

Foreign Element Detection in Chest X-ray Images

by

Fatema Tuz Zohora

B.S., Khulna University of Engineering Technology, Bangladesh, 2010

M.S., University of South Dakota, 2016

A Thesis/Dissertation Submitted in Partial Fulfillment of

the Requirements for the Degree of Master of Science

Department of Computer Science

In the Graduate School The University of South Dakota

December 12, 2016

© 2016 Fatema Tuz Zohora

No part of this work can be reproduced without permission except as indicated by the "Fair Use" clause of copyright law. Passages, images, or ideas taken from this work must be properly credited in any written or published materials.

The members of the Committee appointed to examine
the thesis/dissertation of Fatema Tuz Zohora find it
satisfactory and recommend that it be accepted.

Chairperson, Dr. Quoc-Nam Tran

Dr. Santosh KC

Dr. Erliang Zeng

Abstract

In an automated chest X-ray (CXR) screening (to detect pulmonary abnormalities – Tuberculosis (TB), for instance), the presence of foreign element (e.g., buttons, coins, medical devices) hinders its performance. In this thesis work, I present novel techniques for detecting circle-like element and medical tubes and devices. At first, I focus on detecting circle-like foreign element (both with/without considering the lung region) in the chest X-ray images. I start with pre-processing steps to enhance the CXR images and then detect foreign element. My proposed methods can be categorized in two types: 1) circular assumption-based and 2) training-based. In the first method, candidate selection followed by circular Hough transform (CHT), compute edge images using several different edge detection algorithms and then apply morphological operations to select candidate regions (image segmentation) in the lung region. Finally, CHT is used to detect the circular foreign element. In the second method, I perform the normalized cross-correlation (NCC) followed by an unsupervised clustering. In all tests, both techniques show good performance for a large number of CXR images. I also compare the performance of the proposed techniques with existing methods in the literature (Viola-Jones and CHT). Our methods excelled in performance both in terms of detection accuracy (precision, recall, and F1 score) and computational time. At the output, precision, recall, and F1 score are 90%(96%), 93%(90%), and 91%(92%) from circular assumption-based technique (training-based method). Further, I focus on medical device (with/without tubes) detection in CXRs using training-based, and the results are encouraging.

Keywords: Foreign element detection, chest X-ray (CXR), Medical Tubes, circular Hough transform (CHT), Viola-Jones, Edge Detection, normalized cross-correlation (NCC), unsupervised clustering.

Thesis/Dissertation Advisor: _____

Dr. Santosh KC

Acknowledgement

I would like to express my special gratitude and thanks to my thesis advisor Dr. Santosh KC for his constant instruction, guidance, and encouragement throughout my thesis completion. Without his thoughtful encouragement and careful supervision, this thesis would never have taken shape. I also thank my thesis committee members, Dr. Quoc-Nam Tran and Dr. Erliang Zeng for their contributions to the direction and richness of this work. I am also grateful to Dr. Sameer Antani (Staff Scientist, US National Library of Medicine, NIH), for his guidance during his visit at the USD on 10/17/2016, and of course for the provided dataset (repository, maintained by the US National Library of Medicine (NLM), National Institutes of Health (NIH)). Finally, I extend my deepest thanks to my family who supported and encouraged me every step of the way.

Contents

1	Introduction	1
1.1	Summary	1
1.2	Context and problem	1
1.3	Goal	6
1.4	Contribution	6
1.5	Thesis outline	7
2	Prior work or background	9
2.1	Summary	9
2.2	Foreign element detection in CXRs	9
2.2.1	Circle-like foreign element detection	9
2.2.2	Medical devices (with/without tubes) detection	9
2.3	What's next?	10
3	Circle-like foreign element detection using candidate selection followed by circular Hough Transform in CXR image	12
3.1	Summary	12
3.2	Materials	13
3.2.1	Edge detection	13
3.2.2	Viola-Jones object detection	14
3.2.3	Circular Hough transform (CHT)	15
3.3	Proposed technique	15
3.3.1	Image enhancement	15
3.3.2	Candidate (circle-like elements) selection	18
3.3.3	Finding circle-like elements	20
3.3.4	Annotations	21
3.3.5	Decision	21

3.4	Experiments	24
3.4.1	Dataset	24
3.4.2	Evaluation metrics and protocol	24
3.4.3	Results and analysis	26
3.5	Conclusion	31
3.6	What's next?	32
4	Circle-like foreign element detection using normalized cross-correlation and unsupervised clustering in CXR image	33
4.1	Summary	33
4.2	Materials	34
4.2.1	Normalized cross-correlation	34
4.2.2	Clustering	38
4.3	The proposed technique	40
4.3.1	Image enhancement	41
4.3.2	Annotations	41
4.3.3	Positive training template selection	41
4.3.4	Normalized cross-correlation	41
4.3.5	Negative training template selection	42
4.3.6	Thresholding	43
4.3.7	Training samples: positive and negative	43
4.3.8	Final NCC output	43
4.3.9	Peak detection	44
4.3.10	Decision	44
4.4	Experiments	44
4.4.1	Results and analysis	47
4.4.2	Robustness check	52
4.5	Conclusion	54
4.6	What's next?	55
5	Medical devices (with/without tube) foreign elements detection in chest X-ray	

images using normalized cross correlation based technique	56
5.1 Summary	56
5.2 Proposed technique	57
5.2.1 Decision	59
5.3 Experiments	59
5.3.1 Dataset	59
5.3.2 Evaluation metrics and protocol	62
5.3.3 Results and analysis	62
5.4 Conclusion	63
5.5 What's next?	63
6 Conclusion	64
Appendices	68
A Annotations	69
A.1 Circle-like foreign elements in chest X-ray images	69
A.2 Medical devices in chest X-ray images	69
B Source code	72

List of Figures

1.1	Circle-like foreign element: a) shows CXR images containing circle-like foreign element and b) shows boxes marked by red indicate the location of circle-like foreign element in the images.	3
1.2	Closer view of some of the circle-like elements (see 1.1).	3
1.3	Medical devices in CXRs: a) shows original CXR images containing medical devices and b) shows index line marked by red indicate the location of medical devices CXR images.	4
1.4	Closer view of the medical devices (see 1.3).	4
1.5	Medical devices in CXRs: a) shows original CXR images containing medical tubes and b) shows index line marked by red indicate the location of medical tubes CXR images.	5
1.6	Closer view of selected parts of the medical tubes (see Fig 1.5).	5
3.1	The process begins with image enhancement, using intensity normalization and image adjustment. After that candidate (circle-like element) selection is performed and finally decision is made to decide if the candidates are circle-like elements.	16
3.2	Image Enhancement: a) original image, b) original image histogram, c) enhanced image, and d) enhanced image histogram.	17
3.3	Candidate selection: a), b), c), and d) show results for Prewitt, Sobel, Roberts, and Canny edge detection correspondingly. Steps S1, S2, S3, and S4 correspond to the edge detection, dilation, filling, and erosion.	19

3.4 Decision: e1, e2, e3, and e4 correspond to the Prewitt, Sobel, Roberts and Canny edge images followed by circular object selection steps with CHT. a) and b) show resultant circular object detection intermediate step without and with region of interest (ROI) respectively. Circles marked by green indicate the location of circular shape object on the candidate selection images.	21
3.5 Circle-like element detection with chest segmentation: a), b) and c) show CHT with CS, CHT and Viola-Jones respectively. Here, circles marked by yellow pointed by white arrow indicate false negative, circles marked by red pointed by white arrow indicate false positive and rectangles marked by blue indicate true positive.	22
3.6 Circle-like element detection without chest segmentation in CXR image: a), b) and c) show CHT with CS, CHT and Viola-Jones respectively. Here, circles marked by yellow pointed by white arrow indicate false negative, circles marked by red pointed by white arrow indicate false positive and rectangles marked by blue indicate true positive.	23
4.1 Hierarchical clustering using "bottom up" approach	39
4.2 The process begins with image enhancement. Then perform normalized cross-correlation and hierarchical clustering and finally for automatic detection, it evaluates result by cross-matching with annotated ground-truth and clustered output.	40
4.3 Positive training templates	41
4.4 Negative training templates	41
4.5 Detection with NCC with positive sample without applying thresholding. .	45
4.6 Detection with NCC with positive sample thresholding	46
4.7 Foreign element detection	47
4.8 Detection with NCC: a) shows 3D plot of final NCC coefficient and b) circle-like element detection with final NCC output.	48
4.9 Precision, recall, and F1 score vs SNR (in dB) plot	51

4.10	Detection of NCC with adding Gaussian noise: continue	52
4.11	Detection of NCC with adding Gaussian noise: continue	53
4.12	Detection of NCC with adding Gaussian noise: circles marked by red, rectangle marked by blue, and circle marked by yellow represent true positive, false positive, and false negative respectively.	54
5.1	The process begins with image enhancement. Then perform normalize cross-correlation and hierarchical clustering and finally for automatic detection, it evaluates result by using Dice's coefficient with annotated ground-truth and clustered output.	57
5.2	Positive training templates for medical devices	58
5.3	Negative training templates for medical devices	58
5.4	Positive training templates for medical tubes	59
5.5	Negative training templates for medical tubes	59
5.6	Medical devices detection with positive msk with/without thresholding and final mask.	60
5.7	Medical tubes detection with positive msk with/without thresholding and final mask.	61
A.1	Annotations 2: Circle-like elements in CXR images: a) and b) show original and annotated CXR image respectively.	70
A.2	Annotations 2: Circle-like elements in CXR images: a) and b) show original and annotated CXR image respectively.	71

List of Tables

3.1	Dataset description	24
3.2	Circle-like foreign element detection results for " <u>subset1</u> " (WLS): CHT with CS	26
3.3	Circle-like foreign element detection results for " <u>subset1</u> " (WoLS): CST with CS	26
3.4	Circle-like foreign element detection results for " <u>subset1</u> " (WLS): CHT (without CS)	26
3.5	Circle-like foreign element detection results for " <u>subset1</u> " (WoLS): CHT (without CS)	27
3.6	k-fold cross validation for for " <u>subset1</u> ": Viola-Jones	27
3.7	Circle-like foreign element detection results for " <u>subset1</u> " (WLS, k-fold cross validation): Viola-Jones	27
3.8	Circle-like foreign element detection results for " <u>subset1</u> " (WoLS, k-fold cross validation): Viola-Jones	28
3.9	Circle-like foreign element detection results for " <u>complete data</u> " (WLS): CHT with CS	28
3.10	Circle-like foreign element detection results for " <u>complete data</u> " (WoLS): CST with CS	28
3.11	Circle-like foreign element detection results for " <u>complete data</u> " (WLS): CHT (without CS)	29
3.12	Circle-like foreign element detection results for " <u>complete data</u> " (WoLS): CHT (without CS)	29
3.13	k-fold cross validation for " <u>complete data<td>29</td></u>	29
3.14	Circle-like foreign element detection results for " <u>complete data</u> " (WLS, k-fold cross validation): Viola-Jones	30

3.15	Circle-like foreign element detection results for " <u>complete data</u> " (WoLS, k-fold cross validation): Viola-Jones	30
3.16	Processing time of button detection techniques	30
4.1	Circle-like foreign element detection results for " <u>subset1</u> " (WLS): NCC with unsupervised clustering	49
4.2	Circle-like foreign element detection results for " <u>subset1</u> " (WoLS): NCC with unsupervised clustering	49
4.3	Circle-like foreign element detection results for " <u>complete data</u> " (WLS): NCC with unsupervised clustering	49
4.4	Circle-like foreign element detection results for " <u>complete data</u> " (WoLS): NCC with unsupervised clustering	49
4.5	Circle-like foreign element detection result for " <u>complete data</u> " (WLS, k-fold cross validation): NCC with unsupervised clustering	49
4.6	Circle-like foreign element detection result for " <u>complete data</u> " (WoLS, k-fold cross validation): NCC with unsupervised clustering	50
4.7	Comparison of circle-like foreign element detection result for " <u>subset1</u> " (WLS)	50
4.8	Comparison of circle-like foreign element detection result for " <u>subset1</u> " (WoLS)	50
4.9	Comparison of circle-like foreign element detection result for the " <u>complete data</u> " (WLS)	50
4.10	Comparison of circle-like foreign element detection result for the the " <u>complete data</u> " (WoLS)	51
4.11	Circle-like foreign element robustness check (adding Gaussian noise with variable SNR) for the " <u>subset1</u> " (WoLS) (groundtruth-166): NCC with unsupervised clustering	51
5.1	Dataset description	62
5.2	Medical devices with/without tubes detection result for " <u>subset1</u> ": NCC based technique	62

CHAPTER 1

Introduction

1.1 Summary

In an automated CXR screening (to detect pulmonary abnormalities – Tuberculosis (TB), for instance), the presence of foreign element (circle-like element-buttons, coins, medical devices (with/without tubes) hinders its performance (Santosh, Vajda, Antani, and Thoma 2016). In this chapter, I present the motivation of solving this problem, detailed explanation of foreign element in CXR images and complexity of the problem. After that, I explain the problem solving way to reach the goal and my contribution.

Key topics:

- Motivation of the research work
- Set up goal
- Contribution to foreign element in CXR images

Organization of chapter. The rest of this thesis work is structured as follows: In Section 1.2, I will give detailed explanation of context and problem of our thesis work. After that, I will explain the goal of my work in Section 1.3. Then, I will discuss my contribution towards to solving this problem in Section 1.2. Finally, Section 1.5 describes what is in the next chapters.

1.2 Context and problem

Lung diseases are major threats because significant numbers of people suffer from these diseases such as tuberculosis (WHO 2014), pneumonia (Van Ginneken, Romeny, and Viergever 2001), lung cancer (Firmino, Morais, Mendoça, Dantas, Hekis, and Valentim 2014) (Firmino, Angelo, Morais, Dantas, and Valentim 2016) (Abe, Hanai, Nakano, Ohkubo, Hasizume, Kakizaki, Nakamura, Niki, Eguchi, Fujino, et al. 2005) (Chan, Hadji-

iski, Zhou, and Sahiner 2008) and pulmonary edema (El-Baz, Beach, Gimel'farb, Suzuki, Okada, Elnakib, Soliman, and Abdollahi 2013) across the world. The advent of new powerful hardware and software techniques has triggered attempts to develop computer-aided diagnostic systems for automatic chest X-ray screening (Antani 2015) (Schaefer-Prokop, Neitzel, Venema, Uffmann, and Prokop 2008) (Santosh, Vajda, Antani, and Thoma 2016) (Karargyris, Siegelman, Tzortzis, Jaeger, Candemir, Xue, Santosh, Vajda, Antani, Folio, et al. 2016) (Jaeger, Karargyris, Candemir, Folio, Siegelman, Callaghan, Xue, Palaniappan, Singh, Antani, et al. 2014) (Jaeger, Karargyris, Candemir, Siegelman, Folio, Antani, Thoma, and McDonald 2013). Foreign elements such as button, coin, medical tube, medical device within the chest X-ray images hinder the performance of automatic screening system. Circle-like foreign element such as buttons can be found on medical gown that patients are wearing or coins/buttons can be mistakenly swallowed by patients. Another kind of foreign element i.e., medical devices (with/without tube) can be found by inserting patients chest due to drain air, blood, or fluid from the pleural space. Pleural space which is the space surrounding the lungs. patients and chest tube (medical tube/devices) insertion due to drain air, blood, or fluid from the pleural space, which is the space surrounding the lungs, within the chest X-ray images hinder the performance of the automatic screening process. Fig. 1.1 shows two such chest X-ray images which contain several circle-like foreign elements and Fig. 1.2 shows a closer view of all the circle- like foreign elements in these CXRs. Fig. 1.3 shows another such chest X-ray image that contains medical devices in CXR image and it shows the complexity of the problem. In Fig. 1.4 shows a closer view of the medical devices in those of CXRs. In Fig. 1.5 shows another example of such chest X-ray image that contains medical tubes in CXR image and it also shows the complexity of the problem. Fig. 1.6 shows a closer view of the medical devices in these CXRs.

The presence of such foreign elements hinders the automatic screening system performance (Jaeger, Karargyris, Candemir, Folio, Siegelman, Callaghan, Xue, Palaniappan, Singh, Antani, et al. 2014), as they are not due to any lung abnormalities and therefore should not be considered (Karargyris, Siegelman, Tzortzis, Jaeger, Candemir,

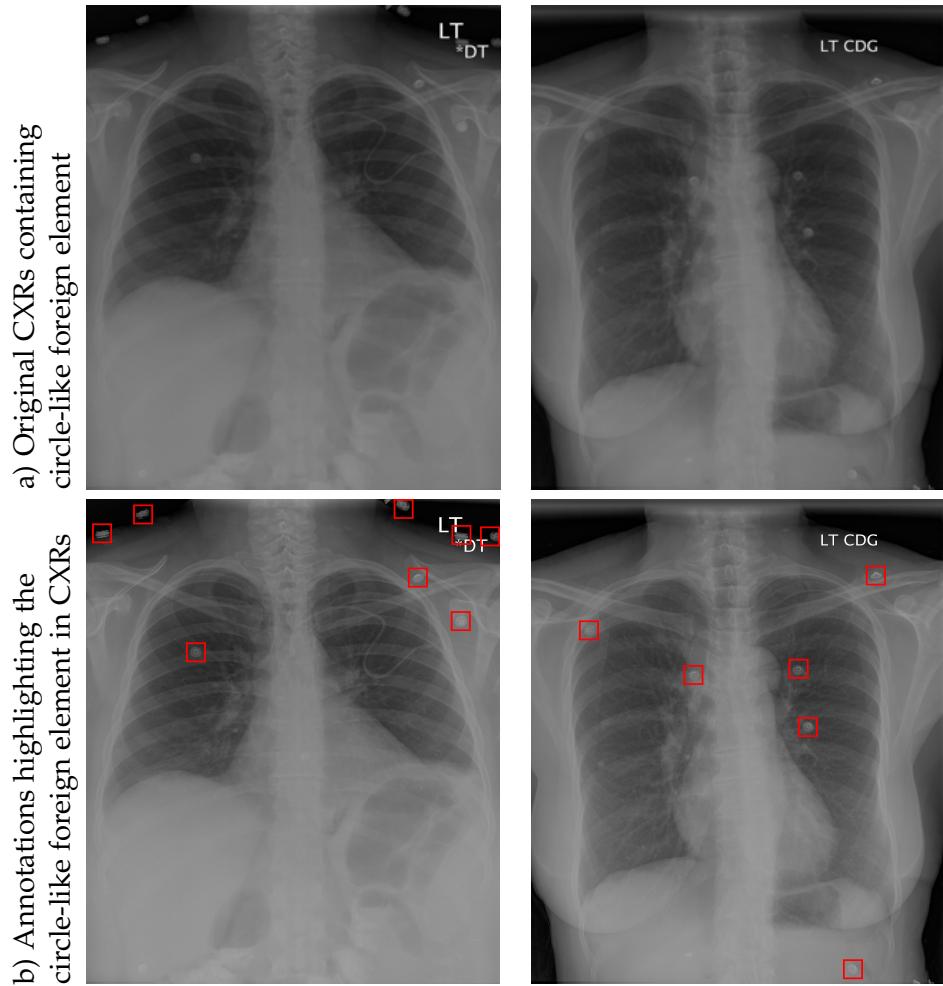


Figure 1.1: Circle-like foreign element: a) shows CXR images containing circle-like foreign element and b) shows boxes marked by red indicate the location of circle-like foreign element in the images.

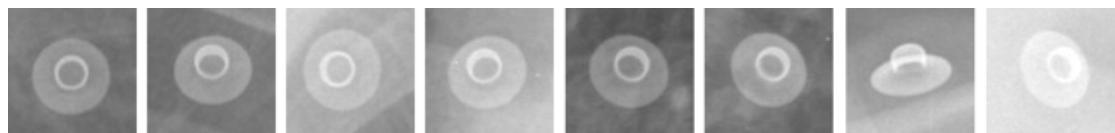


Figure 1.2: Closer view of some of the circle-like elements (see 1.1).

Xue, Santosh, Vajda, Antani, Folio, et al. 2016). Therefore, in the screening process precise detection of foreign element is an important issue for screening of chest diseases in CAD systems.

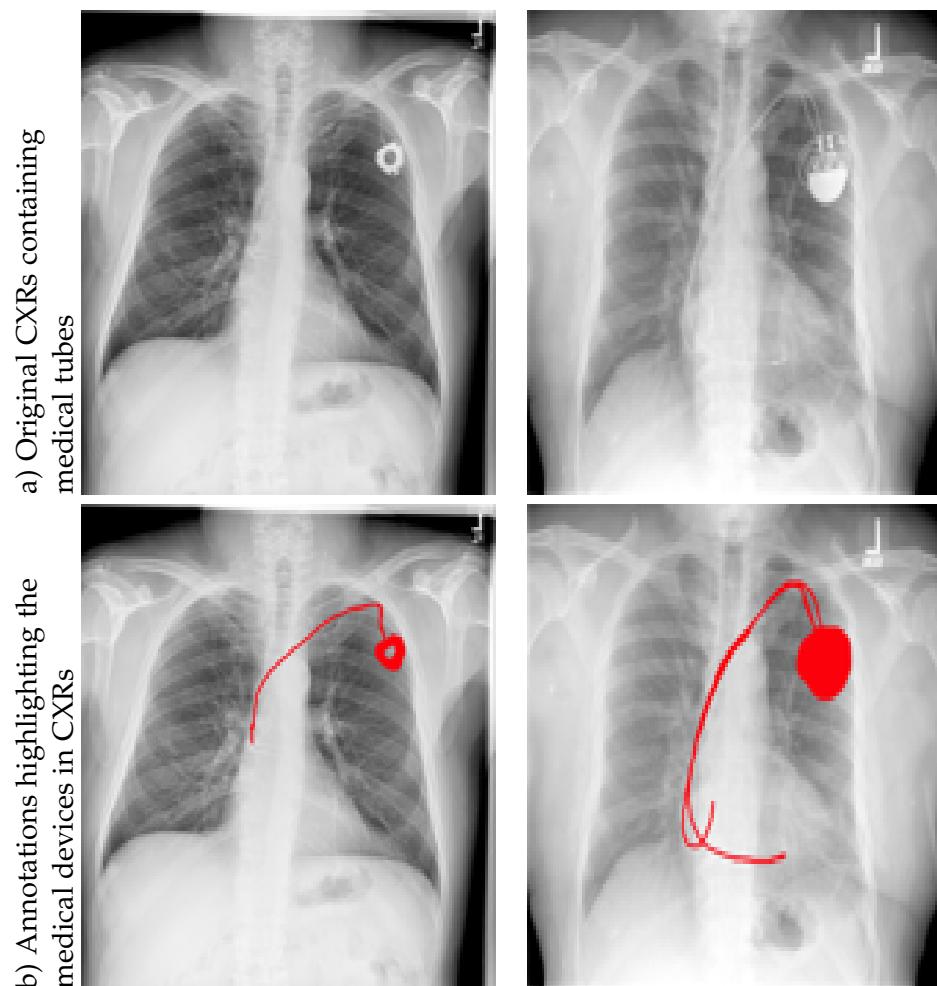


Figure 1.3: Medical devices in CXRs: a) shows original CXR images containing medical devices and b) shows index line marked by red indicate the location of medical devices CXR images.

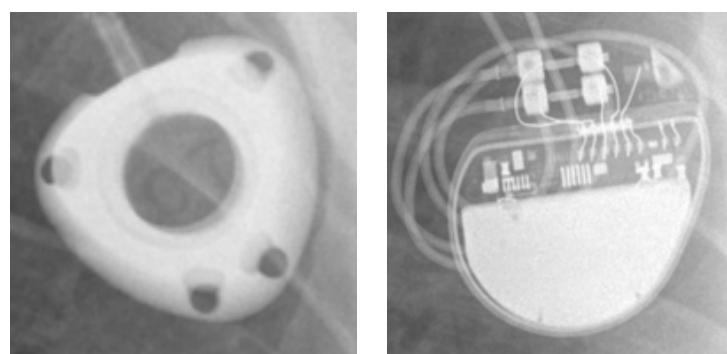


Figure 1.4: Closer view of the medical devices (see 1.3).

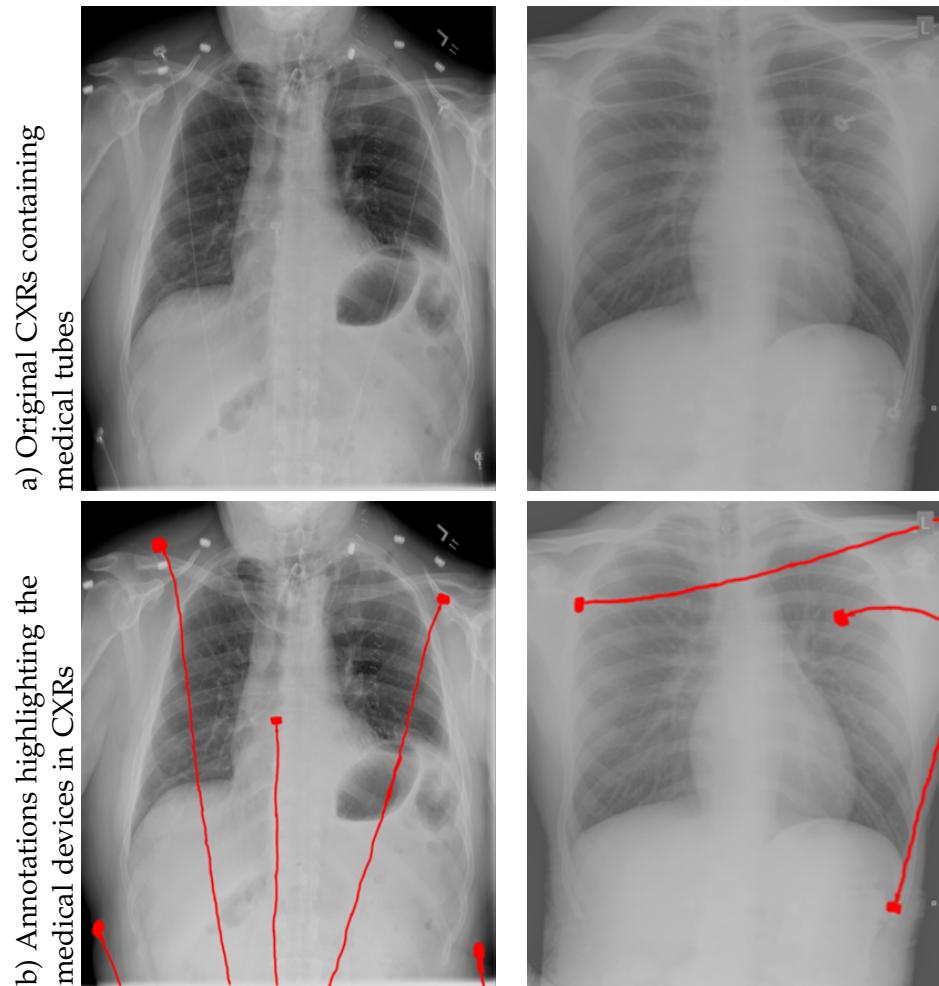


Figure 1.5: Medical devices in CXRs: a) shows original CXR images containing medical tubes and b) shows index line marked by red indicate the location of medical tubes CXR images.

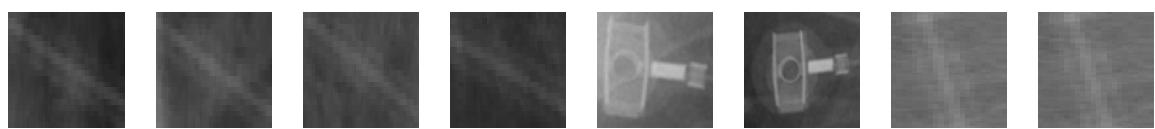


Figure 1.6: Closer view of selected parts of the medical tubes (see Fig 1.5).

1.3 Goal

The main goal of my thesis work is to detect foreign element (circle-like element and medical devices (with or without tubes)) in CXRs by using image processing techniques. This problem has two different type of foreign elements; one of which is circle-like foreign element such as buttons/coins and the other is medical devices (with/without tubes) in CXRs. The pattern of the coins/buttons is circle/elliptical like except some irregularities. In my thesis work, at first, I detected the circle-like foreign element in CXRs. And the second problem is which is complex by nature because medical devices with or without tubes have irregular patterns. For detecting the foreign elements,

- we first analyze the foreign element (coin, button, round element, and medical devices (with/without tube)) characteristics.
- We collect the data set of chest X-ray images including these foreign elements.
- We study the literature.
- We implement and test the existing work.
- We implement our own techniques for detecting the foreign element in the CXRs.
- We compare our techniques with the existing significant work.

1.4 Contribution

I have introduced two methods for detecting circle-like foreign elements and medical devices (with/without tubes) in chest X-ray images including image enhancement. In my first method, circle-like foreign element detection using candidate selection followed by circular Hough Transform in CXR image, I perform edge detection with morphological operation to segment the circle-like candidate. Then, I apply CHT to find the circle-like elements in CXRs. This novel technique is able to excel performance in terms of detection accuracy precision, recall and F1 score for medical devices. This contribution has been published in Communications in Computer and Information Science (CCIS) series, Vol.709, Springer, which was selected from the International Conference on Recent Trends

in Image Processing Pattern Recognition (RTIP2R), Dec. 16 - 17, 2016.

In my second training based technique, circle-like foreign element detection using normalized cross-correlation and unsupervised clustering in CXR image, I perform normalized cross-correlation (using positive and negative template) and unsupervised clustering (hierarchical clustering) for foreign element detection within CXR images using those enhancement CXR images. Finally, I automate our system by annotated my test CXR images for measuring the performance. In all tests, my algorithm performed well under a variety of CXRs. I also compare our proposed technique's performance with all of the benchmarking technique in literature. This novel technique is able to outperform in terms of detection measures precision, recall and F1 score. In circular assumption based technique, we receive 90%, 93% and 91% precision, recall and F1 score for total dataset within lung region. And for our training based technique, I receive 96%, 90% and 92% precision, recall and F1 score within or without lung segmentation regardless of dataset.

For my second problem, I focus on medical devices (with/without tubes) detection in CXRs. I present normalized cross-correlation based technique for detecting medical devices (with/without tube) in CXR images. At first, I detect medical devices without tube and measure detection performance using Dice's coefficient and then calculate precision, recall and F1 score. After that, I perform this technique on medical tubes also.

1.5 Thesis outline

The rest of our thesis is structured as follows:

- In chapter 2, I describe the state of the art of foreign element detection in CXR images including both circle-like element and medical devices.
- In chapter 3, I implement circle-like foreign element detection using candidate selection followed by circular Hough Transform in CXR image.
- In chapter 4, I perform circle-like foreign element detection using normalize cross correlation and unsupervised clustering in CXR image.

- In chapter 5, I apply normalize cross correlation based technique for medical devices with/without tubes in CXR images.
- Finally, I conclude my thesis work on chapter 6.
- In Appendices A.1 and A.2, I present annotations for circle-like foreign element and Medical devices in CXR images respectively.

CHAPTER 2

Prior work or background

2.1 Summary

In this chapter, I explain foreign elements detection in chest X-ray images related benchmarking work in the literature including circle-like foreign element and medical devices(with/without tubes).

Organization of chapter. The rest of the chapter is structured as follows: In Section 2.2, I give detailed explanation of existing work in the literate related to foreign elements detection in chest X-ray images. Finally, Section 2.3 describes what is in the chapter two.

2.2 Foreign element detection in CXRs

In the CXRs, several different type of foreign objects may appear. Such objects can be classified into two categories: i) circle-like objects and ii) medical devices.

2.2.1 Circle-like foreign element detection

Not surprisingly researchers have already put their interest in foreign element (circle-like such as, button, coin) detection (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015). Their system is fairly straight forward because they are limited to round object (e.g., coins, buttons) as shown in 1.1. They mainly focused only chest region in CXR image. Using Viola-Jones algorithm for detecting foreign element in CXR image has high false detection. For improving the performance of foreign element detection in CXR image, I have to focus on new algorithm.

2.2.2 Medical devices (with/without tubes) detection

In the literature, there is no existing research work related to detect medical devices in CXRs. But diagnosing chest radio-graphs accurately, I have to do some more necessary works for detecting medical devices (with/without tubes) in chest X-ray im-

ages.

2.3 What's next?

The rest of my thesis work is structured as follows:

- In chapter 3, circle-like foreign element detection using candidate selection followed by circular Hough Transform in CXR image is structured as follows: In section 3.2, I give detailed information of our all performed experiment's material. In section 3.3, I give detailed overview of our proposed technique. At first, I briefly discuss the image enhancement (see subsection 3.3.1) and then I provide a detailed description of the circular object detection process (see subsection 3.3.2). After that, I give brief description of finding circle-like element in Section 3.3.3 and then in subsection 3.3.5, I explain our decision taking technique. Next, Section 3.4 provides information about the data set and the evaluation protocol (see Section 3.4.1) and results (see Section 3.4.3). Section 3.5 concludes the paper. Finally, Section 3.6 describes what is the next chapter.
- In chapter 4, circle-like foreign element detection using normalize cross correlation and unsupervised clustering in CXR image is structured as follows: In section 4.2, I give detailed information of our all performed experiment's material. In section 4.3, I give detailed overview of our proposed technique. In subsection 4.2.1, I compute normalization cross-correlation with positive and negative template, after that create mask for finding peak using hierarchical clustering. Then in subsection 4.3.10, I explain our decision taking technique. Next, Section 4.4 provides information about the dataset and the evaluation metrics and protocol for performance measurement and results (see Section 4.4.1). Section 4.5 concludes the paper. Finally, Section 4.6 describes what is the next chapter.
- In chapter 5, medical devices (with/without tube) like foreign elements detection in chest X-ray images using normalize cross correlation based technique is structured as follows: In section ??, I give detailed information of our all performed experiment's material. In section 5.2, I give detailed overview of our proposed technique.

In subsection 4.2.1, I compute normalization cross-correlation with positive and negative template, after that create mask for finding peak using hierarchical clustering. Then in subsection 5.2.1, I explain our decision taking technique. Next, Section 5.3 provides information about the dataset and the evaluation metrics and protocol for performance measurement and results (see Section 5.3.3). Section 5.4 concludes the paper. Finally, Section 5.5 describes what is the next chapter.

- In chapter 6, I conclude my thesis work.

CHAPTER 3

Circle-like foreign element detection using candidate selection followed by circular Hough Transform in CXR image

3.1 Summary

In automated chest X-ray screening (to detect i.e, Tuberculosis for instance), the presence of foreign objects (buttons, medical devices) hinders it's performance. In this chapter, we present a new technique for detecting circle-like foreign objects, in particular buttons, within chest X-ray (CXR) images. In our technique, we use a pre-processing step that enhances the CXRs. Using these enhance images, we find the edge images performing four different edge detection algorithms (Sobel, Canny, Prewitt, and Roberts) and after that, we apply some morphological operations to select candidates (image segmentation) in the chest region. Finally, we apply circular Hough transform (CHT) to detect the circular foreign objects on those images. In all tests, our algorithm perform well under a variety of CXRs. We also compare our proposed technique's performance with existing techniques in literature (Viola-Jones and CHT). Our techniques are able to excel performance in terms of both detection accuracy and computation time.

Key topics:

- Image enhancement
- Perform Viola-Jones, circular Hough transform
- Perform couple of morphological operations to select candidates (image segmentation) followed by CHT
- Results, comparative study, and analysis with existing benchmarking techniques

Organization of chapter. The rest of the chapter is structured as follows: In section 3.2, we give detailed information of our all performed experiment's material. In section 3.3,

we give detailed overview of our proposed technique. At first, we briefly discuss the image enhancement (see subsection 3.3.1) and then we provide a detailed description of the circular object detection process (see subsection 3.3.2). After that, we give brief description of finding circle-like element in Section 3.3.3 and then in subsection 3.3.5, we explain our decision taking technique. Next, Section 3.4 provides information about the data set and the evaluation protocol (see Subsection 3.4.1) and results (see Subsection 3.4.3). Section 3.5 concludes the paper. And, section 3.6 tells what is next.

3.2 Materials

In this section, we explain detailed information which is related to our experiments. At first, we explain briefly four edge detection algorithms, after that we introduce Viola-Jones object detection algorithm and its steps. Finally, we explain circular Hough transform, for finding circle-like candidate detection.

3.2.1 Edge detection

In our work, we apply four edge detection algorithms for finding the high gradient pixel value in CXRs. They are given below.

- Canny
- Sobel
- Prewitt
- Roberts

Brief explanation of those four edge detection algorithms are described next.

Canny. This edge detector (Canny 1986) compute edges by looking for local maxima of the gradient of image. It uses derivative of a Gaussian filter for calculating the gradient. This technique uses high and low edges, considering weak edges in the output if they are connected to strong edges. It has five individual steps:

1. Smoothing: At first, blurring is applied to input image for removing the noise.

2. Finding gradients: It performs four filters for detecting horizontal, vertical and diagonal edges in the blurred image for observing the most prominent magnitudes. It gives a value for the first derivative in the horizontal direction (G_x) and the vertical direction (G_y). From this the edge gradient and direction can be determined:

$$\mathbf{G} = \sqrt{\mathbf{G}_x^2 + \mathbf{G}_y^2} \quad \text{and} \quad \Theta = \text{atan2}(\mathbf{G}_y, \mathbf{G}_x). \quad (3.1)$$

3. Non-maximum suppression: Only local maxima should be marked as edges.
4. Double thresholding: Potential candidates are determined by applying two consecutive thresholding. Finally, it is applied hysteresis for edge tracking.

Sobel. This operator (Sobel 2014) is applied in any input gray scale image to find the approximate absolute gradient magnitude at each point. It is convoluted with original input image for calculating horizontal and vertical changes.

Prewitt. This operator (Prewitt) is similar to Sobel operator. It detects horizontal and vertical gradient components. By taking difference between corresponding pixel intensities of an image. The Prewitt kernels can be decomposed as the products of an averaging and a differentiation kernel, they compute the gradient with smoothing.

Roberts. As a differential operator, Roberts cross operator (Lawrence 1963) performs 2-D spatial gradient measurement on an input image. It has two convolution kernels where one kernel is 90° rotation of another kernel.

3.2.2 Viola-Jones object detection

Viola-Jones (Viola and Jones 2001) is a popular object detection algorithm. It has four main steps: i) Haar feature selection, ii) integral image creation for faster computation of the Haar features, iii) training classifier, built on the AdaBoost learning algorithm, and iv) the combination of classifiers in a cascade architecture, which allows background regions to be discarded quickly.

In the training stage, user provides a set of positive and negative samples of the desired objects. In the testing stage, the detector scans across the image at multiple scales and locations and finds the sub-windows that contain the objects. The detector also combines the overlapping multiple detections of one object into a single detection of the object.

3.2.3 Circular Hough transform (CHT)

The circular Hough transform (CHT) is a specialization of Hough transform, with the purpose of finding circles in imperfect image inputs. The first step for this algorithm is to extract edges in the image. Then for each edge point (x, y) a circle is defined in the accumulator space centered at (x, y) with a fixed radius R . The intersection point of all such circles is a maxima in the accumulator space and corresponds to the center point of the original circle that the edge points belong to.

In our experiments, we use the size invariant circle detection method presented in (Atherton and Kerbyson 1999). This method uses a combination of several modifications to the CHT, including the use of edge orientation, concurrent consideration of a span of circle radii, and using a compound accumulator array that has the phase corresponding to the log of the radius.

3.3 Proposed technique

In our proposed algorithm, we first enhance the CXRs to increase the contrast between the button objects with their background, using intensity normalization and image adjustment. Next, we apply candidate selection (CS)/circular shape enhancement step to make the circular candidates more distinct and then we perform Circular Hough Transform (CHT) for extracting circular shape objects (e.g., buttons) in these images. For improving the performance, we segment the chest area to identify the region of interest. Fig. 3.1 briefly presents the proposed method workflow.

3.3.1 Image enhancement

We apply a couple of pre-processing steps on the heterogeneous chest x-ray images to improve the image quality, so that objects of interest (buttons) become more ev-

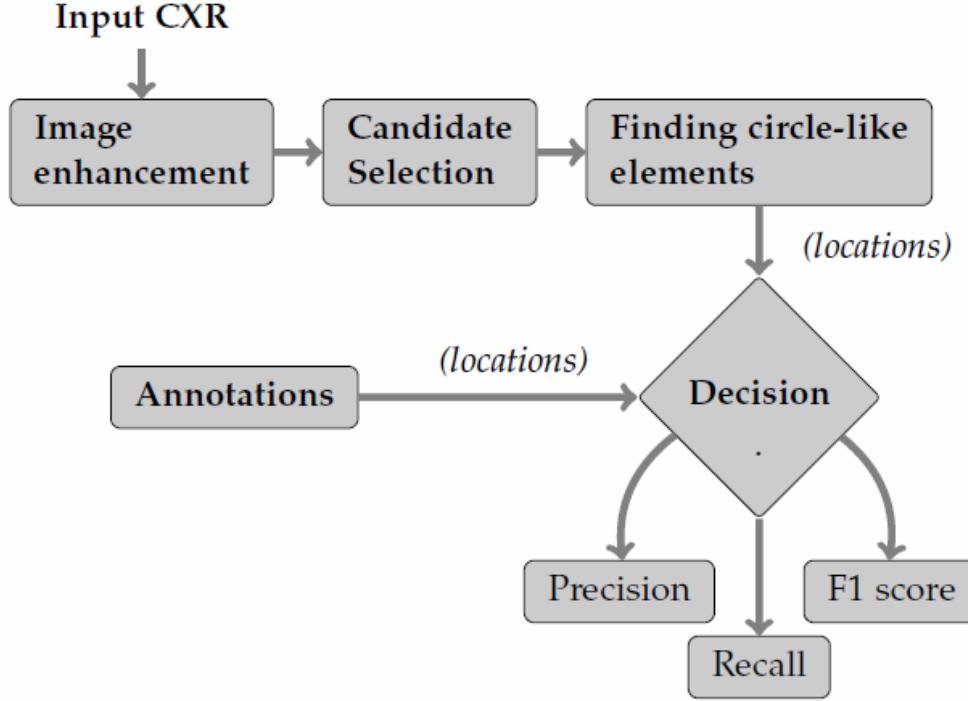


Figure 3.1: The process begins with image enhancement, using intensity normalization and image adjustment. After that candidate (circle-like element) selection is performed and finally decision is made to decide if the candidates are circle-like elements.

ident. The quality of the pre-processing steps strongly affects the performance of the subsequent button detection steps.

In our tests, we apply two pre-processing steps: intensity normalization and image adjustment. Those two steps explain below.

- *Intensity normalization.* In our tests, we use DICOM images as raw input. In DICOM header, we can find window information, which is intensity window manually optimized by radiologists to visually enhance the lung tissue region. If the windowing information is available, then we set the minimum and maximum pixel values of the input CXRs using the following equations:

$$I_{min} = I_{wc} - \frac{I_{ww}}{2} \quad \text{and} \quad I_{max} = I_{wc} + \frac{I_{ww}}{2}, \quad (3.2)$$

where I_{wc} is the value of the window center and I_{ww} is the value of the window width. All pixel values lower than I_{min} are set to I_{min} , and all pixel values higher

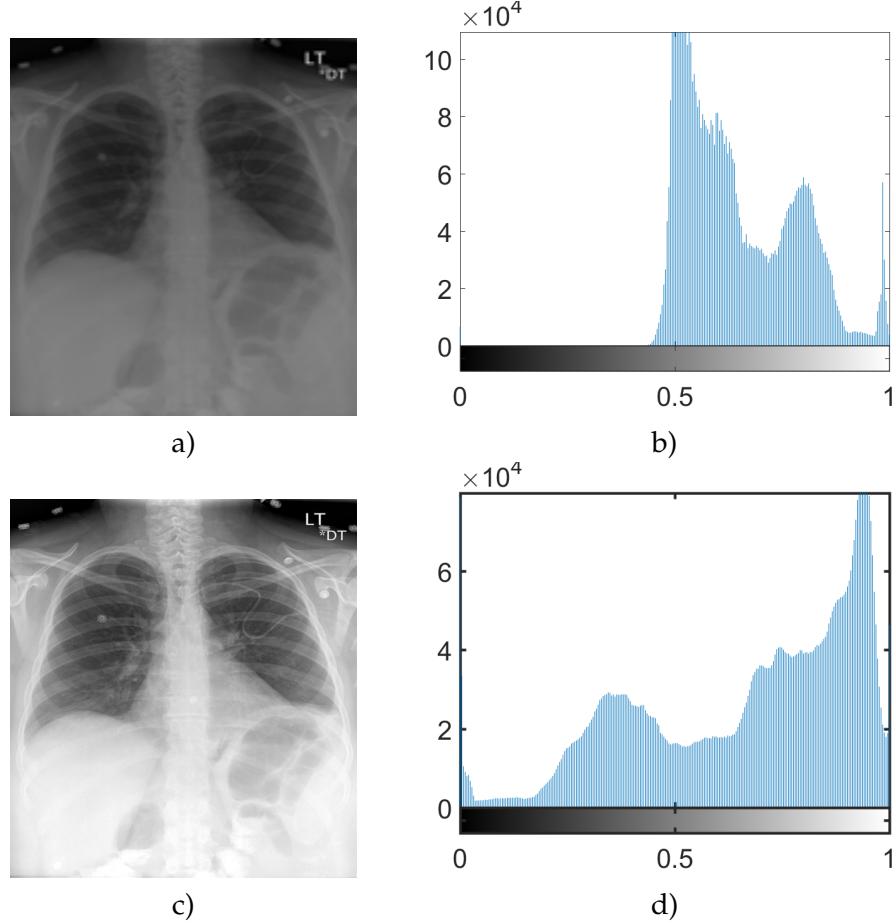


Figure 3.2: Image Enhancement: a) original image, b) original image histogram, c) enhanced image, and d) enhanced image histogram.

than I_{max} are set to I_{max} . After doing this, we normalize the interval to the range $[0,1]$. If there is no window information available, then we calculate the minimum and maximum value of all pixels as I_{min} and I_{max} .

- *Image adjustment.* Next, we adjust the contrast of the CXRs, by mapping the intensities to new values such that the lowest and highest intensities of the image are saturated at 1% of the pixels' intensities. Finally, We check photometric interpretation of the image whether it is monochrome1 or not, if it is monochrome1 then we invert the pixel values. Figure 3.2 shows one such low contrast input and the resultant image after enhancement. It also shows the histogram of pixel intensities before and after the adjustment. It can be seen from the figure that the adjusted image has more uniformly distributed intensity values and a higher contrast. (Xue,

Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015) also follows similar approach for image enhancement. For images without any windowing information, (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015) performs adjustment only for low contrast images. We, however, observe that applying such adjustments always improve performance. Hence, we apply it as a common pre-processing step to all images.

For intensity normalization, we use the same approach as in (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015). For image adjustment; however, we adopt slightly different approach than (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015). Fig. 3.2 shows one such low contrast input and the resultant image after enhancement. It also shows the histogram of pixel intensities before and after the adjustment. It can be seen from the figure that the adjusted image has more uniformly distributed intensity values and a higher contrast.

3.3.2 Candidate (circle-like elements) selection

We observe that the buttons in the lung region are mostly of circular/elliptical shape and their boundaries are sharper than other areas after analyzing the input CXRs. Based on these facts, we extract circular shape object from CXRs. In our proposed technique, we apply edge detection and then perform some morphological operations on these edge-images to enhance the circular candidates for candidate selection.

Edge detection. We apply edge detection to extract high gradient regions in the image. For this, we test with four edge detection operators: Canny (Canny 1986), Prewitt (Prewitt), Sobel (Sobel 2014), and Roberts (Lawrence 1963). Fig. 3.3 S1 represents the four resultant edge-images.

Now, we apply several morphological operations (Soille 2013) on the edge-images to segment and enhance all distinct objects in the image.

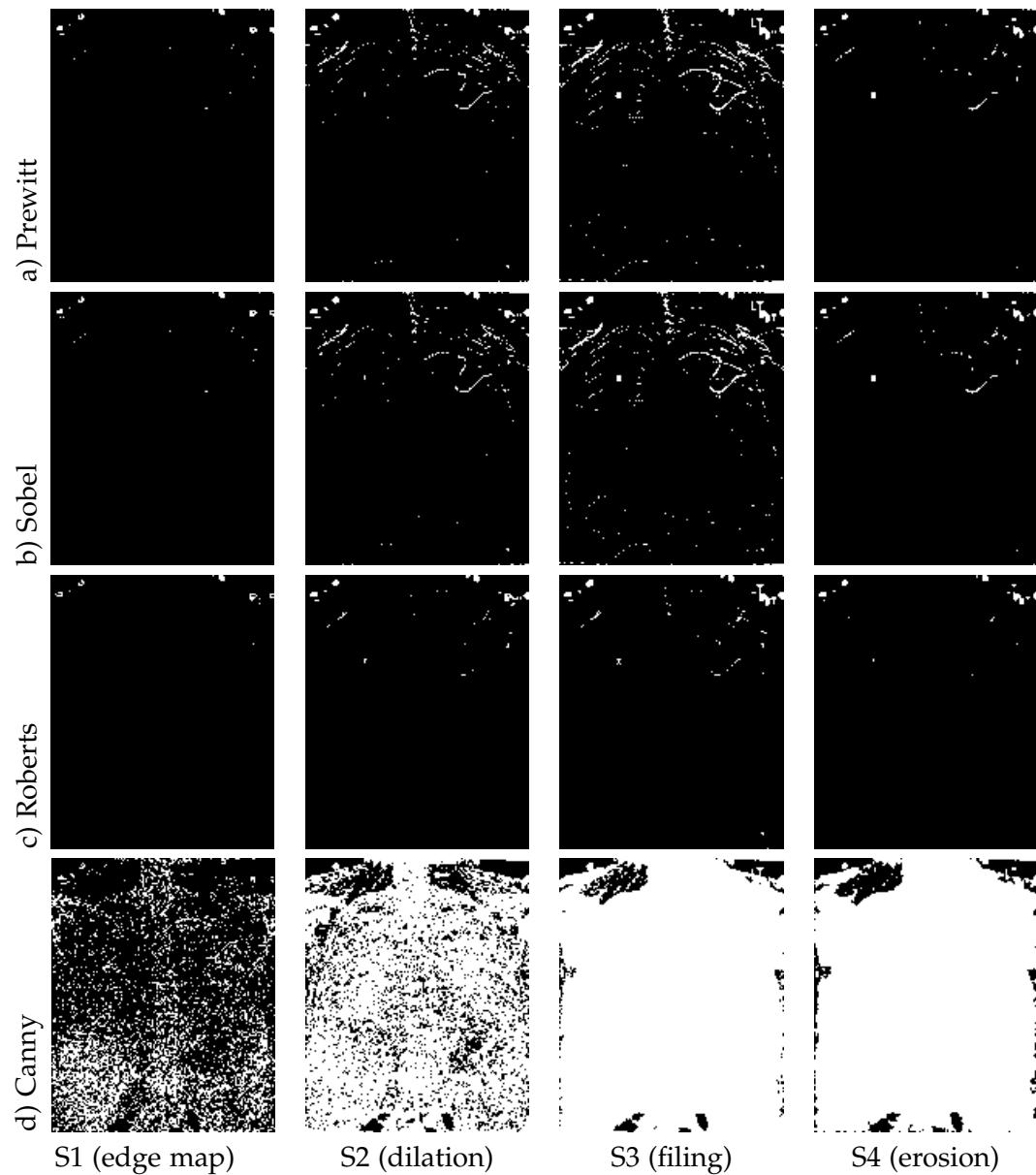


Figure 3.3: Candidate selection: a), b), c), and d) show results for Prewitt, Sobel, Roberts, and Canny edge detection correspondingly. Steps S1, S2, S3, and S4 correspond to the edge detection, dilation, filling, and erosion.

Dilation. We first threshold the edge images to create binary edge masks. Such binary edge masks usually mark the object boundaries in the image. However, due to noise or low background to foreground contrast, the edge mask, in general, is not continuous and hence the corresponding object boundaries have small gaps in between. To overcome this issue, we dilate the edge mask using line-shaped structuring elements. We use two linear structuring element **Sl1** and **Sl2** of length 3, oriented at angles 90° and 0° respectively:

$$\mathbf{Sl1} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \mathbf{Sl2} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \quad \text{and} \quad \mathbf{Sd} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Filling. In this step, we first label all the connected objects in the dilated image. Next, for each such object we perform a flood-fill operation that fills the holes in the objects and generates objects with solid interior.

Erosion. In order to remove noisy small object from the edge images, next we erode the image. For erosion, we use a diamond-shaped structuring element **Sd**. In this step, we also suppress structures that are connected to the image border. After these steps, we apply CHT for button extraction. Fig. 3.3 S1-S4 shows the step-wise outputs of the circular object enhancement process.

3.3.3 Finding circle-like elements

In our experiments, we use the size invariant circle detection method (Atherton and Kerbyson 1999) for finding circle-like element on candidate selection image with and without lung segmentation.

Our main goal is to detect circular shape objects in the lung region of the CXRs, since that is the main area of interest for detecting chest diseases in the CAD systems. The accuracy of the lung segmentation has a big impact on the overall performance of the system. For lung segmentation, we apply anatomical atlases with nonrigid registration algorithm (Candemir, Jaeger, Palaniappan, Musco, Singh, Xue, Karargyris, Antani, Thoma, and

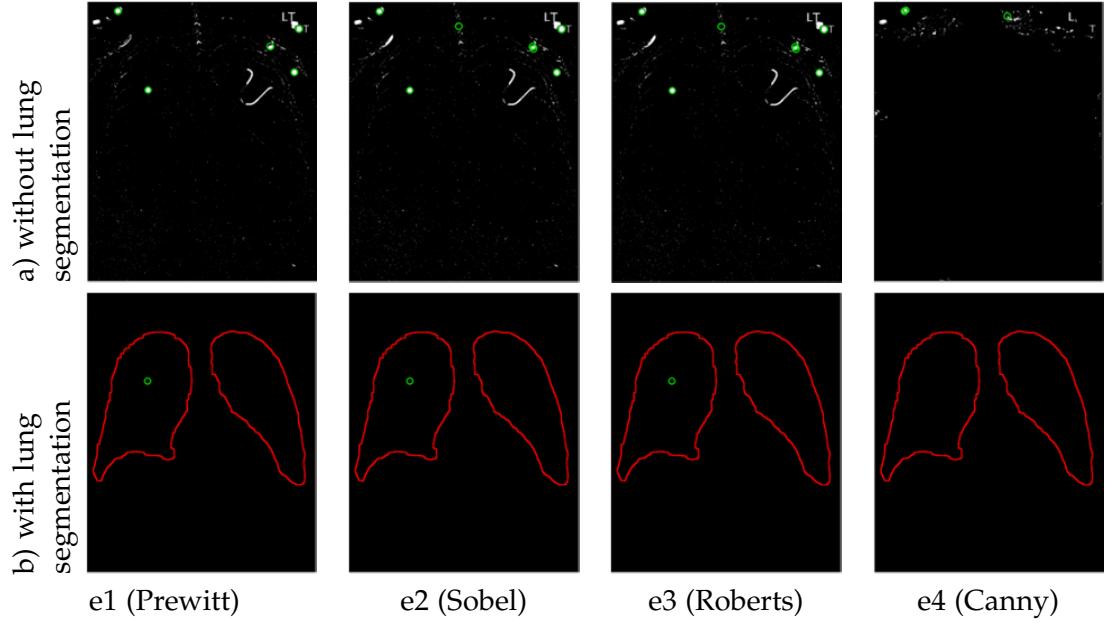


Figure 3.4: Decision: e1, e2, e3, and e4 correspond to the Prewitt, Sobel, Roberts and Canny edge images followed by circular object selection steps with CHT. a) and b) show resultant circular object detection intermediate step without and with region of interest (ROI) respectively. Circles marked by green indicate the location of circular shape object on the candidate selection images.

McDonald 2014). In Fig. 3.4, shows the intermediate resultant circular objects detection on candidate selection image with and with out lung segmentation.

3.3.4 Annotations

For Annotations of cricle-like elements in CXR images please see Appendix A.1.

3.3.5 Decision

After finding circle-like elements in CXRs, we take their locations for computing our final decision. We find the Euclidean distance of those locations with our annotated ground-truth locations. If the distance is less than a specific radius then we consider that position is circle-like elements position of true positive or true detection otherwise we consider it as a false detection (not a circle-like element). For computing non detected circle-like elements, we subtract our annotated ground-truth with true positive or truly detected circle-like elements in CXR image. Figs. 3.5 and 3.6 a), show the complete view of circular foreign object detection on candidate selection image with and without lung

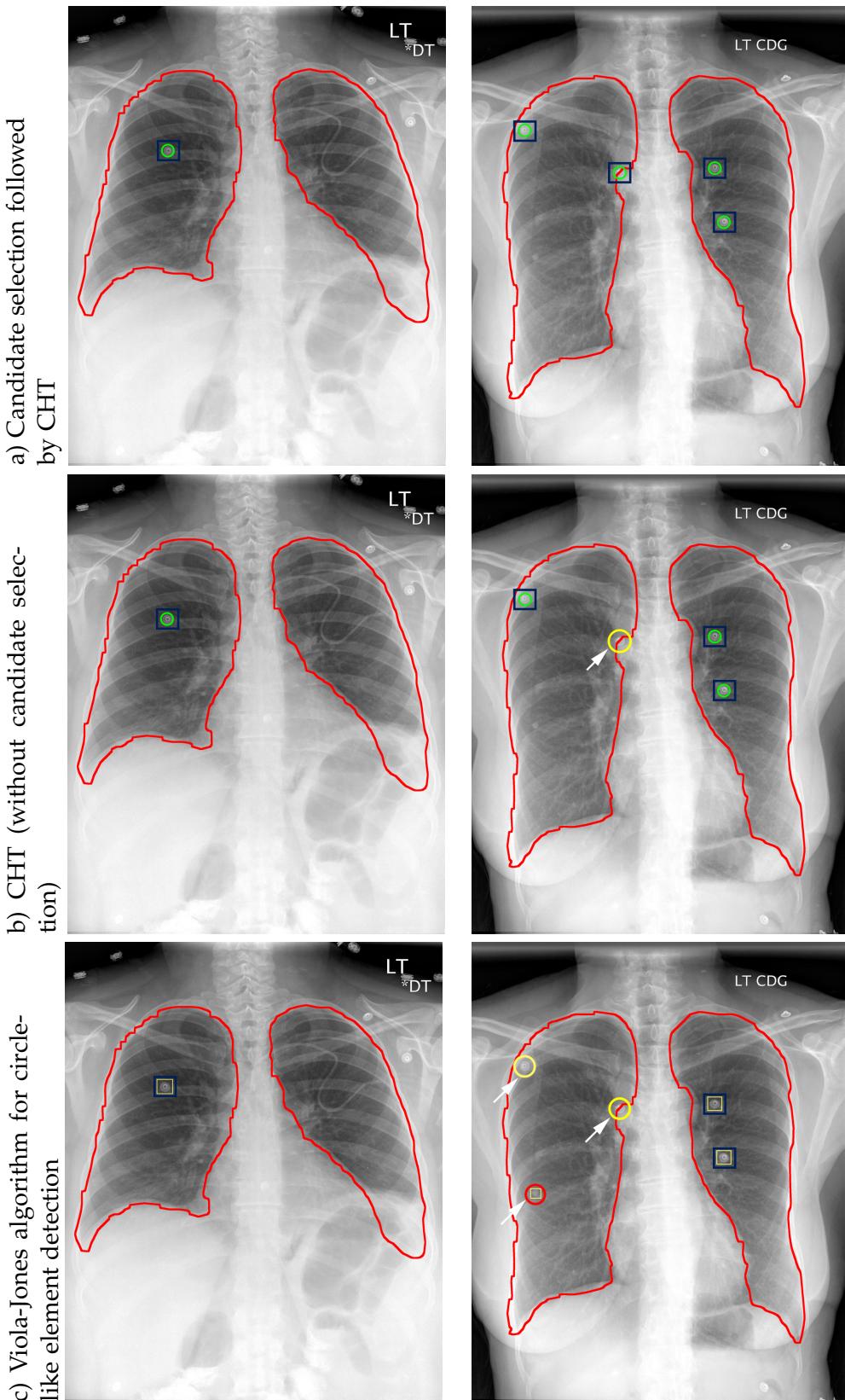


Figure 3.5: Circle-like element detection with chest segmentation: a), b) and c) show CHT with CS, CHT and Viola-Jones respectively. Here, circles marked by yellow pointed by white arrow indicate false negative, circles marked by red pointed by white arrow indicate false positive and rectangles marked by blue indicate true positive.

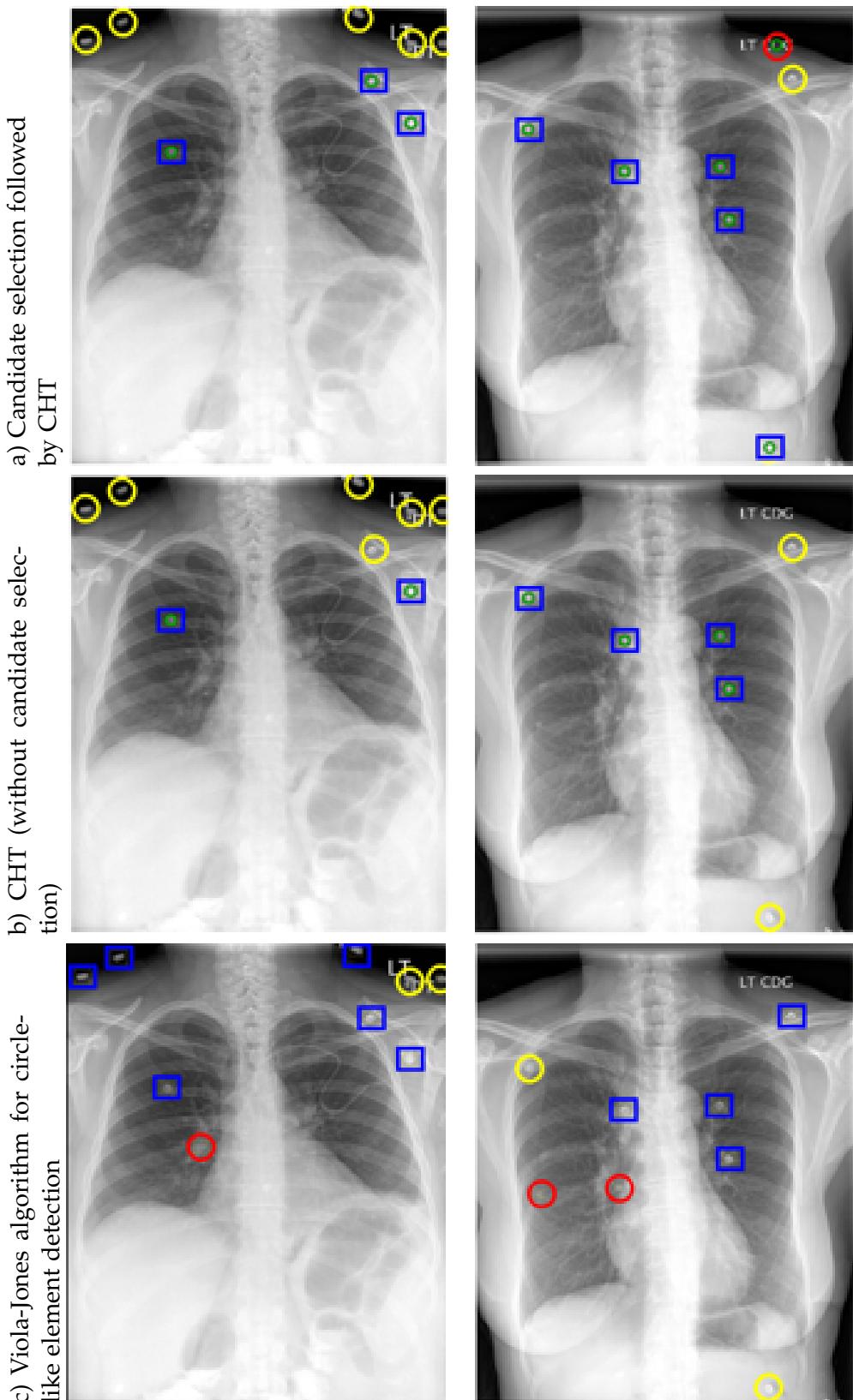


Figure 3.6: Circle-like element detection without chest segmentation in CXR image: a), b) and c) show CHT with CS, CHT and Viola-Jones respectively. Here, circles marked by yellow pointed by white arrow indicate false negative, circles marked by red pointed by white arrow indicate false positive and rectangles marked by blue indicate true positive.

Table 3.1: Dataset description

# of CXRs	# of elements in CXR WLS	# of elements in CXR WoLS
Subset1 = 50	32	136
Complete data = 400	325	1178

segmentation respectively.

3.4 Experiments

In our experiments, we use following setup for finding circle-like foreign element in CXR images.

3.4.1 Dataset

In our study, we use subset of dataset maintained by National Library of Medicine (NLM) - National Institutes of Health (NIH), which is composed of 400 DICOM images. From this subset of dataset, we take 50 DICOM images as subset 1 total of 400 DICOM dataset for our initial experiment. For our final experiment, we use total 400 dataset for robust performance analysis. Total number circle-like foreign element is presented in Table 3.1 within the chest region and all over the CXR images. We annotate corresponding ground-truths since they are not available in this data set (see Appendix A.1).

3.4.2 Evaluation metrics and protocol

We use precision, recall, and F1 score to measure our performance. Precision, recall and F1 score are defined in terms of a set of retrieved and relevant documents.

Evaluation metrics

- *Precision.* Precision (Maimon and Rokach 2005) is defined as the fraction of retrieved documents that are relevant to the query.
- *Recall.* Recall (Maimon and Rokach 2005) is the fraction of the documents that are relevant to the query that are successfully retrieved.

We use following equations for precision and recall for measuring performance of

our experiments.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{and} \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (3.3)$$

where TP = true positive (accurately detected buttons in the chest area), FP = false positive (inaccurately detected buttons), and FN = false negative (un-detected buttons).

- *F1 score.* F1 score (Powers 2011) comprises with the harmonic mean of precision and recall, the traditional F-measure or balanced F-score:

$$F_1 = 2 \cdot \frac{1}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \quad (3.4)$$

Evaluation protocol. For detecting circle-like element in CXR images, we use following evaluation protocol.

- k-fold cross-validation: For observing the experimental results of our test dataset, we use k-fold cross-validation where dataset is randomly segmented into k same size sub partitions. In that k sub partitions, k-1 sub partitions are used for training data set and one of the sub partitions is used for test dataset. For Viola-Jones technique, we use k-folding cross validation where $k = 10$ for the subset1 and complete data. Tables 3.6 and 3.13 represent the indices's of each partition of chest X-ray images with and without lung segmentation respectively.
- With lung segmentation (WSL): In our experiment, we perform lung segmentation in chesst X-ray images for computing the detection result only lung area. In that case, we only take account circle-like foreign element inside the lung area.
- Without lung segmentation (WoSL): We also perform circle-like foreign element detection without lung segmentation. In that case, we take account all circle-like element across the chest X-ray images.

Table 3.2: Circle-like foreign element detection results for "subset1" (WLS): CHT with CS

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
50	32	32	32	0	0	1	1	1

Table 3.3: Circle-like foreign element detection results for "subset1" (WoLS): CST with CS

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
50	136	87	77	10	59	0.86	0.57	0.69

3.4.3 Results and analysis

In this subsection, we explain detailed information of the performance of all the method that we used in this chapter. At first, we give rigorous analysis of our proposed method "CHT with CS" performance. Next, we present the comparison between our proposed method with CHT and Viola-Jones respectively. In our detection results, we represent Ground-truth as GT.

Our Results. In our proposed technique, we apply four different edge operations and additional circular object enhancement step is performed on these edge images. Fig. 3.3 S1-S4 shows all four steps of this approach. It can be seen from the S4 step of the figure that Prewitt edge detection performs the best with distinct circular objects and less amount of outlier high gradient regions. It is also the fastest method, due to the less amount of high gradient regions. Sobel operator also had similar strong circular objects but it has more outlier high gradient regions too, which leads to false detection and more computation time. On the other hand, Roberts does not have distinct circular objects. However, Canny performs the worst with this approach, because Canny edge-image has many strong connected object all over the image. Hence, after the dilation and filling steps, a big portion of the image gets filled and CHT does not work well on such images.

Table 3.4: Circle-like foreign element detection results for "subset1" (WLS): CHT (without CS)

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
50	32	32	22	0	10	1	0.69	0.81

Table 3.5: Circle-like foreign element detection results for "subset1" (WoLS): CHT (without CS)

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
50	136	79	64	15	72	0.81	0.47	0.60

Table 3.6: k-fold cross validation for for "subset1": Viola-Jones

Partition (k)	Dataset indices
1	38, 16, 13, 3, 6
2	11, 25, 40, 10, 4
3	29, 33, 41, 47, 26
4	5, 45, 1, 28, 22
5	14, 21, 48, 15, 49
6	20, 18, 36, 43, 30
7	44, 31, 32, 50, 35
8	23, 46, 8, 7, 17
9	2, 39, 42, 37, 9
10	12, 24, 34, 19, 27

Table 3.7: Circle-like foreign element detection results for "subset1" (WLS, k-fold cross validation): Viola-Jones

Train	Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
1	2,3,4,5,6,7,8,9,10	4	4.	3	1	1	0.75	0.75	0.75
2	1, 3,4,5,6,7,8,9,10	1	3	1	2	0	0.33	1	0.50
3	1,2,4,5,6,7,8,9,10	1	2	1	1	0	0.50	1	0.66
4	1,2,3,5,6,7,8,9,10	4	4	4	0	0	1	1	1
5	1,2,3,4,6,7,8,9,10	5	5	4	1	1	0.80	0.80	0.80
6	1,2,3,4,5,7,8,9,10	6	6	5	1	1	0.83	0.83	0.83
7	1,2,3,4,5,6,8,9,10	8	8	7	1	1	0.87	0.87	0.87
8	1,2,3,4,5,6,7,9,10	6	6	5	1	1	0.83	0.83	0.83
9	1, 2,3,4,5,6,7,8,10	2	4	2	2	0	0.50	1	0.66
10	1, 2,3,4,5,6,7,8,9	4	5	4	1	0	0.80	1	0.88
Average							0.72	0.91	0.78

Hence, we only include results of this approach with Prewitt edge-images.

Figs. 3.5 and 3.6 a) show circular foreign object detection results for this approach with and without chest segmentation respectively. The detection performance is presented in Tables 3.2 and 3.3 for subset1 and in Tables 3.9 and 3.10 for complete data with and without lung segmentation respectively. Since "CHT with CS" does not have any training stage, k-fold cross validation is not performed for this method.

Table 3.8: Circle-like foreign element detection results for "subset1" (WoLS, k-fold cross validation): Viola-Jones

Train	Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
1	2,3,4,5,6,7,8,9,10	10	10	7	3	4	0.70	0.63	0.66
2	1,3,4,5,6,7,8,9,10	18	18	13	5	4	0.72	0.76	0.74
3	1,2,4,5,6,7,8,9,10	15	15	10	5	2	0.66	0.83	0.74
4	1,2,3,5,6,7,8,9,10	12	12	9	3	1	0.75	0.90	0.81
5	1,2,3,4,6,7,8,9,10	14	14	10	4	1	0.71	0.90	0.80
6	1,2,3,4,5,7,8,9,10	20	20	13	7	5	0.65	0.72	0.68
7	1,2,3,4,5,6,8,9,10	18	18	13	5	3	0.72	0.81	0.76
8	1,2,3,4,5,6,7,9,10	16	16	12	4	4	0.75	0.75	0.75
9	1,2,3,4,5,6,7,8,10	16	16	14	2	3	0.87	0.82	0.84
10	1,2,3,4,5,6,7,8,9	14	14	10	4	4	0.71	0.71	0.71
Average							0.73	0.78	0.75

Table 3.9: Circle-like foreign element detection results for "complete data" (WLS): CHT with CS

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
400	325	305	293	12	32	0.96	0.90	0.92

Applying this method, we get 100%, 100%, and 100% precision, recall, and F1 score respectively for subset1 and 96%, 90%, and 92% precision, recall, and F1 score respectively for complete data within lung region.

Comparison with benchmarking techniques. Now, we compare our results with two benchmarking techniques CHT and Viola-Jones that are used in (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015) for circular object detection.

For CHT algorithm (see section 3.2.3), we follow similar approach as in (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015) and use the implementation in MATLAB image processing toolbox. However, instead of using a single radii

Table 3.10: Circle-like foreign element detection results for "complete data" (WoLS): CST with CS

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
400	1178	748	636	112	542	0.85	0.54	0.66

Table 3.11: Circle-like foreign element detection results for "**complete data**" (WLS): CHT (without CS)

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
400	325	256	241	15	84	0.94	0.74	0.83

Table 3.12: Circle-like foreign element detection results for "**complete data**" (WoLS): CHT (without CS)

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
400	1178	654	530	124	648	0.81	0.45	0.58

Table 3.13: k-fold cross validation for "**complete data**": Viola-Jones

Partition (k)	Dataset indices
1	12, 243, 194, 356, 111, 172, 72, 149, 187, 318, 100, 204, 347, 247, 268, 166, 382, 373, 327, 280, 380, 5, 139, 123, 192, 388, 45, 34, 228, 357, 189, 163, 11, 370, 88, 21, 267, 263, 278, 84
2	205, 261, 17, 103, 368, 362, 295, 71, 9, 238, 19, 10, 20, 287, 335, 201, 337, 22, 148, 366, 96, 51, 53, 115, 91, 39, 23, 42, 90, 157, 257, 272, 165, 109, 203, 262, 151, 32, 349, 265
3	383, 24, 74, 33, 87, 27, 237, 202, 161, 144, 251, 184, 122, 322, 360, 256, 79, 330, 217, 182, 28, 110, 120, 43, 364, 159, 363, 252, 399, 181, 321, 333, 391, 73, 58, 218, 63, 381, 239, 271
4	38, 6, 26, 131, 25, 74, 302, 105, 314, 306, 365, 62, 215, 273, 374, 293, 233, 168, 367, 328, 56, 344, 208, 231, 142, 319, 66, 308, 353, 339, 153, 162, 248, 52, 177, 170, 147, 288, 129, 279
5	199, 276, 173, 35, 290, 116, 245, 389, 112, 160, 289, 371, 114, 260, 77, 294, 193, 207, 211, 137, 97, 317, 76, 158, 346, 232, 313, 348, 358, 304, 48, 210, 393, 186, 41, 213, 191, 230, 143, 283
6	241, 196, 133, 117, 376, 390, 174, 190, 185, 340, 101, 355, 284, 176, 132, 130, 89, 258, 60, 298, 301, 336, 315, 395, 154, 244, 134, 332, 140, 55, 343, 386, 369, 198, 234, 29, 379, 93, 286, 266
7	81, 80, 214, 259, 124, 246, 270, 7, 229, 384, 385, 15, 167, 152, 240, 250, 296, 334, 155, 375, 316, 82, 68, 83, 175, 102, 400, 300, 223, 121, 309, 54, 125, 253, 1, 352, 212, 99, 219, 85
8	195, 46, 57, 378, 242, 312, 8, 95, 30, 249, 36, 323, 291, 331, 126, 394, 303, 31, 269, 37, 67, 398, 156, 179, 118, 387, 224, 226, 351, 338, 325, 61, 222, 146, 4, 206, 178, 292, 307, 354
9	254, 285, 342, 264, 311, 281, 341, 135, 275, 227, 94, 104, 220, 180, 59, 16, 197, 188, 98, 392, 377, 106, 305, 225, 221, 236, 310, 108, 128, 150, 65, 13, 69, 70, 2, 282, 396, 119, 138, 345
10	329, 183, 3, 64, 235, 78, 86, 397, 142, 319, 66, 308, 353, 339, 153, 162, 248, 52, 177, 170, 147, 288, 129, 279, 44, 299, 107, 372, 49, 47, 361, 209, 233, 168, 367, 328, 56, 344, 208, 231

Table 3.14: Circle-like foreign element detection results for "complete data" (WLS, k-fold cross validation): Viola-Jones

k	Train	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
1	2,3,4,5,6,7,8,9,10	28	100	28	72	0	0.28	1	0.43
2	1,3,4,5,6,7,8,9,10	34	119	34	85	0	0.28	1	0.44
3	1,2,4,5,6,7,8,9,10	28	103	28	75	0	0.27	1	0.42
4	1,2,3,5,6,7,8,9,10	37	117	37	80	0	0.31	1	0.48
5	1,2,3,4,6,7,8,9,10	45	125	45	80	0	0.36	1	0.52
6	1,2,3,4,5,7,8,9,10	53	142	53	89	0	0.37	1	0.54
7	1,2,3,4,5,6,8,9,10	52	122	52	70	0	0.42	1	0.59
8	1,2,3,4,5,6,7,9,10	41	102	41	61	0	0.40	1	0.57
9	1, 2,3,4,5,6,7,8,10	55	119	55	64	0	0.46	1	0.63
10	1,2,3,4,5,6,7,8,9	48	119	48	71	0	0.40	1	0.57
Average							0.36	1	0.52

Table 3.15: Circle-like foreign element detection results for "complete data" (WoLS, k-fold cross validation): Viola-Jones

k	Train	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
1	2,3,4,5,6,7,8,9,10	103	294	90	204	13	0.30	0.87	0.45
2	1,3,4,5,6,7,8,9,10	117	334	109	225	8	0.32	0.93	0.48
3	1,2,4,5,6,7,8,9,10	100	284	95	189	5	0.33	0.95	0.49
4	1,2,3,5,6,7,8,9,10	108	304	104	200	4	0.34	0.96	0.50
5	1,2,3,4,6,7,8,9,10	85	288	78	210	7	0.27	0.91	0.41
6	1,2,3,4,5,7,8,9,10	144	307	137	170	7	0.44	0.95	0.60
7	1,2,3,4,5,6,8,9,10	129	315	120	195	9	0.38	0.93	0.54
8	1,2,3,4,5,6,7,9,10	118	295	115	180	3	0.38	0.97	0.55
9	1,2,3,4,5,6,7,8,10	135	290	130	160	5	0.44	0.96	0.61
10	1,2,3,4,5,6,7,8,9	138	299	129	170	9	0.43	0.93	0.59
Average							0.37	0.94	0.53

Table 3.16: Processing time of button detection techniques

Method	Time(in sec)
CHT with CS	8.52
Viola-Jones	18.83
CHT	29.02

range, we use a set of three radii ranges ([12, 19], [25, 45] and [46, 60]) and accumulate the results. It not only increases the speed but also accuracy of the detection process. Fig. 3.5 and Fig. 3.6 b) show two output images of this algorithm for subset1 and total dataset respectively with and without in chest segmentation. The detection performance is presented in Table 3.4 and Table 3.5 for subset1 and in Table 3.11 and Table 3.12 for

complete data with and without lung segmentation respectively. Since "CHT" does not have any training stage, k-fold cross validation is not performed for this method.

For Viola-Jones algorithm (see section 3.2.2), we also apply similar approach in (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015). We use 300 and 200 negative samples for subset1 and complete data. We take 389 and 228 positive and negative samples for both dataset. We observe that Viola-Jones detector perform best using Haar-like feature. We use the number of cascade stage 6 and 15 for subset1 and complete data. Fig. 3.5 and Fig. 3.6 c) shows two of the resultant outputs of this method. Since "Viola-Jones" have training stage, we perform k-fold cross validation method for generating unbiased detection result. The detection performance is presented in Table 3.7 and Table 3.8 for subset1 and in Table 3.14 and Table 3.15 for complete data with and without lung segmentation respectively. From the Table 3.14 and Table 3.15, we can see that detection results in terms of precision has poor performance due to large number of positive training samples that leads to false detection. On the other hand, detection detection results in terms of recall improves due to having large number of positive samples. From the Table 3.7 and Table 3.8, we can observe that precision and recall have quite similar performance due to moderate number of training samples.

Processing time. We also measure the computational time for all circular object detection techniques. The computation time is presented in Table 3.16. Our system has following configuration: intel core i7 processor, windows 7 operating system and MATLAB R2016a. We calculated computational time in seconds per image, on average. Average CXR size is around 3K x 3K. Table 3.16 shows the computational time of all circular object detection techniques. Note that in (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015), authors did not report computational time.

3.5 Conclusion

In this work, we have focused on identifying circle-like foreign elements in CXR images such as buttons appearing in lung regions of the chest X-ray images. We present

a novel technique for circular foreign object detection. Our proposed technique is encouraging, both in terms of detection accuracy and computation time. The detection performance of total dataset is quite similar with the detection performance of subset1 dataset. We have showed our results with and without lung segmentation for both datasets. We can see from our results that using our proposed technique, we get 100% precision, recall and F1 score with chest segmentation for subset1 dataset and we get 96%, 90% and 92% precision, recall and F1 score for total dataset. Then, we can say that our proposed technique performed robustly under a variety of dataset for withing chest region. In our system, precision shows marginal difference with (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015). In our recall presents substantial difference with (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015). On the other hand, for without chest segmentation, the detection performance falls due to the irregular shape of the circle-like elements and intensity difference with background with our desire circle-like elements in CXR image. We have also measured the computational time for all circular object detection techniques.

3.6 What's next?

Considering experiment in this chapter, we have achieved encouraging performance in terms of precision, recall and F1 score within chest region. However, detection performance decreases drastically if we consider it without lung segmentation. For that reason, we have carried our research and have implemented a novel technique for circle-like element detection in CXR without focusing on lung segmentation. In the next chapter, we explain circle-like element detection in CXR using normalized cross-correlation with unsupervised clustering.

CHAPTER 4

Circle-like foreign element detection using normalized cross-correlation and unsupervised clustering in CXR image

4.1 Summary

We can see from chapter three that if we would like to perform all of those circle-like foreign element detection algorithms in chest x-ray images then the performance will degrade drastically due to irregular shape and low contrast of the circle-like element with background CXR image. For improving the performance of circle-like foreign element detection in the CXR image without considering chest segmentation and automated the system, we implement a novel technique that is robust in terms of precision, recall and F1 score. In this chapter, we present a novel technique for detecting circle-like foreign element, in particular buttons, within chest X-ray images. In our technique, we use a pre-processing step that enhances the CXRs. Using these enhance images, we perform normalized cross-correlation (using positive and negative template) and unsupervised clustering (hierarchical clustering) for foreign element detection within CXR images. Finally, measuring detection performance circle-like foreign elements, we automate our system to comparing annotated ground-truth of test dataset with output detection result. In all tests, our algorithm performed well under a variety of CXRs. We also compare our proposed technique's performance with previous chapter techniques. This novel technique is able to excel performance in terms of detection accuracy precision, recall, and F1 score.

Key topics:

- Image enhancement
- Fast Fourier transform and convolution for getting cross-correlation
- Implementing integral image for calculating mean and standard deviation

- Perform normalized cross-correlation using positive and negative template
- Perform unsupervised clustering (hierarchical clustering) for candidate selection
- Automate the system using selected candidate and annotated ground-truth in test CXRs
- Results, comparative study, and analysis with existing bench marking techniques

Organization of chapter. The rest of this chapter is structured as follows: In section 4.2, we give detailed information of our all performed experiment's material. In section 4.3, we give detailed overview of our proposed technique. In subsection 4.2.1, we compute normalization cross-correlation with positive and negative template, after that create sample for finding peak using hierarchical clustering. Then in subsection 4.3.10, we explain our decision taking technique. Next, Section 4.4 provides information about the dataset and the evaluation metrics and protocol for performance measurement and results (see Section 4.4.1). Section 4.5 concludes the paper. Finally, Section 4.6 describes what is the next chapter.

4.2 Materials

In this section, we explain all necessary detailed information that is related to our proposed method. For performing normalized cross-correlation, we explain fast Fourier transform (FFT), convolution, integral image, mean, variance, standard deviation and cross-correlation. We also give detailed explanation of unsupervised clustering for finding the peak form the resultant normalized cross-correlation output.

4.2.1 Normalized cross-correlation

For performing normalized cross-correlation we explain following material in details.

Discrete Fourier transform (DFT). DFT plays an important role for performing convolutions. It can be computed efficiently using FFT (Bracewell 1965) (Weisstein 2002a) (Weisstein 2002b).

The sequence of N complex numbers: x_0, x_1, \dots, x_{N-1} is transformed into an N -periodic sequence of complex numbers:

$$\mathbf{X}_k \stackrel{\text{def}}{=} \sum_{n=0}^{N-1} x_n \cdot e^{-2\pi i k n / N}, \quad k \in \mathbb{Z} \text{ (integers).} \quad (4.1)$$

It can also be represented by:

$$\mathbf{X} = \mathcal{F}\{\mathbf{x}\} \text{ or } \mathcal{F}(\mathbf{x}) \text{ or } \mathcal{F}\mathbf{x}. \quad (4.2)$$

The above formula can be interpreted in following way:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} \mathbf{X}_k \cdot e^{2\pi i k n / N}, \quad n \in \mathbb{Z}. \quad (4.3)$$

Trigonometric form of the above formula is given below:

1. Fourier Transform:

$$\mathbf{X}_k = \sum_{n=0}^{N-1} x_n \cdot (\cos(-2\pi k \frac{n}{N}) + i \sin(-2\pi k \frac{n}{N})), \quad k \in \mathbb{Z} \quad \text{and} \quad (4.4)$$

2. Inverse Fourier Transform:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} \mathbf{X}_k \cdot (\cos(2\pi k \frac{n}{N}) + i \sin(2\pi k \frac{n}{N})), \quad n \in \mathbb{Z}. \quad (4.5)$$

where

N = number of time samples,

n = current sample, we are considering $0, \dots, N-1$,

x_n = value of the signal at time n ,

k = current frequency we're considering (0 Hertz up to $N-1$ Hertz), and

X_k = amount of frequency k in the signal (Amplitude and Phase, a complex number).

Fast Fourier transform (FFT). It is a fast computational algorithm factorizing the DFT matrix into a product of sparse (mostly zero) factors. From DFT (see equation 4.1), it is assumed that computing DFT from this equation is too slow, since it uses decomposing technique. It needs $O(N^2)$ operations where X_k produces N output result by adding N terms. FFT can resolve the time complexity problem of DFT by producing the same results from $O(N^2)$ to $O(N \log N)$.

Convolution. Convolution can be expressed as the amount of overlap between two functions as one of them is shifted over the other. The convolution of two functions x and y is represented by $x * y$. It is expressed by the following equation:

$$(x * y)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} x(\tau) y(t - \tau) d\tau = \int_{-\infty}^{\infty} x(t - \tau) y(\tau) d\tau, \quad (4.6)$$

where at any moment t the convolution is defined as a weighted average of the function $x(\tau)$, with the weight being $y(-\tau)$ shifted by the amount t .

Convolution theorem. According to convolution theorem, (Smith et al. 1997) the point-wise product of Fourier transforms is the Fourier transform of convolution. It can be expressed by:

$$\mathcal{F}\{x * y\} = \mathcal{F}\{x\} \cdot \mathcal{F}\{y\}, \quad (4.7)$$

consider x and y denote two signals, \mathcal{F} represents the Fourier transform operator, \cdot represents point-wise multiplication, and $*$ represents the convolution operator. and Fourier transform of x and y are $\mathcal{F}\{x\}$ and $\mathcal{F}\{y\}$ respectively. The convolution of these two signals is expressed by following equation:

$$x * y = \mathcal{F}^{-1}\{\mathcal{F}\{x\} \cdot \mathcal{F}\{y\}\}, \quad (4.8)$$

where \mathcal{F}^{-1} is the inverse Fourier transform.

Mean. The average of a discrete set of numbers is called mean. It is represented by \bar{x} . Consider n number of samples in a set, $x = x_1, x_2, \dots, x_n$. Then mean (\bar{x}) is the sum of all the numbers in the set divided by the total number of samples in the set. It is represented by following equation:

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n}. \quad (4.9)$$

Standard deviation. The standard deviation (SD) (Bland and Altman 1996) (Walker 1929) (Gauss 1816) (Gorard 2005) is the square root of its variance. Let X be a random variable with mean value μ :

$$E[X] = \mu. \quad (4.10)$$

Here, the operator E represents the average or expected value of X . Then the standard deviation of X is the quantity:

$$\begin{aligned} \sigma &= \sqrt{E[(X - \mu)^2]} \\ &= \sqrt{E[X^2] + E[-2\mu X] + E[\mu^2]} = \sqrt{E[X^2] - 2\mu E[X] + \mu^2} \\ &= \sqrt{E[X^2] - 2\mu^2 + \mu^2} = \sqrt{E[X^2] - \mu^2} \\ &= \sqrt{E[X^2] - (E[X])^2}. \end{aligned} \quad (4.11)$$

(derived in terms of expected value). Standard deviation is the square root of the average value of $(X - \mu)^2$.

Integral image. The value at any point (x, y) in the summed area table (SAT) (Lewis) (Crow 1984) is the sum of all the pixels above and to the left of (x, y) , inclusive:

$$I_{\Sigma}(x, y) = \sum_{\substack{x' \leq x \\ y' \leq y}} i(x', y'). \quad (4.12)$$

It can be computed efficiently in a single pass over the image, using the fact that the value in the summed area table at (x, y) is just:

$$I(x, y) = i(x, y) + I(x - 1, y) + I(x, y - 1) - I(x - 1, y - 1). \quad (4.13)$$

Normalized cross-correlation. For image-processing applications in which the brightness of the image and template can vary due to lighting and exposure conditions. Image normalization can be solved this problem. This is typically done at each step by deducting the mean and dividing by the standard deviation (Lewis) (?). That is, the normalized cross-correlation of a template, $t(x,y)$ with a sub-image $f(x,y)$ is

$$NCC(t,f) = \frac{1}{n} \sum_{x,y} \frac{(f(x,y) - \bar{f})(t(x,y) - \bar{t})}{\sigma_f \sigma_t}, \quad (4.14)$$

where n is the number of pixels in $t(x,y)$ and $f(x,y)$, \bar{f} is the average of f and σ_f is standard deviation of f .

4.2.2 Clustering

Cluster analysis/segmentation analysis is the task of grouping a set of elements in such a way that elements in the same group (called a cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). It can be supervised/supervised.

Supervised clustering. Applying an algorithm, an input variables (x) and an output variable (Y) to learn the mapping function from the input to the output:

$$Y = f(X). \quad (4.15)$$

The goal is to approximate the mapping function so well that when we have new input data (x) that we can predict the output variables (Y) for that data.

Unsupervised clustering. It is a learning framework using a specific element functions, where only have input data (X) and no corresponding output variables. Common clustering algorithms are:

- *Hierarchical clustering*
- *k-Means clustering*

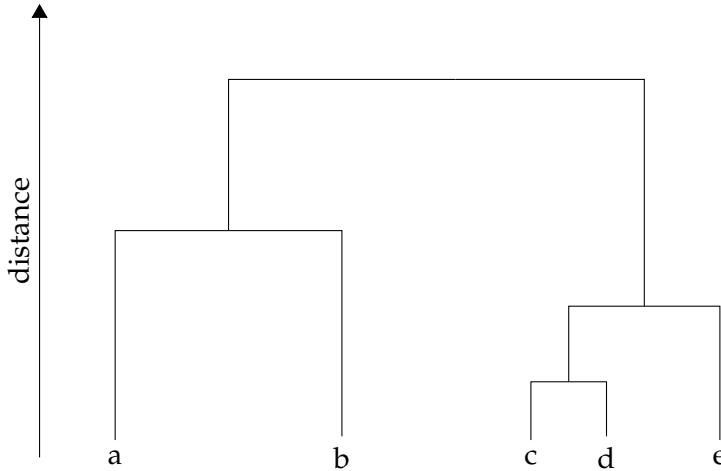


Figure 4.1: Hierarchical clustering using "bottom up" approach

- *Gaussian mixture models*
- *Self-organizing maps*
- *Hidden Markov models*

For our proposed algorithm, we use hierarchical clustering for finding peak from normalized cross-correlation coefficient.

Hierarchical clustering (HC)

In cluster analysis, hierarchical clustering is one of the technique that seeks to build a hierarchy of clusters. It groups data over a variety of scales by creating a cluster tree or dendrogram. The tree is a multilevel hierarchy where clusters at one level are linked with clusters at the next level. This allows us to decide the level or scale of clustering that is most appropriate for our application.

Strategies for hierarchical clustering generally fall into two types

- Agglomerative: It is a "bottom up" approach where every data starts in its own cluster, and two clusters are combined as one advance the hierarchy.
- Divisive: It is a "top down" approach where all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

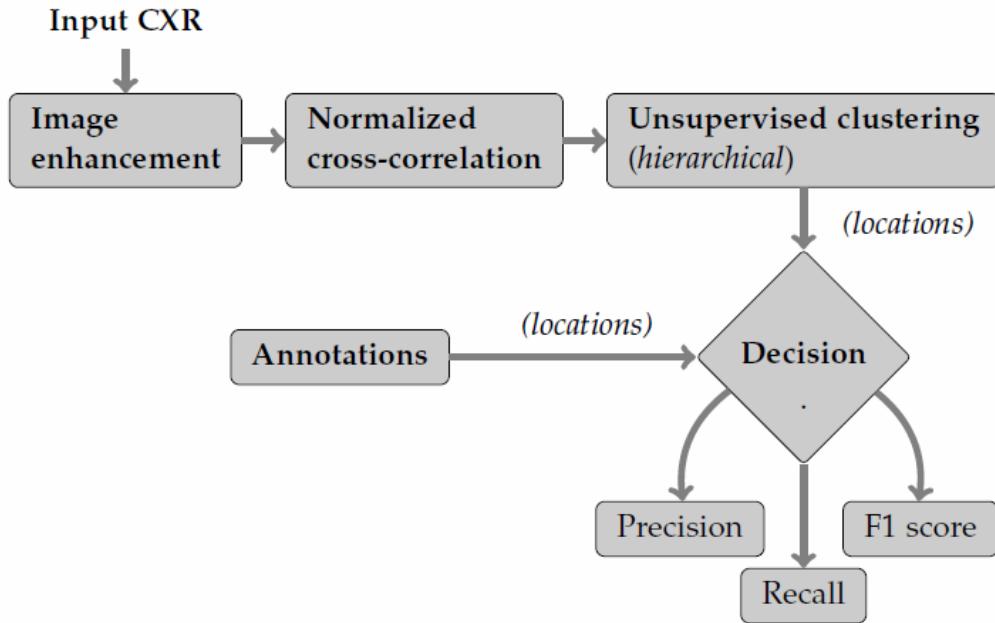


Figure 4.2: The process begins with image enhancement. Then perform normalized cross-correlation and hierarchical clustering and finally for automatic detection, it evaluates result by cross-matching with annotated ground-truth and clustered output..

After analyzing our problem, we use agglomerate hierarchical cluster analysis. Fig. 4.1 presents hierarchical clustering using "bottom up" approach. The detailed explanation of agglomerate hierarchical cluster analysis is given below:

Performing agglomerate hierarchical clustering on a data set, following steps are important:

- Find the similarity or dissimilarity between every pair of elements in the data set.
- Group the elements into a binary, hierarchical cluster tree.
- Determine where to cut the hierarchical tree into clusters.

4.3 The proposed technique

In our proposed algorithm, we first enhance the CXR images to increase the contrast between the button elements with their background, using intensity normalization and image adjustment. Next, we apply normalized cross-correlation step to find the cross-correlation coefficient with positive and negative template. For computing the cen-

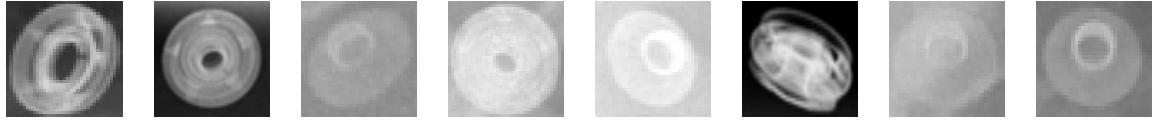


Figure 4.3: Positive training templates

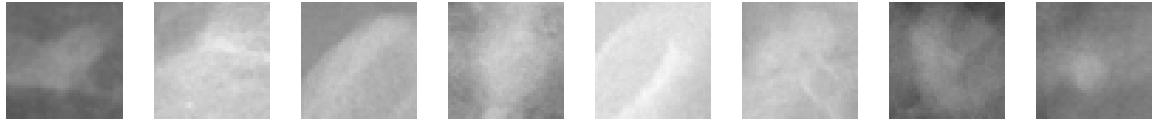


Figure 4.4: Negative training templates

tre of the circle-like foreign element, we perform unsupervised clustering and getting automatic decision, we validate clustered output with annotated ground-truth. Fig. 4.2 briefly presents the proposed method work-flow.

4.3.1 Image enhancement

For input CXR image enhancement, we apply similar approach in chapter 3 (see subsection 3.3.1).

4.3.2 Annotations

For annotations of cricle-like elements in CXR images please see Appendix A.1.

4.3.3 Positive training template selection

For selecting positive training sample, we manually cut the ground-truth 82×82 pixel. Fig. 4.3 shows some of such positive training templates.

4.3.4 Normalized cross-correlation

Cross-correlation. For computing normalized cross-correlation, we perform following steps (see subsection 4.2.1):

- Zero-pad the input signals (add zeros to the end so that at least half of the wave is blank).
- Flip positive and negative template 90 degree counter clock-wise.

- To compute two dimensional FFT, at first we compute one-dimensional DFT of each column of input signal, then of each row of the result.
- Take the FFT of both signals (template and input CXR image).
- Multiply the results together (element-wise multiplication).
- Perform the inverse FFT.
- Compute integral images of the first and second power in order to get mean and variance. After that compute the normalized cross-correlation coefficient at every step by subtracting the mean and dividing by the standard deviation. For details information see subsection 4.2.1.

4.3.5 Negative training template selection

For selecting negative training sample, we perform couple of steps to make it automatic and precise. Those steps are given below:

- Perform normalized cross-correlation (NCC) between positive template and input CXR image
- Analyzing the co-efficient of NCC to set a positive thresholding parameter for filtering the unwanted value for finding the positive samples.
- Perform hierarchical clustering (see section 4.2.2) for finding the peak or mean or candidate of each cluster from the positive sample. We take the location of each and every individual cluster's mean value and update the positive sample. Then we consider the location of each candidate.
- After selecting the location of each and every candidate, we cross validate it with our annotated ground-truth. For our cross validation, we measure the distance between our annotated ground-truth location and the candidate of positive sample . If the distance is less then or equal to a specific value (we take 50 pixel after analyzing circle-like element radius) then we consider that candidate as a positive

candidate or true positive otherwise we consider it as a negative candidate and save those candidate in negative mask. For collecting negative template, we consider the location of negative from the negative sample as center and cut negative template as same size as positive template (82×82 pixel) and save those template as a negative training template. Fig. 4.4 shows some of such negative training templates.

4.3.6 Thresholding

To find similarity between positive template and the input CXR image, threshold plays an important role in template matching using NCC. In the following two subsections; creating positive sample and negative sample we apply threshold on NCC coefficient values for filtering the NCC coefficient values. Fig. 4.5 shows resultant input CXR images, 3D plot of NCC coefficient between positive template and positive sample without threshold. But after applying threshold, we can see most of the false detection is filtered. In Fig 4.6 shows resultant threshold output for positive sample.

4.3.7 Training samples: positive and negative

After performing NCC between positive template and input CXR image, we analyze the score of each pixel and observe that above a certain NCC coefficient value, similarity between them is high which indicate our desire foreign elements. And below of that certain value or threshold we make the positive sample value zero and only keep the high coefficient value. Fig. 4.6 c) shows positive sample after applying threshold value. We follow similar approach for creating the negative sample using negative template.

4.3.8 Final NCC output

After computing the positive and negative sample , we compare their coefficient values for per pixel. If any negative score is greater than the positive score then we make that pixel value zero in the positive sample for creating final NCC output for circle-like element detection. In that way, we can minimize false detection drastically in template matching using NCC. Fig. 4.8 a) shows 3D plot of final NCC coefficient.

4.3.9 Peak detection

After creating the final sample , we apply hierarchical clustering for calculating the peak of that cluster. For detailed information see Section 4.2.2. In our annotation subsection, we mention that circle-like element radius is not greater then 50 pixel. For finding the peak from a cluster, we follow couple of step.

- Take the location of final sample .
- Find the individual cluster using the Euclidean distance metric.
- Calculate the mean of each cluster for computing the peak.

Analyze cluster histogram and threshold the values for being a valid cluster. In Fig. 4.7 b) shows the histogram of clustered locations in final sample .

Here, we get only one location per cluster. That location is our desire/circle-like element in input CXR image. Finally, we plot that locations on input CXR image as indicating foreign circle-like element detection point.

4.3.10 Decision

After finding circle-like elements in CXRs, we take their locations for computing our final decision. We find the Euclidean distance of those locations with our annotated ground-truth locations. If the distance is less than a specific radius then we consider that position is circle-like elements position of true positive or true detection otherwise we consider it as a false detection (not a circle-like element). For computing non detected circle-like elements, we subtract our annotated ground-truth with true positive or truly detected circle-like elements in CXR image. Fig. 4.8 b), shows the complete view of circular foreign object detection on candidate selection image without lung segmentation.

4.4 Experiments

In our experiments, we use following setup for finding circle-like foreign element in CXR images. We use similar dataset (see subsection 3.4.1) and evaluation metrics and protocol (see subsection 3.4.2) in chapter three.

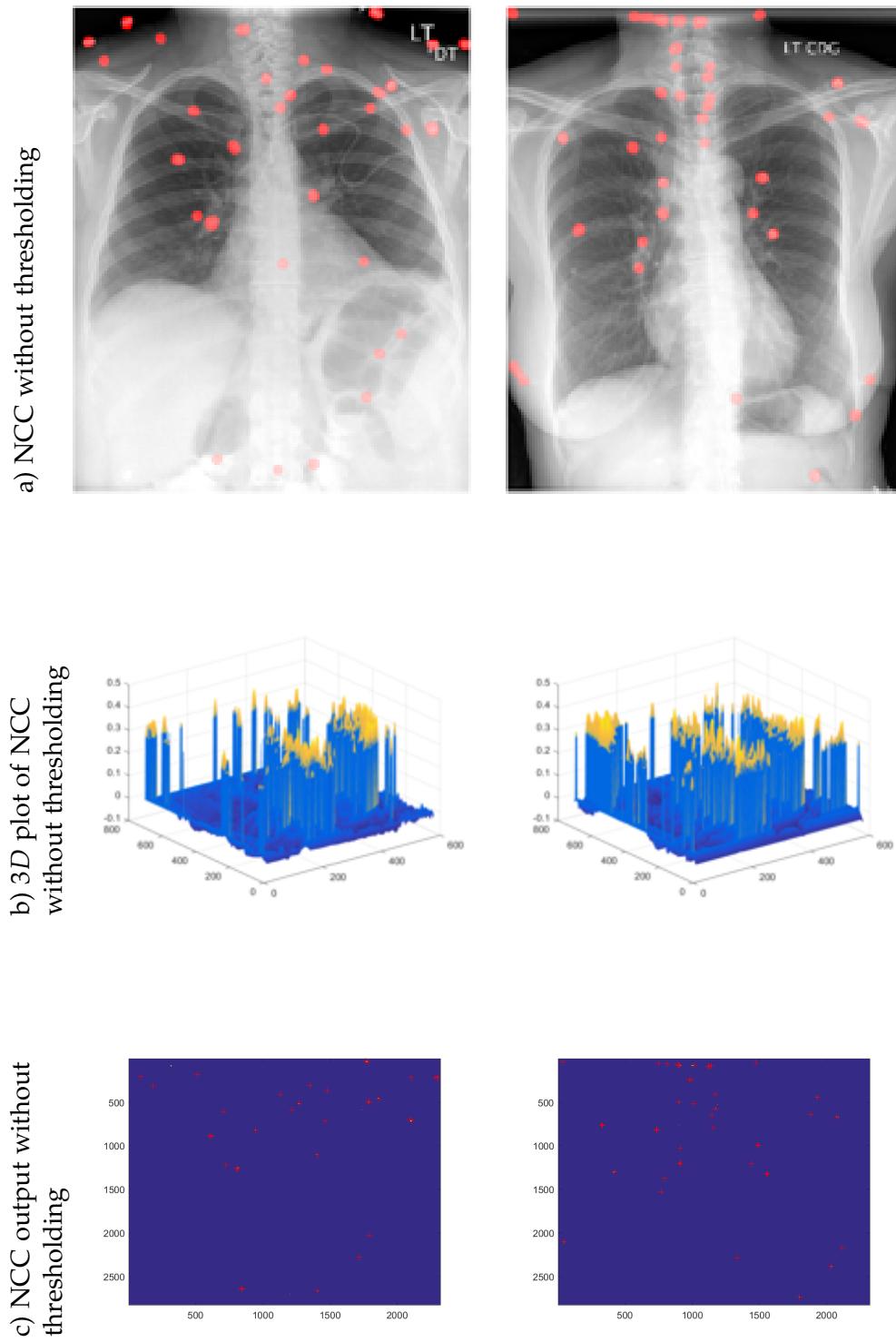


Figure 4.5: Detection with NCC with positive sample without applying thresholding.

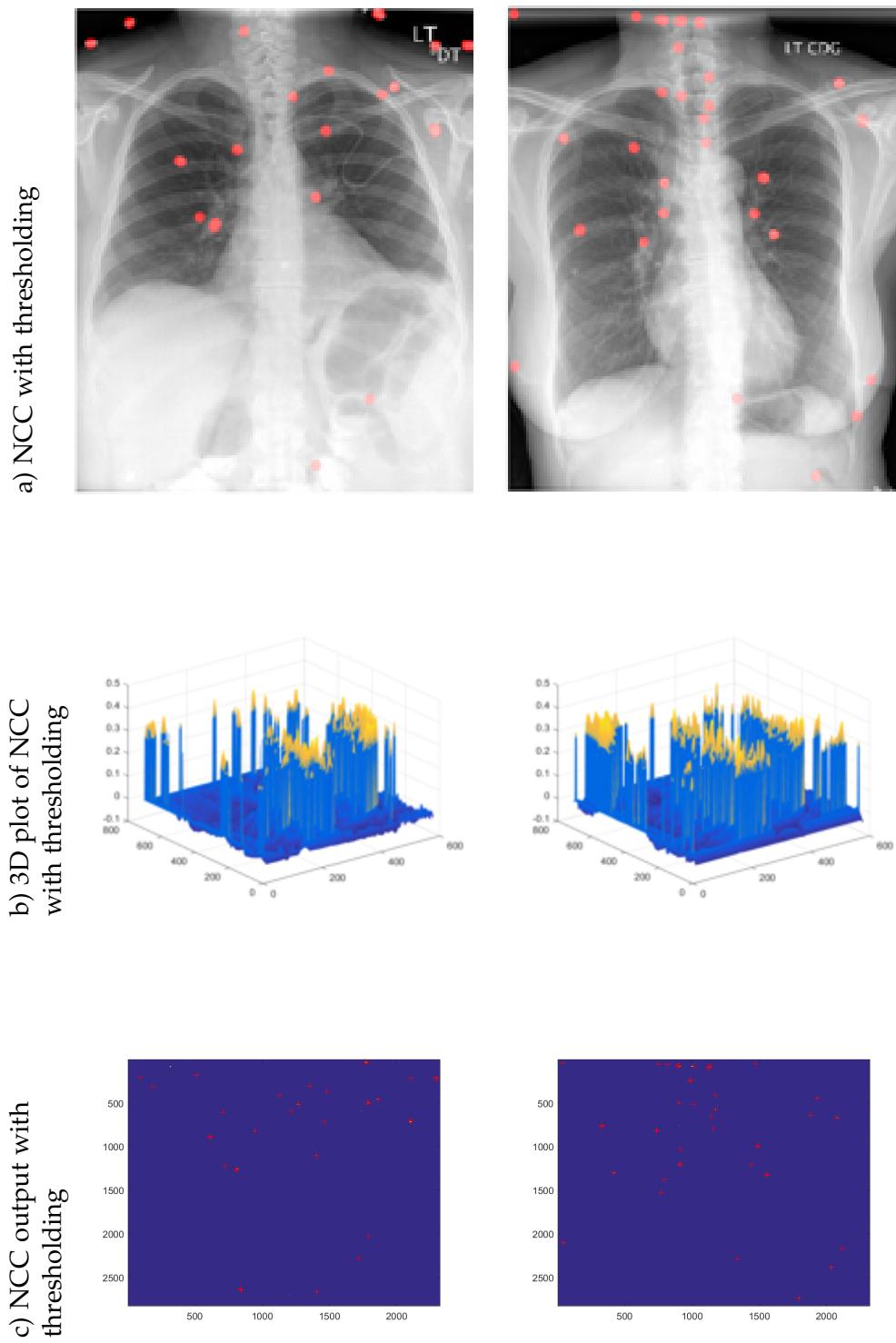


Figure 4.6: Detection with NCC with positive sample thresholding

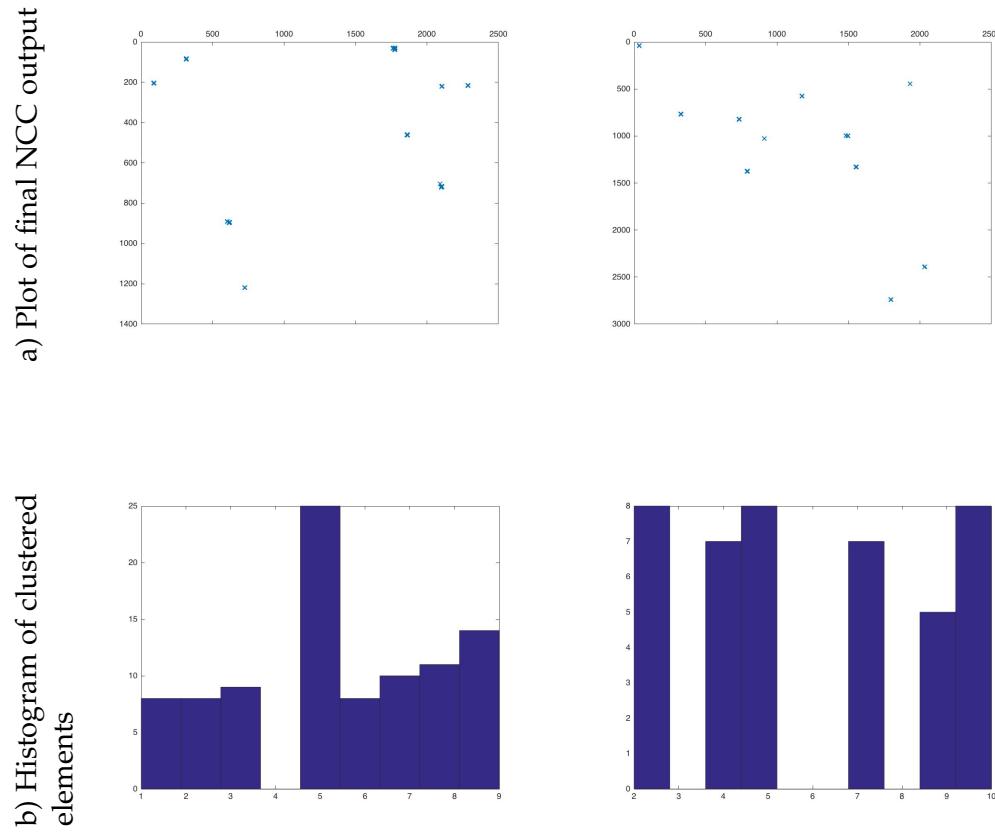


Figure 4.7: Foreign element detection

4.4.1 Results and analysis

In our proposed training based technique, we apply normalized cross-correlation with hierarchical clustering approach to detect the circle-like element in CXR images. Using our positive sample and negative sample for creating final sample and after that applying hierarchical clustering technique, we improve the precision of the circle-like element detection. Fig 4.8 b) shows the final detection result of NCC with hierarchical clustering technique. The detection performance is presented in Tables 4.1, and 4.2 for subset1 and in Tables 4.3 and 4.4 for complete data with and without lung segmentation respectively. Since this technique have training stage, we perform k-fold cross validation method for generating unbiased detection result. The detection performance is presented in Table 4.5 and Table 4.6 for complete data with and without lung segmentation respec-

a) 3D plot of final NCC output
b) Detection with final NCC

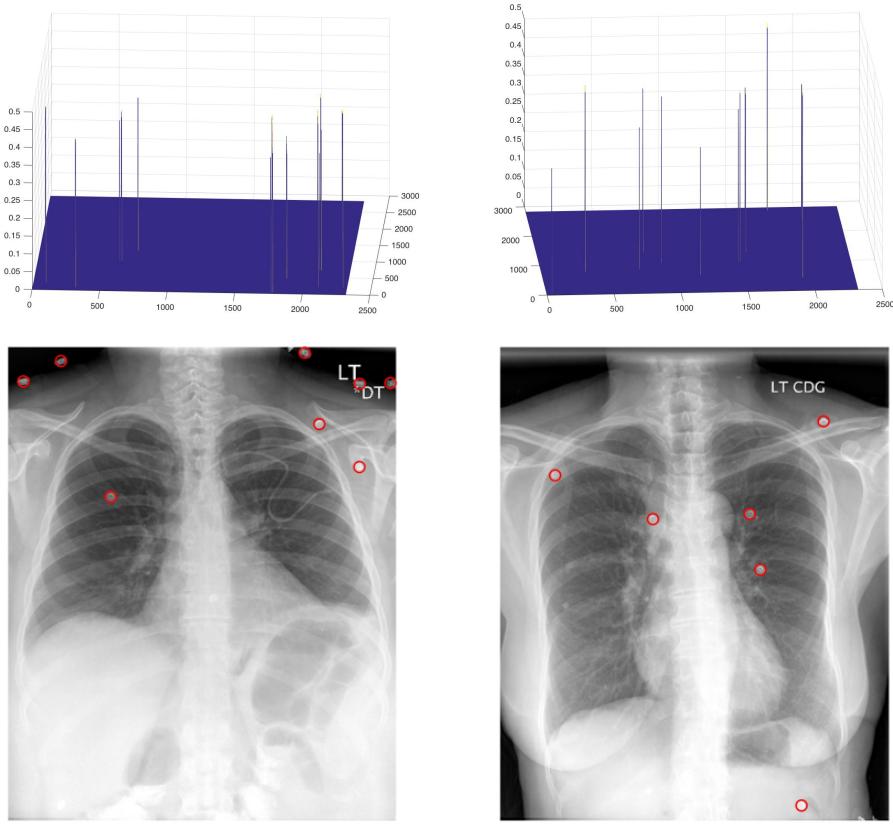


Figure 4.8: Detection with NCC: a) shows 3D plot of final NCC coefficient and b) circle-like element detection with final NCC output.

tively.

Now, we compare our NCC based hierarchical clustering technique with all of the techniques mentioned in chapter three. For comparing the techniques, we take best performance (precision, recall and F1 score) of every technique and each dataset. Tables 4.7, 4.8, 4.9 and 4.10 show the comparing result of four techniques: two is our implemented techniques (Normalized cross-correlation with hierarchical clustering and candidate selection followed by CHT) and other two benchmarking techniques (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015) are CHT and Viola-Jones for both datasets with and without lung segmentation. From those table, we can see that our training based technique's performance is approximately similar regardless with the

Table 4.1: Circle-like foreign element detection results for "subset1" (WLS): NCC with unsupervised clustering

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
50	32	34	32	2	2	0.93	0.99	0.96

Table 4.2: Circle-like foreign element detection results for "subset1" (WoLS): NCC with unsupervised clustering

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
50	136	140	133	7	3	0.95	0.98	0.96

Table 4.3: Circle-like foreign element detection results for "complete data" (WLS): NCC with unsupervised clustering

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
400	325	384	319	65	6	0.83	0.98	0.90

Table 4.4: Circle-like foreign element detection results for "complete data" (WoLS): NCC with unsupervised clustering

Test	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
400	1178	1229	1143	202	35	0.85	0.97	0.90

Table 4.5: Circle-like foreign element detection result for "complete data" (WLS, k-fold cross validation): NCC with unsupervised clustering

k	Train	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
1	2,3,4,5,6,7,8,9,10	28	30	26	4	2	0.87	0.93	0.90
2	1,3,4,5,6,7,8,9,10	34	38	31	4	3	0.82	0.91	0.86
3	1,2,4,5,6,7,8,9,10	28	27	24	2	4	0.94	0.89	0.92
5	1,2,3,4,6,7,8,9,10	45	47	43	4	2	0.91	0.96	0.93
6	1,2,3,4,5,7,8,9,10	53	55	50	5	3	0.91	0.94	0.93
7	1,2,3,4,5,6,8,9,10	52	55	48	7	4	0.87	0.92	0.9
8	1,2,3,4,5,6,7,9,10	41	43	38	5	3	0.88	0.93	0.9
9	1, 2,3,4,5,6,7,8,10	55	56	52	4	3	0.93	0.95	0.94
10	1,2,3,4,5,6,7,8,9	48	51	48	3	0	0.94	1	0.97
Average							0.90	0.93	0.91

dataset. And all of the cases our NCC based hierarchical clustering performs robustly regardless with and without lung segmentation in terms of precision, recall and F1 score. For checking robustness, we test original input images with adding white Gaussian noise. From Table 4.11, we can see that increasing the SNR above 30dB, performance of our tech-

Table 4.6: Circle-like foreign element detection result for "complete data" (WoLS, k-fold cross validation): NCC with unsupervised clustering

k	Train	GT	Detect	TP	FP	FN	Precision	Recall	F1 score
1	2,3,4,5,6,7,8,9,10	103	113	91	21	11	0.81	0.89	0.84
2	1,3,4,5,6,7,8,9,10	117	133	105	27	11	0.79	0.90	0.84
3	1,2,4,5,6,7,8,9,10	100	112	87	25	13	0.77	0.87	0.82
4	1,2,3,5,6,7,8,9,10	108	116	90	25	17	0.78	0.84	0.80
5	1,2,3,4,6,7,8,9,10	85	106	75	30	9	0.71	0.89	0.78
6	1,2,3,4,5,7,8,9,10	144	172	129	43	14	0.75	0.90	0.81
7	1,2,3,4,5,6,8,9,10	129	154	109	44	19	0.71	0.85	0.77
8	1,2,3,4,5,6,7,9,10	118	136	105	31	12	0.77	0.89	0.82
9	1,2,3,4,5,6,7,8,10	135	151	113	37	21	0.75	0.84	0.79
10	1,2,3,4,5,6,7,8,9	138	161	122	38	15	0.76	0.89	0.81
Average							0.76	0.88	0.81

Table 4.7: Comparison of circle-like foreign element detection result for "subset1" (WLS)

Techniques	Precision	Recall	F1 score
NCC with unsupervised clustering	0.93	0.99	0.96
CHT with CS	1	1	1
Viola-Jones	0.72	0.91	0.78
CHT	1	0.69	0.81

Table 4.8: Comparison of circle-like foreign element detection result for "subset1" (WoLS)

Techniques	Precision	Recall	F1 score
NCC with unsupervised clustering	0.95	0.98	0.96
CHT with CS	0.86	0.57	0.69
Viola-Jones	0.73	0.78	0.75
CHT	0.81	0.47	0.60

Table 4.9: Comparison of circle-like foreign element detection result for the "complete data" (WLS)

Techniques	Precision	Recall	F1 score
NCC with unsupervised clustering	0.90	0.93	0.91
CHT with CS	0.96	0.90	0.92
Viola-Jones	0.36	1	0.52
CHT	0.94	0.74	0.83

nique is similar with the original result.

Table 4.10: Comparison of circle-like foreign element detection result for the the "complete data" (WoLS)

Techniques	Precision	Recall	F1 score
NCC with unsupervised clustering	0.76	0.88	0.81
CHT with CS	0.85	0.54	0.66
Viola-Jones	0.37	0.94	0.53
CHT	0.81	0.45	0.58

Table 4.11: Circle-like foreign element robustness check (adding Gaussian noise with variable SNR) for the "subset1" (WoLS) (groundtruth-166): NCC with unsupervised clustering

SNR in dB	TP	FP	FN	Precision	Recall	F1 score
original	143	16	16	0.89	0.90	0.90
5	0	0	166	NAN	0	NAN
10	2	10	164	0.17	0.01	0.02
20	45	14	102	0.77	0.31	0.44
30	134	23	25	0.86	0.85	0.86
40	142	26	17	0.85	0.90	0.87
50	143	22	16	0.87	0.90	0.88
60	143	22	16	0.89	0.90	0.90

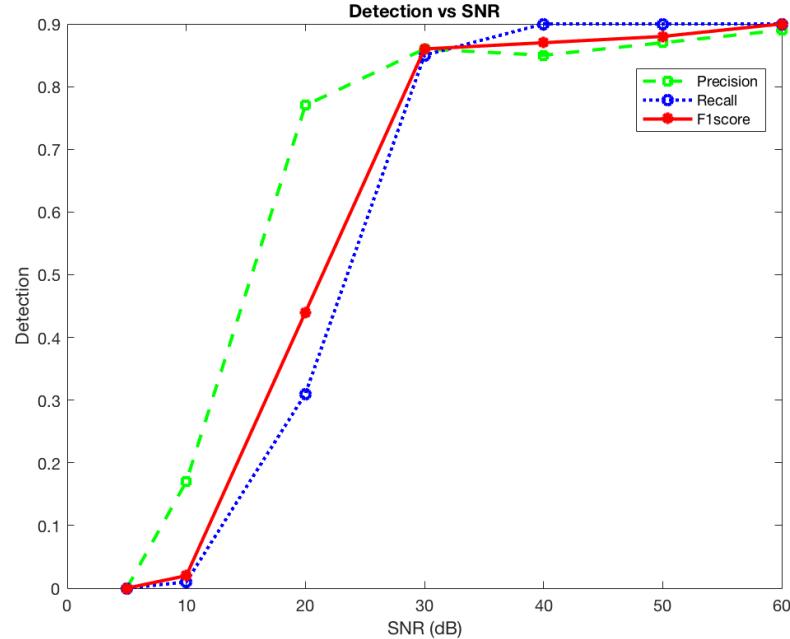


Figure 4.9: Precision, recall, and F1 score vs SNR (in dB) plot

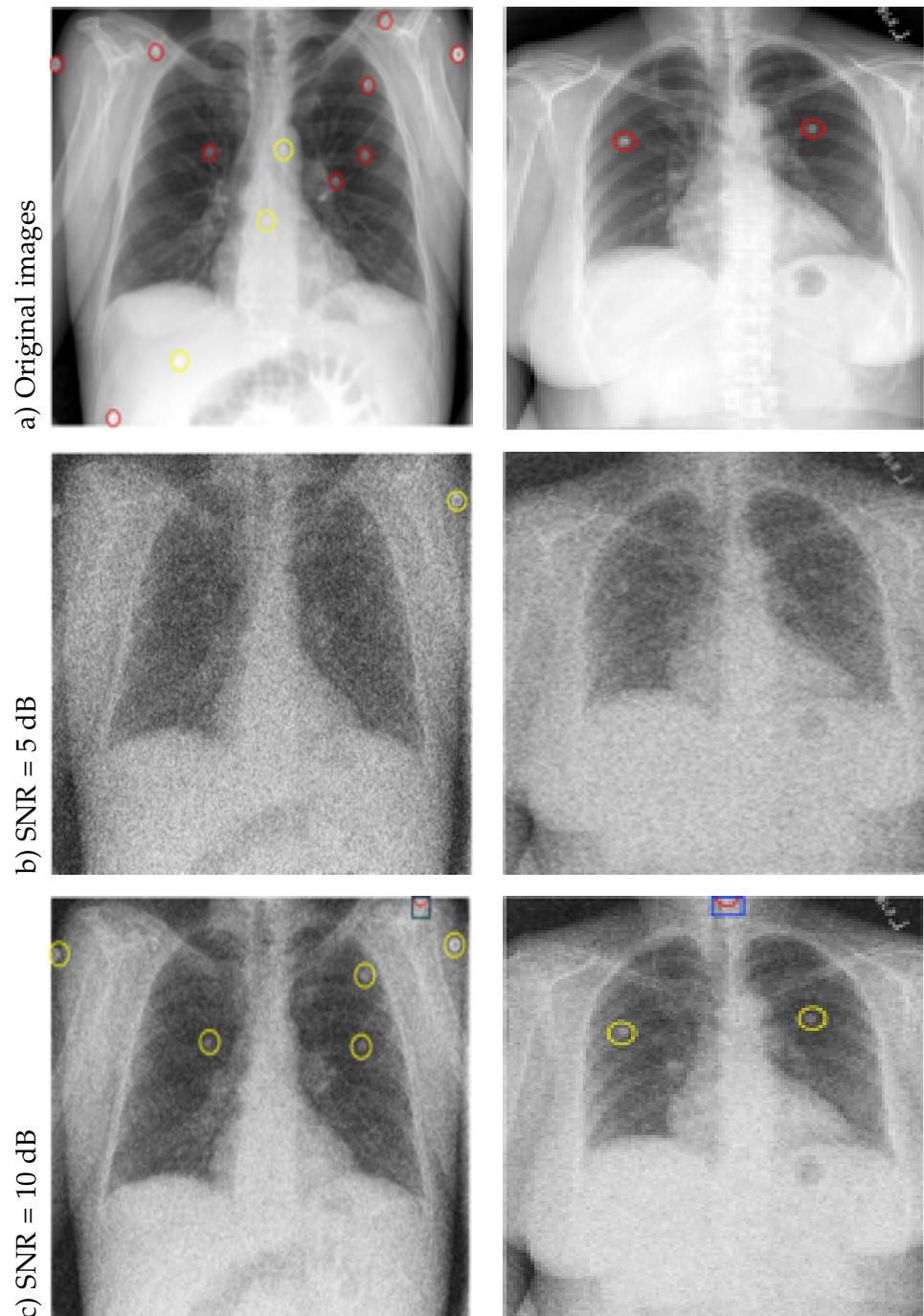


Figure 4.10: Detection of NCC with adding Gaussian noise: continue

4.4.2 Robustness check

For checking the robustness of our algorithm, we add signal to noise ratio (SNR in dB) with increasing order of Gaussian noise in original image due to observe the

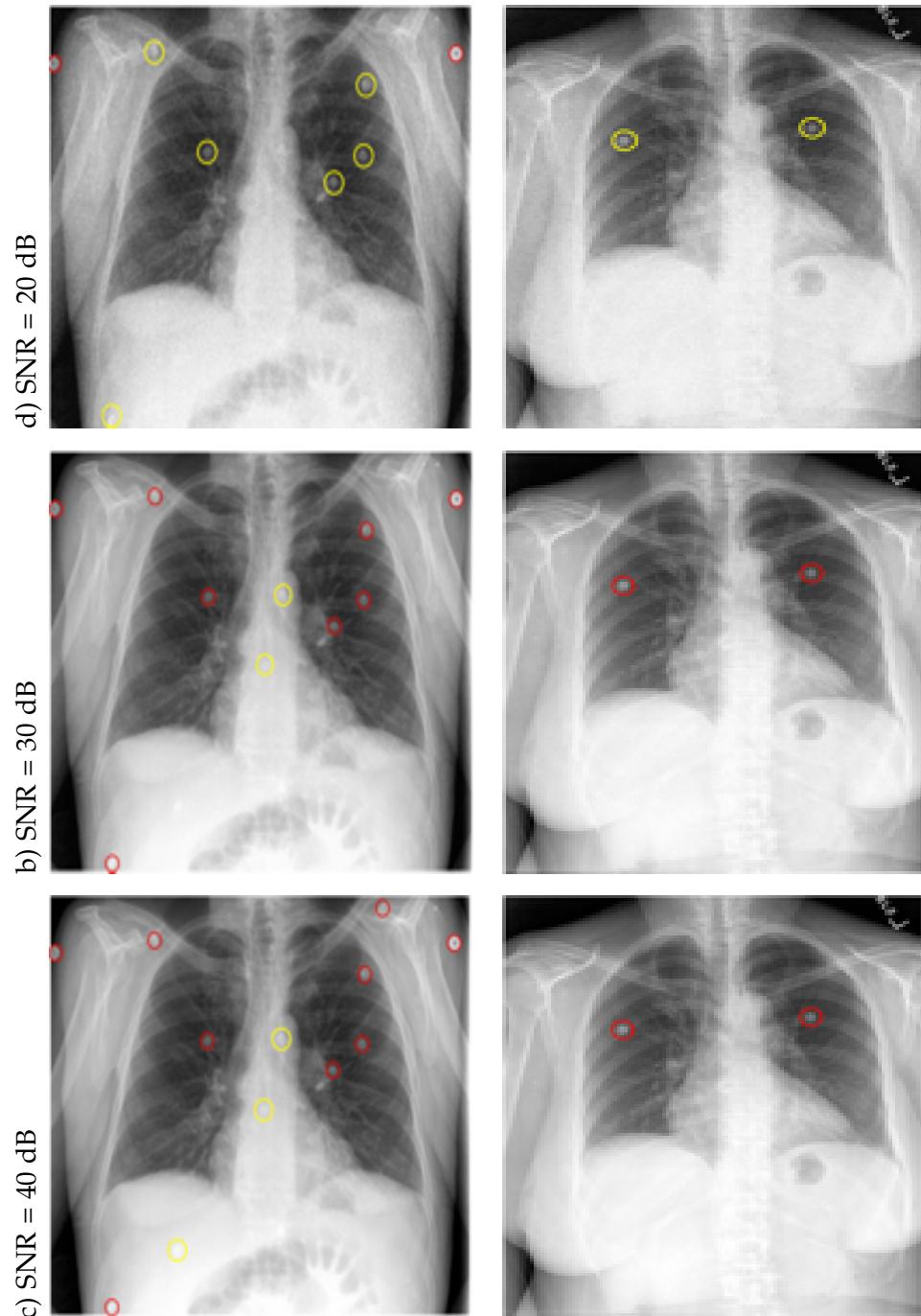


Figure 4.11: Detection of NCC with adding Gaussian noise: continue

performance of the NCC detection technique. For checking the robustness, we test 50 test images. Table 4.11 shows the detection result of NCC with hierarchical clustering technique with adding noise in input test images. Fig 4.9 presents the plot of detection performance (precision, recall, and F1 score) vs SNR plot. From Table 4.11, we can see

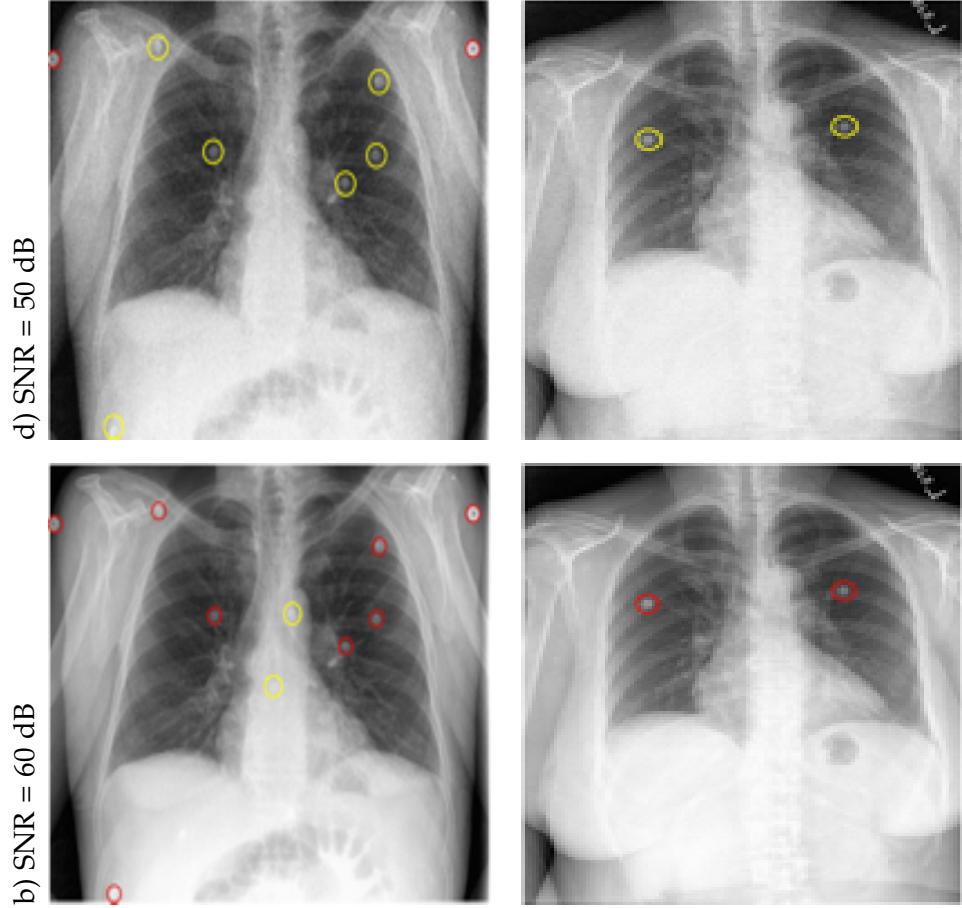


Figure 4.12: Detection of NCC with adding Gaussian noise: circles marked by red, rectangle marked by blue, and circle marked by yellow represent true positive, false positive, and false negative respectively.

that increasing the SNR above 30dB, performance of our technique is similar with the original result. Figs 4.10, 4.11, and 4.11 present the result of circle-like element in CXR images without chest segmentation using varying the SNR value for test dataset.

4.5 Conclusion

In this work, we have focused on identifying circle-like foreign elements in CXR images such as buttons appearing in lung regions of the chest X-ray images. We have presented a novel technique for circular foreign object detection. Our proposed technique is encouraging, both in terms of detection accuracy and computation time. The detection performance of total dataset is quite similar with the detection performance of subset1 dataset. We have shown our result with and without lung segmentation for both datasets.

We can see from our results that using our proposed technique, we have gotten 90%, 93% and 91% precision, recall and F1 score respectively for complete dataset without lung segmentation.. Then, we can say that our proposed technique performed robustly is under a variety of dataset for withing chest region. In our system, precision and recall shows marginal difference with (Xue, Candemir, Antani, Long, Jaeger, Demner-Fushman, and Thoma 2015).

4.6 What's next?

Considering experiment in this chapter, we have achieved substantial performance using NCC with hierarchical clustering in terms of precision, recall, and F1 score regardless of dataset. Inspired by this findings, we want to try similar idea to detect more complex foreign elements in CXR in particular, medical devices (with/without tubes). In the next chapter, we will explain medical devices detection in CXR using such NCC based technique.

CHAPTER 5

Medical devices (with/without tube) foreign elements detection in chest X-ray images using normalized cross correlation based technique

5.1 Summary

In this chapter, we present Normalize cross correlation based technique for detecting medical devices (with/without tube) in CXR images. At first, We detect medical devices without tube and measure detection performance using Dice's coefficient and then calculate precision, recall and F1 score. After that, we perform same technique on medical tubes also. This novel technique is able to excel performance in terms of detection accuracy precision, recall and F1 score for medical devices.

Key topics:

- Image enhancement
- Perform fast Fourier transform and convolution for getting cross-correlation
- Implementing integral image for calculating mean and standard deviation
- Perform normalize cross correlation using positive and negative template
- Perform Sørensen–Dice coefficient for measuring detection overlapping the annotated ground-truth and NCC based resultant output
- Analyze the result for future improvement.

Organization of chapter. The rest of this chapter is structured as follows:

In section 5.2, we will give detailed overview of our proposed technique. Then in subsection 5.2.1, we will explain our decision taking technique. Next, Section 5.3 provides information about the dataset and the evaluation metrics and protocol for performance

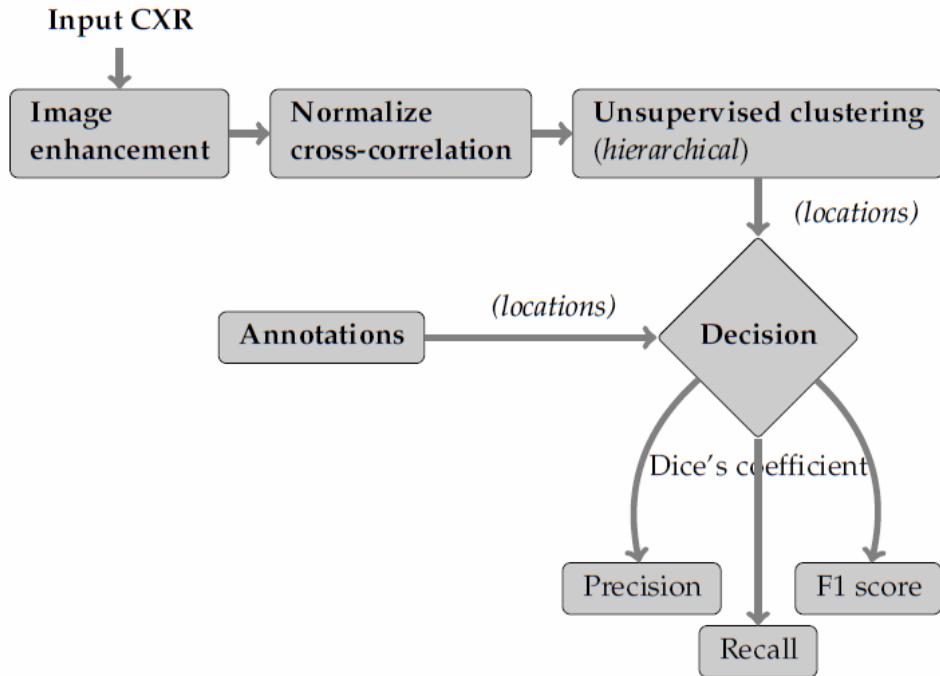


Figure 5.1: The process begins with image enhancement. Then perform normalize cross-correlation and hierarchical clustering and finally for automatic detection, it evaluates result by using Dice's coefficient with annotated ground-truth and clustered output.

measurement and results (see subsection 5.3.3). Section 5.4 concludes the paper. Finally, Section 5.5 describes what is the next chapter.

5.2 Proposed technique

In our proposed algorithm, we first enhance the CXRs to increase the contrast between the button elements with their background, using intensity normalization and image adjustment. Next, we apply normalized cross-correlation step to find the cross-correlation coefficient pixel wise. Then we perform Sørensen–Dice coefficient for measuring detection overlapping the annotated ground-truth and NCC based resultant output for making decision. This proposed technique is quite similar with the chapter 4 proposed technique. For detailed information, please see the chapter 4. Fig. 5.1 briefly presents the proposed method work-flow for medical devices (with/without tube) foreign elements detection using normalized cross correlation based technique. For better understanding, the brief of each step is given below:

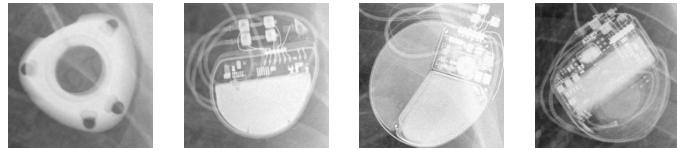


Figure 5.2: Positive training templates for medical devices

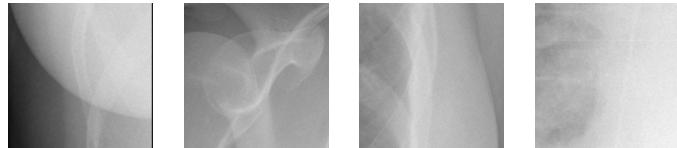


Figure 5.3: Negative training templates for medical devices

- Image enhancement
- Annotations: For annotations of medical devices (with/without tube) foreign elements in CXR images please see Appendix A.2.
- Positive training template selection: For selecting positive training sample, we manually cut the ground-truth 82×82 pixel. Fig. 5.2 and 5.4 show a few positive training templates.
- Normalized cross-correlation
- Negative training template selection: For selecting Negative training sample, we perform similar approach in chapter 4 section 4.3.5 to make it automatic. In this case, we take the negative training template size is $(392 \times 392$ pixel) and $(50 \times 80$ pixel) for medical devices and medical tubes respectively. Fig. 5.3 and 5.5 show a few negative training templates for medical devices and medical tubes respectively.
- Thresholding
- Training samples: positive and negative
- Final NCC output
- Peak detection

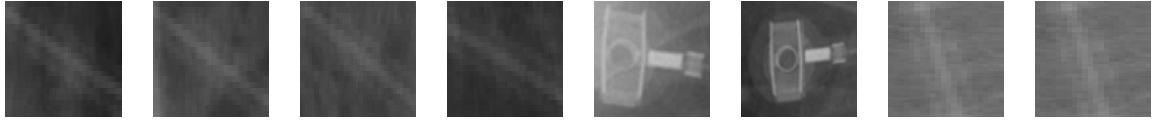


Figure 5.4: Positive training templates for medical tubes



Figure 5.5: Negative training templates for medical tubes

5.2.1 Decision

After detecting, medical devices with/without tubes, we use Sørensen–Dice coefficient to check the overlap between detected CXRs and annotated CXRs. Which (Dice 1945) is a statistics used for comparing the matching of two samples. In our experiment we used following equation for measuring the performance: $Dice\ Coefficient = 2TP/(2TP + FP + FN)$, where TP, FP and FN represent true positive, false positive and false negative respectively. If Sørensen–Dice coefficient score is more than 80% then we consider it as 100% detection. After that we calculate precision, recall and F1 score for measuring the performance result. Fig. 5.6 c) shows the complete view of medical devices foreign element detection in CXR images without lung segmentation. Fig. 5.7 c) shows the complete view of medical tubes foreign element detection in CXR images without lung segmentation.

5.3 Experiments

In our experiments, we use following setup for finding medical devices with/without tubes in CXR images.

5.3.1 Dataset

In our study, we use subset of data-set maintained by National Library of Medicine (NLM) - National Institutes of Health (NIH), which is composed of 400 DICOM images. From this subset of dataset, we take 50 DICOM images as subset1 total of 400 DICOM dataset for our initial experiment. Total number medical devices with/without tubes

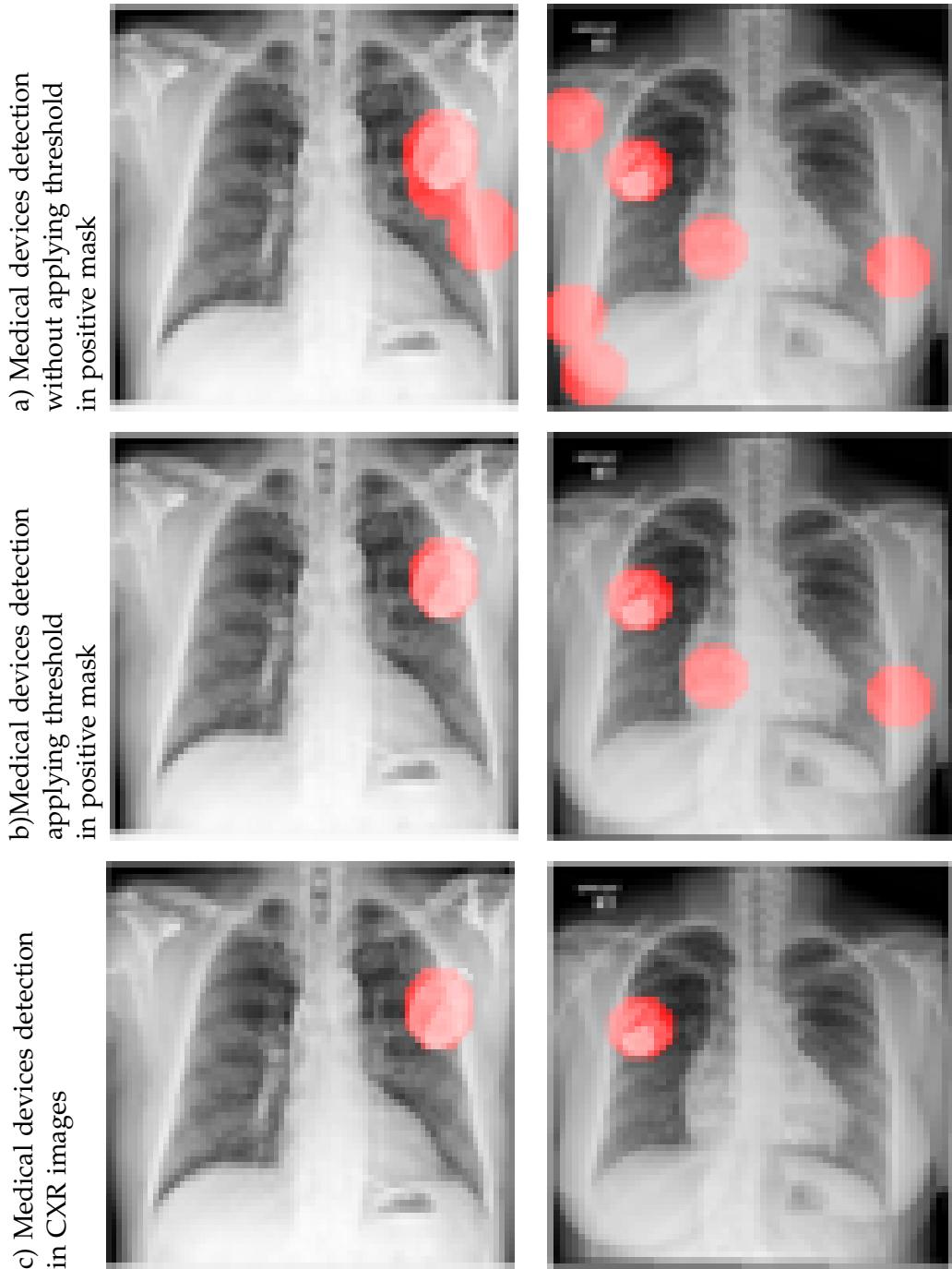


Figure 5.6: Medical devices detection with positive msk with/without thresholding and final mask.

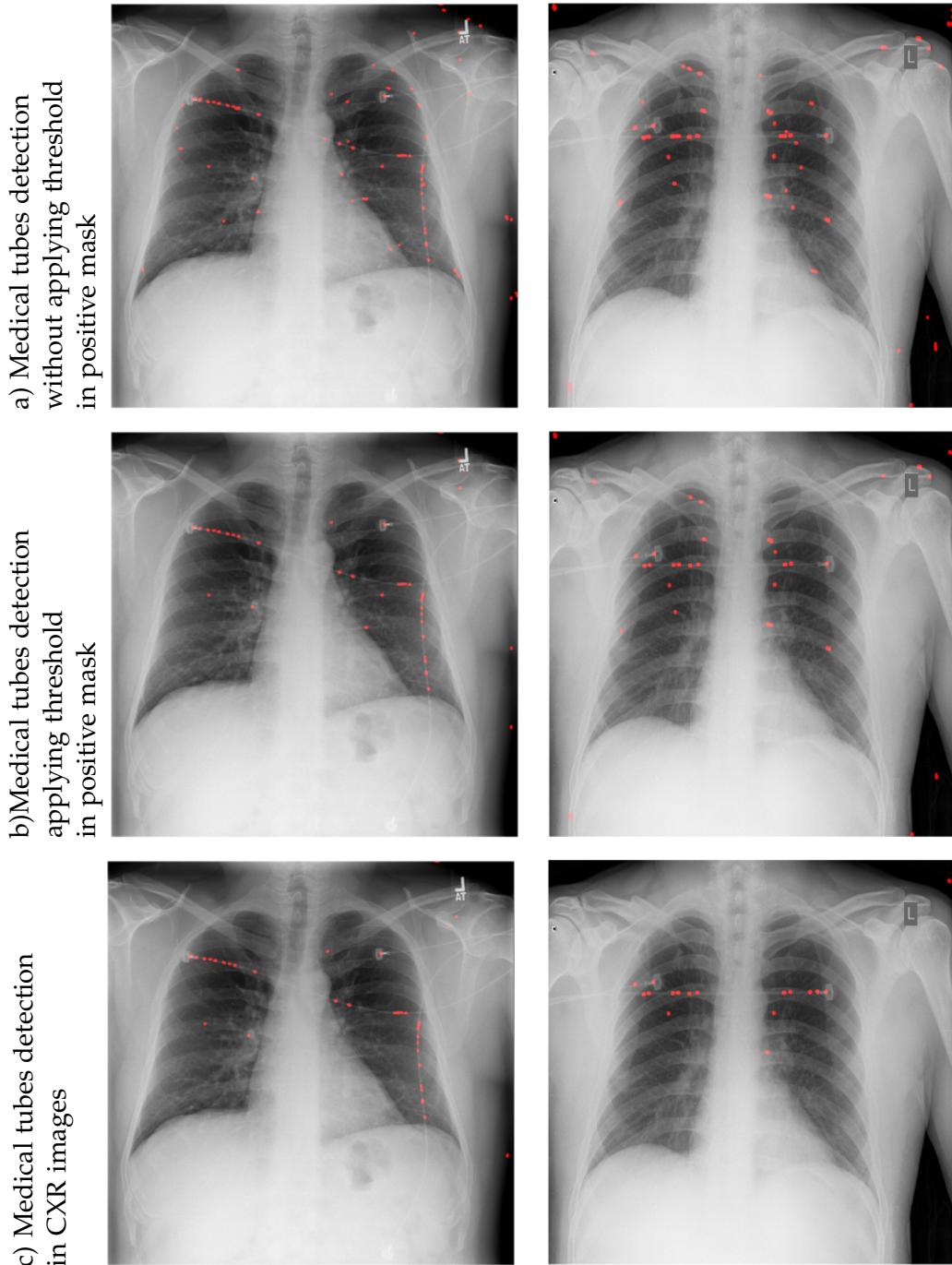


Figure 5.7: Medical tubes detection with positive msk with/without thresholding and final mask.

Table 5.1: Dataset description

# of CXRs	# of medical devices with tubes	# of medical devices without tubes
Subset1 = 50	101	8

Table 5.2: Medical devices with/without tubes detection result for "subset1": NCC based technique

Metrics	Medical devices with tubes	Medical devices without tubes
Ground-truth	101	8
Detection	101	22
TP	61	8
FP	40	14
FN	40	0
Precision	0.78	0.36
Recall	0.60	1
F1 score	0.68	0.53

is presented in Table 5.1 all over the CXR images. We annotate corresponding ground-truths, since they are not available in this data set (see Appendix A.2).

5.3.2 Evaluation metrics and protocol

We use evaluation metrics as precision, recall and F1 score which is similar in chapter three (see Subsection 3.4.2) including Sørensen–Dice coefficient.

5.3.3 Results and analysis

In our proposed training based technique, we apply normalized cross-correlation with hierarchical clustering approach to detect the medical devices with/without tubes detection in CXR images. Using our positive and negative sample for creating final output and after that applying hierarchical clustering technique, we improve the precision of the Medical devices with/without tubes detection. Fig 5.6 and 5.7 c) show the final detection result of NCC with hierarchical clustering technique of Medical devices with/without tubes detection respectively. Table 5.2 presents the result of medical devices with/without tubes detection in CXR images for subset1 dataset.

5.4 Conclusion

In this work, we have focused on identifying medical devices with/without tubes in CXR images of the chest X-ray images. We have presented a novel technique for circular medical devices with/without tubes like foreign element detection in CXR images. We can see from our results that using our proposed technique, we have got 100% recall for medical devices without tubes for subset1 dataset. On the other hand, for medical devices with tubes, the detection performance falls due to the irregular shape of the tubes and intensity difference between background and our desire medical tubes like elements in CXR images.

5.5 What's next?

Considering experiment in this chapter, we have achieved substantial performance using NCC with hierarchical clustering in terms of precision, recall, and F1 score for medical devices without considering the tubes for subset1. In the next chapter, we will conclude our thesis work.

CHAPTER 6

Conclusion

In this thesis work, I have focused on identifying foreign elements (circle-like elements-buttons, coins, medical devices (with/without tubes) foreign elements in the chest X-ray images. I have presented two novel techniques for foreign elements detection in CXR images. In the first circular assumption based - candidate selection followed by circular Hough transform (CHT) technique, is able to excel performance in terms of detection accuracy precision, recall, F1 score and computation time. And second training based technique: normalized cross correlation (using positive and negative template) and unsupervised clustering (hierarchical clustering) for foreign element detection in CXR images is applicable for circle-like foreign elements regardless chest region and medical devices with/without tubus in CXR images. It performs robustly for circle-like foreign elements and medical devices without tubes in CXR images. As future work, I will plan to test the robustness of our algorithms on larger data sets for medical devices without tubes and extend this work to improve the detection performance of medical tubes in CXR images.

References

- Abe, Y., K. Hanai, M. Nakano, Y. Ohkubo, T. Hasizume, T. Kakizaki, M. Nakamura, N. Niki, K. Eguchi, T. Fujino, et al. (2005). A computer-aided diagnosis (cad) system in lung cancer screening with computed tomography. *Anticancer research* 25(1B), 483–488.
- Antani, S. (2015). Automatic x-ray screening for tuberculosis.
- Atherton, T. J. and D. J. Kerbyson (1999). Size invariant circle detection. *Image and Vision computing* 17(11), 795–803.
- Bland, J. M. and D. G. Altman (1996). Statistics notes: measurement error. *Bmj* 313(7059), 744.
- Bracewell, R. (1965). The fourier transform and its applications. New York 5.
- Candemir, S., S. Jaeger, K. Palaniappan, J. P. Musco, R. K. Singh, Z. Xue, A. Karargyris, S. Antani, G. Thoma, and C. J. McDonald (2014). Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. *IEEE transactions on medical imaging* 33(2), 577–590.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence* (6), 679–698.
- Chan, H.-P., L. Hadjiiski, C. Zhou, and B. Sahiner (2008). Computer-aided diagnosis of lung cancer and pulmonary embolism in computed tomography—a review. *Academic radiology* 15(5), 535–555.
- Crow, F. C. (1984). Summed-area tables for texture mapping. *ACM SIGGRAPH computer graphics* 18(3), 207–212.
- Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology* 26(3), 297–302.
- El-Baz, A., G. M. Beach, G. Gimel'farb, K. Suzuki, K. Okada, A. Elnakib, A. Soliman, and B. Abdollahi (2013). Computer-aided diagnosis systems for lung cancer: challenges and methodologies. *International journal of biomedical imaging* 2013.
- Firmino, M., G. Angelo, H. Morais, M. R. Dantas, and R. Valentim (2016). Computer-

- aided detection (cade) and diagnosis (cadx) system for lung cancer with likelihood of malignancy. *Biomedical engineering online* 15(1), 1.
- Firmino, M., A. H. Morais, R. M. Mendoça, M. R. Dantas, H. R. Hekis, and R. Valentim (2014). Computer-aided detection system for lung cancer in computed tomography scans: Review and future prospects. *Biomedical engineering online* 13(1), 1.
- Gauss, C. F. (1816). Bestimmung der genauigkeit der beobachtungen. *Astronomi* 1, 185–197.
- Gorard, S. (2005). Revisiting a 90-year-old debate: the advantages of the mean deviation. *British Journal of Educational Studies* 53(4), 417–430.
- Jaeger, S., A. Karargyris, S. Candemir, L. Folio, J. Siegelman, F. Callaghan, Z. Xue, K. Palaniappan, R. K. Singh, S. Antani, et al. (2014). Automatic tuberculosis screening using chest radiographs. *IEEE transactions on medical imaging* 33(2), 233–245.
- Jaeger, S., A. Karargyris, S. Candemir, J. Siegelman, L. Folio, S. Antani, G. Thoma, and C. J. McDonald (2013). Automatic screening for tuberculosis in chest radiographs: a survey. *Quantitative imaging in medicine and surgery* 3(2), 89–99.
- Karargyris, A., J. Siegelman, D. Tzortzis, S. Jaeger, S. Candemir, Z. Xue, K. Santosh, S. Vajda, S. Antani, L. Folio, et al. (2016). Combination of texture and shape features to detect pulmonary abnormalities in digital chest x-rays. *International journal of computer assisted radiology and surgery* 11(1), 99–106.
- Lawrence, G. R. (1963). *Machine perception of three-dimensional solids*. Ph. D. thesis, Ph. D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA.
- Lewis, J. Fast normalized cross-correlation, vision interface. 1995. *Google Scholar*, 120–3.
- Maimon, O. and L. Rokach (2005). *Data mining and knowledge discovery handbook*, Volume 2. Springer.
- Powers, D. M. (2011). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.
- Prewitt, J. Object enhancement and extraction picture processing and psychopictorics. 1970. *A* 3(3).
- Santosh, K., S. Vajda, S. Antani, and G. R. Thoma (2016). Edge map analysis in chest x-rays for automatic pulmonary abnormality screening. *International journal of com-*

- puter assisted radiology and surgery*, 1–10.
- Schaefer-Prokop, C., U. Neitzel, H. W. Venema, M. Uffmann, and M. Prokop (2008). Digital chest radiography: an update on modern technology, dose containment and control of image quality. *European radiology* 18(9), 1818–1830.
- Smith, S. W. et al. (1997). The scientist and engineer's guide to digital signal processing.
- Sobel, I. (2014). History and definition of the sobel operator. *Retrieved from the World Wide Web*.
- Soille, P. (2013). *Morphological image analysis: principles and applications*. Springer Science & Business Media.
- Van Ginneken, B., B. T. H. Romeny, and M. A. Viergever (2001). Computer-aided diagnosis in chest radiography: a survey. *IEEE Transactions on medical imaging* 20(12), 1228–1241.
- Viola, P. and M. Jones (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, Volume 1, pp. I–511. IEEE.
- Walker, H. M. (1929). Studies in the history of statistical method: With special reference to certain educational problems.
- Weisstein, E. W. (2002a). Discrete fourier transform.
- Weisstein, E. W. (2002b). Discrete fourier transform.
- WHO (2014). World Health Organization (WHO).: global tuberculosis report.
- Xue, Z., S. Candemir, S. Antani, L. R. Long, S. Jaeger, D. Demner-Fushman, and G. R. Thoma (2015). Foreign object detection in chest x-rays. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, pp. 956–961. IEEE.

Appendices

APPENDIX A

Annotations

A.1 Circle-like foreign elements in chest X-ray images

For annotating the ground truth in CXR images at first we analyze our data-set for setting a approximate radius of cricle-like element. We observe that the radius of the circle is approximately 50 pixels. We set 50 pixel as our radius of circle-like element and draw circle corresponding circle-like element in our data-set and save the location of the center of those element as annotated ground-truth. Fig A.1 and Fig A.2 show such original and annotated chest X-ray images.

A.2 Medical devices in chest X-ray images

For annotating the medical devices with/without ground truth in CXR images at first we label the foreign element in in MaTLab and save the position of those corresponding labeled pixel. For creating annotated image, we plot those pixel on our input chest X-ray images. For finding mask CXR images, we subtract annotated CXR images with original CXR images. Fig ?? and Fig ?? show such original and annotated app:mt_annochestX – rayimages.

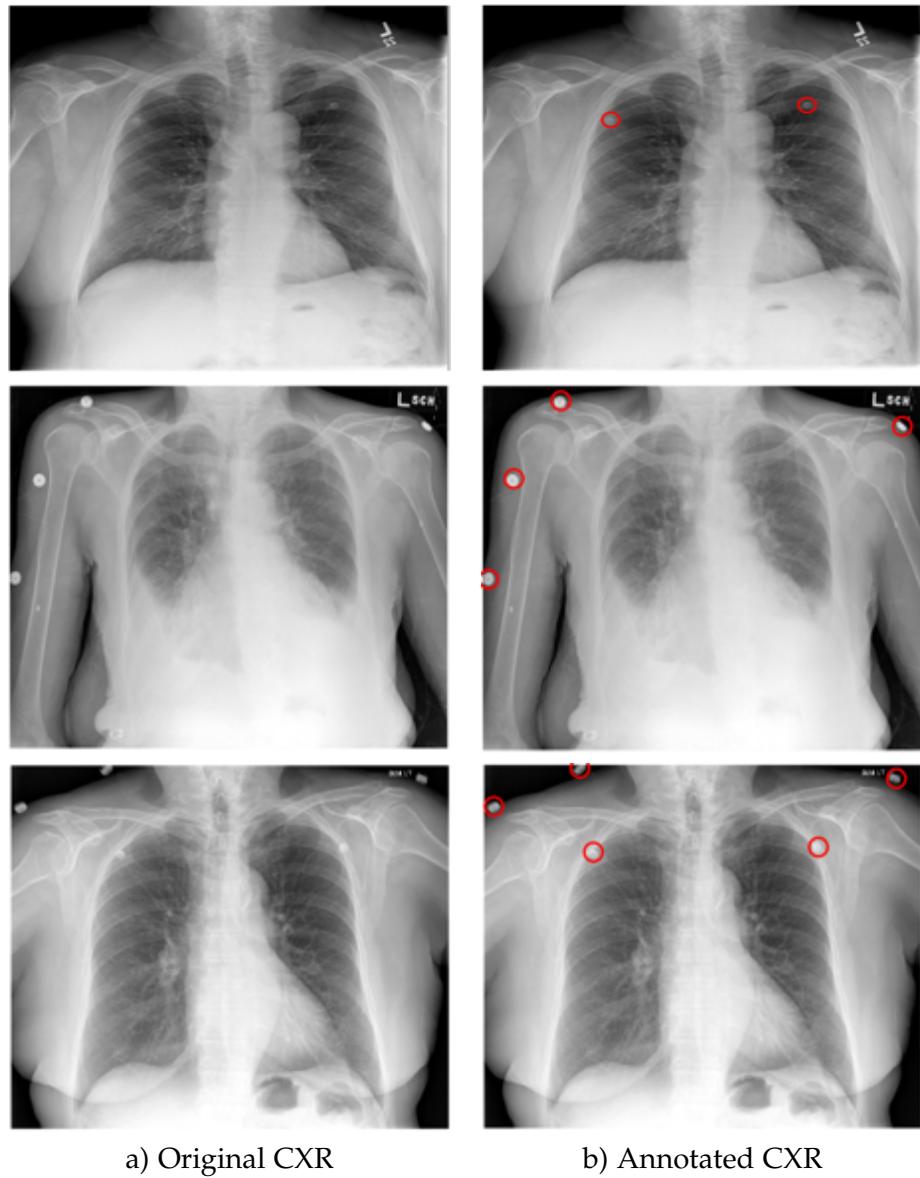


Figure A.1: Annotations 2: Circle-like elements in CXR images: a) and b) show original and annotated CXR image respectively.

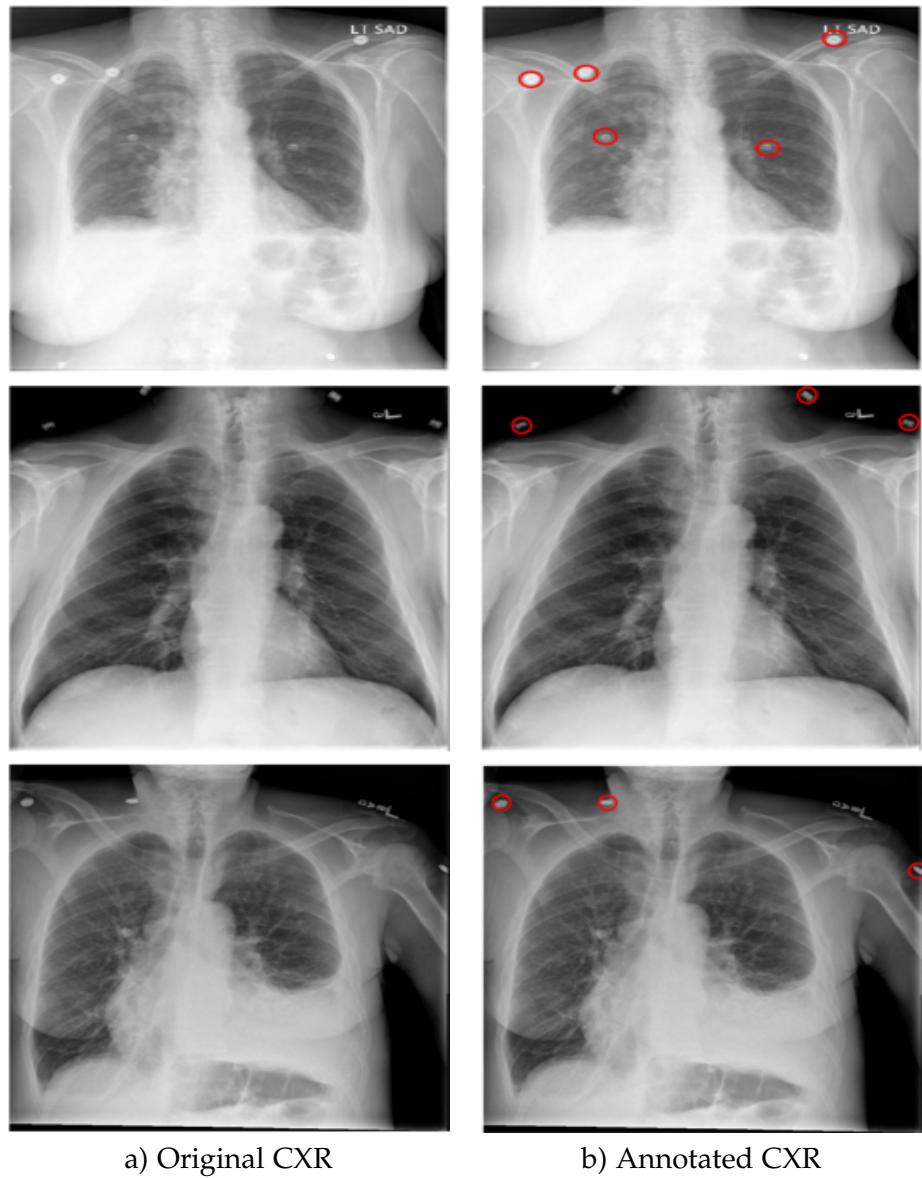


Figure A.2: Annotations 2: Circle-like elements in CXR images: a) and b) show original and annotated CXR image respectively.

APPENDIX B

Source code

```
1 %%%
2 %{
3 author name: fatema tuz zohora
4 date: 04/24/2005
5 purpose: dicom image preprocessing and enhancement: calculate hounsfield ,
6 image adjustment, check photometricinterpretation and convert all imageges
7 into monochrome2.
8 %}
9
10 clear all;
11 close all;
12 %%
13 % initialize path
14 work_dir = pwd;
15 %img_folder ='circle_DICOM';
16 img_folder ='medical_DICOM';
17 img_dir = fullfile(work_dir, img_folder);
18 img_list = dir(fullfile(img_dir, '*.dcm'));
19 img_length = length(img_list);
20
21 for no = 1 : img_length
22     file_name = img_list(no).name;
23     [pathstr , name, ext] = fileparts(file_name);
24     display(['Processing ' file_name]);
25
26 % read image data
27 img = fullfile ( img_dir, strcat (name, ext) );
28 info = dicominfo(img);
29 X = double(dicomread(info));
30 %
31 % figure ,imshow(X,'DisplayRange ',[]);
```

```

32 % imwrite(uint8(255 * mat2gray(X)), [ 'cht/' name '.png'])

33

34 %Calculate Hounsfield Units

35 if(isfield(info, 'WindowCenter') && isfield(info, 'WindowWidth'))
36     low = info.WindowCenter - info.WindowWidth/2;
37     high = info.WindowCenter + info.WindowWidth/2;
38     img_adj = X;
39     img_adj(img_adj < low) = low;
40     img_adj(img_adj > high) = high;
41     img_adj = img_adj - low;
42     img_adj = img_adj / (high - low);

43 else
44     img_adj = imadjust((X - min(X(:))) / (max(X(:)) - min(X(:))));
45 end

46

47 %monochrome1 checking, if true then reverse the image to monochrome2
48 %image

49 if(info.PhotometricInterpretation == 'MONOCHROME1')
50     img_adj = 1 - img_adj;
51 end

52

53 %figure,imshow(img_adj, []);

54

55 %imwrite(img_adj, [ 'cir_op/' name '.jpg'])
56 imwrite(img_adj, [ 'medical_DICOM_mt/' name '.jpg'])

57 end

```

Listing B.1: Image enhancement for CXR images

```

1 %%%
2 %{
3 author name: fatema tuz zohora
4 date: 04/24/2005
5 purpose: circle detection using circular hough transform and checked all
6 parameters in that function.
7 imdistline , for calculate the distance of the circle.
8 %}
9 %%%
10 clear all;
11 close all;
12
13 work_dir = pwd;
14 % img_folder = '..\circle_data_set.jpg';
15 img_folder = '..\cir_data_set_50';
16 img_dir = fullfile(work_dir, img_folder);
17 img_list = dir(fullfile(img_dir, '*.jpg'));
18 img_length = length(img_list);
19
20 % img_folder_chest_seg = 'mask';
21 % img_dir_chest_seg = fullfile(work_dir, img_folder_chest_seg);
22 % img_list_chest_seg = dir(fullfile(img_dir_chest_seg, '*.png'));
23
24 for no = 1: img_length;
25     file_name = img_list(no).name;
26     [pathstr, name, ext] = fileparts(file_name);
27     display(['Processing ' file_name]);
28     input_img = fullfile ( img_dir, strcat (name, ext) );
29     img = imread(input_img);
30
31 %img = rgb2gray(img1);
32 %{
33 figure(no), imshow(img);
34 hold on;
35 %}
36

```

```

37 file_name_chest_seg_img = img_list_chest_seg(no).name;
38 [pathstr, name, ext] = fileparts(file_name_chest_seg_img);
39 display(['Processing ' file_name_chest_seg_img]);
40 input_img_chest_seg = fullfile ( img_dir_chest_seg , strcat (name, ext) );
41 img_mask = imread(input_img_chest_seg);

42

43

44 img_edg = edge(img, 'Prewitt');
45 img_edg = edge(img, 'Sobel');
46 img_edg = edge(img, 'Canny');
47 img_edg = edge(img, 'Roberts');

48

49 %{
50 %using two stage method
51 [centersBright1, radiiBright1] = imfindcircles(img,[20 45], 'ObjectPolarity',
52 'bright', 'method', 'TwoStage');
53 [centersBright, radiiBright] = imfindcircles(img,[46 55], 'ObjectPolarity',
54 'bright', 'method', 'TwoStage');
55 %}
56
57 %using two phase code method which is default method.
58 [centersBright0, radiiBright0] = imfindcircles(img,[12 19], 'ObjectPolarity',
59 'bright', 'EdgeThreshold', .08);
60 [centersBright1, radiiBright1] = imfindcircles(img,[25 45], 'ObjectPolarity',
61 'bright', 'EdgeThreshold', .02);
62 [centersBright2, radiiBright2] = imfindcircles(img,[46 60], 'ObjectPolarity',
63 'bright', 'EdgeThreshold', .02);

64 centersBright = [centersBright0; centersBright1; centersBright2];
65 radiiBright = [radiiBright0; radiiBright1; radiiBright2];

66 idx = [];
67 for i=1:length(radiiBright)
68     if(img_mask(round(centersBright(i,2)), round(centersBright(i,1))) == 0)
69         idx = [idx, i];

```

```

68     end
69 end
70
71 centersBright(idx,:) = [];
72 radiiBright(idx) = [];
73
74 se90 = strel('line', 7, 90);
75 se0 = strel('line', 7, 0);
76 BW1 = edge(img_mask, 'canny');
77 BW2 = imdilate(BW1, [se90 se0]);
78
79 img_w_plot = uint8(zeros(size(img,1), size(img,2), 3));
80 img_w_plot(:,:,1) = img;
81 img_w_plot(:,:,2) = img;
82 img_w_plot(:,:,3) = img;
83
84 img2 = img;
85 img2(BW2 == 1) = 255;
86 img_w_plot(:,:,1) = img2;
87 img2(BW2 == 1) = 0;
88 img_w_plot(:,:,2) = img2;
89 img_w_plot(:,:,3) = img2;
90
91 figure(no), imshow(img);
92 hold on;
93 %create circle
94 viscircles(centersBright, radiiBright, 'EdgeColor','g','LineWidth',1.2);
95 saveas(gcf,['op_cht/' name '.png']);
96
97
98 end

```

Listing B.2: Circular Hough transform (CHT)

```
1 %
2 %
3 clear all, close all;
4
5 tic
6 %%
7 % base code with all image
8
9 work_dir = pwd;
10 img_folder = '..\cir_data_set_50';
11 img_dir = fullfile(work_dir, img_folder);
12 img_list = dir(fullfile(img_dir, '*.jpg'));
13 img_length = length(img_list);
14
15 load('labelingSession_150.mat');
16 %load('labelingSession_png_10.mat');
17 %pos_img ='pos.image';
18 pos_img ='pos_image_real';
19 pos_img_dir = fullfile(work_dir, pos_img);
20 addpath(pos_img_dir);
21 %negative_folder = fullfile(work_dir, 'neg_image');
22 negative_folder = fullfile(work_dir, 'neg_image_with_word');
23
24 % use one at a time for checking the performance
25 trainCascadeObjectDetector('labelingSession_150.xml',labelingSession.ImageSet.
    ImageStruct,negative_folder,'FalseAlarmRate',0.01,'NumCascadeStages',15,
    'FeatureType','Haar','TruePositiveRate',0.999 );
26 detector = vision.CascadeObjectDetector('labelingSession_150.xml');
27
28
29 trainCascadeObjectDetector('labelingSession_with_lung_seg_10.xml',
    labelingSession.ImageSet.ImageStruct,negative_folder,'FalseAlarmRate',0.01,
    'NumCascadeStages',15,'FeatureType','Haar','TruePositiveRate',0.999 );
30 detector = vision.CascadeObjectDetector('labelingSession_with_lung_seg_10.xml');
31
32
```

```

33 trainCascadeObjectDetector('labelingSession_only_chest_area_10.xml',
34   labelingSession.ImageSet.ImageStruct,negative_folder,'FalseAlarmRate',0.01,
35   'NumCascadeStages',15, 'FeatureType', 'Haar', 'TruePositiveRate',0.999 );
36
37 detector = vision.CascadeObjectDetector('labelingSession_only_chest_area_10.xml'
38   );
39
40
41
42
43
44
45
46
47
48
49
50
51
52 %}

```

Listing B.3: Viola-Jones algorithm for circle-like button detection in CXR images

```

1 %%%
2 %{
3 author name: fatema tuz zohora
4 date: 04/24/2005
5 purpose: circle detection using circular hough transform and checked all
6 parameters in that function.
7
8 http://www.mathworks.com/help/images/examples/detecting-a-cell-using-image-
   segmentation.html
9 link: http://www.mathworks.com/help/images/ref/imfindcircles.html?refresh=true
10 http://www.mathworks.com/help/images/ref/viscircles.html?searchHighlight=
      viscircles
11 http://www.mathworks.com/help/images/ref/imdistline.html
12 imdistline, for calculate the distance of the circle.
13 %}
14
15 %%
16 clear all, close all;
17
18 work_dir = pwd;
19 % img_folder = '..\circle_data_set.jpg';
20 img_folder = '..\cir_data_set_50';
21 img_dir = fullfile(work_dir, img_folder);
22 img_list = dir(fullfile(img_dir, '*.jpg'));
23 img_length = length(img_list);
24
25 img_folder_chest_seg = 'mask';
26 img_dir_chest_seg = fullfile(work_dir, img_folder_chest_seg);
27 img_list_chest_seg = dir(fullfile(img_dir_chest_seg, '*.png'));
28
29 for no = 1: img_length
30     file_name = img_list(no).name;
31     [pathstr, name, ext] = fileparts(file_name);
32     display(['Processing ' file_name]);
33     input_img = fullfile ( img_dir, strcat (name, ext) );
34     img = imread(input_img);

```

```

35
36 %img = rgb2gray(img1);
37 %
38 figure(no), imshow(img);
39 hold on;
40 %

41
42 file_name_chest_seg_img = img_list_chest_seg(no).name;
43 [pathstr, name, ext] = fileparts(file_name_chest_seg_img);
44 display(['Processing ' file_name_chest_seg_img]);
45 input_img_chest_seg = fullfile ( img_dir_chest_seg , strcat (name, ext) );
46 img_mask = imread(input_img_chest_seg);

47
48

49 [~, threshold] = edge(img, 'Prewitt');
50 fudgeFactor = .7;
51 BWs = edge(img, 'Prewitt', threshold * fudgeFactor);
52 figure, imshow(BWs);
53 , title('binary gradient mask');

54
55 saveas(gcf,[ 'img_report1/' name '.png']);

56
57 se90 = strel('line',3,90);
58 % se0 = strel('line', 3,0);
59 %% BWsdil = imdilate(BWs, [se90 se0]);
60 %% figure, imshow(BWsdil);%, title('dilated gradient mask');
61 %% saveas(gcf,[ 'img_report2/' name '.png']);

62
63 BWdfill = imfill(BWsdil, 'holes');
64 figure, imshow(BWdfill); title('binary image with filled holes');
65 saveas(gcf,[ 'img_report3/' name '.png']);

66
67 BWnobord = imclearborder(BWdfill, 4);
68 figure, imshow(BWnobord), title('cleared border image');
69 saveas(gcf,[ 'img_report4/' name '.png']);
70

```

```

71 seD = strel('diamond',1);
72 BWfinal = imerode(BWnobord,seD);
73 BWfinal = imerode(BWfinal,seD);
74 figure, imshow(BWfinal), title('segmented image');
75 saveas(gcf,[ 'img_report5/' name '.png']);
76
77 'method','TwoStage'PhaseCode 'Sensitivity'; 'Sensitivity',[0-1], 'LineWidth'
78 ' ,4
79 %
80 %using two stage method
81 [centersBright1, radiiBright1] = imfindcircles(img,[20 45], 'ObjectPolarity',
82 'bright', 'method','TwoStage');
83 [centersBright, radiiBright] = imfindcircles(img,[46 55], 'ObjectPolarity',
84 'bright', 'method','TwoStage');
85 %
86 %using two phase code method which is default method.
87 [centersBright0, radiiBright0] = imfindcircles(BWfinal,[12 19], 'ObjectPolarity',
88 'bright', 'EdgeThreshold', .08);
89 [centersBright1, radiiBright1] = imfindcircles(BWfinal,[25 45], 'ObjectPolarity',
90 'bright', 'EdgeThreshold', .02);
91 [centersBright2, radiiBright2] = imfindcircles(BWfinal,[46 60], 'ObjectPolarity',
92 'bright', 'EdgeThreshold', .02);
93
94 centersBright = [centersBright0; centersBright1; centersBright2];
95 radiiBright = [radiiBright0; radiiBright1; radiiBright2];
96
97 idx = [];
98 for i=1:length(radiiBright)
99     if(img_mask(round(centersBright(i,2)), round(centersBright(i,1))) == 0)
100         idx = [idx, i];
101     end
102 end

```

```

101
102 centersBright(idx,:) = [];
103 radiiBright(idx) = [];
104
105 img2 = img;
106 img2(img_mask == 0) = 255;
107
108
109 figure(no), imshow(img);
110 hold on;
111 %create circle
112 viscircles(centersBright, radiiBright, 'EdgeColor','g', 'LineWidth',1.2);
113 viscircles(centersBright, radiiBright, 'EdgeColor','r', 'LineWidth',1);
114
115 figure(no), imshow(img_chest_seg);
116 hold on;
117
118 %create circle
119 viscircles(centersBright1, radiiBright1, 'EdgeColor','r', 'LineWidth',1);
120 viscircles(centersBright, radiiBright, 'EdgeColor','r', 'LineWidth',1);
121 %
122
123 saveas(gcf,[ 'op_img_seg_cht/' name '.png']);
124
125 end

```

Listing B.4: Image segmentation followed by circular Hough transform

```

1 %%/%%/%%/%%
2
3 clear all;
4 close all;
5
6 work_dir = pwd;
7 img_folder = '..\cir_data_set_50';
8 img_dir = fullfile(work_dir, img_folder);
9 img_list = dir(fullfile(img_dir, '*.jpg'));
10 img_length = length(img_list);
11
12 pos_img_folder = 'pos_train.img';
13 pos_img_dir = fullfile(work_dir, pos_img_folder);
14 pos_img_list = dir(fullfile(pos_img_dir, '*.jpg'));
15 pos_img_length = length(pos_img_list);
16
17 neg_img_folder = 'neg_train.img';
18 neg_img_dir = fullfile(work_dir, neg_img_folder);
19 neg_img_list = dir(fullfile(neg_img_dir, '*.jpg'));
20 neg_img_length = length(neg_img_list);
21
22 %load annotation
23 load('annotations.mat');
24 dist_ = [];
25 tp = 0;
26 fp = 0;
27 fn = 0;
28 neg_counter = 0;
29 save_neg_samples = false; %true
30
31 for j=1:img_length
32
33     file_name = img_list(j).name;
34     [pathstr, name, ext] = fileparts(file_name);
35     display(['.... ' file_name]);
36     img = fullfile (img_dir, strcat (name, ext) );

```

```

37 I = imread(img);
38
39 num = 5;
40
41 %pos images
42 pos_thres = 0.32;
43 pos_mask = zeros(size(I));
44 pos_mask1 = zeros(size(I));
45 pos_scores = -100*ones(size(I));
46
47 for i=1:pos_img_length
48     pos_file_name = pos_img_list(i).name;
49     [pathstr, pos_name, ext] = fileparts(pos_file_name);
50     display(['Processing ' pos_file_name]);
51     pos.img = fullfile ( pos_img_dir, strcat (pos_name, ext) );
52     pos_train_img = imread(pos.img);
53     % Calculate SSD and NCC between Template and Image
54     [~,I_NCC]=template_matching(pos_train_img,I);
55     % Find maximum correspondence in LSDD image
56     s_ncc = sort(I_NCC(:), 'descend');
57
58     pos_mask1 = pos_mask | I_NCC;
59
60     if s_ncc(num) < pos_thres
61         continue;
62     end
63
64     pos_mask = pos_mask | I_NCC > s_ncc(num);
65     pos_scores = max(pos_scores, I_NCC);
66 end
67
68
69 se = strel('disk', 5);
70 pos_mask1 = imdilate(pos_mask1, se);
71 I2 = I; I2(pos_mask1) = 255;
72 I3 = I; I3(pos_mask1) = 0;

```

```

73 I_ = [];
74 I_( :, 1)=I2 ;
75 I_( :, 2)=I ;
76 I_( :, 3)=I ;
77 figure , imshow(uint8(I_)), hold on;
78
79
80 se = strel('disk', 30);
81 pos_mask = imdilate(pos_mask, se);
82 I2 = I; I2(pos_mask) = 255;
83 I3 = I; I3(pos_mask) = 0;
84 I_ = [];
85 I_( :, 1)=I2 ;
86 I_( :, 2)=I ;
87 I_( :, 3)=I ;
88 figure , imshow(uint8(I_)), hold on;
89
90 pos_scores(~pos_mask)=0;
91 mesh(LNCC)
92 saveas(gcf, [ 'output_tm_cir/' name 'LNCC_pos.png']);
93 a = LNCC; a (LNCC<=pos_thres)=0;
94 mesh(a)
95 saveas(gcf, [ 'output_tm_cir/' name 'LNCC_pos_thres.png']);
96 neg_images
97 neg_thres = 0.32;
98 neg_mask = zeros(size(I));
99 neg_mask1 = zeros(size(I));
100 neg_scores = -100*ones(size(I));
101
102 for i=1:neg_img_length
103 neg_file_name = neg_img_list(i).name;
104 [pathstr, neg_name, ext] = fileparts(neg_file_name);
105 display(['Processing ' neg_file_name]);
106 neg_img = fullfile( neg_img_dir, strcat(neg_name, ext) );
107 neg_train_img = imread(neg_img);
108 % Calculate SSD and NCC between Template and Image

```

```

109 [~,I_NCC_]=template_matching(neg_train_img,I);
110 % Find maximum correspondence in LSDD image
111 s_ncc = sort(I_NCC_(:, 'descend');
112 if s_ncc(num) < neg_thres
113     continue;
114 end
115
116 neg_mask = neg_mask | I_NCC_ > neg_thres;
117 neg_mask1 = neg_mask | I_NCC_ > neg_thres;
118 neg_scores = max(neg_scores, I_NCC_);
119 end
120 neg_scores(~neg_mask)=0;
121
122 % mesh(I_NCC_)
123
124 se = strel('disk', 30);
125 neg_mask1 = imdilate(neg_mask1, se);
126 I2 = I; I2(neg_mask1) = 255;
127 I3 = I; I3(neg_mask1) = 0;
128 I_ = [];
129 I_( :, :, 1 )=I2 ;
130 I_( :, :, 2 )=I ;
131 I_( :, :, 3 )=I ;
132 figure , imshow( uint8( I_ )), hold on;
133
134
135 se = strel('disk', 30);
136 neg_mask = imdilate(neg_mask, se);
137 I2 = I; I2(neg_mask) = 255;
138 I3 = I; I3(neg_mask) = 0;
139 I_ = [];
140 I_( :, :, 1 )=I2 ;
141 I_( :, :, 2 )=I ;
142 I_( :, :, 3 )=I ;
143 figure , imshow( uint8( I_ )), hold on;
144
```

```

145 saveas(gcf, [ 'output_tm_cir/' name 'I_NCC_neg.png' ]);
146 n = I_NCC_-; n (I_NCC_-<=neg_thres)=0;
147 mesh(n)
148 saveas(gcf, [ 'output_tm_cir/' name 'I_NCC_neg_thres.png' ]);
149 m= pos_mask + neg_mask*2; imagesc(m)
150 saveas(gcf, [ 'output_tm_cir/' name 'pn_imsec.png' ]);
151 get mask and refine mask
152 mask = (pos_scores > neg_scores);
153
154 imshow(mask)
155
156 %save mask323 mask
157 save mask105 mask
158
159
160 p = cell_access(mask);
161 pp=p'; p = pp(:);
162
163 figure,imshow(I); hold on
164 plot(p(2:2:end),p(1:2:end),'r+','LineWidth',3 )
165
166 length_a = length(annotations{j});
167 length_p= length(p);
168
169 length_a_ = length_a;
170 length_p_ = length_p/2;
171 for m=1:2:length_p
172     flag_pos = false;
173     for n =1:size(annotations{j},1)
174         dist = sqrt((annotations{j}(n,1)- p(m+1))^2 + (annotations{j}(n,2)-
175         p(m) )^2);
176         if(dist < 40)
177             tp = tp + 1;
178             length_a_ = length_a_ - 1;
179             length_p_ = length_p_ - 1;
180             annotations{j}(n,:)=[];

```

```

180         flag-pos = true;
181
182     break;
183 end
184 if flag-pos==false && save-neg-samples
185     r_idx = round(p(m))+(-40:40); r_idx(r_idx<1 | r_idx>size(I,1))=[];
186     c_idx = round(p(m+1))+(-40:40); c_idx(c_idx<1 | c_idx>size(I,2))=[];
187     neg_sample = I(r_idx, c_idx);
188     imwrite(neg_sample,[ neg-img-folder , '\neg-' num2str(neg-counter) , '.jpg']);
189     neg-counter = neg-counter + 1;
190 end
191
192 fn = fn + length_a_;
193 fp = fp + length_p_;
194
195 saveas(gcf, [ 'output_tm_cir/' name '.png']);
196
197 end
198
199 precision = tp/(tp+fp);
200 recall = tp/(tp+fn);
201
202 a = I_NCC; a (I_NCC<=pos_thres)=0;
203 mesh(a)
204
205 a=pos_mask + neg_mask*2; imagesc(a)
206 plot(x,y,'x')

```

Listing B.5: Circle-like foreign element detection using normalized cross-correlation and unsupervised clustering in CXR image

```

1
2
3 function centers = cell_access(mask)
4
5 load mask105 mask
6 centers = [];
7 [x,y] = find(mask ~= 0);
8 mask2 = [x,y];
9 Z = linkage(mask2);%, 'centroid','euclidean');
10 c = cluster(Z,'cutoff',25.7, 'criterion','distance');
11 %p=length(unique(c))

12
13 thres = 5;
14 for i=1:max(c)
15     idx = c==i;
16     if sum(idx) < thres
17         c(idx) = [];
18         mask2(idx,:) = [];
19     else
20         centers = [centers; mean(mask2(idx,:))];
21     end
22 %
23 end
24 plot(x,y,'x');
25 hist(c,max(c));
26
27 imshow(mask)
28 mesh(double(mask2))
29 c = cluster(Z,'cutoff',25.7, 'criterion','distance');

```

Listing B.6: Hierarchical clustering algorithm

```

1 function [I_SSD,I_NCC,Idata]=template_matching(T,I,IdataIn)
2
3 if (nargin<3), IdataIn=[]; end
4
5 % Convert images to double
6 T=double(T); I=double(I);
7 if ( size(T,3)==3)
8     % Color Image detected
9     [I_SSD,I_NCC,Idata]=template_matching_color(T,I,IdataIn);
10 else
11     % Grayscale image or 3D volume
12     [I_SSD,I_NCC,Idata]=template_matching_gray(T,I,IdataIn);
13 end
14
15 function [I_SSD,I_NCC,Idata]=template_matching_color(T,I,IdataIn)
16 if (isempty(IdataIn)), IdataIn.r=[]; IdataIn.g=[]; IdataIn.b=[]; end
17 % Spltie color image, and do template matching on R,G and B image
18 [I_SSD_R,I_NCC_R,Idata.r]=template_matching_gray(T(:,:,1),I(:,:,1),IdataIn.r);
19 [I_SSD_G,I_NCC_G,Idata.g]=template_matching_gray(T(:,:,2),I(:,:,2),IdataIn.g);
20 [I_SSD_B,I_NCC_B,Idata.b]=template_matching_gray(T(:,:,3),I(:,:,3),IdataIn.b);
21 % Combine the results
22 I_SSD=(I_SSD_R+I_SSD_G+I_SSD_B)/3;
23 I_NCC=(I_NCC_R+I_NCC_G+I_NCC_B)/3;
24
25
26 function [I_SSD,I_NCC,Idata]=template_matching_gray(T,I,IdataIn)
27 % Calculate correlation output size = input size + padding template
28 T_size = size(T); I_size = size(I);
29 outsize = I_size + T_size -1;
30 %rot90(T,2)
31 % calculate correlation in frequency domain
32 if (length(T_size)==2)
33     FT = fft2(rot90(T,2),outsize(1),outsize(2));
34     if (isempty(IdataIn))
35         Idata.FI = fft2(I,outsize(1),outsize(2));
36     else

```

```

37     Idata.FI=IdataIn.FI;
38
39     Icorr = real(ifft2(Idata.FI.* FT));
40 else
41     FT = fftn(rot90_3D(T),outsize);
42     FI = fftn(I,outsize);
43     Icorr = real(ifftn(FI.* FT));
44 end
45
46 % Calculate Local Quadratic sum of Image and Template
47 if isempty(IdataIn)
48     Idata.LocalQSumI= local_sum(I.*I,T_size);
49 else
50     Idata.LocalQSumI=IdataIn.LocalQSumI;
51 end
52
53 QSumT = sum(T(:).^2);
54
55 % SSD between template and image
56 I_SSD=Idata.LocalQSumI+QSumT-2*Icorr;
57
58 % Normalize to range 0..1
59 I_SSD=I_SSD-min(I_SSD(:));
60 I_SSD=1-(I_SSD./max(I_SSD(:)));
61
62 % Remove padding
63 I_SSD=unpadarray(I_SSD,size(I));
64
65 if (nargout>1)
66     % Normalized cross correlation STD
67     if isempty(IdataIn)
68         Idata.LocalSumI= local_sum(I,T_size);
69     else
70         Idata.LocalSumI=IdataIn.LocalSumI;
71     end
72

```

```

73 % Standard deviation
74 if (isempty(IdataIn))
75     Idata.stdI=sqrt(max(Idata.LocalQSumI-(Idata.LocalSumI.^2)/numel(T),0));
76 else
77     Idata.stdI=IdataIn.stdI;
78 end
79 stdT=sqrt(numel(T)-1)*std(T(:));
80 % Mean compensation
81 meanIT=Idata.LocalSumI*sum(T(:))/numel(T);
82 I_NCC=(Icorr-meanIT)./(2*stdT*max(Idata.stdI,stdT/1e5));
83
84 % Remove padding
85 I_NCC=unpadarray(I_NCC,size(I));
86 end
87
88 function T=rot90_3D(T)
89 T=flipdim(flipdim(flipdim(T,1),2),3);
90
91 function B=unpadarray(A,Bsize)
92 Bstart=ceil((size(A)-Bsize)/2)+1;
93 Bend=Bstart+Bsize-1;
94 if(ndims(A)==2)
95     B=A(Bstart(1):Bend(1),Bstart(2):Bend(2));
96 elseif(ndims(A)==3)
97     B=A(Bstart(1):Bend(1),Bstart(2):Bend(2),Bstart(3):Bend(3));
98 end
99
100 function local_sum_I=local_sum(I,T_size)
101 % Add padding to the image
102 B=padarray(I,T_size);
103
104 % Calculate for each pixel the sum of the region around it,
105 % with the region the size of the template.
106 if(length(T_size)==2)
107     % 2D localsum
108     s=cumsum(B,1);

```

```
109 c = s(1+T_size(1):end-1,:)-s(1:end-T_size(1)-1,:);
110 s = cumsum(c,2);
111 local_sum_I= s(:,1+T_size(2):end-1)-s(:,1:end-T_size(2)-1);
112 else
113 % 3D Localsum
114 s = cumsum(B,1);
115 c = s(1+T_size(1):end-1,:,:)-s(1:end-T_size(1)-1,:,:);
116 s = cumsum(c,2);
117 c = s(:,1+T_size(2):end-1,:)-s(:,1:end-T_size(2)-1,:);
118 s = cumsum(c,3);
119 local_sum_I = s(:,:,1+T_size(3):end-1)-s(:,:,1:end-T_size(3)-1);
120 end
```

Listing B.7: Normalized cross-correlation

```
1 close all; clear all;
2 work_dir = pwd;
3
4 img_folder = 'orig_img_cir';
5 img_dir = fullfile(work_dir, img_folder);
6 img_list = dir(fullfile(img_dir, '*.tiff'));
7 img_length = length(img_list);
8
9 for no = 1 : img_length
10     file_name = img_list(no).name;
11     [pathstr, name, ext] = fileparts(file_name);
12     display(['Processing ' file_name]);
13     img = fullfile ( img_dir, strcat (name, ext) );
14 %     info = dicominfo(img);
15 %     X = double(dicomread(info));
16 %     figure,imshow(X,'DisplayRange',[ ]);
17     X= imread(img);
18     figure,imshow(X);
19
20     imwrite(X, [ 'op_orig_img_cir/' name '.png' ])
21 end
```

Listing B.8: DICOM to png conversion

```
1
2 close all; clear all;
3 work_dir = pwd;
4 img_folder = 'op_mt';
5 img_dir = fullfile(work_dir, img_folder);
6 img_list = dir(fullfile(img_dir, '*.png'));%,*.jpg'.png
7 img_length = length(img_list);
8
9 for no = 1 : img_length
10     file_name = img_list(no).name;
11     [pathstr, name, ext] = fileparts(file_name);
12     display(['Processing ' file_name]);
13
14 % read image data
15 img = fullfile ( img_dir, strcat (name, ext) );
16
17 X = imread(img);
18 X=imresize(X,0.5);
19
20 imwrite(X,[ 'op_mt/op/' name '.png']);
21
22 end
```

Listing B.9: Image resize

```

1
2
3
4 clear all ,
5 close all;
6 % fullfile matlab built in function.
7 % f = fullfile('myfolder','mysubfolder','myfile.m')
8 % dir - List folder contents;
9 %working directory and length for image
10 workDir = pwd;
11 imgFolder = 'Data_Set_JPG';
12 imgDir = fullfile(workDir, imgFolder);
13 imgList = dir(fullfile(imgDir, '*.jpg'));
14 imgLength = length(imgList);

15
16 %working directory and length for image
17 xmlFolder = 'Data_Set_XML';
18 xmlDir = fullfile(workDir, xmlFolder);
19 xmlList = dir(fullfile(xmlDir, '*.xml'));
20 xmlLength = length(xmlList);
21 info = cell(1, imgLength);

22
23 % Read input image
24 for no = 1 : imgLength
25     fileName = imgList(no).name;
26     [pathstr, name, ext] = fileparts(fileName);
27     display(['Processing ' fileName]);
28     InputImg = fullfile ( imgDir, strcat (name, ext) );
29     img = imread(InputImg);

30
31
32     %org image size er 3d image (rgb); zeros double return kore, we need
33     %char so cast it in uint8
34     %rgb value te graeyy scale value
35     img_w_plot = uint8(zeros(size(img,1), size(img,2)));
36     img_w_plot(:,:,1) = img;

```

```

37 img_w_plot(:,:,2) = img;
38 img_w_plot(:,:,3) = img;
39 figure(no), imshow(img);
40 hold on;

41
42 % Read input xml file
43 xmlName = xmlList(no).name;
44 [pathstr, name, ext] = fileparts(xmlName);
45 display(['Processing ' xmlName]);
46 Inputxml = fullfile ( xmlDir, strcat (name, ext) );
47 xml_name = parseXML(Inputxml);

48
49 %plot xml x,y point on image
50 object_cnt = size(xml_name.Children,2);
51 obj_cell = cell(1,object_cnt-4);
52 for i=5:object_cnt
53     obj = xml_name.Children(i).Children(9);
54     line_cnt = size(obj.Children,2);
55     pt_x = zeros(1,line_cnt-1);
56     pt_y = zeros(1,line_cnt-1);
57     for j = 2:line_cnt
58         x = str2double(obj.Children(j).Children(1).Children.Data);
59         y = str2double(obj.Children(j).Children(2).Children.Data);
60         pt_x(j-1) = x;
61         pt_y(j-1) = y;
62     end
63     pt_x = [pt_x pt_x(1)];
64     pt_y = [pt_y pt_y(1)];
65
66     %obj color assaign kore
67     %color = int32(rand(1,3)*255);

68
69     color = int32(255);

70
71     %pointt -1, loop cholche
72     for j = 2:length(pt_x)

```

```

73      x1 = pt_x(j-1); %line draw x y
74      x2 = pt_x(j);
75      y1 = pt_y(j-1);
76      y2 = pt_y(j);
77      npoints = max(abs(y1-y2), abs(x1-x2))*10; %max koto gula point draw
78      kor
79      if (abs(y1-y2) > abs(x1-x2))
80          dy = 0;
81          dx = 1;
82      else
83          dy = 1;
84          dx = 0;
85      end
86      line = [linspace(x1,x2,npoints); linspace(y1,y2,npoints)]; % plot x1
87      and y1
88      line_width = 2;
89      for k = 1:size(line, 2)
90          xx = round(line(1,k));
91          yy = round(line(2,k));
92          for l = 0:line_width
93              img_w_plot(yy+dy*l,xx+dx*l) = color;
94              img_w_plot(yy+dy*l,xx+dx*l,2) = color;
95              img_w_plot(yy+dy*l,xx+dx*l,3) = color;
96              img_w_plot(yy-dy*l,xx-dx*l) = color;
97              img_w_plot(yy-dy*l,xx-dx*l,2) = color;
98              img_w_plot(yy-dy*l,xx-dx*l,3) = color;
99          end
100      end
101      plot(pt_x, pt_y);
102      obj_cell{i-4} = {pt_x, pt_y};
103  end
104  info{no} = {obj_cell, name};
105  figure,imshow(img_w_plot)
106  imwrite(img_w_plot,[ 'mask_img/' , name, '.jpg']);

```

```
107 end  
108 save info_file info
```

Listing B.10: Medical devices with/without tube in CXR images

```

1 clear all;
2 close all;
3
4 workDir = pwd;
5 imgFolder = 'Data_Set_JPG';
6 imgDir = fullfile(workDir, imgFolder);
7 imgList = dir(fullfile(imgDir, '*.jpg'));
8 imgLength = length(imgList);
9
10 AnFolder = 'annotated.Img';
11 AnDir = fullfile(workDir, AnFolder);
12 AnList = dir(fullfile(imgDir, '*.jpg'));
13 AnLength = length(AnList);
14
15 for no = 1 : imgLength
16     fileName = imgList(no).name;
17     [pathstr, name, ext] = fileparts(fileName);
18     display(['Processing ' fileName]);
19     InputImg = fullfile ( imgDir, strcat (name, ext) );
20     img = imread(InputImg);
21
22     AnfileName = AnList(no).name;
23     [pathstr, name, ext] = fileparts(AnfileName);
24     display(['Processing ' AnfileName]);
25     AnInputImg = fullfile ( AnDir, strcat (name, ext) );
26     Animg = imread(AnInputImg);
27
28     k2 = imsubtract(Animg(:,:,3),img);
29     %imshow(k2);
30     imwrite(k2, ['Img_SSub_annot/' , name, '.jpg']);
31
32 end

```

Listing B.11: Mask creation

```

1 close all; clear all;
2
3 img_path = '../cir_data_set_50/';
4 r = 50;
5
6 files = dir(img_path);
7 file_names = {files(3:end).name};
8 annotations = {};
9 for i = 1:length(file_names)
10     curr_file = file_names{i};
11     im = imread([img_path curr_file]);
12     figure, imshow(im), hold on
13     plot(annotations{j}(:,1), annotations{j}(:,2), 'o');
14
15     saveas(gcf, ['anno_draw/' name '.png']);
16     continue;
17
18     locs = [];
19     a=[];
20     while(1)
21         [x,y,button] = ginput(1);
22         if(button > 1)
23             break;
24         elseif isempty(locs)==0 && (abs(x-locs(end,1)) <= r || abs(y-locs(end,2))
25             ) <= r)
26             pos = [x-r y-r 2*r 2*r];
27             set(a,'Visible','off')
28             a=rectangle('Position',pos,'Curvature',[1 1],'EdgeColor','r',
29             'LineWidth',2);
30             axis equal
31             locs(end,1) = x;
32             locs(end,2) = y;
33         else
34             pos = [x-r y-r 2*r 2*r];
35             a=rectangle('Position',pos,'Curvature',[1 1],'EdgeColor','r',
36             'LineWidth',2);

```

```
34     axis equal
35     locs = [locs; x y];
36
37 end
38 annotations{i} = locs;
39 save annotations.mat annotations;
40 close all;
41 end
```

Listing B.12: Annotaon for circle-like element in chest xray

Listings

B.1	Image enhancement for CXR images	72
B.2	Circular Hough transform (CHT)	74
B.3	Viola-Jones algorithm for circle-like button detection in CXR images	77
B.4	Image segmentation followed by circular Hough transform	79
B.5	Circle-like foreign element detection using normalized cross-correlation and unsupervised clustering in CXR image	83
B.6	Hierarchical clustering algorithm	89
B.7	Normalized cross-correlation	90
B.8	DICOM to png conversion	94
B.9	Image resize	95
B.10	Medical devices with/without tube in CXR images	96
B.11	Mask creation	100
B.12	Annotaon for circle-like element in chest xray	101

Index

- Annotations, 21
 - Canny, 13
 - Chest X-ray (CXR), 1
 - chest X-ray(CXR), 1, 2, 6
 - Circular Hough transform (CHT), 1
 - Convolution, 36
 - Dice's coefficient), 7
 - Discrete Fourier transform (DFT), 35
 - F1 score, 25
 - Fast Fourier transform, 33
 - Integral image, 37
 - intensity normalization, 15, 16, 18
 - k-fold cross-validation, 25
 - Morphological operations, 12, 18
 - Normalized cross-correlation(NCC), 1
 - Prewitt, 14
 - Recall, 24
 - Roberts, 13
 - Sobel, 14
 - Standard deviation, 33
 - unsupervised clustering, 1
 - Viola-Jones, 1

Biographical Notes

Fatema Tuz Zohora is a graduate student in the department of computer science at the University of South Dakota (USD). She earned her BSc in electronics & communication engineering in Khulna University of Engineering Technology (KUET), Bangladesh. Her research focus is mainly in the areas of image processing and computer vision. She also published two research articles in International Conference on Recent Trends in Image Processing & Pattern Recognition (RTIP2R) and International Journal of Computer Vision & Image Processing (IJCVIP). At USD she worked on foreign element detection in chest X-ray images. She also worked on remote sensing satellite imaging. Before joining USD, she worked as a senior software engineer in Samsung R&D institute, Bangladesh. In Samsung she worked on image codec, image enhancement, natural language understanding, and platform specific algorithm optimization for Samsung reconfigurable processor (SRP). In her time in Samsung, she also worked at two other overseas R&D centers, Samsung electronics India and Korea.