Scholars Journal of Engineering and Technology (SJET) **ISSN 2321-435X (Online)**
Sch. J. Eng. Tech., 2014; 2(4C):628-631 **ISSN 2347-9523 (Print)**
©Scholars Academic and Scientific Publisher
(An International Publisher for Academic and Scientific Resources)
www.saspublisher.com

Research Article

Defective Key Detection In Keyboard Using Image Processing Techniques

Shafali Goyal*[1], Reecha Sharma[2]
Department of Electronics and Communication Engineering, Punjabi University, Patiala, Punjab-147002,India

***Corresponding author**
Shafali Goyal
Email: shafali1991@gmail.com

Abstract: There are many limitations related to testing of manufactured item. Solution of these limitations is costly due to the human evaluation procedure, and efficiency is not so promising. Thus, automation of the process has resolved residual issues. With the similar theme, we have proposed an algorithm for the detection of defective keys in keyboard using image processing techniques. Many different types of keyboard's images were acquired for algorithm validation. Proposed algorithm was implemented in MATLAB software and tested on acquired images. Results have shown the defective keys with annotations on the image.
Keywords: Keyboard, adaptive image thresholding, image segmentation, morphological operations, defect detection, automatic detection

INTRODUCTION

Image processing is a large area to solve the problem of manufacturing [1, 2]. There are many applications which make the process automaticusing computational techniques applicable on images. The possible applications of computer vision in today's manufacturing process are virtually endless [3].In every industry, eminence objectives are flattering stringent, production throughput has to be enlarged, and the demand for the throughputexpansions is continual [4-6]. If a visual article can be captured using a camera, a vision system can be used to examine it, gauge it, verify it or guide it. Artificial vision applies digital image processing and analysis to tackle real problems in the industrial production, mainly of standardized products, in real time conditions [7]. Manual inspection or monitoring of any continuous process is commonly agreed to be inefficient, especially because of its repetitive and tedious nature.

Similarly, we have emphasis on the industry that develops keyboards. There is a requirement of enhancement in the process of developing keyboards. There are several keyboard-tester software which test the internal functioning of the keys [8, 9]. But, no software is available for evaluating the keyboard structure physically. We want to develop a system which could be used on assembly line for evaluation of keyboards so that manufactured keyboard can be evaluated if a key is defected. This type of system can guide and automatically assess the quality ofkeyboard which is the final outcome of the process. This type of system can increase the throughput in the overall production of the industry.

Similar types of systems are being used for automation. Here, we want that the evaluation of keyboard structure should also be automatic with the use of computer algorithm.

This paper solves a problem to identify defective keys using image processing techniques. For validation of proposed algorithm, data was generated using some different types of keyboards. Next section describes the methodology of the experiment. Proposed algorithm is discussed in details in third section. Result and conclusions of the experiment is given in last section of this paper.

METHODOLOGY

We had obtained few different types of keyboards for experimentation. Key size and overall structure of the keys was different in each keyboard. Several images were acquired of every keyboard from the perpendicular distance of 40 cm from the keyboard. Each image was acquired with different key defection e.g. with one key defection, with two keys defection, with three or more key defection and with no key defection. Thus, a database was created with several images of keyboards with different key defection.

We have designed an algorithm which automatically detects available defective keys in the image of keyboard. Algorithm is implemented in MATLAB version 13 and tested with the created database images.

A flow chart of the followed steps is shown in Figure 1. Next section discusses the proposed algorithm for defective key detection in keyboard.

Design of algorithm

An algorithm was designed to detect defective keys in keyboard has following steps.

Step 1:- Keyboard image

An image of keyboard with defective key(s) or no defective key was the input of the algorithm. Type of keyboard and nature of defection was not input to the system. Figure 2(a) shows the input keyboard image which has two defective keys.

Step 2:- Reduce the resolution of image for improving the speed

Image processing techniques traverse each and every pixel of the image. Thus, big resolution images take much time to process. Fault identification should be completed at millisecond/microsecond scale for giving aid in the speed of the industry-production. Thus, image resolution was reduced to speed-up for the system.

Step 3:- Convert color image into gray scale image

Similarly, to increase the speed of the process and to process only one frame rather than three frames, Color image was converted into gray scale image.

Step 4:- Segmentation using adaptivethresholding for background subtraction

Histogram was calculated for the gray scale image and a thresholding value was calculated using adaptive thresholding method [10].Image histogram of input keyboard image is shown in Figure 2(b). Image was segmented using calculated thresholding value. For each image, thresholding value was different and dependent on its nature of image neighborhood- pixel values. Resultant image was having only keyboard image as shown in Figure 2(c).Background portion of the image was removed with the segmentation procedure.

Step 5:- Image morphological operations to identify the defect or unusual pixels

Image morphological operations identify the defective portion or unusual pixels of the keyboard image as shown in Figure 2(d). Other non-defective portions remain very little in size and could be ignored. Thus, defective portion could be tracked. We had used image dilation, image erodation, and black & white area open functions for morphological operations. These operations grow the defective portion and compress non-defection portion of the image.

Step 6:- Joining of disconnected components in horizontal and vertical direction

With the morphological operations, some pixels do not appear as 1 or defective portion may have some discontinuity. Thus, to avoid this discontinuity, we have joined disconnected component in horizontal as well as vertical direction. It makes a defective portion big in size so that it can be identified by the computer program. Figure 2(e) shows the defective image pixels after joining of disconnected component.

Step 7:- Morphological operations to detect defects with good confidence

By joining of disconnected component, probable defective portions in keyboard become larger. Some portions which are not defective but having enough amounts of defective pixels, would not be increase through joining of disconnected component. The only defective portions would be increasing with the joining of disconnected component. Thus, these types of portion would be easily identifiable as shown in Figure 2(e). The amount of pixels is increased in defective regions only. Other regions were remained same.

Step:- 8 Annotate defective area on keyboard

With the identification of larger areas in the keyboard image, a rectangle was drawn automatically to show the defective portion as shown in Figure 2 (f). Image shows two defective keys with the annotations drawn over it.

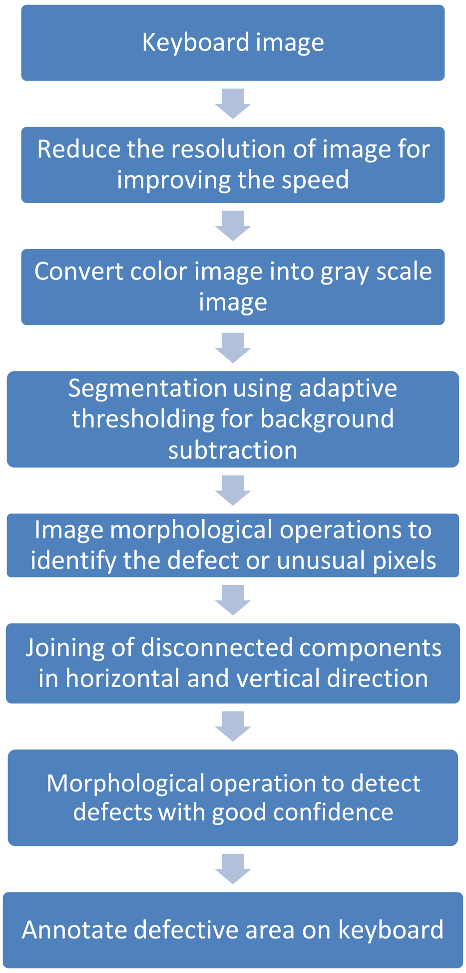
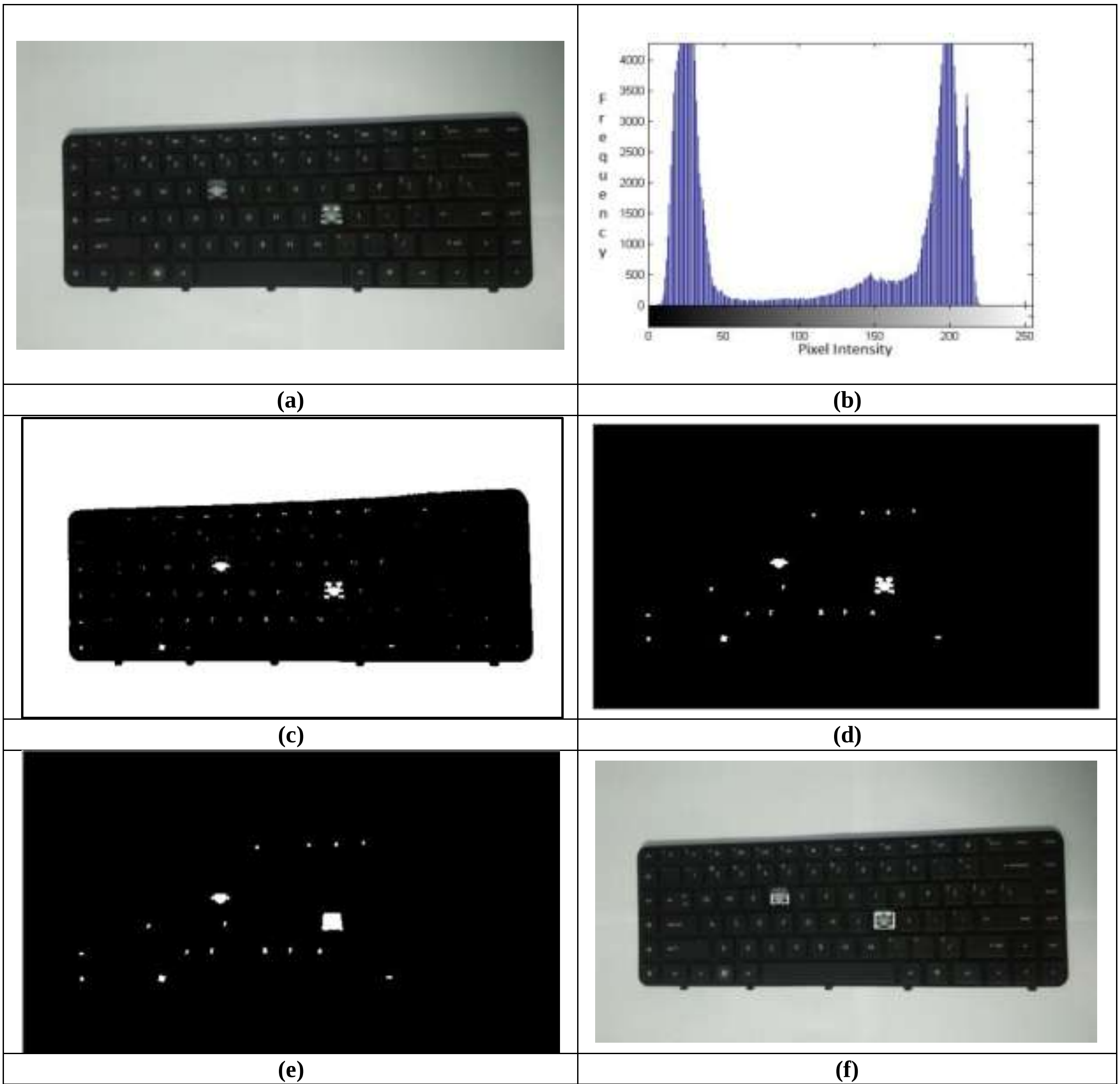


Fig-1: Flow chart of the algorithmic steps



(a) (b) (c) (d) (e) (f)

Fig-2: Algorithmic intermediate steps for defective key detection in keyboard; (a) input keyboard image (b) histogram of input gray scale image (c) image after adaptive threshold segmentation (d) Image after morphological operations (e) Image after connecting disconnected component (f) Defective key detection in keyboard

RESULTS

The proposed algorithm was discussed in previous section. We have implemented this algorithm on various types of keyboards and with different number of defective keys. Available defective keys were successfully detected in all input images. Few results are shown in Figure 3.

With the development of these types of methods of evaluation, automation could be achieved for speed-up the production at industry level. In future, other algorithms might be designed which can detect minor defects also e.g. defect in keyboard annotations, appearance of wrong annotations and crack in any key on keyboard. Human may take time to evaluate these defects and evaluation results would not be so promising. Thus, automation may complete the work as well as may improve the results in very less time compare to human.

Fig-3: Detection of defective keys in different types of keyboards

CONCLUSION

We have proposed an algorithm for detection of defective keys in keyboard. Using different types of keyboard, data-set was generated for validation. Proposed algorithm is working for all types of keyboards to detect number of defective keys. Proposed algorithm is designed in a way that processing could be completed very fast to evaluate the image at manufacturing line and production testing could be completed at same time. We have tested our algorithm with different types of keyboard and found significantly good results.

REFERENCES

1. Unver HO; System Architectures Enabling Reconfigurable Laboratory-Automation Systems, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions, 2011; 41(6):909-922.
2. Lasher B, Narayanan M; Vision systems-an overview. Northcon/93. Conference Record, 1993; 118,121.
3. Lee K, Lee C; Kim S, Kim YH; Fast object detection based on color histograms and local binary patterns, TENCON 2012 - 2012 IEEE Region 10 Conference , 2012; 1-4.
4. Macarron R, Hertzberg RP; High Throughput Screening Assays: Methods and Protocols, 2002.
5. Fricke E, Schuls AP; Design for changeability: Principles to enable changes in systems throughout their entire lifecycle. Syst. Eng. J., 2005; 8(4):342 -359.
6. Alderman E, Elands J; Breaking a screening barrier. A novel approach to ultra-high throughput screening, Genetic Eng. News, 1998; 18:14 -16.
7. Ospina E, Tascon E, Valencia J, Madrigal C; Traffic flow control using artificial vision techniques. Computing Congress (CCC), 2011 6th Colombian , 2011;1-4.
8. Available online at http://www.passmark.com/products/keytest.htm
9. Available online at http://keyboard-tester.software.informer.com
10. Zhang Y, Qu H, Wang Y; Adaptive Image Segmentation Based on Fast Thresholding and Image Merging. Artificial Reality and Telexistence--Workshops, 2006. ICAT '06. 16th International Conference. 2006; 308-311.