Removing Non-uniform Illumination Effects from Deep-sea Floor Images

JIN Shuying  GONG Jianya  ZHANG Xiaodong

1 Introduction

There are more than 70 percent of oceanic areas on the earth surface, where abundant and precious resources exist. In order to calculate the conservation and exploit multi-mental grains effectively, a lot of pictures on deep-sea floors have been taken under East Pacific Ocean with point light source shining, because of the darkness more than 5 000 meters under the sea. The effects of uneven illumination are shown clearly in Fig. 1. It makes common image processing such as segmentation difficult [1].

The deep-sea floor pictures are usually processed in the following steps:
1) digital image acquiring,
2) image smoothing to eliminate isolated noises,
3) removing non-uniform illumination effects,
4) geometric correction of image distortion,
5) image segmentation to extract multi-mental grains and measuring,
6) repeating from Step 1) to Step 5) until all pictures are processed,
7) statistically calculating and analyzing.

Among those steps, Step 3) is very important for accurate measurement. However, commercial image processing software packages (for example, PhotoShop) do not provide ideal algorithms for solving such problems.

Bernd Jähne [2] discussed three concepts for segmentation and pointed out that in a scene with uneven illumination, even if the object clearly cuts out of the background, an adequate threshold does not exist. It is much better to solve the problem at the
root, i.e., to optimize the illumination of the scene that we observed. If it is not possible, we should try to correct uneven illumination before we apply a complex segmentation procedure. ERDAS IMAGINE [3] provides a homomorphic filtering algorithm, which includes four steps:

1) transform the image from multiplicative to additive superposition using a log function,
2) perform Fourier transform to make frequency space image,
3) apply a filter on the Fourier image, which increases the high-frequency components, so as to enhance the reflectance image and de-emphasize the illumination image,
4) perform inverse Fourier transform to make spectral space image.

The result image (Fig. 2(a)) shows it is not ideal for the deep-sea floor images. M. Boninsegna, et al. [4] provided a filtering method based on Kalman filtering to eliminate non-uniform illumination effects on dynamic image series. This algorithm is not fit for single static image, but provides some ideas. R. A. Mckim, et al. [5] subtracted a reference background image, which is estimated by applying low-pass filtering, from the underground pipe image to detect the defects of underground pipe. However, the process of subtracting (or dividing) one image from another will make some of the dynamic range of the original data lose [2].

This paper compares the following three algorithms:

1) (local filtering) a filtering algorithm based on local statistics,
2) (image subtracting) subtracting a reference background image, which is estimated from the original data in the original image automatically,
3) (a new enhanced image stretching algorithm) stretching the original image according to the pixel values in the reference foreground and background images.

2 Image data

Deep-sea floor images are acquired through a camera. Three typical images with different density grains are chosen for the experiment (see Fig. 1).

The images show dark multi-mental grains against light sand background. A portion of the image appears lighter than the rest because of the point light source. Fig. 6(a) shows the brightness variations along with the 90th pixel line in the original image (shown in Fig. 1(b)). White arrows in Fig. 6(a) point out that the lightest grains pixels are lighter than the darkest sand pixels, which makes it impossible to turn a gray scale image to a binary (white and black) image using only a global threshold. Furthermore, the brightness difference between neigh-

![Fig. 1 Original images]

(a) high-density grains (b) moderate-density grains (c) low-density grains

![Fig. 2 Result images for moderate-density grains image]

(a) homomorphic filtering (b) brightness enhancement (c) contrast enhancement
bored sand pixels and grains pixels are also varying with the location. That is to say, the brightness difference is greater in light areas than in dark areas.

Common image enhancement results are shown in Fig. 2(b) and 2(c). Global brightness enhancement (Fig. 2(b)) causes the grains located in the light areas to disappear among sand background. However, global contrastness enhancement (Fig. 2(c)) makes the grains located in the dark areas vanish in sand background. These procedures just illustrate that non-uniform illumination effects is impossibly removed by common global image enhancement techniques.

3 Methods

3.1 Local filtering

As described above, though image enhancement is a widely applicable standard image processing technique, it acts on the image globally. Look at the following local enhancement filter:

\[
f_o(x,y) = (f_i(x,y) - m_i(x,y)) \cdot \left( \frac{1 - b}{s_i(x,y)} + c \right) + \left( 1 - b \right) \cdot m_i(x,y) + b \cdot m_d
\]  

(1)

where \( f_i(x,y), f_o(x,y) \) are the original and output grey value at position \( p(x,y) \); \( m_i(x,y), s_i(x,y) \) are the mean value and standard deviation of pixels in \( n \times n \) neighborhood; \( m_d \) and \( s_d \) are the desired mean value and standard deviation, respectively; \( b(0.0-1.0), c(0.0-1.0) \) are the constant coefficients of mean value and standard deviation.

Rewrite Eq. (1) to Eq. (2):

\[
f_o(x,y) = (f_i(x,y) - m_i(x,y)) \times r_2 + r_1
\]  

(2)

where \( r_1 = (1 - b) \cdot m_i(x,y) + b \cdot m_d \) represents the output mean value of pixels in \( n \times n \) neighborhood. When \( b = 0.0 \), \( r_1 = m_i(x,y) \), the output mean value of pixels in \( n \times n \) neighborhood will keep little varying; When \( b = 1.0 \), \( r_1 = m_d \), the output mean value of \( n \times n \) neighboring pixels of \( p(x,y) \) will become the desired mean value. Set \( 0.0 < b < 1.0 \), the output mean value of pixels in \( n \times n \) neighborhood will exist between \( m_i(x,y) \) and \( m_d \); Set \( m_d < m_i(x,y) \) where \( p(x,y) \in A_{\text{light}} \) and \( m_d > m_i(x,y) \) where \( p(x,y) \in A_{\text{dark}} \), then the output mean value of pixels in \( n \times n \) neighborhood will decrease in the light area but increase in the dark area. As a result, the uneven brightness will be improved to a certain extent.

\[
r_2 = (1 - c) \cdot \frac{s_d}{s_i(x,y)} + c
\]

is the output standard deviation of pixels in \( n \times n \) neighborhood. When \( c = 1.0 \), \( r_2 = 1.0 \), the output standard deviation of pixels in \( n \times n \) neighborhood will keep little varying; When \( c = 0.0 \), \( r_2 = s_d/s_i(x,y) \), the output standard deviation of pixels in \( n \times n \) neighborhood will become the desired standard deviation. Set \( 0.0 < c < 1.0 \), the output standard deviation of pixels in \( n \times n \) neighborhood will exist between \( s_d \) and \( s_i(x,y) \); Set \( s_d < s_i(x,y) \) where \( p(x,y) \in A_{\text{light}} \) and \( s_d > s_i(x,y) \) where \( p(x,y) \in A_{\text{dark}} \), then the output standard deviation of pixels in \( n \times n \) neighborhood will decrease in the light area but increase in the dark area. As a result, the uneven contrastness will be improved to a certain extent.

It is worth to note the local window size \( n \) when calculating the mean value and the standard deviation for \( p(x,y) \) neighborhood. The \( n \) can be neither very small (might have no grains pixels or sand pixels inside the local window), nor very large (might make the calculating process time-consuming). In fact, it should be determined according to the grain size and the grain density. For example, \( n = 37 \) is suitable for our moderate density grains image, shown in Fig. 1(b).

Result images of local filtering algorithm are shown in Fig. 3.

3.2 Image subtracting

Image subtracting is a widely used approach to remove uneven illumination effects. In many cases, it is not practical to acquire the background image separately by removing the objects. There are two methods to acquire the reference background image from the original data\([2.5.6]\). As is the case, the variation of background brightness is a smooth and well-behaved function of location and can be approximated by simple function, such as polynomial.
When the background variation is too irregular to be fit to simple functions, rank neighborhood approach is used. The prerequisite to this method is that the features of interest are limited in size and smaller than the scale of background variations and that the background is everywhere lighter or darker than the features\textsuperscript{6}.

In this experiment, the reference background images are fitted by a polynomial.

3.2 Fitting of the reference background image by polynomial

There are three necessary steps to estimate the reference background image:

1) selecting a number of sampled sand points in the image.

2) performing least-squares fitting of a polynomial function $f(x, y)$ by using the sampled points’ location and brightness.

3) for each point in the reference background image, calculating its brightness with the fitted polynomial function value.

For a 2D second-order polynomial, the functional form is:

$$a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2 = f(x, y)$$

(3)

There are six unknown parameters in Eq. (3), so the minimum number of sampled points is six. In order to get a good fit and minimize the errors, usually much more than six points are in need. These points should be distributed equally throughout the entire image areas. A practical method for locating the sampling points automatically in deep-sea floor images is used for the background fitting. Subdivide the original image into $m \times m$ grids, then find the lightest pixel (i.e., with the maximum grey value, which must be a sand pixel) in each grid, and these $m \times m$ pixels are used as the sampled points.

3.2.2 Subtracting the reference background from the original image

Assuming that $B(x, y)$ is the pixel value in the reference background image, $O(x, y)$ is the pixel value in the original image, $S(x, y)$ is the pixel value in the result image, then we have:

$$S(x, y) = [O(x, y) - B(x, y)] + C$$

(4)

where $C$ is a constant to make positive pixel value.

When the point $p(x, y)$ is a sand pixel (background), the item $[O(x, y) - B(x, y)]$ will be small all over the image, so $S(x, y)$ will be equal to the $C$ approximately. That is to say, the sand pixel’s value in the result image will be even throughout the image area.

Rewrite Eq. (4) to Eq. (5):

$$S(x, y) = O(x, y) + [C - B(x, y)]$$

(5)

When the point $p(x, y)$ is a grain pixel (foreground), $B(x, y)$ will be larger in light area than in dark area, so the item $[C - B(x, y)]$ will be larger in the dark area than in the light area. That is to say, the grain pixel’s value will be improved more in the dark areas than in the light areas, which also makes the grain pixels have close brightness all over the image area.

Result images of subtracting algorithm are shown in Fig. 4.

3.3 Enhanced image stretch

Though the image subtracting method improves the brightness in the dark areas, it does not improve the contrast in the dark areas (Table 1). The most terrible thing is that it reduces the dynamic grey levels of the entire image, which is not expected for image analysis. A new enhanced image stretching algorithm is put forward to improve both the bright and the contrastness in the dark areas.
Like image subtracting method, the enhanced image stretching method needs to acquire both the reference background and foreground images. The reference background image can be estimated by the algorithms described in 3.2.1. The acquisition of the reference foreground image is a little different from the acquisition of the reference background in selecting sampled points. It uses the darkest pixels (i.e., with the minimum grey value, it must be a grain pixel) in each grid for the polynomial fitting.

Point-by-point image stretching is described as follows.

After the background image and the foreground image have been acquired, stretching algorithm is applied to each pixel in the original image as:

$$S(x,y) = \frac{O(x,y) - B(x,y)}{F(x,y) - B(x,y)} \cdot C$$

where $S(x,y)$ is the pixel value in the result image; $O(x,y)$ is the pixel value in the original image; $F(x,y)$ and $B(x,y)$ are the pixel values in the reference foreground and background image respectively; $C$ is a constant to make pixel value $S(x,y) \in [0,255]$; $\cdot \cdot$ indicates the absolute function.

Think of the sand pixel's value in the original image (background) $O(x,y) \approx B(x,y)$, which makes the sand pixel value in the result image, i.e., $S(x,y)$, keeps small throughout the image. Compared with the reference background image, non-uniform illumination effects on the sand pixels are eliminated now. As for the grain pixel's value in the original image (foreground), $O(x,y) \approx F(x,y)$, which makes the grain pixel value in the result image, i.e., $S(x,y)$, keeps constant as $C$ throughout the image. Compared with the reference foreground image, non-uniform illumination effects on the grain pixels are also removed now.

Result images by the enhanced stretching algorithm are shown in Fig. 5.

4 Results and discussions

The images resulting from three algorithms are listed in Fig. 3–Fig. 5. In order to compare the result images objectively, four curves standing for the brightness variations with the locations for the moderate-density grain images and for its processed result images are shown in Fig. 6. The mean values and the standard deviations of the light, moderate, and dark areas in these images are listed in Table 1.

As for the local filtering algorithm (Fig. 3), it can be seen from Table 1 that the contrast are im-
proved throughout the entire image areas and that the brightness are improved only in the dark areas. The brightness difference between the light areas and the dark areas are reduced. However, both the brightness and the contrast are still lower in the dark areas than in the light areas, which means that the effects of non-uniform illumination still exist in the images after local filtering (also shown in Fig. 6). In addition, Fig. 3(c) shows more noises than those in the original image (Fig. 1(c)). This demonstrates that in the local window size (i.e. \( n \)) should be greater in the low-density grain image than in the high-density grains image, which will consume more computing time.

As for the image subtracting algorithm (Fig. 4), it can be seen from Table 1 that the brightness in the dark areas are improved greatly, and are approximately equivalent all over the image areas. However, the contrast is reduced in the whole image areas, especially in the dark areas. Thus some of the dynamic range of the original image are lost (Fig. 6(c)). Though image subtracting algorithm is widely used in removing uneven illumination effects, it is not particularly suggested here.

As for the enhanced image stretching algorithm (Fig. 5), it can be seen from Table 1 that both the brightness and the contrast are improved in the dark areas. Result images in Fig. 5 show that with the method we can completely removes the effects of non-uniform illumination. However, there are more noises in Fig. 5(c) than in the original image (Fig. 1(c)), this is because the sampled points selected automatically to fit the brightness foreground might not be the grain pixels. A solution to this problem is to subdivide the image into fewer numbers of grids and select fewer sampled points.

5 Conclusion

This paper compares three methods to remove non-uniform illumination effects and three images were tested. The work shows that:

1) local filtering algorithm can only reduce the uneven illumination effects and might add noises to the low-density grains image;

2) image subtracting algorithm can remove uneven illumination effects on the brightness but cannot remove those on the contrast, and might lose some of the dynamic range of the original image;

3) the enhanced image stretching algorithm developed by the authors can balance the brightness and contrast all round the image areas and may be the most effective approach to non-uniform illumination effects approach to remove.
in this case gives spikes.

Estimating error is necessary in further calculating the of the de-correlation operator $W(t_1, t_2)$. Denote this error as $\Delta R(t_1, t_2)$, so after calculation the bias appears in $W(t_1, t_2)$. Generally, this means incomplete de-correlation and decreased spatial resolution in comparison with accurate $R(t_1, t_2)$ estimation.

5 Conclusion

This paper introduces stochastic models of the surface, which is characterised by fluctuation of the scattered electromagnetic field and input signal. Main characteristic of the input signal in this case is the correlation function. These methods may be used for post-processing and real-time processing. The results of modelling shows quite good reliability by the real-time algorithms in comparison with post-processing mode. This fact allows using such an algorithm in SAR for on-board processing and de-correlate input signal. De-correlation makes images statistically stable and increase the resolution of the image.

References

1. Bakut (1963) Statistical theory of radiolocation. Moscow.
2. Ksendzuk A. (2001) Optimal processing algorithm in the active remote sensing. KhNU Proceeding, 2: 125-128
3. Ksendzuk A. (2000) De-correlation in synthetic aperture radar. KhPU Proceeding, 80: 12-14

Acknowledgement

Thank Bao Gengsheng and Shentu Haigang for providing the test images.

References

1. Castelmann K.R. (1996) Digital Image Processing. Englewood Cliffs, NJ: Prentice Hall.
2. Jähne B. (1993) Digital Image Processing: Concepts, Algorithms, and Scientific Applications. 2nd ed. Berlin: Springer-Verlag.
3. ERDAS Field Guide. (1999) 5th ed. ERDAS, Inc.
4. Boninsegna M., Bozzoli A. (2000) A tunable algorithm to update a reference image. Signal Processing: Image Communication, 16: 353-365
5. McHale R.A., Sinha S.K. (1999) Condition assessment of underground sewer pipes using a modified digital image processing paradigm. Trenchless Technol. Res., 14(2): 29-37
6. Russ J.C. (1992) The Image Processing Handbook. CRC Press, Inc.