1992

Edge Detection Algorithm for SST Images

Jean-François Cayula

Peter C. Cornillon
University of Rhode Island, pcornillon@uri.edu

Citation/Publisher Attribution
Cayula, J-F., & Cornillon, P. (1992). Edge Detection Algorithm for SST Images. *Journal of Atmospheric and Oceanic Technology, 9*(1), 67-80.
Available at: http://dx.doi.org/10.1175/1520-0426(1992)009<0067:EDAFSI>2.0.CO;2

Edge Detection Algorithm for SST Images

JEAN-FRANÇOIS CAYULA AND PETER CORNILLON

Graduate School of Oceanography, University of Rhode Island, Narragansett, Rhode Island

(Manuscript received 17 August 1990, in final form 14 June 1991)

ABSTRACT

An algorithm to detect fronts in satellite-derived sea surface temperature fields is presented. Although edge detection is the main focus, the problem of cloud detection is also addressed since unidentified clouds can lead to erroneous edge detection. The algorithm relies on a combination of methods and it operates at the picture, the window, and the local level. The resulting edge detection is not based on the absolute strength of the front, but on the relative strength depending on the context, thus, making the edge detection temperature-scale invariant. The performance of this algorithm is shown to be superior to that of simpler algorithms commonly used to locate edges in satellite-derived SST images. This evaluation was performed through a careful comparison between the location of the fronts obtained by applying the various methods to the SST images and the in situ measures of the Gulf Stream position.

1. Introduction

Satellite-derived sea surface temperature (SST) fields are often rich in structure, reflecting important underlying oceanographic processes related to eddies, currents, and regions of divergence or convergence. Many oceanographic studies that make use of satellite-derived data rely on an accurate location of edges associated with these features. To date, although it is labor intensive, the location of edges in oceanographic images has been performed subjectively by human operators. However, because of the constantly increasing amount of satellite-derived data available (Cornillon et al. 1987) and because computers have become faster and less expensive, there has been a rising interest in designing algorithms to automatically detect oceanographic features such as temperature fronts (Cornillon and Watts 1987; Gerson and Gaborski 1977; Holyer and Peckinpaugh 1989; Cayula et al. 1991). The primary advantage of such algorithms, aside from their speed, is that they can be designed to objectively detect fronts. In comparison, decisions by human operators are, in part, subjective and as a result, location of the same front will vary from day-to-day and operator-to-operator. Also, unlike subjective detection, an objective algorithm uses numerical estimates to determine the validity of a front. Consequently, an objective algorithm can produce quantitative information about a front without additional computations.

Existing computer vision algorithms have been designed for purposes other than the study of oceanographic data. As a result, they often work poorly when applied to sea surface temperature fields. In this article, we present a solution to the problem of objectively detecting and locating features of oceanographic interest in SST fields.

We continue our introductory remarks with a brief review of local versus regional operators and conclude them with an overview of the algorithm developed in this work. Subsequent sections deal with the details of the algorithm and a brief discussion of its application to SST fields of the western North Atlantic.

a. Local versus regional operators

The algorithm implemented in this study is derived from a combination of existing algorithms operating at the local level and the regional level. The difference between local and regional algorithms is best demonstrated by example. Temperature fronts, the features to be detected, are step edges. These can be defined in different ways depending on which approach is used. In the case of the regional approach, the edge is defined as the (usually thin) region of separation between two regions of constant temperature. As a result, an edge can only be detected if two populations are found in the area being examined. For example, in the one-dimensional case depicted in Fig. 1, an edge would be detected by the regional approach because two regions are indeed present and well defined in the interval considered. Similarly, an edge would be detected in Fig. 2, although the transition between the two populations is smooth. However, in Fig. 3 no edge would be detected because, aside from a few points, only one population is present.

Corresponding author address: Dr. Jean-François Cayula, The University of Rhode Island, Graduate School of Oceanography, Narragansett Bay Campus, Narragansett, RI 02882-1197.

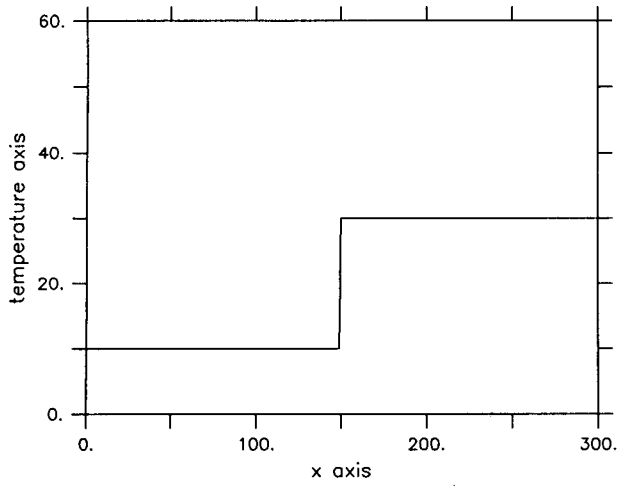

FIG. 1. Two well-defined populations.

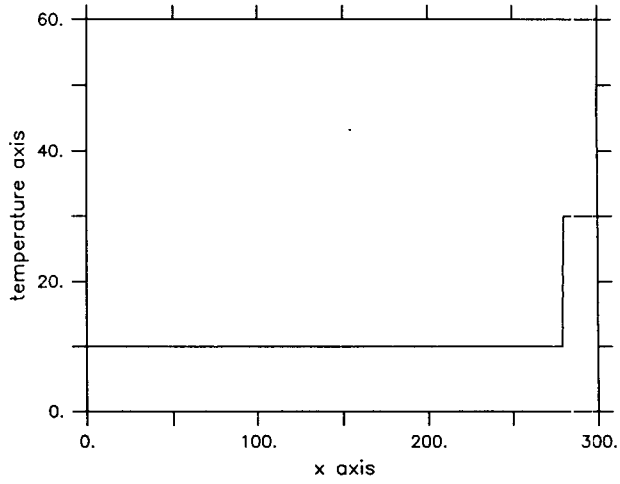

FIG. 3. An edge is present, but one of the populations is not well defined.

On the other hand, the local approach relies on the size and shape of the step separating the two populations. An example of such an approach is the gradient method in which a high gradient magnitude indicates the presence of an edge. The gradient method may be improved by noting that the gradient vector along an edge must have an approximately constant direction. Depending on the threshold used, an edge is likely to be detected in the cases shown in Figs. 1 and 3. The smoothness of the transition in Fig. 2, however, may prevent the detection of the edge.

b. Overview of the algorithm

The algorithm discussed in this work operates at three levels: picture level, window level, and local-pixel level, in much the same way as humans (Cornillon et al. 1987). The flowchart of the algorithm is presented in Fig. 4. Following input of the data, the most obvious clouds (based on temperature and shape) are identified and tagged so that data that do not represent sea surface temperature are not used in the subsequent modules (appendix A). These steps operate at the picture and then at the window level. The procedure continues at the window level with the formal portion of the edge detection. Using techniques for unsupervised learning (Duda and Hart 1973), the temperature distribution (histogram) in each window is analyzed to determine the statistical relevance of each possible front (Kittler and Illingworth 1986; Weszka 1978). To remedy the weakness related to the fact that clouds and water masses do not always form spatially distinct populations, the algorithm also includes a study of the spatial properties, such as cohesion and smoothness, instead of relying entirely on temperatures. In this way, temperature fronts are unequivocally defined. Finally, local operators are introduced to complete the contours found by the region-based algorithm [contour following (Ballard and Brown 1982)]. It should be noted that even though local operators are used, they are used in conjunction with the window-based algorithm, and so the qualities of scale invariance and of adaptivity associated with the region-based approach are not lost. As a result, the algorithm takes advantage of both the regional approach and the local approach while avoiding their drawbacks.

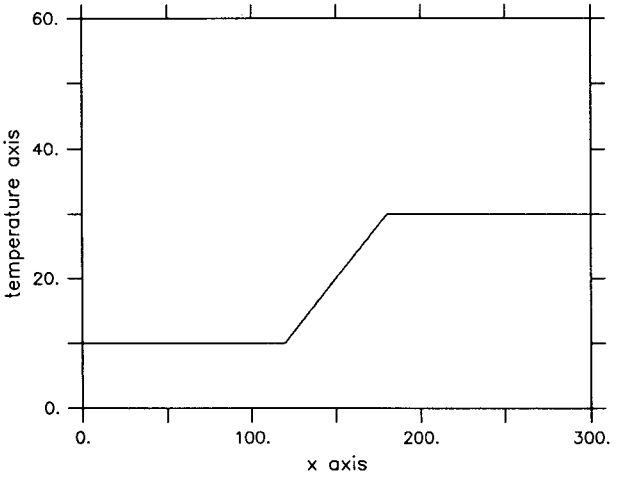

FIG. 2. The smoothness of the edge can be a problem for the local approach.

2. The algorithm

As indicated above, the complete algorithm deals with both cloud and edge detection. However, because the latter is the focus of this research, only the portion of the regional-level algorithm concerned with edge detection and the local-level section will be discussed

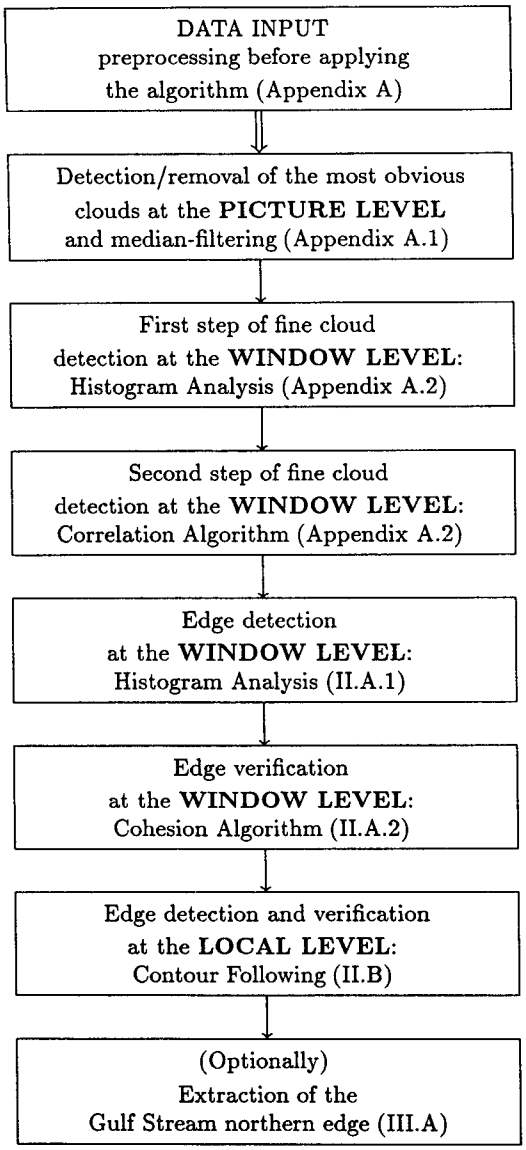

FIG. 4. Flowchart of the algorithm indicating the section of this article in which each step is explained.

in the following. For the sake of completeness, the picture-level portion of the algorithm and the regional-level section dealing with cloud detection are summarized in appendix A.

a. Window-level processing

The first step of the regionally based algorithm is to segment the entire (median-filtered) image into windows. Then each window is processed independently from the others to evaluate the probability of an edge being present in that window. Because an edge located at the boundary between two windows would escape detection, the image is segmented into overlapping windows. In the current configuration, the size of the windows has been set to 32 × 32 pixels. Such a choice in size results from a trade-off between opposing requirements. First, the dataset must be sufficiently large to obtain reliable statistics. Moreover, the windows must not be too small compared to the thickness of the edge. This is because an edge is defined as the boundary of two regions, ω_1 and ω_2, each of approximately constant temperature. If the transition between these regions occupies too much of the window, the two regions of constant temperature cannot be found and the edge is left undetected. On the other hand, by choosing a window that is too large, features that interfere with the detection of the edge may be included in the area under study. Other edges or clouds are examples of such interfering features. Although the window size and the (spatial) scale of the image are important parameters, the algorithm is robust to changes in these parameters: similar results were obtained by using 16 × 16, 32 × 32, or 64 × 64 pixel windows in processing images at scales of 1- and 2-km resolution. Note that the window size is also dependent on such factors as image noise and the resolution desired for edge detection.

1) HISTOGRAM ANALYSIS

The problem addressed in this section is the detection of an edge within a given window. Because of computational considerations, only the temperature distribution (histogram) is used to determine the presence of an edge. Qualitatively, it is easy to see that if an edge is not present in the window, the histogram will likely be unimodal. On the other hand, if an edge is present, then the histogram will be bimodal. Examples of histograms for real data are shown in the next two figures: Fig. 5 represents the histogram of a window with weak edge (see window 2 in Fig. 6a), while in Fig. 7 the

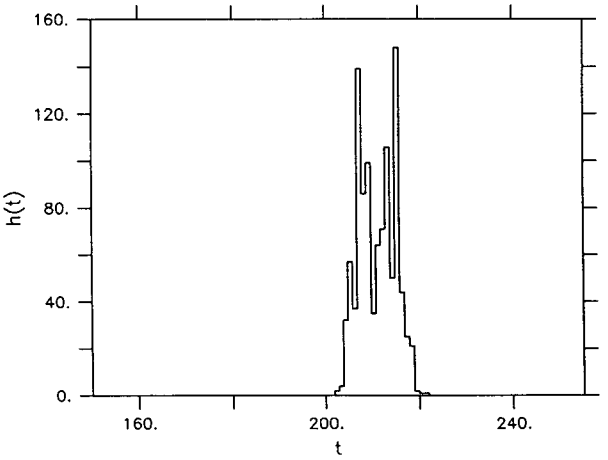

FIG. 5. Histogram of a region containing a weak edge, the boundaries of the region are represented by window 2 in the image shown in Fig. 6a.

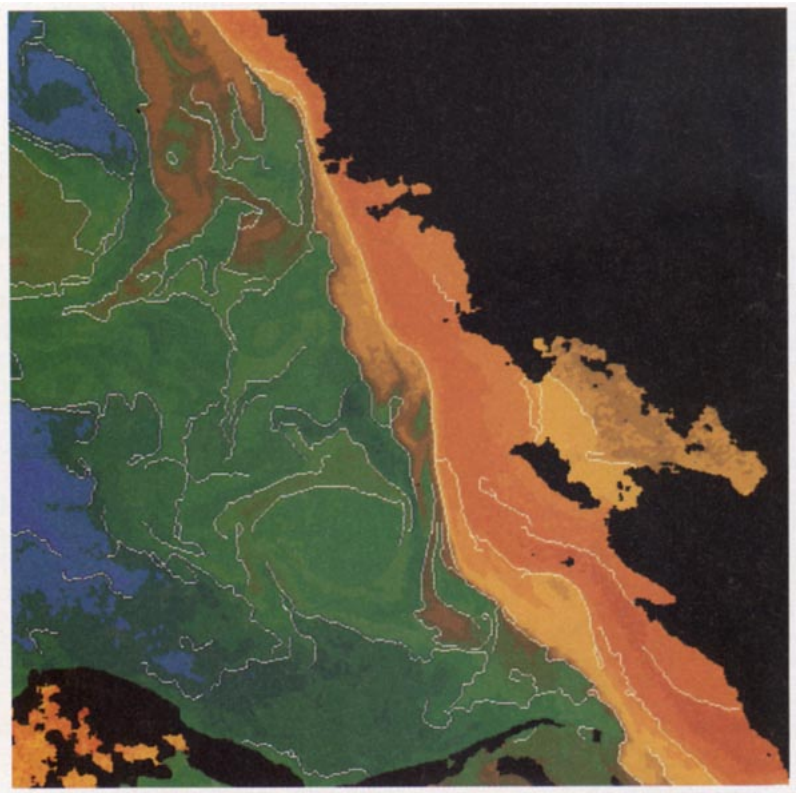

FIG. 6b. The SST fields off Cape Hatteras with clouds zeroed out and edges overlaid.

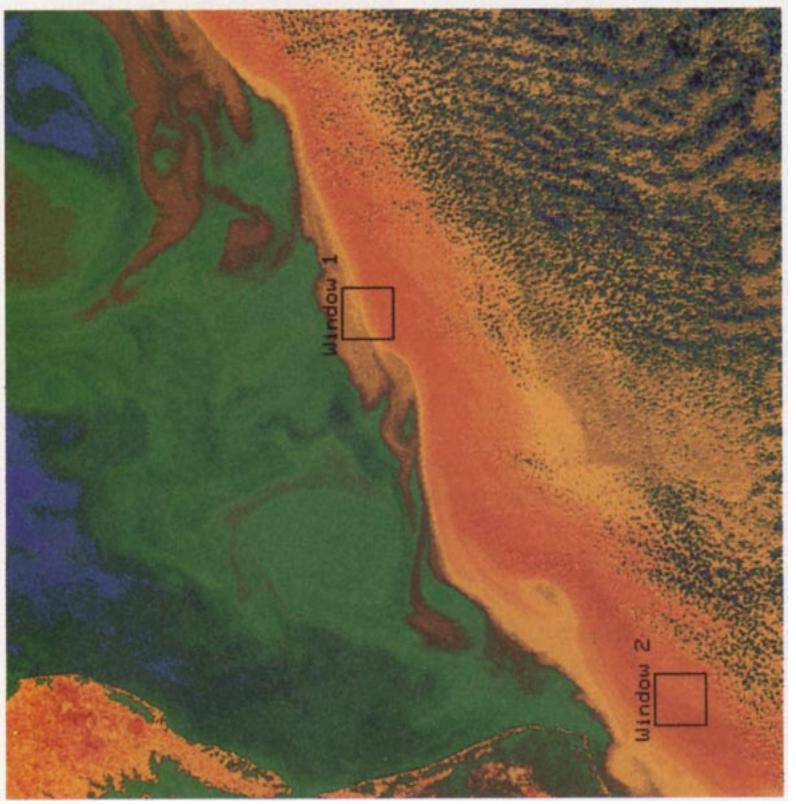

FIG. 6a. The SST fields off Cape Hatteras, North Carolina.

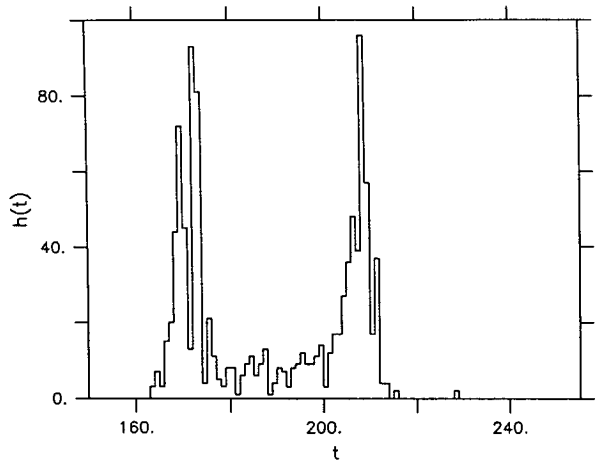

FIG. 7. Histogram of a region containing a strong edge, the boundaries of the region are represented by window 1 in the image shown in Fig. 6a.

histogram of a region containing a strong front is shown (window 1 in Fig. 6a).

The problem addressed by the analysis of the histogram is twofold: the algorithm must determine whether one or two populations are present and, if two populations are present, a threshold must be defined to separate them. To achieve this, the algorithm first assumes that two populations are present and the parameters for the populations are computed. These estimated parameters are then used to determine if the segmentation of the data in two classes is statistically relevant. If the data in the window came from only one class, then there is no edge present in the window.

Before presenting the method used in the algorithm, the theoretical basis for the problem of unsupervised segmentation is examined. Samples from class ω_1 are defined as the cold population and samples from class ω_2 form the warm population. Let x be a sample (pixel) from the set $\mathcal{X}$ (region under study) and $t(x)$ be the temperature of the sample x. The probability density of the mixture can be expressed as

$$p[t(x)] = \sum_{i=1}^{2} p[t(x) | x \in \omega_i] P(x \in \omega_i). \quad (1)$$

Once the parameters of the marginal densities and the a priori probabilities $P(x \in \omega_i)$ are determined, the validity of classifying the data in two classes versus one can be determined. If the two classes hypothesis is verified, the threshold τ that satisfies the Bayes minimum error criterion can be obtained by minimizing the probability of incorrect classification over the entire set $\mathcal{X}$ of samples:

$$P(\text{error}) = P[t(x) \geq \tau | x \in \omega_1] P(x \in \omega_1) + P[t(x) < \tau | x \in \omega_2] P(x \in \omega_2). \quad (2)$$

Exact solutions are available on occasion, in particular when the data are normally distributed (Duda and Hart 1973). However, obtaining the values of μ_i, σ_i and the a priori probabilities can be very complicated, requiring a large amount of computer time. Furthermore, the oceanographic data under consideration do not seem to be normally distributed and so an exact solution may not be attainable.

The problem can be drastically simplified, however, if we assume that the probability of error associated with the Bayes minimum error threshold, Eq. (2), is negligible. In such a case, the overlap between the cold and the warm population is small. Once the optimal threshold is known, parameters such as the mean and the variance of each population can easily be computed:

$$\mu_1(\tau) = \frac{\sum_{t<\tau} th(t)}{\sum_{t<\tau} h(t)}, \quad (3)$$

$$\mu_2(\tau) = \frac{\sum_{t\geq\tau} th(t)}{\sum_{t\geq\tau} h(t)}, \quad (4)$$

$$\sigma_1^2(\tau) = \frac{\sum_{t<\tau} (t-\mu_1)^2 h(t)}{\sum_{t<\tau} h(t)}, \quad (5)$$

$$\sigma_2^2(\tau) = \frac{\sum_{t\geq\tau} (t-\mu_2)^2 h(t)}{\sum_{t\geq\tau} h(t)}, \quad (6)$$

where $h(t)$ is the value of the histogram at temperature t. Although the threshold is not known, we note that the histogram only takes a finite number of values $[0, n]$, where n is the number of quantization levels for the computer representation of the temperature. This means that τ can only take the values 1 to n. Because n is not too large ($n = 255$ for the data available to this study), one can compute an estimate of the parameters of the mixture, for every τ possible. Then, the estimated parameters can be used to maximize or minimize a given criterion over all the possible values of the threshold τ. For example, if the density function of each population is assumed to be known and depends only on the first- and second-order moments, the threshold τ_{best} can be chosen such that,

$$J(\tau) = \sum_{t=0}^{n} \{p[t | \mu_1(\tau), \mu_2(\tau), \sigma_1(\tau), \sigma_2(\tau)] - h(t)\}^2 \quad (7)$$

is minimized. It is also noted that the minimum value of $J(\tau_{\text{best}})$ gives an indication of the goodness of fit of the mixture density (with estimated parameters) to the histogram (estimated mixture density). Some refinements to improve the various estimates are possible; in particular, when the data are assumed to be normally

distributed (Kittler and Illingworth 1986). However, the data used in this project are not normally distributed and the hypothesis of normal distribution is implicitly rejected when the tails of the distribution are neglected.

Because a theoretically optimal criterion function may not exist and would likely require a large number of computations when it exists, it is more practical to heuristically design the criterion function. An example of heuristics that leads to a computationally simple algorithm can be obtained by expressing the total variance S_{tot} as the sum of two terms: $J_e(\tau)$ and $J_b(\tau)$. Here $J_e(\tau)$ represents the sum of the variances within each of the two populations that result from segmenting the region with respect to the threshold τ. We refer to $J_e(\tau)$ as the within-cluster variance and we define it as follows:

$$J_e(\tau) = \frac{N_1}{N_1 + N_2} S_1(\tau) + \frac{N_2}{N_1 + N_2} S_2(\tau), \quad (8)$$

where

$$S_1(\tau) = \frac{\sum_{t<\tau} [t - \mu_1(\tau)]^2 h(t)}{N_1} \quad \text{and}$$

$$S_2(\tau) = \frac{\sum_{t\geq\tau} [t - \mu_2(\tau)]^2 h(t)}{N_2}, \quad (9)$$

with

$$N_1 = \sum_{t<\tau} h(t) \quad \text{and} \quad N_2 = \sum_{t\geq\tau} h(t). \quad (10)$$

Here $J_b(\tau)$ represents the contribution to the total variance resulting from the separation of the two clusters:

$$J_b(\tau) = \frac{N_1 N_2}{(N_1 + N_2)^2} [\mu_1(\tau) - \mu_2(\tau)]^2. \quad (11)$$

Assuming that the histogram exhibits two well-defined peaks, if τ is chosen optimally most of the variance in the window will be due to the difference in temperature between the two populations. Therefore, $J_b(\tau)$ will likely be large compared to $J_e(\tau)$. On the other hand, if pixels from one population are mistakenly included in the other population, some of the terms contributing to the between-cluster variance will be shifted to terms contributing to the within-cluster variance, $J_e(\tau)$. The preceding reasoning leads us to define the optimal threshold τ_{opt} as the value that maximizes the between-cluster variance $J_b(\tau)$. Furthermore, the ratio $J_b(\tau)/S_{tot}$ measures the proportion of the total variance due to the separation between clusters and as such gives an indication on the goodness of the segmentation. In particular this ratio, denoted $\theta(\tau)$, when evaluated at τ_{opt}, indicates how good the best segmentation is. Accordingly, $\theta(\tau_{opt})$ is the criterion used to decide whether one or two populations are present. The reasoning followed to select the best threshold and determine the number of populations is only approximate but we believe that it holds in the present case because the different populations (masses of water) generally have similar variances. Because conditions are not well defined, we are not able to establish the statistical relevance of the criterion $\theta(\tau)$. Although the ratio, $J_b(\tau_{opt})/S_{tot}$, is similar to such familiar statistical quantities as ρ^2 (the ratio of explained variance to total variance), it must be remembered that because the tails of the distributions in the mixture were neglected, the resulting estimates of the means and variances are biased. Furthermore, because no condition is imposed on the shape of the probability density function of each population, common statistical tests cannot be used.

Although an exhaustive analysis is not possible, the behavior of the criterion function $\theta(\tau)$ must be studied to define what criterion threshold leads to a relevant decision on the unimodality or bimodality of the histogram. We begin this study by examining the expected value of $\theta(\tau_{opt})$ for different unimodal distributions. For a normal population, the expected maximum value of the criterion function is

$$E[\theta(\tau_{opt})] = 2/\pi \simeq 0.63, \quad (12)$$

while for a triangular distribution, it is

$$E[\theta(\tau_{opt})] = 2/3 \simeq 0.66. \quad (13)$$

In both the triangular and normal cases, the expected value of $\theta(\tau_{opt})$ is independent of the parameters of the distribution and the size of the sample set and has a maximum value lower than 0.7. In fact, this threshold of 0.7 appears to hold whenever the histogram is clearly unimodal: for a 32×32 pixel window with normally distributed data, a rough approximation (Duda and Hart 1973) gives $P[\theta(\tau_{opt}) < 0.7] \simeq 0.99$. Inversely, to show that the criterion is greater than 0.7 when two modes are present, several different bimodal distributions were simulated (Cayula 1988). The results for one simulation (Fig. 8) are presented in Table 1. In light of all the cases studied, we selected 0.7 as the critical value discriminating between bimodal and unimodal distributions for use in the algorithm devel-

TABLE 1. Case of a mixture composed of two well-separated populations of unequal size that are normally distributed, with very different variances. Although it is difficult to distinguish the two modes by looking at the histogram (Fig. 8), the algorithm correctly identifies these two modes. Simple methods that use fixed thresholds, or local maxima of the histogram, would likely not work in this case.

Parameters	Exact	Estimated
$P(\omega_1)$	0.75	0.76
μ_1	50.0	50.3
μ_2	150.0	153.2
σ_1	3.0	6.4
σ_2	30.0	26.5
τ_{opt}	61.8	74
$\theta(\tau_{opt})$	—	0.91

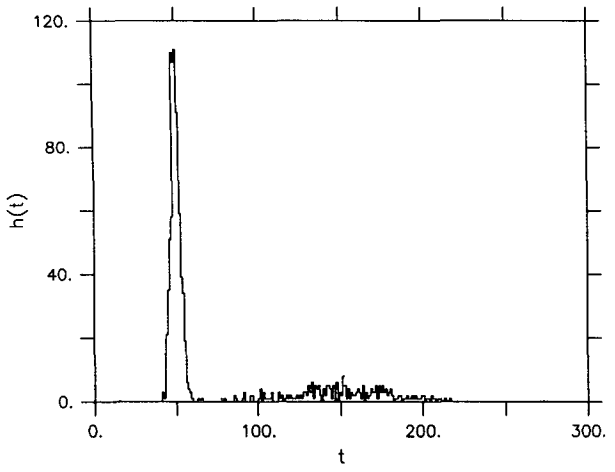

FIG. 8. Histogram of the mixture described in Table 1.

oped in this study. There are, however, potential problems associated with this critical value that require further discussion. First, we note that when the values of such parameters as the size and variance of the two populations are unequal, the algorithm performance deteriorates rapidly for a signal-to-noise ratio of less than 4 (we define the signal-to-noise ratio as $[(\mu_1 - \mu_2)^2/J_e]^{1/2}$. Although this requirement on the maximum amount of noise may seem very strict, it should be remembered that because the images are median filtered, if the populations are assumed to be normally distributed, a signal-to-noise ratio of 4 (or 6) would approximately correspond to a signal-to-noise ratio of only 2.3 (or 3) in the original unfiltered image (Cayula 1988).

It was also noted that with a signal-to-noise ratio of 4, parameter estimates are quite accurate as long as the size of the two populations does not differ by more than a factor of 3. As a result the algorithm was designed to discard any segmentation that would result in an estimated probability that does not obey the condition:

$$\hat{P}(\omega_i) \geqslant 0.25 \quad \text{for} \quad i = 1, 2. \tag{14}$$

In Table 1, the exact value of τ_{opt} is the threshold that satisfies the Bayes minimum error criterion of Eq. (2). The reason for the difference between the estimated and the exact values (in particular, the standard deviations of the two populations tend to average out) is mainly due to the fact that the clusters are not well defined. Indeed, a population with high variance forms a loose cluster. If the second population forms a tight (low variance) cluster, then the algorithm is confused by the two different kinds of clusters. When the loose cluster is compared to the tight cluster, the loose cluster may not look like a cluster at all. However, unequal variances are not a major problem for two reasons. First, the variances in SST associated with two adjacent masses of water are usually similar. Second, significant differences in the value of the variances most often occur when one of the populations is a cloudy or a land region (high variance), and the other population is a mass of water. In such cases, however, the difference in temperature between the two populations is often large, hence, correct partitioning can still be achieved. Table 1 shows that although the exact and estimated τ_{opt} for two unequal normal distributions are different, the error committed in the estimated size of the largest population is only about 1%.

The two other distributions presented highlight a fundamental deficiency of any segmentation algorithm based solely on pixel values. If the data are distributed such that:

$$p[t(x) = k] = p[t(x) = k + 1] = 1/2 \text{ for } x \in \mathcal{X}, \tag{15}$$

then, by looking at the histogram of such a mixture, it seems likely that only one population is present. However, maximizing the criterion function leads to $\theta(\tau_{\text{opt}}) = 1$ (for $\tau_{\text{opt}} = k + 1/2$), indicating that the hypothesis of two populations is highly probable. Until further information is made available, such as limits on the possible range of the variances, an objective decision cannot be reached. A similar problem exists with the uniform distribution $u(m, b)$, such that:

$$p_u(x) = \begin{cases} 0, & \text{if } x < (m - b); \\ 1/2b, & \text{if } (m - b) \leqslant x < (m + b); \\ 0, & \text{if } x \geqslant (m + b). \end{cases} \tag{16}$$

When two populations of equal size, with distributions $u(m_1, b)$ and $u(m_2, b)$ are present and do not overlap ($|m_1 - m_2| \geqslant 2b$), the following expression is derived:

$$\theta(\tau_{\text{opt}}) = \frac{3/4(m_1 - m_2)^2}{3/4(m_1 - m_2)^2 + b^2}. \tag{17}$$

For $m_2 - m_1 = 2b$, $\theta(\tau_{\text{opt}})$ can be found to equal 0.75. This result is satisfying if two populations are present in the mixture. However, this last case could also represent the situation when only one population with the distribution $u[(m_2 - m_1)/2, 2b]$ is present. Again, it is not possible to distinguish between one or two populations in this case unless more information is supplied. In the section on posthistogram analysis, an algorithm that offers a solution to such a problem by using the spatial distribution of the data as a source of new information is discussed.

Thus far, it has been assumed that, at most, two populations exist in a window. Sometimes, however, three populations are present. When the third population represents a small portion of the window under study, the algorithm will only detect the two principal populations. In most cases, such behavior is acceptable because the third population will likely be detected in one of the overlapping windows, in which it represents

a larger proportion of the dataset. If the detection of three populations is required, a new criterion function $\theta(\tau_1, \tau_2)$, which would be similar to $\theta(\tau)$, could be designed. The resulting algorithm would then consist of maximizing the new criterion function for every τ_1 and τ_2 such that $\tau_1 < \tau_2$, and then, the best segmentation into three populations would have to be compared to the segmentation into two populations and to the one population case. However, for an image composed of 8-bit pixels, the computational cost can be multiplied by as much as 128. For satellite-derived fields of 1-km resolution, overlapping 32×32 pixel windows were sufficiently small to eliminate the multiplicity of fronts within a given window as a serious problem; this providing that fronts associated with land or clouds are removed prior to applying the edge detection algorithm (appendix A).

2) COHESION ALGORITHM

By neglecting to take into consideration the spatial distribution of the data in the previous section, a large amount of information was left unused. For example, in the case described by Eq. (15), if all the pixels of value k map to a specific region of the window, then the hypothesis of a front being present in the area would definitely seem more legitimate. On the other hand, if pixels of any value are uniformly distributed throughout the window, then the hypothesis that a front exists in the window would be invalidated. Whenever the histogram indicates the presence of two populations, the spatial distribution of the data should be examined to see if a front is present or if the bimodal distribution is simply a result of an erratic temperature pattern. In studying SST fields, the problem of a bimodal distribution resulting from scattered clouds over water with no front present is encountered on occasion. This problem may also occur when the sensor has intermittent noise, as was the case with some *NOAA-6* data.

In general, if a front is actually present, segmentation obtained by thresholding the data at τ_{opt} will map the two populations to a few large blocks within the window. To test for such spatial compactness (*compact* is intended in the common sense), an algorithm that measures the cohesion of each population has been designed. This algorithm is only applied when two populations, ω_1' and ω_2', have been detected in the histogram analysis. Populations ω_1' and ω_2' are defined such that, for a pixel x with temperature $t(x)$:

$$t(x) \leqslant \tau_{\text{opt}} \Rightarrow x \in \omega_1' \quad \text{and} \quad t(x) > \tau_{\text{opt}} \Rightarrow x \in \omega_2'. \tag{18}$$

The cohesion coefficients for populations ω_1' and ω_2', and for the entire dataset are defined as follows:

$$C_1 \equiv \frac{R_1}{T_1}, \tag{19}$$

$$C_2 \equiv \frac{R_2}{T_2}, \tag{20}$$

$$C \equiv \frac{R_1 + R_2}{T_1 + T_2}, \tag{21}$$

where T_1, the total number of comparisons between center pixels belonging to population ω_1' and neighbor pixels belonging to either population, is given by:

$$T_1 = |\{(x, y), \text{ such that } y \in [\mathcal{N}(x) \cap \mathcal{X}], \quad \forall x \in \omega_1'\}|, \tag{22}$$

and R_1, the total number of comparisons between center pixels and neighbors that both belong to population ω_1', is given by:

$$R_1 = |\{(x, y), \text{ such that } y \in [\mathcal{N}(x) \cap \omega_1'], \quad \forall x \in \omega_1'\}|. \tag{23}$$

Here R_2 and T_2 are similarly defined by substituting ω_2' for ω_1' in Eqs. (22) and (23) and $|\ \ |$ is defined as the cardinality of the set. For reasons of computational economy and simplicity, only the first neighbors of a given pixel are used to evaluate the cohesion. If $x_{i,j}$ is the center pixel, the set of first neighbors is $\mathcal{N}(x_{i,j}) = \{x_{i,j+1}, x_{i,j-1}, x_{i+1,j}, x_{i-1,j}\}$.

High cohesion means that for a given pixel that is not close to the edge, neighboring pixels are likely to belong to the same population. This implies that the spatial segmentation of the area into populations ω_1' and ω_2' is validated. Inversely, low cohesion means that there is a nonnegligible probability that a given pixel has neighbors from a different population and so the presence of the front detected by the histogram analysis is improbable. For a checkerboard pattern the three cohesion coefficients are zero and for an image with the bottom half coming from one population and the top half coming from the other population, the cohesion coefficients will tend to unity when the effects at the edge between the populations are neglected. A threshold of 0.92 for C and 0.90 for C_1 and C_2 was chosen in this work to eliminate edges resulting from noisy distributions; that is, if C is less than 0.92 or C_1 or C_2 was less than 0.90, the segmentation is discarded as unreliable. The selection of 0.92 and 0.90 as thresholds is discussed in appendix B.

To reduce the number of computations needed to estimate the cohesion coefficients, only two neighbors $[\mathcal{N}'(x_{i,j}) = \{x_{i,j+1}, x_{i+1,j}\}]$ of the center pixel are examined. The algorithm has also been slightly modified to make the cohesion coefficients relatively invariant to different edge contours; for example, diagonal or horizontal straight edges result in the same cohesion coefficients.

There are cases, however, for which the contribution of the edge to the estimate of the cohesion may be different from the contribution of a straight edge. It should be noted though, that if $P(\text{error})$ is zero, the threshold on C can accommodate a front about three

times longer than a straight front. This trade-off between length of the front and signal-to-noise ratio is actually an interesting property of the algorithm. Because SST fronts are generally associated with mesoscale oceanographic processes, the fronts are expected to be fairly straight (or at least smooth) on the scale of the 32×32 pixel windows. On the other hand, rugged edge contours are most often found at the boundary of clouds or of land areas; that is, they are not edges between water masses. Because long (not straight) edges often are not valid temperature fronts, they are less reliable than shorter (straight) fronts. Accordingly, the condition on the minimum signal-to-noise ratio should be made more stringent for accepting long edges as valid temperature fronts.

In summary, it can be seen that the cohesion algorithm estimates both the length of the edge contour in a window and the compactness of each population. Although the edge effects on the cohesion could be reduced, physical considerations of the problem justify the approach used to estimate the cohesion coefficients.

3) LOCATION OF EDGE PIXELS

The window-level algorithms discussed above were designed to detect and confirm the presence of an edge in each individual window. The last step in this process is to locate the edge pixels in the windows where the presence of a front was detected and confirmed. The output of this algorithm is an edge image in which any pixel that is not an edge pixel is set to a digital count of zero. Pixels that are determined to be edge pixels are set to a value equal to the temperature threshold obtained by histogram analysis. Using the indicator function $\Omega(x)$ such that,

$$\forall x \in \mathcal{X},\ \Omega(x) = \begin{cases} 0, & \text{if } x \in \omega_1'; \\ 1, & \text{if } x \in \omega_2', \end{cases} \tag{24}$$

an edge pixel can be simply expressed:

$$\forall x \in \mathcal{X}, \quad \text{if } \exists y \in \mathcal{N}'(x), \text{ such that } \Omega(x) \neq \Omega(y), \Rightarrow x \text{ is an edge pixel.} \tag{25}$$

b. *Local-level processing*

The edge image obtained by window-level processing does not really contain edges, but independent edge pixels. Because oceanographers are interested in studying the statistics associated with a temperature front, the edge image is inadequate, and further processing is required to link the independent edge pixels so that they form continuous contours. For this reason, a contour following algorithm (local-level processing) is included to complete the algorithm for frontal detection. Basically, the algorithm assigns to the nth pixel of the pth contour the value (n, p). Among the edge pixels detected at the window level that are neighbors of contour pixel (n, p), the algorithm selects the pixel that least changes the direction of the contour to become contour pixel $(n + 1, p)$. However, no pixel is added if the contour direction has to change by more than 90° in 5 pixels. When no previously detected edge pixels can be added to the contour, the algorithm examines the ratio of the magnitude of gradient sum to the sum of gradient magnitude in a 3×3 pixel window centered on the last contour pixel. If the ratio is greater than 0.7, one of the pixels in the 3×3 pixel window centered on the last contour pixel is added to the contour. The algorithm selects the pixel for which the scalar product of the gradient vector at that pixel with the gradient vector at the last contour point is maximized.

A few of the problems associated with the edge image obtained by the window-level algorithm can be corrected at the local level by a contour-following algorithm. For example, in windows where more than one front is present, fronts may often be missed. However, portions of the fronts missed in one window can often be detected in neighboring windows. Because the local-level algorithm only relies on the first neighbors of the last edge pixel to find a new edge pixel to add to the existing contour, the algorithm can still follow a particular front even when it comes close to another one. As a result, fronts can often be detected in windows containing more than one front. A second problem often encountered in the edge image is that there may exist isolated edge pixels. This effect is mainly due to the fact that $P(\text{error}) \neq 0$. By imposing a minimum length for a valid contour, isolated points can be removed from the edge image. In the present version of the program, contours containing fewer than 15 pixels are deleted from the list of contours: all contour pixels (n, p) are set to $(0, 0)$ to delete contour p.

3. Validation

a. *Extraction of the Gulf Stream northern edge*

To evaluate the performance of the edge detection algorithm, it was applied to a set of 98 satellite-derived (*NOAA-7*) SST fields used in a previous comparative study (Cornillon and Watts 1987). In the previous study, the location of the northern edge of the Gulf Stream off Cape Hatteras, determined subjectively from satellite imagery by a trained analyst, was compared with the location of the Gulf Stream determined from inverted echo sounders (IES) moored on the sea floor. With each subjective determination the analyst also assigned a quality factor Q_f from 0 (no useful data due to excessive cloud cover) to 4 (completely free of clouds). The position of the Gulf Stream determined from IESs corresponds to the surface projection of the 15°C isotherm at 200 m, T_{15}, the generally accepted definition of the northern edge. A detailed description of the inverted echo sounders can be found in Chaplin and Watts (1984). For the present purpose, it is sufficient to note that three inverted echo sounder lines

were available for the period covered by the images used here (see Fig. 9). Also shown in Fig. 9 is the edge of the Gulf Stream extracted from the edges shown in Fig. 6b. The algorithm designed to extract only those edges associated with the Gulf Stream uses three main criteria. First, only the edges inside the envelope shown in Fig. 9 are considered. Second, the average gradient along a contour must be directed toward the southeast. Third, edges are kept only if the temperatures of the adjacent masses of water are consistent with the temperature in the Gulf Stream region. Note that the extraction algorithm also defines its own objective quality factor Q_f as a function of the temperature of the water masses adjacent to the contour selected. The algorithm is described in more detail in Cayula (1988). To compare the position of the northern edge of the Gulf Stream located by the edge detection algorithm with the IES data, x_{SAT} is defined as the intersection of an IES line with the temperature front (if it exists) located by the edge detection algorithm. Here x_{IES} denotes the location of the intersection of T_{15} with the IES line. Watts and Johns (1982) have estimated that the standard error associated with the IES location of T_{15} compared with the location determined by XBT (expendable bathythermograph) is about 5 km.

To compare the satellite-derived Gulf Stream edge with the IES-derived path, statistics related to the separation of x_{SAT} and x_{IES} are generated. The distance $|x_{SAT} - x_{IES}|$, corrected for the fact that the Gulf Stream does not, in general, cross the IES line orthogonally, is the variable used in the comparison. Because isotherms intersecting the sea surface are not vertical and because the surface projection of T_{15} is being compared with the surface front, a mean offset between the two is expected. This offset is given by

$$m_{SAT-IES_\perp} = \frac{1}{N} \sum_{x_{SAT}/Q_f \geq q} \cos(\theta)(x_{SAT} - x_{IES}). \quad (26)$$

The figure of merit used to evaluate the edge detection algorithm is the scatter about the mean offset given by

$$\sigma^2_{SAT-IES_\perp} = \frac{1}{(N-1)} \sum_{x_{SAT}/Q_f \geq q} [\cos(\theta)(x_{SAT} - x_{IES}) - m_{SAT-IES_\perp}]^2, \quad (27)$$

where θ is the angle between the normal to the Gulf Stream edge and the IES line, and N denotes the number of points that satisfy the minimum quality factor. In obtaining these values it is noted that one IES line may intersect several edges identified in the satellite imagery as being associated with the Gulf Stream. In such cases, only the warmest southernmost intersection is used in the comparison.

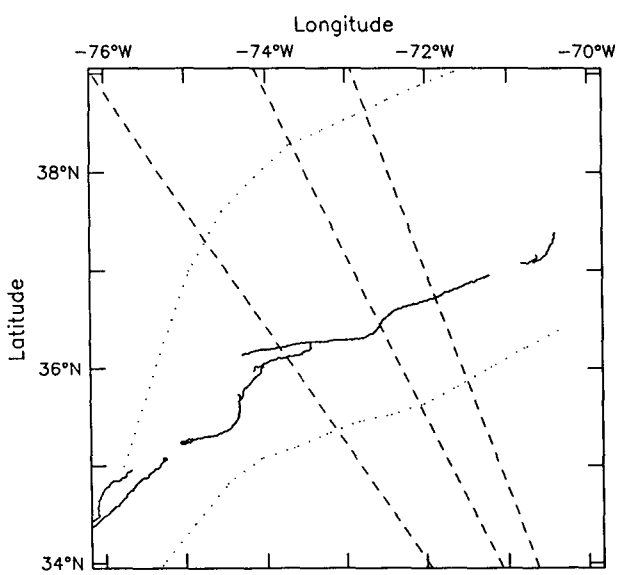

FIG. 9. This figure represents the area shown in Fig. 6. The Gulf Stream northern edge is indicated by a solid line, while the three IES lines are represented by dashes. The Gulf Stream envelope is shown in dots.

In Table 2, the edge detection algorithm (alg), the subjective detection (sub), the gradient method (grd), and the skew method (skw), where the last two methods are objective edge detection algorithms used in the previous study (Cornillon and Watts 1987) are compared. Subjective detection means that an analyst manually located the northern edge of the Gulf Stream. On the other hand, the gradient method is an automated approach that relies on the Sobel operator (Duda and Hart 1973). The skew method consists of computing the skew of the data in a 7 × 7 pixel window and detecting an edge whenever the skew changes sign. Because the various methods operate differently, they assign different quality factors for the same edges: the quality factor is different if the edge is studied globally, regionally or locally. As a result, quality factors for different methods are not directly comparable and Table 2 is structured so that in a particular row, the bias and standard deviation associated with each method are given as a function of the number N of intersections, between IES lines and the Gulf Stream northern edge, which satisfy a minimum quality factor. The variations in the value of N on a particular row are not significant and are simply due to the fact that N is not a continuous function with respect to Q_f.

In comparing the results in Table 2, one notes the following: first, subjective detection leads to the lowest scatter about the mean and as a result, it is still the most accurate method to locate the northern edge of the Gulf Stream from satellite-derived images. However, scatter for temperature fronts obtained with the edge detection algorithm is very close to that obtained subjectively. The other methods lead to $\sigma_{SAT-IES_\perp}$'s which are larger than the $\sigma_{SAT-IES_\perp}$ associated with the subjective method or with the edge detection algorithm. Second, the mean offsets associated with the edge detection algorithm and the subjective method are the

TABLE 2. Comparative results of the edge detection algorithm (alg), the subjective detection (sub), the skew method (skw), and the gradient method (grd). Here N is the number of intersections between the IES lines and the selected Gulf Stream fronts for those edge points that satisfy a minimum quality factor. When the minimum quality factor decreases, N increases since intersection points, which are not as reliable (lower quality factor), are selected.

N				$m_{\text{SAT-IES}_\perp}$				$\sigma_{\text{SAT-IES}_\perp}$			
alg	sub	skw	grd	alg	sub	skw	grd	alg	sub	skw	grd
39	45	49	42	−9.26	−13.32	−21.02	−23.58	14.12	13.29	20.90	21.87
87	80	82	90	−11.08	−13.52	−18.94	−23.97	15.65	13.46	18.40	22.48
110	113	110	109	−12.50	−13.59	−17.37	−24.50	15.61	14.62	18.97	21.92
131	141	131	126	−14.18	−13.80	−16.91	−25.56	17.17	14.50	18.14	22.40

smallest observed. This means that the warmest, southernmost front being detected is further to the south; that is, closer to T_{15}. Indeed, one can often detect two fronts in the region of the Gulf Stream northern edge: a first and weaker front between warmer and warm waters, and a second and stronger front between warm and cold waters. It is the first front that this edge detection algorithm and the analysts are able to detect, while the gradient and skew methods will, in general, detect only the stronger front farther north.

b. *Running time*

The processing time required by the edge detection algorithm varies from image to image. In the set of the 98 images, an average of 30 s of DECstation 5000/200 CPU time was used to process each image.

4. Summary

In this article we have presented an algorithm that is able to detect edges reliably without intervention from a human operator. To accomplish this task the algorithm operated at different levels to detect and differentiate between true and false edges.

For comparison purposes, the algorithm was applied to a test set of 98 sea surface temperature images to detect the northern edge of the Gulf Stream. For this dataset, the algorithm was shown to perform better than automated methods previously used to detect the edge of the Gulf Stream from satellite-derived sea surface temperature fields: the algorithm successfully detected valid temperature fronts and ignored false edges. Furthermore, the cloud-detection portion of the algorithm successfully identified a large number of clouds and, in particular, those that were the most likely to interfere with the edge detection. Finally, the algorithm produced statistics about the temperature fronts that are useful in the subsequent analysis of these fronts. Although the comparative evaluation has been performed on Gulf Stream edges, it is assumed that the algorithm performs equally well on other SST fronts, such as those associated with rings, the subtropical convergence, and shelf-slope fronts.

Acknowledgments. This research project was funded by the National Aeronautics and Space Administration under Contract NAGW-858.

APPENDIX A

The Data—Preprocessing and Cloud Identification

The algorithm was developed for application to SST fields. These fields were obtained from the AVHRR/2, an infrared sensor flown on the *TIROS-N* series of polar-orbiting satellites (Schwalb 1978).

Prior to application of the edge detection algorithm to the infrared data, several preprocessing steps were performed to obtain SST fields. These steps are described in detail in Cornillon et al. (1987) and that discussion will not be reproduced here. We simply note that the result of these steps are SST fields in a common coordinate system with a rms uncertainty for sea surface temperature in cloud-free regions of approximately 0.7°C. We also note that infrared sensors operating in the thermal portion of the spectrum cannot "see through" clouds. Hence, clouds represent a special problem in the analysis of the data and, in particular, in the operation of the edge detection algorithm discussed herein. For this reason, additional cloud detection preprocessing steps at the picture level and window level are applied to the data prior to edge detection.

a. *Cloud identification at the picture level*

Processing at the picture level is accomplished in four basic steps. First, clouds are usually colder than the underlying sea surface. Therefore, a simple temperature thresholding is applied to the image to flag cloudy regions. Second, cloudy regions are often characterized by a high gradient magnitude. The algorithm therefore also applies a threshold to the temperature gradient to achieve a rough segmentation of the data between noncloudy and possibly cloudy regions. The next two steps are used to refine the segmentation resulting from the thresholding; that is, to determine whether those regions flagged as possibly cloudy are indeed cloudy. In effect this results in the algorithm being less dependent on the threshold values. Unlike

a real edge, gradient vectors associated with pixels inside a cloudy area do not have a coherent direction. Regions flagged as possibly cloudy, for which the ratio of the magnitude of gradient sum to the sum of gradient magnitude is lower than 0.3, are therefore assumed to consist of clouds. On the other hand, for a ratio greater than 0.7, the region is flagged as clear. Finally, because clouds often form bulky shapes while edges produce elongated profiles, a threshold is applied to the aspect ratio of the regions flagged that have a gradient ratio between 0.3 and 0.7. An aspect ratio (larger eigenvalue of the spatial covariance matrix divided by smaller eigenvalue) that is more than 6 indicates a clear region.

Note that land can also lead to false temperature fronts. However, for the images used in this study, the location of land is well defined; hence, edges detected at the shoreline are easily identified as land–water edges by their location in the image. To avoid detecting the shoreline as an edge, land regions in the image can be masked before the edge detection algorithm is applied.

The final step before window-level processing involves filtering the original image with a 3×3 median filter (Huang 1981) and then removing the clouds detected by the previous algorithm. The cloudy regions can be removed from the median-filtered image by logically multiplying (logical AND) the image and the cloud map.

b. *Cloud identification at the window level: Correlation algorithm*

The idea behind this portion of the algorithm stems from the observation that cloudy areas, and to a lesser extent land areas, are regions that usually have high variability in temperature when compared with water areas. The problem, when trying to remove clouds, is that fronts (the features to be detected) are also associated with high variability.

In order to separate high variability due to clouds and high variability due to edges, the image is segmented into overlapping windows (see beginning of section 2a) and the histogram analysis (section 2a.1) is applied. If the histogram in a window is found to be bimodal, then parameters estimated by the histogram analysis define two populations for that window and the variability inside each population can be analyzed separately from the edge effect. Such an algorithm has the property of refining the result obtained by the cloud detection algorithm at the picture level (i.e., better detection of clouds near edges and of small clouds).

A measure of temperature variability can be obtained by estimating the autocorrelation or autocovariance functions, but this method would be computationally expensive and, to fully use its results, complex models of the autocorrelation functions for every type of population would have to be defined (Kittler and Pairman 1985).

The approach taken here is much simpler: the average of the absolute differences is used. To eliminate the possible linear trend of the data, the absolute value of the expected value is subtracted:

$$\gamma_X = E(|x - y|) - |E(x - y)|,$$
$$\forall (x, y) \in X^2 \quad \text{such that,} \quad y \in \mathcal{N}'(x), \quad \text{(A1)}$$

where X is either population ω_1' or population ω_2'.

If the correlation between neighbors is close to unity, or if the variance is not too large, (low variability area) then γ_X will be small. On the other hand, if high variance and low correlation are observed (high variability area) then γ_X will take a value close to the estimated variance of the population (which is high). As a result, a decision threshold can be defined so that if γ_X is larger than the threshold, then the population is marked as a cloudy area. Because cloud detection depends on the value of the population variance, it is not temperature-scale invariant. The algorithm would have to include a much more sophisticated analysis of the clouds (such as making use of the correlation function, Markov random fields or fractals) to obtain this property of temperature-scale invariance. However, although the present cloud detection algorithm is not scale invariant, it is translation invariant in the temperature domain. For a given image the average temperature of a cloud does not affect the probability of detecting a cloud.

After running the program on a number of images, it was decided to use two thresholds. Because clouds are usually colder than their surroundings, when two populations are present in the area, the cold population is more likely to be a cloudy area than the warm population. As a consequence, a lower decision threshold of 4 was selected for γ_X associated with the cold population, while a higher threshold of 8 was chosen for γ_X associated with the warm population.

APPENDIX B

Determination of the Cohesion Threshold

The algorithm that examines bimodal distributions for cohesion makes use of threshold values on the cohesion coefficients defined in the body of this paper. In this appendix, we derive the threshold values when the noise in the image is independently distributed. Note that although the hypothesis of independence is not always verified for actual SST images, the theoretically derived threshold values match those obtained through experiments.

Note that ω_1' and ω_2' are the populations obtained by thresholding the temperature field [Eq. (18)], while ω_1 and ω_2 are the actual underlying populations. If we define P_{e_1} as the probability of misclassifying a sample belonging to ω_1, and P_{e_2} as the probability of misclassifying a sample belonging to ω_2:

$$P_{e_1} = P(x \in \omega_2' \mid x \in \omega_1) \quad \text{and}$$
$$P_{e_2} = P(x \in \omega_1' \mid x \in \omega_2), \tag{B1}$$

we can express the probability associated with ω_1' and ω_2':

$$P(\omega_1') = P(\omega_1)(1 - P_{e_1}) + P(\omega_2)P_{e_2}, \tag{B2}$$

$$P(\omega_2') = P(\omega_2)(1 - P_{e_2}) + P(\omega_1)P_{e_1}. \tag{B3}$$

If only one population is actually present, Eqs. (B2) and (B3) still hold, with either $P(\omega_1) = 1$ and $P(\omega_2) = 0$ or $P(\omega_1) = 0$ and $P(\omega_2) = 1$.

More generally, assuming that the noise added to the image is a random process, the expected value of the cohesion coefficients can be determined as a function of $P(\omega_1)$ and $P(\omega_2)$ and P_{e_1} and P_{e_2}. These relations will help define the value of the cohesion coefficients that indicate a reliable segmentation. To obtain Eqs. (B4) and (B5) we assume that a pixel from ω_1 and a pixel from ω_2 cannot be neighbors. The deterministic effect on the cohesion coefficients due to the presence of an edge, where the two populations are in contact, is reintroduced later:

$$C_1 = \frac{P(\text{center} \in \omega_1' \text{ and, neighbor} \in \omega_1')}{P(\omega_1')},$$

$$C_1 = \frac{P(\omega_1)(1 - P_{e_1})^2 + P(\omega_2)P_{e_2}^2}{P(\omega_1)(1 - P_{e_1}) + P(\omega_2)P_{e_2}}. \tag{B4}$$

$$C_2 = \frac{P(\text{center} \in \omega_2' \text{ and, neighbor} \in \omega_2')}{P(\omega_2')},$$

$$C_2 = \frac{P(\omega_1)P_{e_1}^2 + P(\omega_2)(1 - P_{e_2})^2}{P(\omega_1)P_{e_1} + P(\omega_2)(1 - P_{e_2})}. \tag{B5}$$

The expected value of the global cohesion coefficient can be obtained as the weighted average of C_1 (B4) and C_2 (B5) or from its fundamental definition:

$$C = P(\text{center and neighbor from the same population}),$$

$$C = P(\omega_1')C_1 + P(\omega_2')C_2,$$

$$C = 1 - 2[P(\omega_1)P_{e_1} + P(\omega_2)P_{e_2}] + 2[P(\omega_1)P_{e_1}^2 + P(\omega_2)P_{e_2}^2]. \tag{B6}$$

When only one population, for example ω_1, is present, $P(\omega_1) = 1$ and $P(\omega_2) = 0$. As a result, from Eq. (B2), $P(\omega_2') = P_{e_1}$ and if the noise is a completely random process, the global cohesion coefficient C becomes $[1 - 2P(\omega_2') + 2P(\omega_2')^2]$. For the condition imposed by Eq. (14), $P(\omega_2') \geq 0.25$, we find $C \leq 0.625$. Usually, the noise is not completely random and even though a front does not exist in the area, C can be found to be greater than 0.625. However, if we assume that a front is actually present under normal conditions P_{e_1} and P_{e_2} should be small with respect to unity and the quadratic terms may be ignored. Furthermore, Eq. (2) can be used in Eq. (B6) to obtain a simple approximation for the expected value of the global cohesion coefficient:

$$C \simeq 1 - 2P(\text{error}). \tag{B7}$$

Although the pixels physically located at the edge between populations ω_1 and ω_2 have been ignored, their effect on the value of the cohesion coefficients is not completely negligible. For a window of $n \times n$ pixels, containing a straight edge that crosses the window diagonally from corner to corner, the value of the global cohesion factor is decreased by n^{-1}. In particular, for a window of 32×32 pixels, C is decreased by $1/32 \simeq 0.03$:

$$C \simeq 0.97 - 2P(\text{error}). \tag{B8}$$

While studying the histogram based algorithm in the case of normally distributed data, it was noted that a signal-to-noise ratio of at least 4 was needed to obtain reliable results. This condition is also equivalent to a maximum tolerable probability of error, $P(\text{error})$, of approximately 0.025. By replacing $P(\text{error})$ by its limiting value in Eq. (B8), it becomes apparent that the estimate of C must be greater than 0.92 to obtain a reliable segmentation. Consequently, in this program, any segmentation that yields a value of C lower than 0.92 is discarded as unreliable. Similar results are obtained for C_1 and C_2. However, it should be noted that the effects on C_1 and C_2 due to the edge depend on the size of each population. Furthermore, because the estimates of the partial cohesion coefficients are computed with fewer samples, they are less reliable than C. As a result, a larger allowance was made for possible edge effects:

$$C_1 \simeq 0.95 - 2P_{e_1}, \tag{B9}$$

$$C_2 \simeq 0.95 - 2P_{e_2}, \tag{B10}$$

and the minimum value of C_1 and C_2, to accept the hypothesis of a front, was chosen to be 0.90.

REFERENCES

Ballard, D. H., and C. M. Brown, 1982: *Computer Vision.* Prentice-Hall, 523 pp.

Cayula, J.-F., 1988: Edge detection for SST images, M.S. thesis, Department of Electrical Engineering, University of Rhode Island, 91 pp.

——, P. Cornillon, R. J. Holyer, and S. H. Peckinpaugh, 1991: Comparative study of two recent algorithms designed to process sea-surface temperature fields. *IEEE Trans. Geosci. Remote Sens.*, **GE-29**, 175–177.

Chaplin, G., and D. R. Watts, 1984: Inverted echo sounder development. *Ocean '84 Conference Record,* vol. 1, 249–253.

Cornillon, P., and D. R. Watts, 1987: Satellite thermal infrared and inverted echo sounder determination of the Gulf Stream northern edge. *J. Atmos. Oceanic Technol.,* **4,** 712–723.

——, C. Gilman, L. Stramma, O. Brown, R. Evans, and J. Brown, 1987: Processing and analysis of large volume of satellite-derived thermal infrared data. *J. Geophys. Res.,* **92,** 12 993–13 002.

Duda, R. O., and P. E. Hart, 1973: *Pattern Recognition and Scene Analysis.* John Wiley and Sons, 482 pp.

Gerson, D. J., and P. Gaborski, 1977: Pattern analysis for automatic location of ocean fronts in digital satellite imagery. U.S. Naval Oceanographic Office, TN 3700-65-77, 67 pp.

Holyer, R. J., and S. H. Peckinpaugh, 1989: Edge detection applied to satellite imagery of the ocean. *IEEE Trans. Geosci. Remote Sens.,* **GE-27,** 46–56.

Huang, T. S., 1981: *Two-Dimensional Digital Signal Processing II: Transforms and Median Filters.* Springer-Verlag, 222 pp.

Kittler, J., and D. Pairman, 1985: Contextual pattern recognition applied to cloud detection and identification. *IEEE Trans. Geosci. Remote Sens.,* **GE-23,** 855–864.

——, and J. Illingworth, 1986: Minimum error thresholding. *Pattern Recognition,* **19,** 41–49.

Schwalb, A., 1978: The *TIROS-N/NOAA A-G* satellite series. National Environmental Satellite Service, NOAA, U.S. Dept. of Commerce, 77 pp.

Watts, D. R., and D. R. Johns, 1982: Gulf Stream meanders: Observations on propagation and growth. *J. Geophys. Res.,* **87,** 9467–9476.

Weszka, J. S., 1978: A survey of threshold selection techniques. *Comput. Graph. Image Proc.,* **7,** 259–265.