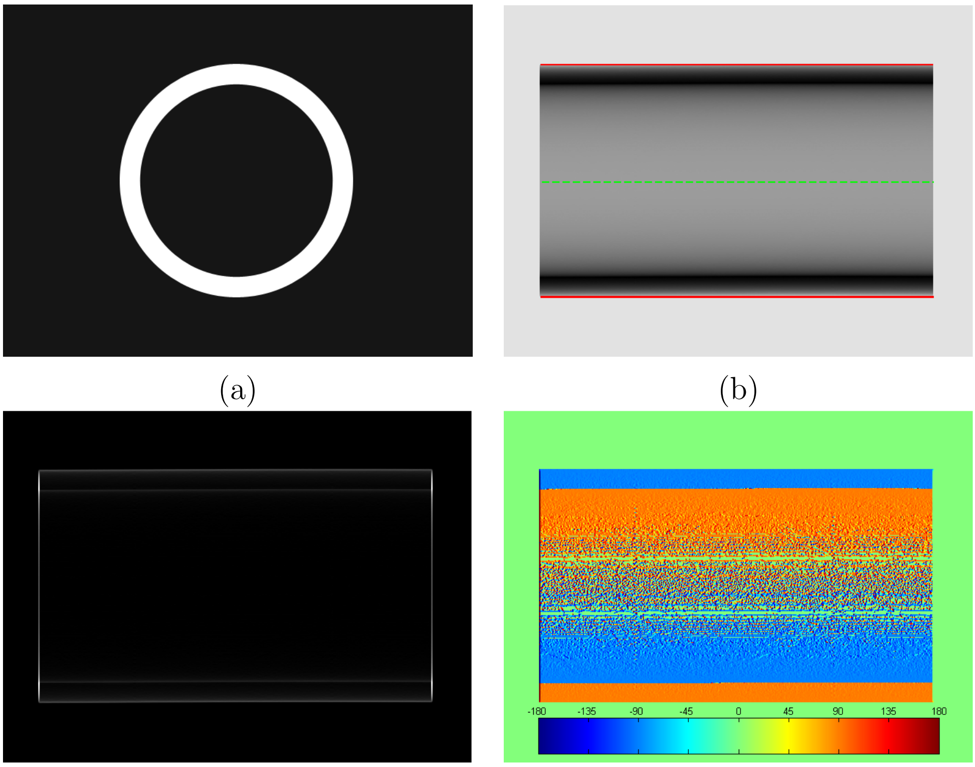
# Chapter 7

# Fracture detection using gradient analysis

In Section 2.5.1 on page 25, long-bone diaphyses were identified as the location in which fractures should be detected. Chapter 6 demonstrated that the location of the diametaphyses, and therefore the diaphysis, could be accurately identified using a segmentation method that utilised the bone curvature. In this chosen segmentation method, the diametaphyses were located by calculating the average points at which the bone deviated away from the straight line approximations determined in Chapter 5. This was equivalent to marking the diametaphyses as the points at which the centre-lines ended. The result of the segmentation was a mask that could be placed over the image to retain only those areas that corresponded to the diaphysis.

Once the diaphysis is identified, a fracture detection algorithm can be applied to the unmasked region, to determine if there are any fractures present within the long-bone shaft. The fracture detection process is split into two parts. The first involves extraction of the required features from the image, and the second involves a decision process to determine if the identified features constitute a fracture. The algorithm that was created to detect these fractures is outlined in this chapter.



(a)

(b)

(c)

(d)

*Figure 7.1: (a) The cross section of a long cylindrical tube used to approximate a portion of a human long-bone, and (b) the corresponding x-ray of the tube, showing the centre-line (green dashed) and bone edges (red dashed). (c) The magnitude of the gradient shows that the only significant gradients are present at the walls of the bone, and that the magnitude of the gradient does not vary along the length of the bone. (d) The gradient direction is constant at the bone edges at ±90°, but contains some random changes within the bone.*

## 7.1 Assumptions

In section 2.1.1 on page 6 it was explained that long-bone diaphyses consist almost exclusively of uniform cortical bone. Figure 7.1a shows the cross section of a cylindrical tube used to approximate the cortical bone present in the shaft of a human long-bone. The corresponding x-ray of this segment of tube is shown in Figure 7.1b. Disregarding the absence of the anatomical variations in the wall width and texture that are present within a real long-bone, the approximation is very good, and can be used to explain the assumptions used in detecting long-bone fractures.

X-rays of a typical diaphysis containing no major pathology should exhibit gradient changes in the direction normal to the bone centre-line, due to the interface between the bone and soft tissue (i. e. at the periosteum—the edge that is approximated in Chapter 5), and the variation in cortical wall thickness through which the x-rays must pass. This cortical thickness change produces the visible vertical intensity gradient across the image in Figure 7.1b. In contrast, the cortical wall thickness and density change relatively slowly along the length of the long-bone, so the gradient parallel to the centre-line is relatively small. Accordingly, at any point along the length of the diaphysis, the direction of the gradient at the cortical walls is orthogonal to the centre-line. In Figure 7.1c there is no gradient change along the length of the bone because the image is artificial; but in a real long-bone x-ray image there would be some small changes, but these should not be as large as those orthogonal to the centre-line. Figure 7.1d shows that within the bone, the relatively isotropic image intensity produces no significant gradient magnitude changes, so the gradient direction appears to change somewhat randomly.

This means that any edges that produce relatively large gradients that occur at angles that are not orthogonal to the centre-line are indicative of some type of abnormality. The fracture detection algorithm in this chapter was designed to detect and then mark these abnormalities.

### 7.1.1 The basis for detecting fractures

In Chapter 5, the modified Hough Transform and peak detection methods were used to calculate line parameters that approximated the long-bone shaft. Chapter 6 then showed how the image could be segmented to retain only the diaphyseal segment of the bone. From this information, the particular gradients (both magnitude and direction) that should be present in the image could be calculated. Modifying the image to remove all of these normal gradients left behind only the abnormal gradients, such as the ones that belonged to any fractures or other pathologies. To remove all normal regions from the image, a tool called the gradient composite measure (GCM) was developed.

## 7.2 The gradient composite measure

Previously, in Section 7.1, it was suggested that some edges within an image of a fractured long-bone would be identified as being abnormal, and may correspond to fractures. In addition, these edges could be identified on the basis of their gradient. In order to remove the normal regions from the image, both the magnitude and the direction of the gradient at each image pixel had to be taken into account. This was achieved by utilising a combined measure of the magnitude $|\bigtriangledown I(x,y,t)|$ and direction $\phi(x,y,t)$ of the gradient of the smoothed image $I(x,y,t)$ at scale $t$, termed the gradient composite measure. Here the scale was critical, and needed to be suitably fine for small feature analysis, since too large a scale would result in subtle fractures disappearing. As discussed in Section 4.6.1 on page 74, the scale chosen for this small feature analysis was $t_1 = 5$.

The gradient composite measure $C(x,y,t,\rho,\theta,p)$ was calculated using the magnitude of the gradient, and two scaling factors. The two scaling factors were used to incorporate the direction of the gradient $\phi(x,y,t)$, along with the angle and distance information from the long-bone shaft approximation parameters calculated in Chapter 5. The first scaling factor was called the importance rank $R(x,y,t,\theta,p)$, and was a measure of how well the direction of the gradient matches the angle $\theta$ of the approximation lines, at any given pixel within the image. The second scaling factor was called the distance rank $D(x,y,\rho,\theta,p)$, and was a measure of how close any given pixel within the image was to all of the approximation lines. That is, it represented how well each pixel in the image matched the $\rho$ parameters of the approximation lines. The gradient composite measure was the product of the magnitude of the gradient, and the two scaling factors:

$$C(x,y,t,\rho,\theta,p) = |\bigtriangledown I(x,y,t)|\, R(x,y,t,\theta,p)\, D(x,y,\rho,\theta,p) \tag{7.1}$$

Therefore to calculate the gradient composite measure of an image at a particular transform angle, it was first necessary to determine the two scaling factors.

### 7.2.1 The importance rank

The magnitude data was already ranked in terms of importance and could be easily normalised to the range $[0,1]$, such that values close to 1 indicated the presence of a large intensity change in that region. However, the direction data was in the range $[0,360°]$ and had to be ranked in terms of which angle was most important, since large angles did not necessarily correspond to more important regions. As a result, a transform was required to convert the gradient direction data from its raw form, so that every pixel was given an importance rank $R(x,y,t,\alpha,p)$ that was also in the range $[0,1]$. The importance rank was based on the direction of the gradient $\phi(x,y,t)$ (calculated using Equation 5.5 on page 91) at that pixel $(x,y)$, the scale $t$, the chosen transform angle $\alpha$, and an importance weighting coefficient $p$.

The importance rank was calculated using a sinusoid of period 180° that was translated and scaled so that the transform angles $\alpha$ and $\alpha \pm 180°$ were assigned the minimum value 0, while orthogonal angles $\alpha \pm 90°$ were assigned the maximum value 1 . To achieve this, the sinusoid used was a standard *cosine* function, that had a period of 180° rather than 360° and was translated horizontally by $\alpha$. In addition, it had one half the amplitude and was vertically offset by one half, so that all values were in the range $[0,1]$. A power relationship was also applied so that higher powers $p$ decreased the width of the trough, thereby increasing the importance of angles close to the chosen $\alpha$, and decreasing the importance of angles further from the chosen $\alpha$:

$$\begin{aligned} R(x,y,t,\alpha,p) &= 1-\left[\frac{1+\cos\left(2\left(\phi(x,y,t)-\alpha\right)\right)}{2}\right]^{p} \\ &= 1-\left[\cos^{2}\left(\phi(x,y,t)-\alpha\right)\right]^{p} \\ &= 1-\left[\cos\left(\phi(x,y,t)-\alpha\right)\right]^{2p} \end{aligned} \tag{7.2}$$

As a result, pixels in $I(x,y,t)$ where the direction of the gradient was close to $\alpha$ had a small importance rank, while those angles orthogonal to $\alpha$ had a large importance rank. Equation 7.2 produced better results than other methods such as a linear importance rank, or a hard binary threshold where only the angles $\alpha \pm \mathit{threshold}$ were cleared. Comparing these methods showed that it was better to use a smooth function that decreased the importance of angles close to $\alpha$ but still allowed them to contribute to the final output, rather than simply cancelling them so that they had no effect. It was possible for a pixel having an angle just outside a chosen hard threshold to be important, especially if it had a very large gradient, yet this pixel would be excluded by the thresholding process. When utilising Equation 7.2 that pixel could still contribute to the output, albeit at a reduced intensity due to its lower importance rank.

The importance rank for the peak $\alpha = 94.5°$, with a range of powers $p$ is shown in Figure 7.2. This plot illustrated that any pixel in $I(x,y,t)$ where the direction of the gradient $\phi(x,y,t)$ was close to 90° would have a low importance rank, and a correspondingly small composite measure, regardless of the gradient magnitude $|\bigtriangledown I(x,y,t)|$ at that point. However, a pixel in $I(x,y,t)$ where the direction of the gradient was close to the (arbitrarily chosen) angle 124° would retain 75% of its intensity in the composite measure, since it was further from $\alpha$. The importance rank was also shown for all integer powers in the range $p = [1,10]$. Higher powers reduced the trough width and made the importance rank measure more specific for the chosen angle, while lower powers increased the trough width and made the composite measure exclude a greater range of angles.

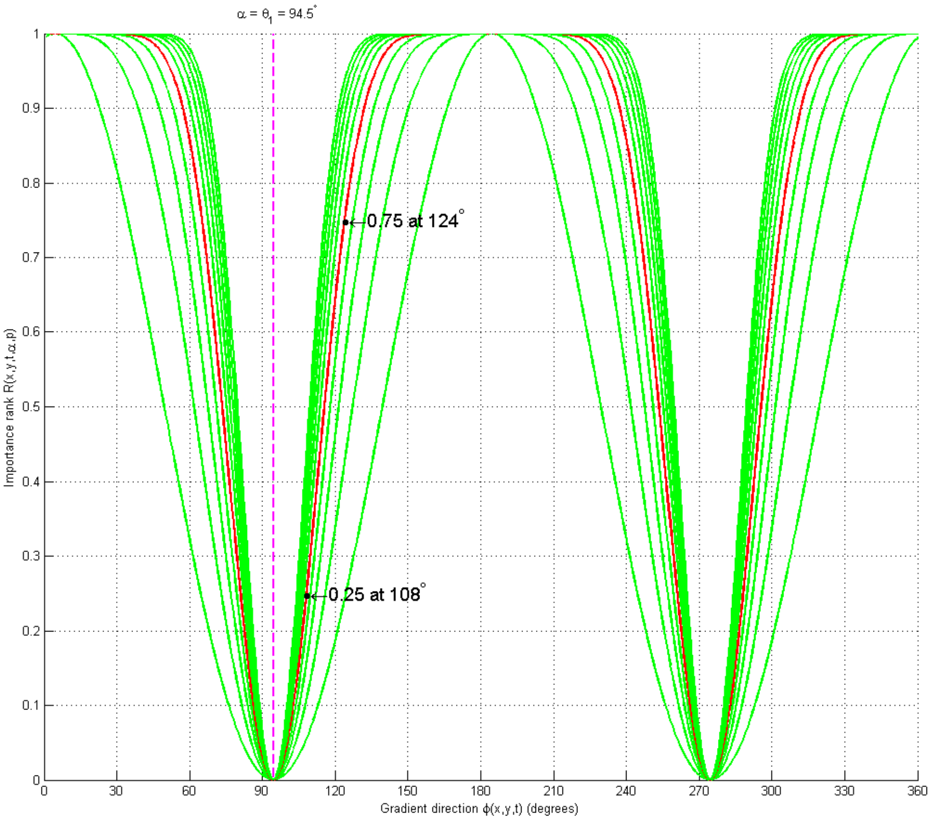


*Figure 7.2: The importance rank $R(x,y,t,\alpha,p)$ for the detected peak $\alpha = \theta_1 = 94.5°$ (from Figure 5.7 on page 104) identified by the pink dashed line. A range of powers $p = [1,10]$ are displayed, with the chosen power $p = 5$ shown in red.*

### The combined importance rank

In almost all cases, the importance rank would need to be calculated for more than one transform angle. This was because in many cases the segments of a fractured long-bone had different $\theta$ values, as demonstrated in the Hough Transforms shown in this thesis. The number of transform angles at which the importance rank had to be calculated was equivalent to $e$, the number of peaks in the modified Hough accumulator to be detected by the ranked sums method. Fractures and other abnormalities were located by setting the transform angle $\alpha$ equal to the angle parameter $\theta_i$ of each of the approximating lines. To calculate the combined importance rank for a particular pixel, it was first necessary to calculate the individual importance ranks for each of the $e$ transform angles. The combined importance rank was then simply the product of all the individual importance ranks. Equation 7.2 was modified to reflect this change, and became:

$$R(x,y,t,\theta,p) = \prod_{i=1}^{e} \left(1 - [\cos(\phi(x,y,t) - \theta_i)]^{2p}\right) \tag{7.3}$$

An example of the combined importance rank for two arbitrary transform angles $\theta_1 = 67°$ and $\theta_2 = 100°$with the power $p = 5$ is shown in Figure 7.3. In this plot there were multiple troughs, corresponding to the two chosen transform angles.

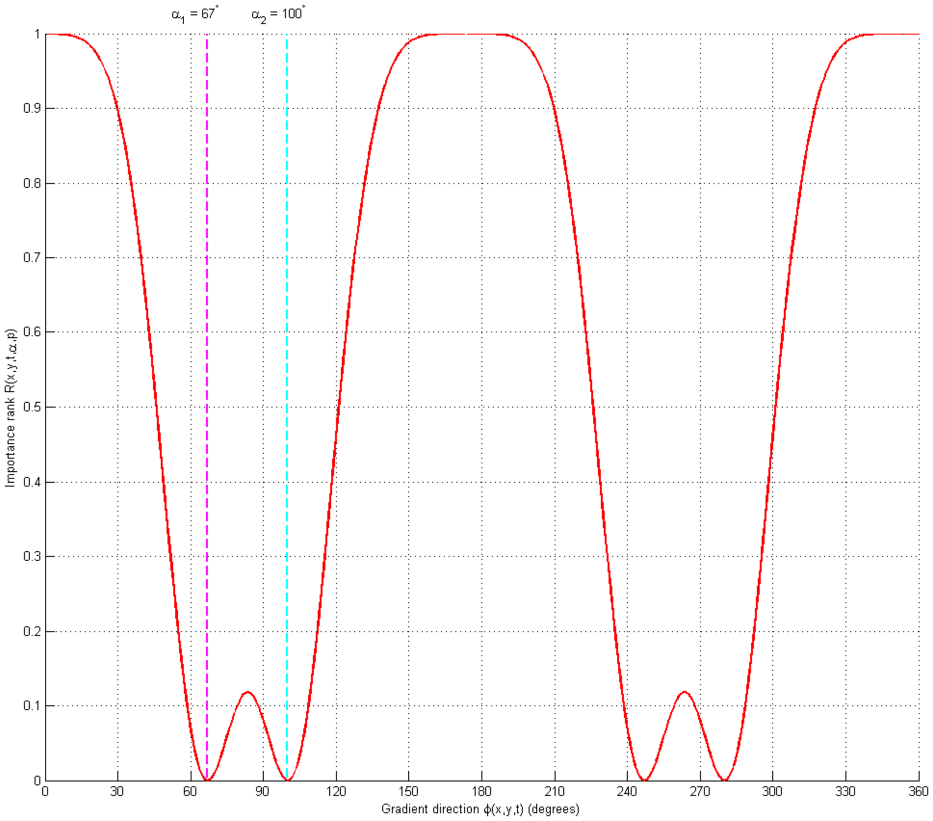


*Figure 7.3: The combined importance rank $R(x,y,t,\alpha,p)$ for the two arbitrary transform angles $\alpha_1 = 67°$ and $\alpha_2 = 100°$, identified by the pink and blue dashed lines, respectively.*

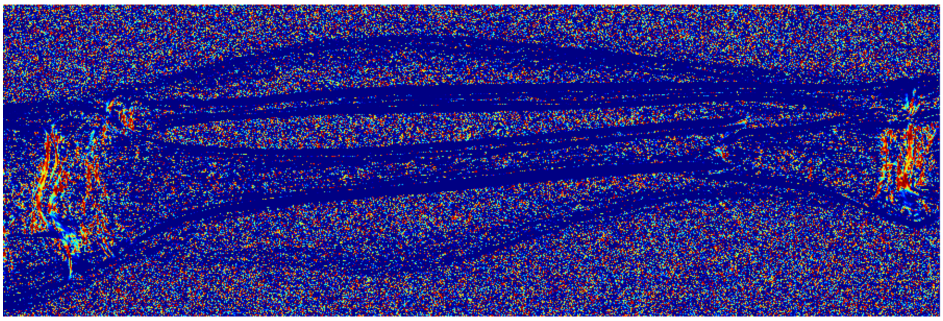


*Figure 7.4: The importance rank for the image shown in Figure 6.18d (with no segmentation applied here), calculated using Equation 7.3. Red regions are the most important, and blue regions are the least important.*

Testing on the six development images showed that although the importance rank $R(x,y,t,\alpha,p)$ was not overly sensitive to the value of $p$ provided $p \geq 4$, a power of $p = 5$ produced the best measure of importance ($R = 0.9$ occured at $\approx \theta_i \pm 37°$ and $R = 0.5$ occured at $\approx \theta_i \pm 21°$). Smaller values such as $p = 2$ allowed angles a long way from the peak to make too large a contribution ($R = 0.9$ occured at $\approx \theta_i \pm 55°$ and $R = 0.5$ occured at $\approx \theta_i \pm 33°$), and therefore many insignificant regions of the output image were classified as being important. While not as dramatic, greatly increasing $p$ (i. e. $p > 10$) caused significant areas to be removed. The chosen value of $p = 5$ is shown in red in Figure 7.2.

An example of the combined importance rank for the segmented development set image shown in Figure 6.18d is shown in Figure 7.4. In this image, the fracture, joints and epiphyseal plates were all marked as important. In contrast, the bone edges were not marked as important, since they had been removed from the image by the ranking process.

### 7.2.2 The distance rank

The combined importance rank calculated using Equation 7.3 utilised only one of the long-bone shaft approximation parameters calculated in Chapter 5. Only the information about the angle $\theta$ of the approximation lines was used, while the location $\rho$ of the lines was unused. A consequence of using only the angle parameter was that any region a long distance from a line with the parameters $(\rho_i, \theta_i)$ could be ranked highly if it matched the angle parameter $\theta_i$, despite the distance parameter $\rho_i$ not matching.

To prevent this from happening, another measure called the distance rank was used.

The distance rank was a measure of how far a point in the image was from all of the long-bone shaft approximation lines. For a single line with parameters $(\rho, \theta)$ the distance rank for a point $(x, y)$ was simply the length of the normal between that point and the approximation line. A power relationship $p$ was again applied, so that regions close to the line were given a much higher importance than those further from the line. The distance rank was calculated for each point in the image using:

$$D(x, y, \rho, \theta, p) = |\rho - [(x - x_{origin})\cos\theta + (y_{origin} - y)\sin\theta]|^{\frac{1}{p}} \tag{7.4}$$

where the equation is calculating the distance between the line with parameters $(\rho, \theta)$ and the line parallel to it passing through the point $(x, y)$.

### The combined distance rank

Like the importance rank, it was necessary to calculate the distance rank for more than one value of $\rho$. Again, the number of lines for which the distance rank had to be calculated was equivalent to $e$, the number of peaks in the modified Hough accumulator to be detected by the ranked sums method. To calculate the combined distance rank for a particular pixel, it was first necessary to calculate the individual distance ranks for each of the $e$ parameter pairs. The combined distance rank was then the product of all the individual distance ranks. Equation 7.4 was modified to reflect this change, and became:

$$D(x, y, \rho, \theta, p) = \prod_{i=1}^{e} \left( |\rho_i - [(x - x_{origin})\cos\theta_i + (y_{origin} - y)\sin\theta_i]|^{\frac{1}{p}} \right) \tag{7.5}$$

Altering the power $p$ changed the rate at which the distance rank decreased further away from the approximation lines. Testing on the six development images showed that again $p = 5$ was the best choice. Lower powers tended to rank too much of the image as being important, thereby reducing the effectiveness of the distance rank as a scaling factor. On the other hand, higher powers tended to have the opposite effect, and resulted in the regions within the bone—where fractures are to be detected—being ranked as unimportant. Figure 7.5a shows an example of the combined distance rank for the image shown in Figure 6.18d. The red points had the highest rank, while the blue regions had the lowest rank. Figure 7.5b contains the cross section along the black line in 7.5a, and shows how the rank varied with the distance from the lines.

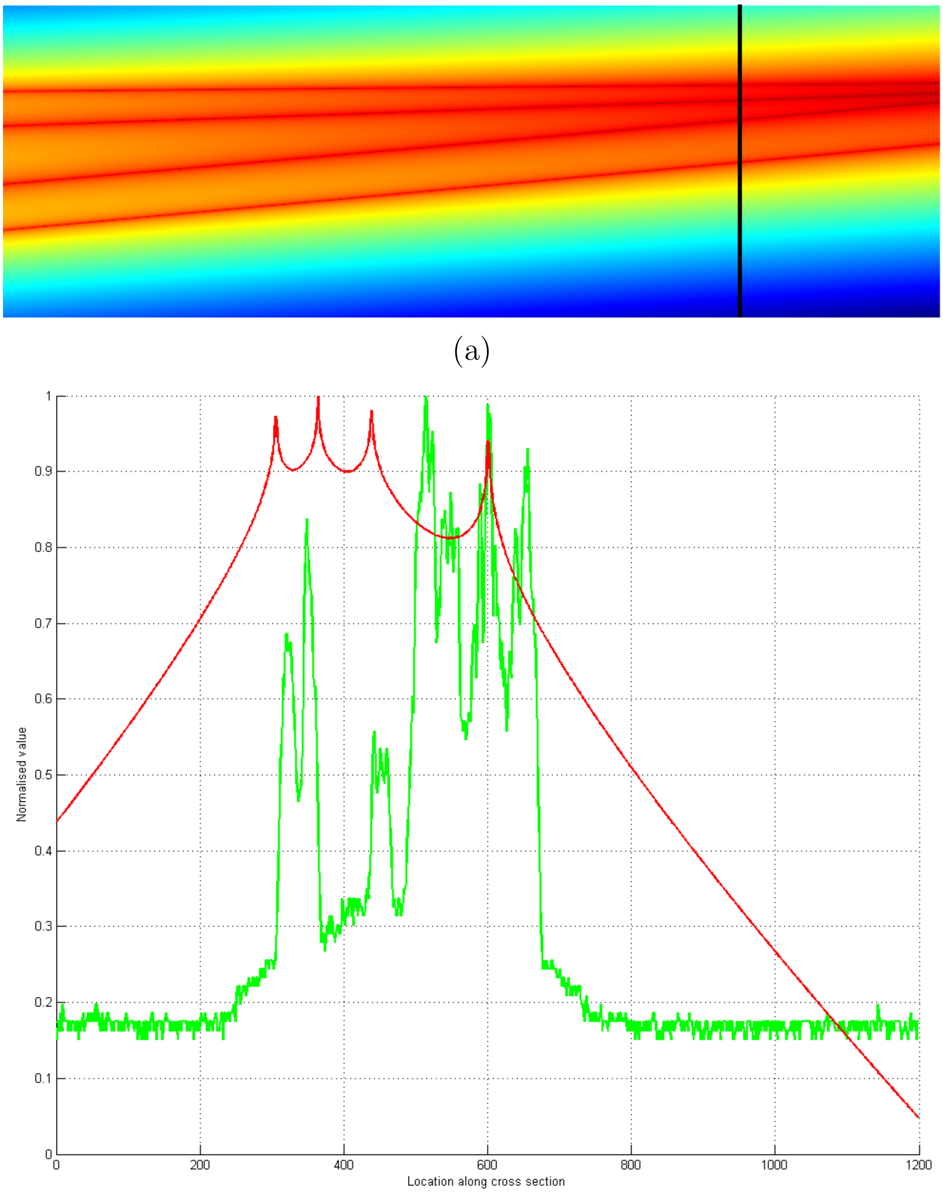
### 7.2.3 The combined rank

The product of the two scaling factors calculated in the previous sections was referred to as a combined rank. Figure 7.6a illustrates the resulting combined rank for the same example development set image shown in Figure 6.18. This time the segmented image was used, so that only the diaphysis was shown. The most important regions were again red, while the least important regions were dark blue. The gradient composite measure for this image, calculated using Equation 7.1, is shown in Figure 7.6b. It was clear that in this image, the only remaining region of high intensity corresponded to the fracture. Since the spatial resolution of the GCM was much higher than the Gabor orientation maps utilised by Yap, et al. [110] for the detection of gross neck of femur fractures (shown in Figure 3.2 on page 37), it was much more likely to be able to detect the texture changes produced by subtle long-bone fractures.

The calculation speed of the gradient composite measure was not sufficiently slow to be of concern. However, if required, both the importance rank and distance rank could also be implemented using a look up table to decrease the calculation time that was associated with both the cosine and power functions.

## 7.3 Fracture identification

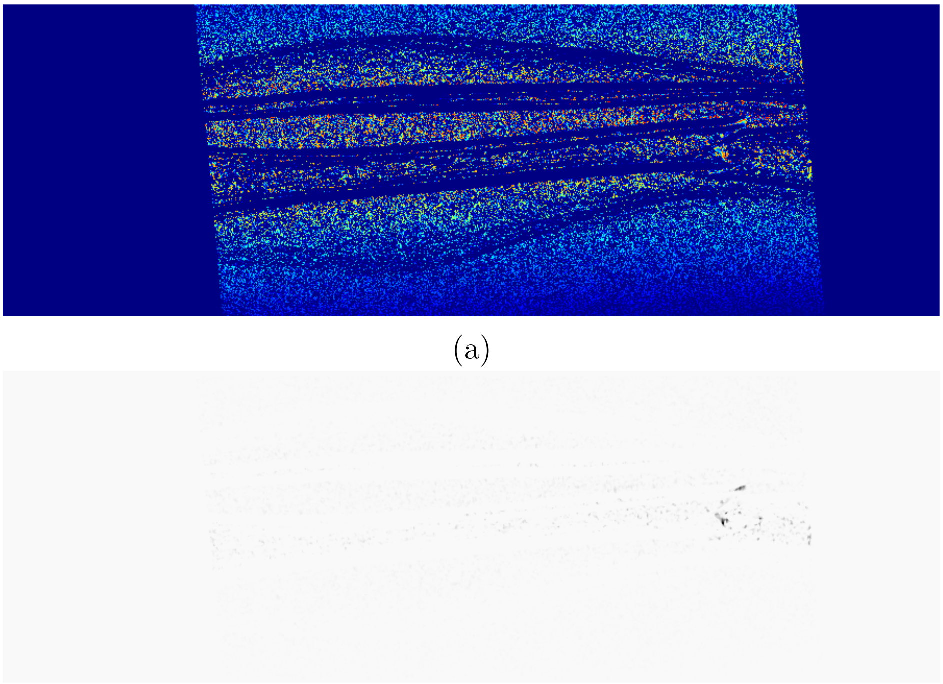
Using the gradient composite measure, the magnitude of the gradient was artificially lowered in the regions where the composite values were small, but remained high in all other regions. Accordingly, by applying the GCM and ranking equations (7.1, 7.3 and 7.5) for each of the peaks detected by the ranked sums method, those regions a long distance from the approximation lines and those regions where $\theta$ matched $\phi(x,y,t)$ were removed. Since the areas containing straight sections deemed to be normal were removed from the image, any remaining locations where the gradient was large were



(a)

(b)

*Figure 7.5: (a) The distance rank for the image shown in Figure 6.18d (with no segmentation applied) calculated using Equation 7.5. (b) The cross section of both the distance rank (red) and the original image (green) along the black line.*



(a)

(b)

*Figure 7.6: (a) The combined rank is the product of the importance rank and the distance rank shown in Figures 7.4 and 7.5, respectively. Here the mask from the diaphysis segmentation was also applied. (b) The gradient composite measure, in which the high intensity regions corresponded to the fracture.*

classified as being abnormal. As shown in Figure 7.6b, the GCM did indeed indicate the location of the fracture, but some further processing was required to make better use of the information that the image contained. Two methods of presenting the location of the fractures contained within the image were examined, and are outlined below.

### 7.3.1 Artificial colouring for fracture identification

The first presentation method examined, simply presented the information in the gradient composite measure to the user by artificially colouring the long-bone x-ray image. The image was adjusted so that all normal areas remained in greyscale, while all abnormal areas—those identified in the GCM—were coloured in a user adjustable colour (red, in this case). The intensity of the colouring was used to show the magnitude of the GCM at that point, and therefore the possibility of that point being abnormal. An example of the artificial colouring is shown in Figure 7.7.

While this method highlighted any abnormalities—including fractures—in the test images, it suffered from a number of drawbacks:

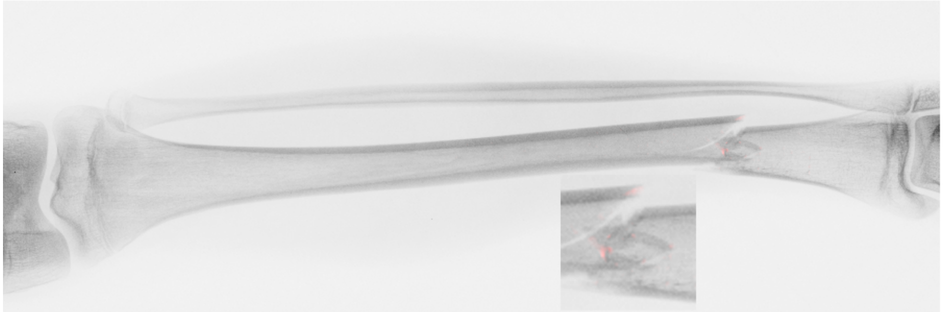


*Figure 7.7: To present the information to the user, the original image was artificially coloured, so that the abnormal regions were highlighted red, while the normal regions remained greyscale. The inset shows an enlargement of the fracture region, in which the bone fragments were highlighted in a soft red colour.*

1. Since artificial colouring was used to show the magnitude of the GCM, the highlighting was sometimes so subtle that it was not visible to the naked eye. As a result, some fractures would still be easily missed during interpretation of the x-ray by a physician or radiologist.

2. The highlighting was placed directly over the abnormality, and as a result it could interfere with the radiologist's interpretation of the image. This would be compounded by the fact that at the highlighted points—where the radiologist normally searches for subtle changes in intensity—the intensity was modified to show the magnitude of the GCM.

3. Most importantly, the artificial colouring method neither clearly identified nor classified regions containing fractures.

Of the three identified drawbacks, the first two were minimised by allowing the user to manually alter the colour scaling, and to selectively apply the colour highlighting, respectively. Thus, when the highlighting is very subtle, the user can interactively drag a slider bar to increase the intensity of the highlight. Alternatively, when the highlight masks an abnormality and complicates the image interpretation, the user can turn the highlighting off. The third drawback outlined above still required resolution, and to do this a second method of presenting the location of fractures was examined.

### 7.3.2 Marking regions for fracture identification

A second method of fracture identification was designed to address the drawbacks of the artificial colouring method. Firstly, instead of showing the magnitude of the GCM, markers that could not be easily missed during the interpretation of the x-ray were used. Secondly, the markers were not placed directly on the identified abnormalities, but were instead placed at a distance to ensure that all the intensity variations corresponding to the fracture were still visible, and could still be examined. Thirdly, only the points in the GCM that met the criteria for representing a fracture were marked, rather than all GCM features. This was achieved using a three step process:

1. The GCM image was filtered to remove noise.
2. The regions of the GCM where the intensity and surface area were likely to be part of a fracture were retained.
3. Any matching regions were marked as abnormalities that required further examination by the radiologist.

Each of these steps is discussed below in greater detail.

**Filtering the GCM**

In the first stage of fracture identification, a 9 x 9 median filter was used to remove salt and pepper type noise that was created by random matches between $\phi$ and $\theta$ in the gradient composite measure calculation. The relatively large size of this filter also had the effect of smoothing the image, so that points of high intensity that were clustered very closely were joined together. The size of this filter was found to be optimum because smaller filters did not provide a sufficient degree of smoothing, while larger filters were prohibitively slow. In fact, this size was chosen as it was the largest median filter that could be applied in a timely manner. The result of applying the median filter to the GCM image shown in Figure 7.6b is shown in Figure 7.8a. This image shows that the location of the fracture had become clearer due to both the smoothing and removal of noise.

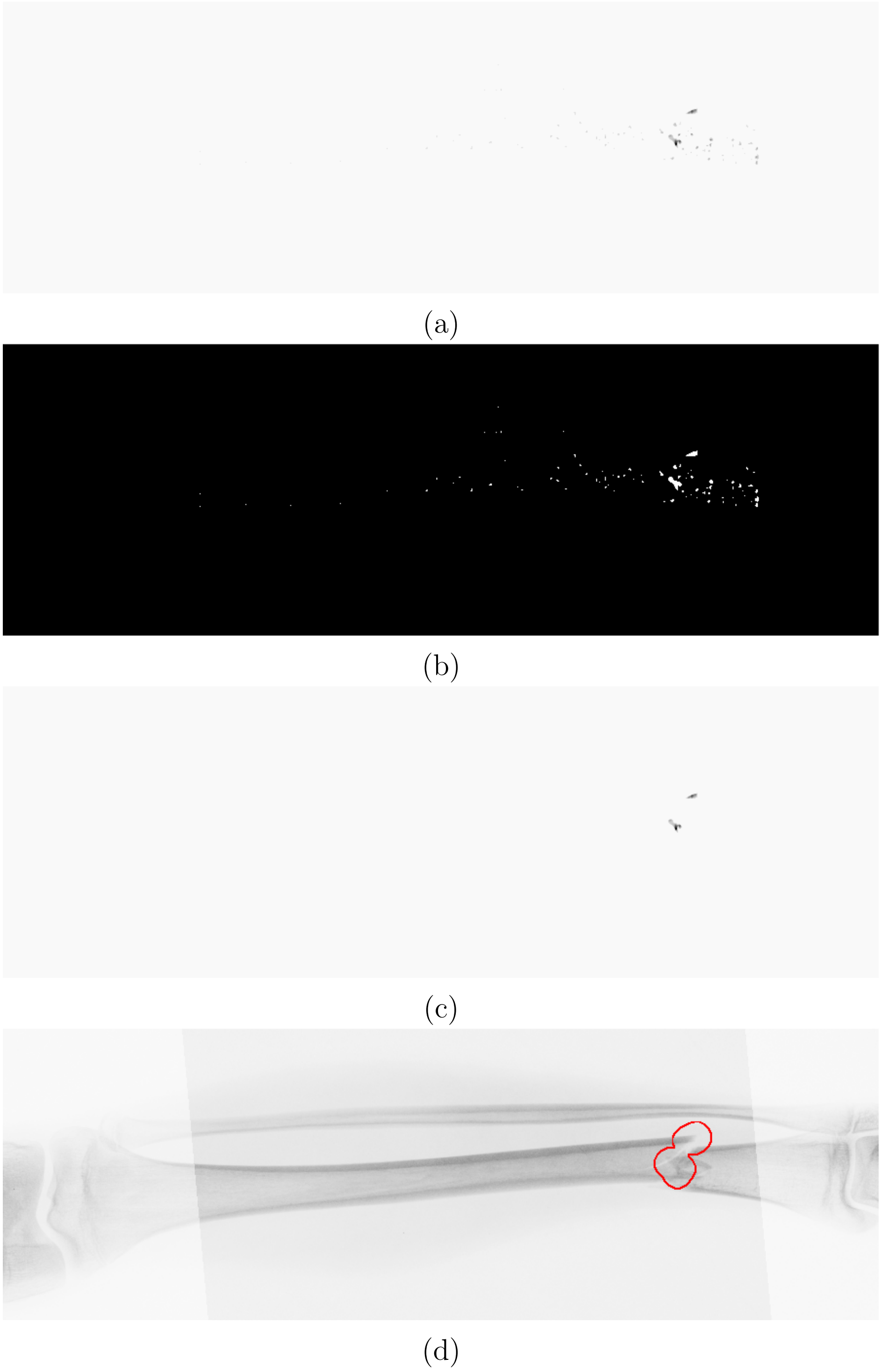
A strip around the image boundary was also cleared as it was prone to edge effects from the AMSS smoothing, and otherwise produced false detections. From the AMSS implementation equation 4.20, it was known that these edge effects (discussed in detail in Section 8.2.1) were limited to a distance less than or equal to the number of iterations $n$ of the AMSS smoothing equation. Finally, the segmented image was also normalised so that the range of values was correct for the subsequent thresholding stages.

### Retaining suitable clusters

To isolate the regions in the filtered gradient composite measure that were likely to correspond to a fracture, an empirically chosen threshold $T_3$ was applied to the image, so that only those pixels that had a sufficiently large composite measure were retained. The threshold $T_3$ was chosen to be relatively low, so that a large number of candidate points were retained at this stage. Figure 7.8b demonstrates that the resulting image contained many points that did not belong to a fracture. A low threshold was chosen because testing using the development images revealed that it was possible for fracture regions to have a relatively low filtered GCM intensity, in comparison to other regions of the image. That is, the individual pixels in the filtered GCM image with the highest intensity did not necessarily correspond to the fracture. However, in the GCM, the fracture regions were generally larger, and their greater surface area had to be taken into account. This was done in the following stage.

In the second thresholding stage, each cluster in the binary image produced by the first thresholding stage was examined. The location of every pixel in the cluster was recorded, and then the sum of the pixels in the corresponding locations in the filtered GCM image was calculated. As a result, this simultaneously measured both the area of a filtered GCM cluster, as well as the magnitude of the points within that cluster. This was different to measuring the sum of the pixels in a cluster in the binary image, which measured only the cluster area. The result was a list of clusters, along with the sum of the filtered GCM pixels corresponding to each cluster.

At this stage, a second empirically chosen threshold $T_4$ was applied, so that only the significant clusters were retained. It was mostly this threshold that was responsible



(a)

(b)

(c)

(d)

*Figure 7.8: (a) The gradient composite measure shown in Figure 7.6b was filtered using a 9 x 9 median filter to both remove salt and pepper type noise and join clustered high intensity pixels. (b) A threshold was then applied to remove the smaller regions and create clusters. (c) The clusters were analysed, and only those that were large enough, and had a high enough intensity were retained. (d) The clusters were used to mark the identified regions.*

for determining the sensitivity of the fracture identification algorithm. An example of the results produced by the second thresholding stage is shown in Figure 7.8c.

#### Marking the detected regions

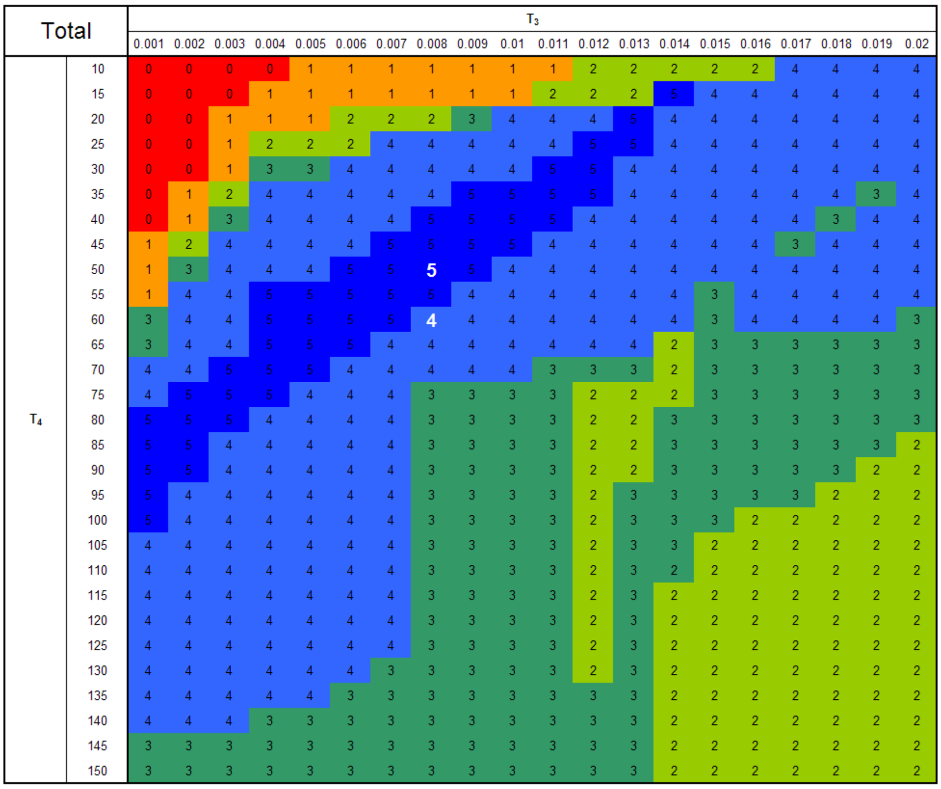
At this point, the GCM points that corresponded to an abnormality had been identified, and they were not modified any further. The final stage of fracture identification only involved displaying them in an appropriate manner. As discussed in Section 7.3.1, highlighting placed directly on an abnormality could interfere with the viewer's interpretation of the image, so it was determined that the markers should not be placed directly on any identified abnormalities. As a result, the GCM points were enlarged to a user adjustable size using morphological dilation and their boundaries were marked on the original image. Figure 7.8d demonstrates the results produced. In comparison to the artificial colouring method, the fracture was identified much more clearly, and was also not obscured by the marking.

### 7.3.3 Threshold sensitivity analysis

The accurate identification of fractures was much more highly dependent on the choice of the thresholds $T_3$ and $T_4$ than any other parameters such as the power $p$ and the size of the median filter. In addition, the two thresholds $T_3$ and $T_4$ were dependent on each other, since increasing one required that the other be decreased to provide a satisfactory detection. This meant that the choice of $T_3$ and $T_4$ had to be made very carefully. Not surprisingly, independently increasing either of these values resulted in fewer fractures being detected, while independently decreasing either of them resulted in normal regions being identified as fractures. Initially, the thresholds $T_3 = 0.008$ and $T_4 = 60$ were empirically chosen, after testing the six images in the development set. These choices resulted in six of the seven fractures being correctly detected—that is, four of the six images were completely correct—with two images containing false positives. To help more accurately choose the thresholds, a sampling based sensitivity analysis was performed.

Each of the six development set images was tested using a range of values for $T_3$ and

$T_4$ centred around the thresholds initially chosen. The range $[0.001, 0.02]$ with a step size of 0.001 was used for $T_3$, and the range $[10, 150]$ with a step size of 5 was used for $T_4$. This resulted in a total of 3480 images that needed to be manually examined. When this examination was performed, the combination of the two thresholds $T_3$ and $T_4$ was marked as producing either a correct or incorrect detection result. A correct detection result was defined as one in which there were no false negatives or false positives, while an incorrect detection result was defined as one in which there was one or more false positives or false negatives. The results for each of the development images were then combined to produce the results shown in Table 7.1. This plot showed that there was a small range of values for which five images (83%) could be correctly detected, but that no threshold choices resulted in all six images being correct. Choosing any pair of thresholds within the dark blue region (i. e. a value of 5) resulted in all seven fractures (across five of the images) being detected correctly, with the trade-off being that one image (that did not contain a fracture) displayed some false positive regions.



| Total | | $T_3$ | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0.001 | 0.002 | 0.003 | 0.004 | 0.005 | 0.006 | 0.007 | 0.008 | 0.009 | 0.01 | 0.011 | 0.012 | 0.013 | 0.014 | 0.015 | 0.016 | 0.017 | 0.018 | 0.019 | 0.02 |
| | 10 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 4 | 4 | 4 | 4 |
| | 15 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 5 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 20 | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 2 | 3 | 4 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 25 | 0 | 0 | 1 | 2 | 2 | 2 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 30 | 0 | 0 | 1 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 35 | 0 | 1 | 2 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 |
| | 40 | 0 | 1 | 3 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 |
| | 45 | 1 | 2 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 |
| | 50 | 1 | 3 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 55 | 1 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 4 |
| | 60 | 3 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 3 |
| | 65 | 3 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 2 | 3 | 3 | 3 | 3 | 3 | 3 |
| | 70 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 2 | 3 | 3 | 3 | 3 | 3 | 3 |
| | 75 | 4 | 5 | 5 | 5 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 |
| $T_4$ | 80 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| | 85 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 2 |
| | 90 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 2 | 2 |
| | 95 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 |
| | 100 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 |
| | 105 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 110 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 115 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 120 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 125 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 130 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 135 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 140 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 145 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | 150 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

*Table 7.1: The results of the sampling based sensitivity analysis, performed on the six images in the development set. The numbers represented the number of images for which those particular thresholds produced the desired output. The initial (4) and final (5) thresholds are shown in white text.*

The sensitivity analysis showed that there was a range of thresholds that could produce the correct output. As such, a new pair of thresholds $T_3 = 0.008$ and $T_4 = 50$, inside the dark blue region were chosen. It was decided that the first threshold $T_3$ should not be adjusted, since it was already appropriate for retaining a large number of candidate points, so only the second threshold $T_4$ was changed from its original empirically chosen value. These thresholds were then fixed for testing on the complete image test set to determine how suitable they were for identifying fractures.

## 7.4 Fracture detection evaluation

The fracture detection algorithms described in this chapter were evaluated by testing them on the 44 diaphyses in the test set that were previously segmented using the algorithm described in Chapter 6. After the algorithms were applied, the results were manually compared to the film images interpreted by a trained radiologist.

### Gradient composite measure calculation

Although the importance rank, distance rank and gradient composite measure were calculated for each image, they were not individually analysed. It is interesting to note that—disregarding any further analysis—in almost all cases, subjectively the unfiltered GCM produced a very good representation of the location of the fracture. This also meant that although the artificial colouring method described in Section 7.3.1 had a number of limitations, it too almost always showed the correct fracture location, despite doing so with a consistently poor clarity. This indicated that the GCM was a good tool for detecting fractures within long-bone diaphyses. The following section examines how well features within the GCM were classified, using the fracture identification algorithm.

### Fracture identification

The fracture identification algorithm used to analyse the gradient composite measure was then examined. When the manual comparison to the radiologist's results was performed, the number of true and false positive and negative results were recorded, along with the reasons for any false positives that occurred. A summary of the results is shown in the confusion matrix in Table 7.2. The test image set contained a total of 47 fractures, 39 of which were correctly detected by the algorithm, corresponding to a detection rate of 83% of all fractures. The algorithm correctly identified all the fractures present in 37 of the 44 images (84% of all the images), while in the remaining 7 images at least one fracture was not detected correctly (in one image two fractures were not detected correctly). In 13 of the 44 images (30% of all the images), all fractures were correctly identified and no non-fractures were identified—that is, there were no false positives or false negatives. The remaining 31 images contained at least one false positive region. Unfortunately, of the 9 normal images, only one was correctly identified as containing no abnormalities. By modifying the two thresholds $T_3$ and $T_4$, it was possible to drastically reduce the number of false positives, although this also simultaneously reduced the number of true positives. Therefore the two thresholds were retained at the values determined in the sensitivity analysis.

| | Predicted Non-fracture | Predicted Fracture |
|---|---|---|
| Non-fracture | 1 (11%) | 68 |
| Fracture | 8 (17%) | 39 (83%) |

*Table 7.2: A confusion matrix showing the results produced by the algorithm on a sample of 47 fractures present in the 44 image test set.*

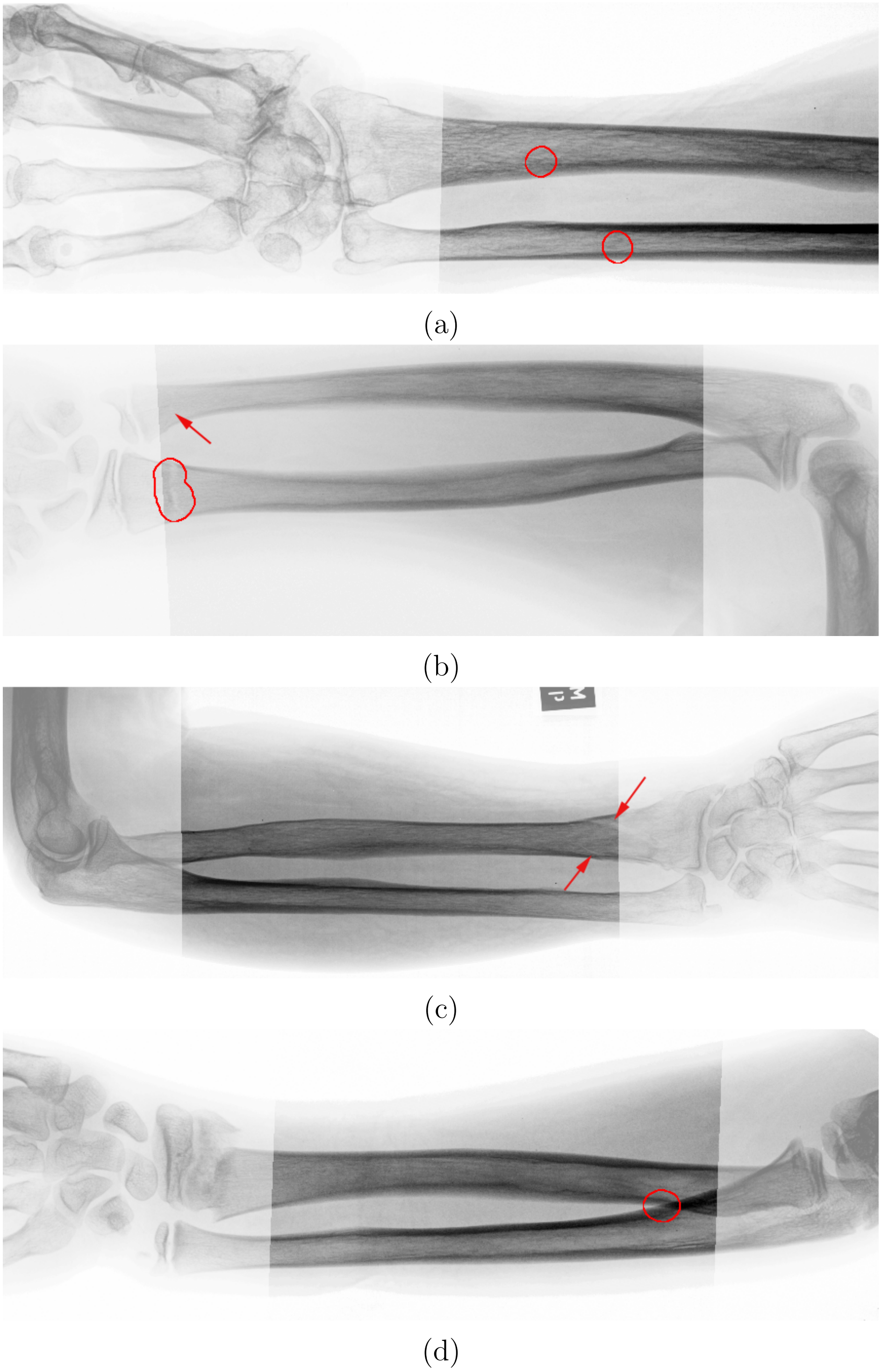
The causes of all the false positives were recorded, and were split into two categories: those that were caused by algorithm errors, and those that were caused by misinterpreted biological phenomena. Details are shown in Table 7.3. A second examination by a radiologist revealed that 49 of the 68 false positives (72%) were due to biological phenomena such as Harris growth arrest lines, soft tissue shadows or anatomical features unrelated to the fracture. Harris growth arrest lines (Figures 7.10c and 7.10d) represent episodes of growth arrest followed by recovery, and are often a marker of systemic illness that causes transient slowing of growth throughout the body during childhood. The soft tissue shadows (Figure 7.10d) were caused by the density variation of the soft tissues surrounding the bone, including fat under the skin which can mimic a fracture but extends beyond the bone. Other anatomical features such as nutrient foramen and some textures (Figure 7.10b) also caused false positives. As they were regions that should still be examined carefully during x-ray interpretation, highlighting of these regions was deemed acceptable, reducing the number of false positives to 19.

| | Cause of false detection | Number | Individual % | Group % |
|---|---|---|---|---|
| Algorithm | Bone overlap | 7 | 10.3% | 27.9% |
| | Image artifact | 12 | 17.6% | |
| Biological | Harris growth arrest line | 5 | 7.3% | 72.1% |
| | Feature not related to fracture | 38 | 55.9% | |
| | Soft tissue injury | 6 | 8.8% | |

*Table 7.3: The causes of false detection, split into those from the algorithm and those from biological variation. False detections due to biological variation were deemed by the radiologist to still be acceptable regions of interest.*

Of the remaining false positives, 7 (37%) resulted from the algorithm confusing overlapping bones as being a fracture, and 12 (63%) were caused by image artifacts. The overlapping bones (such as the overlapping radius and ulna in Figure 7.9d) generally caused a false detection because the magnitude of the gradient tended to be large at that point, and the curvature of the meeting point ensured that the direction of the gradient was parallel to the bone centre-line. According to the assumptions in Section 7.1, the point at which the bones overlap therefore met the same criteria as a fracture. Fortunately in many cases, the overlap points on the distal long-bones were removed during the diaphysis segmentation. Image artifacts (such as those in Figure 7.10a and 7.10b) on the other hand, were not necessarily even located over the bone. The example images show that these artifacts could be detected some distance from the bone, despite the distance rank greatly reducing their importance. This was possible because the algorithm described in Chapter 6 only determined the location of the long-bone diametaphyses (thus retaining the diaphysis), rather than masking out the non-bone areas. In some cases the artifacts corresponded to important features—so they were retained—although the majority were caused by scratches on the x-ray film, medical equipment such as casts or intravenous lines, pillows and creased clothing or sheets.

Despite the high false positive rate, the fracture detection algorithm algorithm was capable of detecting many fractures, some of which were very hard to identify visually. Two examples of these are shown in Figure 7.9a and 7.9b. Most importantly, the fracture shown in Figure 7.9a was not detected during the radiologist's initial visual



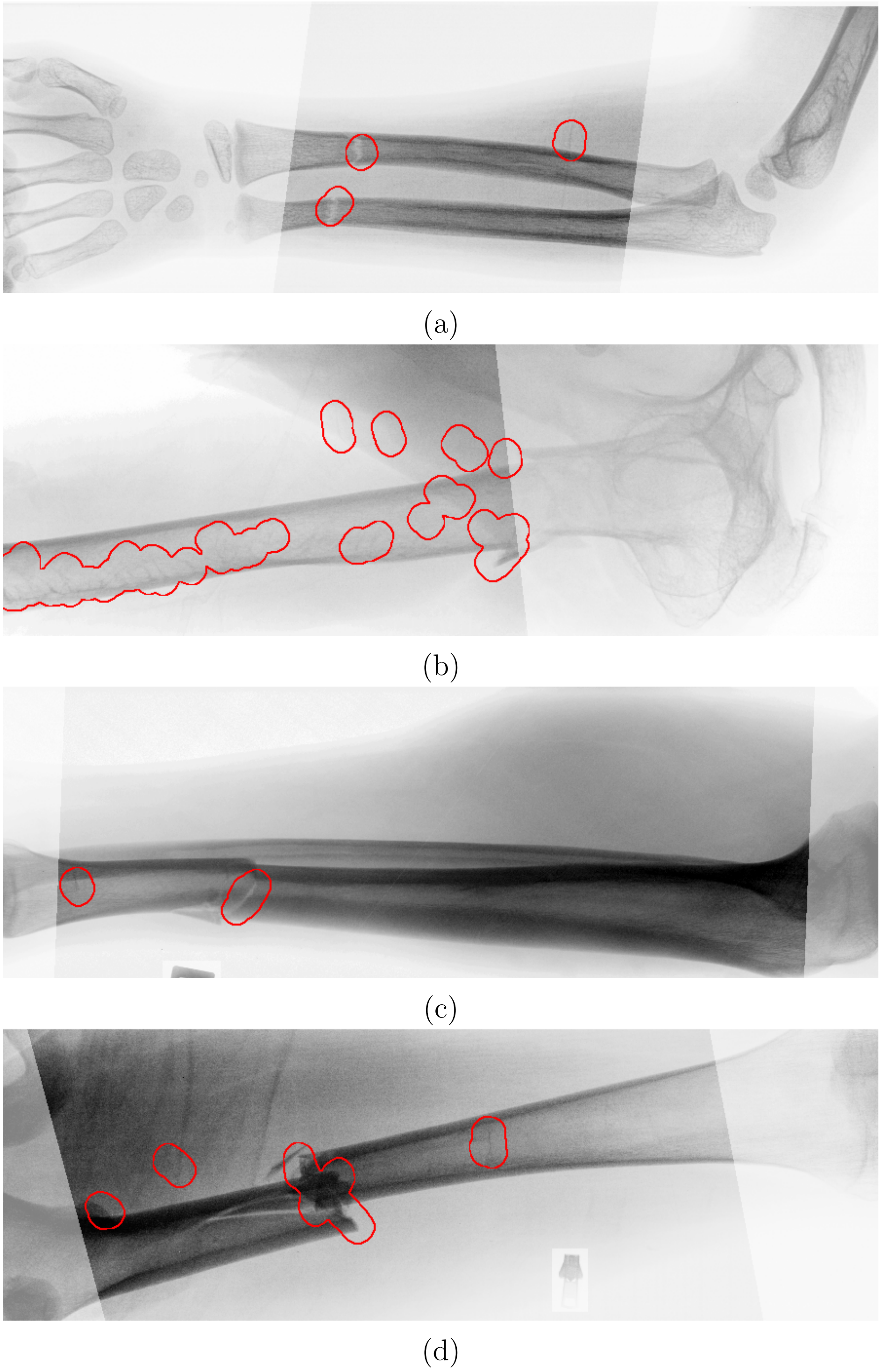
(a)

(b)

(c)

(d)

*Figure 7.9: Examples of the results produced by the fracture detection algorithm. (a) Two extremely subtle midshaft forearm fractures were detected correctly. (b) One subtle radius fracture was detected correctly, but the corresponding subtle ulna fracture (arrow) was missed. (c) Subtle radius and ulna fractures (arrows) were missed. (d) A false detection due to overlapping bones.*



(a)

(b)

(c)

(d)

*Figure 7.10: Examples of the results produced by the fracture detection algorithm. (a) Two midshaft forearm fractures were detected correctly, but an image artifact also caused a false detection. (b) An upper humerus fracture was detected correctly, but artifacts and non-fracture related features were also detected. (c) A midshaft tibia fracture was correctly detected, but a Harris growth arrest line was also detected. (d) A midshaft femur fracture was detected correctly, but soft tissue artifacts and a possible Harris growth arrest line were also detected.*

interpretation, but was only detected during the second reading using the results from the fracture detection algorithm. During the first interpretation, the more distal radius fracture outside the segmented area was identified, indicating that the miss was possibly an example of the SOS effect. In this case, the missed fracture diagnosis would have had little effect on the clinical outcome because a cast would have been required anyway. It has been suggested [41] that a CAD system is still useful even when its sensitivity is less than perfect, especially when the lesions detected by the algorithm do not overlap with those detected by the radiologist. In addition, a high sensitivity at the cost of an acceptable number of false positives is preferred, because false negatives generally produce a significantly poorer clinical outcome. The algorithm certainly showed that despite the high false positive rate, it was capable of reducing the number of false negatives.

It was unfortunate that the image test set contained many obvious fractures that were unlikely to be missed by an untrained observer, let alone an expert radiologist. It was these obvious fractures that caused most of the segmentation and detection difficulties throughout this thesis. Had the image data set consisted primarily of subtle fractures that are more difficult for an expert to detect, it is possible that the results would have been improved significantly. Therefore, although the complete fracture detection system was only partially successful, its ability to identify fractures that experts could either not see, or had difficulty seeing, was successful.

One identified problem was that the detection sensitivity of the fracture detection algorithm was a function of the angle at which the fracture occurred. Fractures parallel to the bone centre-line were not always as well detected as those perpendicular to it. This was because the gradient composite measure assumes that large gradients normally occur only in the same directions $\theta_i$ as the bone edges, and that all other gradients belong to some type of abnormality. Typically, most fractures are of this type. However, if a fracture was parallel to the bone edge, then it was classified as being normal, and was not highlighted. In the test image set, 3 fractures were either not detected, or only partially detected because portions of them were parallel to the bone edge.

The biggest problem with the long-bone fracture detection algorithm was that the two thresholds $T_3$ and $T_4$ were not always appropriate. The results demonstrated that using the current method, it was not possible to choose a set of global thresholds that were suitable for all images. When the sensitivity was adjusted by the user on a per image basis, it was possible to simultaneously lower both the false positive and false negative rates, so that close to 100% of the fractures were detected correctly. In a clinical setting, a computer aided long-bone fracture detection system could be configured to allow the radiologist to control the sensitivity of the computer output based on their own personal preference, or the nature of the case being examined. This sensitivity adjustment would be performed in a similar manner to setting a window level and width, or the use of edge enhancement when available on a workstation. As a result, the detection rate could be significantly better than the results presented here.

### The radiologist's comments

During testing the radiologist commented that the image development and test sets contained a good range of images. In addition, according to the radiologist, some of the false negatives were also very difficult to manually classify as either fractured or unfractured. For four images the radiologist would have requested a different view of the bone to better determine if a fracture was present. To make their decision, they would also have used visual and verbal cues, including a description about how the injury occurred, as well as the location and severity of any pain. This system does not currently utilise any of these cues. After examining the results produced by the detection algorithm, the radiologist stated that the system outlined in this thesis is very useful, and while the false positive rate is higher than desired, the highlighted areas could be more closely examined to determine whether there was indeed a fracture present.

## 7.5 Summary

Once the diaphysis was correctly segmented, a fracture detection algorithm could be applied to the segmented region. This chapter described a method by which fracture detection could be performed. Fracture detection was based on the assumption that an x-ray of a typical diaphysis containing no major pathology should exhibit gradient changes in the direction normal to the bone centre-line, but not in the direction parallel to the centre-line. Therefore, any large edges occurring at angles not normal to the centre-line were deemed to be part of an abnormality. These regions could be detected by utilising the long-bone shaft approximation parameters and a tool called the gradient composite measure.

The GCM was a combined measure of both the magnitude and direction of the gradient. It was calculated using the magnitude of the gradient and two scaling factors, which were used to incorporate the direction of the gradient with the angle and distance information from the long-bone shaft approximation parameters. These two scaling factors were referred to as the importance rank and distance rank, respectively. The importance rank was a measure of how well the direction of the gradient at any point within the image matched the $\theta$ parameter of the approximation lines. It was calculated using a scaled and translated sinusoid that had the effect of increasing the importance of all the angles close to $\theta$, while decreasing the importance of those further away. When applied to an image, the importance rank removed the regions corresponding to the bone edges, and retained features such as fractures, joints and epiphyseal plates. The distance rank was a measure of how close any pixel within the image was simultaneously located to all the approximation lines. When applied to an image, the distance rank reduced the intensity of regions a long way from the approximation lines, while retaining the bones at close to their original intensity. Finally, the GCM was the product of the magnitude of the gradient, the importance rank, and the distance rank.

Fractures could then be identified by using the GCM to artificially colour the original x-ray image. Normal areas remained greyscale, while abnormal areas were coloured in a user selectable colour. Although this method did highlight abnormalities such as fractures, it suffered from drawbacks that made it unsuitable. Therefore a method of marking the regions containing an abnormality was examined. This was a three step process consisting of filtering the GCM image to remove noise, performing dual stage thresholding to retain the appropriate regions, and marking any matching regions as abnormalities. The dual stage thresholding ensured that only regions of both sufficient size and intensity were retained, and allowed the sensitivity of the algorithm to be changed. Initially, the two thresholds were empirically chosen using the six images in the development set. However, the algorithm was very sensitive to these two thresholds, and so to ensure that the best values were chosen a sampling based sensitivity analysis was performed to find their optimum values.

Testing the fracture detection algorithm on the entire image test set showed that 83% of fractures were detected correctly, despite the presence of a high false positive rate. The causes of false detection were also analysed, and a large percentage of them were found to be from biological causes. One extremely positive feature was that the algorithm detected one additional fracture that was not detected by the radiologist during the first interpretation, thereby proving the value of this type of system. This was further reinforced by the radiologist stating that the system is very useful, after examining the results produced by the detection algorithm. Therefore, although the false positive rate is not ideal, the algorithm is well suited to long-bone fracture detection.

Most of the algorithms described so far in this thesis were very fast to compute, resulting in a relatively rapid diagnosis—one of the aims outlined in Section 3.3.4. The exceptions to this were the AMSS smoothing algorithm and the modified Hough Transform, which were time consuming due to the large number of iterations that were required for their calculation. As a result, an evaluation of the speed of these two algorithms was performed, and methods of decreasing the calculation time were also examined. These results are presented in the following chapter.