Delineation of Mesoscale Features of Ocean on Satellite IR Image①

Li Jun (李 俊) and *Zhou Fengxian* (周凤仙)
Institute of Atmospheric Physics, Academia Sinica, Beijing 100029,
and *Gao Qinghuai* (高清怀)
Information Science Center, Peking University, Beijing 100871
Received December 22, 1989

ABSTRACT

An ICSED (Improved Cluster Shade Edge-Detection) algorithm and a series of post-processing technique are discussed for automatic delineation of mesoscale structure of the ocean on digital IR images. The popular derivative-based edge operators are shown to be too sensitive to edge fine-structure and to weak gradients. The new edge-detection algorithm is ICSED (Improved Cluster Shade Edge-detection) method and it is found to be an excellent edge detector that exhibits the characteristic of fine-structure rejection while retaining edge sharpness. This characteristic is highly desirable for analyzing oceanographic satellite images. A sorting technique for separating clouds or land well from ocean at both day and night is described in order to obtain high quality mesoscale features on the IR image. This procedure is evaluated on an AVHRR (Advanced Very High Resolution Radiometer) image with Kuroshio. Results and analyses show that the mesoscale features can be well identified by using ICSED algorithm.

I. INTRODUCTION

Infrared (IR) images of the ocean obtained from NOAA satellite sensor are widely used for the study of ocean dynamics. Fig.1 is a 256×256 pixels IR image with 1.8 km / pixel spatial resolution obtained from the AVHRR aboard the NOAA-9 satellite, the intensity value is represented by 8-bit digits. Some icelands and clouds are blanked out at the right side of the image (a late section will point out how to seperate clouds and land from IR image). The image shows the existence of Kuroshio and its meander. The Kuroshio and its associated eddies are the examples of mesoscale features. 'Mesoscale' is the term commonly applied to features with spatial scale of the order of 50 to 300 km. Mesoscale features are important to the study of ocean dynamics, to fisheries, and to many other application fields. Studies have been carried out on the mesoscale features of Gulf Stream. One of the difficult aspects to obtain the high quality mesoscale features is how to detect the cloudy area effectively. In this paper, three problems are discussed. The first one is separating clouds on IR image. The second one is delineation of mesoscale features by use of ICSED (Improved Cluster Shade Edge-Detection) algorithm. The last one is the post-processing of mesoscale features.

The radiance recorded by the IR channel from the radiometer of meteorological satellite is the infrared signal emitted by the surface (sea or land) and cloud top. Clouds are always present to some degree on the infrared image. It must be identified if one wants to get the sea

---

①Special Announcement: Starting with Volume 8, 1991, Advances in Atmospheric Sciences will be both published and distributed worldwide by China Ocean Press. For subscription information please contact us directly (see subscription sheet for details).

surface temperature with accuracy. Clouds are fairly easy to be distinguished on an satellite image from its temperature and structure. The lighter color generally corresponds to lower temperature and the darker one to higher temperature. The very bright region denotes cloud area because it is much colder than the surface. If the cloud is a dense mass, the ocean signal beneath is obscured. If the cloud is dense but scattered, then the ocean features can be seen from the gaps of the clouds. If cloud is low and thin, then they are barely distinguishable from the ocean. Finally if the cloud cell is smaller than the sensor resolution (1 km) and can not be seen on the image, then it is difficult to be removed.

Fig.1. NOAA–9 AVHRR image showing sea surface temperature in the Kuroshio region for March 3, 1986. The image is AVHRR channel four with a spectral bandpass of 10.5 μm to 11.5 μm.

Numerous cloud–detection methods have been developed to identify clouds on satellite image for the specific purposes. Some methods are for producing the parameters of global cloud climatology for general circulation models of the atmosphere. The information produced is the cloud amount, albedo and cloud height etc rather than the individual measurement based on the pixel. In the method used by NOAA / NESDIS for the Multichannel Sea Surface Temperature (MCSST) determination, subsets of the images are subjected to a series of standard cloud–detection tests to remove the effect from cloud (McClain et al, 1983). For the oceanographic purpose, someone wants to examine the time evolution or the statistics of SST patterns. In this case, the more important thing is to preserve the temperature patterns than to obtain the precise temperature. Nevertheless, cloud contaminated data must be eliminated effectively to prevent confusion between clouds and underneath sea if the data are to be used for quantitative analyses. Another method was developed by K.A.Kelly (1985). Three basic techniques are used by him in sorting procedure: a threshold test; a comparison of the two infrared channels and a uniformity test. The procedures work well at daytime and nighttime by use of AVHRR Channel 3 and Channel 4. But low, thin clouds and sparse scattered clouds are sometimes missed using this method. Also there are still some problems in choosing thresholds for the algorithm.

All methods above use only spectral information of the satellite images. In this paper, a combination of box–classification method (Liljas, 1982; 1986) and clustering technique is adopted for classifying the imagery data. The new method uses both spectral and spatial information of the multispectral satellite images and can distinguish almost all types of clouds at day and night. Especially, sunglint which is very difficult to be separated by other techniques is well identified by this new method.

II. DETECTING CLOUDS OVER OCEAN

In this study, dynamic clustering technique is used for separating clouds from ocean. The method and the usefulness of clustering technique on cloud image identification has been described in detail by Li and Zhou (1990). In that paper, Local Standard Deviation image—LSD image, was introduced as the spatial characteristics. In LSD image, the value attributed to each pixel is the Local Standard Deviation in the 3×3 neighborhood of the pixels. Also, the difference image between Channel 3 and Channel 4 was constructed as a new kind of spectral characteristic. LSD image provides the most important information about surface and clouds on the satellite images especially at night. For example, surface–fog discrimination can be achieved using Channel 3 / 4 (difference image between Channel 3 and Channel 4) and their LSD images. Because surface is more homogeneous on both Channel 4 and Channel 4 / 3 images at night. The main procedures of separating clouds can be summarized as follow:

DAY–TIME

Step D1: Use Channel 1, Channel 4 and their LSD images based on clustering technique to identify surface from satellite image. In classified image, the class which is the most homogeneous spatially in Channel 1 and Channel 4 and also very dark and warm, is the class of surface.

Step D2: On the region of surface, using Channel 1 and Channel 2 images based on box–classification method (Liljas, 1982; Liljas, 1986) to separate the ocean from surface. Using Channel 1, Channel 2 and their LSD images based on clustering technique can also identify sea from surface. Because on the classified image, sea has lower contour magnitude and is more homogeneous spatially in Channel 2 than the land.

NIGHT–TIME

Detection of clouds at night using satellite images is possible mainly because clouds are relatively cold and appear as light grey or white colors. The main problem at night is that the thermal contrast between the fog or stratus top and the surface is usually very small. But it can be solved by using Channel 3 / 4 image. In the region which is relatively cloud–free, the Channel 3 / 4 image is nearly featureless with a small magnitude. Also the region is most homogeneous spatially on the Channel 3 / 4. So the procedure of obtaining ocean (cloud–free) data at night can be described with the following steps.

Step N1: Land–sea image using Channel 1 and Channel 2 images based on box–classification method at daytime is created previously. Each pixel on the land–sea image is represented by 0 (land) or 1 (ocean).

Step N2: Use Channel 3 / 4, Channel 4 and their LSD images based on clustering technique to distinguish surface from cloud. In classified image, the class which represents the most homogeneous area spatially in both Channel 3 / 4 and Channel 4 and has the lowest magnitude of Channel 3 / 4 is the class of surface.

Step N3: Unify the classified image and the land–sea image to identify the ocean area.

III. IMPROVED CLUSTER SHADE EDGE DETECTION ALGORITHM

1. *Edge Detection*

Edge detection is one of the most common problems encountered in image analysis. Applications of edge detection have spread out to various fields such as remote sensing, industrial inspection, optical character recognition, medical imaging, robotics, and many others. Edge detection is often an important first stage in many types of image segmentation such as the present case, where edge detection can be used to simplify complex imagery in preparation for subsequent feature identification. The plethora of edge operators is very difficult to evaluate and compare. Trade–offs between edge detectability, noise sensitivity, and computational efficiency are always involved in selecting the appropriate edge operator for a given application. We will attempt to complete discussion of existing edge–detection techniques here, but will comment briefly on some of the more common approaches in order to show the inadequacy of these methods for finding edges of mesoscale features in satellite IR imagery of the ocean.

The edge of an image is characterized by the gray level difference and the gray level change rate. An edge exists if both the absolute difference and the change rate of the gray level are larger than the prespecified thresholds. Thus we can extract edges by performing a differentiation operator to an image.

The most common and earliest image edge operators are based on approximation of the intensity gradient with the image. The two orthogonal components of the gradient vector g_x and g_y are calculated separately then combined to give the gradient magnitude g

$$|g| = \sqrt{g_x^2 + g_y^2} \tag{1}$$

or for increased computational efficiency

$$|g| = |g_x| + |g_y| \tag{2}$$

Table 1. Kernels of Well–Known Image Edge Operators

	g_x	g_y
ROBERTS	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
SOBEL	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & -1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
PREWITT	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$
FREI–CHEN	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$
LAPLACIAN	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	

Researchers have proposed many ways to estimate g_x and g_y from a discrete function. Normally these estimation techniques are implemented as a convolution of the image with an appropriate gradient–approximating kernel. Examples of such kernels are given in Table 1. Of those shown in table, the Sobel operator is most widely used. Another common approach to edge detection arises from the mathematical Laplacian. The discrete Laplacian can also be implemented as a convolution with a 3×3 kernel. The Laplacian kernel is also given in Table

1.

The gradient and Laplacian techniques, which are based on first and second derivatives respectively, share a common shortcoming in the present application. Namely, they are toosensitive to noise, to fine–structure in the edges, and to weak gradients. To illustrate this problem, the g_x and g_y Sobel operators, selected as typical of the 3×3 kernel derivative approximation methods, have been applied to Fig.1, and the gradient magnitude has been calculated using (1). Fig.2 is an image of Sobel gradient values. The mesoscale features have been captured in the Sobel image, but many areas of the image void of mesoscale structure also exhibit edges comparable to those contained within the mesoscale features. Furthermore, the edges within the mesoscale features are very complex structures that show great detail. In many application this enhancement of detail by edge operator is desirable. However, a preferable result in the present application would be an edge operator that resulted in simple, smooth representation of edges within the mesoscale features and no edges at all in the oceanographically bland areas of the image.

The Laplacian, being an approximation to the second derivative, shows the disadvantage that it tends to enhance noise and weak gradient in the image. Using the Laplacian detector, the mesoscale features are almost not detectable.

Fig.2. Result of applying a Sobel edge operator to Fig.1.

It is desirable to devolep an edge–detection technique that would reject detail in the edges to a higher degree than show in Fig.2 while maintaining sharp edge definition and accurate localization of the edge. This objective is difficult because edges are characterized by much high–frequency content; henceforth, an edge detector must be, in effect, a high–pass filter. Passing high frequencies almost guarantees that the edge sharpness will be preserved. This paper present a new edge–detection method that possesses these desired smoothing characteristics to a degree that is not obtainable from small convolution operators such as those shown in Table 1.

2. *Improved Cluster Shade Edge–Detection Algorithm*

The edge algorithm described here is based on the cluster shade texture measurement. It is derived from the Gray Level Co–occurrence(GLC). The GLC matrix contains edge information (R.W.Conners, 1984). R.J.Holyer et al (1989) use GLC matrix to investigate edge measurement of Gulf Stream. Here we use ICSED algorithm to investigate mesoscale features of Kuroshio.

The $(i, j)^{th}$ element of the GLC matrix $P(i,j,\Delta x,\Delta y)$ is the relative frequency with which two image elements, separated by distance $(\Delta x,\Delta y)$, occur in the image with intensity levels i and j. Mathematically speaking, consider a $M \times N$ (e.g. 9×9) pixel image or local neighborhood within an image with L intensity levels ranging from 0 to L−1 (here L=256) and let $f(m,n)$ denotes the intensity level of the pixel at line m, element n. Then

$$P(i,j,\Delta x,\Delta y) = \sum_{m,n} A \tag{3}$$

Where

$$A = \begin{cases} \dfrac{1}{(M-\Delta x)(N-\Delta y)}, & f(m,n) = i, \quad f(m+\Delta x),n+\Delta y) = j \\ 0 & \text{otherwise} \end{cases}$$

both pixels at locations (m,n) and $(m+\Delta x, n+\Delta y)$ must lie within the $M \times N$ image space. $P(i,j,\Delta x,\Delta y)$ is therefore a $L \times L$ matrix of second–order probabilities.

The elements of GLC matrix could be combined in many different ways to give a single numerical value which would be a measure of edginess. One such measure, called cluster shade $S(\Delta x,\Delta y)$, has been found to be an especially good edge detection for the present problem.

$$S(\Delta x,\Delta y) = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} (i + j - \mu_i - \mu_j) \cdot P(i,j,\Delta x,\Delta y) \tag{4}$$

Where

$$\mu_i = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} i \; P(i,j,\Delta x,\Delta y)$$

$$\mu_j = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} j \; P(i,j,\Delta x,\Delta y)$$

The quantities μ_i and μ_j are estimates of mean intensity based on weighted summations of rows and columns within the GLC matrix. The edge detection in the present study is done by computing $P(i,j,\Delta x,\Delta y)$ and $S(\Delta x,\Delta y)$ in overlapping local neighborhood within the image. The center point of the neighborhood is then replaced by the $S(\Delta x,\Delta y)$ value computed from its neighborhood. Thus a "cluster shade image" with $S(\Delta x,\Delta y)$ values is created. Holyer et al. (1989) pointed out that the Δx and Δy values have very little effect in the resulting edge detect. So $\Delta x = 0$; $\Delta y = 0$ are set for the simplicity of mathematics, and the reduction of computation time. It can be simply proved that:

if $\Delta x = 0,\ \ \Delta y = 0$ then

$$S(\Delta x,\Delta y) = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N} (f(m,n) - \mu)^3 \tag{5}$$

where $$\mu = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N} f(m,n)$$

After the cluster shade image is computed by (5), then the edges can be detected by finding the significant zero crossings in the cluster shade image. The way to find the significant zero crossings is as follows. For each 3×3 pixel neighborhood in the cluster shade image the absolute value of the center pixel is tested against a defined threshold. If this value does not exceed the threshold, then a '0' will be output to a 'binary edge image' which is new created. If, however, the absolute value of any of the eight neighbors also exceeds the threshold but its sign is opposite to the center pixel, a '1' will be output to the binary edge image to indicate the presence of an edge. This algorithm is called CSED (Cluster Shade Edge–detection), it finds the major zero crossings in the cluster shade image and puts 1's in the resulting 2 or 3 pixels wide binary edge image. Fig. 3 is the example of CSED detector applied to Fig. 1. The window size in this case is 9×9. The CSED algorithm has the advantage of retaining sharpness of the edge, but sometimes there will still be some noise in the CSED image. To avoid this inadequacy, we develop a ICSED algorithm by computing (6) instead of (5). From (6) it is obvious that the cluster shade is proportional to gradient magnitude and exponent k in an absolute sense. This fact results in that with a big k, detection of the large gradients is associated with the major features while ignoring the smaller gradients.

$$S(\Delta x,\Delta y)=\frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}(f(m,n)-\mu)^{k} \tag{6}$$

$$k=3,5,7,9\cdots\cdots$$

Fig.3. Application of CSED algorithm to Fig. 1.

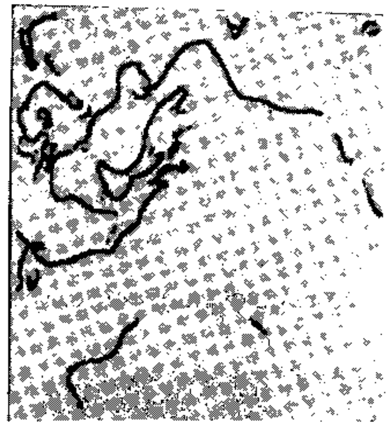

Fig.4. Application of ICSED algorithm to Fig. 1 with $k=7$

Fig. 4 is the result of ICSED algorithm applied to Fig. 1, it retains the sharpness of the mesoscale features while ignoring the noise comparison to Fig.3.

The choice of exponent k is very important to the detection of edge, theoretically, the big k will result in great contrast between large gradient and small gradient. Our experiment shows that when $k \geqslant 7$, the value has very little effect in the edge detection.

The question of optimal window size is highly dependent upon the application, no strict rules can be stated, in our regard, 9×9 pixel window gives the most suitable result when consider both the quality of edges and the computational expense.

IV. POST–PROCESSING OF THE EDGE

To get a high quality image for displaying and following analyzing, it is necessary to do a series of post–processing on the binary edge image. It means noise removing, semi–pixel thickening and thinning.

1. *Removing Noise*

On the binary edge image there are many isolate points or little point–clusters which are usually considered noise or false edges. They might be the real edges. But when we consider the main edges, they can be ignored. So we remove them from the image by a window–filter.

We choose a $n\times n$ window and move it from the upper–left corner through the image by one pixel each step. If there is a pixel on the boundary of the window which value is 1, nothing to do with the window. If the boundary pixels of the window are all 0′s, set all the pixels in the window 0′s. The window–filter fits for removing noise from the edge image because it saves the long edges and removes the short ones.

Choosing the size of the window depends on the application, we set it 15 and get a good noise–free image.

2. *Semi–pixel Thickening*

On the non–noise image there are edges which are multi–connective or broken. That is, there are one–pixel holes or broken points on the edge. To make the edge simple connective and avoid the adherence of the neighbor edges whose distance is 2 pixels, a semi–pixel thickening on the binary edge image is implemented.

First we enlarge the image A(256,256) to the image B(512,512) by the following definition: $B(i,j) = B(i-1,j) = B(i,j-1) = B(i-1,j-1) = A(i/2,j/2)$, where $i,j = 2,4,...,512$. Then, set the values of its 8 neighbors '1' if a pixel's value is 1 on image B and get image C(512,512). At last, define the resulting binary image D(256,256) as following: $D(i,j) = 1$if and only if $C(2i,2j) + C(2i+1,2j) + C(2i,2j+1) + C(2i+1,2j+1) \geqslant 3$, where $i,j = 1,2,...,256$.

3. *Thinning*

After semi–pixel thickening, the edges are 3 or 4 pixels wide. To get one–pixel edges, the image must be thinned.

Fig.5. Edges from Fig. 4 after cleaning / semi–pixel thickening / thinning.

Fig.6. Edges shown in Fig. 5 overlaid on the original image Fig.1.

The thinning algorithm here can be called stripping thinning. The main idea is deleting the deletable pixels until no pixel can be deleted. On a binary image, a pixel is deletable if and only if it satisfies: (1). its value equals 1; (2). change its value to 0 without changing the connectivity number of the whole image; (3). it is not an end point of edge. To get a clear formula , we define the connectivity number of the pixel x0 whose value is 1 as:

$$NC(x_0) = \sum_{k \in S} (\bar{f}(x_k) - \bar{f}(x_k)\bar{f}(x_{k+1})\bar{f}(x_{k+2}))$$

$$\text{where} \qquad \bar{f}(x) = 1 - f(x)$$

where $x_1,...,x_8$ is the 8 neighbor of x_0 and $S = \{1,3,5,7\}$. It can be proved that the condition (2) is equivalent with $NC(x_0) = 1$.

A pixel x_0 is an end point if and only if (1) one of its 8 neighbors is 1; (2) $NC(x_0) = 1$.

To get smooth and thin edges, some fine processing has been done in the algorithm to make the edges as natural as possible. It is important in the thinning algorithm. To cut the paper short, we omit the details.

Thinning is a very useful process in image processing. It can not only be used in post–processing. Its real importance is that after thinning the features can be extracted from the image easily. For example, we can compute the lengths and the shape features of the edges which are useful in the following analyses.

Fig.5. is the edge image after a series of post–processing operated to Fig. 4, the edges become connective and single–pixel wide.

V. ANALYSES

The Fig.5 edges overlaid on the original false color image Fig.1 is displayed in Fig.6. Though the image quality is degraded in Black / White picture, the wall of the Kuroshio, its associated cold eddies and its meander are still clearly identified.

The atmospheric attenuation has influence on the edge detection from the IR imagery of the ocean. It can lower the sea surface temperature (SST) by 1 to 3 degree. The overall lowering of the temperature value has no obvious effect on the edge detection. However, the atmospheric attenuation also reduce the magnitude of SST gradients. Under certain condition the atmosphere can completely obscure the SST patterns. The severity of the atmospheric degradation to SST gradients depends on the temperature profile and water vapor content. The warmer, more humid the atmosphere is the greater degradation of SST gradient is caused.

Multispectral techniques are frequently used to correct the atmospheric effects [Strong et al,1984]. However, the correction techniques introduce considerable noise into the resulting SST image. Also it does not reserve the SST gradient magnitude as they should do [Violette et al, 1988]. Therefore, it is not clear for edge detection to use raw image or processed image with atmospheric correction. In the present study raw imagery data from Channel 4 of AVHRR are used. The relationship between edge–detection performance and atmospheric attenuation effects should be carefully investigated.

VI. CONCLUSION

All types of clouds (includes sunglint) at day and night can be well identified using clustering technique and box–classification method before ICSED algorithm being carried out. Spatial information of meteorological satellite images plays an important role in cloud detection. The GLC matrix based edge–detection algorithm presented here has done an excellent job of delineating mesoscale ocean structure in the test image. The algorithm has the desirable characteristics of smoothing out the fine structure and reserving the edge sharpness and accurate location. Edge detectors with these characters show promise as a tool for segmentation of satellite IR imagery of the ocean. The ICSED algorithm should be also useful in other image processing applications where these characteristics are required. Application of the satellite images on atmospheric and oceanic sciences, theory and technique development of meteorological satellite image processing are the further works which will be carried on at the Institute of Atmospheric Physics, Chinese Academy of Sciences.

The authors are grateful to the Remote Sensing Division, National Research Center for Marine Environment Forecasts for their providing the AVHRR images.

REFERENCES

Conners, R.W., M.M.Trivedi, and C.A.Harlow (1984), Segmentation of a High-resolution Urban Scene Using Texture Operators, *Computer Vision, Graphics, and Image Processing* **25**: 273–310.

Cornillon, P. et al (1988), Processing, Compression and Transmission of Satellite IR Data for Near-Real-Time Use at Sea, *J. of Atmos.and Ocean. Tech.*, **5**: 320–327.

Gerson, D.J. and P.Gaborski (1977) Pattern analysis for automatic location of oceanic fronts in digital satellite imagery, Naval Oceanographic Office, TN 3700–65–77, Oct.1977.

Hoyler, R.J. and S.H.Peckinpaugh (1989), Edge Detection applied to satellite IR Imagery of the Ocean, *IEEE, Trans. Geosci. Remote Sensing*, **GE-27**: 46–56.

Janowitz, M.F. (1985), Automatic Detection of Gulf Stream, Office of Naval Research, *Tech. Rep*, TR-J8501, contr. N-00014-79-c-0629, June 1985.

Kelly, K.A (1985), Srparating Clouds from Ocean in IR Images, *Remote Sensing Environ.*, **22**: 432–447.

Li, J. and F.X.Zhou (1990), Computer Identification of Multispectral Satellite Cloud Imagery, *Adv, Atmos Sci.* **7**: 366–375.

Liljas, E. (1982), Automated Techniques for the Analysis of Satellite Cloud Imagery, *Nowcasting*, K. A. Browning, Academic Press, 177–190.

Liljas, E. (1986), Use of the AVHRR 3.7 Micrometer Channel in Multispectral Cloud Classification, SMHI, 1–23.

McClain, E.P.et al (1983), Multi-channel Improvments to Satellite-drived Global Sea Surface Temperatures, *Adv, Space Res.*, **2**: 43–47.

McMillin, L.M. and D.S.Crosby (1984), Theory and Validation of the Multi-windowSea Surface Temperature Technique, *J. Geophys. Res.*, 3655–3661.

Phulpin, T., M.Derrien and A.Brard (1983), A Two-Dimensional Histogram Procedure to Analyze Cloud Cover from NOAA Satellite High-Resolution Imagery, *J.Climate Appl. Meteor.* **22**: 1332–1345.

Robert, P. et al (1987), Interpreting Meteorological Satellite Images, *Bull. of Amer. Met. Soc.*, **68**: 762–767.

Strong, A.E. and E.P.McClain (1984), Improved Ocean Surface Temperatures from Space-comparisons with Drifting Buoys, *Bull. of Amer. Met. Soc.* **65**: 138–142.

Violette, P.E.La and R.J.Holyer (1988), Noise and temperature Gradients in Multichannel Sea Surface Temperature Imagery of the Ocean, *Remote Sensing Environ.*, **25**: 231–241.