Fuselage Damage Locator System

Yair Wiseman¹,

¹ Computer Science Department
Ramat-Gan 52900, Israel
wiseman@cs.biu.ac.il

Abstract. Damaged fuselage can cause a fatal catastrophe that will claim numerous lives. This paper suggests the use of a regular digital camera that can generate JPEG images so as to locate the damaged part of the fuselage. The digital camera takes pictures of the fuselage. The signal processing method of JPEG assumes that the taken image is to a certain extent smooth. As a result when there is a sharp change in a particular block of an image, the value of many of its frequency coefficients will be high and it will be compressed into many more bits; therefore, if the image is overly large, an embedded computer system can track down the damage. In that case, sometimes an action can be taken so as to reduce the devastative effects of the damage.

Keywords: Fuselage, Discrete Cosine Transform, JPEG

1 Introduction

At April 1, 2011, a Boeing 737-3H4 flew from Phoenix, Arizona to Sacramento, California. Near Yuma, Arizona, a mysterious sound was recorded on the cockpit area microphone. Few seconds later, the captain noticed that the airplane had rapidly lost cabin pressurization and called for oxygen masks on. Steady sounds with increased wind noise were heard on the cockpit voice recording. The captain announced an emergency with air traffic control. The cabin crew began to search the source of the noises and found a 2-foot hole in the fuselage. The aircraft was diverted to Yuma International Airport. The airplane landed there without further incident. The damaged part of the fuselage can be seen at Figure 1.

The cabin crew in this case eventually indeed succeeded to find the damage part; however it took some time to find the damaged part and in such cases time is very crucial.

We suggest using an ordinary digital camera to locate the damaged part. Nearly all digital cameras can produce JPEG pictures. JPEG is a very common method for image compression and it is also extensively used by electronic devices like scanners and digital cameras [1] as well as vehicle equipment like GPS [2].

JPEG images have many advantages like the ability of being decoded in parallel [3], the straightforwardness of adaptation for new compression methods [4,5] and the capability of flexible implementation for hardware from different vendors [6].
Fig. 1. Dangerous fuselage damage

Images are often stored in a compressed standard. A naive approach for image processing on compressed images would be to decompress the image and then running the image processing algorithm on the original image data. Instead, for some image operations, we can act on the compressed data directly. This gives us two benefits: first, we can use the standard digital cameras without a need to adjust the digital camera; second, the signal processing used by JPEG can let us use the frequency information embedded in the compressed data.

The rest of the paper is organized as follow: Section 2 explains how JPEG can be used for detecting damaged fuselage; section 3 describes the experiments we have performed while section 4 concludes the paper.

2 Fuselage Damage Locator System Using JPEG

JPEG is a well known standardized image compression technique. JPEG loses information, so the decompressed picture is not the same as the original one. By adjusting the compression parameters, the degree of loss can be adjusted. The wide use of JPEG is because of two fundamental reasons: storing full color information and reducing the size of image files, so as to avoid high traffic on the network and avoiding memory pressure [7]. JPEG is an eminent format and is described in many places e.g. [8,9].

The JPEG standard is based on the DCT paradigm [10]. The DCT changes the picture into frequency space. The frequency coefficients, which are very low magnitude, are rounded to zero. When most of the coefficients in a block are zero or very low magnitude: The compression algorithm will give a very short bits sequence for such a block. Zero sequences are treated very efficiently by JPEG compression and the results will be only few bytes.
When there is a drastic change in a block of 8X8, the value of many frequency coefficients will be high. Such a sequence will be compressed into many more bits. JPEG’s standard stipulates that the block’s size will be 8X8 pixels, but the algorithm will be obviously good for other small NXN pixels size too.

When there is a good algorithm that is very common in the market and does a good job, it seems to be the commonsensical choice. There might be some other good algorithms, but using those algorithms means you have to push a new standard which can be a very hard task.

When looking for the contour of an object, the goal will be to locate the object's border. The idea is to break the compressed file into its original blocks, then look in the compressed file for long bit sequences. The blocks which are compressed into long bit sequences, are presumed to be the object's border. In our implementation we took a simpler approach. We take pictures of the fuselage. We actually take a close picture of each part of the fuselage. If the entire size of a picture is above a certain threshold, we will consider this part of the fuselage as a damaged part.

If we have no idea what the threshold value should be, we can examine the probability density function (PDF) of the block representation to select a suitable value. In the uncomplicated case the PDF should be mono-modal and we set the value in the inflection point.

![Sample image and how JPEG can be used for contouring](image)

**Fig. 2.** Sample image and how JPEG can be used for contouring

**Table 1.** JPEG format of the upper left corner of the black square.

<table>
<thead>
<tr>
<th>12</th>
<th>18</th>
<th>-8</th>
<th>-2</th>
<th>3</th>
<th>-1</th>
<th>-1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>-25</td>
<td>8</td>
<td>2</td>
<td>-3</td>
<td>0</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>-18</td>
<td>20</td>
<td>-7</td>
<td>-2</td>
<td>-2</td>
<td>0</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>14</td>
<td>4</td>
<td>1</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>-11</td>
<td>10</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>-5</td>
<td>1</td>
<td>-3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>-2</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 2 demonstrates how JPEG is used for contour extraction. The original image was compressed in grayscale baseline JPEG format with 75% quality. The figure shows the original image, which is a high-resolution picture of 1000X1000
pixels. Table 1 shows the JPEG format in the upper left corner of the black square. The size of the black square is 200X200 pixels and the square is not aligned relative to JPEG’s 8X8 blocks.

The JPEG file reports the difference of magnitude between the DC’s coefficients of a previous block relative to the current block. In the case of a white or black area there are no changes in the coefficients’ magnitudes. This type of block is encoded as six bits by the JPEG standard:

0, 0, 1, 0, 1, 0.

The “00” reflects that there are no differences between the values of the previous and the current block’s DC coefficients, and “1010” symbolizes the end of the block. If there is a difference between the intensity of the DC coefficients of the previous and the current blocks, the size of the encoding block will be slightly larger. For example, a block which encodes a sharp change from white to black is represented by a wide range of frequency coefficients. It is easy to select a threshold that delimits the edges of the shape from the rest of the image [11,12,13].

Figure 2 and Table 1 show a sample of the block, which contains the upper left corner of the black square. In order to compress these values in JPEG standard, 243 bits are needed. The difference between 6 to 243 is obviously significant. By using three parameters the length of the block, its magnitude and the number of consecutive blocks the threshold can extract the contour with a range of scalar values [14,15]. The extra parameters allow more control over the resulting mechanism.

The algorithm is very simple and can be described as follows:

- Take a picture of the fuselage part and create an image - I
  - Set L = \text{size of image (I)}
  - If (L > T)
    - then the fuselage is deemed to be damaged and should be checked.

Where T is a threshold and its value will be discussed below.

3 Experiments

We examined our technique on some damaged fuselages and tried to check whether we succeed to locate the damaged part. Clearly, obvious cases like in Figure 2 do not exist in real fuselages, but we still succeed to find most of the damaged parts by taking many picture of it in all of its parts.

We used an Olympus FE-170 digital camera with resolution of 2112X2816. The Images with no damage in the fuselage produced images in size of less than 1.3MB; whereas images in size of more than 1.5MB usually had damage in the fuselage.

Many of the fuselage damages are clear like the damage in Figure 3. It appears that many wildlife strike aircrafts [16] and in such cases the damage is usually quite large. Since the establishment of the Federal Aviation Administration’s (FAA) National Wildlife Strike Database at 1990, 99,411 reported wildlife strikes to airplanes have been reported. The result of these strikes causes more than 200 human lives lost worldwide as well as financial losses (direct and indirect) of at least $1.2 billion
annually to civil aviation worldwide. Particularly, more than $625 million of financial losses annually just in the United States [17].

Fig 3. Large and clear damage

However, sometimes there are smaller damages. One major reason for smaller damages is hail [18]. At July 14, 2011, More than 100 flights have been called off at Denver International Airport after hail damaged about 40 aircrafts and stranded about 1,000 passengers overnight. The damages were usually not large as can been seen in Figure 4.

Fig 4. Hail Damage in Denver International Airport

There also some other small objects that sometime soar over airports and can cause tiny damages like the damage in Figure 5.
Actually, there are seasons that there more such accidents [19]. Usually the months of Aug-Oct have the highest priority of having such damages as can been seen in Figure 6.

![Fig. 5. Small damage](image)

<table>
<thead>
<tr>
<th></th>
<th>Nov-Jan</th>
<th>Feb-Apr</th>
<th>May-Jul</th>
<th>Aug-Oct</th>
</tr>
</thead>
</table>

**Fig. 6.** Strike's distribution during a year

As a matter of fact, the use of our system for large and clear damages is less important, although sometimes a cabin crew member under stress can need a help even such cases; however when a small and indistinguishable damage can make lower the pressure in the cabin and locating it without delay can be an imperative.

There are also cases where our system has produced false alarms. Figure 7 shows an example for such a false alarm. The dirt on the fuselage in Figure 7 can be easily noticed. This dirt made the system assuming that the fuselage is damaged; whereas this fuselage was actually undamaged.

The quality factor of JPEG is used to produce two quantization tables - one for the luminance (brightness) information and the other for the chrominance (color) information. Applications that support IJG (Independent JPEG Group) produce quantization tables according to the IJG specification; however, many other
applications employ different quantization tables. At this paper we have used IJG quantization tables. Actually, in our system, we obviously preferred the highest quality factor (100%), so as to get the best distinguishableness, which means in IJG that the entire quantization table is filled with 1s i.e. the frequency coefficients are not divided.

Fig 7. Dirt on the Fuselage

4 Conclusions and Future Work

Fuselage is very risky and can cause too many loss of life. Automatic inspection of aircraft fuselage is critical and vital as was noted in [20]. The suggested system in this paper can be mounted in an aircraft and it may save lives. In this paper, we explained how a system using simple equipment – digital camera can detect damages in a fuselage. In the future we would like to adapt our system to a ground camera. Conventional digital cameras cannot take a picture of so fast moving object and so far objects, so we should find a practical way how the pictures can be taken.

We also consider developing more sophisticated algorithm that can not only identify damaged fuselage, but also identify various kinds of damages. Looking at existing machine-vision [21,22,23] and machine-learning algorithms [24,25] may help in significantly extending the suggested method.

References

2. P. Hongyan, H. Hong, J. Hengtian, "Drive design for ship GPS navigation equipment based on Linux operating system", International Conference on Educational and Network Technology (ICENT), pp. 384-388, Qinhuangdao, China, June, 2010.