15th IAPR International Conference on Machine Vision Applications (MVA)
Nagoya University, Nagoya, Japan, May 8-12, 2017.

A Fast Sparse Reconstruction Approach for High Resolution Image-based Object Surface Anomaly Detection

Woon Huei Chai, Shen-Shyang Ho[⋆]*, Chi-Keong Goh*[†]*, Liang-Tien Chia, and Hiok Chai Quek*
Nanyang Technological University, [⋆] Rowan University, [†] Rolls-Royce Singapore Pte Ltd
Chai0093@e.ntu.edu.sg

Abstract

We propose an approach to resolve two issues in a recent proposed sparse reconstruction based anomaly detection approach as a part of automated visual inspection (AVI). The original approach needs large computation and memory for high resolution problem. To solve it, we proposed a two-step sparse reconstruction, 1) the first sparse representation of input image is estimated in a sparse reconstruction with low resolution downsampled images and 2) the high resolution residual values is generated in another sparse reconstruction with the sparse representation. The first step provides the flexibility of freely adjusting the computation and the demand of memory storage with small trade-off of detection accuracy. Moreover, an illumination adaptive threshold with morphological operators is used in the anomaly classification. Empirical results show that the proposed approach can effectively replace the original approach with better results.

1 Introduction

Anomaly detection is important and widely used in various domain to detect patterns which do not conform normal patterns. Its application includes intrusion detection [1], fraud detection [2], electronics fault diagnosis [3] terror-related activities [4], visual inspection [5], etc.

Visual inspection is a process to detect defects which affect quality of product. It is difficult to develop a robust AVI technique which can tackle all the common industrial inspection challenges such as 1) small anomalous region embedded within majority normal background, 2) characteristics within normal regions are diverse, 3) characteristics within anomalies are diverse, 4) small availability of anomalous samples, 5) the inspected data at extreme high resolution.

Sparse representation is used to represent and reconstruct signals and is widely used in image-based problem. The recently proposed sparse reconstruction approach [5] for defect detection is severely limited by its large computation and memory requirements.

This paper presents a fast sparse reconstruction approach to overcome the above mentioned limitations. The details of the proposed approach is elaborated in the next section. Section 3 shows a two-step optimization algorithm for learning sub-optimal task dependent parameters from a global convex and local non-convex optimization problem. Section 4 shows the empirical results including the comparisons of anomaly detection results of both new proposed and original sparse reconstruction approaches [5]. Section 5 concludes this paper.

2 Methodology

Figure 1 illustrates the flow chart describing the methodology of our proposed approach. The flow chart of our proposed anomaly detection approach is in Figure 1. The novelty of the new proposed fast sparse reconstruction is that it is capable to detect anomalies on high resolution images which the original approach [5] is limited to. It provides the flexibility of freely adjusting the computation and the demand of memory storage with small trade-off of detection accuracy. Moreover, a novel threshold which adapts the illumination condition in input image replaces the original universal threshold.

2.1 Fast Sparse Reconstruction

Both original $w' \times h'$-dimensional input image I' (w' and h' are width and height of original input image) and its $w \times h$-dimensional downsampled image I are firstly concatenated either row-wise or column-wise to form p' ($p' = w' \times h'$)and p ($p = w \times h$)-dimensional vectors, x' and x. The p-dimensional input data vector, $x = (x_1, x_2, \ldots, x_p)^T$, can be reconstructed with a $p \times m$-dimensional matrix, D (dictionary/design matrix with m words/atoms), and a m-dimensional vector, α, in the equalization form of $x = D\alpha + \lambda e$, where λ is a constant and λe is a p-dimensional residual vector caused by noises or anomaly. The α and the equalization are called sparse representation of image I' and I and sparse reconstruction if α is a sparse vector. The values of α and e can be found through the following optimization problem:

$$\min_{\alpha,e} \|\alpha\|_1 + \|e\|_1 \quad (1)$$
$$\text{s.t.} \quad x = D\alpha + \lambda e$$

where λ is a parameter which controls the trade-off between sparsities of α and e. $\|\alpha\|_1$ ensures only few relevant atoms are chosen in the reconstruction and $\|e\|_1$ is used due to the prior knowledge that anomaly is only a small portion in x (most values in residual vector should be around zero).

In our problem, D stores the global information of normal samples. Each atom of D is a p-dimensional vectorized normalized downsampled anomaly free image. In general, D can be constructed with dictionary learning [6, 7] too. After the sparse representation α is estimated, the sparse reconstruction at high resolution (2) is conducted to have the p'-dimensional residual values e' which can be further reorganized to $w' \times h'$-dimensional residual image at original resolution for anomaly classification, where atoms in D' are vectorized normalized anomaly free images at original resolution with the same order of atoms in D.

$$x' = D'\alpha + e', \quad (2)$$

Note that formula (1) is the computation and memory storage bottleneck of the whole sparse reconstruction process if p is large. However, we can adjust the value of p freely with any downsampled resolution with a small trade-off in final anomaly detection accuracy to speed up the process and reduce the memory storage

© 2017 MVA Organization

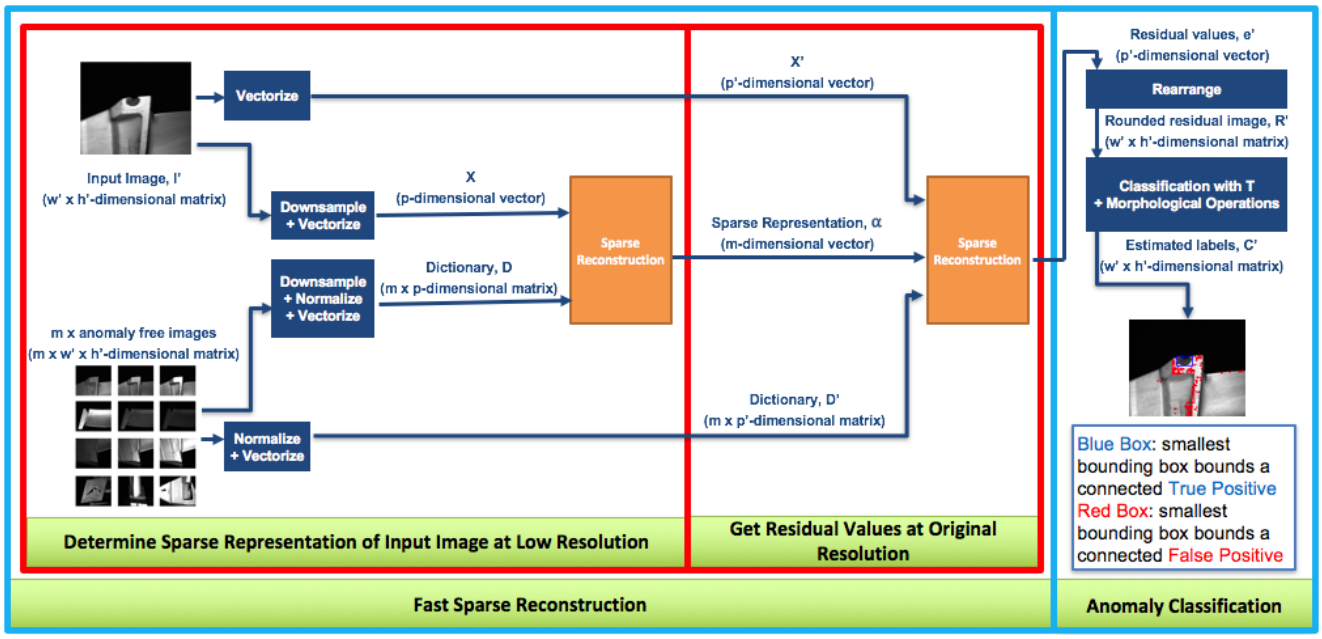

Figure 1. The flow chart of our proposed anomaly detection approach. I' is the original input image; m is the number of anomaly free images to construct the dictionaries; w', h', w and h are widths and heights of original and downsampled images; p' and p are equal to $w' \times h'$ and $w \times h$ respectively. The description of other symbols can be found in Section 2.1 and 2.2.

requirement for the process. Hence, the whole process is called fast sparse reconstruction.

Also note that it saves the computation for estimating α and high resolution e using high resolution (high resolution images) x and D with original approach does in [5].

2.2 Anomaly Classification

Firstly, all the values in e' are rounded to integers between 0 and 255. The vector of rounded e' is called residual vector e'_r and the image form of e'_r is called preprocessed residual image R'. All pixels x'_i are classified into two classes, normal and anomalous pixels, based on the classifier $C(x'_i)$ as follows:

$$C(x'_i) = \begin{cases} 1 & |e'_{r,i}| - T > 0 \ \& \ D_l \le D_i(T) \le D_u \\ 0 & otherwise, \end{cases} \quad (3)$$

where x'_i is a pixel in the input image. $C(x'_i)$ returns 1 (estimated anomalous pixel) if the absolute residual value, $|e'_{r,i}|$, at that pixel is higher than threshold T, the pixel is in a connected region which absolute values are higher than threshold and the dimensions $D_i(T)$ of the smallest bounding box which bounds the connected region is between the lower and the upper bounds, D_l and D_u. The bounding box constraint is for removing noise and fulfilling the inspection requirements (i.e. the tolerance of acceptable defect and unacceptable anomaly specifications such as sizes of defects). Otherwise, $C(x'_i)$ returns 0 (estimated normal pixel). Morphological operations (opening and closing morphological operations of square kernel of size S_k) may be applied on the thresholded residual image before forming the connected region and estimate the labels for removing noise.

Note that, the threshold T in formulation (3) is designed to adapt the illumination condition in the image. It replaces the universal threshold used in the original approach [5].

In (3), the main challenge is the selection of the threshold T. Let $N_c(t)$ be the number of connected estimated anomalous pixel regions after applying (3) with $T = t$. Define

$$T = min(\{t | t = 0, 1, ..., 255, N_c(t) \ge \eta\}), \quad (4)$$

where t is an integer between 0 and 255 and η is a predefined parameter. It is the lowest integer threshold which has at least η connected estimated anomalous pixel regions. η is dataset dependent (i.e. the characteristics of noise and the number of anomaly occurrence in an image). Hence, η needs to be re-estimated when new dataset has different characteristics (i.e. the changes of target defect type, the changes of equipment such as camera and the setup of illumination sources and the changes of inspected product, etc.).

3 Parameter Optimization

In formulas (1) and (4), there are two dataset dependent parameters, λ and η need to be learned. These two parameters depend on the characteristics of anomaly to be detected, the environment, the image acquisition devices, the inspected products, etc. Based on our analysis[1], the optimization problem of accuracy (5) with changing either one of the two parameters is convex globally and non-convex locally respectively. Hence, we proposed a specific supervised parameter learning algorithm to tackle this special non-convex problem.

The parameter learnings are done with a two-step greedy search in Algorithm 1. Algorithm 2 and 3 learn λ and η respectively. The two Algorithms (Step 4 and 5 in Algorithm 1) are iterated until either the validation accuracy converges (depends on the stopping criterion of accuracy, S_1) or the maximum number of iterations S_2 is reached.

In Algorithm 1, Step 1 initializes all the parameters and variables required for the two-step greedy search. η_o and λ_o are the optimized η and λ; D_l and D_u, the lower and upper bounds of the dimensions of the smallest bounding box size which bound a connected region

[1]Due to limitation of article's length, the details of the analysis will be presented in a journal which we are going to submit soon.

Algorithm 1: Finding best λ, η and T

Input: I_t, set of labeled training images; D, D', dictionaries with atoms at low and high resolutions respectively; S_1, S_2, stopping criteria of accuracy and number of iteration; N, search granularity; S_k, kernel size; β, trade-off coefficient

Output: λ_o and η_o

1: $\eta_o = 1000; \lambda_o = 0; D_l = 0; D_u = 1000; N_m = 0; maxAccp = 1; maxAcc = 0; \lambda_{min} = 0.0001; \lambda_{max} = 5; \eta_{min} = 0; \eta_{max} = 1000$
2: **while** $\|maxAcc - maxAccp\| > S_1$ *or* $N_m < S_2$ **do**
3: $N_m = N_m + 1; \lambda_p = \lambda_o; \eta_p = \eta_o; maxAccp = maxAcc$
4: λ_o, $maxAcc$ =Parameter optimization $(I_t, D, D', S_1, S_2, N, S_k, D_l, D_u, \beta, \lambda_{min}, \lambda_{max}, \eta_o, 1)$
5: η_o, $maxAcc$ =Parameter optimization $(I_t, D, D', S_1, S_2, N, S_k, D_l, D_u, \beta, \eta_{min}, \eta_{max}, \lambda_o, 2)$
6: **if** $maxAccp > maxAcc$ **then**
7: $maxAcc = maxAccp; \lambda_o = \lambda_p; \eta_o = \eta_p$
8: break
9: **end if**
10: **end while**

which is being classified as anomaly; N_m is the iteration number of the main loop; $maxAccp$ and $maxAcc$ are the validation accuracies with previous and current optimized parameters respectively; λ_{min}, λ_{max}, η_{min} and η_{max} are the lower and upper bounds of λ and η for the parameter searches. Steps 2 to 10 is the main optimization loop. The whole loop ends when either the difference between the previous and the latest optimized accuracies is not longer larger than S_1 or the iteration number is equal to S_2. Step 3 calculates the iteration number N_m of the main loop and update the most recent optimized parameters and theirs corresponding accuracies. Steps 4 and 5 optimize λ' and η respectively by fixing the other parameter. Step 6 to 9 check whether the current optimized accuracy is lower than the previous one. The loop is broken if it is lower and the optimized parameters are set to be equal to the previous one. Otherwise the loop continues with the stopping criteria stated at Step 2.

In Algorithm 2, the input P_c (1 or 2) states the parameter (λ or η) to be learnt. The specified parameter is learnt and is returned with its corresponding anomaly detection accuracy. Step 1 initializes the iteration number N_s of the optimization loop (Step 2 to 16) and the variables *acca* and *accb*. *acca* and *accb* are the highest two accuracies determined from the parameter search list. The optimization loop stops when either one of the stopping criteria, S_1 or S_2 is met. Step 3 calculates the iteration number N_s. Step 4 to 14 compute all accuracies with different parameter values defined at Step 5. The values depend on the search granularity N, the lower and upper bounds, P_{min} and P_{max}, of parameter search range. The accuracy of each training image is computed with Algorithm 3 at either Step 8 or 10 depending on which parameter is learnt. Step 15 updates the lower and upper bounds of the search range for next iteration.

Algorithm 3 computes the weighted anomaly detection accuracy with formula (5).

$$Acc_b(\beta) = \beta\,(sensitivity) + (1-\beta)\,(specificity) = \beta\frac{TP}{P} + (1-\beta)\frac{TN}{N}, \quad (5)$$

where TP, TN, P and N are the number of true positives, true negatives, positives and negatives respectively. Acc_b is the sum of the weighted sensitivity (recall) [8] and specificity [8] measures. Acc_b alleviates the imbalanced dataset challenges in many real-world problems especially for anomaly detection task which has infrequent anomalies.

Algorithm 2: Parameter optimization

Input: I_t; D; D'; S_1; S_2; N; S_k; D_l; D_u; β; P_{min}; P_{max}, lower and upper bound of parameter search range; P'_o, fixed parameter; P_c, parameter choice , 1 for λ, 2 for η

Output: P_o and Acc_o

1: $N_s = 0; acca = 1; accb = 0$
2: **while** $\|acca - accb\| > S_1$ *or* $N_s < S_2$ **do**
3: $N_s = N_s + 1$
4: **for** $i = 0$ to N **do**
5: $P[i] = P_{min} + \frac{i}{N}(P_{max} - P_{min})$
6: **for** $j = 1$ to N_t **do**
7: **if** $P_c = 1$ **then**
8: $acc_t[j]$ =Accuracy computation $(I_t[j], D, D', P[i], P'_o, S_k, D_l, D_h, \beta)$
9: **else if** $P_c = 2$ **then**
10: $acc_t[j]$ =Accuracy computation $(I_t[j], D, D', P'_o, P[i], S_k, D_l, D_h, \beta)$
11: **end if**
12: **end for**
13: $acc[i] = \sum_{n=1}^{N_t} \frac{1}{N_t} * acc_t[j]$
14: **end for**
15: $k1 = \arg\max_i(acc); k2 = \arg\max_i(acc \setminus acc[k1]); acca = acc[k1]; accb = acc[k2]; P_o = P[k1]; P_{min} = \min(P[k1], P[k2]); P_{max} = \max(P[k1], P[k2])$
16: **end while**
17: $Acc_o = acc[k1]$
18: Return P_o and Acc_o

Algorithm 3: Accuracy Computation

Input: I, input image; D; D'; λ; η; S_k; D_l; D_h;β

Output: *Accuracy*

1: Find sparse representation, α, using (1) for I with D and λ
2: Find residual vector, e', using (2) with D' and α
3: Round all the values in the residual vector to integers between 0 and 255 and to have the residual image R'
4: Find the threshold T using (4) with given S_k and η
5: Apply (3) to obtain the labels of all pixels with T, D_l, D_h and S_k
6: Calculate *Accuracy* with β and (5)
7: Return *Accuracy*

4 Experimental Results

4.1 Metallic Object Dataset

The dataset is a subset of a large number of images collected from an AVI process for quality control in a metallic object manufacturing[2]. Anomalies are the defects on metallic surfaces. The dataset consists of 891 images with original resolution of 2050×2448. There are three defect types: "Melt", "Plus Metal", and "Scuff". Each 2 instances of same defect type are grouped together for a experimental trial. There are 7 instances: "Melt A" (58 [images contain defects], 69 [defect free images]), "Melt B" (58, 161), "Melt C" (58, 69), "Plus Metal A" (53, 69), "Plus Metal B" (52, 23), "Scuff A" (53, 46) and "Scuff B" (57, 63). All instance pairs are organized as follows: "Melt 1" ("Melt A" and

[2]Due to confidentiality, the description is limited.

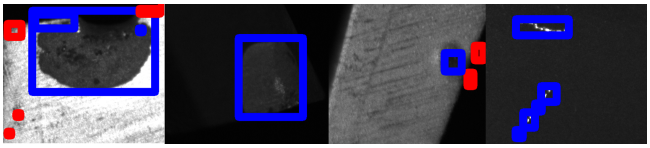

Figure 2. Images contain "Melt" (Left), "Plus Metal", "Melt" and "Scuff" defects. Blue and red boxes are the smallest bounding boxes which bound individual connected True Positive and False Positive respectively.

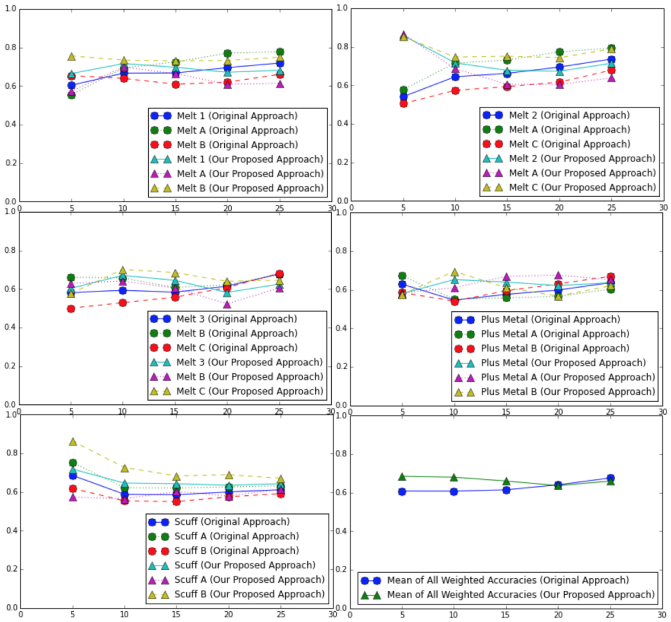

Figure 3. The means of balanced accuracies (weighted accuracy in (5) with $\beta = 0.5$) and the individual balanced accuracies of all experimental trials and the overall means of all means of balanced accuracies of original and our proposed approaches are presented. X-axis represents widths and heights of the output images. The resolution of downsampled images used in determining sparse representations of input images in the our proposed approach is fixed at 5×5. No morphological operation is used.

"Melt B"), "Melt 2" ("Melt A" and "Melt C"), "Melt 3" ("Melt B" and "Melt C"), "Plus Metal" ("Plus Metal A" and "Plus Metal B") and "Scuff" ("Scuff A" and "Scuff B"). All images in the same instance are taken from the same viewpoint with different illumination conditions. In each experimental trial, each instance takes turn to be training and testing sets.

4.2 Comparisons of Proposed and Original Approaches

The default values for parameter and variable initialization are declared in all algorithms presented in Section 2.

Figure 2 shows four examples of anomaly detection results based on our proposed approach. Balanced accuracy (weighted accuracy in (5) with $\beta = 0.5$) is used as performance measure in all experiments described in this subsection. Figure 3 shows the results of all experimental trials. The new proposed approach outperforms the original approach in low resolution output. The performance of both approaches converges in high resolution output.

Note that our proposed approach spends average of 0.0446 seconds to solve the optimization problem (1) for one image at resolution of 25×25 using linear programming function in Matlab. Original approach takes average computation time of 3.5727 seconds to solve the optimization problem which estimates the optimal sparse representation. Two identity matrices corresponding to e in the equality consumes memory of 10GB RAM for an input image with resolution of 160×160 in the original approach.

5 Conclusions

In this paper, we present a new proposed approach to resolve two critical issues, large computation and high memory storage demand, of a recent impractical sparse reconstruction anomaly detection approach for anomaly detection on high resolution images. Moreover, a new threshold which is able to adapt the illumination in the image is proposed to replace the universal threshold used in the original approach. Empirical results show that the new proposed method can replace the impractical original approach with better anomaly detection performance for low resolution outputs and approximately equal performance for high resolution outputs.

6 Acknowledgment

This work was conducted within the Rolls-Royce@NTU Corporate Lab with support from the National Research Foundation (NRF) Singapore under the Corp Lab@University Scheme and Energy Research Institute@NTU under Interdisciplinary Graduate School in Nanyang Technological University.

References

[1] Miao Xie, Song Han, Biming Tian, and Sazia Parvin, "Anomaly detection in wireless sensor networks: A survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1302–1325, 2011.

[2] Clifton Phua, Vincent Lee, Kate Smith, and Ross Gayler, "A comprehensive survey of data mining-based fraud detection research," *arXiv preprint arXiv:1009.6119*, 2010.

[3] M El Hachemi Benbouzid, "A review of induction motors signature analysis as a medium for faults detection," *IEEE transactions on industrial electronics*, vol. 47, no. 5, pp. 984–993, 2000.

[4] Y Elovici, A Kandel, M Last, B Shapira, and O Zaafrany, "Using data mining techniques for detecting terror-related activities on the web," *Journal of Information Warfare*, vol. 3, no. 1, pp. 17–29, 2004.

[5] Woon Huei Chai, Shen-Shyang Ho, and Chi-Keong Goh, "Exploiting sparsity for image-based object surface anomaly detection," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 1986–1990.

[6] Cewu Lu, Jiaping Shi, and Jiaya Jia, "Online robust dictionary learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 415–422.

[7] Cewu Lu, Jianping Shi, and Jiaya Jia, "Scale adaptive dictionary learning," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 837–847, 2014.

[8] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.