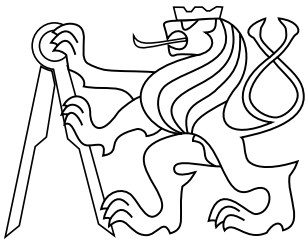
Diploma Thesis

Automatic Optical Inspection of Printed Circuit Boards

Bc. Ondřej Kunte



January 2018

Thesis advisor: Mgr. Radoslav Škoviera, Ph.D.

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

DIPLOMA THESIS ASSIGNMENT

Student: Bc. Ondřej K u n t e

Study programme: Cybernetics and Robotics

Specialisation: Robotics

Title of Diploma Thesis: Automatic Optical Inspection of Printed Circuit Boards

Guidelines:

The student´s task will be to design and implement an automatic optical inspection tool for printed circuit boards (PCB). The system will be primarily intended for a small sized production. The inspection system will be equipped with a camera for capturing the image of the printed circuit board. The inspection will be focused on defects occuring during production of the PCBs and their assembly.

The students´s tasks will be:

1. Study issues linked to image capturing and image processing.
2. Design and build the image capturing device.
3. Design and implement the algorithm for automatic PCB defects detection.
4. Demonstrate the functions of the equipment and perform testing in a real application.

Bibliography/Sources:

[1] Sonka, Milan; Hlavac, Vaclav; Boyle, Roger - Image Processing, Analysis and Machine Vision - Cengage Learning Standford, 2015
[2] Duda et al. - Pattern classification - NY, USA 2001
[3] Marshall, Dave A. and Martin, Ralph R. - Computer Vision, Models and Inspection - World Scientific Publishing, NJ, USA 1992

Diploma Thesis Supervisor: Mgr. Radoslav Škoviera, Ph.D.

Valid until: the end of the winter semester of academic year 2018/2019

L.S.

prof. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, June 15, 2017

Acknowledgement

I would like to thank all people who have helped me to successfully complete this work. I would like to thank my supervisor, Mgr. Radoslav Škoviera, Ph.D. for his excellent guidance. He could not lead me personally in most of the time, due to my research abroad. Nevertheless, he always provided me with advice and suggestions to solve the arising problems during my work. I would like to thank Intronix s.r.o. company, namely the Michal Ditrich, for working with the development of the experimental station and its subsequent funding. I would like to thank the company for wetting the printed circuit boards of real production used in system testing. I would like to thank the International office of Czech Technical University in Prague for supporting the foreign research. I would also like to thank Dr. A.N. Rajagopalan for accepting me in his laboratory at Indian Institute of Technology Madras and for providing me with the necessary support for research there. Last but not least, I would like to thank my University of CTU in Prague for possibility to work on my thesis at the prestigious foreign university of the IIT Madras in India, and I also thank IIT Madras university for supporting me and other foreign students in their research.

Declaration

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

I declare that I worked out the presented thesis independently and I quoted all used sources of information in accord with Methodical instructions about ethical principles for writing academic thesis.

V Praze dne ....

Abstrakt

Tato práce ověřuje hypotézu, zda-li je možné vytvořit systém automatické optické inspekce plošných spojů z běžných materiálů a za pomoci volně dostupných softwarových prostředků. Práce klade důraz na celkovou jednoduchost a cenovou dostupnost výsledného řešení, kdy výsledný systém je určen pro malosériovou produkci a soukromé jednotlivce, oproti dnes dostupným AOI zařízením. Práce diskutuje rozsah možných vad pro inspekci osazených a neosazených plošných spojů, navrhuje výběr metod pro zpracování obrazu a uvádí výsledné algoritmy pro inspekci včetně jejich testování. Součástí práce je návrh a realizace zařízení pro snímání plošných spojů a návrh grafického uživatelské rozhraní pro ovládání celého systému.

Klíčová slova

Automatická Optická Inspekce, AOI, zpracování obrazu, deska plošných spojů, DPS, defekt

Abstract

This thesis verifies the hypothesis whether it is possible to create an Automatic Optical Inspection system of printed circuit-boards from common resources, using freely available software frameworks. The work emphasizes the overall simplicity and affordability of the resulting solution. The resulting system is designed for the small-scale production and the private individuals versus today's large-scale focused AOI devices. The paper discusses the range of possible defects for inspection of assembled and non-assembled printed circuit-boards, suggests methods for image processing and presents final inspection algorithms, including their testing. Part of the thesis is the design and construction of a device for printed circuit-board capturing and the design of a graphical user interface for controlling the entire inspection system.

Keywords

Automatic Optical Inspection, AOI, Image processing, Printed circuit board, PCB, defect

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

AOI	Automated Optical Inspection
AM	Amplitude Modulation
FM	Frequency Modulation
PCB	Printed Circuit Board
CD	Compact Disk
SMT	Surface-mount Technology
THT	Through-hole Technology
CCD	Charged Coupled Device
SW	Software
RGB	Red-Green-Blue
CIE	Commission Internationale de l'Eclairage
HSV	Hue-Saturation-Value
IDE	Integrated Development Environment
SDK	Software Development Kit
BSD	Berkeley Software Distribution
STL	Standard Template Library
CAD	Computer-Aided Design
MIT	Massachusetts Institute of Technology
ISO	International Organization of Standardization
DSLR	Digital Single Lens Reflex
DC	Direct Current
LED	Light Emitting Diode
TP	True Positive
FP	False Positive

1. Introduction

Optical Inspection of printed circuit boards is not a new technology. It started in 1940s with traditional optical comparators. The control was performed by a human worker by basically comparing two magnified images of tested boards. The testing device was purely mechanical and the whole system depended on the workers experience and eye-sight.

During the years the systems were improved and with incoming automation began to be more and more complex, assuring a higher level of inspection and wide range of fault detection. Currently, the Automated Optical Inspection (AOI) system is generally a very expensive and sophisticated machine focused on large-scale production.

1.1. Motivation

As a student of Faculty of Electrical Engineering the author has had a long-term interest in the electronic device prototyping and manufacture as it is usual among the technically focused people in his country. This particular interest has roots in history, when citizens of former Czechoslovakia before 1989 struggled with the unavailability of basic electronic devices. As a result, it was common to create devices, like amplifiers, transistor AM/FM radios and other systems at home, using the necessary equipment and a gained knowledge from magazines. This legacy has been passed onto following generations and today a unique community of amateur enthusiasts is settled across the nation. With expanding accessibility and affordability of electronic parts and equipment, this area is increasingly popular.

With such an extensive and culturally rich background, Czech industry and production go hand in hand. There are many small-sized companies and start-ups focused on developing electronic devices and electronic prototyping. This development can be seen not only in Czechia and Europe but also globally. These companies generally produce their products in small series and quantities. PCB production is outsourced and assembling and soldering is performed manually or without serial production. Products are produced mainly in limited and small series. Nevertheless, to assure the quality of the output product, output control is always required as it is usual in a large-scale production.

In the beginning of the year 2015 the autor started to work a part-time job in a small-sized company, called Intronix s.r.o.[1], located in Prague. This company is focused on development and manufacture of electronic systems mostly used for automation and non-electrical parameters measurements in industry production. Size of the company perfectly fits the previous description since it is mainly focused on small size production. Authors job was to perform output control and testing of final assembled PCBs. The author carried the idea of automated PCB inspection a certain period of time before, but exactly at this time the author started to realize its true potential when he got more familiar with the standard manufacturing process and its economical aspects.

1.2. Draft proposal

Since this research is rather practically focused, more emphasis is placed on the suitable combination of existing methods, rather than developing a new ones. In the end, this approach should have a greater probability of success.

In this work an Automated Optical Inspection will be developed and designed. The system will consist of a hardware part and a software part. The hardware part includes the design of an image capturing device and its support equipment. More types of image capturing devices will be used such as industrial solution cameras and consumer electronics cameras as well. The software part includes selection of suitable image processing techniques and inspection methods or rules for sufficient quality inspection of PCBs. The system should also contains understandable user interface for comfortable use.

The main benefit of this work should be to verify possibility to create a functional, affordable and accessible automated optical inspection system which could be beneficial either for small-sized companies, developers or even enthusiasts and their home projects. Since most of available AOI systems are mainly focused on a large-sized serial production, the goal of this work is to use the exactly opposite philosophy and try to find the solution for a small-sized companies or private individuals.

1.3. State of the art

As mentioned in the Introduction, most AOI systems present currently are complex machines capable of inspecting enormous quantities of samples in short period of time. Sometimes the inspection is performed even with multiple samples at the same time. Examples of AOI producing companies are Saki Corporatio[2] (Japan), modus high-tech electronics GmbH[3] (Germany) or the pioneer of AOI, Orbotech Ltd.[4] (Israel). A product of the last one, Orbotech Ltd. company, can be seen in Figure 1. In this section are presented some of the working systems used in industry and practical research and applications of this topic.

Fig. 1. Discovery^{TM} II 9000 AOI System by Orbotech Ltd., taken from [4].

These systems are focused on plain and assembled PCBs as well. Depending on the particular product they can use the latest 3D screening technology and multiple light source reflection. Inspection is performed using multi-image technology to increase the accuracy of inspection by producing multiple images of the inspected board. Achieved

accuracy then varies in range of decades of microns. For presented Discovery system above, it is exactly 30 μm [4].

Fig. 2. Bungard QualityScan 3000 (left) and Flatbed Scanner Platform (right). Taken from [5]

The Bungard Elektronik GmbH & Co.KG company has in its portfolio an AOI workstation focused on non-serial production. As can be seen in Figure 2, the system is based on a similar topology as we can all know from office equipment we generally use but in this case with larger dimensions and accuracy. This system is less complex than the previously mentioned system from Orbotech, but it still belongs to the group of high or middle-high class of inspection systems with accuracy up to 50 microns [5].

Neelum Dave et al. presented a simple and inexpensive solution for AOI of bare printed circuit boards. Their system captures the image via movable camera. After the image is captured pre-procesing methods are applied followed the image processing based on image extraction[6].

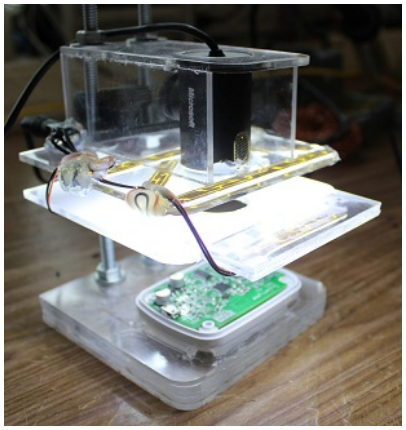


Fig. 3. Automated Optical Inspector by Robert Baddeley

A low-cost AOI solution of assembled board inspection was investigated by Robert Baddeley. He proposed a system equipped with a 720p web-camera stationed above the tested printed circuit board as can be seen in Figure 3. The system is complemented by support lighting. Extraction method used by Neelum Dave et al. was changed to a process based on the template matching (2.5.4). As Baddeley presents, final cost of the device was settled up to $50 and with usable results. The final system was able to detect missing and misplaced components [7].

2. Methodology

Some of the most used techniques suitable for PCB inspection and related theory are presented in this chapter. Each method is followed with a basic explanation of its functionality. However, as this work does not substitute a textbook. Deeper explanation of the methods is beyond the scope of this work.

2.1. Printed circuit board introduction

A printed circuit has been a basic interconnecting technique for electronic components for decades. Since this technology and its problematic is the main element of this work, it is required to familiarize the reader with the concept itself, its fabrication, usage and possible manufacturing errors occurring during the fabrication of PCB. There are many types of manufacturing processes, materials and technology suitable for various output products. However, only a fraction of available and the most widely used technologies are presented.

2.1.1. Fabrication

A new type of error could occur in every step of production. Faults of plain boards and assembled boards, including the manner of their occurrence and cause are mentioned in section 2.1.2.

When describing manufacture of PCB, it is necessary to distinguish it by its construction, which goes hand in hand with the manufacture process. By this description we can separate printed circuits into three groups:

- Single and double layered, rigid base
- Multi-layered, rigid base
- Flexible single and double layered

The dielectric base material is selected by required assembly to fulfill expected parameters such as water absorption, flexural strength, tensile strength, thermal parameters and hardness. Materials used for most applications include[8]:

- Phenolic-resin-impregnated paper
- Acrylic-polyester-impregnated random glass mat
- Epoxy-impregnated paper
- Epoxy-impregnated fiberglass cloth

The last mentioned option of fiberglass laminate is currently the most used one since the previous mentioned technologies are becoming obsolete.

The manufacturing process begins with input material. Generally, it is the base plate covered by copper foil from one side in case of one side circuit board and analogically the base plate covered from both sides in the case of double layered circuit board. The copper thickness usually varies in range from 9 μm to 105 μm. The most used thickness

is 35 μm[9]. General manufacturing process is based on the photo-resist technique[8] which consist of these steps:

Cleaning
Surface of the board is chemically cleaned and degreased. During the process even the smaller impurities on microscopic level and surface corrosion are removed. This is crucial for the resulting quality of the final product. Cleaning is also repeated several times in between the other manufacture steps.

Resist application
A layer of resist is applied on the cleaned copper layer. The resist holds the circuit pattern and prevents of removing an unwanted areas of copper during the next step, etching. Resist can be applied directly in a form of pattern or as a photoresistive continuous layer which is then transformed into the pattern via photoresistive method.

Etching
During this operation all copper not protected by the resist material is removed by an etchant. The used etchants could be various types. The most common are based on ferric chloride, combination of hydrochloric acid and hydrogen peroxide or the compound of ammonium peroxydisulfate.

Resist removal
Resist layer is removed after etching. The board is then cleaned to ensure no etchant remains on the board. The board is then ready for next fabricate by drilling, trimming, application of soldermask etc.

The described process is suited mainly for single-sided printed circuit boards. However, this procedure can be applied with minor changes to double-sided boards as well. Double-sided technology also includes necessity of side matching and copper electrodecomposition for creating conducting vias through holes which is beyond the scope of this short description.

The fabrication process of a multiple-layer board (three layers and more) is different. Single inner layers are etched separately, into thin conducting foils. Etched and cleaned foils are then laminated together. A condusive connection between layers is created by vias using drilling and copper electrodecomposition in a similar way as for the double-layered circuit boards.

The final circuit board is then assembled. At this stage of manufacturing process the board is fitted with electronic parts, mounting parts, connectors and others. Nowadays, boards using SMT technology prevail so the assembly process can be described as:

1. Cleaning
2. Flux application
3. Parts assembly
4. Reflow of the flux
5. THT parts assembly, soldering
6. Cleaning

2.1.2. Possible board defects

At first it is necessary to divide possible defects into two groups:

- Plain board defects
- Assembled board defects

The first type of errors occurs during the manufacture of the board itself and it is mainly caused by imperfections during copper pattern establishing or prefabrication defects of the base material. The second type of errors is related to the assembled parts themselves and it is focused on quality of soldering, parts position, occurrence of parts and their placing.

Plain board defects

In Figure 4 one can see an example of circuit pattern. The most common fault in the circuit pattern is "short" and "open circuit". Other examples could be unetched copper, mousebite nick, scratches or cracks, overetching and underetching, pad size violations, spurious metal, violations of spacing and many others. From destription of Figure 5 we can see most of the faults which could occur during the PCB fabrication [10].

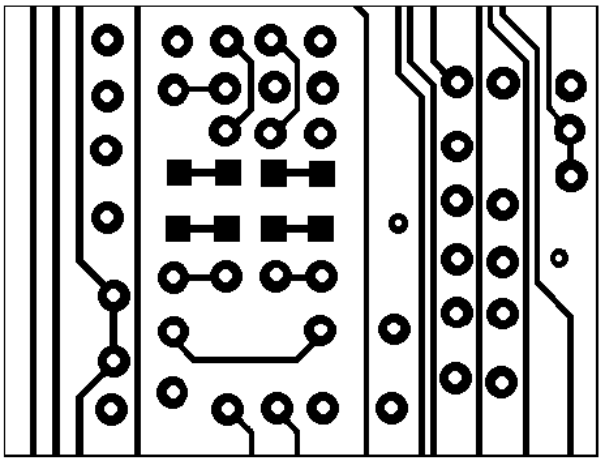


Fig. 4. Example of good PCB pattern, taken from [10]

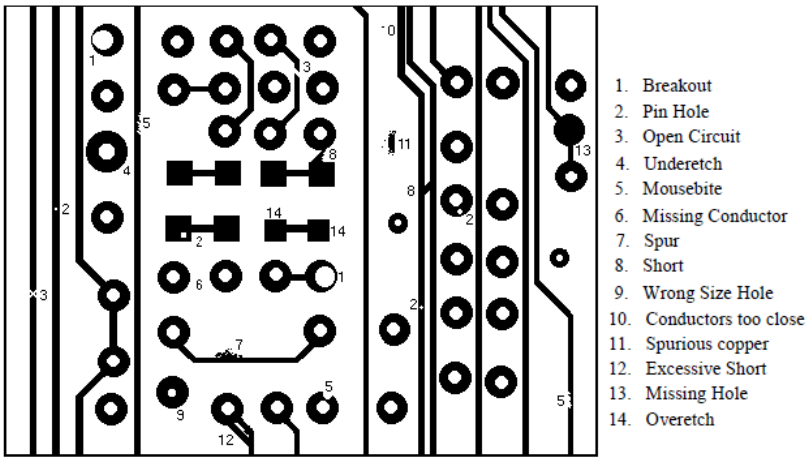


Fig. 5. Example of defective PCB pattern, taken from [10]

Assembled board defects

Most common and main inspected defects can be divided into two groups:

- Soldering defects
- Component placing defects

Soldering defects are related to soldering itself when SMT components are soldered by flux reflow as mentioned in 2.1.1. THT devices are soldered using wave soldering technology. As the reader can imagine, the process of soldering is basically connecting two metal parts by a non-diffusive method using another metal in liquid form. A connection is established by the hardening of the filler metal. Fluid mechanics in general always brings many conditions which are difficult to meet. Because of this fact the range of possible defects is tremendously wide. The defects most fatal for proper functionality of the final electronic device are:

- Solder ball
- Bridge (short)
- Skip (open)

An "open" defect can be caused by placing inadequate amount of flux or by impurity or corrosion of the future joint. On the other hand a bridge or "close" defect and solder balls are caused by excess of added material. Solder balls are all the more dangerous because without removal they can cause fatal damage of the device even if their effects could be initially untraceable. Examples of all presented faults are visualized in Figure 6.

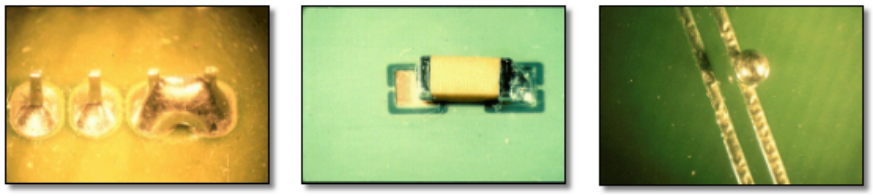


Fig. 6. Examples of common types of soldering defects. Bridge (left), Skip (center), Solder ball (right), taken from [11]

Component placing defects as the name suggest are related to the positioning of the components. The component could be misplaced, switched with another part or it's position related to the solder pads could be insufficient.

All these defect could take place before flux reflow by inappropriate assembly or after flux reflow where they can be caused by improper flux application, assembly or their combination. Some position defects can be caused even by the design of a printed pattern itself. In this case, the design fault combined with surface tension of melting flux can dramatically influence the final position of the soldered component [8].

2.2. Optical acquisition system

As the word "Optical" in AOI suggests, the process of inspection includes the processing of imagery in particular. The process involves digitization of the object to be inspected for visual data and for this, suitable equipment is required. Possible methods of image capturing are presented in this section.

2.2.1. Image capturing

Procedure is usually covered by a sensor in the form of a camera or other digitizer. There are several types of cameras used in AOI, where most of them are based on CCD sensors, laser scanners or others. Comparison of sensor capability itself is inappropriate since the best performance of the camera is achieved after matching with suitable lenses and it is hard to say which technology would be more suitable for different usage. In general the CCD sensors are cheaper but as mentioned the final price can be changed significantly by adding the optical system and other support equipment.

One of the most important condition which the camera system has to fulfill is a sufficient resolution. The pixel size of the smallest fault to be detected should be at least twice that of the vision system [10]. Increasing the resolution with a single camera results in a smaller field of vision. Compensating techniques like multiple cameras or multi-image techniques (1.3) can be used. Multi-image technique requires only one camera which is movable. Multiple camera system consists of several cameras positioned stationary. In general the multiple camera system can achieve better accuracy and resolution but it is more expensive and more demanding for maintenance and calibration. Multi-image technique has higher structural requirements for the handling system and higher demands on SW image pre-procesing. Using only one camera sensor make this technique less expensive.

2.2.2. Light illumination

Illumination of a captured scene is an important part of the image acquisition. Correct light adjustment has a major effect on the resulting image. By choosing the suitable lighting technique it is possible to avoid the need for complex image processing algorithms by increasing sustainability of variable parameters. The main lighting parameters that characterize the system's sustainability are:

- Intensity
- Uniformity
- Directionality
- Spectral profile

The relative importance of these parameters and the degree to which each one must be controlled are largely governed by the surface of a given PCB and constraints imposed by the camera.

Light source

The following lighting sources are now commonly used in machine vision:

- Fluorescent
- Quartz Halogen - Fiber Optics
- LED
- Xenon
- Metal Halide (Mercury)
- High Pressure Sodium

The most widely used lighting types in machine vision are quartz-halogen, flurescent and LED light sources. LEDs, mostly because of its minimal required dimensions, are used in small and middle sized applications. When a very bright light source is required, halide, xenon or high pressure sodium is applied. Metal halide, when complemented with suitable band-pass filter, is also suitable for fluorescence studies since it has many discrete wavelength peaks. During the last years the versatility of LED technology is rapidly improving its stability, intensity and cost-effectiveness but still this technology is more suitable for small sized applications however its usage is increasing.

The spectral content of the light source is also important. In microscopy applications to which the AOI is relatively close a full spectrum light source is often required, due to which the quartz halogen or xenon are often used as well as white the LED heads complemented with all color RGB light heads.

Depending on the required light parameters more than one light source may be used to adequately solve the lighting issues.

Illumination techniques

One of the important parts in illumination is the use of correct technique. This involves setting up the best suitable geometry of three main components: light source, investigated object and sensor. Some of illumination techniques are presented in Figure 7.

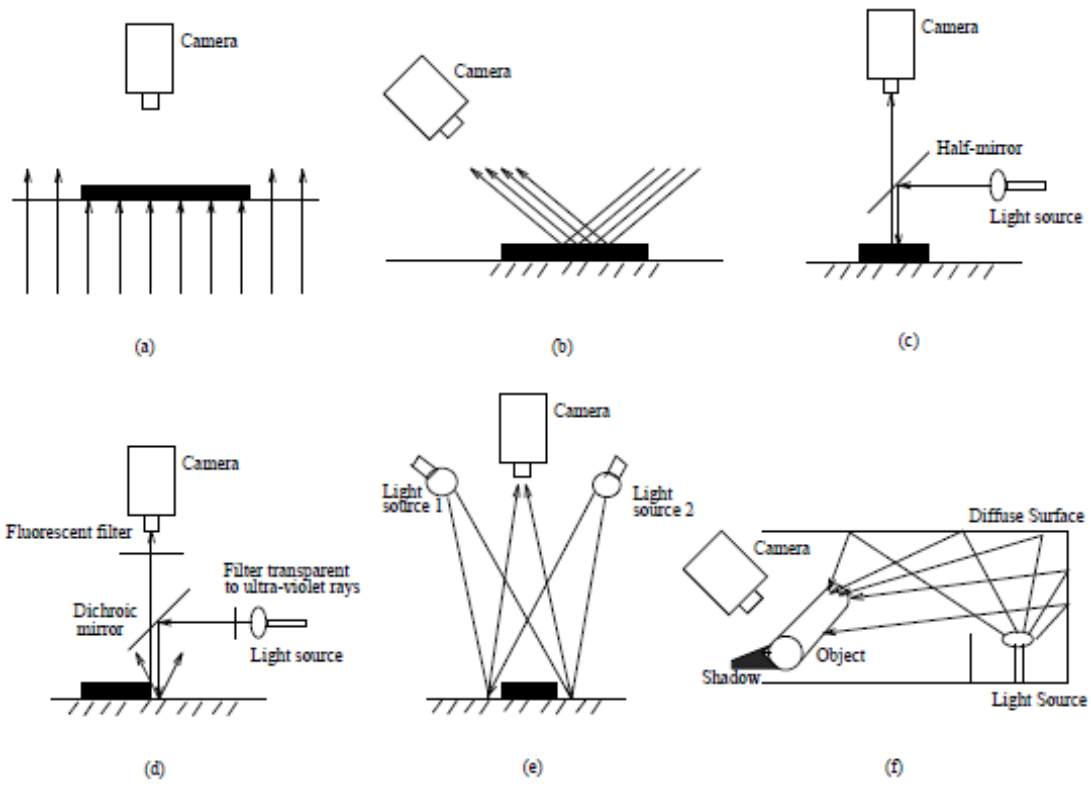


Fig. 7. Different illumination techniques (a) Back lighting, (b) Directed lighting, (c) Vertical lighting, (d) Fluorescent lighting, (e) Bidirectional lighting, (f) Diffuse lighting, taken from [10]

Back lighting generates dark silhouettes against a bright background. Commonly a monochrome light with light control polarization is used. This technique is used for object measuring, edge detecting etc. because its precision.

Partial bright field light or so-called Directional lighting (7.b) is the most commonly

used lighting technique. It is simple to perform, generates the contrast well and enhances a topographic detail. It is much less resistant against flares and "hotspot" reflections.

Vertical lighting and fluorescent lighting are members of the so called "On-axis" lighting technique. This type of lighting is used in highly specialized application. On-axis lights are suitable for flat objects by enhancing their textured, angled or topographic features.

Diffuse lighting is used in applications where multi-directional light is needed. This type of lighting suppresses unwanted shadows of spatial objects and helps balance the brightness of the scene. Using bi-directional lighting has a similar effect. There are two ways to achieve diffuse lighting, either by a diffuse dome (7.f) or by a flat diffuse where direct lighting is complemented with a diffusive layer.[12, 13].

All types of illumination have to deal with presence of ambient light. Ambient light negatively affects consistence of a scene setup and can significantly influence the output image. The goal is not to completely remove the ambient light since in many cases it is not even possible but it is possible to stabilize its fluctuations or suppress its appearance in the system. Generally it is accomplished by decreasing the light ratio R

$$R = \frac{I_{amb}}{I}, \tag{2.1}$$

where I is the intensity of the light source and I_{amb} is the intensity of the ambient light. This can be done either by limiting incoming ambient light using black boxes or by increasing the intensity of the light source which might not be possible in every application. There are also more sophisticated solutions to suppress ambient light based on measuring the ambient light and adjusting the parameters of the sensor or light source via control loop.

Color contrast enhancement

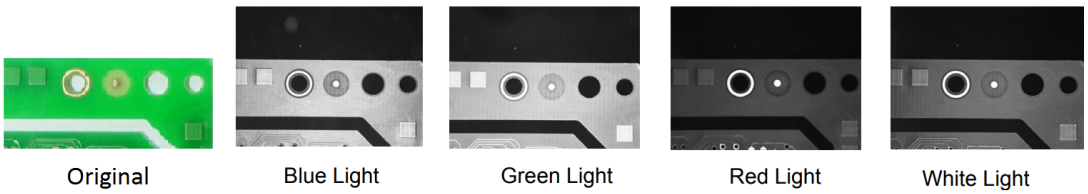


Fig. 8. PCB footage taken using various color illumination

As per common knowledge, materials reflect and absorb various wavelengths of light differently to create its visual color. Like colors reflect and surfaces are brightened, opposing colors are absorbed and surfaces are darkened as we can see in Figure 8. By using the color wheel in Figure 9 we can enhance contrast of wanted color spectrum in the image.

High contrast images can be obtained by capturing multiple photos with different color illumination and blending them together in a desired way or processing them separately. Wanted areas like parts and paths on PCB can be brightened and unwanted areas like soldermask can be suppressed [14].

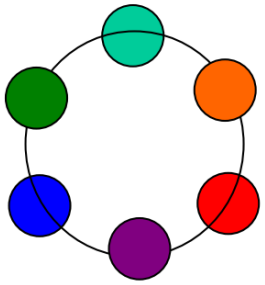


Fig. 9. Color wheel of Warm vs. Cool colors

2.3. Image pre-processing

Before an image can be used as a source of information, it should be somehow prepared in a way a data acquisition would be as effective as possible, resulting in a stable and uniform results. Pre-processing is used for this task. It generally involves global or local techniques which operate with the whole image and its output is another modified image with enhanced features. Pre-procesing does not increase image information [15].

2.3.1. Color spaces

There always has been the question how to define a color. For that reason we have a color space which allows us to described it in a linear space. Based on the trichromatic color perception in humans, we have 3-dimensional spaces. Different types of color spaces were established during the years based on different physical principles or human perception as presented below.

RGB

Color space based on additive color primaries when the output color is produced by blending the primary colors red, green and blue. RGB is mostly used in electronic devices since it is historically based on the physical principle of color television. Not every wavelength can be generated by RGB components and the color space is device-dependent. Due to this the same color can look differently across a different devices which is not useful.

CIE L*a*b*

Color space created by International Commission on Illumination. The Lab space is based on CIE XYZ color space. It is a perceptually uniform color space which tries to simulate the way how humans perceive the colors. The base colors are L* (light), a* (color pair red-green) and b* (color pair blue-yellow). It is mainly used as a base for color transferring among a different devices and for other color management.

HSV

Color space based on the painters color palette, where HSV stands for Hue-Saturation-Value. In this color space the hue vector carries an information about the color base from the palette, i.e. dominant wavelength. Saturation then describes how far the color is from a neutral grey color. This color space is very useful for image color processing because it can hold the information about the color unaffected even during varied lighting conditions.

2.3.2. Pre-processing methods

Camera calibration

In reality we are generally not working with an ideal optical system. Lenses, mirrors and other optical systems have defects which cause color changes, geometrical distortion etc. In practical application the radial distortion is the most important, presented in Figure 10. This defect is even more affecting the scene for closeup applications such as AOI and it needs to be compensated. Applications including precise measurements would not be possible without compensation.

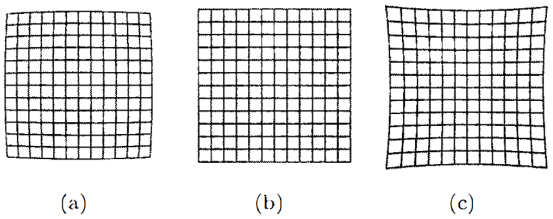


Fig. 10. Radial distortion. (a) Barrel distortion. (b) Undistorted. (c) Pincushion distortion, taken from [15]

Radial distortion is generally approximated as a rotationally symmetric function with respect to the principal point and dependent on the distance r of the measured pixel (x', y') from the principal point (x_0, y_0)[15].

$$r = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{2.2}$$

To assure rotational symmetry, an approximation polynomials up to order of six at most are used. If we mark the pixel after correction as $x = x' + \Delta_x$, the approximation can be written as

$$\begin{aligned} x = &\quad x'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y = &\quad y'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6), \end{aligned} \tag{2.3}$$

where k_1, k_2, k_3 are coefficients representing the radial distortion.

Another aberration which occurs during the image capturing is tangential distortion. This type of distortion is caused when the image plane and the lenses are not perfectly parallel. This can be corrected by

$$\begin{aligned} x = &\quad x' + \left(2p_1 xy + p_2(r^2 + 2x^2\right) \\ y = &\quad y' + \left(p_1(r^2 + 2y^2) + 2p_2 xy\right), \end{aligned} \tag{2.4}$$

where coefficients p_1 and p_2 are coefficients representing the tangential distortion. In total we have five distortion coefficients

$$Distorsion_{coefs} = \begin{bmatrix} k_1 & k_2 & p_1 & p_2 & k_3 \end{bmatrix} \tag{2.5}$$

.

These coefficients are obtained experimentally by observing a known calibration pattern. The pattern usually has the form of some lines or shapes. Chessboard or similar circular pattern is also suitable.

White balance

In general the colors of an image have to be adjusted to match the real color appearance. The imperfection of color representation is mainly caused by camera sensors or light illumination. In reality it means that for example white surfaces could appear having a color tinge. Basically, white balance represents a camera color calibration.

Globally we are adjusting the intensity of each color channel for calibration. Particularly the neutral colors such as white or gray colors are used as a reference. In reality the image of reference area (white or grey surface) is captured. The color channels are then rescaled and normalized.

Edge detection

Major contribution of edge detection was made by Canny in 1986 [16]. Since then, his edge detector became one of the most used algorithms for edge information extraction. The detector itself is based on optimal boarder detection of steps and ramp edges. The algorithm can be described as:

- Image smoothing using Gaussian filtering (see below).
- Finding the intensity gradient for every pixel.
- Threshold edges with hysteresis to eliminate spurious responses.
- Edge tracking. Elimination of weak edges and smaller edges not connected to strong ones.

Smoothing

Smoothing or blurring the image is a technique primarily focused on image noise removal. Noise can be defined as small fluctuation in the image. As well as sharpening both techniques fall into the category of local pre-processing. In these methods a small local area of neighboring pixels are used to deliver a new brightness value to the focused pixel.

Linear filters calculate the resulting value by a linear combination of pixel values from the local area. This area is also called a convolution mask and it usually has the form of a rectangular kernel. The calculation is basically performing a discrete 2D convolution and can be described by the function

$$f(i,j) = \sum_{(x,y)\in \mathbf{O}} \sum h(i-x, j-y) g(x,y), \tag{2.6}$$

where the pixels are weighted by the coefficient h, kernel.

Gaussian filter is presented as the main representative for the linear filters. Convolution kernel $h(x,y)$ is created according to the normalized Gaussian distribution formula

$$h(x,y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \tag{2.7}$$

Linear filters reduce the noise by image averaging which poses blur and the edge information is decreased. Improvement would be the application of filters only to pixels with identical properties. Using this process, the edge information could be preserved. These types of filters falls into group of **non-linear filters**. Two examples of non-linear filters are Median filter and Bilateral filter.

A Median filter computes a median from subset of neighboring pixels. In probability theory, the median divides the probability distribution in the lower and higher half.

The median is computed by selecting the middle value in the ordered set of pixels by their intensities. The advantage of median is its resistance against extremely distant values. It also better preserves edge information in comparison to Gauss filtering.

The main disadvantage of median filtering is the damage to thin lines and sharp corners when the rectangular neighborhood set is used. Especially vertical and horizontal lines are suppressed at most. This problem can be solved by using the kernel of different shape, such as "+" or "×" shape.

Bilateral filter assigns the value of issued pixel from a weighted average of intensity values from neighboring subset. The weights can be based on Gauss distribution and depend on two parameters:

- Euclidean distance from the pixel
- Color intensity difference

In reality that means if the filter subset performs in relatively uniform area it acts almost like a linear filter with Gauss distribution. However, if the subset "hits" an edge, pixels behind the edge with high intensity difference are assigned with a lower weight and their impact on filtering is suppressed. The filter well preserves edges. When used on a large scale the filter starts to "segment" individual areas and the image starts to look like a cartoon.

2.4. Image subtraction

Image subtraction is a special but yet very simple technique in image processing. The method operates on pixel-to-pixel level where the pixels of the input images should operate in the same value type. Color images are separated by its channels and handled in a way similar to a single channel greyscale images. The output pixel can be estimated by computing an absolute difference of input images/channels, such as:

$$D = |P_1(i,j) - P_2(i,j)|, \tag{2.8}$$

where P_1 and P_2 are input images/channels. The method is used for suppressing imperfections of a scene mostly caused by inappropriate or ambient lighting as well for spotting the difference between two images, such as tracking a moving object in a video stream or in this case detection of defects when comparing to the reference board.

2.5. Segmentation

The main task of segmentation is to divide the image into parts or areas where the containment of the area best represents the object captured in a real world. By the outputs of the algorithm we can divide the segmentation into two groups [15]:

- Complete segmentation
- Partial segmentation

In global segmentation the image is divided into a set of separated regions corresponding the best with the real objects in the captured scene. On the other hand outputs of partial segmentation does not fully correspond with the image objects. Separated regions are divided based on a chosen property such as brightness, color, reflectivity, etc. Partially segmented image can be input for further processing.

2.5.1. Thresholding

Thresholding is a useful method for setting up boundaries of objects resting on a contrasting background by using a threshold rule. When applying the threshold rule initially, the threshold value of pixel intensity is selected. All pixels of threshold or above are marked as foreground and all pixels under the threshold are selected as background as it is also called image binarization. The boundary can be also assigned from the set of interior points each of which has at least one neighbor in the background.

There are more ways to establish the threshold value. **Fixed thresholding** holds the threshold greylevel value constant throughout the image. If the background intensity is reasonably constant and the foreground intensity contrast is approximately equal with respect to the background the fixed threshold can give good results when properly selected.

In another way **adaptive thresholding** can be used. In this case the threshold value is calculated from a small region of the image where the threshold value could be the mean of neighborhood area or weighted sum. Weights are usually a gaussian window. This brings a big improvement for example that makes it possible to process images with uneven light conditions.

2.5.2. Hough transforms

The Hough transform falls into group of edge based segmentation. If it is possible to describe the issued object or feature in image with a mathematical term, the object can be found using this process. The method uses description of the investigated object in image space and it's transformation in parameter space. The unknown parameters are denoted from the mathematical description of the object by process of maximalization. There are two main uses of the Hough transform.

Hough line transform

A line, as the investigated object can be represented as

$$y = ax + b \tag{2.9}$$

or

$$\rho = x\cos\theta + y\sin\theta, \tag{2.10}$$

where ρ is the perpendicular distance from origin to the line and θ is a degree related to its slope as shown in Figure 11.a. Usage of (ρ, θ) parameters is more sufficient due to the description (2.9) causes difficulties for vertical line detection, where $a \to \infty$.

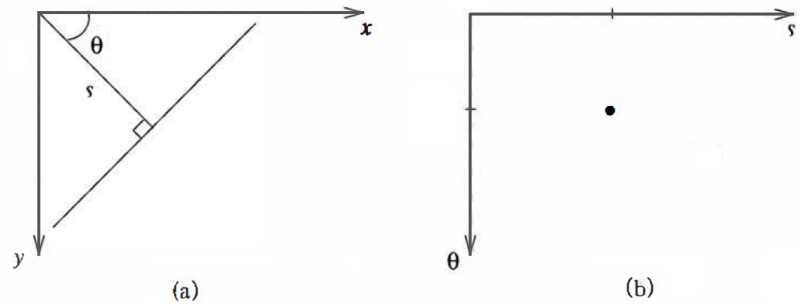


Fig. 11. Hough line transform. (a) Image space. (b) ρ, θ parameter space.

An edge image is the input to the algorithm. An infinite number of lines can go through any of the edge pixels however in reality the number of possible lines is always limited. By limiting possible directions the parameter θ is discretized and the parameter ρ is sampled by then. Parameter space (fig. 11.b) is transformed to a rectangular array which is generally called **accumulator array**.

The search is performed by increasing the particular cell in the accumulator array every time the issued line goes through the edge pixel. Cells represents the actual line will be incremented many times more than other combination of (ρ, θ). The resulting line is estimated by detection of local maxima in the parameter space.

Hough circle transform

The mathematical description for circles or curves is

$$(x - x_c)^2 + (y - y_c)^2 = r^2, \tag{2.11}$$

where (x_c, y_c) are center coordinates of the circle and r is its radius. Accordingly to the Hough line transformation the accumulator array is used, however in this case it is three-dimensional. If the pixel at the position (x_c, y_c) is at distance r from the edge point, the related cell in accumulator is incremented. Cell of the potential center has higher probability to be incremented and in the end of algorithm it is being found by local maxima detection.

As one can assume, computational demands are much higher than for line transformation. Prior knowledge about the object can decrease this demand significantly. To do that, we can simply provide limits for radius. Hough circle transformation implemented by OpenCV library (see 4.2) uses trick based on gradient information of the edges to suppress computational time.

2.5.3. Watershed segmentation

Technique is part of the region-based segmentation. The segmentation itself is performed in the greyscale image using its intensity peaks and valleys ie. its local minima and maxima in a gradient image. The process of the watershed algorithm, visualized in 2D, is presented in Figure 19.

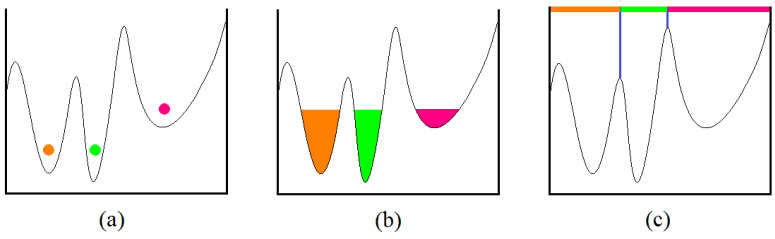


Fig. 12. Watershed algorithm. (a) Input labels of local minima, (b) Pouring of "water", (c) Segmented areas

The process of segmentation is quite straightforward. Every isolated valley is marked with a label. The index number of labels sets the dimension of final segmented set. The index-colored "water" is poured into valleys. As the water rises, depending on the gradients, water from different labels meets. At the boarder of different colors the segmented border is established. The process continues until the whole image is covered.

The main advantage of watershed algorithm is that it provides a solid and homogeneous output without holes, which could be useful in other processing such as computing center of mass of segmented area (see 2.7) or others. It is used for object localization or extraction. Matched templates can be very small or with large dimension. Increasing the dimensions of the pattern increases it's computational time significantly.

2.5.4. Matching

Template matching

As it is obvious from the name, the method is locating an object in the image by matching it to a known pattern or template. Because in reality the matching and matched objects are not an exact copies and the image usually corrupted by noise, geometrical distortion etc. it is not possible to create an absolute match. For that reason the best match has to be found using a maximal match criteria[15]. The matching criteria function can be represented by

$$C_(u,v) = \frac{1}{1 + \sum\limits_{(i,j)\in V} |f(i+u, j+v) - h(i,j)|}, \tag{2.12}$$

where f is the image to be processed, h is the matching template and V is the set of processed pixels in the image f. Function represents the match between f and h at position (u, v).

Template matching is simple technique used for object tracking and recognition. It is less effective with a rotated patterns which can be solved by rotated pattern matching. Nevertheless, this process increases the computation demands tremendously.

Feature matching

The process consists of two steps. At first it is necessary to obtain the points of interest i.e. features. There are many ways how to perform this operation. One of the most used methods is eigenvalue computation from the gradient image presented by Harris and Stephens[17] known as Harris corner detector.

The detector is based on 2×2 symmetric auto-correlation matrix

$$M(x,y) = \sum_{u,v} w(u,v) * \begin{bmatrix} I_x^2(x,y) & I_xI_y(x,y) \\ I_xI_y(x,y) & I_y^2(x,y) \end{bmatrix}, \tag{2.13}$$

where I_x and I_y are local image derivatives and $w(x, y)$ is a weighting window over the area.

For denoting the points of interest, eigenvalues of M are estimated for every pixel. Corner is evaluated if both eigenvalues λ_1, λ_2 are large. Score of pixel "cornerness" can be constructed by calculating

$$C(x,y) = det(M) - k(trace(M))^2, \tag{2.14}$$

where k is scaling parameter and

$$det(M) = \lambda_1\lambda_2 \tag{2.15}$$

$$trace(M) = \lambda_1 + \lambda_2 \tag{2.16}$$

Area with a score greater than a certain given threshold is considered a corner. Matching of the sets of features can be preformed based on nearest neighbors, but more techniques exist.

Instead of template matching, feature matching is a more complex and robust technique which handles both rotated images and noisy images well. With some limits, the feature matching is capable of handling differently scaled objects and images of the object captured from a different angle. One of many possible features matching usages can be blending of panorama photos or detection of partially hidden objects in the scene.

2.6. Mathematical morphology

By using convolution (eq. 2.6), the method of mathematical morphology passes a fixed pattern over the binary image. Size and shape of the of the structure kernel with the nature of the logical operation determines the outcome. A simple kernel could be a rectangular element of size 3×3 pixels, containing all ones. The basic morphological operation are erosion and dilation and their combinations opening and closing.

Erosion and dilation

Simple erosion is the process of elimination of the boundary pixels from an object. For a 3×3 kernel the result is the object decreased by two pixels after this single operation. Objects smaller than three pixels in diameter are removed. Because of its features, erosion is useful for removing objects that are too small to be considered for further processing or they are just of interest.

Dilation is the process of assigning into the object all the pixels touching it, increasing the size of output object by a layer of these pixels. If two objects are separated by the distance less than three pixels, they will become connected.

Opening and closing

Opening is a process of erosion followed by dilation. The main effect is elimination of small and thin objects, separating objects and smoothing peaks in boundaries of larger objects without modifying their area significantly.

Closing encloses dilation followed by erosion. The process fills small and thin holes in objects, connects nearby objects and smooth boundaries of large objects without significantly changing their area.

After image thresholding (2.5.1) often a noisy image is obtained. The resulting boundaries are ragged, the object is inhomogeneous with inner holes and the image contains small objects caused by noise. Significant improvement can be achieved with opening or closing operations. The desired effect is sometimes obtained with several iterations.

2.7. Shape representation

Convex hull

Convex hull estimation of the object is only one from many shape representation techniques. The boundary information is used to estimate various information about the object itself, such as its rectangularity and elongatedness. By comparing concave areas and inner area we can estimate compactness. Convex hull helps us to place less complex bounding shapes around the object to estimate it's position or rotation.

Many algorithms can be used. Sklansky in 1972[18] presented an algorithm which is based on a region concavity tree, where the tree is constructed recursively during the convex hull building. At first the convex hull of the whole region is estimated and from this area the concave residua are found.

Moments

As one can see from equation 2.17, moments describe properties of a probability density of a 2D random variable. They can be used for estimation the center of mass, also called center of gravity of the object.

Moments are generally classified by the order of the moments. The order of a moment depends on the indices p and q of the moment $m_{p,q}$. The sum $p + q$ is the order of the moment, where p and q are the indices along its first and second axes.

Considering this, the following moments of a function f are defined:

- zero order moment ($(p, q) = (0, 0)$)

$$m_{0,0} = \iint dx\, dy\, f(x, y) \tag{2.17}$$

- first order moments ($(p, q) = (0, 1)||(1, 0)$)

$$m_{1,0} = \iint dx\, dyx\, f(x, y) \tag{2.18}$$

$$m_{0,1} = \iint dx\, dyy\, f(x, y) \tag{2.19}$$

where $f(x, y)$ is the nominal binary value of each pixel which can be classified by value of $\{0, 1\}$.

The first order moments contain information about the center of gravity of the object. We get the required center coordinates of the object by computing:

$$\begin{aligned} x_c &= \frac{m_{1,0}}{m_{0,0}} \\ y_c &= \frac{m_{0,1}}{m_{0,0}} \end{aligned} \tag{2.20}$$

In digitized images the integrals are replaced with sums [19].

3. Inspection design

3.1. Plain board inspection process

Based on the types of defects listed in the Methodology (2.1.2), the range of observed defects was narrowed down. For the first version of this system, it is assumed the capability of tracking these plain circuit board defects:

- Bridge (short)
- Skip (open)

The limitation of the observed errors was made for several reasons. The range of possible errors is too large and it is not possible to cover the whole range within the time-frame. Defects listed in the Methodology have different occurrence, so monitoring was limited only to the errors of the conductive path itself. Defects that have been selected were most easily recognizable and well-described. At the same time, these two errors types are fatal for the operation of future electronic devices. Last but not least, low influence by the covering soldermask was expected.

In the beginning, two methods of image processing were proposed for plain board inspection. Both methods were based on the image subtraction (Section 2.4) where two images are compared globally and differences between them are further processed as they can be used as sign of defects. Methods differ with used type of segmentation where the first method uses an area based segmentation and second proposed method should use an edge segmentation. However, the hypothesis was such that the edge segmentation would not be as sufficient in the case of blurred edges.

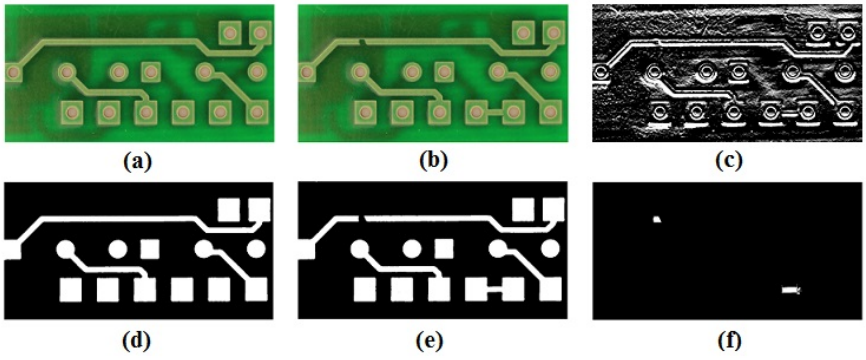


Fig. 13. Testing of image subtraction. (a) Reference image. (b) Tested image. (c) Subtracted image. (d) Segmented reference image. (e) Segmented tested image. (f) Subtracted and processed images (d) and (e), showing defect areas

The reason for the importance of segmentation can be seen in Figure 13 where a reference image of plain PCB and its reference are presented in Figure 13.a and 13.b. If we read these images straight away, we get the image presented in Figure 13.c. The reader recognizes that this image is basically unusable for the detection of possible defects. The appearance of the resulting image is primarily due to the fact that the

images can never really be taken the same way every time. Thus, the reference frame is always slightly affine transformed. Affine transformation, primarily image shifting and rotation, prevents the possibility of direct subtraction of images. For this reason, it is necessary to limit information entering the subtraction process. This can be accomplished by filtering or, in the best case, segmentation, as indicated in Figures 13.d and 13.e. The resulting subtraction on image 13.f is ideal when all imperfections are removed with morphological operations.

For initial testing and algorithm development an example testing board was created. The testing board is visualized in Figure 14. The board is double sided, the top side is provided with twelve soldering pads for packages of size 0805. The bottom side contains a simple pattern of conductive paths for simulating presented defects. The board files for manufacture are attached in CD (see Appendix B).

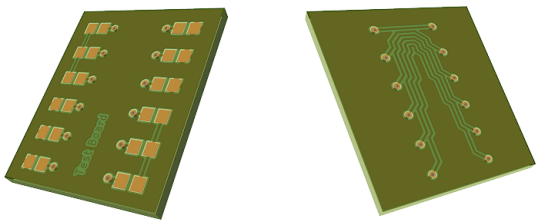


Fig. 14. Test board visualization

The test board should have been used to test algorithms primarily to monitor non-assembled boards so that the conductive pattern on the bottom layer deliberately contained the observed defects. The upper layer with solder pads was designed to develop algorithms that focus on the correct assembly of components on a printed circuit board.

Unfortunately, this board was not built, mainly due to lack of communication from the manufacturer and complicated logistics between the Czech Republic and India. For this reason, a virtual test set of images was created (like in Fig. 13) for further testing (see section 5.2.1).

Edge-based segmentation method

As mentioned above, the first proposed method was based on edge segmentation. Edges were obtained with usage of Canny edge detector (Section 2.3.2), processed and compared with edge subset from a reference board.

This solution proved inadequate behavior during testing. The process had problems with the identification of fuzzy edges and corners. However, the biggest problem was the high volatility of the results and the high degree of false positive results. Because this method is very sensitive, the image noise was often mistakenly marked as a defect. Therefore, it was necessary to filter the image significantly, which resulted in the removal of minor defects. For this reason, this method was not further developed and the development was directed towards a area segmentation method.

Area-based segmentation method

Area-based segmentation, unlike edge segmentation, segments individual image areas. A complete global image segmentation has been considered, where the conductive paths of the printed circuit in one group and the background of the other group are separated. Segmented objects directly correspond to the objects, such as paths, pads and the rest (background, soldermask). This segmentation method was the one used in final solution and it was proposed for testing.

The method is using color segmentation based on the known color threshold. Output of the threshold goes to the watershed algorithm which provides a global segmentation of the image. Because the core of this method was originally developed for an assembled board inspection, it is described in next section 3.2 more in detail. The only difference in this section is related to the color thresholds, driven by the user as it was presented in section 4.5, where the user set up segmentation properties through user interface. In detail the user specified single-layered board or double-layered board option as well as the board with or without the soldermask. These options change the way how the board is segmented. For example the color of background (soldermask) changes with number of copper layers. In Figure 13.a the reader can notice that the background color has two shades of green color instead of single-layered board, which has only one shade of green. This fact had to taken in consideration, because when the soldermask is applied on the surface of PCB, all areas including the conductive paths show a green color with close tones.

A block overview of the inspection process is visualized by process diagram in Figure 15. The process starts with loading a tested image and a reference Golden board. The Golden board in this case is in a form of a defect-less board image. The tested image is calibrated against camera distortion, the automatic white-balance is performed and the image is cropped to the board dimensions. Global smoothing filtering is applied. This arrangement of pre-processing methods showed the best results in subsequent segmentation.

After that the inspection part follows. The image is segmented with color thresholding and watershed segmentation. The threshold limits are known in advance since they are experimentally estimated. After segmentation, an output binary image is subtracted with the reference binary image. The differential image is then subjected to morphological erosion which was selected as the best method for suppressing imperfections caused by affine transformations of the images. The residua after morphological filtering are marked as possible defects. The method can distinguish the "Bridge" and "Skip" fault by double differentiation. The "Skip" type of defect is positively marked in the differential image when subtracting the tested image from the reference. The "Bridge" type is positively marked when subtracting the reference from the tested image.

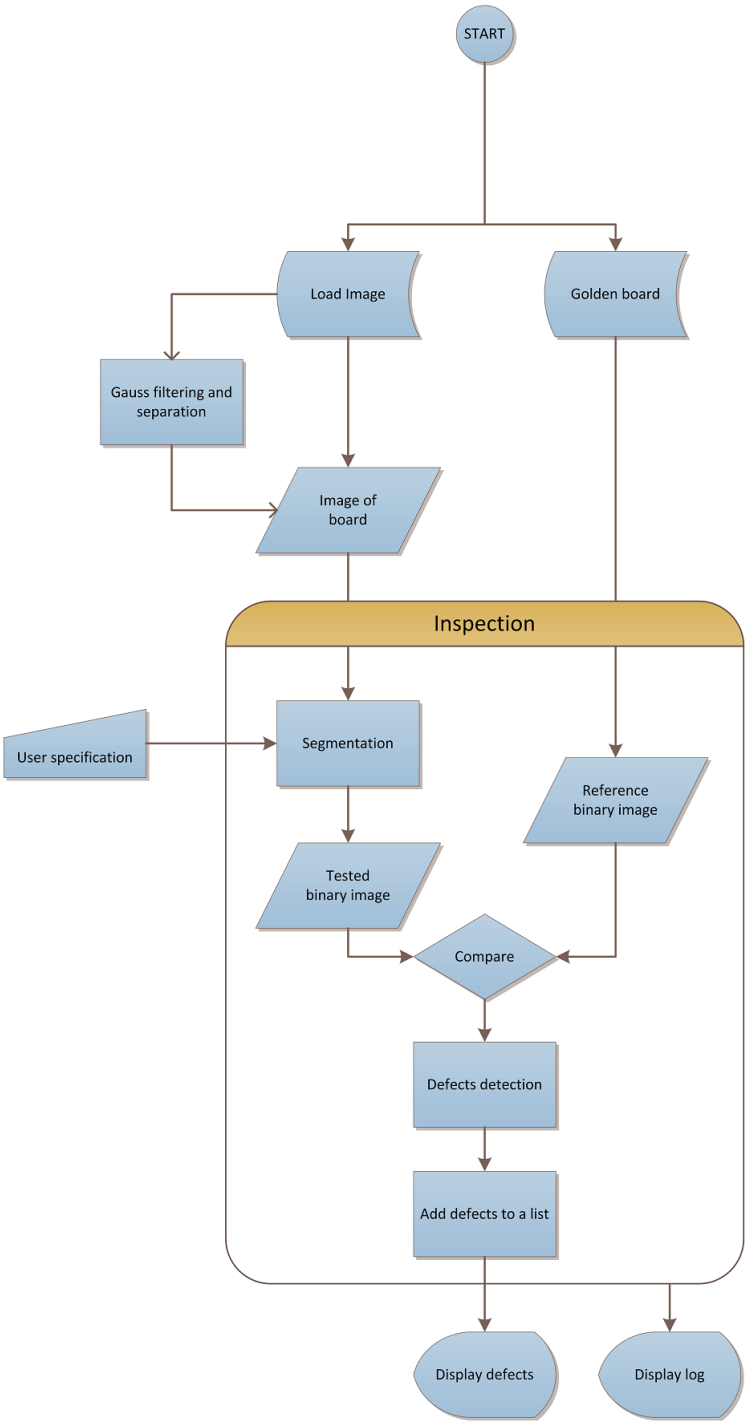


Fig. 15. Plain board inspection process

3.2. Assembled board inspection process

Based on the possible assembled board defects, the inspection was focused on defects related with component placing. These defects are one of the basic and primary errors observed on assembled circuit boards. As shown in the user interface in section 4.5, the inspection was focused on specific errors:

- Correct type of component
- Missing component
- Correct polarity
- Correct position/rotation

The aim was to create the most versatile method that could be used as much as possible. Such method could, at best, be used to track as many of the types of defects. For that reason, a method based on color thresholding combined with watershed algorithm was introduced. Color thresholding is a relatively simple technique, but it requires stable color and illuminance condition. Watershed segmentation algorithm provides globally segmented images which perfectly fits the specified needs. On the other hand it requires precise initial segmentation conditions for correct segmentation.

For the best understanding of the assembled board inspection, explanation of the method core follows. The process is described on an example board which the reader can further see in Section 4.6 dedicated to the Golden board creation. The example board is originally a cropped image of a test board provided by the Intronix company[1] as shown in Figure 16.

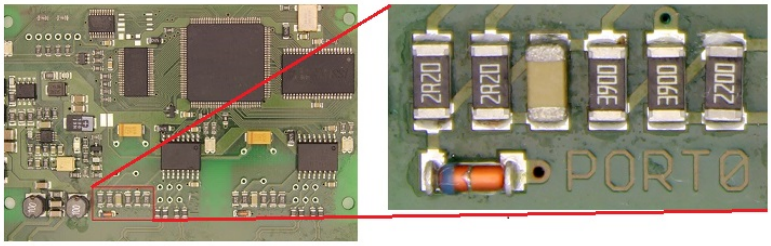


Fig. 16. Example board photo

The entire inspection process is illustrated on the flowchart in Figure 17. The first part of pre-processing is essentially the same as in the case of plain board inspection and therefore the reader is referred to this section (3.1). The process differs mainly in the inspection part as well in a form of Golden board data structure.

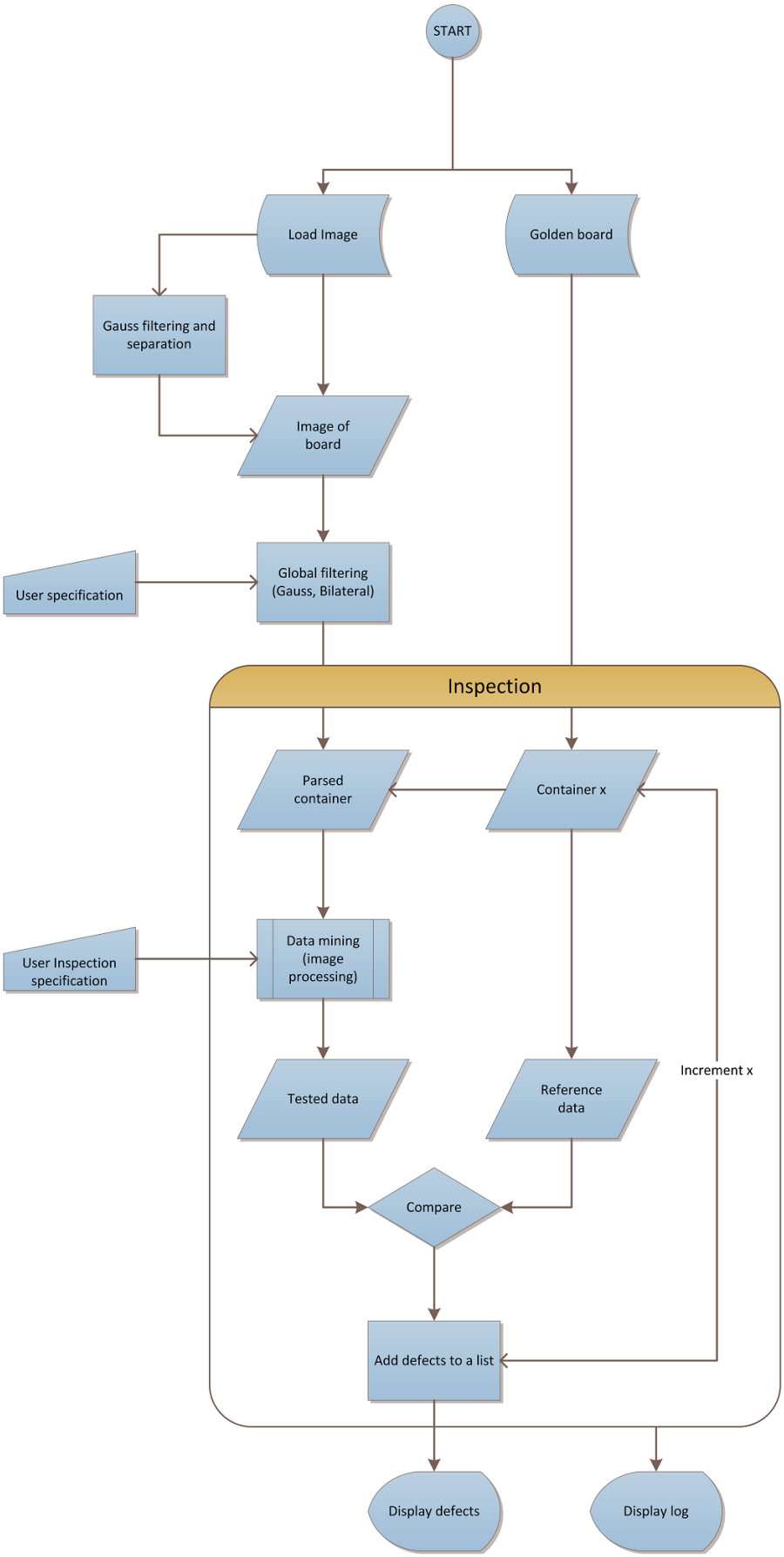


Fig. 17. Assembled board inspection process

The main image processing is being done in the block "Data mining" at the beginning of Inspection. Before that, the Golden board is loaded. The board inspection is divided into containers as can be seen in Figure 24 where the Golden board structure is presented. Inspection is performed on container after container for all components of the board. One can see the tested area is exactly the outer area set up by the green rectangle during the Golden board estimation (see Figure 25).

Initially, three color space representations of the image are obtained, i.e. RGB, HSV and CIE L*a*b* color space. Then, the color thresholds are obtained separately for each color space as it visualized in Figure 18 in a such way to threshold the component package from the background.

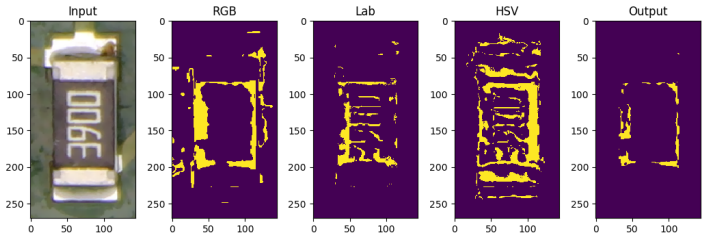


Fig. 18. Color threshold in multiple color spaces

Threshold limits were taken from the color library which had been estimated in advance experimentally. The final operation of threshold process is a combination of results from all color spaces. Experimentally, it was discovered that the logical combination of

$$Out = thresh_{Lab} \cdot (thresh_{HSV} + thresh_{RGB}) \tag{3.1}$$

provided the best results. Symbol "·" represents the logical conjunction and symbol "+" represents the logical disjunction. A good result contains the amount of least false positive pixels as possible. This is important, because the output of thresholding is used as a marking set for the watershed algorithm, which is described below and the output threshold does not have to be eroded so significantly.

After a similar threshold process where the background set of points is obtained, the segmentation part follows. As the most suitable method chosen was the watershed segmentation, since it globally segment the whole area. Watershed segmentation well overlays the part package and provides precise boundaries which can be used for further data acquisition.

When providing the input set of foreground and background seed pixels obtained by thresholding, the result is a homogeneous segmented area without any holes as can be seen in middle of Figure 19. At this point the image processing part ends and the data mining specified by the user can be started.

Correct type control

The feature is that the vast majority of components of different types differ in the colors of their package. When the correct type control is selected, the data from multiple color thresholds are being examined. Output from equation (3.1) is processed for every color from color library and the correct part type is estimated by pixel appearance maximalization, where the biggest thresholded area from the color threshold is selected.

In the next step, the segmented image from watershed algorithm is used for approximate size estimation of the parts package. This serves for type determination of

components with similar color but different size, such as SMT resistors and integrated circuits. At this point the part type is estimated.

With knowledge of the reference part type from the Golden board the possible defect can be verified when the tested part does not match the reference part.

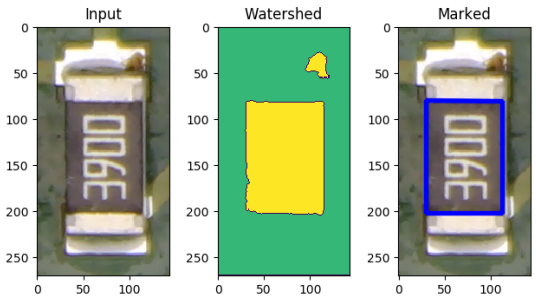


Fig. 19. Watershed output

Misplaced component & Missing component

These two types of defect use basically a similar method developed for the correct part type estimation. Differentiation procedure does report an defect when the difference between tested and reference part appear (misplaced part) and it reports the defect when an empty pad is being found at the place where the component should be assembled (Missing component).

Translated/rotated component

Control uses the segmented image from watershed algorithm. The segmented area which has mostly a rectangular shape is encapsulated with convex hull. This method perfectly fits for next steps i.e. enclosing the shape to another rectangle as it is visualized in Figure 19. As can be seen from the figure, the method is resistant to the occurrence of secondary objects in segmentation and can filter them so that the position of the component is not affected by this imperfection.

Position of this rectangle is used to determine position of the center and translation of the parts. With knowledge of a correct position and rotation, based on the position of pads from Golden board, the error of absolute rotation can be estimated as well as position error. Position error is calculated as euclidean distance of center of a part and a pad. Defect is reported when some of these values exceed the given limit.

Polarity check

Polarity control uses outputs of the threshold procedure as well as from the watershed algorithm. If the information about a part from the Golden board suggests the fact the tested part holds polarity, the secondary color threshold is performed. In this case only colors of polarity markers are used. This procedure finds polarity marks which can be distinguished on the bases of its colors, such as side color strips. After segmentation the center of its mass is determined.

Because the polarity marks usually also holds predefined shape, identification based on it is also suitable. Thus, the Hough circle transform is performed to indicate the positions of any circular polarity marks, such as circular dots or hollows on the packages. This method perfectly fits for marking the circular objects and for that reason it was selected.

Correct polarity is determined by correlating the coordinates of the center of the part with estimated center of the mark. Left, right, top or bottom polarity is estimated and compared with the information from the Golden board. This method is not resistant against a rotation of the test board. The error is reported even if the polarity marker has not been found.

4. Implementation

4.1. Programming environment

Selecting the programming environment and the associated programming language is the first important choice before the implementation itself starts. It is necessary to take into account all factors related to the development of the new software, i.e.:

- The developer's language experience
- The suitability of the programming language for the application
- Availability and variability of libraries and frameworks
- Periphery compatibility
- Documentation

On the other hand, for prototyping, the initial development of new software, there is little emphasis on overall system stability and end-use. This includes mainly computational difficulty and memory management.

Additionally, the overall user-friendliness and appearance of the resulting user interface have to be emphasized, as this serves primarily to test the system itself. It is important to note that the GUI is one of the most important benchmarks for user comparison, where the user evaluates the system primarily from the point of view of its control, friendliness and overall appearance. If the developed system has ever been so produced, it is necessary to take this fact into account.

Due to the author's experience, three programming languages, C++, Java and Python, were considered. The availability of suggested libraries (see section 4.2) is similar for all these languages. Also, the documentation of these libraries is well maintained. However, for fast prototyping, Python was assumed as the best choice. The biggest advantages of Java (platform independence) and C++ (speed and memory management) were not considered as important for development.

Because in this work the industrial periphery is used, it is necessary to count with the set of languages which are compatible. Because the camera manufacturer, the Basler company, provides their Pylon library (see 4.2) written in C++, this language had to be considered for the developing as well. The light controlling system of RGB light strip (see 5.1.2) was developed in C++ as well, since this language is more suitable for embedded microprocessor systems and its firmware development.

Python was chosen because it meets all the previous criteria very well and because the development in this language is fast and the resultant code is easy to read even for people with basic programming experience. Nevertheless, Python comes with Anaconda distribution which contains multiple useful side-packages as well as Spyder IDE. Spyder provides well maintained debugging options and with comprehensive memory and variable explorer it perfectly fits for application development.

4.2. Frameworks

In this section, a brief description of the utilized framework is presented. It would be very ineffective and time consuming to code all methods which others managed to program before. For that reason, different frameworks and libraries where used in various parts of the research work. Different frameworks were applied for GUI, image processing, image capturing, serial communication and others.

PyQt

PyQt is a multiplatform toolkit made by a british company Riverbank Computing Ltd.[20]. It provides solutions for GUI programming in Python and is freely available under GNU license.

There are other alternatives, such as PyQTK, Tkinter or wxPython but PyQt was selected mainly because it is generally the most used one and mainly because Python Anaconda distribution comes with PyQt packages pre-installed. Complemented with Qt designer software, which allows to build the interface mainly by drag&drop principle, the GUI development with PyQt is fast. Nevertheless, PyQt's visual impression of final interface looks very good which is one of the important factors as well.

Particularly PyQt5.9.2 has been used in this works, since it was the latest version available at the time of developing the application.

Pylon

Pylon is a collection of drivers and additional software which the Basler company (see section 5.1.1) provides as the supporting software pack for all their products. It has an interface independent SDK and it allows to develop applications for Windows, OS X or Linux for x86 and ARM based platforms [21].

Pylon drivers are natively written in C++ and the development of python application had to be adapted accordingly. There were two options, to build a python wrapper around the pylon application and use it for further development or build the pylon application separately and use its output in python application afterwords.

Because Pylon software was used mainly for image acquisition by the industrial camera, the second option was selected, where the image is obtained separately by the Pylon application and then it is processed by the inspection application. This solution is less effective but sufficient enough for developing work of this manner.

OpenCV

OpenCV is one of the most used frameworks for computer vision and image processing. It is an open-source released under a BSD license and is free to use for academic or commercial applications. The library has more than 2500 optimized algorithms, which can be used in many applications such as face recognition, object identification and tracking, scenery recognition and many others. OpenCV's user community has more that 47 thousand users and the estimated download count of this library exceeds 7 milion (in the year 2015).

The OpenCV library supports Mac OS, Android, Windows and Linux in C++, C Python, Java and MATLAB interfaces. The library is written natively in C++ and has a templated interface that works seamlessly with STL containers. OpenCV version 3.2.0 was used for the development of image processing algorithms in this work. [19, 22]

4.3. Project libraries and additional software

It is necessary to briefly introduce other supporting software used in various parts of this work, although they do not directly relate to image processing itself.

Matlab was used for initial testing, color segmentation design and it was also used for initial algorithm testing during the design of image processing structure. Its built-in threshold color application was also used for color library estimation (see section 4.6). Version Matlab 2015b was used.

Arduino IDE is an integrated development environment specifically created for Arduino firmware development. Arduino microprocessor platform is such well known tool that the author presumes that further describtion is not necessary. However the Microsoft Visual Studio was partially used for Arduino firmware development, the beginning of the programming was done in Arduino IDE. Specifically, the color illumination control had been developed to run on Arduino platform (see section 4.4).

FastLED 3.1 is a library for Arduino platform providing easy and efficient control for a wide variety of LED chipsets. It also contains low-level functions for 8bit mathematic operations usable in manipulating RGB values of color channels and also color space transformations HSV→RGB and vice versa [23].

Inventor is a 3D CAD modeling system developed by Autodesk. Inventor is mainly used for mechanical design in industry and production. It allows direct 3D design including destruction simulations and many other functions. It was mainly used for designing of a white-box, major part of the image capturing device (see section 5.1.2). Inventor was used due to necessity of 3D modeling and selected because of authors prior experience with this software.

Eagle is a PCB design and schematic software which was used to design a several testing boards which were supposed to use for experiments.

Pylon Viewer is a part of pylon suite presented above. It is used for quick operability testing of Basler cameras and their initial setup.

Qt designer is a tool which allows to graphically build a graphical user interfaces (GUIs) using Qt functionalities or so-called widgets. It comes as a part of Anaconda Python distribution.

digiCamControl is an open-source camera controlling software. It is even free to use under MIT license for commercial applications. It provides PC remote control via USB protocol for a wide range of commercial cameras. It provides functionality for most generally known brands such as Nikon, Canon and Sony cameras. It supports remote functions such as image capture or live view, and allows the remote control of image capturing properties such as exposure, aperture, ISO, white balance and many others. It was mainly used for remote control of DSLR camera used in experiments (see section 5.1).

Project data structures, libraries

The implementation of the project itself includes several parts that are grouped into single entities. Two libraries with different purpose were created:

- colorLib.py
- processLib.py

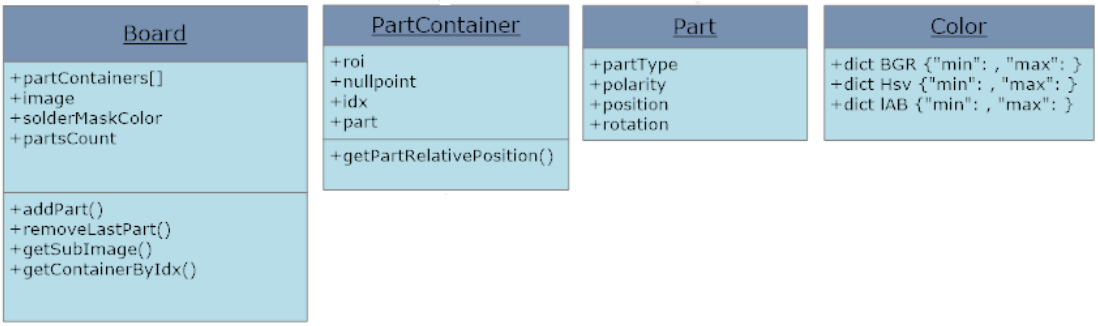


Fig. 20. Board management classes

ColorLib basically stores color thresholds in all used color spaces (RGB, HSV, CIE L*a*b*) and it defines the major Color class for color handling.

ProcessLib contains definitions of all necessary classes which are presented in Figure 20. The classes serve for various operations during the process of inspection and they carry the database for inspection data. The processLib also implements various methods, presented in the Methodology section, which are used during the inspection as well.

4.4. Illumination control

Illumination control (hardware is discussed in section 5.1.2) provides remote control of white-boxs RGB LED-strip. It substitutes an information converter combined with power amplifier. As seen in the structure presented in Figure 21, the color requirements are transformed via serial communication to the microcontroller. Microcontroller processes the information and controls the RGB driver via another serial line. Commands to the driver are performed with usage of FastLED library.

Commands to the microcontroller are propagated in a form of char sequences separated with endlines, as it is usual for serial communication. Each "frame" contains:

- Mode option (On/Off)
- Color space (HSV/RGB)
- Channel1 value
- Channel2 value
- Channel3 value

Process of Arduino loop is described below in Algorithm 1. Every time a new line is received, it is processed and directly transmitted to the driver in one loop.

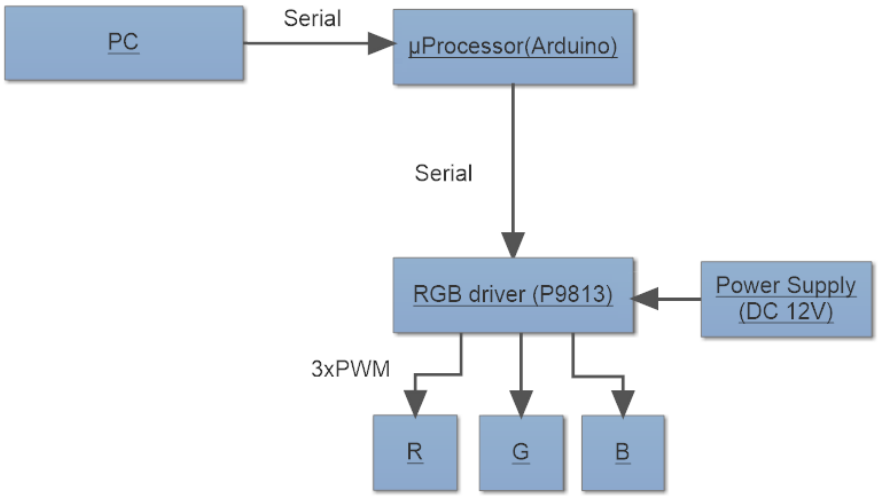


Fig. 21. RGB LED-strip control

Input: Char_input_line
Output: Command
if *new line available* **then**
  Parse *mode*, *colors*, *color_space* **from** *Char_input_line*
  if *color_space* ***is*** *HSV* **then**
    convert *colors* to RGB
  end
  Build *Command* **with** *mode*, *colors*
  Send *Command*
else
  do nothing
end

Algorithm 1: Arduino loop

4.5. User interface

For controlling the inspection application a graphical user interface was proposed as one can see in Figure 22. GUI was created using widgets from PyQt framework and the layout itself was created in Qt designer.

When creating a graphical user interface layout, emphasis was placed on the overall clarity of the entire composition. The Figure 22 shows the individual user interface components numerically. These components are:

1. Main control panels
2. Image display area
3. Message/log window
4. Status bar
5. Defects information window
6. Semaphore

In the "Menu" icon at the top bar the user can select the type of used camera, which formally changes the parameters of calibration and selects a related color library during the process of inspection. Application can be also terminated here.

The whole window is scale-able to a certain limit. The main components of the

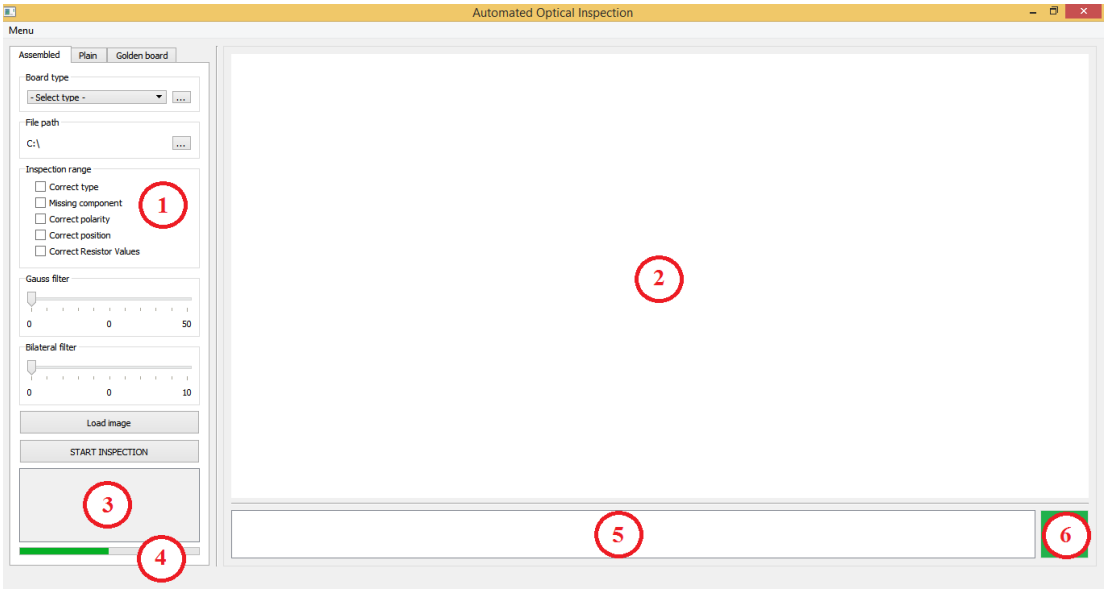


Fig. 22. Automated Optical Inspection GUI

interface are the Main control panels and Image display area. The Image display area shows currently inspected image and during the process defects are visualized there eventually. When the Golden board is created, it serves as an interactive window for data acquisition (section 4.6).

Control panels are positioned on the right side of the window as it is a widely used layout which we can meet very often among many different applications. Control panel consists of three different cards, each for different functionality. The cards are:

- Assembled - Enables an assembled board inspection
- Plain - Enables a plain board inspection
- Golden board - Serves for Golden board creation

The bottom part of every control panel's card is identical for all of them and it contains the message/log window and the status bar. Status bar informs the user about current state of the process and the message/log window provides further information about the inspection itself.

The last two components are situated below the image display area and they are the Defects information window and the Semaphore in the bottom-right corner. The window provides information about an error or errors that may have been found. The semaphore informs the user if the inspection was successful or not. If a defect was found, it turns red. If no defects or errors occurred during inspection it turns green. This allows the operator to check the result at one glance and possibly to investigate the defect when red semaphore occurs.

Control panels are presented in Figure 23.(a) and 23.(b). Panel 23.(c) serves for assembled Golden board creation and is discussed below in section 4.6.

The assembled card can be used for inspection of assembled printed circuit boards. Top to bottom, the board type can be selected from the list of prepared Golden boards or a new Golden board can be loaded from another direction by pressing the push button on the right. Below that, the file path of the inspected image can be typed in. Next, the operator can select the range of performed inspections by checking the

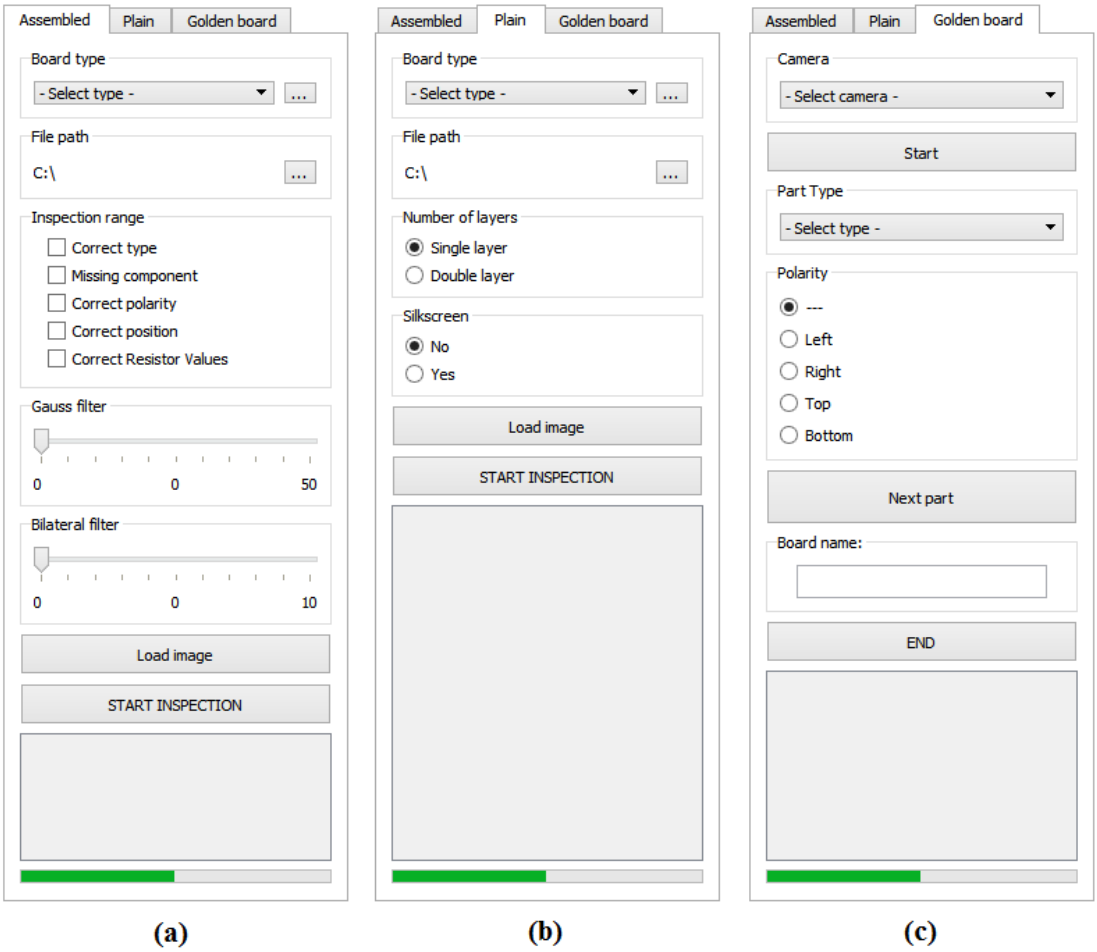


Fig. 23. AOI control panels

provided check boxes. When checking the position check box, user is allowed to type in the limit values. The check box for correct resistor value inspection is prepared, however this function was not completely implemented (see Discussion).

Performance of the inspection can be adjusted by two horizontal sliders which manipulates the Gaussian and Bilateral filter (2.3.2). For Gaussian filter it is the size of rectangular kernel in pixels and for Bilateral filter the user is modifying number of iteration. When the particular slider is set to zero, the filtering is disabled. Maximal size of Gaussian filter kernel is 30 pixels, the highest number of Bilateral filter iteration is 10. For simplicity of the user interface, it was assumed that the users do not need to know what exact parameters of the filters they are actually modifying. The logical manner of increasing number with increasing influence of the filter was found to be sufficient.

The last two push buttons are placed below the horizontal sliders. By pressing the Load image button, the image from inserted file path is loaded. After pressing the START INSPECTION button the process of inspection is initiated.

The last two elements of the panel (i.e. message/log window and status bar) are already mentioned.

The plain board inspection card is similar to the assembled card where the components for the selection of Golden board and file path of the inspected image are the

same.

Inspection can be adjusted with two sets of radio buttons. In the first set the operator informs whether the board is single layered or double layered. This fact affects process of segmentation during image processing. In the second set, information about the existence of soldermask can be provided to the system, which is also a required parameter necessary in the segmentation process.

The rest of the card is similar to the assembled inspection card and contains Load image and START INSPECTION buttons with same functionality as in previous case and message/log window with status bar.

4.6. Golden board creation

The so-called "Golden board" must be created before the inspection begins. The Golden board represents the ideal board without any defects to serve as a reference for comparison with the tested board. An interactive process has been created for that reason, where the user can create a new Golden board for assembled board inspection. For plain board inspection a reference image only serves as a Golden board and it was not implemented. For that reason, only assembled board problematic is presented below.

4.6.1. Data structure

Creation of Golden board represents filling a board database with necessary information. Structure of the board database is constructed using some objects presented in section 4.3 and relation of these objects can be seen in Figure 24.

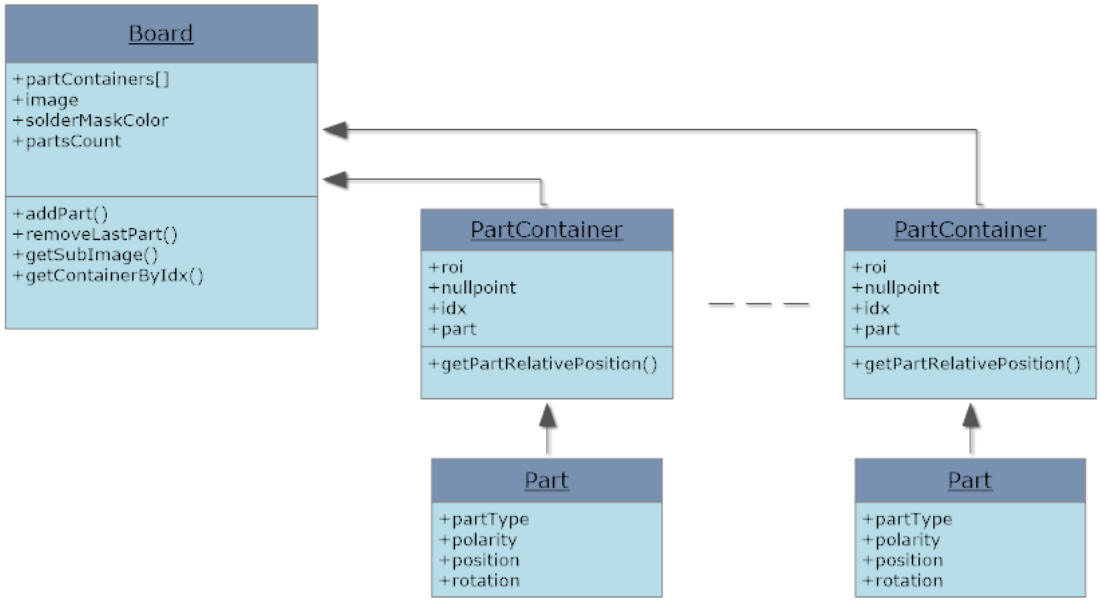


Fig. 24. Board database structure

The presented board structure represents not only Golden board structure, but it is used for general board representation among the process of inspection, where the top-level object Board basically carries all necessary information about each issued board.

In Figure 24 we can see the Board class carries an image of the board with additional information about total count of assembled parts and color or appearance of the soldermask. It also contains a list of all part areas i.e. part containers where each container

contains exactly one component or part. The number of containers is equal to number of parts in the Board class counter.

These parameters are being saved for every issued part:

- Part type (resistor, capacitor, diode etc. including type of housing)
- Polarity
- Position of part in container
- Rotation of part

The part type represents a specific color threshold from colorLib and it is used for color identification of the part during inspection.

4.6.2. Color library

Color library contains a palette of color thresholds known to the inspection system. This information is carried by the PartType value of the part. Each color is represented by its minimum and maximum threshold values for each color space i.e. RGB, HSV and CIE L*a*b*. For RGB color space minimal and maximal values of each channel are stored. For HSV color space the hue channel threshold values are stored and for CIE L*a*b* color space the threshold values of a, b channels are being stored. Thresholds were experimentally estimated and some of those colors (parts) can be seen in Table 5 in section 5.2.2.

By creating each color, we essentially create the identification or distinguishing mark of the component package. When identifying the component during inspection, it is possible to compare color sets from this library and determine the component that is really involved according to the best match method (see section 3.2).

4.6.3. Interactive process

In Figure 23.c the Golden board control panel is visualized. This panel serves for creating a new Golden board in an interactive process which is further explained.

Instead of adding new components to the color library, it is assumed that the creation of new Golden boards will be more frequent, mainly due to the production of smaller series. Therefore, this process has been included in the user interface itself to help automate this activity and make it more user-friendly.

The process itself was designed to be as simple as possible and did not take too much time. Thus, the overall simplicity of the resulting data input was emphasized and thus a partial interactive solution was chosen. The user enters certain parameters, namely the position and rotation of the component, by clicking in the interactive area i.e. the image display area in Figure 22. This solution greatly simplifies the entire process. Otherwise, the user would have to manually position the coordinates and some method of relative measurement would have to be created. In this way, the system receives the data directly in the format in which it processes it further.

The process of creating the Gold Plate is as follows. The user first uploads the board image to the system using the appropriate fields on the Assembled or Plain cards. The image needs to be taken on a white background. The image is automatically adjusted its white balance, and the board is calibrated and displayed in an interactive environment.

First, the appropriate camera has to be selected and the process is started by pressing the "Start" button. When the button is pressed, the other menus are unlocked and the user is navigated through the message/log window. The user is then prompted to indicate the position of the pad of the component. The user double-clicks the four

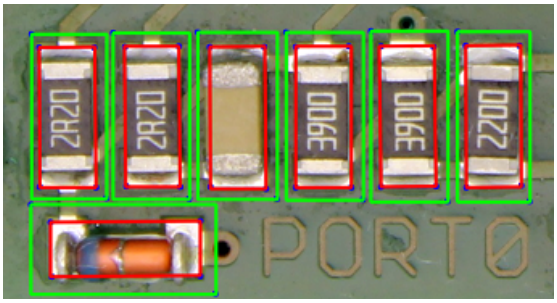


Fig. 25. Golden board creation, marked board

outer corners of the pad of the component, creating a red rectangle after the fourth marking, which determines the correct position of the component by its position and rotation. In addition, the user is prompted to mark the area in which the component is located. The area must be marked with two points. These two points determine the diagonal edge points of the rectangle, which is always parallel to the boundaries of the image. For the proper inspection functionality, it is good to mark a sufficiently large area around the component to include the entire component, including the surrounding area with a soldermask. The surrounding area is marked with a green rectangle. This is where the interactive part of the entry ends. Result are presented in Figure 25 where the interactive process is performed on an example board.

In addition, the user in the control panel selects the relevant component type from the menu and selects the polarity of the component. The component is confirmed by pressing the "Next part" button. The process is necessary to repeat the component after the component for all the parts of the board. If all the parts are entered, it is necessary to fill in the name of the new Golden Board, exit the process and save the board by pressing the "END" button. If the wrong information is entered during creation, it is necessary to restart the process. An application in this regard is still in prototype mode and it is not implemented for this type of error.

4.7. Image pre-processing

Before the image can be processed it has to be prepared for the inspection. There are basically three necessary steps which have to be completed:

- Distrotion removal
- White-balance
- Cropping

Because usage of a real camera is presumed in the application, camera calibration is required for suppressing distortion of the image. The OpenCV library provides well prepared functionality of camera calibration based on obtaining calibration images of known pattern, in this case chess board pattern of size 9×6 points.

In the process of calibration, the pattern is captured several times in various positions in order to cover most of the visual range of the camera. By increasing the number of captured images, the precision of calibration increases as well. Theoretically at least two snapshots are required for calibration parameters estimation, however for good results more snapshots have to be captured. In the end the calibration parameters (eq. 2.5) are obtained. The calibration process has been done for used cameras in advantage so the user of the application can only select the calibration parameters from the list by

setting the correct camera option in user interface, as it was presented in Section 4.5.

White-balance was implemented based on the method from David Kay[24]. It is based on normalization and re-scaling of the image histogram and it can be described in several steps, such as:

Input: Input image
Output: Balanced image
for *all channels* ***in*** *Input image* **do**
  Find the low and high percentile values **in** *channel*
  Saturate below the low percentile and above the high percentile
  Re-scale *the channel*
end
Merge *channels*

Algorithm 2: White-balance process

Such simple white-balance algorithm could be used with two constraints. The image of the board always has to be captured with white reference background and the board image should not contains flares. Both constraints were satisfied by using a white-box which is more discussed in section 5.1.2.

The final step to prepare an image for the inspection process is cropping of the image. The fact that examined board dimensions and shape are the same every time was taken into consideration. For that reason and the reason the board is always captured on white background from same distance the method of board separation was implemented.

Since the board shape is easily segmented from the white background, the binary image of foreground (board) and background (white space) is created. The board object is then encapsulated to the rotated bounding rectangle which sets borders for cropping. After that, the cropped image is rotated in the way that the longer side of the rectangle is horizontal after transformation. For this reason the position of the board does not have to be precisely the same every time, but the exact location is beneficial since every affine transformation of the image brings deformation in the picture. The board in the picture should not be rotated more than $\pm 90°$ from the wanted position for inspection.

5. Testing

In this section, the physical testing of the resulting inspection system is presented. Used cameras and device for image acquisition are listed (section 5.1.1). Their choice and use is discussed. Last but not least, the testing data acquisition processes for the plain board and assembled board inspections are presented.

5.1. Experimental setup

For the control of printed circuit boards, a physical station had to be designed. This station had to fulfill three purposes:

- Ensure the constant lighting conditions
- Create a stable composition of camera and test object.
- Provide high-quality image data

The resulting design had to be composed from the generally available construction elements and had to be affordable to preserve idea mentioned in the Draft proposal. Another important factor should be transportability. The entire station should be of a lightweight construction and this structure should be dismountable. This had to be ensured as most of the software development took place abroad, specifically in Chennai, India. The experimental setup had to be transported personally to the final destination.

The setup should be able to hold a different types of cameras since the multiple options were suggested for the PCB inspection.

5.1.1. Camera equipment

For the inspection application two types of a camera system were proposed. As an industrial solution the Dart camera type daA2500-14uc from Basler company was used. The Dart camera is a part of the basic area scan cameras. Some parameters are presented in Table 1. Further information and technical description reader can find at the vendors website[25]. The camera was complemented with 12 mm S-mount lenses

Parameter	**Value**
Resolution	2592 × 1944 px (5 Mpx)
Sensor Size	5.7 × 4.3 mm
Mono/Color	Color
Pixel Bit Depth	8, 12 bits
Interface	USB 3.0
Lens focal length	12 mm

Tab. 1. Basler Dart camera - technical description

As the customer electronic camera solution the Canon EOS T3i Rebel with set 18-55 mm lens was used. In Europe this camera has been sold as type EOS 600D. This

camera falls into the low-end DSLR group of cameras. Further information and technical description reader can find at the Canon website[26]. Selected technical parameters are listed in the Table 2.

Parameter	Value
Resolution	18 MPx
Sensor Size	22.3 × 14.9 mm
Mono/Color	Color
Interface	USB 2.0
Focal length	Equivalent to 1.6x the focal length of the lens (lens 18-55 mm)

Tab. 2. Canon camera - technical description

Both cameras are capable to provide data transfer via USB interface which is suitable for the application and can be remotely controlled through this communication bus. Finally, the cameras are equipped with a mounting elements, thus the easy assembly to the experimental station is possible.

5.1.2. White-box

To ensure a sufficient light conditions for the capturing scene the so-called white-box composition was used. White-box generally consists from a closed space covered with a diffusive surface illuminated by a dispersed light source. Dispersed multi-directional light suppresses shadows and it enhances contrast of the scene.

As an illumination technique the combination of the Bidirectional lighting and the Diffuse lighting was used (see section 2.2.2). This ensures a shadow-less environment when capturing the assembled boards.

For the best idea the photos of the real constructed device the reader can find in Attachment A. The aluminum modular structural system provided by Alutec company[27] was used as the main construction element. The smallest offered profile of 20 mm was used to meet the weight conditions. The sides of the chassis were fitted with a white diffusive desks from polycarbonate material. The top of the box was equipped with a light source. Approximately in the middle of the box the dispersion layer is mounted. Underneath, there is the capturing space with hollow bottom. When capturing an image, the whole system can be lifted and put on the tested board on white background prepared for capturing.

As the main light source the high luminous white LED strip was used, complemented with supporting RGB LED strip required for color contrast enhancement. Technical parameters of the LED strips provided by the dealer are presented in Tables 3 and 4. Manufacturer of the light strips is unknown.

Parameter	Value
Input voltage	12 V
Power	20 W/m
Light temperature	4000 - 4500 K
Light flux	2070 lm/m
LED density	120/m
Angle of light	120°

Tab. 3. White LED-strip, technical parameters

Parameter	Value
Color variant	RGB
Input voltage	12 V
Power	14.4 W/m
Light flux (total)	800 lm/m
LED density	60/m
Angle of light	120°

Tab. 4. Color LED-strip, technical parameters

In total, the light consisted of 84 white LEDs and 48 RGB LEDs, resulting in 14/11.5W (white/RGB) light source. For cooling the LED strips were attached to the special aluminum frame. The photos of the frame are presented in Appendix A.

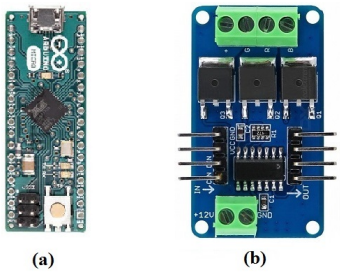


Fig. 26. RGB control hardware. (a) Arduino Micro. (b) P9813 based RGB LED driver

For the final testing the color enhancement was not used, however the RGB light strip was installed to the image capturing device. This was caused due to the late delivery of the RGB light driver (Figure 26.b). Although the color enhancement system was designed, the driver was delivered in the last week of research at a foreign university and could not therefore be used for testing.

Overall dimensions of the white-box were adapted in a way the both cameras could be used without any modifications, based on cameras dimensions and its lens focal distances. This included the position of the disperse layer and correct handling of mounting device for both cameras. In final design the cameras can be easily replaced just by attaching the relevant removable mounting profile.

5.2. Inspection testing

Inspection testing was performed on a set of real-production test boards. Unassembled plain printed circuit-boards were donated by Pragoboard s.r.o.[9] and assembled PCBs were donated by Intronix s.r.o.[1].

5.2.1. Plain board testing

As the board directly designed for testing was not delivered to the place of testing in India, a replacement board was selected (see Figure 27). Obtained images of a flawless reference board were taken and defects were embeded manually in the image using a raster graphical editor.

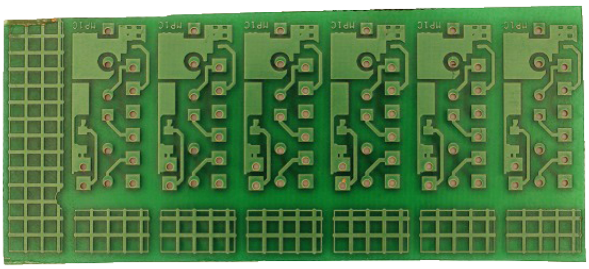


Fig. 27. Plain board for inspection testing

Because the test plate was not defective, it was recorded as a Golden board in the first stage of experiment preparation. In the second phase of the experiment preparation, errors were artificially created in the board. Range of errors follows:

- Bridge errors with various minimal thicknesses
  1. 20 mil
  2. 10 mil
  3. 5 mil
  4. 1 mil
- Skip errors with various maximal thicknesses
  1. 20 mil
  2. 10 mil
  3. 5 mil
  4. 1 mil

For comparison, the 20 mil thickness roughly corresponds to the width of the conductive paths of the printed joint in the Figure 27. The set of 10 images were obtained by the Canon and the Basler camera for the quantitative comparison. Each pattern on the board was defected by the presented defects. In total, every thickeness of bridge error was presented on the board pattern once and every skip error was presented in the same amount. The white LED illumination was used with maximum input power.

In total 10 images were captured, holding 80 various defects of conductive paths.

5.2.2. Assembled board testing

The board for the assembled board testing was provided by the Intronix company and can be seen in Figure 28. The board is assembled with various components due to it is suitable as a test example. Following assembled board defects were examined:

- Correct part type recognition
- Missing part
- Correct polarity estimation
- Correct position of the component

The board provides a wide range of components including several integrated circuits. For the testing the set of part types from Table 5 was examined, since it correlates with the experimentally estimated color library.

PartType	**Description**
resBlack	SMT thick film resistor, various packages, black color
ceramicCapWhite	SMT ceramic capacitor, 0805 package, white color
ceramicCapLightBrown	SMT ceramic capacitor, 0805 or 1206 package, color light brown
ceramicCapDarkBrown	SMT ceramic capacitor, 0805 or 1206 package, color light brown
tantalYellow	SMT tantalum capacitor, A - C package, color yellow
tantalBlack	SMT tantalum capacitor, A - C package, color black
LEDdiode	SMT LED diode in 0603, 0805 or 1206 package,
diode	Universal SMT rectifying diode, glass MINIMELF package

Tab. 5. Examined part types

The set of 25 images were obtained by the Canon and Basler camera for the quantitative comparison. The board was slightly moved before each capture to verify the systems resistance against changes in the scene. The white LED ilumination was used

with maximum input power as for the plain board inspection. The Golden board containing 26 containers were created. Counts of the various part types is following:

- resBlack - 5 packages
- ceramicCapWhite - 3 packages
- ceramicCapLightBrown - 5 packages
- ceramicCapDarkBrown - 5 packages
- tantalYellow - 2 packages
- tantalBlack - 2 packages
- LEDdiode - 2 packages
- diode - 2 packages

Because a single piece of testing board was available a certain process of obtaining the testing data had to be created to expand the required data-set and fulfill the range of defects. The process was following:

- Capture 5 shoots of defect-less board.
- Unsolder 5 different components:
  - resBlack, 2 pieces
  - ceramicCapWhite
  - ceramicCapLightBrown
  - ceramicCapDarkBrown

  capture 5 shoots.
- Return components on different places, capture 5 shoots.
- Destroy a correct position of 5 components, capture 5 shoots.
- Change polarity of 2 components, capture 5 shoots.

In the end of this process the data-set of 650 package containers was obtained, which contained:

- 100 containers with incorrect type of component.
- 25 containers with missing component.
- 50 containers with insufficient position.
- 10 containers containing a component with incorrect polarity.

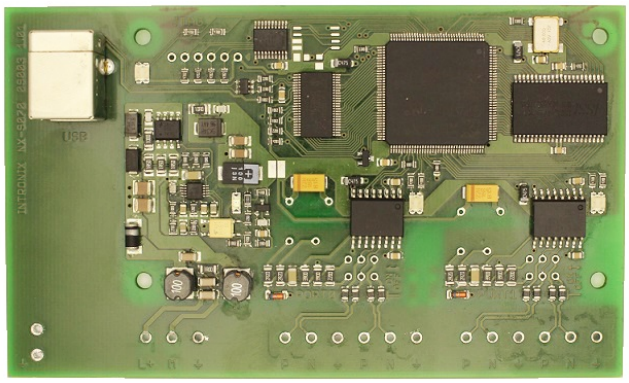


Fig. 28. Assembled board for inspection testing

The process was created in order to simulate the real ratio of defects, where from experience the substitution of a correct part type is the most common defect for hand-assembled circuit board, following with incorrect position caused mostly in flux reflow

process. During the process both Basler and Canon cameras were used to obtain dual data-set for their comparison.

Since there was no possibility of verifying whether the reported position deviation and rotation of the component reported by the inspection system corresponds to reality, only the ability of the system to designate a component as defective was considered. Defective component is the one which exceeds a predetermined limit value that was set as:

- Maximal position deviation: 0.4 mm
- Maximal rotation deviation: 0.5°

6. Results

6.1. Plain board testing

Because both types of defects in all ranges were presented simultaneously in the images, it is impossible to assign correct counts of "True Negative" errors to appropriate groups of thicknesses. For that reason only "True Positive" and "False Positive" data could be presented.

The success rate of the inspection process is presented, where the value is calculated as

$$Success\,rate = \frac{True\,positive}{Error\,appearance} \cdot 100 \quad [\%]. \tag{6.1}$$

6.1.1. Canon camera

Results of skip defect inspection testing and bridge inspection testing are shown in Table 6 and Table 7.

Thickness (mil)	Error appearance	TP	FP	Success rate (%)
20	60	59	1	98.33
10	60	58	2	96.67
5	60	13	47	21.67
1	60	0	60	0.00

Tab. 6. Plain board testing results, skip error, Canon camera

Thickness (mil)	Error appearance	TP	FP	Success rate (%)
20	60	60	0	100.00
10	60	55	5	91.67
5	60	6	54	10.00
1	60	0	60	0.00

Tab. 7. Plain board testing results, bridge error, Canon camera

6.1.2. Basler camera

Set of results follows similarly as for the previous type of camera.

Thickness (mil)	Error appearance	TP	FP	Success rate (%)
20	60	50	10	83.33
10	60	18	42	30.00
5	60	0	60	0.00
1	60	0	60	0.00

Tab. 8. Plain board testing results, skip error, Basler camera

Thickness (mil)	Error appearance	TP	FP	Success rate (%)
20	60	55	5	91.67
10	60	11	49	18.33
5	60	0	60	0.00
1	60	0	60	0.00

Tab. 9. Plain board testing results, bridge error, Basler camera

6.2. Assembled board testing

The part type identification and the correct polarity identification are measured by computing its success rate, where the the value is estimated by eq. 6.1. In the case of correct polarity testing the *Correct estimation* value is used instead of *True Positive*.

In the case of the missing component testing and the correct position testing the classification accuracy of inspection is presented. Value is calculated as

$$Classification\,accuracy = \left(1 - \frac{Error\,appearance - TP + FP}{dataset}\right) \cdot 100 \quad [\%] \quad (6.2)$$

Dataset is the count of containers obtained for testing and it is 650 containers for both inspection tests.

6.2.1. Canon camera

Part type identification testing

Part Type	Appearance	True Positive	Success rate (%)
resBlack	120	111	92.50
ceramicCapWhite	70	56	80.00
ceramicCapLightBrown	125	120	96.00
ceramicCapDarkBrown	120	118	98.33
tantalYellow	50	50	100.00
tantalBlack	45	45	100.00
LEDdiode	50	41	82.00
diode	45	44	97.78

Tab. 10. Part type error testing, Canon camera

From total count of 625 containers were 585 correctly identified, which results to the average success rate **93.60 %**

Missing component testing

From the set of 650 containers, 25 containers contained the empty pad, simulating the missing component.

Error	Error appearance	TP	FP	Classification accuracy (%)
Missing component	25	25	0	100.00

Tab. 11. Missing part error testing, Canon camera

Correct position testing

From the set of 650 containers, 50 containers contained the incorrectly positioned component.

Error	Error appearance	TP	FP	Classification accuracy (%)
Incorrect position	30	25	10	97.69
Incorrect rotation	20	20	0	100.00

Tab. 12. Correct position error testing, Canon camera

Correct polarity testing

From the set of 650 containers, 200 containers contained the component with polarity and 10 of these containers were incorrectly positioned.

	Appearance	Correct estimation	Success rate (%)
Polarity check	200	150	75.00

Tab. 13. Correct polarity error testing, Canon camera

6.2.2. Basler camera

Part type identification testing

Part Type	Appearance	True Positive	Success rate (%)
resBlack	120	105	87.50
ceramicCapWhite	70	48	68.57
ceramicCapLightBrown	125	115	92.00
ceramicCapDarkBrown	120	111	92.50
tantalYellow	50	50	100.00
tantalBlack	45	45	100.00
LEDdiode	50	35	70.00
diode	45	44	97.78

Tab. 14. Part type error testing, Basler camera

From total count of 625 containers were 553 correctly identified, which results to the average success rate **88.48 %**

Missing component testing

From the set of 650 containers, 25 containers contained the empty pad, simulating the missing component.

Error	Error appearance	TP	FP	Classification accuracy (%)
Missing component	25	15	20	60.00

Tab. 15. Missing part error testing, Basler camera

Correct position testing

From the set of 650 containers, 50 containers contained the incorrectly positioned component.

Error	Error appearance	TP	FP	Classification accuracy (%)
Incorrect position	30	25	75	87.69
Incorrect rotation	20	20	125	80.77

Tab. 16. Correct position error testing, Basler camera

Correct polarity testing

From the set of 650 containers, 50 containers contained the incorrectly positioned component.

	Appearance	Correct estimation	Success rate (%)
Polarity check	200	105	52.50

Tab. 17. Correct polarity error testing, Basler camera

7. Discussion

Before discussing the results, it is necessary to take into account that the testing was carried out under limited conditions and using limited test samples. In particular, this fact has had a significant effect on the results of most experiments, since the data set is highly affected by systematic defects resulting from repeating the scanning of the same test specimen. Nevertheless, it is possible to draw conclusions from the results obtained and obtain valuable conclusions about the functionality of the resulting system.

7.1. Plain board inspection

When testing a non-assembled printed circuit board, only the segmented image based method produced good results. The method based on simple subtraction did not provide results suitable for correct defect inspection. This was caused mainly because of the high noise in the output image. This shortcoming could not been limited by filtering, because the noise of the image exceeded the defect appearances. Therefore, only the methods used to subtract the segmented image are presented.

Let's first take a look at the results obtained with the Canon camera. The tracking error method quite well identified mistakes up to a width of 10 mil, 112 out of a total of 120 occurrences (Skip + Bridge). The method began to fail for 5 mils and 1 mils, when it identified only 19 errors from 120 occurrences. The fact the tables do not show is that the method showed a high incidence of "True Negative" results. This is not evaluated in the tables because it was not possible to assign the "True Negative" occurrences to an individual groups of errors, as described in the Testing chapter. The low inspection success rate for the 2 smallest sizes of observed errors can be explained by a significant use of morphological operations to suppress faults caused by imperfect aligning of images at subtraction. Unfortunately, this operation also led to the unwanted elimination of minor defects and also caused a reduction in the sensitivity of the inspection.

For the Basler camera, this sensitivity limitation was even more important, given its lower resolution. Although morphological operations for this camera have been modified, defect suppression has already occurred for errors of 10 mil width and for lower widths this method has ceased to identify errors. This behavior was assumed, as the camera resolution of 1 mil corresponded roughly to the width of one pixel of the image. The inspection results using the Basler camera are worse than the Canon camera in general. In the case of the biggest error (10 mil), the system detected 105 errors out of a total of 120. In other cases, it detected only 29 errors from 360 (Skip + Bridge).

Most of the problems were caused by the presence of soldermask on the printed circuit board. This fact required more filtering before segmentation and some of defect were being lost at this procedure too. Combining this with imperfect aligning and morphological operation, the resulted method was not effective for error under 10 mils for Canon camera and 20 mils for Basler camera.

7.2. Assembled inspection

Testing of correct part type identification did not detect a major differences between the Basler camera and the Canon camera. Because the inspection method is looking for the best match in color library, a individual part is always assigned. From the Table 14 can be seen the lower success rate values of white ceramic capacitor, black resistor and LED diode. This was mainly caused by interchange assignment of components. Because the images were taken with slightly different positions, the light conditions may varied more, with could suppress multiplicative errors as they did not occur in a such large scale. The tantalum capacitors show an interesting values for both used cameras where they have been flawlessly identified. This was caused either by its distinctive color (tantalYellow) or by its case of larger dimensions. Because the method reacts to the size of the package, these two types were identified in 100% of all cases.

Missing component testing showed significant difference between the cameras as shown in Tables 11 and 15. Though the Canon camera showed 100% classification accuracy in the un-assembled pad detection with flawless "False Positive" values, the Basler camera identification was not successful completely. The results of Basler camera testing showed high rate of assembled pads marked as free pads i.e. "False Positive" error. This was caused by incorrectly identifying a single component, in this case white ceramic capacitor, as the empty pad. The error was multiplied by the multiplicative data acquisition, which is obvious from the multiples of five in the results. This error influenced the results of type part identification, where in the Table 14 it is possible to observe significantly decreased success rate of the white ceramic capacitor in comparison to the other types of components.

In the testing of position identification result of "True Positive" are the same for both cameras, which signifies independence of the camera usage. Five unidentified cases of incorrect position may be caused by mistaking the component with an empty pad. The Basler camera results in Table 16 showed significant increase of "True Negative" errors, i.e. correctly soldered components marked as defective. This was mainly caused by lower resolution of camera where the low resolution noise processed with filtering badly influenced watershed segmentation. Resulting shape of segmented package was not rectangular because of side-effect blending with noise elements. Position and mainly rotation of fitted rectangle has been shifted. The multiples of five suggest the multiplicative error involvement again.

Although the polarity inspection test may appear to carry the least accurate results, when the success rate of correct estimation was 75% for the canon camera and only 60% for the Basler camera, detailed description is required. In the case of Canon camera usage, the method was unable to successfully identify the polarity mark of the LEDdiode package. Count of this package (for example visible in the Table 10) is exactly equal to the number of missing correct estimations. Same happened for the Basler camera, where the method was not able to identify the LEDdiodes polarity mark as well as the polarity strip on the black tantalum capacitor, due to a threshold error. This behavior caused a drastic worsening that was reflected in the results.

8. Conclusion

In this work, the development process of the Automatic Optical Inspection system for the small-scale production was presented. The motivation of investigation was presented and presentation of the existing solutions followed. The goal was to create an experimental system for inspecting the un-assembled and assembled printed circuit boards with usage of a freely available software tools and a hardware for image acquisition. This would make the technology of AOI more accessible.

Image capturing device was designed and constructed with relation to studied issues, linked to the image capturing. The final design allowed usage of two different cameras and it was equipped with internal lighting and light dispersion unit. For image capturing the two types of camera were used, industrial camera and commercial DSLR camera. Proposed color contrast enhancement controlled by light illumination was designed but not implemented.

Related theory was investigated to create suitable set of methods for image processing algorithms. To simplify the development as much as possible, the single general method was implemented for usage in plain board inspection as well as in assembled board inspection. The method was based on the color thresholding combined with watershed segmentation. This method core was used for most of the inspection parts with suitable modifications. For testing the inspection algorithm with real images other support procedures had to be implemented. This included an implementation of image preprocesing methods, process for Golden board creation and the suitable user interface design.

The results showed it is possible to use the industrial camera as well as the commercial camera for PCB inspection. Nevertheless, the results of plain board inspection proved that the system cannot be used for inspection of skip and bridge errors smaller than 10 mil in diameter. This resolution is insufficient, especially if we realize a usual dimensions of bridge or skip defects often do no reach even a tenth of the presented value. For this reason, the system was found to be unable to check plain PCBs. The system was more successful in the inspection of assembled printed circuit boards. Unfortunately, most of the obtained results contained multiplicative errors due to the imperfect data acquisition. The part type inspection results can be presented as the least affected results where the success rate for commercial camera was 93.6 % and the success rate for the industrial camera was 88.48 %. These results may seem sufficient for an experimental prototype system, but in a real production this system would not be usable. For example we can imagine a situation where the system with a success rate of 93.6 % incorrectly estimates at least one component on every board containing at least 15 parts.

However the obtained results does not approve the system directly for the real production it suggests it can be accomplished with further development. To improve the inspection performance, the future work has been suggested:

- Optimize the inspection algorithm for a single camera usage.
- Optimize the inspection algorithm for the assembled board inspection only.
- Implement the color contrast enhancement system.
- Fix the board position during capturing.

The scope of work was fulfilled in most points to the last point, when the quality testing of the system in the industrial application could not be achieved mainly due to processing the work abroad. The foreign research was personally very enriching, but the impact on the research itself was rather negative, considering the limited resources, cultural and communication distance, which have greatly complicated this practically focused research work and even some parts of the develop such as image capturing device had to be adapt to it. However, the experience gained from cross-border research has outweighed the negative aspects of work significantly.

Appendix A.

Photographs of the experimental setup

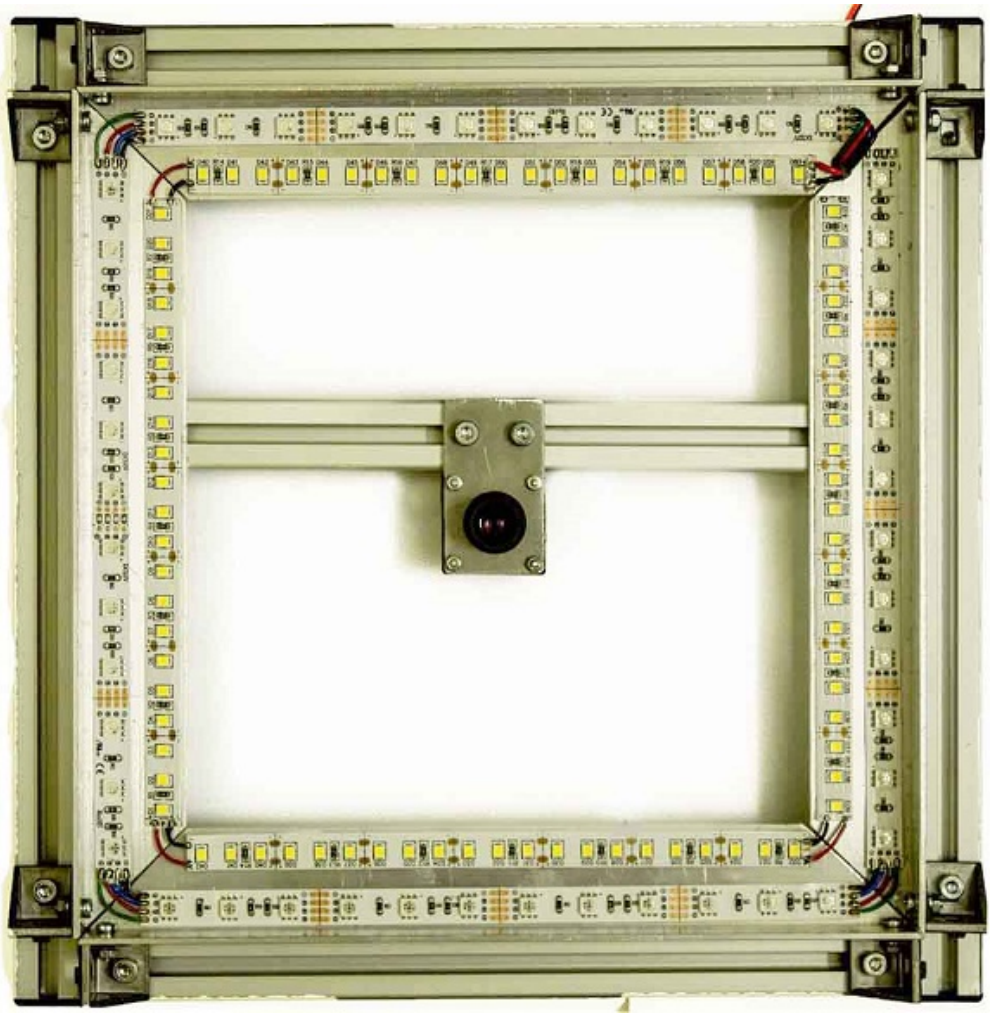


Fig. 29. Light source frame (White+RGB), Basler camera

Fig. 30. Image capturing device - white-box

Fig. 31. White-box, lights turned on

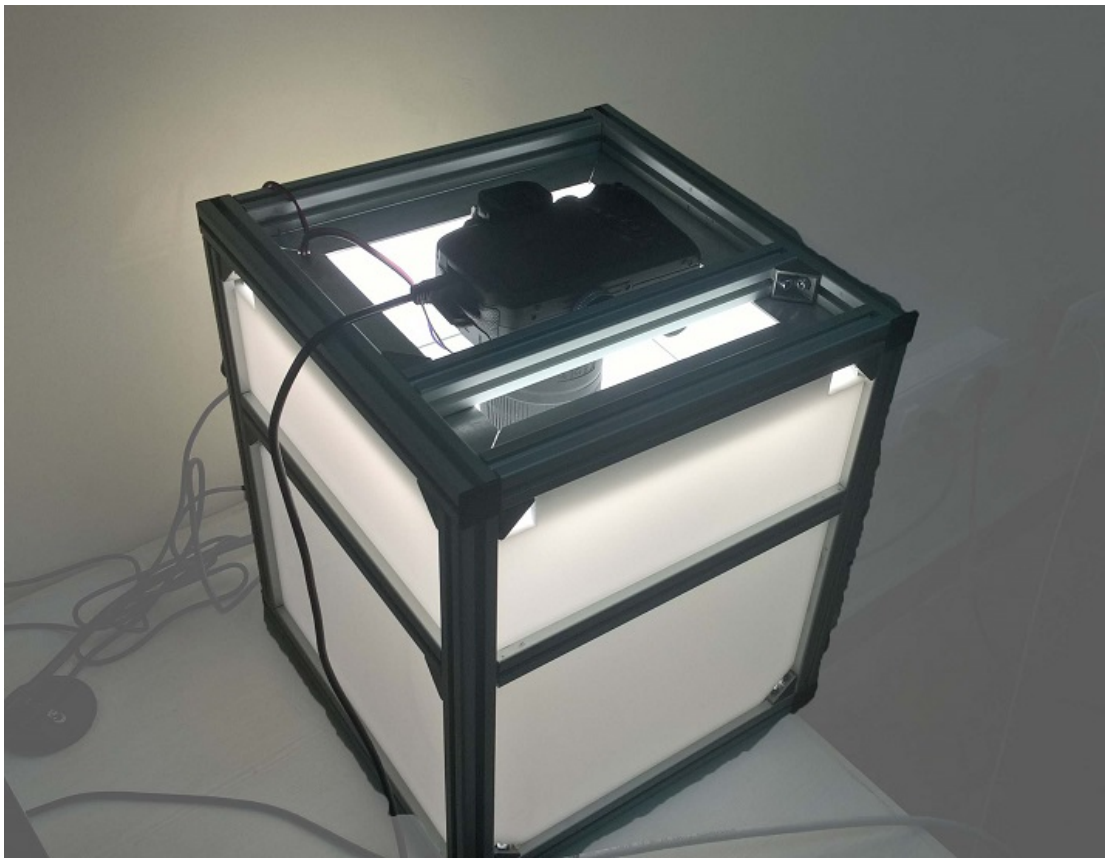


Fig. 32. White-box, mounted DSLR camera

Fig. 33. Experimental setup

Appendix B.

CD Content

```
/root
├── code ..................................Implemention of AOI systems
│   ├── Basler ..........................Pylon scripts for image acquisition
│   ├── Arduino ..................................Color enhancement system
│   ├── GUI ........................................User interface design
│   └── Inspection ..................................Inspection algorithms
├── Eagle ..............................................Test board design
├── LaTeX ..................................Source files of the document
├── Photo ..............................................Photo documentation
└── Thesis ..........................................Thesis in PDF format
```

Bibliography

[1] INTRONIX s.r.o. URL: http://www.intronix.cz (visited on 10/14/2017).

[2] Saki Corporation. URL: http://www.sakicorp.com/en (visited on 06/08/2017).

[3] modus high-tech electronics GmbH. URL: https://www.modus-hightech.de (visited on 06/08/2017).

[4] Orbotech Ltd. URL: https://www.orbotech.com (visited on 06/08/2017).

[5] Bungard Elektronik GmbH & Co.KG. URL: https://www.bungard.de/index.php/en/ (visited on 06/10/2017).

[6] Neelum Dave et al. "PCB Defect Detection Using Image Processing And Embedded System". In: *International Research Journal of Engineering and Technology (IRJET)* 03 (May 2016).

[7] Robert Baddeley. *Creating an Automated Optical Inspector for $50*. URL: http://bobbaddeley.com/2015/12/creating-an-automated-optical-inspector-for-50/ (visited on 06/28/2017).

[8] Clyde F. and Coombs Jr. *Printed Circuits Handbook*. Edition 2. McGraw-Hill Book Company, 1979. ISBN: 0-07-012608-9.

[9] PragoBoard s.r.o. URL: http://www.pragoboard.cz/en/o_firme (visited on 06/15/2017).

[10] Madhav Moganti et al. "Automatic PCB Inspection Algorithms: A Survey". In: *Computer Vision and Image Understanding* 63.2 (1996), pp. 287–313. ISSN: 1077-3142. DOI: https://doi.org/10.1006/cviu.1996.0020. URL: http://www.sciencedirect.com/science/article/pii/S107731429690020X.

[11] LLC. Epec. *Wawe Soldering Defects*. URL: http://www.epectec.com/pcb/wave-soldering-defects/ (visited on 06/17/2017).

[12] Gordon T. Uber and Kevin G. Harding. "Illumination and viewing methods for machine vision". In: *Proc.SPIE* 10258 (1991), pp. 10258 - 10258 - 14. URL: http://dx.doi.org/10.1117/12.57599.

[13] Inc Advanced illumination. *A Practical Guide to Machine Vision Lighting*. URL: http://www.advancedillumination.com/sites/all/themes/advill/images/resource/files/practicallightingv3.pdf (visited on 07/28/2017).

[14] Aven tools. *Illumination for Machine Vision Guide*. URL: http://www.aventools.com/wp-content/uploads/2016/02/mv-illumination-guide.pdf (visited on 07/28/2017).

[15] Milan Šonka, Václav Hlaváč, and Roger Boyle. *Image Processing, Analysis, and Machine Vision, International Student Edition*. Edition 3. Tompson Learning, part of the Thompson Corporation, 2008. ISBN: 0-495-24438-4.

[16] J. Canny. "A Computational Approach to Edge Detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6 (Nov. 1986), pp. 679–698. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1986.4767851.

[17] C. Harris and M. Stephens. "A Combined Corner and Edge Detector". In: *Proceedings of the 4th Alvey Vision Conference*. 1988, pp. 147–151.

[18] J. Sklansky. "Measuring concavity on a rectangular mosaic". In: *IEEE Transactions on Computers* 21(12) (1972), pp. 1355–1364.

[19] Ondřej Kunte. "Intelligent Heliport for Autonomous Helicopters". Czech Technical University in Prague, Faculty of Electrical Engineering, May 2015.

[20] Riverbank Computing Ltd. URL: `https://riverbankcomputing.com/news` (visited on 11/10/2017).

[21] Basler AG. *Pylon software overview*. URL: `https://www.baslerweb.com/en/products/software/` (visited on 11/28/2017).

[22] OpenCV - Open Source Computer Vision. URL: `http://opencv.org` (visited on 07/18/2017).

[23] FastLED 3.1 library. URL: `https://github.com/FastLED/FastLED` (visited on 06/11/2017).

[24] David Kay. *Simple white-balance algorithm*. URL: `https://gist.github.com/DavidYKay/9dad6c4ab0d8d7dbf3dc` (visited on 09/12/2017).

[25] Basler AG. *daA2500-14uc - Basler dart camera*. URL: `https://www.baslerweb.com/en/products/cameras/area-scan-cameras/dart/daa2500-14uc/` (visited on 11/12/2017).

[26] Canon Inc. *Canon EOS 600D*. URL: `https://www.canon.cz/for_home/product_finder/cameras/digital_slr/eos_600d/` (visited on 11/13/2017).

[27] s.r.o. ALUTEC K & K. *Aluminium structural system*. URL: `https://www.aluteckk-uk.com/aluminium-structural-system` (visited on 07/24/2017).