

High-Resolution Inline Video-AOI for Printed Circuit Assemblies

Benjamin Guthier, Stephan Kopf, Wolfgang Effelsberg Praktische Informatik IV
University of Mannheim, Germany
{guthier, kopf, effelsberg}@informatik.uni-mannheim.de

ABSTRACT

We enhance an existing in-circuit, inline tester for printed circuit assemblies (PCA) by video-based automatic optical inspection (Video-AOI). Our definition of *video* is that we continuously capture images of a moving PCA, such that each PCA component is contained in multiple images, taken under varying viewing conditions like angle, time, camera settings or lighting. This can then be exploited for an efficient detection of faults. The first part of our paper focuses on the parameters of such a Video-AOI system and shows how they can be determined. In the second part, we introduce techniques to capture and preprocess a video of a PCA, so that it can be used for inspection.

Keywords: Video-based AOI, circuit boards, fiducial marks, camera calibration, stitching

1. INTRODUCTION

Flaws in the process of populating *printed circuit boards* with electronic components often lead to malfunctioning of the resulting *printed circuit assembly* (PCA). To guarantee that the right components were placed at the correct positions and work as expected, PCAs must be tested subsequent to the assembly. Various methods for testing exist. The most straightforward approach is to do functional testing. This means running a sequence of tests on the assembly as a whole and monitoring the results. However, there may be faults that are more subtle and not covered by the functional tests. It is therefore common to perform in-circuit tests to gain a more fine-grained understanding of the components' functionality. They allow for checks at any level of granularity varying from a single component to the entire assembly. In-circuit testing is done by connecting electrical probes to a PCA that allow for measuring conductivity, resistance, capacity and other electrical properties.

The two major techniques used are the so-called *bed of nails testers* and the *flying probe testers*. While the former uses a static arrangement of connectors that is pressed against the PCA, the latter uses a typically much smaller set of movable probes to connect. In practice, creating a specialized bed of nails adapter for one particular type of board is expensive. Only for high volumes, the initial cost is redeemed by the high testing speed achieved through extensive parallelization. On the other hand, the flexibility of flying probe testers takes effect for smaller volumes and prototyping. These testers can be easily configured to test new types of boards as needed. Their main drawback is the lower testing speed due to the smaller number of probes and the time it takes to reposition them between the individual checks.

An alternative to functional and in-circuit testing is *automatic optical inspection* (AOI).¹⁻⁴ In this approach, line scan cameras or area scan cameras with strong magnification are used to capture high-resolution images of the PCA. The digital images are then processed by machine vision algorithms to search for faults. This technique reveals faults that are hard or impossible to detect by the other two approaches. Examples of such faults are bent pins on a connector, badly aligned components, bad solder joints or faults in regions that are unreachable by probes. Another advantage of optical inspection is contact-less testing. Assemblies tested with in-circuit tests often show traces of the pointy tips of the connectors on the soldering pads. In-circuit testing damages the tested PCA to some extent which can be avoided by AOI.

The approach we chose in this paper is the enhancement of a flying probe tester by AOI. This combination achieves a high coverage of fault classes as it joins the sets of detectable faults of flying probe testers and AOI. Additionally, increased confidence in the test results can be gained by checking crucial elements redundantly. When testing speed is of major concern, individual checks can be completely shifted to the AOI and performed in parallel to the electrical tests, speeding up the entire process significantly. Depending on the particular application, an optimal weighting between electronic testing and optical inspection can be determined.

As a novel idea, we use video sequences instead of still images in this scenario. We call our technique *Video-AOI*. Our definition of *video* is that we continuously capture images of a moving PCA, resulting in a large number of images, such that each electrical component on the PCA is contained in multiple images, taken under varying viewing conditions.

Having multiple images of each component can then be exploited for inspection. If the images are captured at known camera positions, the height of a component can be determined using *stereo vision* and *structure from motion* approaches, facilitating the detection of missing components. Another way of taking advantage of Video-AOI is by varying the lighting conditions between the individual shots, controlling the casting of shadows and light reflections. Similarly, the shutter speed of the camera can be varied to capture images at varying exposure that each emphasize a different dynamic range of the object under consideration.

Applying Video-AOI inside a flying probe tester gives rise to new difficulties to overcome. Optical inspection of PCAs requires an image capturing system with a very high resolution. The image may need to span up to 40cm of a board while still resolving details with a size of $100\mu\text{m}$. Such a resolution can only be achieved by line scan cameras, but only area scan cameras are capable of capturing videos. It is therefore necessary to employ several area scan cameras at once, leading to increased costs of the system, high data rates to be processed and shortness of space inside the narrow-built tester. The narrowness of the tester is also a challenge for the lighting used. Additionally, due to the moving probes, it is not possible to capture a video of the PCA during the electronic tests. In order to not add overhead for capturing to the overall duration of the test, the video must be captured inline while the board is transported into the tester via a conveyor belt. In an industrial environment, the speed of the conveyor must be considered a given constant. Only by using short shutter times and cameras capable of capturing videos at high frame rates can the capturing system keep up with the conveyor speed. And lastly, the capturing system will produce a large number of images taken under varying lighting conditions and from differing angles. Before they can be used for inspection, they need to be aligned with respect to each other. We refer to the process of computing each image's place in the "big picture" as *stitching*.⁵ Having a fully stitched set of captured images of a PCA, the Video-AOI system is capable of determining the set of images containing a component to inspect and its exact pixel position in the images.

It should be noted that calibrating the cameras, capturing image sequences and computing the offsets between the images so they can be used for inspection is the main focus of this paper. Going into the details of a particular inspection algorithm is not our goal.

The remainder of this paper is structured as follows. In Section 2 we describe the prototype of such a system and analyze its parameters and their relationship to each other. At the end of the section, we will give the parameters of our prototype as an example. Section 3 focuses on the capturing and preprocessing of image sequences with our system and explains the steps necessary to capture videos that can be used for AOI. They include camera calibration, coordinate system transformations and four forms of image stitching. The Section ends with an example application implemented by us that allows the capturing of *high dynamic range videos* (HDR videos) with our system. Section 4 contains experimental results.

2. PARAMETERS OF THE SYSTEM

Before building a Video-AOI system to be used inside a flying probe tester, one must first examine the system's parameters, understand their interrelation and ultimately decide upon the values to be used. We begin this section with an overview of the system as a whole and its relevant components. We then analyze the parameters by grouping them into three categories: Constraints by the tester (Section 2.2), parameters determined by the application (2.3) and the freely adjustable parameters (2.4). The hardware constraints and the application requirements are the starting point for the choice of adjustable parameters. The currently available camera hardware, optics and bus technology then determine how well the requirements can be met. At the end of this section, we list the choice of parameters we made when building our prototype.

2.1 System Overview

Figure 1 depicts the arrangement of conveyor, camera array and flying probe tester in our Video-AOI prototype. The system is built as an inline facility that can be directly connected to the production line. The assembled board is transported into the flying probe tester on a conveyor band. On its way in, it passes an array of area scan cameras. A photo sensor below the conveyor detects the approaching board and starts a clock generator to trigger the cameras. The cameras then synchronously capture a sequence of images until the board has completely passed the array. Finally the captured images are processed while the PCA is tested electronically inside the flying probe tester.

For the remainder of this paper, we will refer to the board axis perpendicular to the board motion and parallel to the camera array as the horizontal axis. The direction of board movement defines the vertical axis respectively.

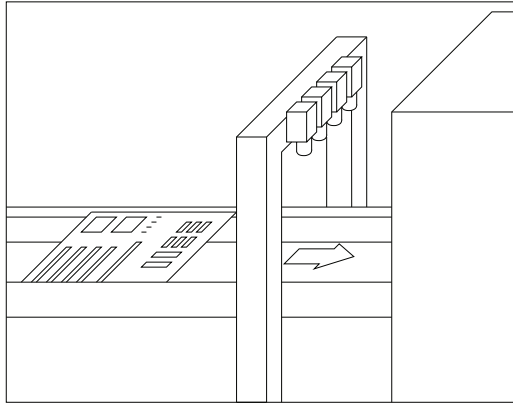


Figure 1. Simplified representation of the flying probe tester and its preceding Video-AOI unit. The camera array captures images of the PCA as it is transported into the tester.

2.2 Hardware Parameters

The first category of parameters of the Video-AOI system for flying probe testers consists of the hardware parameters. Their values are usually given by the production environment and we assume them to be fixed in our paper. The following parameters are relevant:

Maximum PCA width. The size of the biggest PCA to be inspected is determined by the width of the conveyor. Its value has an impact on the number of cameras used, their resolution and the required optics. For our further considerations, we will assume that a PCA to be inspected has the maximum width which we denote by w .

Minimum/maximum vertical camera position. The camera array may not always be as freely positionable as shown in Figure 1. It may be tightly integrated into the production line or the tester itself. In these cases, constraints regarding the distance to the conveyor at which the camera array can be installed apply. They limit the achievable depth of field and determine the optics to be used for the cameras. The lower and upper bound for the distance of the camera to the PCA will be denoted by d_{min} and d_{max} respectively.

Conveyor band speed. Making changes to the speed at which a PCA is transported into the tester is difficult in an existing production line. We therefore assume it to be constant throughout this paper and assign it the letter v . As a result, the cameras' frame rate, shutter speed and lighting must be chosen carefully to allow the capturing of motion blur-free images under the given board movement speed.

2.3 Application Parameters

The second parameter set consists of those that are determined by the particular inspection application. Variations may be due to the type of boards and components to be inspected as well as the inspection task to be performed. The reconstruction of 3D data through stereo vision for example requires the cameras' fields of view to overlap largely. Application parameters are in general more flexible than hardware parameters. The parameters to be considered are:

Color. Many industrial cameras are available in the two variants color and monochrome. A common trick is to apply a color filter array * to an image sensor to make the sensor cells color sensitive. While color cameras using this technique are similar in price to the corresponding b/w models, they effectively trade off resolution for the ability to detect colors. Color cameras should thus only be used if color information is relevant to the application at hand.

Depth of field. The height of the highest inspectable component on a PCA determines the depth of field required for the Video-AOI system. The achievable depth of field mainly depends on the camera's focal length, lens aperture and the distance to the PCA. It is measured in units of length and we denote it by d_f .

*often: Bayer filter

Image brightness. Assuming that the lighting illuminating a scene was chosen to be as bright as possible, the brightness of the captured images only depends on the shutter speed and lens aperture used. Increasing the exposure time by adjusting the shutter speed leads to brighter images but also to motion blur when capturing a fast moving board. Its upper bound is therefore determined by the conveyor speed. If the desired brightness cannot be achieved by adjusting the shutter speed alone, depth of field must be traded off for image brightness by widening the lens aperture.

Spatial resolution. The size of the smallest structure to be inspected through Video-AOI is a parameter determined by the application. If we assume that a fixed number of pixels is required to resolve a structure, dividing this number of pixels by the size of the smallest structure directly leads to the required spatial resolution of the optical system. Hence the spatial resolution r is the number of pixels required per unit of dimension of the PCA.

Image multiplicity. We refer to the number of captured images in which a PCA component is contained as image multiplicity. Multiplicity can be generated by capturing images that overlap horizontally or vertically. Horizontal overlap is the result of overlapping fields of view of the cameras in the array. In the vertical direction, increasing the cameras' frame rate will increase multiplicity. The horizontal and vertical multiplicity factor m_h and m_v must be chosen according to how the captured images are to be processed. Capturing images of a component under varying lighting conditions for example can only be achieved through vertical overlap, since all cameras are triggered synchronously.

2.4 Adjustable Parameters

The final category of parameters are those of the optical system that must be chosen to meet the hardware and application requirements determined before. In practice, not every combination of parameters is possible and restrictions of the available camera hardware must be considered. It is then necessary to review the hardware and application parameters and to relax the constraints until they can be met. In this section we give guidelines and formulae for their choice.

Number of cameras and resolution. When deciding on the camera type and the number of cameras to be used for the Video-AOI system, the important values to consider are the maximum PCA width w , the spatial resolution r and the horizontal image multiplicity m_h . The horizontal camera resolution, i.e. the number of cells per row on the camera's sensor, and the number of cameras used must be big enough to achieve the desired spatial resolution over the entire width of a PCA at the desired multiplicity. Mathematically, this can be expressed as follows: Let n be the number of cameras and p_h the number of camera pixels per row. n and p_h must be chosen so that $wrm_h \leq np_h$. A suitable compromise between number of cameras and resolution is one that minimizes the overall cost. Typical values for p_h range from 500 to 2000 for current industrial cameras.

Focal length. Once the type and number of cameras are determined, the cameras need to be configured with suitable optics. We found that for optical inspection, it is desirable to employ telephoto lenses to keep the distortion due to short focal lengths to a minimum. We therefore position the camera array at d_{max} within reasonable bounds. With the width of the desired field of view of a camera being p_h/r and knowing the width of the camera's sensor w_s , the required focal length f can be approximated by

$$f = \frac{d_{max}}{1 + p_h/(rw_s)}. \quad (1)$$

Lens aperture. If the chosen lens has an adjustable f-number N , it can be set to achieve the desired depth of field. Giving objective directives for setting the camera's f-number is difficult, since the definition of depth of field depends on the maximum size of the *acceptable circle of confusion* c , which is strongly subjective. A suitable value for c must be chosen for the given application (see Section 2.6 for our choice). If the camera's focus is set to the surface of the PCA, the achieved depth of field d_f can be roughly estimated by

$$d_f = d_{max} - \frac{d_{max}f^2}{f^2 + Nc(d_{max} - f)}. \quad (2)$$

Given the desired depth of field, and values for the other parameters from the previous considerations, this equation can be used to estimate the required f-number setting:

$$N = \frac{d_f f^2}{c(d_{max} - f)(d_{max} - d_f)}. \quad (3)$$

Shutter speed. The shutter speed s – also called *exposure time* – is the duration for which the camera’s sensor is exposed to the light of the scene. It is measured in microseconds. As stated before, we assume a constant conveyor band speed v throughout this paper. The longest usable shutter speed is therefore limited to avoid motion blur. In other words, the distance in pixels a point on the PCA moves during one exposure period must be lower than a threshold τ

$$vrs \leq \tau \quad (4)$$

leading to an upper bound for the shutter speed of

$$s \leq \frac{\tau}{vr}. \quad (5)$$

Frame rate. The last parameter to be chosen is the frame rate t of the cameras in the array. It is a crucial limiting factor of the attainable capture speed. Most industrial cameras have an adjustable frame rate, so the question is: What is the lowest frame rate sufficient to capture images of the moving PCA with the desired vertical multiplicity m_v ? The cameras must then be chosen to support at least this rate. More precisely, this requirement can be formulated as

$$t \geq \frac{vr}{p_v} m_v \quad (6)$$

where vr is the conveyor band speed in pixels per time unit and p_v the frame height in pixels. It should be noted, that in theory the time between two frames cannot be shorter than one exposure period, so the frame rate must also be less than $1/s$. In practice though, exposure periods are very short and this restriction does not apply.

2.5 Further Considerations

Capturing videos with the VAOI system described above produces high data rates and large amounts of data. For example, capturing 1392×1040 pixels at 15 frames per second results in a data rate of roughly 175 MBit/s . By employing several of these cameras, the bandwidth quickly exceeds the limit of a single FireWire bus. Special care must be taken when choosing the image processing hardware to cope with the occurring data.

As a result of basing the design of the VAOI system and its parameters on the constant conveyor band speed, the total time to capture a video of a PCA only depends on the speed of the conveyor. Capturing ends once the PCA has passed the camera array completely. Though before the captured video can be used for inspection, the individual frames need to be aligned with respect to each other in a process called “Stitching”. Time taken to perform this step is evaluated in Section 4.

Achieving proper lighting for the VAOI system is a challenge. Little advice on its choice can be given here as it strongly depends on the availability of space, mounting and power inside the tester. Generally speaking, the lighting should be as bright as possible to attain more freedom in choosing other parameters like shutter speed and f-number. We chose to use an array of LED light sources that are triggered synchronously to the cameras. By using the LEDs in a pulsed mode rather than operating them continuously, more brightness can be achieved by the same LEDs without damaging them. The pulse only needs to be as long as the exposure time, giving the LEDs time to cool down while the CCD sensors are read out.

2.6 Our Choice of Parameters

Our VAOI prototype is built as box separate from the flying probe tester and is preceding the tester in the conveyor line. The box is opaque to allow for constant lighting conditions, independent of the surrounding light. Upon entering the VAOI prototype, the PCA triggers a light barrier that will start the capturing process. The same light barrier will then signalize the end of the capturing process as the PCA exits.

In our scenario, the hardware requirements are as follows: The widest PCA to be inspected by Video-AOI has a width of $w = 400 \text{ mm}$. PCAs are transported on the conveyor at a speed of approximately $v = 100 \text{ mm/s}$. Our Video-AOI prototype allows a maximum height of the camera array above the surface of the PCA of $d_{max} = 500 \text{ mm}$.

The upper limit for the height of an inspectable component on a PCA – and thus the required depth of field – was set to $d_f = 10 \text{ mm}$. Our application requires a resolution of $r = 40$ pixels per millimeter. The horizontal image multiplicity was set to $m_h = 1.05$ for roughly 5% of overlap as tolerance. In the vertical direction, we capture with a multiplicity of $m_v = 2.1$ to get two shots of each component with some tolerance that can be used for stitching. The cameras we use are monochrome 1394b FireWire cameras with a resolution of 1392×1040 ($p_h \times p_v$) pixels and $1/2''$ sensors.

Using the formulae described in Section 2.4, we get the following values for the adjustable parameters: We need at least $n = 12$ cameras. With a sensor width of $w_s = 6.4mm$, Equation 1 gives a focal length of $f = 77.7mm$. For reasons of availability, we used a lens with a fixed focal length of $75mm$ and moved the camera array to a distance of $d = 483mm$ from the PCA to achieve the desired resolution. In Equation 5, we require that the PCA moves at most one pixel during one exposure period which results in an upper limit for the allowed shutter speed of $s \leq 250\mu s$. The shutter speed can be varied under this constraint to capture High Dynamic Range videos using varying exposure settings for example. And finally the cameras must be capable of capturing images at a rate of at least $t = 8$ frames per second.

The total data rate produced by the twelve cameras in our setup is $1111.8MBit/s$ over a duration of up to 5 seconds. This data rate can be handled by two 1394b interface cards and the amount of data produced conveniently fits into the main memory of a modern PC.

3. CAPTURING VIDEOS FOR INSPECTION

This section describes in detail how high-resolution videos of PCAs can be captured and preprocessed in order to be used for video-based automatic optical inspection. We start with an overview of the coordinate systems involved and their relationships in terms of mathematical transformations in Section 3.1.

In a first offline step, the cameras in the array need to be calibrated with respect to each other and the conveyor band. For this step, we use a calibration board tailored to our camera array. The board and the calibration process are described in Section 3.2.

In the online phase of the VAOI system, a PCA to be tested is transported into the system where it triggers a light barrier and starts the capturing process. Periodic trigger signals are sent to all cameras in the array and all light sources until the PCA exits the VAOI unit. In each cycle, a row consisting of n images is captured by the n cameras. We denote the number of total rows captured by m . It is important that all cameras are triggered at exactly the same time so the relative positions of the images in one row correspond to those determined in the calibration process. The video consisting of $m * n$ frames is first captured into the main memory of the PC the cameras are connected to. Preprocessing of the video starts once capturing is completed. The preprocessing mainly consists of estimating the transformations between images of the video. After this, one last transformation needs to be computed that relates the “big picture” with the CAD description of the PCA. We refer to the entire preprocessing step as *Stitching*. It is described in full detail in Section 3.3.

Once capturing and preprocessing is done, the video can be used for inspection. How this can be done is beyond the scope of this paper. We end this Section with an example of how our VAOI system can be used to capture high dynamic range (HDR) videos of PCAs.

3.1 Coordinate Systems and Transformations

A multitude of two-dimensional coordinate systems are involved in capturing videos of a moving PCA. Each type of PCA has its own coordinate system called the *CAD coordinate system*. It is used to describe positions and sizes of components placed on the PCA, which is important for AOI. Its unit is usually a physical unit of length and its axes and origin can be arbitrarily placed on the PCA.

Every camera of the array has a pixel coordinate system with the origin residing in the top left pixel of the camera image and the positive horizontal and vertical axes pointing right and down respectively. We refer to them as *camera coordinate systems*.

For the sake of understandability, we imagine the PCA to be standing still on the conveyor and the camera array moving once over the entire PCA while capturing $m * n$ images. It then becomes clear that each image has its own *image coordinate system*. The coordinate systems of the first row of images are identical to the camera coordinate systems. The coordinate systems of each subsequent row of captured images are then related to the camera coordinate systems by an Euclidean transformation.

We introduce an additional virtual 2D coordinate system between CAD and the images which we call the *tester coordinate system*. It lies in the plane defined by the conveyor band and is established during the camera calibration process. It serves as an intermediate coordinate system to simplify the stitching.

Using homogeneous coordinates, a point in any of these coordinate systems is a tuple with three components. Let \bar{x} be a point on a PCA, specified in CAD coordinates. It is then represented in tester coordinates as *one* 3-tuple x' , since we

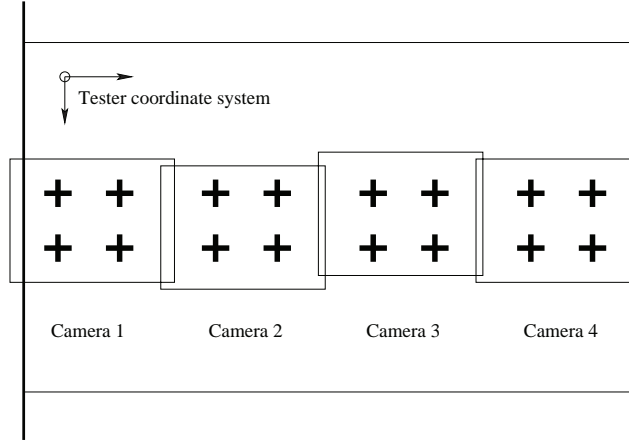


Figure 2. Calibration board placed on the conveyor belt underneath the camera array. The cross-shaped fiducial marks are printed onto the board so that each camera can see at least four marks. Their coordinates are specified in an arbitrary coordinate system which later constitutes the tester coordinate system.

imagine the board to be standing still. For i in $\{1, \dots, n\}$ and j in $\{1, \dots, m\}$ the 3-tuple $x_{i,j}$ represents the same point in the coordinate system of image $I_{i,j}$, where $I_{i,j}$ is the j th image captured by camera i .

Transformations between the various coordinate systems can be expressed by 3×3 matrices. In the whole process of capturing and stitching $m * n$ images, there are $m * n + n + 1$ matrices involved. The *image matrix* $M_{i,j}$ transforms a point from tester coordinates into the coordinate system of image $I_{i,j}$:

$$M_{i,j}x' = x_{i,j}, \quad i \in \{1, \dots, n\}, \quad j \in \{1, \dots, m\}. \quad (7)$$

Once during the calibration process, the *camera matrices* N_i are established, that transform tester coordinates to the coordinate system of camera i (see Section 3.2). The first row of image matrices are set to the camera matrices:

$$M_{i,1} := N_i, \quad i \in \{1, \dots, n\}. \quad (8)$$

All further image matrices must be estimated in the stitching process as described in Section 3.3. One more matrix, the *CAD matrix* C , transforms CAD coordinates into tester coordinates:

$$C\bar{x} = x'. \quad (9)$$

It is estimated in the final step of stitching (see Section 3.3.4).

The matrices $M_{i,j}$ and N_i represent projective transformations. The $M_{i,j}$ are interrelated by Euclidean transformations, which becomes clear when imagining the cameras to be moved over the fixed PCA. C is a similarity transformation, consisting of an Euclidean transformation with scaling. Its Euclidean part can be explained by the PCA residing fixed in the tester, translated and rotated relative to the tester coordinate system. The scaling is due to the potentially differing units of length used in the CAD and tester coordinates. Note that all matrices are invertible and can also be used to transform coordinates in the opposite direction.

The results of the entire capturing and stitching process described throughout this paper are thus $m * n$ images $I_{i,j}$ with corresponding image matrices $M_{i,j}$ and a CAD matrix C . Using the matrices $M_{i,j}$, a single stitched high-resolution image of the PCA can easily be obtained. However for AOI, this step is unnecessary.

3.2 Camera Calibration

Before any videos can be captured with the VAOI system, the array of cameras must be calibrated once. This means to estimate the camera matrices N_i , that relate the pixels of all cameras to positions in the common tester coordinate system. The matrices N_i are saved and used in stitching later.

For calibration, we place a *calibration board* on the conveyor band, which is similar to a PCA in size. A number of fiducial marks is printed onto the calibration board and it is positioned in a way that allows each camera to see at least four

marks. An example calibration board with cross-shaped marks and the fields of view of four cameras is shown in Figure 2. The positions $x_{i,1}$ to $x_{i,4}$ of the four fiducial marks in pixel coordinates of camera i can be accurately detected by template matching, thresholding and computation of centers of gravity.

Let $x'_{i,1}$ to $x'_{i,4}$ be the coordinates of the marks on the calibration board in a fixed coordinate system with arbitrary origin and scale. These coordinates must be known prior to calibration. The arbitrary coordinate system constitutes the intermediate tester coordinate system.

For each camera i , the eight parameters of the camera matrix N_i are calculated by solving the system of equations

$$N_i x'_{i,k} = x_{i,k}, \quad k \in \{1, \dots, 4\}. \quad (10)$$

3.3 Stitching

As mentioned before, throughout this paper, we imagine the PCA to be standing still on the conveyor while the camera array is moved for scanning. From a mathematical point of view, this scenario is identical to a moving PCA and stationary cameras, so the choice between the two philosophies is arbitrary. We believe that our view helps the comprehensibility.

The problem of stitching can be formulated as the estimation of the image matrices $M_{i,j}$ and the CAD matrix C . The matrices $M_{i,j}$ relate the pixel coordinates of image $I_{i,j}$ to tester coordinates. For the first row of images, the image matrices are identical to the calibration matrices. Every additional row of image matrices can then be obtained by multiplying the matrices of the previous row by an Euclidean matrix which is estimated from two subsequent images from the same camera. Three approaches to the problem of estimating $M_{i,j}$ will be introduced in the following sections.

In the end, only *one* mapping C between tester and CAD coordinates needs to be estimated. This will be the focus of Section 3.3.4.

3.3.1 Fiducial-based Stitching

We now show how the image matrices $M_{i,j}$ can be estimated by setting $M_{i,1} := N_i$ for i in $\{1, \dots, n\}$ and showing a method for estimating $M_{i,j+1}$ from already known $M_{i,j}$ for j in $\{1, \dots, m-1\}$.

Here, we assume that in each pair of subsequent image rows $(I_{1,j}, \dots, I_{n,j})$ and $(I_{1,j+1}, \dots, I_{n,j+1})$ there are two fiducial marks that are visible in any two images of row j and the corresponding two images of row $j+1$. Let k be the camera index so that the images $I_{k,j}$ and $I_{k,j+1}$ contain the first mark, and let l be the index so that $I_{l,j}$ and $I_{l,j+1}$ contain the second. The pixel coordinates $x_{k,j}, x_{k,j+1}, x_{l,j}, x_{l,j+1}$ of the marks in the four images are determined through template matching just like during calibration. We must now estimate the matrices $M_{k,j+1}$ so that

$$M_{k,j+1}^{-1} x_{k,j+1} = M_{k,j}^{-1} x_{k,j}. \quad (11)$$

Index l likewise. Equation 11 means that the pixel coordinates of a mark in two images must both map to the same position in the tester coordinate system.

We accomplish this by first transforming $x_{k,j+1}$ and $x_{k,j}$ by the known matrix $M_{k,j}^{-1}$ and transforming $x_{l,j+1}$ and $x_{l,j}$ by the known matrix $M_{l,j}^{-1}$ into tester coordinates. We then estimate an Euclidean transformation T that maps the transformed coordinates onto each other:

$$TM_{k,j}^{-1} x_{k,j+1} = M_{k,j}^{-1} x_{k,j} \quad \wedge \quad TM_{l,j}^{-1} x_{l,j+1} = M_{l,j}^{-1} x_{l,j}. \quad (12)$$

This system of four equations has three variables and can be solved by non-linear least-squares fitting. We now set $M_{k,j+1} := M_{k,j} T^{-1}$, so that $M_{k,j+1}^{-1} = TM_{k,j}^{-1}$. It follows, that

$$M_{k,j+1}^{-1} x_{k,j+1} = TM_{k,j}^{-1} x_{k,j+1} = M_{k,j}^{-1} x_{k,j}. \quad (13)$$

It can be seen that $M_{k,j+1}$ fulfils Equation 11 as required. Index l likewise. Using the same matrix T , all other $M_{i,j+1}$ are now calculated as: $M_{i,j+1} := M_{i,j} T^{-1}$

Adding fiducial marks that can be used for fiducial-based stitching to a PCA can be done by putting the PCA into a fixture that already contains the required fiducials. Mounting PCAs in such a way has to be done manually, which can be an unacceptable drawback in some production lines. The advantage of fiducial-based stitching is clearly its speed. The indices k and l of the cameras that can see the fiducial marks as well as their approximate position in the camera's field of view are usually known. In a relatively small search area properly printed marks on a fixture can be detected quickly and robustly. Estimating the $M_{i,j}$ is even faster, as is shown in Section 4.

3.3.2 Feature-based Stitching

The feature-based stitching is similar to the fiducial-based version. Again, we set $M_{i,1} := N_i$ for i in $\{1, \dots, n\}$ and show a method for estimating $M_{i,j+1}$ from already known $M_{i,j}$ for j in $\{1, \dots, m-1\}$.

For the feature-based stitching, we relax the requirement of having fiducial marks and only assume that detectable features like corners and dots are present in the images. We use Harris feature points,⁶ SIFT^{7,8} and RANSAC⁹ for feature detection and matching. For each pair of subsequent images $I_{i,j}$ and $I_{i,j+1}$, $\forall i$, we obtain a list of coordinate pairs $(x_{i,j}^{(k)}, x_{i,j+1}^{(k)})$. For each k , this pair represents the coordinates of a feature that has been detected in two subsequent images. We refer to this pair as a *feature match*. Similar to Equation 11, $M_{i,j+1}$ must be estimated, such that both coordinates of a match are transformed to the same tester coordinates:

$$M_{i,j+1}^{-1}x_{i,j+1}^{(k)} = M_{i,j}^{-1}x_{i,j}^{(k)}, \forall i, k. \quad (14)$$

In practice, only a small number of feature matches in only two distant images must be considered. The more features and images are considered, the higher the accuracy.

Again, we transform both coordinates of a match $(x_{i,j}^{(k)}, x_{i,j+1}^{(k)})$ by the same known matrix $M_{i,j}^{-1}$ and estimate *one* Euclidean transformation T , that approximates

$$TM_{i,j}^{-1}x_{i,j+1}^{(k)} \approx M_{i,j}^{-1}x_{i,j}^{(k)}, \forall i, k. \quad (15)$$

This system of three variables and two equations per feature match can be approximated using non-linear least-squares fitting. Again, by setting $M_{i,j+1} := M_{i,j}T^{-1}$, we get

$$M_{i,j+1}^{-1}x_{i,j+1} = TM_{i,j}^{-1}x_{i,j+1} \approx M_{i,j}^{-1}x_{i,j} \quad (16)$$

and approximate Equation 14. The quality of this approximation is evaluated in Section 4.

Note that the detection and matching of features is less accurate and robust than the detection of fiducial marks. We therefore use a higher number of matches (in the magnitude of tens or hundreds) for an average, and the correspondence in Equation 14 cannot be achieved perfectly for each of the matches. In order to be able to detect enough feature matches, sufficiently structured PCAs and a higher vertical multiplicity m_v than for fiducial-based stitching are required. The latter leads to a higher frame rate requirement to retain the same conveyor velocity as can be seen in Equation 6. In addition, the process of detecting, matching and selecting suitable features is computationally expensive. As an advantage, the overall accuracy is higher than for fiducial-based stitching due to averaging over all the feature matches considered. Another major advantage is the independence from fiducial marks on the PCA, allowing feature-based stitching to be used without fixture and inline.

3.3.3 Reference-based Stitching

In order to cope with high conveyor speeds using a small m_v while still being mostly independent of a fixture, we developed a third stitching mechanism called reference-based stitching.

Here, we use a fixture with fiducial marks only *once* for each type of PCA to be inspected. A reference PCA is manually mounted to a fixture, and a reference video is captured and stitched based on the fiducial marks on the tray. Its images $J_{i,j}$ and image matrices $R_{i,j}$ are saved and later used as a reference for stitching. For a fixed i and j , the image $I_{i,j}$ captured of a PCA to be inspected overlaps with $J_{i,j}$ by nearly 100%. The difference is only an Euclidean transformation. We therefore estimate $M_{i,j}$ using $J_{i,j}$ and $R_{i,j}$. Since only images and matrices with the same i and j are used at a time, we omit the indices here for simplicity's sake.

We first detect feature matches $(x_{(k)}, y_{(k)})$ in the images I and J as was done in the previous section. M must be estimated, so that

$$M^{-1}x_{(k)} = R^{-1}y_{(k)}, \forall k. \quad (17)$$

Using the same method as before, we estimate an Euclidean transformation T , such that

$$TR^{-1}x_{(k)} \approx R^{-1}y_{(k)}, \forall k. \quad (18)$$

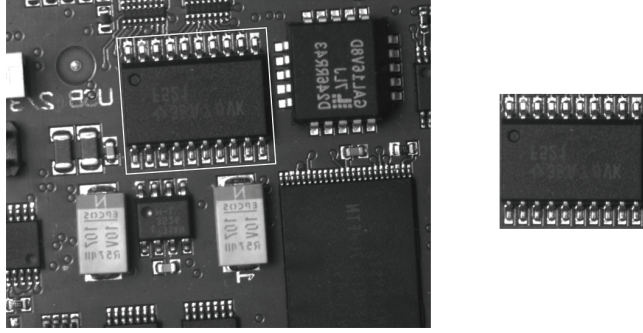


Figure 3. The left image shows a part of a PCA as seen by a camera. The camera’s rotation was exaggerated for clarity. The white box represents the bounding box of a component. It can be seen that the CAD coordinate system is rotated and translated with respect to the camera’s pixel coordinates. The right image is a temporary image created for inspection. Its coordinates are aligned with the bounding box.

Setting $M := RT^{-1}$ yields

$$M^{-1}x_{(k)} = TR^{-1}x_{(k)} \approx R^{-1}y_{(k)}, \forall k, \quad (19)$$

and Equation 17 is approximated. In practice, T is approximately equal for all i and must be computed only once using a small set of images.

This stitching method allows for conveyor speeds as fast as those achieved by fiducial-based stitching while taking as much processing time as the feature-based approach. The overall accuracy is limited by the accuracy achieved by the initial stitching of the reference video.

3.3.4 Mapping to CAD Coordinates

So far we introduced different approaches of obtaining the matrices $M_{i,j}$ that transform tester coordinates into pixel coordinates. As a last step once this is done, a similarity transformation C is computed that performs the final mapping between CAD coordinates and tester coordinates (see Equation 9). The four parameters of C denote the position of the PCA inside the tester (in our view of a moving camera array), its rotation and the difference in scale of the two coordinate systems.

To define the CAD coordinate system, a PCA always has at least two fiducial marks with known CAD coordinates, which are also used for populating the PCA. The indices of the images in which they appear are manually selected once when examining the captured video of a reference PCA. Since all future PCAs will be captured under similar conditions, knowledge about potential search areas for the fiducial marks gained from the reference video can be used to facilitate the fiducial detection mechanism during operation.

Let $\bar{x}_{(k)}$, $k \in \{1, 2\}$ be the CAD coordinates of two fiducial marks. Let $x_{(k)}$ be the pixel positions at which the fiducials have been detected in the images $I_{(k)}$. We transform them into tester coordinates $x'_{(k)}$ using the image matrices $M_{(k)}$:

$$x'_{(k)} := M_{(k)}^{-1}x_{(k)}, \forall k. \quad (20)$$

We can now calculate the four parameters of C by solving the following system of four equations:

$$C\bar{x}_{(k)} = x'_{(k)}, \forall k. \quad (21)$$

3.4 Using Videos for Inspection

Knowing C and all matrices $M_{i,j}$, coordinates can be freely transformed between the various systems. This permits the inspection of PCA components using the video captured as images $I_{i,j}$. Each component to be inspected will be visible in $m_h * m_v$ images in the average. We obtain a component’s bounding box from the CAD data of the PCA. For inspection, we create roughly $m_h * m_v$ temporary images containing exactly the component, captured under varying application-dependent viewing conditions like angle, time, camera settings and lighting. The size of the images is easily obtained by multiplying the bounding box size by the resolution r . For each pixel in the temporary image, we calculate the corresponding CAD position by linear interpolation of the bounding box coordinates.

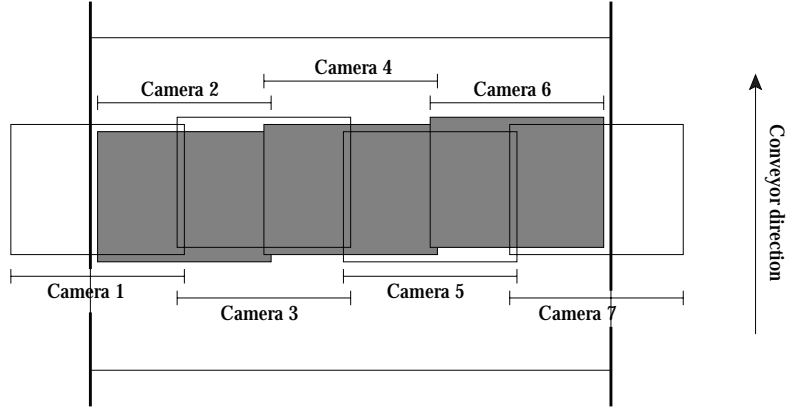


Figure 4. A row of images of a PCA on the conveyor is captured by seven cameras. The horizontal multiplicity is $m_h = 2.2$. Cameras with an even index have a lower shutter speed setting, resulting in a darker image. This assures that each position on the PCA is visible in one bright and one dark image.

Let \bar{x} be the CAD coordinate corresponding to a pixel position. We transform it into tester coordinates $x' = C\bar{x}$. The selection of a source image from which the color value is retrieved is highly application dependent. Generally speaking, indices i and j must be determined, so that

$$M_{i,j}x' = x \in [0, p_h - 1] \times [0, p_v - 1]. \quad (22)$$

The color value at x in image $I_{i,j}$ is calculated using bilinear interpolation and inserted into the temporary image. This process must be repeated for each pixel of each temporary image.

In the new images, pixel positions can be easily mapped to CAD coordinates and vice-versa using linear interpolation. This allows for efficient AOI. See Figure 3 for an exemplary camera image and a temporary image that was created for inspection. An example for the selection of source images is given in Section 3.5.

3.5 Example: Capturing HDR Videos

As an example application for our VAOI system, we implemented the capturing of *high dynamic range* (HDR) videos.

When capturing videos of a PCA for inspection, achieving proper lighting is difficult. An IC for example can have highly reflective metal pins and a dark grey label on a black surface. For an industrial camera with a CCD sensor with linear response, it may be difficult to find a suitable shutter speed that shows details in dark and bright areas at the same time. We thus use our Video-AOI prototype to capture two videos of the PCA using two different brightness settings and combine them into one video covering a higher dynamic range.

For this purpose, we set the horizontal multiplicity to $m_h = 2.22$ and set m_v to an arbitrary high value. We set each camera with an even index to a short shutter speed and cameras with odd index to a longer value within the upper bound specified in Equation 5. Like this, two subsequent images captured by the same camera have the same brightness level which increases the robustness of feature-based stitching.

Due to the horizontal image overlap of $(1 - \frac{1}{m_h}) \approx 55\%$, each position on the PCA is guaranteed to be contained in at least one bright and one dark image. When creating a temporary image as described in Section 3.4, Equation 22 will be satisfied for an odd and an even index i for each pixel. We retrieve both the dark and the bright pixel value, divide them by the shutter speed of the respective cameras and combine them into one. For details of the creation of HDR images, see:^{10,11} We create m_v temporary HDR images for each component to be inspected.

4. EXPERIMENTAL RESULTS

We used the prototype described in Section 2.6 to perform measurements of the time taken for stitching and the accuracy achieved. The tests were done with a camera resolution of 1032×776 and a spatial resolution of $r = 12$ pixels per millimeter. The other parameters remained unchanged.

Four PCAs of the same type were used. Their width is $45mm$ and their height $215mm$. The width was small enough to be captured with a single camera in this setup. With a vertical multiplicity of 2.1, this resulted in seven rows of one image per row. The first board was used as a reference and stitched by feature-based stitching. The other three were stitched based on the reference video. In each image, we selected five components that were visible in the top left and right corner, the bottom left and right corner and the center of the image. The real position of each component as seen in the images was selected manually using a mouse and compared to the estimated position obtained by transforming the component's CAD position by the estimated matrices. The average error over all 35 components was calculated. Out of the four videos, the reference video had the lowest total error, as expected. Its total stitching error was $0.53mm$. For the other three videos, the error was $0.59mm$, $0.64mm$ and $1.23mm$ respectively. Since stitching is a pixel-based operation, this error is inversely proportional to the resolution r .

We processed the captured video on a PC with an Intel Quad Core CPU with 2.4GHz. Detecting Harris feature points in a full image and computing SIFT keys took $300ms$. The feature threshold was set to a value so that roughly 600 features were detected. Matching them with the same number of features in another image took $220ms$.

For reference, detecting a fiducial mark in a full image took $135ms$. The computation time is proportional to the size of the search area. Prior knowledge about fiducial positions will thus speed up the process significantly.

Estimating an Euclidean transformation from k feature matches took about $k * 0.05ms$ with a lower bound of $0.5ms$ for exactly two features, as is the case for fiducial-based stitching.

5. CONCLUSIONS AND FUTURE WORK

We showed how a prototype for video-based optical inspection of PCAs can be built. We gave an overview of all the parameters involved and gave advice on how they can be set. The process of capturing high-resolution videos for AOI was described. The focus was put on preprocessing the videos in a way that allows to locate parts of the PCA in the captured images.

Future work includes the development of inspection techniques that benefit from the video aspect of our system. We also aim to conduct more detailed measurements of the performance of our prototype. In this process, we hope to speed up the stitching and increase its accuracy.

REFERENCES

- [1] Moganti, M., Ercal, F., Dagli, C. H., and Tsunekawa, S., "Automatic pcb inspection algorithms: a survey," *CVIU: Computer Vision and Image Understanding* **63**, 287–313 (Mar. 1996).
- [2] Kishimoto, S., Kakimori, N., Yamamoto, Y., Takahashi, Y., Harada, T., Iwata, Y., Shigeyama, Y., and Nakao, T., "A printed circuit board (pcb) inspection system employing the multi-lighting optical system," in [*Electronic Manufacturing Technology Symposium, 1990, IEMT Conference., 8th IEEE/CHMT International*], 120–129 (May 1990).
- [3] Teoh, E., Mital, D., Lee, B., and Wee, L., "Automated visual inspection of surface mount pcbs," in [*Industrial Electronics Society, 1990. IECON '90., 16th Annual Conference of IEEE*], **1**, 576–580 (Nov. 1990).
- [4] Guerra, E. and Villalobos, J., "A three-dimensional automated visual inspection system for smt assembly," *Computers & Industrial Engineering* **40**(1-2), 175–190 (2001).
- [5] Szeliski, R., "Image Alignment and Stitching: A Tutorial," *Foundations and Trends in Computer Graphics and Vision* **2**(1), 1–105 (2006).
- [6] Harris, C. and Stephens, M., "A combined corner and edge detector," in [*Alvey Vision Conference*], **15**, 50 (1988).
- [7] Lowe, D., "Object recognition from local scale-invariant features," *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on* **2**, 1150–1157 (Sept. 1999).
- [8] Lowe, D., "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision* **60**(2), 91–110 (2004).
- [9] Fischler, M. and Bolles, R., "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM* **24**(6), 381–395 (1981).
- [10] Debevec, P. E. and Malik, J., "Recovering high dynamic range radiance maps from photographs," in [*Proc. of the 24th annual conference on computer graphics and interactive techniques*], 369–378 (1997).
- [11] Kang, S. B., Uyttendaele, M., Winder, S., and Szeliski, R., "High dynamic range video," *ACM Transactions on Graphics (TOG)* **22**, 319 – 325 (July 2003).