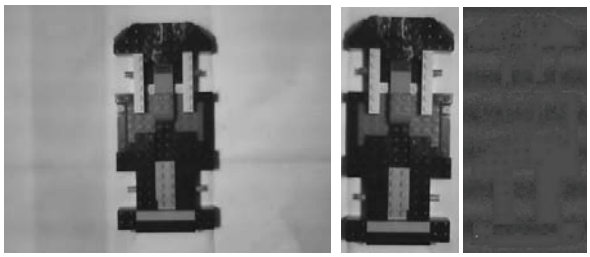


Figure 4. Image data monitoring process.



(a) (b) (c)

Figure 5. Image pre-processing process where (a) grayscale image captured. (b) adjust image size and remove uneven illumination (c) subtract the nominal image.

For Phase II monitoring, we assume that the first 20 images are normal, and the defective images start from the 21st. The same image pre-processing steps were applied on each of these images to obtain the image data. Then, we calculated the statistics based on (2), (4), and (5). The control chart is shown in Fig. 6, where the horizontal axis represents the sequence of real-time image set in the image stream, and the vertical axis represents the accuracy statistics. We can see that the control chart gives an alarm when s=23, and after that, the statistic increases continuously, and tends to be stable when s=30. Furthermore, we use mean intensities of each ROI to identify the fault locations. The 31st image is used to recognise the fault location. The plot of estimated mean intensities against their in-control values of regions is shown in Fig. 7 as proposed by [20]. Apparently, the estimated mean intensities of region 45, 46 and 55 are quite different from their in-control values. And the faulty regions are shown in Fig. 8(b) by hollow rectangle. We compare the normal image with faulty image in Fig. 8. From Fig. 8(a) and Fig. 8(b), we can see that the regions marked by hollow rectangle are different from normal image. There is an extra part in the centre of vehicle.

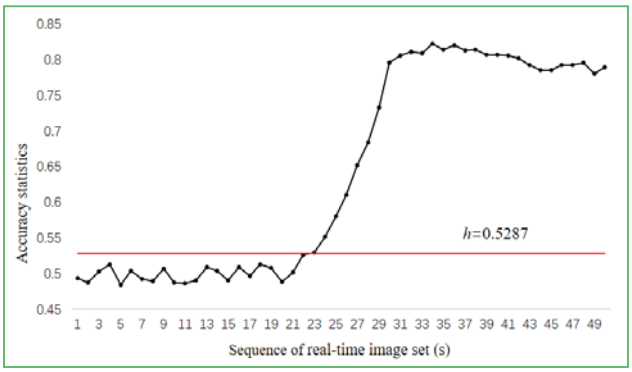


Figure 6. Fig. 6 The control chart of applying the proposed RTC method to the case study.

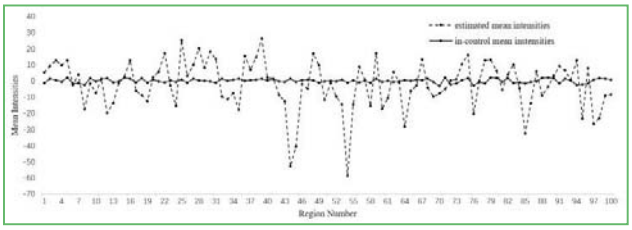
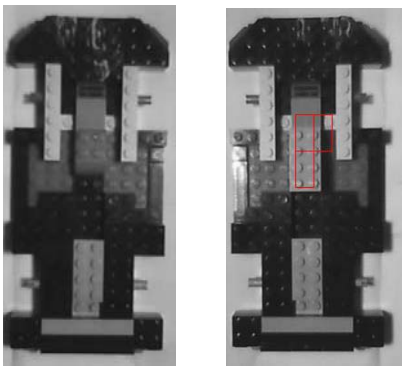


Figure 7. In-control and estimated mean intensity for each region.



(a) (b)

Figure 8. Identification of the faults where (a) normal vehicle image and (b) identified faults marked with hollow rectangles.

V. Conclusions and Recommendations

The paper proposes a real-time contrast method to monitor the defects in grayscale images of industrial parts, of which quality characteristics are uniformities or with a specified pattern. The proposed method converts the image monitoring problem to a classifying problem. Random

forests are used as the classifier and the classification accuracies are utilized to calculate the statistics for control chart. An exampling approach is used to solve the imbalanced problem for image monitoring. Through simulations, we show that the proposed method has a relatively better performance in monitoring the mean shifts in image data. What's more, it can also detect multiple defects. At last, a laboratory experiment is carried out, and the application steps of the proposed method are described in detail. It demonstrates the proposed method can not only provide a good estimation for the occurrence of fault/faults, but also identify the faulty location and size, which can help practitioners find the reason of fault so that they can correct the process back to in-control state in time.

There are two limitations of the paper. First, for big faults, the results SSMRL1 in Table 1 are worse than [4], indicating that detection capability is influenced by window size of the real-time image set. According to the study of [21], when shift size is big or change magnitude is large, a small window size is more efficient. Otherwise, a big window size is better. How to choose the specific window size for different fault sizes and different shift magnitudes needs further study. The second problem is that the statistics in the proposed method are discrete because they are calculated with classification accuracy of decision trees. This may lead to some degree of insensitivity to small differences between faults. Thus, new method needs to found to improve the classifier to make statistics more sensitive to small shifts.

ACKNOWLEDGMENT

The authors would like to thank Dr. Ling Zuo from Shanghai Maritime University, for providing vehicle images used in experiments. This work was supported by the National Natural Science Foundation of China [grant number 71661147003].

REFERENCES

[1] Megahed F. M., W. H. Woodall, and J. A. Camelio, "A Review and Perspective on Control Charting with Image Data," *Journal of Quality Technology A Quarterly Journal of Methods Applications & Related Topics*, vol. 43, no. 2, pp. 83-98, 2011.

[2] Armingol, J. M., Otamendi, J., Escalera, A. D. L., Pastor, J. M., and Rodriguez, F. J., "Statistical Pattern Modeling in Vision-Based Quality Control Systems," *Journal of Intelligent and Robotic Systems: Theory and Applications,* vol. 37, no. 3, pp. 321-336, 2003. (2003). Statistical pattern modeling in vision-based quality control systems. *Journal of Intelligent and Robotic Systems: Theory and Applications, 37*(3), 321-336.

[3] Tunák, Maro, and A. Linka, "Directional Defects in Fabrics," *Research Journal of Textile and Apparel,* vol. 12, no. 2, pp. 13-22, 2008.

[4] Megahed, F. M., Wells, L. J., Camelio, J. A., and Woodall, W. H., "A Spatiotemporal Method for the Monitoring of Image Data," *Quality & Reliability Engineering International*, vol. 28, no. 8, pp. 967-980, 2012.

[5] He, Z., Zuo, L., Zhang, M., and Megahed, F. M., "An image-based multivariate generalized likelihood ratio control chart for detecting and diagnosing multiple faults in manufactured products," *International Journal of Production Research,* vol. 54, no.6, pp. 1771-1784, 2015.

[6] Deng, Houtao, G. Runger, and E. Tuv, "System Monitoring with Real-Time Contrasts, "*Journal of Quality Technology,* vol. 44, no. 1, pp. 9-27, 2012.

[7] Gonzalez, Rafael C, and Wintz, Paul. *Digital image processing*.3rd ed. Upper Saddle River, NJ: Prentice Hall. pp.156-193, 2010.

[8] Hwang, Wookyeon, G. Runger, and E. Tuv, "Multivariate statistical process control with artificial contrasts." *IIE Transactions,* vol. 39, no. 6, pp. 659-669, 2007.

[9] He, Shuguang , W. Jiang , and H. Deng, "A distance-based control chart for monitoring multivariate processes using support vector machines," *Annals of Operations Research,* vol. 253, no. 1, pp. 191-207, 2016.

[10] Wei, Q., Huang, W., Jiang, W., and Zhao, W., "Real-time process monitoring using kernel distances," *International Journal of Production Research,* vol. 54, no. 21, pp. 6563-6578, 2016.

[11] Breiman L. "Random forests." *Machine Learning,* vol. 45, no. 1, pp. 5-31, 2001.

[12] Lin, W. J., and J. J. Chen, "Class-imbalanced classifiers for high-dimensional data," *Briefings in Bioinformatics,* vol. 14, no. 1, pp. 13-26, 2013.

[13] Liu, Xu Ying, "Exploratory Under-Sampling for Class-Imbalance Learning," Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006), Dec 2006, pp: 965-969, doi: 10.1109/ICDM.2006.68

[14] Steiner, Stefan H., and M. Jones, "Risk-adjusted survival time monitoring with an updating exponentially weighted moving average (EWMA) control chart," *Statistics in Medicine,* vol. 29, no. 4, pp.444-454, 2010.

[15] Zhou, Qin, Y. Luo, and Z. Wang, "A control chart based on likelihood ratio test for detecting patterned mean and variance shifts," *Computational Statistics & Data Analysis, vol*. 54, no. 6, pp. 1634-1645, 2010.

[16] Ryan, Anne G., and W. H. Woodall, "Control Charts for Poisson Count Data with Varying Sample Sizes." *Quality Control & Applied Statistics,* vol. 56, no. 3, pp. 23-24, 2010.

[17] Han, Dong, and F. Tsung, "A Reference-Free Cuscore Chart for Dynamic Mean Change Detection and a Unified Framework for Charting Performance Comparison," *Publications of the American Statistical Association,* vol. 101, no.473, pp.368-386, 2006.

[18] Zou, Changliang and P. Qiu, "Multivariate Statistical Process Control Using LASSO," *Journal of the American Statistical Association,* vol. 104, no. 488, pp. 1586-1596, 2009.

[19] Yeh, Arthur B., B. Li, and K. Wang, "Monitoring multivariate process variability with individual observations via penalised likelihood estimation," *International Journal of Production Research*, vol. 50, no. 22, pp. 6624-6638, 2012.

[20] Wang, Sai, and M. R. Reynolds, "A GLR Control Chart for Monitoring the Mean Vector of a Multivariate Normal Process," *Journal of Quality Technology,* vol. 45, no.1, pp. 18-33, 2013.

[21] Jang, Seongwon, S. H. Park, and J. G. Baek, "Real-time contrasts control chart using random forests with weighted voting," *Expert Systems with Applications,* vol. 71, pp. 358-369, 2017.