

CONF-980117 - -

# Rule-based inspection of printed green ceramic tape

D.R. Patek<sup>a</sup>, J. S. Goddard, T. Karnowski,  
 Oak Ridge National Laboratory<sup>b</sup>, MS6011, Oak Ridge, TN 37831  
 D. Lamond, T.A. Hawkins  
 Coors Electronic Package Company, Chattanooga, TN

RECEIVED  
 FEB 25 1998  
 OSTI

## ABSTRACT

A template-based vision system for the 100% inspection of printed flaws on green ceramic tape has been developed. Design goals included a requirement for the detection of flaws as small as two thousandths of an inch on parts up to 8 by 8 inches in size. The inspection engine is a Datacube, Inc., MV200 pipeline processor. As each part is inspected, four 2K by 2K pixel quadrant images are stitched together to construct a single 4K by 4K pixel image with the aid of multiple fiducials located in each quadrant. The part fiducial locations, mask image, and punched-hole position data are generated, beforehand, from CAD designs using a defect map editor (DME), a preprocessing software package developed for the PC. The DME also generates a part "defect map". Each unique structure in the printed pattern is defined as an object. Objects are grouped into user-defined categories such as die pads, contact fingers, traces, and electrolysis buses. The map is used during the runtime inspection to associate each detected defect with an object group and a particular defect specification for that group. Repeat defects are optionally tracked for up to three consecutive parts.

**Keywords:** machine vision, ceramic green tape, defect inspection, referential inspection, defect specification

## 1. THE INSPECTION TASK

Coors Electronic Package Company manufactures ceramic component carriers for the electronics industry. One such product involves the printing of conductive metallic material onto "green" ceramic tape before it is fired. The conductor material on the post-fired product is frequently plated with gold. Voids in printing can lead to open conductors in the final product. Excess printing material inadvertently deposited on the tape can lead to shorts between the conductors. It is particularly cost effective to catch printing flaws before the plates are fired and further processed. However, many voids and extraneous printed marks do occur in non-critical part areas and the part must be allowed to pass inspection. The goal of this project is to provide an online system that can perform 100% inspection of up to 8 by 8 inch printed areas on green ceramic tape. The system is required to catch printing flaws that are unacceptable to their customers, yet allow non-critical variations to pass. In particular it is necessary to catch repetitive defects in the same area on successive parts that would indicate developing problems with either the printing "ink" or the silk screen.

## 2. IMPLEMENTATION

The system consists of a VME bus Datacube MV200 pipeline processor equipped with 96M bytes of virtual surface image memory (VSIM) and a 1K by 1K byte miniwarper module. The host computer

<sup>a</sup> D.R.P. (Correspondence): Email: patekdr@ornl.gov; WWW: http://www-ismv.ic.ornl.gov; Telephone 423-574-4674; Fax: 423-574-6663

<sup>b</sup> Work performed under DOE CRADA No. Y-1292-0078 and prepared by OAK RIDGE NATIONAL LABORATORY, Oak Ridge, Tennessee, 37831-6285, managed by LOCKHEED MARTIN ENERGY RESEARCH CORP. for the U.S. DEPARTMENT OF ENERGY under contract DE-AC05-96OR22464.

19980402 049

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DTIC QUALITY INSPECTED 3

MASTER

## DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

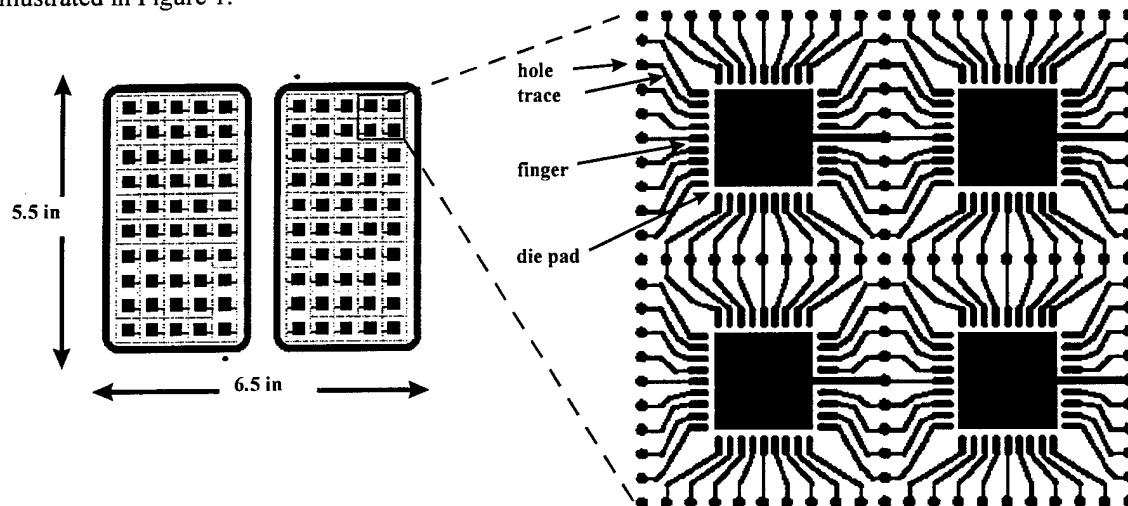
supplied with the MV200 is a Motorola VME 167 (using the Motorola 68040 microprocessor) equipped with 32M bytes of RAM and operating at 32MHz. The host runs the Lynx operating system. The runtime inspection software is written in the C language. The runtime system is controlled via an ethernet connection using the TCP/IP protocol by an industrial PC running an operator interface written in Microsoft Visual Basic. Images are acquired using a Kodak Megaplus (Model 4.2i) 2K by 2K area digital camera. A separate software package called the defect map editor (DME) was developed to generate part setup data needed by the runtime inspection system.

### Camera Resolution

Our first task was to determine the minimal camera resolution that would be required. Given the size of the printed part and the relatively tiny size of the potential defects, cost would rise prohibitively if unnecessary resolution were employed. Through a series of experiments and demonstrations it was determined that a 2-mil per pixel resolution could do the job<sup>c</sup>. The possibility of using a line scan camera together with a precision scanning stage was investigated but we learned that vibration and scanning stage durability became overriding issues. We decided to use a high-resolution 2K x 2K area camera mounted on a Linear Industries Ltd, Lintech Division, X & Y axis stage to acquire images in quadrants<sup>d</sup>. Four 2K by 2K pixel images are assembled into a composite 4K by 4K pixel (8 bits per pixel) image which is then analyzed for defects.

### The Inspected Part

A typical printed pattern as imaged by the camera after the four quadrants have been reassembled is illustrated in Figure 1.



**Figure 1. Simulation of a printed green tape pattern. It consists of two multi-component carrier arrays with an overall dimension of approximately 5.5 in by 6.5 in. Each array consists of 50 components. The blowup in the figure to the right shows four such components.**

It consists of two multi-component carrier arrays with an overall dimension of approximately 5.5 in by 6.5 in. Each array consists of 50 components. The blowup in the figure above shows four such components.

<sup>c</sup> While below the Nyquist sampling limit, use of this camera resolution leads to the **detection** of 2-mil **high-contrast** voids and excess paste marks most of the time. The defect signal rapidly approaches the noise level as the size drops below 2 mils.

<sup>d</sup> Z-axis motion capability was added for auto-focusing capability.

We refer to the center black square as the die pad, the elongated rectangular objects lined up around the four edges of the die pad as fingers, and the lines moving away from the fingers to the circular objects bordering each component as traces. The circular objects bordering each component are holes punched through the tape. The holes are typically 23 mils in diameter. Traces range in width from 8 to 12 mils (thousandths of an inch) depending upon the design of the part.

### Referential Inspection – Use of the “Golden” Template

The use of the well-known “golden” template reference-based inspection was an obvious choice for this inspection task. Images of printed parts are precisely aligned with a flawless template and are compared by pixel subtraction. Two images that are identical will yield an image containing all zero pixels. However, any differences between the template and the part will appear as a non-zero pixels. In a pipeline processor such as the MV200, two images can be subtracted in 0.85 seconds at a rate of 20M pixels per second.

The entire process is summarized in Figure 2.

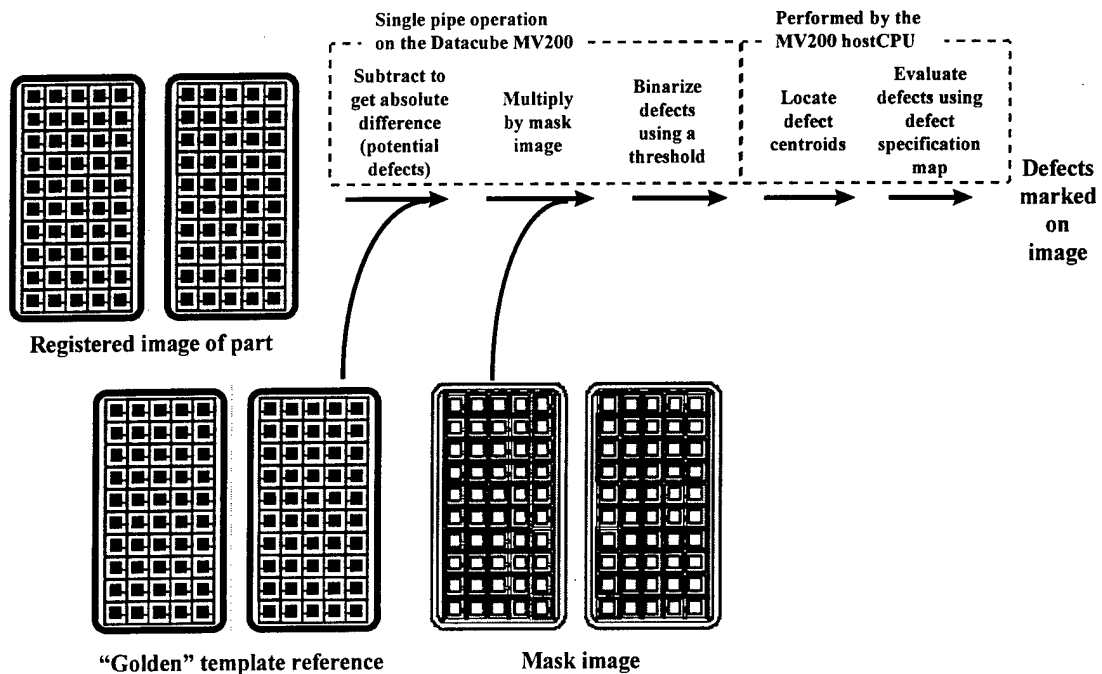


Figure 2. Summary of the entire inspection process. The single pipe portion performs the template subtraction and difference masking. The MV200 host CPU performs the remaining operations including hole-to-trace continuity inspection (not shown).

### The Composite Image

Input images from the Kodak camera are tiled together to form a composite 4K by 4K pixel image as illustrated in Figure 3. To reassemble the 4K by 4K image, the individual 2K by 2K quadrant camera images are registered with fiducial tie-points to the original CAD drawing so that each final image is geometrically registered with the template to sub-pixel accuracy. To meet the processing time requirements, this operation requires the use of the miniwarper module in the Datacube MV200 board. The miniwarper is capable of handling up to 1K by 1K size images at a 10 MHz. pixel rate. The hardware uses a 2<sup>nd</sup> order bi-quadratic mapping of the form

$$x_s = ax_d^2 + bx_d + cy_d^2 + dy_d + ex_dy_d + f$$

where  $x_s$  is the source pixel location and  $x_d, y_d$  are the destination pixel coordinates. A similar expression is used to determine the  $y_s$  source pixel location. Fiducial points in each 2K by 2K image are located using either a normalized correlation technique or a sub-pixel edge locator. The fiducial locations are used as tie points to calculate the six warping coefficients. A minimum of three fiducials is required for the second order warp although additional ones give more accurate results from a least square error fit. The 2K by 2K image is warped using nine tiles each of approximately 670 by 670 pixels. During the warp, the tiled image may shrink so that the source tile must be larger than the destination tile. The maximum amount of shrinkage is a parameter defined during machine setup. Overlaps between tiles are required to prevent gaps. The composite 4K by 4K image is thus made up of thirty-six individual tiles.

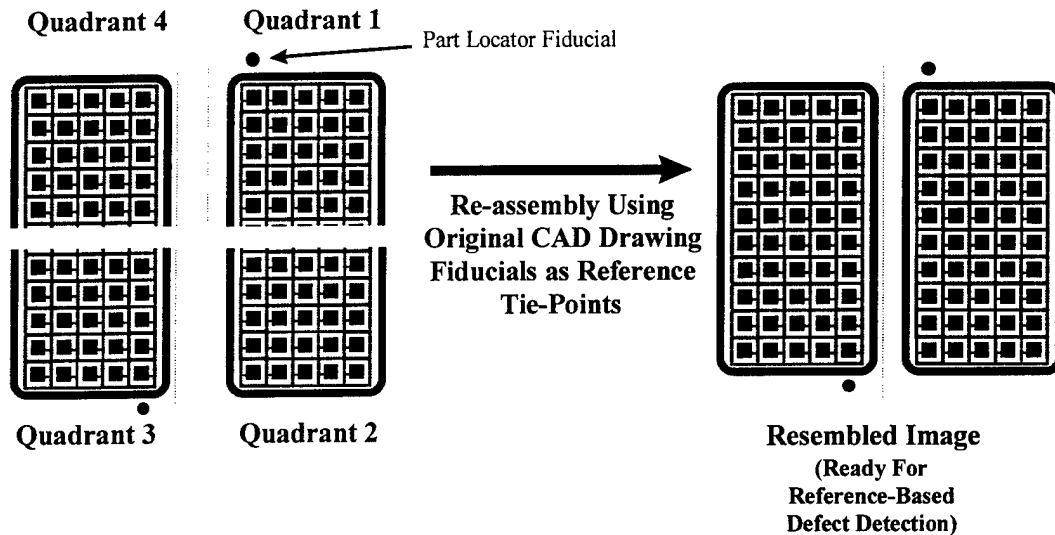


Figure 3. A single image of the part is reassembled from four quadrant images with the help of fiducial tie-points.

### The Mask Image

The mask image referred to in Figure 2 is derived by the DME beforehand from the original CAD drawing of the part. The complement of this image is shown in Figure 4. The absolute difference image obtained by subtracting the image of the inspected part from the template is multiplied by the mask complement image to reduce unavoidable pixel differences that arise from a variety of sources. Where the complement mask image is black, the non-zero pixel differences are reduced to zero. The extent of this reduction depends upon the blackness of the mask complement region.

The edges of the printed objects are extracted using the Sobel operator. Edge differences arise from slight misregistrations that exist between the part image and the template. Other differences arise from slight variations in the size of the printed features due to changes in the consistency of the ink. For example, if more ink is pressed through the screen during printing, the printed object becomes slightly dilated. Dilations of the holes in the form of "donuts" are also added to the mask at the location of the punched holes to eliminate differences arising from variations in the positioning of the punched holes relative to the printed pattern. The DME also adds masking to the inside corners of all traces to eliminate differences that arise from ink flowing across. The mask is also used to eliminate acceptable defects in the template since a perfect template in reality doesn't exist. Finally, masking is added to eliminate differences due to other marks and punched holes that are present as part of the manufacturing process.

## Rule - Based Defect Specification

In addition to masking, the system tolerates non-critical printing flaws by using rule-based defect specification. Differences between the part image and the template that are not masked are evaluated with respect to position by using a defect map. In the setup phase, the DME is used to assign a number to each object in the printed pattern. The defect map overlays the template so that every pixel of the object in the defect map is assigned the same object identifier. When a significant difference "blob" is detected, its centroid is located in the defect map so that its object identifier may be extracted. The identifier then serves as a key to lookup the defect specification rule for that printed object. For example, if the defect is located in a narrow trace, the rule would probably indicate that this defect is significant and should be called to the operator's attention. The rule would simply state that one or more defects of any size found in a trace of this type should be considered to be significant. On the other hand, a small void located in a die pad region would be registered for that particular die pad but would otherwise be ignored. For the sake of this example, the rule would state that up to three voids are allowed as long as the accumulated area does not exceed 10% of the die pad area. The assignment of defect rules is discussed in a later section.

## Defect Map Compression Method

The defect map is too large to reside in fast RAM during runtime inspection. Access from disk would adversely impact performance. Standard compression methods can achieve high compression ratios but have the disadvantage that the entire file must be decompressed before a data value can be determined. During operation, access to the defect map is only required for a relatively small set of defect candidates. These defect locations, moreover, can be located anywhere on the image. The method used for compression, then, must be able to quickly access locations anywhere within the defect map without decompressing the entire image.

We developed a pyramid-based compression technique for this application. The defect map image is characterized by numerous regions of constant gray level and by regions with similar patterns. Initially the image is mapped into an array of blocks of size  $n$  by  $n$  where  $n$  is the block size in pixels.  $n$  is required to be a power of 2. If all the pixels within a block have the same pixel value, the pyramid is terminated for that block and the block is set to the common value. This operation is performed for all blocks at this level. For those blocks that do not have a constant value, the block is divided into a set of 2 by 2 sub-blocks. Each sub-block in turn is examined for a common pixel value. If the sub-block is not a constant value, it is also examined for the same pixel value pattern as other sub-blocks at this same level. As in the first level, a sub-block with a constant value is terminated with that value. For the case of multiple blocks with the same pattern, only one copy of the pattern is retained. This processing is repeated for each level until the block size becomes one. At this point the compression is complete. Access to an image data point requires traversing through a maximum of  $\log_2 n$  levels. A tradeoff then exists between the compression ratio that

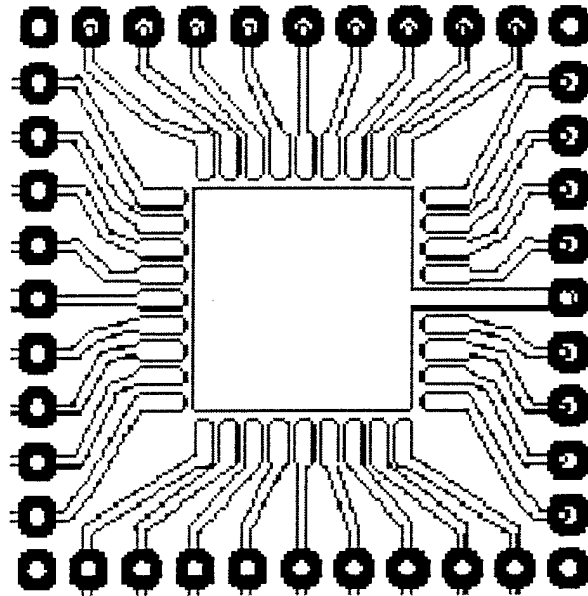


Figure 4. The mask image contains printed pattern edges and dilated hole information to allow the accommodation of acceptable misregistrations. Masking is placed at the inside corners of printed pattern objects to enable the accommodation of ink "wicking"

increases as initial block size  $n$  increases and the speed of access to individual data points. Typical block sizes used during operation of the system are 32 and 64. These values give compression ratios on the order of 2.5 to 1.

### **Illumination**

Near flat illumination is provided by a pair of fluorescent lamps one located on each side of the part being inspected. The lamps are powered by high-frequency ballasts. Moveable shades were added to each lamp fixture to help flatten the illumination field. A pseudo-color tool was added to provide a flat-field "tuning" aid during setup. The system sensitivity must be changed for each part type to provide maximum contrast. The operator does this by adjusting the camera lens (Schneider Componon-S 80/4.0 Leica Preset) aperture together with the camera's electronic gain and offset.

### **Initial Camera Location Setup**

Registration is greatly improved if the camera image array is adjusted to be as nearly parallel to the plane of the part as possible. A measurement aid for use during initial setup and alignment was developed. The technique measures the six-degree-of-freedom position and orientation (pose) of the camera with respect to the part inspection area. A white sheet with an 11 by 15 array of filled circles at 1-inch center to center spacing is placed on the inspection area. The system then calculates the position and orientation information from an image of the dot pattern acquired by the camera. In particular, the distance of the camera from the sheet and the vertical off-axis angle are used to ensure that the camera is aligned within tolerance at each of the four staging positions where part images are taken.

The measurement method uses a point-based algorithm and least square error fitting to a stored model to compute the position and orientation.<sup>1,2</sup> First, the black dots are segmented from the white background, and blob analysis is performed to calculate the x and y centroids of each dot. The centroids are matched to the stored pattern based on the relative arrangement of dots within the image. At least four points are needed to guarantee a unique result. Additional points reduce the error. Using all matched centroids found from the blob detection, the pose is calculated using an iterative least square error algorithm. The least square fit pose algorithm is based on a modified Newton's method procedure with an over-determined number of feature points. The pose solution is found iteratively with termination based on an error criterion. The algorithm runs until the correction between iterations is below a limit value or the squared error does not change sufficiently. A maximum number of iterations terminates the algorithm if convergence is not reached. The result is valid if the sum of the squared errors in the distances between the projected model points and the image points is less than a threshold value. An initial pose is assumed. The initial values are zero except for z translation which has a value near the midpoint of the expected range. The algorithm minimizes the least square error between the model points projected to the 2-D image plane and the actual points found in the acquired image.

### **Inspection Throughput**

The overall throughput varies from approximately 9 to 20 seconds. The time required for the template subtraction pipe shown in Figure 2 is approximately 0.85 seconds. However the time required to acquire and reassemble the part image is 5 seconds and the remaining inspection time varies depending upon the number of defects detected and the number of holes that the MV200 host CPU is required to process. The 68040-based Model VME 167 host computer is the rate-limiting. The inspection of trace to hole continuity is also performed by the host CPU and requires more than a second. Newer and faster MV200 host processors with larger quantities of RAM memory have become available since this effort was begun and we believe that the overall inspection time can be shortened significantly.

### 3. THE DEFECT MAP EDITOR

The DME provides the Datacube MV200 runtime inspection engine with most of the required preprocessing data. This data is derived from CAD drawings of the part paste pattern and the hole punch layout. The information generated by the DME is summarized in Figure 5. Processing of the CAD files to a BMP file is performed using ASM 501 - DXF to Gerber translator available from Artwork Conversion Software, Inc. The DME converts it to a 500 pixel-per-inch TIF format prior to processing. Once produced for a given part type, the data is saved on disk for re-use by the runtime system.

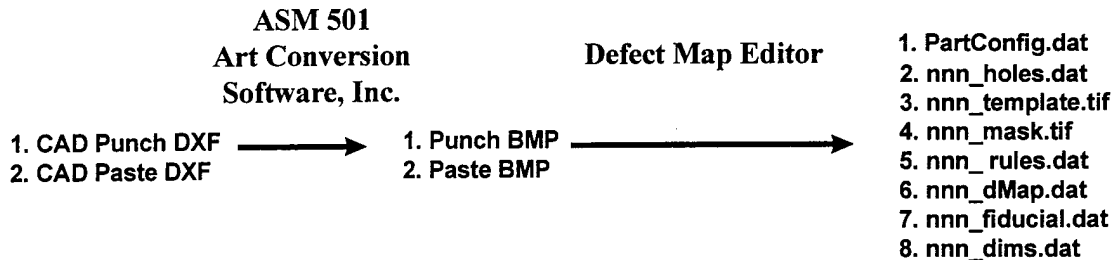


Figure 5. The Defect Map Editor provides the runtime inspection with part specific data.

Figure 6 and Figure 7 illustrate portions of the hole punch and paste patterns CAD images seen by the DME user prior to defect map editing. The punched hole drawing is needed to generate the mask "donuts" for incorporation into the mask image, "nnn\_mask.tif". This drawing and the paste pattern are used to generate hole location data including trace-to-hole connectivity data, "nnn\_holes.dat", which is needed during the hole inspection phase that follows the referential inspection phase described earlier. The

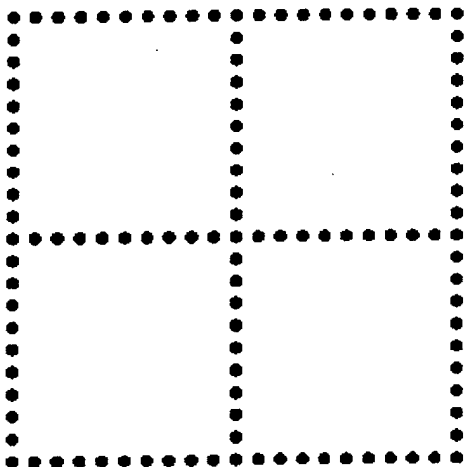


Figure 6. The hole punch pattern prior to defect map editing.

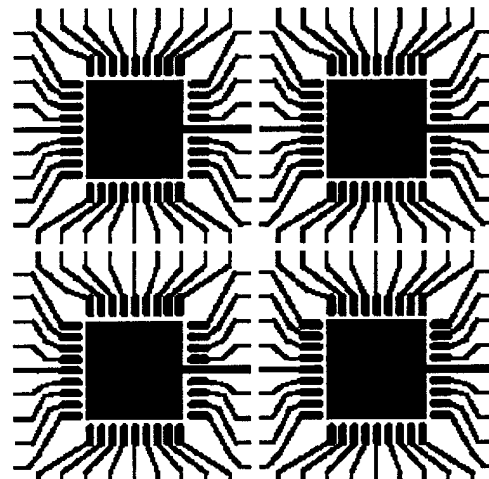


Figure 7. The paste pattern prior to defect map editing.

punched hole and paste patterns are overlaid to generate a CAD template, "nnn\_template.tif", which is needed during system setup. Both the actual template images and the part images that are acquired by the camera are registered to fiducial tie-points originally identified on the "CAD" template. The DME user identifies the fiducial locations and fiducial type using a "point-and-click" tool and the results are placed in the "nnn\_fiducial.dat" file. The compressed defect map, "nnn\_DMap.dat", and the defect specification rules, "nnn\_rules.dat", are generated with input from the DME user. The "nnn\_dims.dat" provides the runtime system with some miscellaneous part dimension information.

## Defect Map Generation Using the Defect Map Editor

The DME allows the part setup operator to extract the various part structural features from the CAD drawings and to assign the appropriate defect specification rules. Our intent was to make this process operator friendly. The DME provides the user with a combination of automatic feature extraction tools and interactive editing tools. The step-by-step application of these tools is recorded into scripts which can be replayed during later editing sessions. The interactive editing operations need only be applied to the upper-left component of each multi-component array. Prompt messages can be used to instruct the user on the next operation to be executed in the overall defect map generation sequence. Prior to any interactive editing, the pitch of each array is analyzed and recorded by the DME with interactive input from the user.

In general each object type (e.g. die pads, fingers, traces, etc.) is extracted into its own object image which is saved by the DME for later processing. Each object image is called a "layer" and will ultimately be assigned its own defect specification rule. Figure 8 illustrates use of the rectangular region-of-interest (ROI) extraction tool. Figure 9 shows usage of the cut and click tool was designed to extract individual fingers and place them into the finger object image.

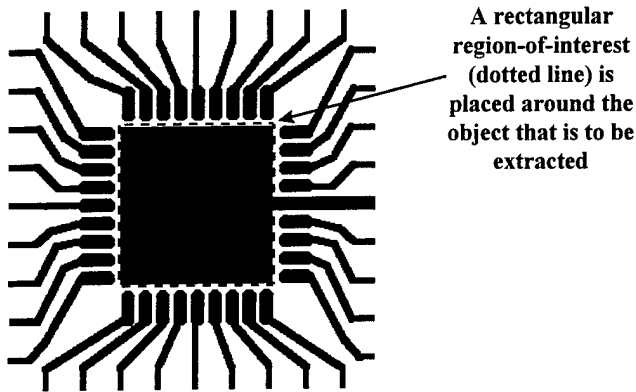


Figure 8. The region-of-interest (ROI) object extraction tool. In this example the user has selected the die pad for

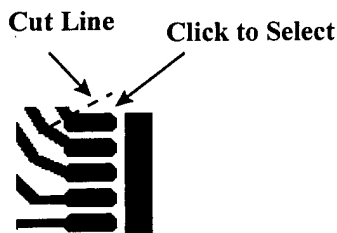


Figure 9. The cut-and-click extraction tool. The user has drawn a cut line at the back of the finger to be extracted.

DME has extracted the object layers and the operator has entered a defect specification rule for each, the DME reassembles them into the defect map and the object to rule lookup table. Rules are assigned using the DME dialog box shown in Figure 12 following the syntax shown in Figure 13. The operator indicated that 3 or more defects must be found in any single die pad or the combined area of the defects must be greater than 10 % of the die pad's area before the part is considered to be defective. Also, up to 10 small

A rectangular region-of-interest (dotted line) is placed around the object that is to be extracted

The "object" extraction tool extracts objects based upon the distance of the edges of an object from its medial axis (skeleton). The algorithm diagrammatically illustrated in Figure 10 shows a combination of two distance transform routines that can assign a number based on size to the pixels of an object so the object can be extracted simply by the like value of its pixels. In the illustration shown, the pixels of a 5-pixel wide trace are assigned the value of 3 (the width divided by 2, plus 1). Using the same algorithm, a 7-pixel wide trace would be assigned the value of 4 (not illustrated). During execution of the

object extraction tool, the distance transform skeletonization proceeds while each layer of newly eroded pixels is assigned the iteration number. Iteration ceases when no further erosion occurs. A copy of the skeleton that was generated is used as a seed for iterative constrained dilation back to the original shape. Each newly added pixel is assigned the iteration number. The result is added to the previous result to get the sum which reflects the original width. The final segmentation occurs when all pixels of a given value or between a range of values is extracted. The algorithm yields useful results if the skeleton branch points or endpoints are employed as seeds for the constrained dilation step instead of only the medial axis.

An example of the sequence of steps that would be used to extract the layers of a printed pattern is shown below in Figure 11. Once the

printing errors (less than 3 pixels in width or height) could be allowed in unprinted areas that were not assigned to other layers.

The “allpaste” layer receives all printed objects that were **not** previously extracted into one of the printed object layers. The “remainder” layer receives all areas of unprinted tape that were not assigned to layers containing unprinted objects (e.g. the narrow spaces between fingers).

A visual image is shown in Figure 14 illustrates the defect map layers. Each object **type** is represented with a different gray level. The actual defect map is a 16-bit per pixel 2D array where each pixel contains an object number. The object number serves as a key to find its object type and thus defect specification rule.

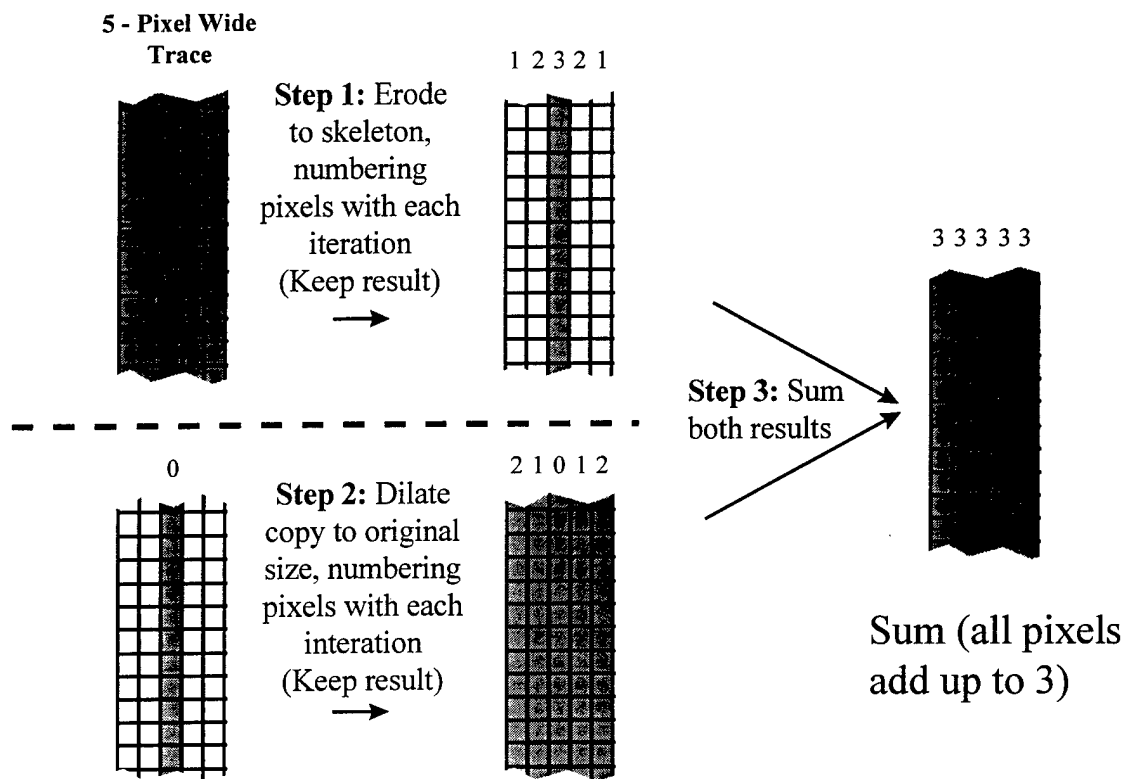


Figure 10. The “object” extraction tool extracts objects based upon the distance of the edges of an object from its medial axis (skeleton).

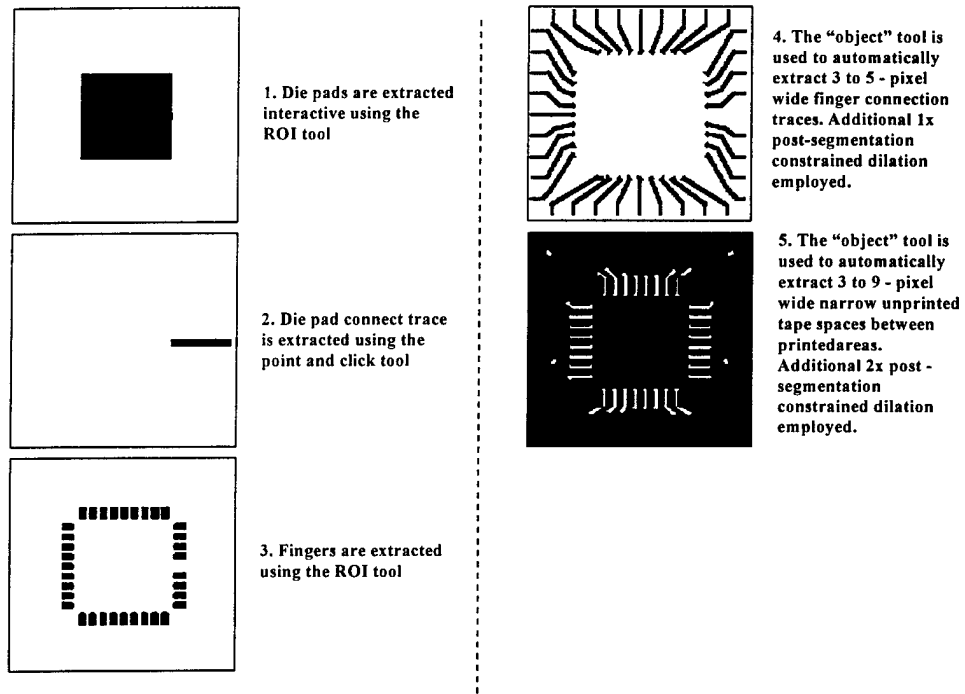


Figure 11. Sequence used to extract object type layers from the paste pattern image. All editing is performed on the upper-left component of each array.

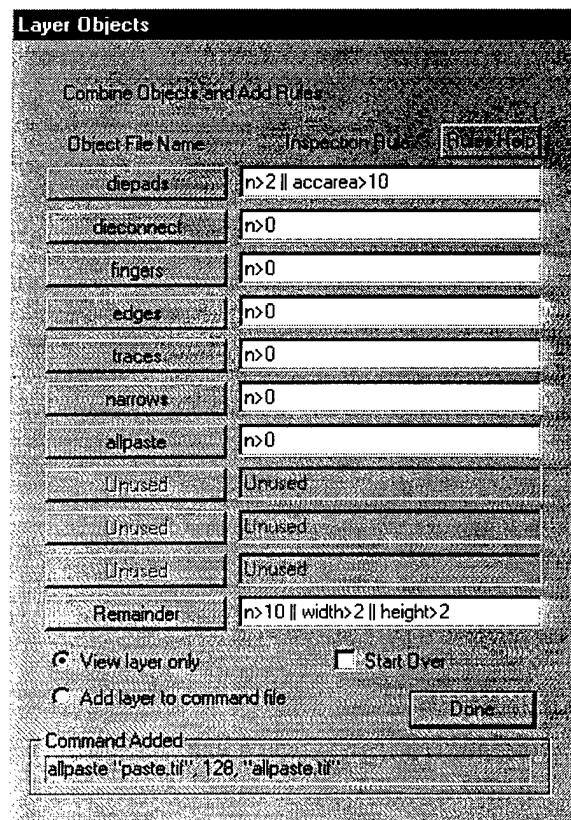


Figure 12. DME dialog to aid in the reassembling the object layers.

<p><b>Relational Operators</b></p> <p>&lt; Less than          &lt;= Less than or equal          &gt; Greater than          &gt;= Greater than or equal          == Equal to          != Not equal to</p>	<p><b>Reserved Variables</b></p> <p>area Area in square pixels          accarea Cumulative area          (% of object area)          n number of defects          width width in pixels          height height in pixels</p>
<p><b>Logical Constants</b></p> <p>T true          F false</p>	<p><b>Related to Consecutive parts</b></p> <p>part0 current part (1 is failure,          0 is passed)          part1 preceding part (1 is failure,          0 is passed)          part2 two parts previous (1 is failure,          0 is passed)</p>
<p><b>Logical Operators</b></p> <p>&amp;&amp; And             Or</p>	
<p><b>Miscellaneous</b></p> <p>: Separates the object rule from the part rule for each object type</p>	
<p><b>Default rule</b></p> <p>n&gt;0: part0==1 &amp;&amp; part2==1 &amp;&amp; part3 ==1</p> <p>which is translated "the process will stop if a single defect occurs on the same object on 3 consecutive parts"</p>	

Figure 13. Rules consist of relational statements that are constructed from the elements shown above. The default rule is an example.

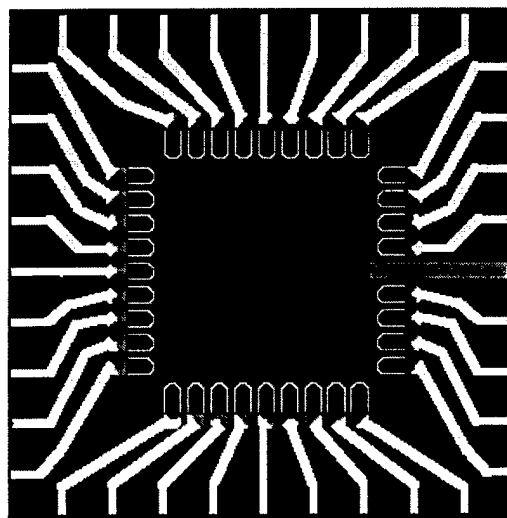


Figure 14. The visual graphic that illustrates the different object types extracted by the DME and overlaid into a single image. This is not the actual defect map.

#### 4. SUMMARY

In summary, a template-based vision system for the 100% inspection of printed flaws on green ceramic tape has been developed. Design goals included a requirement for the detection of flaws as small as two thousandths of an inch on parts up to 8 by 8 inches in size. The system is defect tolerant through the use of defect specification rules that can help the system distinguish between critical and non-critical regions on the printed part.

#### 5. REFERENCES

1. W.B. Jatko, J.S. Goddard, R.K. Ferrell, S.S. Gleason, J.S. Hicks, K. Varma, "Crusader Automated Docking System Phase III Report," Oak Ridge National Laboratory, Oak Ridge, TN, ORNL/TM-13177, March, 1996.
2. D.G. Lowe, "Fitting Parameterized Three-Dimensional Models to Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 5, pp. 441-450, May 1991.

M98002728



Report Number (14) CORN/CP-- 96066  
CONF-980117 --

Publ. Date (11) \_\_\_\_\_  
Sponsor Code (18) 199801  
JC Category (19) DOE IXF  
UC/900, DOE/ER

DOE