AGING AIRCRAFT RIVET SITE INSPECTION USING MAGNETO-OPTIC IMAGING: AUTOMATION AND REAL-TIME IMAGE PROCESSING

Yuan Fan\textsuperscript{1}, Yiming Deng\textsuperscript{1}, Zhiwei Zeng\textsuperscript{1}, Lalita Udpa\textsuperscript{1}, William Shih\textsuperscript{2}, Gerald Fitzpatrick\textsuperscript{2}
\textsuperscript{1}Dept. ECE, 2120 Engineering Building, Michigan State University, East Lansing, MI 48824, USA
\textsuperscript{2}PRI Research & Development Corporation, 25500 Hawthorne Blvd., Torrance, CA 90505, USA

Abstract
Magneto-optic imaging (MOI) technique, developed in the last two decades, is becoming increasingly popular for aging aircraft inspection. In recent years, it is widely used in detecting surface and subsurface cracks and corrosion in aircraft skins. The merits of MOI that make it attractive include rapid and large-area inspection, insensitivity to liftoff variations, easy interpretation and low false call rate. Although interpreting MOI images is much easier than interpreting complex impedance signals in conventional eddy current inspection, the decision of acceptance or rejection is subjective in nature and is affected by human factors and fatigue leading to a spread in the confidence level of POD curves. To reduce this detection variability from inspector to inspector, a possible solution is the development of a system that provides automated inspection including detection of rivets and defects, classification of defects and accept/reject decision. This system can also be used for automated scanning by mounting the MOI imager on a robot. This paper presents the development of a technique for processing MOI image data to eliminate background noise with automatic detection and classification of surface and subsurface cracks at rivet sites. Real-time rivet detection and classification is implemented on a TMS320C6000 DSP platform. Results of automated signal analysis are presented on experimental magneto-optic image data obtained from the AANC at Sandia National Laboratories.

Introduction
Magneto-optic Imaging (MOI)\textsuperscript{1} is a recently developed nondestructive evaluation (NDE) technique that is widely used for detecting cracks and corrosion in aircraft skin. The MOI technique uses an induction foil to induce eddy currents in the specimen, and uses a magneto-optic (MO) sensor for imaging the magnetic fields associated with the eddy currents interacting with surface and subsurface anomalies such as rivets and cracks\textsuperscript{2,3}. The main advantage of the MOI is rapid inspection and ease of interpreting image data in contrast to complex impedance signals from conventional eddy current instruments.

MOI images are obtained manually by scanning the MOI imager over the inspection surface. While easy to interpret, the inspector must use judgment to determine whether or not a defect exists. This leads to variability in POD (probability of detection) from inspector to inspector. Background noise inherent in the MO detector also contributes to some uncertainty in image interpretation. This is particularly true for very small fatigue cracks at rivet sites. An automated MOI system which can eliminate background noise and provide automatic detection and classification of structural defects offers the capability to increase reliability as well as speed of the inspection. In addition to achieving high accuracy of detection, the algorithms for filtering and classification need to meet real-time processing conditions.
This paper is organized as follows. Section 2 introduces the automated rivet inspection algorithm. Section 3 describes the DSP hardware system for the real-time automated rivet inspection. Finally, section 4 describes the experimental results along with a discussion of issues related to automated rivet inspection.

Automatic Rivet Inspection System

The overall approach for automated rivet inspection is depicted in Figure 1. The raw image obtained from the MOI system is applied to the image processing module to remove the unwanted serpentine noise present in the image and generate a ‘clear’ binary image. The binary image is used in the subsequent rivet detection and rivet classification modules. The descriptions of algorithms are explained in detail in the following sections.

In a sequence of MOI images, noise associated with the domain structures in the sensor is stationary as the sensor moves while image features associated with rivet or corrosion in the test sample move from frame to frame as the MOI sensor is scanned over the sample. Using this fact, a motion based filtering (MBF) algorithm can separate out the defect signals from the static background noise. The MBF contains two parts, Multiple Frame Subtraction (MFS) and post-processing. MFS algorithm detects the moving object based on the detection of pixel-intensity changes in a frame sequence. The difference image $D_i(x,y)$ is obtained by:

$$D_i(x,y) = f_{n-i}(x,y) - f_n(x,y)$$  \hspace{1cm} (1)

where $f_n(x,y)$ is the current frame and $f_{n-i}(x,y)$ is the $i$-th previous frame. However the difference image above also removes any overlap between frames in the object of interest. Complete information about the object of interest can be recovered by using multiple difference images. This is achieved by subtracting the current image from several past images, and combining the difference images using a MAX operation to get the result image, $S(x,y)$, as described below.

$$S(x,y) = \max_{1 \leq i \leq w}(D_i(x,y))$$  \hspace{1cm} (2)

where $w$ is the number of difference images combined within the Max operation. Each pixel value $S(x,y)$ is the maximum value of $D_i(x,y)$, $1 \leq i \leq w$.

The resultant image of MFS is typically of low-contrast which is caused by the subtraction and in addition contains some arbitrary isolated noise pixels. In order to enhance the image contrast and remove the noise pixels, a post-process stage is applied. The post-process includes three stages: a 5x5 median filter to remove the isolated noise pixels; a grayscale stretching function to enhance the image intensity; a threshold function to remove the negative valued and small positive valued noise pixels. Figure 2 shows the resulting gray image of motion based filtering algorithm. Four connective image frames are...

![Figure 1 Block diagram of the automated rivet inspection system](image-url)
used to extract the moving object. As seen in Figure 2 (e), background noise is almost removed while the moving objects are retained.

Figure 2.1 (a) - (d) MOI images in time sequence. Rivets are moving to the left while sensor is moving to the right; (e) filtered image of (d) by MBF

Rivet Detection

A typical MOI image after MBF filtering is shown in Figure 3, which contains two rivets. The left rivet site is good but the rivet site on the right has a radial crack. We can see the normal rivet image is roughly circular in shape while the abnormal rivet is non-circular. Rivet detection can be performed using different techniques such as circular Hough transform\(^7\); morphological image processing; and 2-D convolution.

Figure 3 A typical MOI image after MBF filtering

The Hough transform is a traditional circle detection algorithm. However, when the rivet edge is concavo-convex, the detected circle is not satisfactory. Also, we need to predefine the radius range and threshold value which depends on the test sample. The convolution method simply involves segmenting the image, extracting the rivet region and convolving using a known kernel. There is no user specified parameter involved but the convolution operation is computation-intensive. The morphologic operation based method is the fastest of the three methods. Considering the real-time requirement, we chose the morphological operation based rivet detection algorithm.

The basic idea of this method is using the morphological operation of erosion, to find the center of the rivet object. The erosion of binary image B by structuring element S is expressed as\(^8\):
If the structuring element $S$ was chosen as a circular element whose diameter is close to the object rivet size, the eroded result image will only contain a few non-zero pixels at the rivet center. The real center of the rivet is given by the centroid of the result after erosion.

The iterative implementation procedure using the erosion operation is shown in Figure 4a. The first step is to segment out the rivet region. The segmented result is shown in Figure 4 (b)-(d).

The initial center position, $c$, is chosen as the center of the rectangular boundary box. The initial radius is defined as

$$ r = \min(c_x - x_1, x_2 - c_x, c_y - y_1, y_2 - c_y) $$

(4)

Usually the initial radius is larger than the actual rivet radius and the erosion result image will be empty. The radius of the structure element $S$ is decreased to the largest value for which the erosion produces a non-empty result.

After erosion, the result image only contains a few non-zero pixels at the rivet center as seen in Figure 4(c). The true rivet center $c'$ is derived as the center of the non-zero object. The true rivet radius was defined as:

$$ R = \frac{1}{N} \sum_{i=1}^{N} d_i $$(5)

where $d_i$ is the distance between the edge pixel and the true center $c'$.

**Rivet Classification**

An important issue in classification is the identification of the critical features of the rivet image that quantifies the presence of a crack. From Figure 3, we see that a normal rivet is roughly circular in shape and is contained inside the detected circle while the abnormal rivet has an additional ‘blob’ which lies outside of the detected circle. Using this fact, a simple skewness function $d$ is defined as the distance from the edge to the rivet center. In this paper, we introduce two skewness functions.
Skewness function $S_i$ is defined as:

$$S_i = \sqrt[3]{\frac{R}{100 + B r^2 + \sum (D_i - r)^2}}$$

where $f$ is the eddy current frequency and $B$ is the ‘width’ of the additional region outside the rivet in the image, $r$ is the average measured radius of rivet and $\{D_i, i = 1,2,\ldots,n\}$ are the distances from the rivet center to the edge pixels. Finally, $\hat{R}$, the mode of $\{D_i\}$, is computed by calculating the histogram of $D_i$ and picking the value at the peak location.

As seen in figures 5, images of rivets close to the vertical edge of a horizontal lap joint are distinctly different from the interior rivet images. This is due to the constraint imposed by the vertical edge on eddy current flow direction. In order to take this difference into account, a second skewness function $S_e$ is applied for the edge rivet images and is defined as:

$$S_e = \frac{A_1 - A_2}{A_1 + A_2} \frac{|X_m - Y_m|}{X_T}$$

where $A_1$ and $A_2$ are the areas of each lobe. $X_m, Y_m$ are dimensions of the largest lobe in the x and y direction respectively, and $X_T$ is the total dimension of lobes in the x direction.

![Figure 5 MO images](image)

Figure 5 MO images (a) interior image (b) edge image without crack (c) edge image with crack. The red triangular represents the center of the rivet image.

Hence an additional step is introduced to classify the detected rivet into ‘interior’ and ‘edge’ type. Appropriate skewness functions of the detected rivets are calculated for classification into ‘good’ and ‘bad’ rivets.

### Implementation of Automated Rivet Inspection on the Digital Signal Processor System

Real time image processing systems are often data-intensive, as well as computation-intensive. The automated rivet inspection algorithm needs to process large amounts of image data in real time which requires efficient data transfer mechanisms as well as high computing power. The TMS320DM642 (DM642) device is based on the second-generation high-performance, advanced VelociTI™ very-long-instruction-word (VLIW) architecture (VelociTI.2™) developed by Texas Instruments (TI), making the DSP an excellent choice for digital media applications. The DM642 Evaluation Module (EVM) is a low-cost high performance video & imaging development platform designed to jump-start application development. We choose this evaluation module as the hardware platform for the real-time automated rivet inspection system. Figure 6 shows the dataflow of the hardware system.

The dataflow follows the following sequence:

Step1: A frame is captured from the MOI camera.
Step 2: The captured frame is converted to the gray image and stored in the original image buffer.

Step 3: The input frame is processed using the MBF algorithm and converted into a binary image which is then stored in the processed image buffer. A classification decision is made based on the rivet detected in the binary image. In order to show the classification result, we put a color bar in the both original image and corresponding processed image. A red bar means the rivet site is abnormal while the green bar means the rivet site is good.

Step 4: The original image and processed image are sent to the video encoder to display on the monitor.

**Figure 6: Dataflow diagram of the real-time rivet inspection hardware system**

All algorithms are coded in C programming language. Using Texas Instrument’s integrated development environment, Code Composer Studio (CCS), the C code can be converted to DSP assembly language that can run on the DSP hardware system. In order for the algorithms to work in a real-time system, there must be an application framework to connect algorithms with DSP hardware system. In a typical DSP application, the framework is a software module or a group of software modules that resides on top of algorithms and peripheral I/O drivers. The framework controls transfer of input data from peripheral devices, sending the data to algorithms for processing and transferring the processed data to peripheral devices for display. The architecture of the framework is shown in Figure 7. The framework uses a three task setup. Before initiating the BIOS task scheduler, the code performs initialization of various modules used in the system which includes the board and processor initialization, the RF-5 modules initialization and the capture channel and display channel creation. After these initializations, the system enters three task configuration managed by BIOS scheduler. The three tasks are:

Input task:
The input task will capture the frames from the input device. The input YUV format image is converted into RGB format and stored in the original image buffer. It then sends the message to process task. The task then waits for the message from the output task to continue.

Process task:
The process task is responsible for the image processing, rivet detection, and rivet classification. It waits till it receives the message from the input task. After the rivet is classified, it sends a message to output task. The system then waits for the message from the input task to continue.

Output task:
The output task is responsible for displaying both the original image and processed image with the color bar which shows the classification result on the monitor. After sending the data to the output frame it sends a message to input task. The system then waits for the message from process task.

The real-time processing system built in NDEL lab is shown in Figure 8. The raw image is stored in a video tape. In place of an MOI a VCR is used to play the video tape and output the raw image signal. Both the raw image and processed image with classification result color bar are shown on a PC monitor.
DM642 processor initialization
- BIOS initialization
- CSL initialization
- DMA initialization

eXpressDSP Reference
Framework5 (RF5) Modules initialization
-- Channel initialization

Capture/Display driver configuration
- Create the capture channel
- Create the display channel

BIOS scheduler starts

Input Task
{ Acquire a frame
   Send the message to process task
   Wait for the message from output task }

Process Task
{ Wait the message from the input task
  MBF image processing
  Rivet detection
  Rivet classification
  Send the message to output task }

Output Task
{ Wait the message from the process task
  Display the result image
  Send the message to input task }

Figure 7 Architecture of the software framework

Figure 8 Real-time processing hardware system and result image
Experimental results and discussion

POD curves obtained using the MOI on a set of fatigue crack samples produced to evaluate various inspection techniques are shown in Figure 9. The four curves to the right were obtained using the earlier MOI 301 and an eddy current based sliding probe. The furthest curve on the left was obtained subsequently using the newer MOI 303.

![Figure 9 POD curves for the manual MOI inspection on fatigue cracks](image1)

All the data in Figure 9 for the POD curves were obtained manually. The automated and real-time MOI inspection system was tested on the same set of lap splice samples consisting of 36 panels with 720 rivet sites containing first layer fatigue cracks of various sizes. An MOI 303 was used to scan the panels and the resultant images were recorded on video tape. This video tape was used to provide the raw MOI images that were processed.

The results from the automated algorithm used to process the MOI images are shown in Figure 10. POD curves for two skewness function thresholds are shown. Lower threshold values yielded better PODs at the expense of higher false call rates. For example, using a single classification algorithm on both interior and edge rivets and a threshold of 0.210 for the skewness function $S_1$, the performance was 0.9 for the POD and a false call rate of about 8.3% for a flaw of length 0.094 inches.

![Figure 10 POD fits for Automated MOI inspections of surface cracks.](image2)

By implementing the improved classification algorithm which treats interior and edge rivets separately with skewness function $S_e$ for edge rivets the performance improved significantly in terms of lower false calls. For a flaw of 0.094 inches and skewness threshold of 0.210 the POD was 0.9 and false call rate was 1%. The results are comparable to the “field” tests for the MOI 301, but not as good as the MOI 303.
results shown in Figure 9. A lower threshold of 0.15 show comparable results to the MOI 303 manually obtained data, but has higher false call rate of 21%. However with the improved classification algorithm using $S_e$ for edge rivets, the false call rate was reduced to 1%.

The algorithms are currently being further optimized and will be tested on third layer crack data from AANC as well.

**Acknowledgement**

The authors are grateful to Michel Bode and Floyd Spencer, Sandia National Laboratories for providing experimental MOI data from aircraft panels and helping with POD analysis on the results.

This work was supported by funding from the Federal Aviation Administration. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Federal Aviation Administration.

**Reference:**