Dawson Burgess
10/25/24
CS555: Machine Vision
Assignment 6

Final Project: Proposal 1

1 **Project Name**
1.1 PCB Defect Detection System for Quality Control

2 **Scope**
2.1 The goal of this project is to develop an automated vision system to detect surface defects on Printed Circuit Boards (PCBs). This system will focus on identifying and flagging common issues such as missing components, broken traces, incorrect soldering, or misplaced elements. PCB defects are important to detect in manufacturing, as even minor faults can cause functional failures. The system will utilize a combination of image processing techniques to enhance PCB images and detect faults. Machine learning will be used for defect classification to improve accuracy. After some looking around online, there are several datasets I can make use of to train and test on.

3 **Problem Description**:
3.1 Manufacturing defects in PCBs can be subtle and difficult to identify through manual inspection. PCB defect detection requires accuracy in identifying small variations, and these defects often exhibit subtle visual differences from non-defective boards. Traditional automated inspection systems are expensive and tailored for specific boards. This project aims to develop a flexible and open source solution for detecting common PCB defects using computer vision.

4 **Custom Image Processing Algorithm**:
4.1 Initial detection of defects will be based on traditional image processing techniques like edge detection, template matching, and contour analysis. These methods will help to highlight differences between expected and actual patterns on the PCB.

5 **ML Model:**
5.1 A classification model will be trained to identify specific types of defects (e.g., soldering issues, missing components). This model will improve the detection of subtle defects that may be missed by basic image processing algorithms.

6 **Challenges/Depth**:
6.1 Handling the variability in PCB images (e.g., lighting, different designs, partial occlusion) will be a challenge. Developing a robust system that works on various PCB designs would make the project more substantial. You could also extend the project to categorize the types of defects or provide precise locations for each defect.

7 **Approach**:
7.1 **Image Preprocessing**: Use high-resolution images and apply preprocessing techniques like histogram equalization, adaptive thresholding, and denoising to enhance defect visibility.
7.2 **Feature Extraction**: Implement traditional methods like edge detection, contour analysis, and template matching to identify potential defects. For example, broken traces can be detected using edge-following algorithms.

7.3 **ML Component**: Train a defect classification model using a labeled dataset of PCBs. Create custom models to differentiate between minor surface scratches and actual defects.

7.4 **Defect Classification**: Segment the board into different regions of interest (ROI) and use a defect classification model to analyze each region.

8 **First Check-In Deliverables** (Mid-November):

8.1 **Image Preprocessing**: Implement basic image preprocessing steps such as morphological operations, denoising, and adaptive thresholding to make the defects more visible.

8.2 **Initial Defect Detection Algorithm**: Implement and test early defect detection methods such as edge detection and template matching to identify key defect types (e.g., broken traces or missing components).

8.3 **Dataset Collection**: Start gathering or creating a dataset of PCB images with known defects. This dataset will be used for both testing the traditional algorithms and training the ML model.

9 Final goal

9.1 I would like to get a hold of some defective and non-defective PCB boards to do live webcam testing on. This may prove challenging but it is a end project goal. If I am not able to do this, printed out images may have the be the alternative solution for testing.