KNOWLEDGE BASED GEOMETRIC OBJECT RECOGNITION

R. J. Morris (1), K. V. Mardia (1), C. C. Taylor (1), J. D. Burrows (2)

(1) University of Leeds, U.K., (2) PLSD, Fort Halstead, UK

INTRODUCTION

The task of recognising rigid objects in an image can be greatly eased by exploiting particular geometric features of the objects. Rather than trying to match the whole object we can just match these particular features. A considerable speed advantage can be obtained over techniques like template matching, Burrows et al (1),as there are efficient algorithms for detecting various geometric primitives. This approach can also cope with small differences in the actual composition of the objects as a geometric description can ignore these differences. Similarly the algorithm can be made robust with respect to clutter and occlusion/obscurement problems.

Here, we will concentrate on detecting detonators in two channel X-ray images. A selection of dummy detonators is shown in figure 1. We will exploit two particular geometric features:

1. Each detonator contains at least one dark object. We shall call these blobs *cold-spots.*
2. Each detonator is approximately a cylinder which we can represent as two parallel edges lying on either side of the cold-spot.

The algorithm works by first identifying cold-spots, which give a possible location but not an orientation for the object. The second stage identifies pairs of parallel edges, which give an orientation. The first stage can be carried out fairly quickly but generates a large number of false hits. Each test in the second stage is quite slow but as we are only interested in parallel lines near the cold-spots not many tests need to be carried out and the whole stage can be completed quite quickly. A third, much slower, stage can then be used to eliminate the remaining false hits. For example using template matching with a library of sample detonators.

The algorithm represents a hierarchical method of object recognition: first easy to find features, like cold spots, are identified. Then more discriminating features are used to refine the results.

We will also look at the use of Hough transforms for detecting suitcase edges. The two channel X-ray data has some specific characteristics and we look at how these can be be exploited.

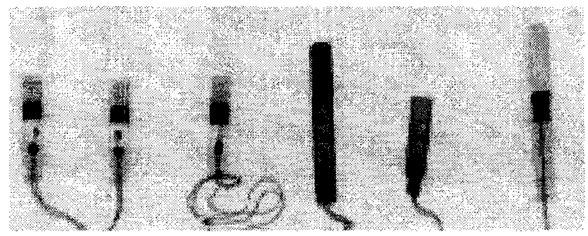


Figure 1: A selection of dummy detonators

1 FINDING COLD-SPOTS

The first stage of the algorithm is finding candidate cold-spots in the image. Several techniques are available for this stage.

- Convolving an image with a specific mask
- Identifying local minima
- Simple thresholding
- Using morphological operations
- Using local correlations between the two channels of the image

In the previous work (1), cold-spots were found by convolving with four different nine by nine masks, which is fairly computationally expensive. Here, we directly find the local minima in one of the channels of the image, by comparing each pixel in the image with all the pixels in a surrounding neighbourhood. The size of the neighbourhood is quite important; small neighbourhoods will be fast but yield many false hits whereas large ones take much longer to calculate and may miss some of the actual detonators. We have used a fairly large (21 by 21) neighbourhood. For the local minima test to fail we only need to find one pixel in the neighbourhood with a lower value. Hence the test can be completed efficiently taking on average 35 tests per pixel using a 21 by 21 neighbourhood.

Once found we can reduce the number of cold-spots by thresholding on one of the channels, only accepting cold-spots which are dark enough. To avoid the chance of eliminating true hits we use a high threshold here.

The *black top-hat* morphological operator removes large dense objects like batteries and preserves small and thin object, such as the detonators. We can threshold on the results of this operator to further reduce the number of cold-spots.

1.1 Local Correlations

Another technique available to us is to exploit the two channel nature of the data. Two different energies of X-rays are used and different materials absorb these at different rates which gives us two images, *back* and *front*. We can construct a third image by taking the difference between these. There are also standard techniques to combine the back and difference information to give a rough approximation of the atomic numbers of each pixel. We felt that this atomic number information was not sufficiently discriminating to be useful at this stage.

We can combine these four images by constructing *local correlations*. Let x_{ij}, y_{ij} be two channels of data. For each pixel we choose a neighbourhood and calculate the correlations between the channels:

$$\gamma_{kl} = \frac{\sum(x_{ij} - \overline{x})(y_{ij} - \overline{y})}{\sqrt{\sum(x_{ij} - \overline{x})^2 \sum(y_{ij} - \overline{y})^2}}.$$

This gives a number between -1 and 1 for each pixel. Here sums are taken over the neighbourhood of the pixel (k, l), and $\overline{x}$ and $\overline{y}$ are the two means calculated over the neighbourhood.

If we calculate the local correlation between the back and the difference we observe a couple of features:

1. The edges of the detonators and wires seem to have a strong negative correlation, this could be a useful criteria for refining the results of the cylinder finding algorithm.
2. The cold-spots in the detonators have a high positive correlation.

By thresholding on this local correlation we can exploit the second of these to further reduce the number of cold-spots found.

2 FINDING PARALLEL EDGES

The second feature we will use to detect detonators is the fact that they can be approximated by cylinders of a given width. The algorithm we use consists of a number of stages

(a) Use a logarithm based correction factor.

(b) Calculate the gradient of the image.

(c) Identifying pairs of parallel lines in the gradient image which lie on either side of a cold-spot.

2.1 Logarithmic Corrections

The intensity of X-rays received after passing through an object depends on an exponential decay law.

$$I = I_0 e^{-\mu x}$$

where I_0 is the intensity of the source, I is the intensity at the detector, μ is some constant depending on the type of material and the energy of the X-rays and x is the thickness of the material. See Knoll (3) and Wilks (4) ch. 30,for more details on the properties of X-rays. Experimentation on test objects suggests that the actual pixel values depend linearly on the intensities.

The X-rays also obey a product rule when they pass through two objects. Let A be a detonator and B be some other object, with attenuation factors f_A and f_B. The intensities after passing through A or B are $I_A = f_A I_0$ and $I_B = f_B I_0$. If the X-rays pass through A *and* B then the final intensity will be $I_{AB} = f_A f_B I_0$.

When considering gradients we are essentially concerned with the differences between neighbouring pixels. If we just calculate the gradient of the raw data we would expect a greater response across the edges of a detonator when it lies on a light background (high f_B) than if it lies on a dark background (low f_B). Ideally we would like to have and edge detector which does not depend on the material the detonator lies upon. This can be achieved by taking logarithms which convert the product rule into a sum rule. We have

$$\log(I_{AB}) = \log(f_A) + \log(f_B) + \log(I_0).$$

In practice we can not directly use logarithms as some pixels will have a zero intensity giving a logarithm of minus infinity. For simplicity lets assume our data lies between 0 and 1. If we use a correction factor of the form

$$b_i' = \frac{\log(1 + \alpha b_i)}{\log(1 + \alpha)}$$

where b_i, b_i' are the original and corrected pixel values and α is some constant then this always maps $[0, 1]$ onto $[0, 1]$. For α slightly greater than 0 this is almost the identity map and for large α we get closer to a true logarithmic function (with appropriate scaling). There is a certain trade-off with the value we choose for α, higher values are closer to true logarithms but they will tend to lighten the whole image reducing detail for light objects.

Once we have used the correction factor we then calculate gradients in the x and y directions using the standard Sobel mask. This gives two values g_x and g_y for each pixel which we can combine these to give a magnitude of the gradient and also the direction of greatest change.

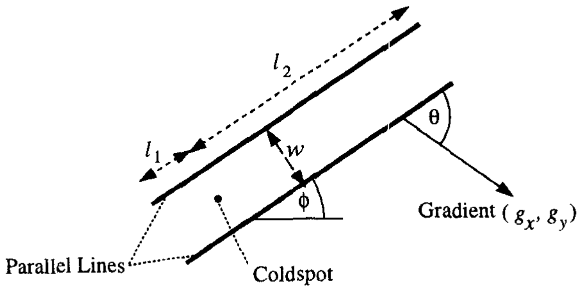


Figure 2: Setup for parallel lines test.

2.2 Matching Lines

Consider two parallel lines of length $l_1 + l_2$ and separation w lying on either side of a cold-spot and at orientation ϕ as shown in figure 2. The values l_1, l_2, w can be calculated by examining sample detonators. Now count up all the points of the lines which satisfy two conditions on the gradient:

1. The magnitude is greater than some tolerance T_m.
2. The gradient points outwards and is approximately at right angle to the lines, i.e. $|\theta - 90°| < T_\theta$.

If the number of passes on each line is sufficiently large then the pair of lines is taken to be a hit. To locate all the detonators we repeated this test for each cold-spot in the image, and for varying angles ϕ.

Figure 3 shows the result of a typical run of the algorithm on a small portion of a suitcase. This has successfully located the detonator even though it lies across another object. Using a counting technique allows for the discontinuities in the edges and the logarithmic correction helps compensate for the difference in gradient response.

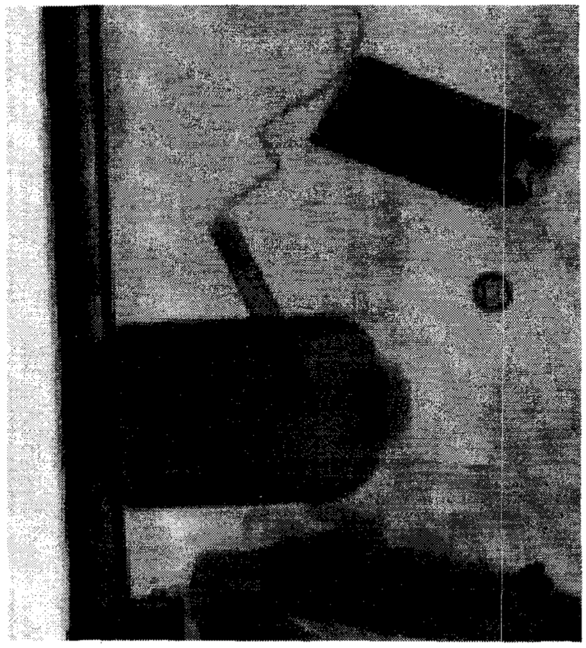
3 HOUGH TRANSFORMS

The edges of a suitcase can yield a number of false hits with the above algorithm. We can eliminate some of these by using a *foot of normal Hough transform* Davis (2),to identify the edges of the suitcases.

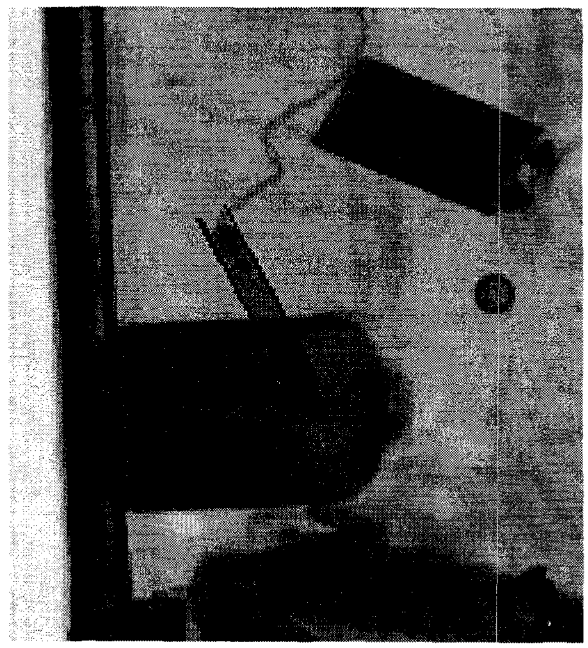
Consider a point (x, y) with gradient (g_x, g_y) as shown in figure 4. This defines a line orthogonal to the gradient l. The point, (x', y'), on the line which is closest to the origin is called the *foot of the normal.* We can calculate this point quite easily:

$$(x', y') = v(g_x, g_y)$$

where $v = (g_x, g_y) \cdot (x, y)$.



(a)



(b)

Figure 3: A detonator lying across another object (a), and the results of the algorithm (b) showing a cold-spot and a pair of matching edges.

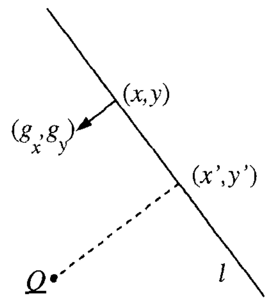


Figure 4: Calculating the foot of the normal

Figure 5: Lines detected using the foot of normal Hough transform

To identify edges in the image we calculate the foot-of-normal for each point in the image where the gradient is greater than some given threshold. Here we use the centre of the image as the origin. Each such point adds one vote to an accumulator representing possible foot-of-normals. Points in the accumulator space with a large number of votes corresponds to long lines in the image.

Figure 5 shows edges detected in this way for a typical suitcase. We have drawn one line in the image for each point in the accumulator space with a sufficiently high number of votes. Once we have identified the suitcase edges we can eliminate the false hits which lie on the edges. However, there is a danger in missing detonators which are positioned near the edges.

4 CONCLUSIONS

The above algorithm has several advantages over a template matching approach. It is considerably faster, on a sample 545 by 760 pixel image it took 35 seconds running on 200MHz Silicon Graphics Indy workstation. It is also more robust and can cope with situations where the detonators are obscured.

Improvements could be made in eliminating the false hits and in coping with a wider variety of detonators.

Acknowledgements

We gratefully acknowledge funding for this project from the Department of Transport.

References

[1] Burrows J.D., Little J.A., Mardia K.V., Taylor C.C., Walder A.N., 1995, "Statistical Approaches to Image Restoration", Proc. ECOS 95, IEE conf. pub. 408, pp 59–63.

[2] Davis E.R. 1990, "Machine Vision: Theory, Algorithms, Practicalities", Academic Press.

[3] Knoll G.E., "Radiation Detection and Measurement (2nd edition)", Wiley, 1989.

[4] Wilks R.J., 1987, "Principals of Radiological Physics", Churchill Livingstone, Edinburgh.