

University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

CSE Conference and Workshop Papers

Computer Science and Engineering, Department
of

2008

An Automatic Bridge Detection Technique for Multispectral Images

D. Chaudhuri

Defence Electronics Application Laboratory

Ashok Samal

University of Nebraska-Lincoln, asamal1@unl.edu

Follow this and additional works at: <https://digitalcommons.unl.edu/cseconfwork>



Part of the [Computer Sciences Commons](#)

Chaudhuri, D. and Samal, Ashok, "An Automatic Bridge Detection Technique for Multispectral Images" (2008). *CSE Conference and Workshop Papers*. 29.

<https://digitalcommons.unl.edu/cseconfwork/29>

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Conference and Workshop Papers by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

An Automatic Bridge Detection Technique for Multispectral Images

D. Chaudhuri and Ashok Samal, *Member, IEEE*

Abstract—Extraction of features from images has been a goal of researchers since the early days of remote sensing. While significant progress has been made in several applications, much remains to be done in the area of accurate identification of high-level features such as buildings and roads. This paper presents an approach for detecting bridges over water bodies from multispectral imagery. The multispectral image is first classified into eight land-cover types using a majority-must-be-granted logic based on the multiseed supervised classification technique. The classified image is then categorized into a trilevel image: water, concrete, and background. Bridges are then recognized in this trilevel image by using a knowledge-based approach that exploits the spatial arrangement of bridges and their surroundings using a five-step approach. A river extraction module identifies the rivers using a recursive scanning technique and geometric constraints. Using a neighborhood operator and the knowledge of the spatial dimensions of a typical bridge, we identify the possible bridge pixels. These potential bridge pixels are then grouped into possible bridge segments based on their connectivity and geometric properties. Finally, these bridge segments are verified on the basis of directional water index along different directions and their connectivity with the road segments. The approach proposed in this paper has been implemented and tested with images from the IRS-1C/1-D satellite that has a spatial resolution of 23.5×23.5 m. The results show that this approach is both efficient and effective in extracting bridges.

Index Terms—Classification, feature extraction, graph theory, multiseed clustering, pattern recognition, remote sensing.

I. INTRODUCTION

AUTOMATIC detection of geographical objects such as bridges, big buildings, or road crossings in satellite images is useful in many important applications. These applications include maintaining geographical databases, assessing the extent of damages in case of natural disasters such as floods or earthquakes, and military applications. Extraction of high-level features is also important in the development of content-based indexing schemes for satellite images [14], [21].

Many approaches to detect linear features have been reported in computer vision, remote sensing, and photogrammetry literature. These include the gradient direction profile analysis method [23], adaptive template matching [10], snake models [7], [8], [12], [13], perceptual grouping [11], the hyperspectral

approach [18], multiscale or multiresolution approaches [5], [13], [24] and GIS-data-guided methods [7], [12]. The concept of road extraction is relatively simple, but developing reliable methods remains a difficult challenge. As Fortier *et al.* [6] concluded: “Due to its importance, much effort is devoted to finding solutions to this problem. Unfortunately, no existing methods yet allow a complete and robust automation of the process.”

We briefly describe the few approaches proposed for detection of bridges in remotely sensed data. Houzelle and Giraudon [9] suggest a data fusion technique using SPOT and synthetic aperture radar (SAR) images for bridge detection and urban-area extraction. For bridge detection, they first segment water in the SPOT image to spatially constrain the location of bridges in the SAR image. Mandal *et al.* [15] use a method of analyzing the IRS satellite imagery for detecting various man-made objects including bridges. Initially, a multivalued recognition system is used to classify the image pixels into six land-cover types. In order to identify the targets, knowledge of their characteristics and their interrelationship is incorporated into the clustered image using heuristic rules. Trias-Sanz *et al.* [22] suggest techniques to automatically detect bridges on small high-resolution panchromatic satellite images that rely on radiometric features (texture information) and geometric models. Using neural networks, they classify each pixel into several terrain classes. Although the approach is effective, there are several drawbacks to extending it to the general case. First, computation of texture parameters for a large image takes a significant amount of time. In addition, the analysis windows span more than one texture at the boundaries of texture regions and, therefore, give imprecise classifications. This technique also is unlikely to extract bridges over larger regions such as big rivers.

This paper focuses on a method to extract bridges that divide water bodies in multispectral images. We define a bridge as a narrow concrete structure between two disparate water bodies, which is connected to a road. Concrete is considered to be a broad class of material that comprises the foundation for buildings, roads, railroads, and metallic structures. Initially, majority-must-be-granted logic based multiseed supervised classification technique [16] is used to classify the image pixels into eight land-cover types. The classifier is trained using a database of spectral information of different terrain classes from multispectral satellite sensor images. The eight-class image is then converted into a trilevel image: water, concrete, and background. Because bridges on rivers are important targets in many tactical applications, identification of rivers is a central step in extracting them. Separation of river and nonriver parts in the image is done by using a recursive scanning technique that exploits some geometric constraints such as area, perimeter, and elongatedness. A low-level operator is then

Manuscript received June 20, 2007; revised August 28, 2007.

D. Chaudhuri is with the Defence Electronics Application Laboratory, Dehradun, Uttaranchal 248001, India.

A. Samal is with the Department of Computer Science and Engineering, University of Nebraska—Lincoln, Lincoln, NE 68588 USA (e-mail: samal@cse.unl.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TGRS.2008.923631

used to identify possible bridge pixels. This operation is based on the following observations: 1) the bridge on a water body is linear, and 2) the width of bridges is relatively small. The bridge pixels are then grouped to determine the final location and span of the bridges. We have evaluated our approach with real IRS-1C/1-D satellite images.

We briefly compare our approach with the few other approaches proposed in literature. For the bridge detection method proposed by Houzelle and Giraudon [9], both SPOT and SAR images are required. Thus, an additional step to register both images is necessary. Sometimes, it is difficult to collect the ground control points, resulting in reduced accuracy of the registration process. In addition, the availability of both sensor images for the same time period is questionable. The approach proposed here requires a single satellite image in the multispectral mode. Hence, there is no additional registration step. The method proposed by Mandal *et al.* [15] uses a contour tracing method. Misclassification results in many nonbridge segments to be detected as bridges. In addition, the approach is computationally more expensive. Unlike the approach proposed here, there is no confirmation test for small segments to be labeled as bridges. The approach proposed in this paper exploits geometric constraints of bridges to distinguish them from other geometric configurations. The novel decision-making logic is another advantage to confirm the presence of small bridges, resulting in reduced false-alarm rates.

The main contributions in the proposed method are as follows: 1) a tunable algorithm to identify candidate bridge pixels using a simple Boolean and/or logical low-level operator within a $d \times d$ window; 2) a new merger-separator operation that exploits geometric constraints to group the possible bridge pixels into small bridge segments; and 3) a novel decision-making logic for confirmation of bridges with low false-alarm rate.

II. ROAD EXTRACTION

In this paper, we define a bridge on the water body as a narrow concrete structure between two disparate water bodies, which is connected to a road. Using the connectivity constraint of the bridge with road will reduce the false-alarm rate. Therefore, road extraction is used as a part of our approach to detect bridges.

Road networks are important features in satellite imagery, which are characterized by unique spectral and spatial properties. However, low-level feature extraction is not always sufficient to accurately identify the roads in a satellite image. Incorporation of high-level knowledge is a necessary step in the process of road extraction. Our road extraction method is based on the work described in [4] and [17] and is designed to take advantage of certain characteristics of the IRS-1C/1-D multispectral images. We assume that the visible road structures are piecewise linear, 1–5 pixels wide, darker than their surroundings, and are nearly homogeneous in their appearance.

Many of the characteristics are either directly applicable or can be easily extended for other types of imagery. The process of road extraction can be divided into the following three major steps: line enhancement, segmentation, and linking. The enhancement process uses only local operations in which linelike structures get enhanced while the edges get suppressed, thereby avoiding false alarms. During the segmentation process, non-

road pixels are successively eliminated from the image. At this point, only the road pixels remain for processing. However, some road pixels may have been eliminated. Therefore, a linking step is used to recover the eliminated road pixels.

III. MULTISEED SUPERVISED CLASSIFICATION TECHNIQUE

Like most remote sensing applications, we begin our analysis with an initial ground-cover classification. Supervised classification is widely used to accomplish this task. One of the critical issues that directly affects the accuracy of classification is the choice of the training set. The bridges considered here are essentially concrete structures that act as dividers of water bodies. Thus, it is important to extract concrete and water from a multispectral image. Many supervised classification algorithms, e.g., minimum-distance classifiers, use the mean of the training samples as the representative (seed) of a particular class. The main problem with seed-point-based algorithms is that they are not appropriate for classes that are not spherical in shape and are unlikely to yield good results for complex and elongated shapes [16]. In such cases, using multiple seeds to represent a class is likely to be more effective. This is particularly true for satellite images, where the variation within a class may be high. We can treat a nonhomogeneous class as the union of many homogeneous classes [16] and represent it by a seed from each of the constituting homogeneous classes. This approach will be more effective and will result in increased accuracy of classification. We have demonstrated the effectiveness of a multiseed-based classifier and have compared it to many other classifiers [16]. The algorithm used in this paper has the following three main steps: 1) identification of seed points; 2) parameter estimation; and 3) classification.

A. Seed Point Detection

The central idea is to eliminate outliers from the training sample to arrive at the central points of the clusters. The number of seed points needed for accurate training depends on the shape of the class. We have shown that a single seed cannot accurately classify elongated data (classes with large gray-value variation), and using multiple seeds is an effective way to solve this problem [1], [2], [16]. The seed point depends on the highest density, and this paper [3] proves that the estimated density is consistent and asymptotically unbiased. The training data of each class are collected by the guidance of the topographic maps and by visiting sites.

B. Statistical Parameter Extraction Algorithm

After extracting the seed points of each class in the training data, we determine the statistical parameters for each class. The main idea is to identify the key subclusters in each class and determine their first-order statistics: maximum, minimum, mean, median, and mode for the gray levels. We use a k -means clustering approach to determine the clusters.

C. Minimum-Distance Logical-Based Classifier

Using the statistical parameters for each class, we can classify the unknown pixels of any image. Our classification algorithm uses the following two types of logic:

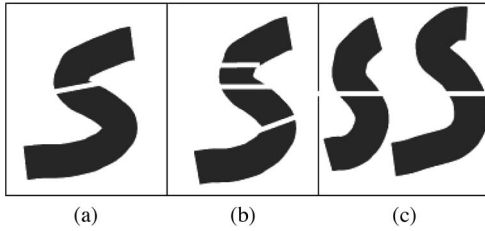


Fig. 1. Different organization of bridges on the rivers.

1) minimum-distance-of-majority-must-be-granted logic and 2) minimum-of-minimum-distance-granted logic. Both are designed for multiseed classification and improve the accuracy of classification [16]. In both cases, we first compute the distance between the pixels to be classified and then each of the subclusters of each class. In majority-must-be-granted logic using the minimum distances, a class is assigned to an unknown pixel if three of the five minimum distances (min, max, mean, median, and mode) correspond to the same class. In minimum-of-minimum-distance-granted logic, we assign a pixel to a class whose minimum distance among mean, median, mode, max, and min is the smallest.

IV. BRIDGE DETECTION TECHNIQUE

In this paper, we define a bridge as a narrow concrete structure that connects two water bodies and is connected to a road. Due to the difficulty in collecting a large number of samples, we consider concrete as a broad class that includes buildings, roads, railroads, and metallic structures. In addition to the spectral properties, we also exploit many geometric constraints of and world knowledge about bridges. These constraints include the following.

- 1) A bridge should be a concrete structure.
- 2) A bridge should be a straight line on the water body.
- 3) A bridge acts as a divider of the water body; it is surrounded by water on both sides.
- 4) A bridge should have constant width; the actual width depends on the sensor resolution.

Fig. 1 shows different cases of bridges on the water body (river). Case 1 [Fig. 1(a)] occurs when there is only one bridge on the river. Case 2 [Fig. 1(b)] arises when there is more than one bridge on the river. The bridges may be parallel or nonparallel. Case 3 [Fig. 1(c)] occurs when there is more than one river in the image and the bridges lie on a straight line. Note that the bridges across a series of rivers do not have to be on a straight line. Fig. 1(a) handles the scenario when they are not, and Fig. 1(c) shows the scenario when they are. Our algorithm can extract bridges from all such cases including the case of an island.

The bridge detection algorithm is performed in four steps. We first extract river segments in the image. We then identify possible pixels that can be part of bridges. The candidate bridge pixels are then grouped to form bridges. Finally, a verification step validates the identified bridges. Each of these four steps is described in detail hereafter.

A. River Extraction Technique

We begin by classifying a scene into eight different land-cover classes using the multiseed supervised classification

technique. The eight classes are as follows: ice/snow, shrubs, concrete, water, sand, forest, soil, and rock. For bridge detection, concrete and water are the most important classes. Therefore, we convert the eight classes in the classified image into three classes by merging the rest of the classes into one class called “background.” After this step, each pixel in the image is labeled as water, concrete, or background.

Due to noise or other problems, some bridge pixels may have been classified as water pixels. Similarly, a few water pixels may have been misclassified as bridge, resulting in many small regions that are not significant. We fix this by removing the regions whose area falls below a predefined threshold (5 pixels in our experiments). This makes a computation significantly faster.

Segmentation is the process of grouping similar pixels into regions. The result of segmentation is a set of disjoint regions that covers the image. Many approaches have been used in literature, including classical blob coloring as well as split and merge [20]. Because our image is trilevel (water = 1, concrete = 2, and background = 0), we propose a specialized recursive scanning algorithm that is fast and efficient for our application. For recognition of rivers, we only consider the pixels that are labeled 1 (water) and ignore the rest.

Recursive Scanning Method: We begin by selecting a pixel with label 1 (water) and label it as “marked.” The procedure simultaneously looks for all other pixels in its 3×3 neighborhood and marks them if they are also water pixels. This is a variation of the region growing method, but growing and marking take place in all directions within a 3×3 window simultaneously. Once the pixel is “marked,” it will not be considered in the search procedure. Therefore, this method is very fast. Unlike many other growing methods, this method is independent of a starting point, and there is no requirement to select suitable properties for including points in the various regions during the growing process, except to check for water pixels. The procedure terminates when all the water pixels are “marked.”

After water regions are found, we compute some shape features. Our goal is to identify rivers, i.e., elongated water bodies. Three shape parameters, namely, area, perimeter, and elongatedness, are used to find elongated water bodies. Area and perimeter are computed in a straightforward manner. To compute the elongatedness, we first compute the center of the region. If the center of the region does not lie inside the region, we choose the nearest point to be the center. We then compute the maximum and minimum distances from the center to the boundary points of that region. We define elongatedness as

$$\text{elongatedness}(E) = d_{\max} - d_{\min}$$

where d_{\max} and d_{\min} are the maximum and minimum distances from the center to the boundary of the region, respectively. If d_{\min} is zero, then we take the average of the 5% of the nearest pixels to the region boundary. We classify a region as a river if the area, perimeter, and elongatedness all exceed predefined thresholds, which are determined during training.

B. Candidate Bridge Pixel Extraction Technique

The next step in our analysis is to identify the bridge pixels. We view a bridge as a narrow concrete structure that divides rivers in the image. We identify candidate bridge pixels using a

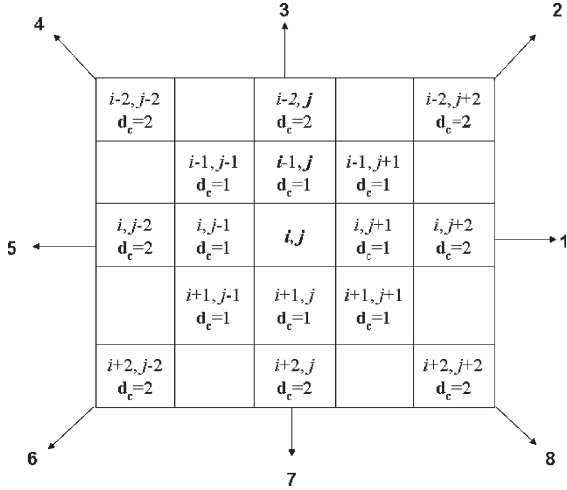


Fig. 2. Possible bridge pixel extraction window.

simple Boolean and/or logical low-level operator within a $d \times d$ window. Direction of small elongated objects is something that is generally difficult to estimate due to the discrete nature of the image. Looking for pixels in all directions can be very expensive. While the window size depends on the spatial resolution of the sensor, our experiments showed that considering only four directions (horizontal, vertical, right diagonal, and left diagonal) for extraction of possible bridge pixels provides a good balance between computational efficiency and extracting the bridge pixels accurately.

Fig. 2 shows a 5×5 window along with the eight directions. We consider only the $[1 \leftrightarrow 5]$, $[2 \leftrightarrow 6]$, $[3 \leftrightarrow 7]$, and $[4 \leftrightarrow 8]$ directions. It also shows the chessboard distance between the center pixel and the pixels in the window. The chessboard distance d_c between two q -dimensional points $X = (x_1, x_2, \dots, x_q) \subseteq R^q$ and $Y = (y_1, y_2, \dots, y_q) \subseteq R^q$ is defined as

$$d_c(X, Y) = \max_{i=1}^q |x_i - y_i|.$$

Based on the chessboard distance, we define a layered neighborhood for a pixel. Because every direction has two senses, positive and negative, a pixel has two immediate neighbors along a particular direction at a particular distance. If we choose a $d \times d$ (d is odd) window for our analysis, we can only identify bridges of thickness 1 to $d - 2$. The value of d depends on the width of the bridges and the spatial resolution of the sensor. The number of layer neighbors required to develop a logical low-level operator within a $d \times d$ window is $(d - 1)/2$. The basic logic for extracting the possible candidate bridge pixel within a window of size $d \times d$ along a horizontal direction is given hereafter. The same logic applies for the other three directions, but adjusting for the orientation. For a candidate pixel (i, j) to be labeled as a bridge pixel, the following conditions must hold in any one of the four directions (see Fig. 2).

- 1) The candidate pixel must be concrete.
- 2) At least one pixel in the all layer- k , ($k < (d - 1)/2$), neighborhood must be concrete.
- 3) All layer- k , where $k = (d - 1)/2$, neighbor pixels must be water.

It should be noted that if $d = 3$, then $k < 1$, i.e., $k = 0$. Thus, a concrete pixel may not have any neighbor to be labeled as a

bridge pixel. Therefore, we can detect bridges of thickness 1 also. By considering the broad definition of the concrete class and the complexity of the scenes, some nonbridge pixels may get classified as bridge pixels. They often appear as isolated pixels, which can be removed easily by using a 3×3 mask or through a morphological operation.

C. Grouping Bridge Pixels to Form Bridge Segments

Once a set of bridge pixels is identified, we group them into contiguous sets that correspond to bridge segments. We begin the grouping operation by using the recursive scanning algorithm (described in Section IV-A). Because the number of regions extracted is typically more than the actual number, a merging step is required.

Merger-Separator Operation Technique: Segmentation of bridge pixels to form bridge segments does not usually result in perfect groupings. A single bridge structure may be broken into multiple segments due to the noise in the original image or misclassification of constituent pixels. These multiple segments, often separated by a small distance, should be considered for merging. The following two factors must be considered during any merging process: 1) the distance and 2) the angle between the segments. All segments of a single bridge, when combined, should form a linear structure with two parallel edge lines. Using these principles, we have the following rules.

- 1) If a set of segments aligns on a straight line and is proximate, it must be merged.
- 2) If a set of segments is proximate but does not form a linear structure, it must be kept separate.

We have developed a merger-separator operation that exploits these distance and orientation constraints. Suppose there are m -segmented regions that need to be analyzed for merger-separator operation. Therefore, there are ${}^m C_2 (= m \times (m - 1)/2)$ combinations of regions that are checked. If m is large, the merger-separator operation takes an unacceptable amount of time. To reduce the complexity of the problem, we use the minimum spanning tree (MST) [1], [2].

Definition: Let S_1 and S_2 be the two sets of pixels, where $S_1 = \{(x_{1i}, y_{1i}), i = 1, \dots, n_1\}$ and $S_2 = \{(x_{2j}, y_{2j}), j = 1, \dots, n_2\}$. The distance $D(S_1, S_2)$ between two groups is defined as

$$D(S_1, S_2) = \min_{\substack{i=1,2,\dots,n_1 \\ j=1,2,\dots,n_2}} \{d[(x_{1i}, y_{1i}), (x_{2j}, y_{2j})]\}$$

where $d(X, Y)$ is the Euclidean distance between the two points X and Y .

Assume that there are C groups of bridge segments. Let $(x_{j,i}, y_{j,i}), i = 1, \dots, n_j$ be the n_j points of j th group of segments, where $j = 1, 2, \dots, C$. The centroid of the j th group, (\bar{x}_j, \bar{y}_j) , is given by

$$\bar{x}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} x_{j,i} \quad \bar{y}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} y_{j,i}, \quad j = 1, \dots, C.$$

We begin our merger-separator operation with the computation of the centroids of each segment. Using the centroids as nodes in a graph, we compute the MST. The Euclidean distance between two connecting nodes is used as weights for the edges in the graph. MST gives the connectivity of these

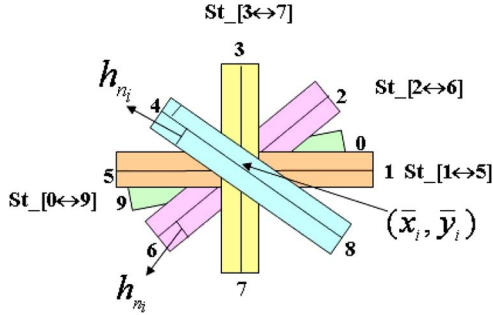


Fig. 3. Possible directional strips for computing the directional water index.

centers. A pair of segmented regions is considered for the merger–separator operation only if their centroids form an edge in the MST. The MST of the centroids of the segmented regions preserves the connectivity while reducing the complexity of the merger–separator operation.

To determine if two segments should be merged, we first determine the lines that best fit each group of pixels. This is done by using a standard least squares regression technique that minimizes the square error [19]. Let θ_j and θ_k be the angles of inclination of the best-fit lines of the j th and k th groups, respectively. If the difference between the angles is large, i.e., $(|\theta_j - \theta_k| > \alpha)$, the two groups are different. If $(|\theta_j - \theta_k| \leq \alpha)$, either the two groups lie in a straight line or the two groups are parallel to each other. They can be distinguished as follows. We first compute the angle of the line that connects the centers of the j th and k th groups. Let θ_{jk} be the angle of the line joining the two centers. Compute the two angle difference parameters

$$\theta_{\min} = \min(|\theta_{jk} - \theta_j|, |\theta_{jk} - \theta_k|).$$

If $|\theta_{\min}| \leq \delta$, a small predefined threshold, then the j th and k th groups approximately lie in a straight line and should be checked for proximity. Otherwise, the two groups are parallel and should not be merged. If they lie in a straight line, they are merged if the distance between them, d_{ms} , is small.

Decision Logic Regarding Directional Water Index and Connectivity: The bridges recognized in the previous step must be verified using the presence of surrounding water. We use the concept of directional water index to verify that an identified structure is a bridge. Directional water index is defined as the number of water pixels within a strip along a particular direction. Water surrounds the bridge in all directions, except along the direction of the bridge. Thus, the water index along the bridge direction will be less than along any other direction. We consider five directions to analyze from the center of a possible bridge for computation of the water index, as shown in Fig. 3. They are horizontal (1 ↔ 5), right diagonal (2 ↔ 6), vertical (3 ↔ 7), left diagonal (4 ↔ 8), and the bridge direction (0 ↔ 9). Fig. 3 also shows the center of the bridge and strips of width $2h_{n_i}$ along all directions.

One can increase the number of directions, but it will increase the computational cost. For the confirmation of a bridge, we test the water index in different directions. First, we compute the direction of the segmented region by determining the direction of the best-fit line of the pixels in the segmented region. This segment will be confirmed as a bridge if the directional water index along this direction is less than the index along other directions.

The selection of the width of a strip is critical in the computation of the directional water index in that direction. The perfect width of the strip is half of the bridge width. If the width is very large, too many water pixels may be included along the bridge direction. On the other hand, if it is very small, the water index along all directions will be low, and it will be difficult to verify the bridge. We estimate the width of the strip by using a data-driven procedure described in [3] and given by

$$h_{n_i} = \left(\frac{l_i}{n_i} \right)^{1/2}, \quad i = 1, 2, \dots, m$$

where h_{n_i} is the half-width of the strip, n_i is the number of points in the bridge group, and l_i is the sum of the edge weights (Euclidean interpoint distance) of the MST of the bridge group.

In order to make this more efficient, we compute this only around the bridge pixels that are within a certain distance. The search region of water body is a collection of water pixels within an open disk, with the center being located at the centroid of a bridge segment. The radius of the open disk is called a search radius (SR) and is given by

$$SR_i = h_{n_i} \times gd_{\max}$$

where SR_i is the search radius for the i th bridge segment, h_{n_i} is the estimate of the half bridge width, and gd_{\max} is the maximum distance between any two bridge pixels in the same segment and is given by

$$gd_{\max} = \text{Max}_{i=1,2,\dots,m} \left[\text{Max}_{\substack{j,k=1,2,\dots,n_i \\ j \neq k}} d(p_{i,j}, p_{i,k}) \right]$$

where m is the total number of bridge segments and d is the Euclidean distance between two points in the same bridge segment. It should be noted that this is a global max and is not defined with respect to any bridge segment. Therefore, it is maximized over all the bridges. Using the constrained search space, we compute the directional water index along all ten directions (see Fig. 4).

Confirmation Test for Bridges: In order to confirm that a hypothesized bridge segment is actually a bridge, we use the constraints described earlier in this section. To summarize, a bridge is a narrow and linear concrete structure between two disjoint water bodies and is connected to a road. Water surrounds it in all directions, except in the direction of the bridge. Therefore, the water densities in different directions and connectivity between the bridge and road can be used as the basis for confirmation.

We have already computed the signed directional water index along different directions (0–9). We define unsigned directional water densities as the sum of the water index along the positive and negative directions. Thus

$$\begin{aligned} \text{uwi}(0 \leftrightarrow 9) &= \text{dwi}[0] + \text{dwi}[9] \\ \text{uwi}(1 \leftrightarrow 5) &= \text{dwi}[1] + \text{dwi}[5] \\ \text{uwi}(2 \leftrightarrow 6) &= \text{dwi}[2] + \text{dwi}[6] \\ \text{uwi}(3 \leftrightarrow 7) &= \text{dwi}[3] + \text{dwi}[7] \\ \text{uwi}(4 \leftrightarrow 8) &= \text{dwi}[4] + \text{dwi}[8] \end{aligned}$$

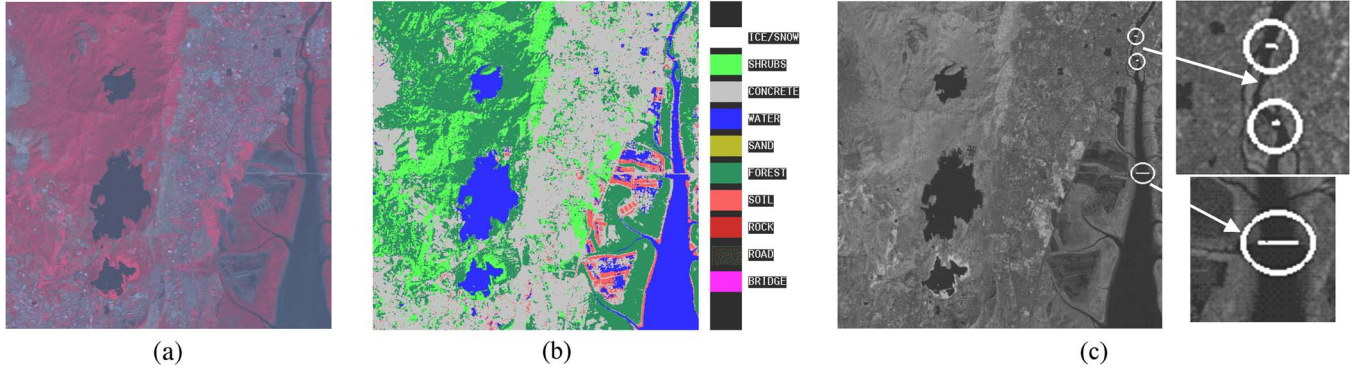


Fig. 4. IRS-1C satellite multispectral image of ground resolution of 23.5×23.5 m. (a) Original image, (b) classified image, and (c) mapped image of the infrared band image and detected bridges.

where uwi is the unsigned directional water index and dwi is the signed directional water index. Using these, we define the total water index (twi), the minimum directional water index (mwi), and the percentage of water index (pwi) along the direction of the bridge as follows:

$$\begin{aligned} twi &= uwi[0 \leftrightarrow 9] + uwi[1 \leftrightarrow 5] + uwi[2 \leftrightarrow 6] \\ &\quad + uwi[3 \leftrightarrow 7] + uwi[4 \leftrightarrow 8] \\ mwi &= \text{Min} \{uwi[0 \leftrightarrow 9], uwi[1 \leftrightarrow 5], uwi[2 \leftrightarrow 6], \\ &\quad uwi[3 \leftrightarrow 7], uwi[4 \leftrightarrow 8]\} \\ pwi &= \frac{mwi}{twi} \times 100. \end{aligned}$$

A hypothesized segment is confirmed as a road segment if the minimum water index occurs only along the bridge direction, and the value of the percentage of water index is less than a certain threshold and it is connected to a road segment.

V. EXPERIMENTAL RESULTS

To test the effectiveness of the approach, we use the multispectral images obtained using the IRS-1C satellite. The IRS-1C satellite comprises the following two parts: a standard multipurpose platform and a sensor system payload. The payload consists of two identical high-resolution visible (HRV) sensor systems, a wide field sensor (WiFS) system, and a package comprising two magnetic tape data records and a telemetry transmitter. The HRV sensors can operate in the following two modes in the visible and reflective infrared portions of the spectrum: a panchromatic mode, corresponding to observation over a broad spectral band (similar to a typical black-and-white photograph), and a multispectral (color) mode, corresponding to observation in four narrower spectral bands. In IRS-1C satellite panchromatic mode, the wavelength of the band is $0.50\text{--}0.75 \mu\text{m}$. The ground spatial resolution is 5.8×5.8 m. In the multispectral mode, the wavelengths of bands 2, 3, 4, and 5 are $0.52\text{--}0.59 \mu\text{m}$ (G), $0.62\text{--}0.68 \mu\text{m}$ (R), $0.77\text{--}0.86 \mu\text{m}$ [near-infrared (NIR)], and $1.55\text{--}1.75 \mu\text{m}$ [shortwave infrared (SWIR)], respectively. The ground spatial resolution is 23.5×23.5 m. Moreover, IRS-1C also has WiFS sensors, which operates in $0.62\text{--}0.68 \mu\text{m}$ (R) and $0.77\text{--}0.86 \mu\text{m}$ [near-infrared (NIR)] with a spatial resolution of 188×188 m.

In this paper, we illustrate our approach with two scenes obtained by using the IRS satellite multispectral imagery. Both the images are taken from the plains area but from different seasons, one from winter and the other from the summer season. Both the scenes are classified into eight different land-cover classes using the multiseed supervised classification technique. The classes obtained are ice/snow, shrubs, concrete, water, sand, forest, soil, and rock for both images, and the average numbers of pixels of each class for both seasons used for training the classifier are 0, 2500, 4323, 4868, 1252, 3215, 768, and 1019, respectively. Note that no ice/snow training samples are available in low-altitude plain areas of India.

Fig. 4(a) shows an original multispectral image (of a plains scene in summer) of size 512×512 generated from green, red, and near-infrared bands of IRS-1C. Statistical parameters for each class are obtained from the training samples by using SPD and ESP algorithms, which are described in Sections III-A and III-B, respectively. Then, the multiseed-based supervised classification, described in Section III-C, is used to obtain a classified image [see Fig. 4(b)]. The circularity parameter θ in the SPD algorithm is set to 3.0. We have checked the classification accuracy from the known number of pixels of each class by multiseed supervised classification technique.

We have tested the road detection algorithm [17] on a number of PAN and LISS images from IRS-1C/1-D and SPOT satellites. For IRS-1C/1-D LISS, there are four spectral bands—three in the visible range and one in the near-infrared range. We consider only the near-infrared band because the roadlike structures are not significant in the first three bands. We have seen from the road output image that the roads are well connected with all three bridges.

The results of applying the bridge detection algorithm are shown in Fig. 4(c). The bridges are shown in white lines. Although the two bridges in the upper right corner of the original image of Fig. 4(a) are not clearly visible, the proposed algorithm is able to detect them. The detailed areas around the bridges are also shown in Fig. 4(c).

Fig. 5(a) shows another original multispectral image (plains scene in winter) of a different part in India. The size of the image is 512×512 . Statistical parameters for each class are obtained from the training samples by using SPD and SPE algorithms. The classification is performed based on these statistical parameters. Fig. 5(b) shows the final output of the classification method. After applying all steps in the bridge detection technique, we get the final bridge output. Fig. 5(c)

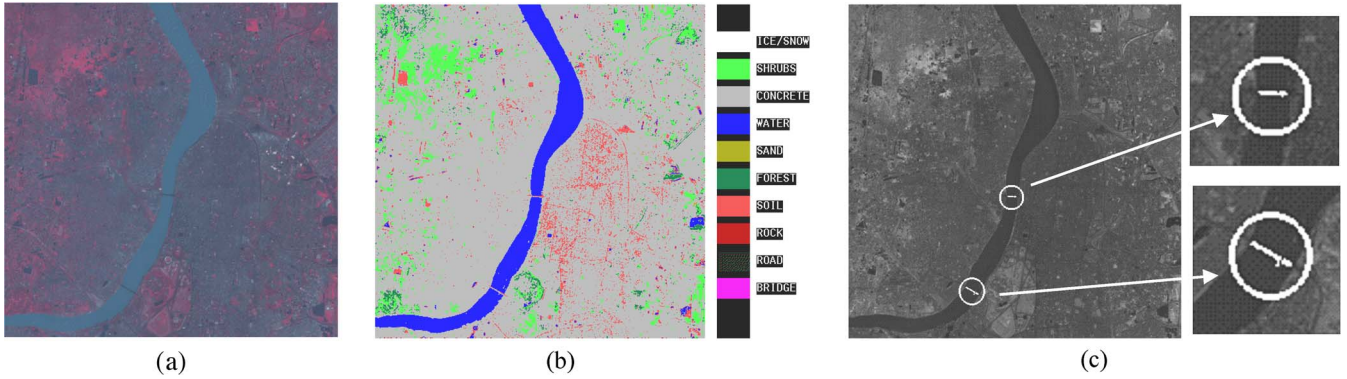


Fig. 5. IRS-1C satellite multispectral image of ground resolution of 23.5×23.5 m. (a) Original image, (b) classified image, and (c) mapped image of the infrared band image and detected bridges.

shows the combined images of the near-infrared band and detected bridges.

We compare the results for both images with a Survey of India (SOI) map and find that the detected bridge locations are correct. SOI publishes the most geologic and geographic maps of India in the form of 392 numbers $1^\circ \times 1^\circ$ quadrangles sheets. An exercise was to register the SOI toposheets and pick selected features for the base map. The scale of the map is 1 : 25 000. Maps are updated every five to ten years and are based on site visits to verify the features including bridges.

The parameters that are used in several modules of bridge detection technique are same for both images. In this experiment of the IRS-1C LISS image for river extraction, the threshold values that we take for area, elongatedness, and perimeter are 400, 20, and 300, respectively. We use 5×5 ($d = 5$) windows for the extraction of possible candidate bridge pixels. We have chosen the value of parameters $\alpha = 0.5$, $\delta = 0.3$, and $d_{ms} = 5.0$ for the merger–separator operation technique.

We have applied the method proposed by Mandal *et al.* [15] and have summarized the results of this method and the method proposed in this paper for detection of bridges in both images. For the scene shown in Fig. 4, Mandal’s approach detected one of the three bridges and gave one false alarm. In Fig. 5, Mandal’s approach recognized both of the bridges correctly. In contrast, our approach extracted all the bridges in both scenes correctly.

VI. SUMMARY AND FUTURE WORK

We have presented an approach for detecting bridges from multispectral satellite images. A bridge is characterized by a narrow concrete structure between two disjoint water bodies and is connected to a road. There are many civilian, commercial, and military applications for this paper. For example, this technique can also be used after disasters to see if a bridge still exists.

The following are the three main modules in our bridge detection approach: 1) pixel classification; 2) road extraction; and 3) bridge detection. It is very difficult to collect the training samples of different materials from low-resolution multispectral images like IRS-1C/1-D satellite images of ground resolution of 23.5×23.5 m. Hence, we group several classes of material (buildings, roads, railroads, and metallic structures) into a single class called concrete. A majority-must-be-granted-logic-based multiseed supervised technique is

incorporated to classify the image pixels into eight land-cover types. The classified image is converted into a trilevel image, i.e., comprising water, concrete, and background. Because bridges over water are important targets in many applications and the focus of this paper is to extract such bridges, water information is one of the most important features in our approach. Separation of river and nonriver parts from the water class information of the classification image is done by using a recursive scanning technique that exploits some geometric constraints such as area, perimeter, and elongatedness. A low-level operator is used within a $d \times d$ window for extraction of the possible bridge pixels. This operation is based on the following observations: 1) the bridge on the water body is a straight line, and 2) the bridges are typically no more than a few meters wide. For grouping the possible bridge pixels, a grouping operator based on connectivity property within a 3×3 window is used at the edges of different regions of the possible bridge pixel image.

The grouping operation may not be able to find the correct group due to a noisy image or misclassification. Some of the groups may lie in the same straight line with a minimum distance that is greater than one but less than a specific threshold. These two straight lines will not form a single group due to the grouping operation, but they should be considered for merging. This situation can be overcome by increasing the window size of the grouping operation. If we increase the size of the window, then the groups that lie in the same straight line with a minimum distance that is greater than one will merge. At the same time, the groups that are parallel with a minimum distance that is greater than one will also merge. Hence, a merger–separator operation is introduced to overcome this situation. The merger–separator operation is based on geometric constraints and an MST. The centers of each group are computed, and the MST of these centers is drawn. The minimum distance D between the two groups that are connected by the MST is computed. The angle difference ϕ between the best-fitted lines of the MST connected groups is determined. If $\phi > \alpha$ (α is a predefined threshold) then they are different groups. If not, then the angle θ_{\min} can be determined as the minimum value of the pair of difference values of centers fitted line of the two groups and individual fitted lines of the two groups, which are connected by the MST. If θ_{\min} lies between a specific range then they are collinear; otherwise they are parallel. The two groups can be merged on the basis of the values of D and ϕ , and the condition of parallelism. We

apply decision-making logic regarding the water index in ten directions and connectivity with roads to derive the confirmed bridge. An efficient algorithm for road detection is used in our proposed method of bridge detection. To test the efficiency of the algorithm, several real-life data are considered.

The approach described here is tuned to the IRS-1C/1-D satellite imagery. The size of an object depends on the spatial resolution of the satellite sensor. The parameters will need to be adjusted for a different type of sensor, e.g., IKONOS that has a higher spatial resolution and different multispectral bands. IKONOS has a blue band instead of a SWIR band, and the spatial resolution of the bands is approximately six times higher than the IRS-1C image. Consequently, bridges that were 1–3 pixels wide in the IRS-1C image can reach a width of about 20 pixels in an IKONOS image. Thus, we have to collect different training samples for each class, and by SPD and SPE algorithms, we can calculate the statistical parameters for each class. In the road extraction algorithm, we have to change the mask size for low-level operations to enhance the linelike structures. Some soft threshold values in the segmentation process of road detection also have to be tuned. The rest of the road detection algorithm will remain unchanged. In the bridge detection algorithm, we have to change the window size for the extraction of possible bridge pixels. We have to search in some additional directions for finding the directional water index. The segmented region, which is possibly a bridge, will be thicker in the higher resolution image. Thus, a thinning algorithm may have to be introduced in the merger–separator algorithm. The rest of bridge detection techniques will remain unchanged.

This paper can be extended in many different directions. As described before, it can be tuned for higher resolution satellite imagery. In addition, it will be interesting to detect bridges in panchromatic data. In this case, we have to develop a new supervised classification technique for classifying the panchromatic data.

REFERENCES

- [1] D. Chaudhuri and B. B. Chaudhuri, "A multiseed non-hierarchical clustering technique for data analysis," *Int. J. Syst. Sci.*, vol. 26, no. 2, pp. 375–385, Feb. 1995.
- [2] D. Chaudhuri and B. B. Chaudhuri, "A novel multiseed non-hierarchical data clustering technique," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 27, no. 5, pp. 871–876, Sep. 1997.
- [3] D. Chaudhuri, B. B. Chaudhuri, and C. A. Murthy, "A data driven procedure for density estimation with some application," *Pattern Recognit.*, vol. 29, no. 10, pp. 1719–1736, Oct. 1996.
- [4] D. Chaudhuri, A. Mishra, V. Gohri, and J. K. Sharma, "Automated detection of tracks in mountainous terrain using IRS-1C/1D images," in *Proc. ICVGIP*, 2002, pp. 394–399.
- [5] I. Couloigner and T. Ranchin, "Mapping of urban areas: A multi-resolution modeling approach for semi-automatic extraction of streets," *Photogramm. Eng. Remote Sens.*, vol. 66, no. 7, pp. 867–874, Jul. 2000.
- [6] M. F. A. Fortier, D. Ziou, C. Armenakis, and S. Wang, *Survey of Work on Road Extraction in Aerial and Satellite Images*, 1999. [Online]. Available: <http://www-l2ti.univ-paris13.fr/~auclair/Publi/Auclair-TR247.pdf>
- [7] M. F. A. Fortier, D. Ziou, C. Armenakis, and S. Wang, "Automated correction and updating of road databases from high-resolution imagery," *Can. J. Remote Sens.*, vol. 27, no. 1, pp. 76–89, Feb. 2001.
- [8] A. Gruen and H. Li, "Semi-automatic linear feature extraction by dynamic programming and LSB-snakes," *Photogramm. Eng. Remote Sens.*, vol. 63, no. 8, pp. 985–995, 1997.
- [9] S. Houzelle and G. Giraudon, "Data fusion using SPOT and SAR images for bridge and urban area extraction," in *Proc. IGARSS*, 1991, vol. 3, pp. 1455–1458.
- [10] X. Hu, "Automatic extraction of linear objects and houses from aerial and remote sensing imagery," Ph.D. dissertation, Wuhan Univ., Wuhan, China, 2001. (in Chinese).
- [11] X. Hu and C. V. Tao, "Automatic extraction of main-road from high resolution satellite imagery," in *Proc. IAPRS*, Xi'an China: Commission II, Aug. 20–23, 2002, vol. XXXIV, pp. 203–208. Part 2.
- [12] D. Klang, "Automatic detection of changes in road databases using satellite imagery," in *Proc. IAPRS*, Stuttgart, Germany, 1998, vol. 32, pp. 293–298. Part 4.
- [13] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner, "Automatic extraction of roads from aerial images based on scale space and snakes," *Mach. Vis. Appl.*, vol. 12, no. 1, pp. 23–31, Jul. 2000.
- [14] N. Lomenie, J. Barbeau, and R. Trias-Sanz, "Integrating textural and geometric information for an automatic bridge detection system," in *Proc. IGARSS*, Toulouse, France, Jul. 2003, pp. 3952–3954.
- [15] D. P. Mandal, C. A. Murthy, and S. K. Pal, "Analysis of IRS imagery for detecting man-made objects with a multi-valued recognition system," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 26, no. 2, pp. 241–247, Mar. 1996.
- [16] V. Mehre and D. Chaudhuri, "Multi-spectral classification technique based on data driven procedure and MDLBC classifier," in *Proc. Int. Conf. CODIS*, 2004, vol. IS-34, pp. 554–557.
- [17] A. Mukherjee, S. K. Parui, D. Chaudhuri, B. B. Chaudhuri, and R. Krishnan, "An efficient algorithm for detection of road-like structures in satellite images," in *Proc. 13th Int. Conf. Pattern Recog.*, 1996, vol. III, pp. 875–879.
- [18] D. Roberts, M. Gardner, C. Funk, and V. Noonha, "Road extraction using spectral mixture and Q-tree filter techniques," *NCRST Technical Reports: 3rd Quarter Progress*, Feb. 2001.
- [19] T. P. Ryan, *Modern Regression Methods*, ser. Wiley Series in Probability and Statistics. New York: Wiley-Interscience, 1996.
- [20] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision*, 2nd ed. New York: PWS Publishing, 1998.
- [21] R. Trias-Sanz and N. Lomenie, "Automatic bridge detection in high-resolution satellite images," in *Proc. 3rd ICVS*, J. L. Crowley et al., Eds., Graz, Austria, Apr. 2003, vol. 2626, pp. 172–181.
- [22] R. Trias-Sanz, N. Lomenie, and J. Barbeau, "Using textural and geometric information for an automatic bridge detection system," in *Proc. ACIVS*, Brussels, Belgium, Aug. 31–Sep. 3, 2004, pp. 325–332.
- [23] J. Wang and Q. Zhang, "Applicability of a gradient profile algorithm for road network extraction sensor, resolution and background considerations," *Can. J. Remote Sens.*, vol. 26, no. 5, pp. 428–439, 2000.
- [24] Q. Zhang and I. Couloigner, "A wavelet approach to road extraction from high spatial resolution remotely sensed imagery," *Geomatica*, vol. 58, no. 1, pp. 33–39, Jan. 2004.



D. Chaudhuri received the B.Sc. degree (with honors) in mathematics from Visva-Bharati University, Santiniketan, India, in 1985, the M.Sc. degree in applied mathematics from Jadavpur University, Calcutta, India, in 1987, and the Ph.D. degree in image processing and pattern recognition from the Indian Statistical Institute, Calcutta, in 1994.

He is currently a Scientist with the Defence Electronics Application Laboratory, Dehradun, India. He has published over 25 papers in international journals and conferences. He is an Associate Editor of the

International Journal of Computational Vision and Robotics. His research interests include image processing, pattern recognition, computer vision, remote sensing, and target detection from satellite imagery.

Ashok Samal (S'85–M'88) received the B.Tech. degree in computer science from the Indian Institute of Technology—Kanpur, Kanpur, and the Ph.D. degree in computer science from the University of Utah, Salt Lake City.

He is currently an Associate Professor with the Department of Computer Science and Engineering, University of Nebraska—Lincoln, Lincoln. He has published over 70 papers in international conferences and journals. He is currently an Associate Editor for *Pattern Recognition* and *Pattern Analysis and Applications*. His research interests include biometrics, document image understanding, geospatial data mining, and computer science education.