

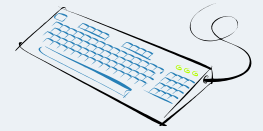


# Wire Segmentation from X-ray image

- ▶ 2014년 4월 4일
- ▶ F조
- ▶ 000,000,000

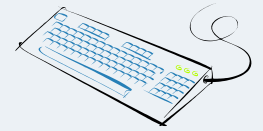
# Contents

---

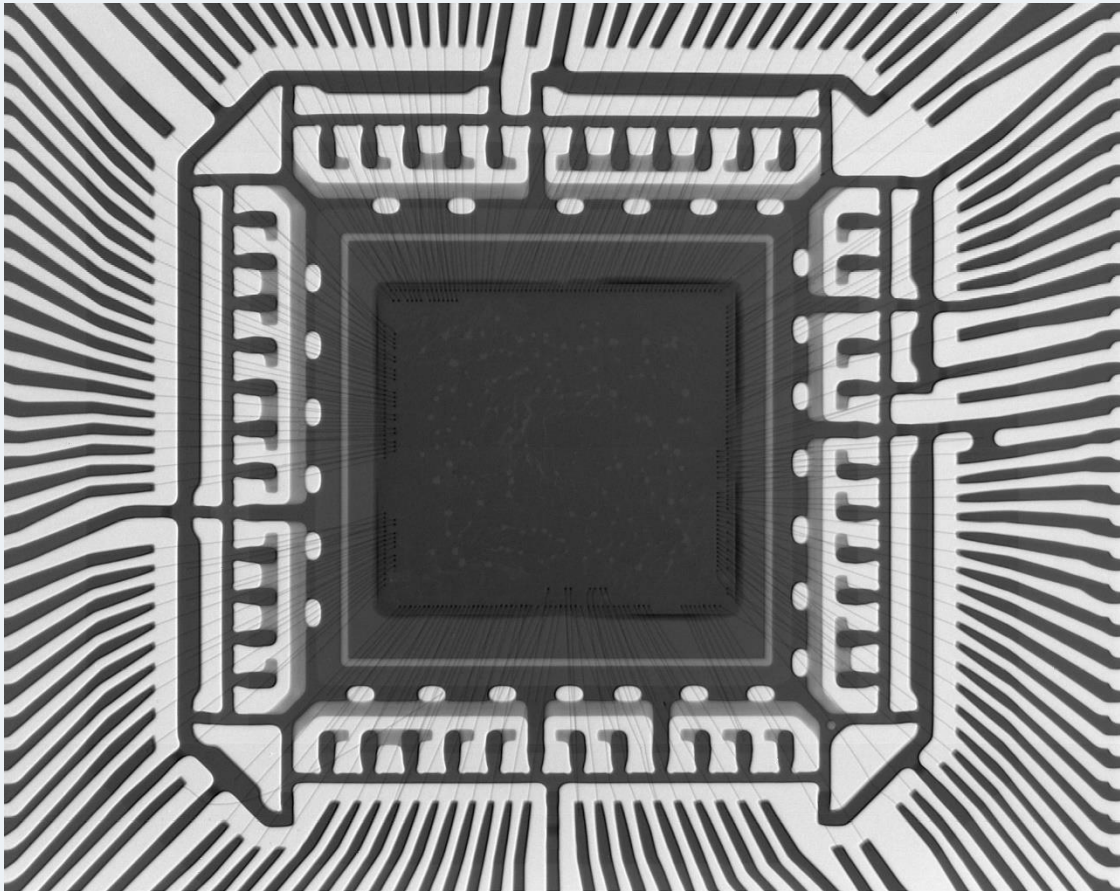


- ▶ Overview
- ▶ Approach
- ▶ Development Environment
- ▶ Architecture
- ▶ Basic Spec
- ▶ Current Status
- ▶ Further plan
- ▶ Division and Assignment of work
- ▶ Schedule

# Overview

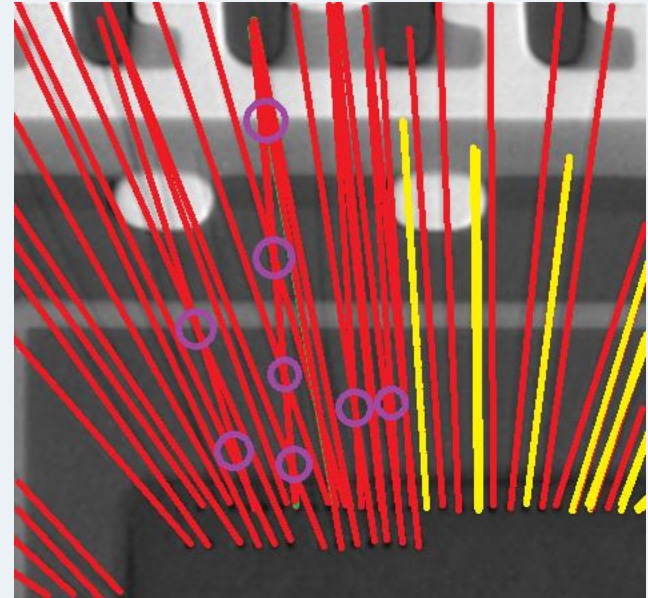
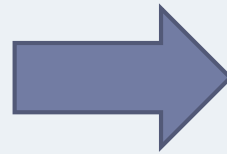
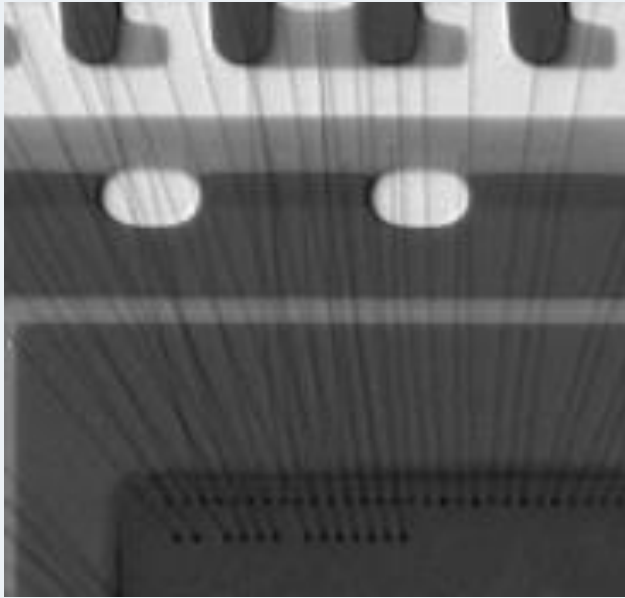


- ▶ Image segmentation problem



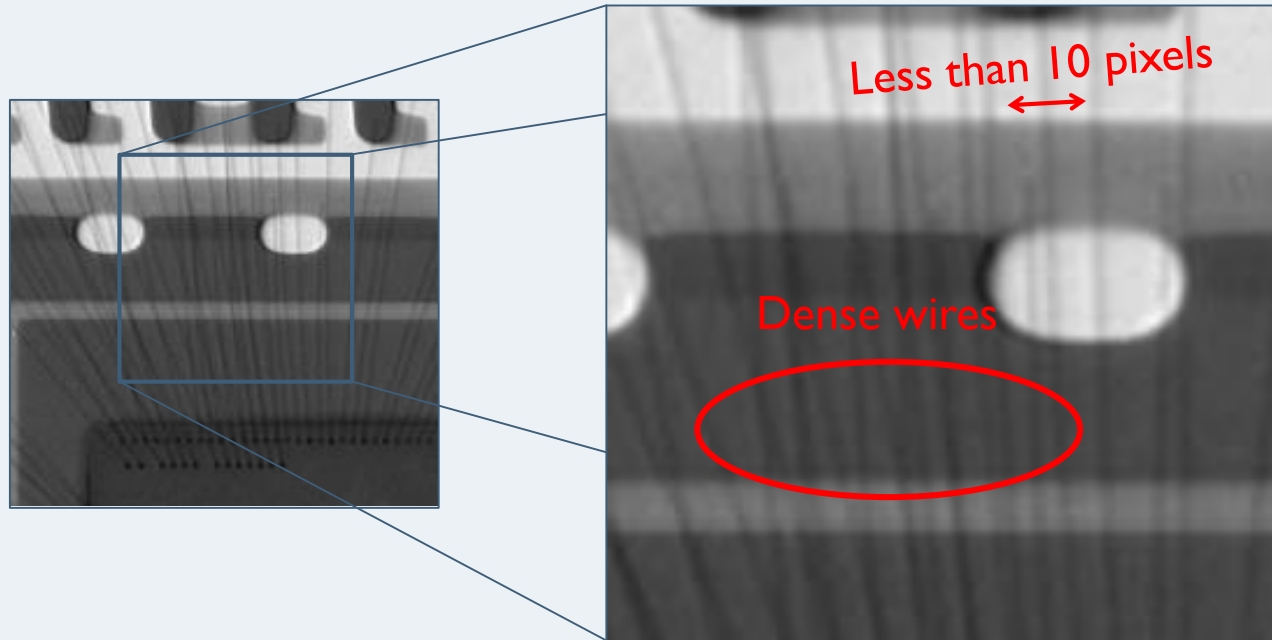
Find wires on  
this image!

# Goal



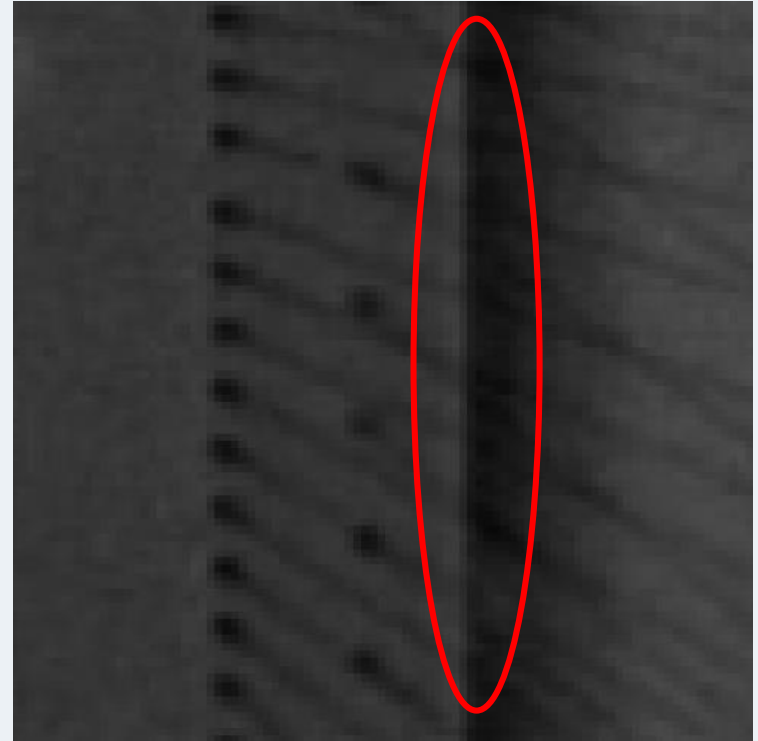
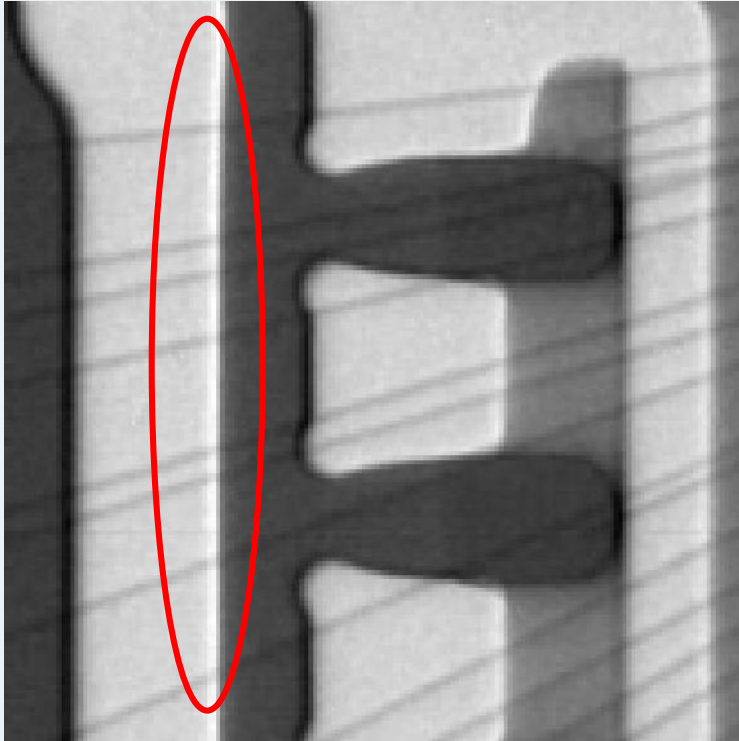
- ▶ Find every wire on target image
- ▶ Classify broken and crossed wires

# Problem



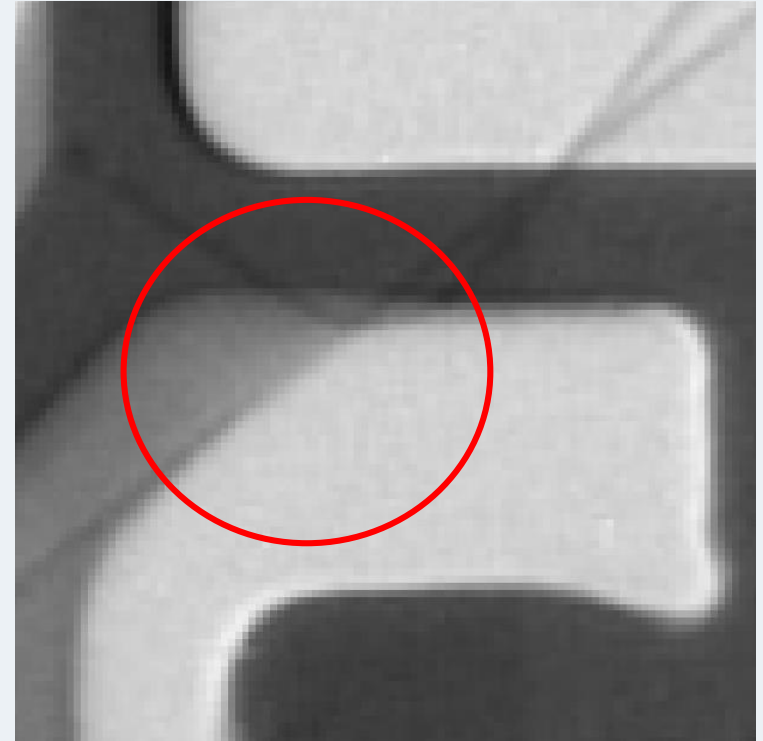
- ▶ Target wires are too thin. ( $< 10$  pixels)
  - ▶ Target wires are too dense.
- line tracking is difficult

# Problem *(contd.)*



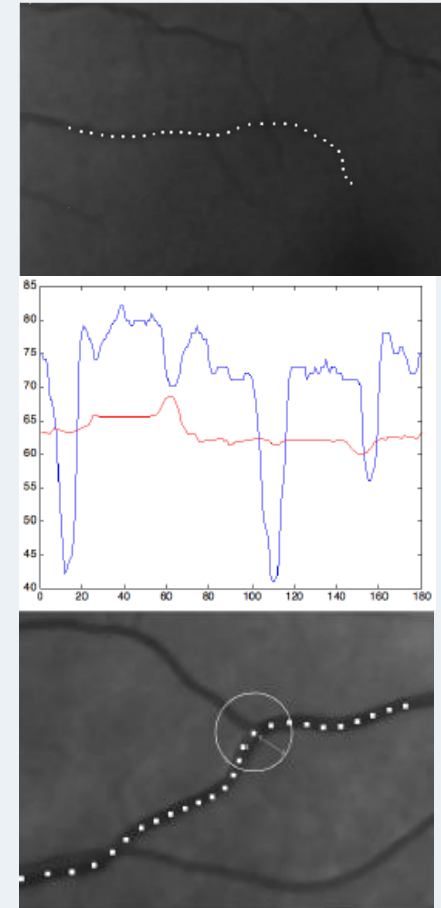
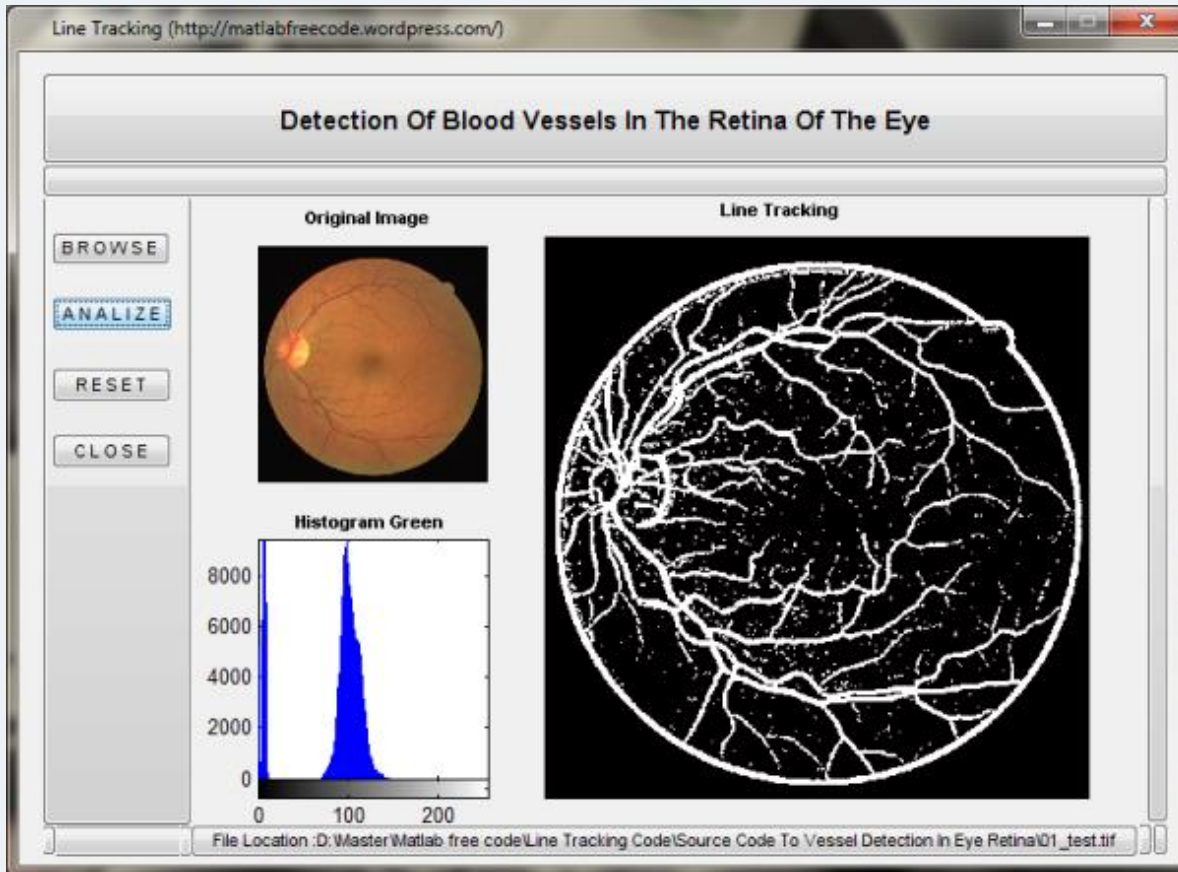
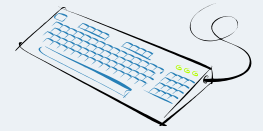
- ▶ Wire seems to be broken due to strong background object

# Problem *(contd.)*



- ▶ Hard to segment start points
- ▶ Wire almost diminishes

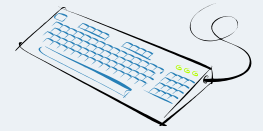
# Approach





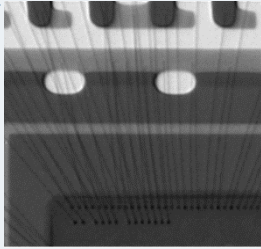
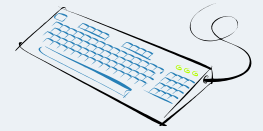
# Development Environment

---



- ▶ Windows
- ▶ Matlab 2013a
  - Image Processing Toolbox 6.5
- ▶ Visual studio 2013
  - C++
  - OpenCV

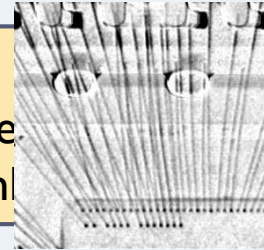
# Architecture



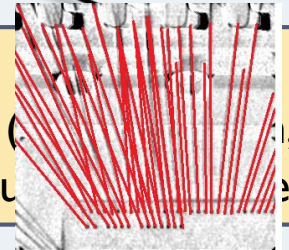
Wire image on X-ray  
(\* .jpg, Gray scale)

Input

(Despeckle, Contrast Enhancement, etc.)

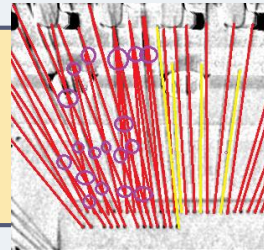


(Gaussian Blur, etc.)



Wire Segmented Image

Output

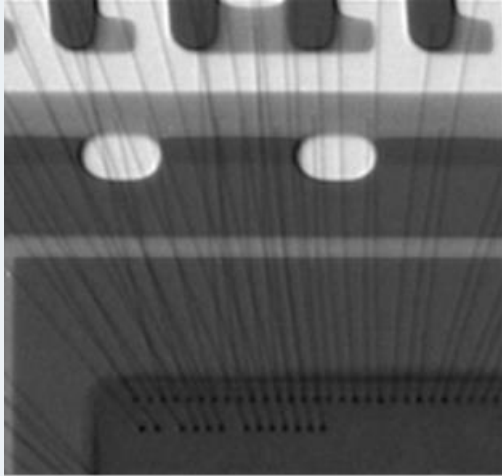


# Basic Spec

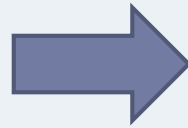


- ▶ [Mode 1] 사용자가 Wire의 시작점과 끝점을 지정하면 두 점을 양 끝으로 하는 Wire를 찾아내 표시하는 기능.
  - 사용자가 지정한 시작점과 끝점이 유효하지 않은 경우 에러 메시지를 출력하고 다시 시작점과 끝점을 지정할 수 있도록 하는 기능.
- ▶ [Mode 2] 시작점과 끝점을 지정하는 과정 없이 주어진 Image에 있는 모든 Wire를 찾아내 표시하는 기능.
- ▶ [Mode 1]과 [Mode 2]의 작업을 진행하는 과정에서 대상 Wire가 도중에 끊어져 있을 경우 다른 Wire들과 구별할 수 있도록 별도로 표시하여 사용자가 이를 인지할 수 있도록 하는 기능.
- ▶ [Mode 1]과 [Mode 2]의 작업을 진행하는 과정에서 대상 Wire가 다른 Wire와 교차하는 경우 이를 찾아내고 교차점을 별도로 표시하여 사용자가 이를 인지할 수 있도록 하는 기능.
- ▶ 사용자가 상기 기능을 효과적으로 활용할 수 있도록 UI를 제공.

# Current Status



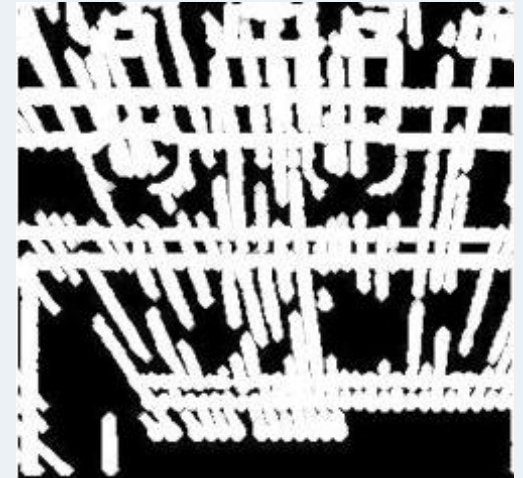
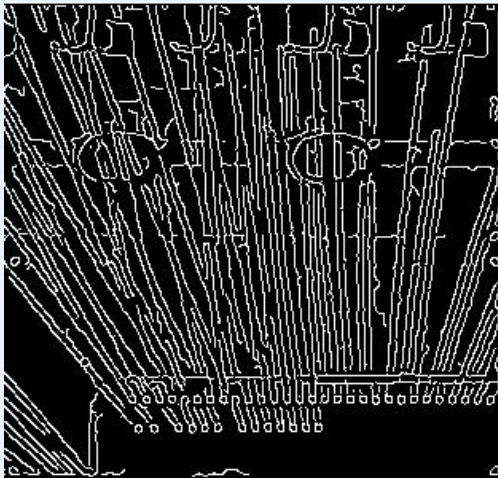
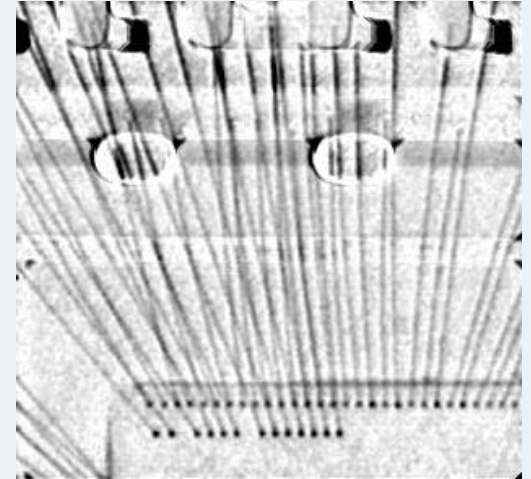
Preprocessing



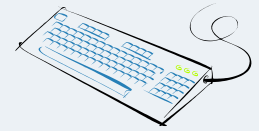
Edge detection



Binarization  
Dilation



# Further Plan



본 프로젝트와 유사한 문제들을 찾고 이 문제들의 해결 방법들 중 본 프로젝트에 응용할 수 있는 idea를 선별하여 알맞게 고치고 기존의 idea들과 융합하는 방법을 모색한다.

## User Interface module

### Wire Segmentation Algorithm module

Preprocessing

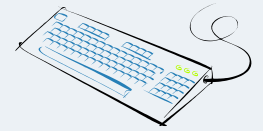
Segmentation

격주로 회사 담당자와 미팅을 진행하여 프로젝트의 진행 상황과 개선할 점에 대한 논의를 진행한다. 이를 통해 회사담당자와의 긴밀한 협업 시스템 유지한다.

매주 1~2차례의 조모임을 통해 개발에 사용할 언어와 도구에 대한 study를 진행하여 구현에 필요한 함수와 알고리즘을 테스트하고 선별한다.

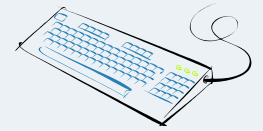
IBM Jazz Tool 을 이용하여 요구사항의 추적성을 확보하고 이를 바탕으로 세부적인 요구사항의 구현 여부를 지속적으로 확인한다. 또한 이를 활용하여 중간 목표를 단계별로 설정하고 진행 상황을 수시로 확인하여 프로젝트가 체계적으로 진행될 수 있도록 한다.

# Division and Assignment of Work



분류	항목	담당자(예시)
Preprocessing	Denoising image	OOO
	Enhancing contrast	OOO
	Dilating target object	OOO
	Detecting edges	OOO
	Extracting binary image	OOO
Implementing Tracking Algorithms	Testing vessel tracking techniques	OOO
	Implementing EM algorithm	OOO
	Implementing Gaussian model fitting algorithm	OOO
	Improving Algorithm for Wire Segmentation	OOO
User Interface	Constructing MATLAB GUI	OOO
	Constructing Visual C++ GUI (Optional)	OOO

# Schedule



내용	3월			4월				5월				6월		
	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주
Background Study	■	■	■											
Write SHR Document				■										
Spec. Presentation				■										
Write SR Document				■	■									
Implement Preprocessing					■	■								
Write DR Document							■							
Implement Tracking Algorithms							■	■	■	■				
Midterm Presentation								■						
Implement UI										■	■			
Optimization												■	■	
Final Presentation														■

---

감사합니다

OOO  
(nobody@snu.ac.kr)