Use of Salient Features for the Design of a Multistage Framework to Extract Roads From High-Resolution Multispectral Satellite Images

Sukhendu Das, T. T. Mirnalinee, and Koshy Varghese

***Abstract*—The process of road extraction from high-resolution satellite images is complex, and most researchers have shown results on a few selected set of images. Based on the satellite data acquisition sensor and geolocation of the region, the type of processing varies and users tune several heuristic parameters to achieve a reasonable degree of accuracy. We exploit two salient features of roads, namely, distinct spectral contrast and locally linear trajectory, to design a multistage framework to extract roads from high-resolution multispectral satellite images. We trained four Probabilistic Support Vector Machines separately using four different categories of training samples extracted from urban/suburban areas. Dominant Singular Measure is used to detect locally linear edge segments as potential trajectories for roads. This complimentary information is integrated using an optimization framework to obtain potential targets for roads. This provides decent results in situations only when the roads have few obstacles (trees, large vehicles, and tall buildings). Linking of disjoint segments uses the local gradient functions at the adjacent pair of road endings. Region part segmentation uses curvature information to remove stray nonroad structures. Medial-Axis-Transform-based hypothesis verification eliminates connected nonroad structures to improve the accuracy in road detection. Results are evaluated with a large set of multispectral remotely sensed images and are compared against a few state-of-the-art methods to validate the superior performance of our proposed method.**

***Index Terms*—Feature extraction, image analysis, image classification, image processing, image region analysis, image segmentation, neural network application, pattern classification, pattern recognition, remote sensing.**

I. Introduction

AUTOMATED road network extraction from remotely sensed imagery is a challenging and important research topic. Roads are the backbone and essential modes of transportation, providing many different supports for human civilization. Cartographic object extraction from digital imagery is a fundamental operation for geographic information system (GIS) update. Based on the sensor used for satellite data acquisition and geolocation (urban areas only) of the area, the type of processing varies and users have to often tune several heuristic parameters of their algorithms based on the requirements to suit the data. Most of the algorithms suggested in literature for road extraction consist of one or more of the following operations: image segmentation (often using a classifier), detection of a pair of parallel linear segments (using edge detector, Hough transform), snakes (active contours), morphological operations for cleaning and bridging discontinuities, matching of road templates, etc. However, the complete automation of the road extraction processes is still a hard problem which has not been solved with a reasonable degree of success (accuracy) over a large set of images of different categories of urban and suburban areas.

Due to large variations in the appearance of roads, as observed in high-resolution multispectral (three-band red–green–blue (RGB) data) satellite images, it is difficult to consider and model all the variants in the different types of phenomena (spectral reflectance, shape, contrast, shadow, and occlusion) as observed in roads and incorporate them into a single processing module. The design of such a robust powerful module to solve this problem with some degree of satisfaction and provide results with any degree or acceptability will be a complicated task. We also doubt if this would be feasible at all, given the current technology of pattern classification (mathematical model, classifier, and expert system) and image analysis algorithms.

The road network, as observed in a satellite image, is a layout (due to wide variability in visual characteristics and features) that is complex to model mathematically. Even a powerful classifier will not be able to extract accurately such an object exhibiting wide variations in features (with overlapping classwise distributions) such as spectral response, shape, contrast, occlusion, and shadow. Also, roads cannot be defined as an object exhibiting a unique shape and/or having a uniform contrast with respect to the background. Hence, existing target recognition and foreground segmentation algorithms will not be able to capture any meaningful layout of roads.

From an extensive literature survey, we found that most researchers have attempted to solve this problem using a semisupervised or semiautomatic system. Semiautomatic systems require manual identification of seed points for road-tracking operators to delineate (and detect) the roads. On the other hand, automatic systems generally do not require any human interaction at the testing phase when an input image is processed to

Manuscript received March 19, 2010; revised October 22, 2010 and February 25, 2011; accepted March 3, 2011. Date of publication May 11, 2011; date of current version September 28, 2011.

The authors are with the Indian Institute of Technology Madras, Chennai 600036, India (e-mail: sdas@iitm.ac.in; mirna@cse.iitm.ac.in; koshy@iitm.ac.in).

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.2011.2136381

detect a road network. Of course, significant amounts of training and empirical evaluation of the optimal set of parameter values are required for these automatic systems to be successful. This process of training is done offline, which makes the system powerful enough to operate independently with no interaction from the user, during the testing (or operational) stage. We have proposed a framework for such an automatic system for road detection, with minimal (almost none) interaction from the user at the testing phase. Hence, we kept the discussion focused on automatic methods. Most of the published works on automatic methods have exhibited results on a very limited set (below ten) of images within an urban area, using one to two modules for processing. We were almost certain that a multistage, well-formulated, and planned strategy can only conquer this problem, with applicability over a wide variety of road networks across many places over the globe. This drove us to formulate and design a multistage framework to solve this problem with some degree of satisfactory performance and make it usable for the GIS and remote sensing community. To accomplish this task, we relied principally on two salient features of roads, as observed by a human being, to be first extracted from a satellite image.

We have exploited two salient (as observed by a human being) features of roads: 1) spectral contrast with respect to the surroundings and 2) locally linear trajectory. The main focus of our road extraction method is to extract the salient features first. Saliency is defined as the part of a scene projecting out of the surroundings due to discriminability and the power of seeking attention. The salient features help to identify the road boundaries enclosed by homogeneous regions on either side. Based on local linearity and spectral changes as salient features, roads are detected in our proposed approach. A dominant orientation descriptor and spectral discriminating distribution give the salient features in the image. Integration of these pieces of information using an optimization framework for integrating complimentary features provides a better result in situations only when the road has very few obstacles (trees, large vehicles, over bridge, nearby tall buildings, etc.). Misclassification of small roadlike regions is also one of the major difficulties in this automatic process. To improve the results in such situations which provide discontinuities in road detection (or a road may be merged with an adjacent region with similar spectral features), we designed a set of suitably selected postprocessing stages to filter the undesired effects and thus improve the accuracy in road detection. The overall framework thus consists of two stages. Stage I deals with the detection of candidate regions for roads, using salient features from satellite imagery. Stage II deals with a set of suitable postprocessing modules to enhance the road extraction system, mostly eliminating false alarms.

The rest of this paper is organized as follows: Section II deals with the review of existing road extraction techniques. Section III deals with the research issues, design strategies, and brief description of overall proposed methodology. A description of Stage I in the proposed framework is presented in Section IV. Section V deals with the process of road network formation. We present experimental results in Section VI and conclude this paper in Section VII.

II. Review of Previous Work

In the last years, many approaches have been developed to deal with the detection of roads from satellite images [1]–[15]. Various road detection techniques include knowledge-based methods [16], mathematical morphology [2], [12], [17], snakes [18]–[22], classification [23]–[26], differential geometry [27], region competition [28], active testing [29], perceptual grouping [30], and dynamic programming [31]. Mena [32] and Fortier *et al.* [33] provide extensive surveys of the literature on road extraction technique. We briefly highlight the key aspects of the various categories of work in the following.

Knowledge-Based Methods: Trinder and Wang [16] proposed a road model which includes geometric and radiometric properties of roads. The hypotheses for the roads are generated using these rules, and a top-down process is applied to verify the road hypotheses. A road tracker, based on angular texture signature, has been proposed by Shen *et al.* [34] based on the knowledge of the roads on high-resolution imagery. The current limitations are that the algorithm may not work on a road cast by much shadow and occlusion in complex scenes. It can only track long ribbon roads on grayscale imagery, and it needs more computing time. Results are shown using a single high-resolution satellite image sample. In [8], a road tracker based on shape classification of the road footprints effectively detects road intersections. It extracts almost all inscribed lines of road networks and forms a road tree. A toe-finding algorithm finds the dominant directions of the road footprint and makes it possible to initialize and track a road segment automatically. However, the multidirectional road tracker suffers from overextraction. It is necessary to remove portions in the road tree that do not appear to be roads. An original road tree pruning approach has been presented, which is based on a Bayesian decision model defined on the road tree, and takes into account the basic property of road networks: The width of a road is constant or piecewise constant. Results were shown for six images, where completeness of 84%–94% and correctness of 82%–92% were reported.

Mathematical Morphology: Zhang *et al.* [35] proposed an approach for detecting road networks from high-resolution images using a combination of mathematical morphological operations. In the preprocessing stage, the image is first segmented to separate the road network regions from their surroundings, followed by a morphological trivial opening operation. A criterion T is defined using the major axis of the minimal ellipse which encloses an object. Only the connected components that satisfy the criterion T are retained after morphological reconstruction. This process preserves the elongated road areas and filters almost all the houses and small clusters of noise. The result is further refined by filling holes, removing small paths and recovering shadowed areas. In high-resolution simulation images and aerial photos, the approximate road centerline network is finally extracted. However, road gaps may still exist, if road surfaces are completely broken and there is no other information supporting the linkage. A binary and grayscale mathematical morphology with a line segment match method is proposed in [12]. Mean-shift clustering approach is used within the hue–saturation–intensity space. Conditional morphological

techniques are used to significantly improve the segmentation results in [9].

Snakes: A model and a strategy based on the multiscale detection of roads in combination with geometry-constrained edge extraction using snakes are presented in [21]. Results are shown for three sample images, and this approach is mostly intended for rural areas. A family of cooperating snakes [36], which are able to split, merge, and disappear as necessary, is used to extract disconnected road networks and enclosed regions. The energy functional comprises of gradient vector flow computed as a diffusion of the gradient vectors of a gray-level or binary edge map derived from the image.

Classification: Most of the existing road extraction methods for multispectral imagery rely on an automated and reliable classification of road surfaces [24], [37]. Unfortunately, the classification accuracy of roads is far from satisfactory whether a supervised classification method or an unsupervised method is used. The main difficulty lies in the high misclassification between roads and other spectrally similar objects, such as parking lots, buildings, crop fields, etc. Song and Civco [24] used two shape measures, namely, smoothness and compactness, to further reduce the misclassification between roads and other spectrally similar objects in a support vector machine (SVM) classifier. In [23], Nager *et al.* proposed SVM-based road extraction using the intensity, edge length, edge gradient, and width between a pair of edges as features. They have reported the correctness measure of their algorithm to a maximum of 35%, with a completeness measure of 75%–90%.

In [38], Mantero *et al.* describes a classification strategy that allows the identification of samples drawn from unknown classes through the application of a suitable Bayesian decision rule [39]. This approach is based on SVMs for the estimation of probability density functions, which uses a recursive procedure to generate prior probability estimates for known and unknown classes. SVMs were also exploited by Yager *et al.* [23] as a classifier for road extraction, which involves two stages of processing. Here, SVM is trained using edge-based features such as edge length, gradient, and intensity within the edge pair. In level 1, SVM is used to classify edges as road edges or nonroad edges. Edges classified as road edges are given as input to the SVM in level 2 where opposite edges are paired as road segments. However, they have reported a very low correctness measure. A new method [40] is presented for the extraction of buildings from light detection and ranging digital elevation models on the basis of segmentation principles. The accuracy of supervised classification largely depends on the quality of the training data. The locations and sample size of training data are difficult to be optimized depending on image data types and classifiers to be used.

A Self-Organized Road Map was developed by Doucette *et al.* [37] for extracting road networks from classified imagery. Huang *et al.* [14] proposed a road detection system based on multiscale structural features and SVMs. A new unsupervised aerial image segmentation algorithm is presented in [41]. It is built on the basis of the modified two-phase Mumford–Shah model [42] with the combined feature constraints. Results are shown for three images of high resolution.

Bogess [43] proposed an approach to identify roads in LANDSAT images using artificial neural networks (ANNs). A feedforward neural network was trained with the spectral information. The obtained results contain a lot of false alarms and disconnected road segments. Tu-Ko [44] proposed an approach in which a feedforward neural network was trained with the spectral information and edges. The results contain many nonroad edge segments classified as edge segments.

Region Competition: Amo *et al.* [28] proposed a combined approach consisting of region growing and region competition to extract road centerlines and sides. The initial seeds are given manually, and hence, this is a semiautomatic process. Then, a first simple region-growing-based model is applied to obtain a rough road approximation. This model is refined by the region competition algorithm. Hu *et al.* [45] presented a two-step approach, i.e., detecting and pruning, for the automatic extraction of road networks from aerial images. Results are shown for four images of high resolution.

Active Testing: This approach was proposed by Geman and Jedynak [29] originally for road tracking from SPOT satellite images with 10-m resolution. Given a start point and an orientation, the algorithm searches the road by constructing a decision tree. Candidate road pixels are selected in a limited number of directions using hypothesis tests. The road centerline is obtained from the extracted road segments. However, if the road borders are incomplete or the road width changes, the centerline position might not be accurate. Given a starting point and starting direction, road networks from medium-resolution satellite imagery were extracted and results are shown for only two images.

Perceptual Grouping: Gamba *et al.* [30] detected urban road networks from high-resolution optical and SAR images. An adaptive filtering procedure captures the predominant directions of roads and enhances the extraction results. Then, to both discard redundant segments and avoid gaps, a perceptual grouping algorithm is devised, exploiting collinearity and proximity concepts. Finally, the road network topology is considered, checking for road intersections and regularizing the overall pattern. Results are shown for six images of high resolution. Sahar and Alireza [46] proposed a road tracing approach using extended Kalman filter and a special particle filter module. These modules, making use of clustering algorithms, are able to pass severe obstacles on the road and trace all road branches at a junction. All road segments in a connected road network are traced using a single seed point. However, the performance of the algorithm depends on the parameters of the module. Results were shown for two images, with average completeness of 88% and correctness of 98%.

Dynamic Programming: Barzohar and Cooper [31] proposed a geometric–stochastic model of roads for finding main road networks, but this model is not optimal. First, a method was developed to model roads in low-resolution imagery. Hence, it focused on modeling road edges using intensity variations. Second, the model was developed for road finding using dynamic programming. In order to apply dynamic programming in road finding, some unpractical assumptions have to be imposed into the model. Results are shown for four images of high resolution.

TABLE I
TABULARIZED SUMMARY OF A SURVEY ON ROAD EXTRACTION TECHNIQUES

Method (Algorithm)	Feature	Number of Samples used as test cases	Performance (%)
Road tracker, angular texture signature [34][25][8]	Intensity and edge direction	2 - 4	Accuracy: 70 - 79
Mean Shift clustering [9][15]	Histogram of the HSI image	1 - 8	Accuracy: 86 - 90
Snakes [21][20]	Intensity Gradient	3	Completeness: 72 - 84 Correctness: 95 - 99
Adaboost [49]	Intensity, LBP	3	Completeness: 64 - 91 Correctness: 66 - 90
SVM [24][23]	Intensity, edge gradient edge length, width	3-6	Completeness: 75 - 90.7 Correctness: 13 - 35
Region competition [28]	Intensity	4	Accuracy: 80
Binarization, optimization using Markov random field [7][5]	Mean Radiance value	5	Completeness: 76.9 - 90.9 Correctness: 63.5 - 93.4
Fuzzy inference algorithm [26][50]	Wavelet based filterbanks	1 - 2	Accuracy : 82 - 93
ANN [51][45][46]	Intensity, edge & normalised distance	1 - 2	Accuracy : 95
Cooperating snakes [37]	Gradient Vector flow	4	Precision: 51.88 - 85.8 Recall: 79.87 - 98.3
Morphology [36][35][15]	Length & direction of line	3 - 4	Accuracy 91.76
Filtering and grouping [30][48]	Line direction, intensity	2 - 6	Completeness: 66-95 Correctness: 80-98

Table I gives a summary of the methods and features used for automatic road detection from high-resolution satellite images. Among them, snakes and ANN (classifier)-based methods have provided the best performance. Snakes need an initialization, making it a semiautomatic method, which relies on gradient and spectral contrast as a part of the energy functional. Most methods have only used two to eight different image samples to observe the performance, which is low for any conclusive study of performance analysis in pattern recognition tasks. A road network is too complex to be completely modeled using any of the existing processing techniques, and hence, a framework that integrates suitable processing modules has been formulated in this research work. In the design of our framework for road detection, the following processes are novel (never used earlier for this purpose): optimization using constraint satisfaction neural network (CSNN), extraction of linear features using Dominant Singular Measure (DSM), and the use of region part segmentation (RPS). We used 100 test samples, which is significantly more than (typically) one to ten samples used in all the previous efforts discussed by us. Grouping and linking segments using neighborhood/connectivity and morphological operations (dilation, erosion, thinning, etc.) have been used in earlier work of road detection. Most of the features are based on intensity and histogram, which have also been used earlier.

Most of the previously developed methods successfully recognize roads which exhibit a homogeneous surface and have a strong contrast with the adjacent surface. However, in cases where surrounding objects like trees, buildings, and cars occlude the road or cast shadows on it, road extraction often fails, resulting in gaps and discontinuities in the detected roads. The most popular and successful methods rely on one or two methods of processing: 1) classification and 2) geometrical property. We also use both but integrate these as a pair of salient properties using a multistage framework. The focus of this paper is on the design and development of a technique, which enables us to extract road segments from an input image without much of user interaction. The motivation of our work comes from the fact that the complimentary information of regions (road pixels in our case) and edges (road boundaries) have not been exploited together to obtain a better road map from satellite images. Performance analysis using various categories of images have not been provided for a rigorous study of the applicability of the methods published in literature. In the next section, we describe our proposed multistage method based on the issues discussed in this section, followed by design details of the road extraction modules listed in Table II.

III. DESIGN STRATEGY AND PROPOSED METHODOLOGY FOR ROAD DETECTION

A. Design Strategy

The difficulties in the design of an automated road network extraction system using remotely sensed imagery lie in the fact that the image characteristics of road feature vary according to sensor type, spectral and spatial resolution, ground characteristics, etc. Even for an image taken over a particular urban area, different parts of the road network reveal different characteristics. In the real world, a road network is too complex to be modeled using a mathematical formulation or an abstract structural model. The existence of other objects (e.g., buildings and trees) casts shadows to occlude road features, thus complicating the extraction process.

Human perceptual ways of recognizing a road involves [50] extracting geometric, radiometric, and topological characteristics of an image. Humans usually recognize a road using, first, its geometric characteristics, considering a road to be a long and elongated feature with uniform width and similar radiometric variance along its path. Even though the spectral characteristics of a road vary within an image, its physical appearance tends to exist as long continuous features. Humans fuse these vital clues to identify a foreground road object from the background layer. This motivated us to develop a generic framework that integrates suitable processing modules necessary for extracting the different types of features present in road objects from satellite scenes. We next present the characteristics of roads, followed by suitable modules designed specifically to address these issues. We also validate the efficiency of the extraction system using experimental results.

TABLE II
ROAD CHARACTERISTICS AND CORRESPONDING PROCESSING MODULES

Sl No	Characteristics	Strategy/Module	Remarks
1.	Elongated linear Structures	DSM on edge map	Discontinuity due to occlusion
2.	Mostly homogenous; Contrast w.r.t. background	SVM classier using mean and variance of spectral response	Misclassication of non-road objects with identical spectral response
3.	Discontinuities and distortions in linear paths	CSNN-CII and Segment Linking	Chance of linking roads with other structures
4.	Rarely terminate, Not appearing in isolation	Region Part Segmentation	Removal of small road fragments
5.	Roads have near-parallel boundaries with constancy in width	MAT based hypothesis verication	Removal of parking lots

A set of robust, relevant, and sufficiently complete semantic descriptions of the object of interest in the entire image is termed as salient features. Most significant characteristics of roads, which appear in high-resolution satellite imagery, are considered to be salient in our work. Salient features used for the extraction of roads from high-resolution satellite imagery are as follows (***I: Salient features***).

1) Roads have a distinctively contrasting spectral signature (both locally and globally) with respect to the background layer (e.g., vegetation, soil, waterways, man-made structures, etc.).
2) Roads are mostly elongated structures, with locally linear properties.
3) The road surface is usually homogeneous, with occasional variations.

Among the different properties, the salient features of roads are their geometrical shape and spectral contrast [as previously stated in 1) and 2)]. Roads in high-spatial-resolution images of urban areas appear as piecewise linear segments with spectrally homogeneous characteristics. These are vital clues, which form the basis of the design of our framework for automatically detecting roads in satellite imagery. Our proposed strategy attempts to extract (perceive) the salient features from images globally and quickly. In the design of a framework for road detection, we first need to exploit these two vital characteristics of roads. In such a case, one may be tempted to use a foreground extraction algorithm trained with spectral patterns for roads and then use linear features on top of it. However, a classifier trained with only spectral features will produce false alarms (identify nonroad objects as roads and filter parts of roads as background). On the other hand, a pattern classifier (for roads) trained with geometrical features cannot be used unless the target (itself) is available for feature extraction. It is difficult to simultaneously extract and fuse this pair of distinct (unrelated) features together. It is more complex to design an operator (or mask) for this purpose, as that would need to simultaneously extract and use spectral and RTS (rotation-translation-scale)-invariant shape (geometrical) features of various target templates from the image data. It is also a nontrivial task to formulate a mathematical (parametric) model for a road network, which will work for all complex variations in the geometric design patterns (linear and curvilinear) formed by roads in an urban scenario. The method of extracting the salient features separately and then fusing them in Stage I of our proposed framework is described in Section IV.

The second stage of processing deals with certain characteristic (properties) of roads in satellite imagery. These characteristics are used to depict the missing road parts occurring due to occlusion, shadow, etc. This enables a more efficient use of the knowledge about roads, in the automatic road extraction process, to improve the accuracy in extraction. Characteristics used for the extraction of roads from high-resolution satellite imagery are as follows (***II: Road Characteristics***).

1) Discontinuities appear in a road structure mainly due to occluding objects, such as trees, buildings, large vehicles, etc., or even shadows.
2) Near-parallel boundaries with constancy in width.
3) Road structures are rarely nonsmooth and occur generally without much of sharp bends.
4) Roads do not appear as a small segment or patch, either in isolation or attached to a large linear segment.
5) Roads rarely terminate (no abrupt ending) within short distances. In fact, they intersect, occlude one another (bridges and highways), and bifurcate to build a network (global appearance).

Due to the existence of these complex phenomena for roads, it is almost impossible to consider and model all these situations and incorporate them in a single module or processing stage for road network extraction. This drove us to formulate and design a hierarchical (pipelined) framework, consisting of the classification (supervised), information integration, filtering, and local neighborhood analysis to obtain better results with acceptable quality.

Hence, in most cases with multispectral data sets, the spectral information alone is not sufficient to define roads. We need an integrated multistage framework to achieve our goal. The characteristics of roads aforementioned are exploited by a multistage framework. Each stage of the framework deals with a particular characteristic of roads and are given in the left column of Table II. The center column gives the corresponding strategy (processing module) used by us to solve the problem, while the right-hand side column specifies the difficulties/drawbacks that one may face with that stage. In the next section, we provide the justification of different processing stages of our proposed multistage framework, based on the issues discussed in this section, followed by the design section on details of the road extraction modules listed in Table II. Results will be compared with two state-of-the-art methods [26], [49] published in literature and one GIS-based software [53] used for raster image analysis.

B. Justification of the Different Stages of Processing

A multistage framework for road extraction has been proposed in this paper. Fig. 1 shows the framework of our proposed

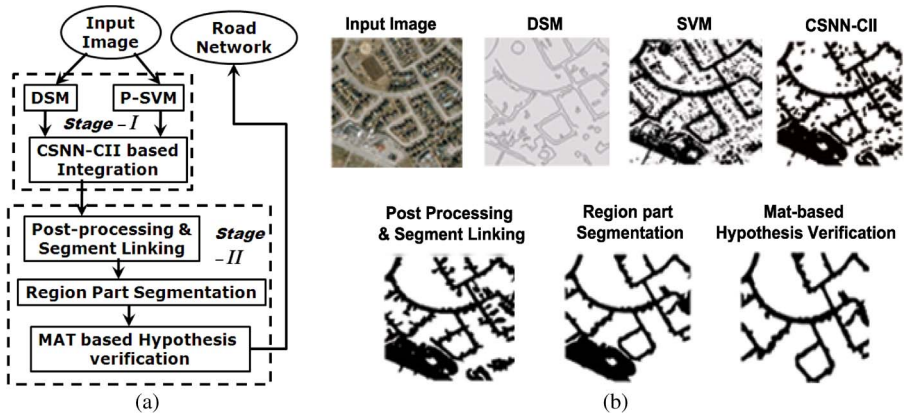


Fig. 1. (a) Framework of the proposed method for road detection. Stage I detects the salient features. Stage II improves the road layer formation using a set of processing modules. (b) Illustration of the results for the intermediate stages of processing for the "input image" shown at the top.

road extraction system, which is a pipelined multistage framework based on the details specified in Table II and discussed in the previous section. At the first stage, we require a process to merge information of two salient features. This is accomplished by an iterative merging process of region- and edge-based information using a set of constraints. Road edges (boundaries) are extracted from edge features using DSM. We assume roads appearing in satellite images to be locally linear. DSM (a novel method) was found to be better than parametric methods such as Hough transform [26]. Soft class labels (probabilities) for each pixel belonging to either road or nonroad regions are produced by the Probabilistic SVM (P-SVM). Road regions are separated from nonroad regions in our proposed framework using a P-SVM classifier.

Either of these techniques (DSM or P-SVM), when solely applied, produce errors which do not occur together (simultaneously) in general. This is due to the fact that the criteria for classification of pixels as road regions (using P-SVM) look for continuity and local smoothness, whereas methods to detect road boundaries (using DSM) look for discontinuities in raster images. We use a modified CSNN, termed CSNN Complementary Information Integration (CSNN-CII) [52], for integrating the complimentary information from the outputs of edge- and region-based processing. A fruitful cooperation could be established between region- and edge-based methods to accurately extract elongated thick objects like roads in high-resolution satellite imagery. A modified CSNN has been used for this task, which uses a modified dynamic window to merge the complimentary information of edges and regions. This completes the requirement for the set of processes in Stage I.

Modules in Stage II have been designed for the purpose of improving the accuracy of road network detection by removing false alarms as well as recovering a few short (road) segments neglected due to false rejection. It works based on some natural and artificial characteristics exhibited by roads due to external influence around its neighborhood. The output of CSNN-CII (at Stage I) needs to be processed further to remove some undesired artifacts and errors. This is performed with the help of three tasks/substages in Stage II [see Fig. 1(a)]. The steps of the algorithm, depicting the process shown in Fig. 1(a), are given in Algorithm 1. Fig. 1(b) shows the results of intermediate stages of processing within our proposed framework. The output of Stage I, labeled as CSNN-CII in Fig. 1(b), has undesired isolated patches, few discontinuities (will be clear with another example, discussed later on), and large nonroad structures that must be eliminated. Postprocessing and segment linking steps help to eliminate the shadows and occlusions which produce discontinuities by bridging narrow gaps within road segments. This is shown in Fig. 1(b) with the example labeled as "Postprocessing and segment linking." The "RPS" module separates the roads from protruding (or attached) nonroad regions, thereby improving the accuracy. Finally, a Medial-Axis-Transform (MAT)-based hypothesis verification is used to automatically complete the road network by integrating the contextual knowledge (Item 5 in Table II) into the framework. Fig. 1(b) also shows the results of these two final steps of processing at the end of Stage II of our proposed framework.

Algorithm 1 Multistage framework for road detection

- Input: Satellite Roadmap.
- Output: Segmented Image.
- ***Stage I***
  - Obtain edge maps of the image using DSM.
  - Compute the probability of class (region) label using P-SVM.
  - Integrate region and edge information (outputs of previous two steps) using CSNN-CII [52]:
    - Initialize all neurons in CSNN-CII using the probabilities obtained from P-SVM.
    - Iterate and update the probabilities and edge map to get an improved segmented map [52].
- ***Stage II***
  - Postprocess the CSNN-CII output to remove undesired patches and unnecessary artifacts.
  - Use segment linking to bridge the gaps.
  - Perform RPS algorithm to eliminate nonroad regions (false positives).
  - MAT-based road hypothesis verification for filtering road layers.

Results will be shown using four categories of database of high-resolution satellite images from the following areas: 1) developed suburban; 2) developed urban; 3) emerging suburban; and 4) emerging urban. Performance analysis is presented using completeness and correctness measures [53] and compared with two state-of-the-art methods and a commercial package FeatureObjeX [51]. In the next section, we present the description of the different modules of Stage I of our proposed framework, along with intermediate results of processing using a satellite image sample. Details of Stage II will be discussed in Section V.

IV. Road Feature Detection (Modules in Stage I)

Stage I of our proposed framework deals with the extraction of salient features and is described in this section in detail. Section IV-A describes the extraction of DSM-based edge features. Road segment extraction using SVM is discussed in Section IV-B. Section IV-C deals with the process of integrating region and edge information.

A. DSM-Based Road Boundary Extraction

Roads are expected to be locally linear. Hence, we extract the local orientation from the image of the road network. Extracting linear features from satellite images has been of interest to pattern recognition community [41], [54]–[58]. Our method of obtaining the dominant direction using Singular Value Decomposition (SVD) of the gradient (obtained using multiscale [59] 1-D Canny operator [60]–[62]) matrix, for orientation estimation to extract road segments, is efficient and produces more robust results.

For each pixel in the image, we first calculate the local image gradients using the 1-D Canny operator [60] and then perform SVD of the covariance of the gradient matrix. The gradient of image $f(x, y)$ at point (x_k, y_k) is denoted by

$$\nabla(f(x_k, y_k) = \left[\frac{\delta f(x_k, y_k)}{\delta(x)}, \frac{\delta f(x_k, y_k)}{\delta(y)}\right]^{\mathrm{T}} \tag{1}$$

which involves two pairs of 1-D processing along orthogonal directions (for details, see [60] and also [61]–[63]), where $\delta f/\delta(x) = f_y \bigotimes \delta G(x)$, $\delta f/\delta(y) = f_x \bigotimes \delta G(y)$, $f_y = f \bigotimes G(y)$, $f_x = f \bigotimes G(x)$, $\bigotimes$ indicates a 1-D convolution function, and

$$\begin{aligned} G(x) &= \frac{1}{\sqrt{2\pi}\sigma_1}\exp\left(\frac{-x^2}{2\sigma_1^2}\right) \\ G(y) &= \frac{1}{\sqrt{2\pi}\sigma_1}\exp\left(\frac{-y^2}{2\sigma_1^2}\right) \end{aligned} \tag{2}$$

$$\begin{aligned} \delta G(x) &= \frac{-x}{\sqrt{2\pi}\sigma_2^3}\exp\left(\frac{-x^2}{2\sigma_2^2}\right) \\ \delta G(y) &= \frac{-y}{\sqrt{2\pi}\sigma_2^3}\exp\left(\frac{-y^2}{2\sigma_2^2}\right). \end{aligned} \tag{3}$$

A similar processing is hence applied along the x and y directions, where the two operators interchange their directions of processing. A major problem of computing the edge map is the noise sensitivity of the gradient operator. One solution to suppress the effect of noise is to use a larger window, since more neighboring gradients will be used to get the estimate and the averaging process will depress or eliminate the noise effect. However, this will create a localization error. A mechanism with both noise robustness and feature localization is needed. A multiscale model [59] provides an efficient way to combine the information from coarse to fine scales. This method is described in Appendix D. As the data are multispectral (RGB), we have used this processing separately across all the bands and then used a vector addition to obtain the gradient (∇f) at each pixel (x_k, y_k).

In order to get the local orientation estimate, we obtain the gradient vectors over a window size of $N \times N$, around each pixel, as $GF_i = \nabla f(x_i, y_i)$, $i = 1, 2, \ldots, N^2$ [see (1)]. We then compute the SVD [64] of the covariance of the gradient matrix [63] obtained for pixels within a window of size "$N \times N$." The SVD of the covariance of the gradient map is computed as

$$\mathrm{Cov}(G) = USV^{\mathrm{T}} \tag{4}$$

where $\mathrm{Cov}(G) = \sum_i GF_i.GF_i^{\mathrm{T}}$, U is an orthogonal 2×2 matrix, in which the first column represents the dominant orientation of the gradient field, S is a 2×2 matrix of eigenvalues representing the energy along the dominant directions, and V is an orthogonal matrix of size 2×2, representing each vector's contribution to the singular value.

DSM is computed as the ratio between the singular value of the major axis and the sum of the singular values. This measure approaches "1" for an elongated shape; DSM is defined as

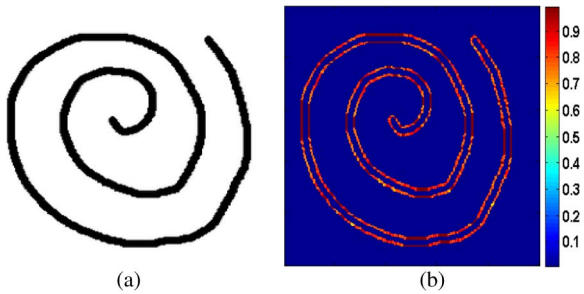
$$\mathrm{DSM} = \frac{s_1}{s_1 + s_2}, \qquad s1 \geq s2 \geq 0. \tag{5}$$

When all the gradient components have the same direction, only one singular value (s_1) is nonzero, which, in turn, makes the DSM value equal to 1. If both the singular values are equal (nonzero), the DSM value is 0.5. The range of values of the DSM thus lies in the range [0.5–1]. We use the DSM measure to distinguish between scattered (or disoriented) image patterns and an image region with a uniformly orientated pattern. If the DSM is less than a threshold ρ $(0.5 < \rho < 1)$, it is very likely that the corresponding image block is noisy, contains no dominant orientation, and is hence considered as background (nonroad). The concept of DSM is similar to that used in [63] to detect corners. The default values for the parameters used in this process are given in Table III.

The result of this orientation detection process is robust and performs well on both synthetic and natural images. The results of DSM, which is an unsupervised classification method, includes the orientation information about streets/roads, rivers, and other linear structures. However, if additional information and knowledge are available, nonroad structures may be masked [65]. Fig. 2 shows the result of DSM applied to a synthetic image. We observed that the most dominant structure has the DSM value "1." Image patches which do not have any oriented pattern produce a DSM value just greater than "0.5," and these patches are considered as nonroads. The color bar in

TABLE III
VALUES OF THE DIFFERENT PARAMETERS USED IN OUR PROPOSED APPROACH

Road image type	DSM				CSNN-CII		Post Processing		Segment linking		RPS		MAT	
	σ_1	σ_2	N	ρ	δ	n	T_E	T_A	Distance	Angle	T_C	Perimeter	th1	th2
Suburban	2	2.5	9x9	0.6	0.04	31x21	0.7	50	10	30	0.01	40	10	20
Urban	3	3.5	11x11	0.7			0.6	35						



(a) (b)

Fig. 2. Results of DSM on a synthetic image. (a) Input image. (b) Corresponding DSM with the color bar indicating the values of DSM in the normalized range [0–1].

Fig. 2 show the DSM value in a normalized range [0–1] (actual range [0.5–1]). The measured orientations and strengths accurately reflect the linearity in the oriented structures of the input image. This method fails in case of junctions, certain textures, and overlapping objects, which do not contain a dominant local orientation.

B. Road Segment Extraction

1) SVM: Region-based techniques are used to segment an image into clusters. SVM classifiers have yielded some excellent results in many such application domains. The main idea of SVM is to separate the classes with a hyperplane surface so as to maximize the margin among them. A brief discussion on SVM and learning methodology is given in Appendix A. A polynomial kernel of degree two was used due to its superiority over other kernels for most of the applications.

The interaction of the user is necessary to generate the set of training samples for SVM. Positive samples as well as negative samples (template windows of size 21 $\times$ 21) representing the road and nonroad classes form the training set, which are collected with the aid of a human being (manually). The negative samples were obtained from image regions which do not contain any roads. In this case, 7200 negative samples are selected manually from these image regions. Positive samples (5000) were similarly collected from image regions which contain roads. For each subimage, representing either a road or nonroad region, a set of multispectral (three-band, RGB) histogram-based features, such as mean, standard deviation, skew energy, and entropy, are computed using a window of size 21 $\times$ 21 around a pixel (for road or nonroad class) to form a feature vector for each sample. Hence, each sample is represented by a feature vector, which is computed from the multispectral data within a window around a pixel. These sample points are demarcated manually within subimages (regions). This manual process of demarcation and selection of sample points have been done over 100 images. The feature vectors for these training samples were used for training the SVM to obtain the alpha values. SVM is then used to classify every pixel into either road or nonroad groups based on the sign of the discriminant function $(y = \mathrm{sgn}(f(x)))$.

SVM is principally a binary classifier. Considering a classifier to produce a posterior probability is very useful in practical recognition problems. Posterior probabilities are also required when a classifier is making a small part of an overall decision, and the classification output is combined for overall decision. However, SVM [66] produces an uncalibrated value that is not a probability. In the next section, we use a mechanism to obtain probabilistic classification of pixels as roads or nonroads, using soft-class labels from SVM.

2) Soft Class Labels Using P-SVM: SVM, in general, provides a crisp class label and does not provide any estimation of their classification confidence. Thus, SVM does not allow us to incorporate any *a priori* information. Hence, we use P-SVM to produce posterior probability $P(\mathrm{Class}|\mathrm{Input})$. The posterior probability outputs of SVMs are based on the distance of testing vectors and support vectors. Following a method presented in [67], a sigmoid model is used to map binary SVM scores into probabilities as shown in the following:

$$\mathbf{P(y=1|f_s)} = \frac{\mathbf{1}}{\mathbf{1+\exp(Af_s+B)}} \tag{6}$$

where "$\mathbf{y}$" is the binary class label and "$\mathbf{f_s}$" is a scalar obtained from the output of SVM decision function [(11) in Appendix A]. The parameters A and B (scalars) are obtained using the process of Maximum Likelihood Estimation on the training set, by minimizing the log-likelihood (negative) of the training data $(\mathbf{f_s}, \mathbf{y})$. An image block is said to be a road if the value of P, computed at the block, is larger than a predetermined threshold. As a result, the model has a probabilistic output for further processing. The probabilistic output of a classifier makes it possible to use existing results for fusion theories, particularly in cases when a classifier is making a small part of an overall decision.

The sample subimages shown in Fig. 3 illustrate the discriminative feature between road and nonroad samples of each category. Fig. 3 shows 80 samples each from the road and nonroad classes, among those used for training. They are bunched in groups of 20 (in two rows) per category. Fig. 3(a) shows the road samples, whereas Fig. 3(b) shows the nonroad samples for four categories: developed suburban, developed urban, emerging suburban, and emerging urban areas. Spectral characteristics vary for both the classes, which is analyzed by P-SVM. The local homogeneous orientation visible for the road class, as seen from the samples in Fig. 3(a), will be captured by P-SVM, whereas nonroad structures, as shown in Fig. 3(b), will produce distributed orientations. Our system is trained with 5000 samples of road and 7200 samples of nonroad classes. Once the classifier is trained, it is asked to

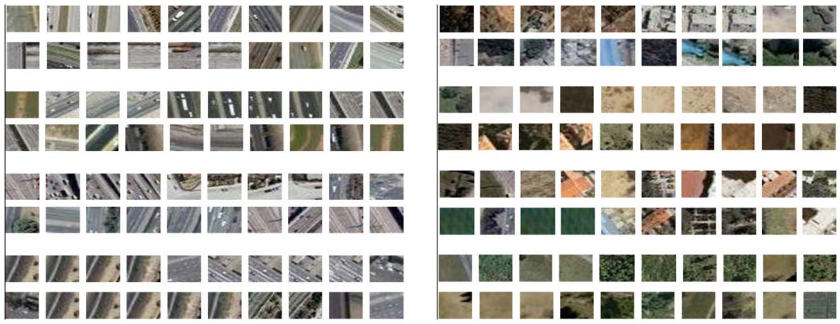


Fig. 3. Eighty multispectral samples (of size 21 × 21) each for (a) road and (b) nonroad classes used for training the P-SVMs.

predict the labels for the test image pixels. Experimental results for different scenarios, namely, urban and suburban areas of developed and emerging countries and their discussions, are presented in Section VI. In the next section, we discuss the method of fusing the two complementary information (segment class from P-SVM and linear edge map obtained from DSM) using a CSNN-based integrator.

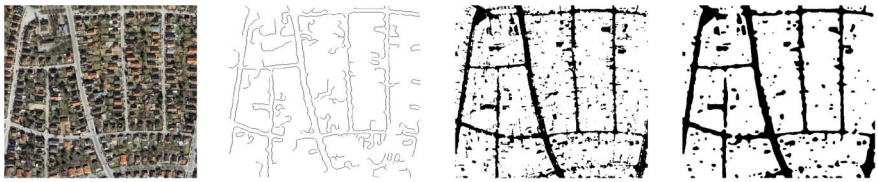
C. CSNN-CII to Integrate Road Regions and Boundaries

The quality of the edges vary due to noise, changes in radiometry of the road surface, and its background occlusion. Thus, edge extraction from satellite images often delivers partly fragmented and erroneous results. Therefore, the extraction of edges based on local criteria often results in an incomplete detection of a few significant edges or in the detection of many irrelevant edges. Attributes describing geometrical and radiometric properties of the line segments can be helpful in sorting out the most probable false alarms. However, these attributes may be ambiguous and are not considered to be reliable enough when used alone. Region-based segmentation produces oversegmentation, whereas edge-based segmentation may lead to undersegmentation. We used a fusion strategy proposed by Lalit *et al.* [52], which uses a constraint to iteratively correct both these erroneous outputs to produce a better result. The method is described next for the sake of completion in this paper.

CSNN-CII is a feedback neural network, where each neuron contains two fields: probability and rank. The rank field stores the rank of the probability in a decreasing order for that neuron. We exploit the soft class labels produced by P-SVM to compute ranks, which, in turn, is used to initialize the interconnection weights of the CSNN. The initial class probabilities are also obtained from the P-SVM [67] output. In our case, the size of CSNN is $I \times J \times K$, where $I = J = M$ ($M * M$ is the resolution of the image) and $K = 2$ (the number of classes). After initialization, the CSNN-CII is allowed to "relax"—this is an iterative process which helps to refine the P-SVM output, using a set of constraints formulated based on the principle of region segmentation in an image (for details, readers must refer to [52]). Discussion and details of CSNN architecture are given in Appendix B. In addition to region-based constraints, CSNN-CII also incorporates edge constraints. The initial edge maps are also obtained using the DSM-based technique for road edge extraction. The number of neighbors considered for computation during the process of relaxation is determined using edge information.

CSNN-CII [52] uses the concept of a dynamic window for computation, which helps to iteratively integrate region and edge information for better segmentation. CSNN-CII neurons are initialized using probabilities obtained from P-SVM-based segmentation results. Neurons in a layer hold the probability (confidence) that pixels belong to the segment (class) represented by the corresponding layer. The rank field is initialized based on the rank of the probability in the nondecreasing order (see [52] for details). The interconnection weights of the CSNN-CII are computed only for those neurons which are within the effective size of the dynamic window. The size of the rectangular template (dynamic window) is fixed at (31 × 21). However, the effective set of pixels used for computation varies based on the location and orientation of the edge within the template (dynamic) window. Thus, the effective width of the dynamic window is based on the presence of edge information around the pixel at the center of the window. The advantage of using a dynamic window manifests at the region (road segments in our case) boundaries, where the segmentation algorithm (SVM-based) generally fails to provide the correct class levels. Here, only a majority of the neurons (corresponding to the pixels) which correspond to a single class are chosen for processing and other neurons which may confuse the network are not used for computation. CSNN-CII uses the "dynamic window" to adaptively select an appropriate neighborhood for computation. The number of neighbors considered for computation is determined using the edge information. The dynamic window thus controls the effective width for computation. The size of the template is, thus, not dynamic; it is rectangular and oriented along the edge. The effective width is dynamic (again, window size is constant), as the effectiveness is based on edge pixels.

The stopping criterion is based on the presence of the edge pixels. Hence, this process helps to mutually exploit both the complementary information of regions and edges inside the window. The window is considered to be dynamic (or adaptive), as its effective size depends on both these pieces information—one (region) for initial estimation and the other (edge) for convergence. The optimal size of the dynamic window was obtained empirically as 31 × 21 (size is constant, but orientation and effectiveness are based on the presence of edge pixels). Lalit *et al.* [52] used a square window, whereas we propose the rectangular window in our work. The orientation



(a) (b) (c) (d)

Fig. 4. Results of Stage I on a satellite image of an urban scene of size 512 × 512. (a) Input image. (b) Corresponding DSM output. (c) Result of P-SVM-based segmentation. (d) Segmented map obtained using CSNN-CII.

of the rectangular window is set equal to the orientation of the gradient value at regions with large DSM values (i.e., at edges only). For all other regions, the dynamic window is considered to be square (21×21). It was observed from experimentation that when a larger window size was used, small regions (or small sections of a region) were merged with larger adjacent regions. The use of a smaller window size makes the CSNN-CII take a longer time to converge to the final solution.

Results of integration using output of P-SVM and DSM are shown in Fig. 4. Fig. 4 shows the results of processing at the end of Stage I on a satellite image of an urban scene. Fig. 4(a) shows the input image of size $(M * M)$. Fig. 4(b) shows the edge map obtained using the algorithm proposed in Section IV-A. Fig. 4(c) shows the segmented map for the satellite image shown in Fig. 4(a) obtained using the method described in Section IV-B1. Fig. 4(d) shows the segmented output obtained using the proposed method (CSNN-CII). This map is obtained using the binary decision of "max rule" (see end of Appendix B) applied on the probabilities of the two classes (road and nonroad) for each pixel after convergence. It can be observed that the segmented map has improved significantly using the proposed method.

V. Road Network Formation (Modules in Stage II)

The output of roads detected at Stage I reveals two main characteristics: 1) discontinuity due to occlusion and 2) nonroad regions with roadlike spectral response. Thus, there are two main concerns in a road network formation. The first problem is to bridge the gaps caused by artifacts such as trees, shadows, or occlusions from buildings. This affects the completeness of the extracted road network and the topological correctness of the network. The second problem is to eliminate the false region detected as roads. This can improve the correctness of the extracted road network. The basic procedures of road network formation are studied with an emphasis on grouping the road segments. Road network formation enables the link between individual road segments to form meaningful road lines, and it generates the topological structure of the network. It usually includes a set of processes such as bridging gaps between road segments and removing overshooting and undershooting.

A. Postprocessing

At the outset of Stage II of processing, a refinement process is necessary to eliminate the false segments which do not belong to roads. The result of CSNN-CII integration produces a few undesired patches, which do not correspond to road segments. In the case of satellite images, a few undesired or noisy structures will be erroneously classified as road segments. To eliminate these false alarms (segments), we use connected component labeling [68] to extract the disjoint segments from the output of our algorithm. Segments with an area less than a prefixed threshold T_A are deleted.

The major and minor axis lengths of each component are computed using normalized second central moments [69] for each segment as shown in the following:

$$\mu_{20} = M_{20} - \bar{x}M_{10} \quad \mu_{02} = M_{02} - \bar{y}M_{01}$$

where

$$\bar{x} = \frac{M_{10}}{M_{00}};\ \bar{y} = \frac{M_{01}}{M_{00}};\ M_{pq} = \sum_x \sum_y x^i y^i I(x,y).$$

We computed the ratio of the major axis length to the minor axis length of each component as

$$E = \mu_{20}/\mu_{02}.$$

Components having a value of E less than a threshold T_E are usually nonroad structures and hence deleted. The steps of the algorithm, depicting this postprocessing module, are given next in Algorithm 2.

Algorithm 2 Steps of postprocessing to refine the result
1) Compute the connected components.
2) Obtain area (A) of each connected component.
3) Compute eccentricity (E) of each connected component.
4) For each component
    if$(E \leq T_E)$**then**
        delete that component
    else
        if$(A \leq T_A)$**then**
            delete that component
        end if
    end if

B. Segment Linking

A road structure may have discontinuities or appear to be fragmented into multiple segments, due to the noise in the original image, occlusion by other structures, or misclassification of constituent pixels. A pair of nearby collinear segments separated by a small distance should be considered for merging

to properly regenerate the road segment. The following two factors must be considered during any merging process: 1) the distance between the end points and 2) the angle of orientation between the segments. We used a region linking algorithm [70] to eliminate the discontinuities between a pair of road segments.

Initially, a dilation operation is performed on the input image. Since dilation is an operation that thickens or grows objects in the original image, edge segments which are very close to each other are automatically linked. In our algorithm, the structural element used for the dilation operation is a disk of radius 10. The image is then thinned [71], and the edges are broken down into smaller straight line edge segments. For each segment, the best neighbor is determined based on the difference in direction (orientation analysis) and the minimum distance between the end points.

C. RPS

It is necessary to eliminate some large patches of nonroad structures which appear to be fused to roads. These patches are man-made structures, such as rooftops and parking lots, with similar spectral characteristics as roads. The proposed algorithm for RPS is based on a part segmentation [72] algorithm, consisting of the following steps:

1) Smooth the inner and outer image contours.
2) Compute the smoothed curvature of the contours.
3) Determine the local extrema, where the derivative of smooth curvature equals zero and curvature value is greater than a threshold.
4) Compute convex/concave dominant points (CDPs) at which the interior angle is greater/less than 180° by tracing the outer/inner contours of the region. (See Fig. 19 in Appendix C, for illustration.)
5) Compute effective convex $(CDP_{\rm cx})$ and concave $(CDP_{\rm ce})$ dominant points, on outer and inner contours, respectively, by logical AND operation of the output in steps 3) and 4).
6) The CDPs (both $CDP_{\rm cx}$ and $CDP_{\rm ce}$) are moved along the normal for a fixed number of iterations (all the CDPs must move simultaneously) on the respective contours.
7) A moving CDP will stop (freeze) only if it touches another moving CDP or a point on the same contour within a specified path distance from it. If the outer contour of the segment touches the boundary of the image, then the respective CDPs are not frozen.
8) Trace back all the frozen CDPs and join the pair of corresponding CDPs or the CDP and the contour point using a line segment.
9) For each line segment obtained in step 8), form two adjacent regions within a closed contour, using the line as the new boundary.
10) Merge the new pair of adjacent regions, if they have similar structural properties (orientation of line segments near the CDPs).
11) Set a threshold and eliminate all the connected components with the area below the threshold.

Of the steps aforementioned, the discussion and details of two crucial steps 2) and 4) are given in Appendix C. A synthetic example has been used in Appendix C to illustrate the RPS algorithm. This processing is similarly performed on both the inner and outer contours of the roadmap to remove patches of nonroad segments.

D. MAT-Based Hypothesis Verification

We now need to remove the patches appearing due to parking areas and rooftops which are attached with road segments. To address this issue, we introduce a stable and robust decomposition of regions into their salient parts. The region decomposition is computed from the medial axis skeleton of the image. Each subregion corresponds to a portion of a branch of the medial axis skeleton.

A MAT-based road hypothesis is generated using the knowledge about the properties of road segments (Table II, item 5). This requires *a priori* knowledge of contextual information (i.e., relations between roads and other objects like trees, parking lots, and buildings) to be integrated into the road extraction methodology [73]. This high-level information is used to automatically complete the process of road network extraction from satellite images. The hypotheses of a road segment is based on the assumption that the widths of roads do not vary abruptly. The road network (obtained at the output of the stage described previously in Section V-C) is thinned by a morphological operation [74], [75]. Parts of the road segments with a local width less than a fraction of the average width are preserved. Else, a simple thresholding operation removes the rest as nonroad segments, thus increasing the accuracy of road extraction. The steps of the algorithm depicting the process is given in Algorithm 3. The four major substeps of MAT-based processing are discussed in the following:

Algorithm 3 MAT-based road hypothesis verification

- Input: Binary image (I), after RPS.
- Output: Segmented Image.
- Steps:
  1) Compute $I_{\rm dist}$—distance transform of I.
  2) Compute $I_{\rm skel}$—the MAT of I.
  3) Compute the branch points as $bp(I_{\rm skel})$ and end points as $ep(I_{\rm skel})$.
  4) Identify subcomponents in $I_{\rm skel}$ using bp and ep.
  5) For each subcomponent do
     - Compute the width $(W_{\rm sub})$ using $I_{\rm dist}$ and length $(L_{\rm sub})$.
     - Compute variance $(V_{\rm sub})$ and mean $(MW_{\rm sub})$ of width along skeleton trajectory.
     - Select and add in set $Salient(I_{\rm skel})$, if $(V_{\rm sub}) < th1$ and $(MW_{\rm sub}) < th2$; else, remove it.
  6) Perform morphological dilation operation on $Salient(I_{\rm skel})$ with a structuring element (3×3).
  7) Perform logical AND operation of the result from step 6) with the input Image I.

---

1) Distance Transform [Step 1) of Algorithm 3]: Distance transform [76] is computed normally for a binary image where each pixel in the binary image is assigned a number that is the distance between that pixel and the nearest nonzero pixel of the binary image. Formally, for a binary image I, its distance transform image I^d is given by the following:

$$I^d(x,y) = \begin{cases} 0 & \text{if } I(x,y)\ is\ 0 \\ d & \text{otherwise} \end{cases} \tag{7}$$

where d is the Euclidian distance of pixel (x,y) to the nearest nonzero pixel. From a measurement perspective, the Euclidean distance is the most useful because it corresponds to the way objects are measured in the real world. The Euclidean distance metric uses the L_2 norm and is defined as

$$\|x,y\|_{L_2} = \sqrt{x^2+y^2}. \tag{8}$$

It contains optimum meaningful information about the neighborhood of a pixel. The width of each subcomponent obtained using MAT is computed from the distance transform.

2) MAT [Step 2)]: The medial axis skeleton is a thin line graph that preserves the topological relationships of the salient parts of a region. The skeleton has often been cited as a useful representation for shape description, region interpretation, and object recognition. The skeleton provides a decomposition of the region into salient subparts. It also provides a description of the connectivity of the subparts. Unfortunately, the computation of the skeleton is extremely sensitive to variations in the bounding contour of a region. Tiny perturbations in the contour often lead to spurious branches of the skeleton. It is nontrivial to determine which of the branches are spurious and which correspond to significant subregions. There have been numerous attempts to find a robust algorithm for computing the medial axis skeleton (see, for example, [75], [77], and [78]). Most algorithms use some deviation of morphological thinning. We have used an implementation of MAT, as given in [68]. Often, spurious branches are eliminated based upon some approximate property of the bounding contour or some property of the branch itself. Each branch of the skeleton corresponds to a subregion of the region bounded by the contour.

3) Computing Width of Road Segments [Step 5)]: The variation of the width is used as a criterion, because nonroad segments, in most cases, produce nonparallel road boundaries. The algorithm for determining the width of a road uses the skeleton and its distance map (I_{dist}) of the binary input image. The skeleton acts as a guide for tracking the distance map, and distances at all points along the skeleton (which coincide with the center of the objects in the distance transformed image) are recorded to compute the road width. Finally, the recorded results are doubled to obtain the estimate of road width.

4) Identifying the Salient Skeletons [Steps 6) and 7)]: Road segment hypothesis is framed according to the properties of roads. The hypothesis used in this paper is as follows: There is a constancy in the width of the road segment, and the mean width of the road segment should be within a specific threshold. This threshold is computed from *a priori* information about the width of the road in high-resolution satellite images. From the resulting set of skeletons for road segments, a graph is constructed. The road segments define the edges of the graph, and their end points and junctions produce a set of vertices. Various attributes computed for each segment (or edge) are as follows: 1) length of the segment; 2) width of the segment; and 3) standard deviation (σ) of width. While traversing the graph from the source node (arbitrary), the associated attributes of each edge are verified according to the hypothesis. Segments which do not satisfy the hypothesis (step 5) in Algorithm 3) are removed. The resulting graph gives us the salient skeleton. Now, the salient skeleton image is dilated iteratively using a structuring element of size 3×3 [74]. The number of times the dilation operation has to be performed is dictated by the width of the road (estimated earlier). A logical AND operation of the dilated salient skeleton with the original image gives the final road network.

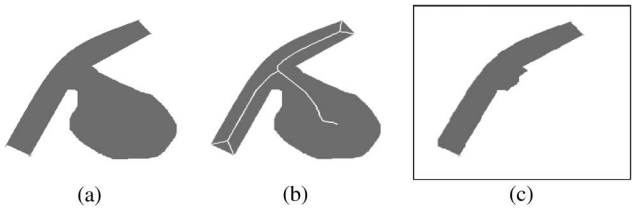


Fig. 5. Results of MAT shown on synthetic images. (a) Input image. (b) MAT superimposed on the input image. (c) Results of MAT-based hypothesis verification.

Fig. 5 shows the result of this algorithm on a synthetic image. Fig. 5(a) shows the input image, Fig. 5(b) shows the MAT superimposed on the input image, and Fig. 5(c) shows the final result after processing. As seen in Fig. 5(c), a part of the image which is a nonroad structure is removed from the input image shown in Fig. 5(a).

Fig. 6 shows the intermediate results of Stage II of our proposed algorithm. Fig. 6(a) shows the results of postprocessing obtained on the image of Fig. 4(d). Fig. 6(b) shows the results of the segment linking algorithm, which links a few disconnected segments together, thus enhancing the accuracy of detection. Some nonroad patches associated with the road structure exist. Fig. 6(c) shows the result of the RPS algorithm applied to the image in Fig. 6(b). It is observed that most of the nonroad structures have been eliminated, enhancing the accuracy further. Fig. 6(d) shows the result of the MAT-based hypothesis verification algorithm applied to the image in Fig. 6(c) to eliminate any big nonroad structures associated with the road structure. Since this example does not have any nonroad structure to eliminate, the result of this module is similar to the previous module.

The power of MAT-based processing (final step in Stage II) is shown in Fig. 7 with two real images. Fig. 7(a) shows the input images, and Fig. 7(b) shows the result obtained after RPS module (see Section V-C) overlaid by MAT. It is observed that RPS fails to eliminate large nonroad regions associated with the road region. Fig. 7(c) shows the result of the MAT-based hypothesis verification module. As seen in Fig. 7(c), large nonroad structures are eliminated using this approach. Thus, the accuracy of extracting roads from satellite images have improved to a great extent, producing the desired road layer.

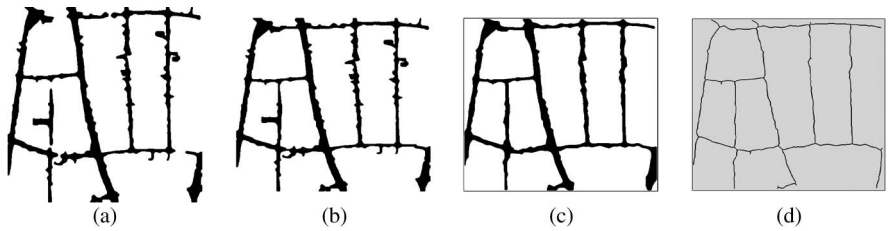


Fig. 6. Results of processing modules in Stage II on a satellite image [in Fig. 4(a)] of an urban scene: (a) Postprocessing, (b) segment linking, (c) RPS, and (d) MAT.

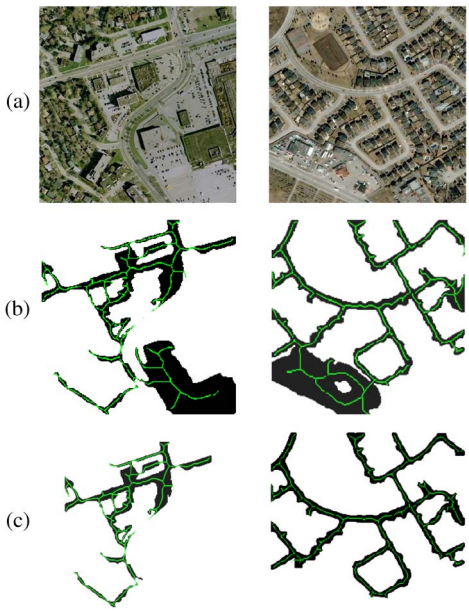


Fig. 7. Results of MAT shown on real images. (a) Input image. (b) MAT superimposed on RPS output for the image in (a). (c) Results of MAT-based hypothesis verification.

Fig. 8 shows the centerline of the road extracted using our proposed algorithm from the satellite images of four different categories. Fig. 8(a) shows the input satellite images from four different categories. Results of our proposed algorithm are shown in Fig. 8(b). Fig. 8(c) shows the reference maps obtained manually from the input images. The accuracy of the proposed technique is excellent, based on visual observations, which is also verified quantitatively using extensive experiments, as discussed in the next section.

VI. Experimental Results and Comparative Study

We now describe the results of experimentation using our proposed framework. The performance of the proposed method is verified on four categories of multispectral satellite images of size 512 × 512 each. The data set (200 images) used here consists of four categories of satellite images, with 50 images per category. This is split equally into two parts—one (100) used for training and another for testing. We have trained four different P-SVMs, one for each category, with 25 images for a particular category. The performance of the proposed technique is compared with two state-of-the-art techniques, those of Tuncer [26] and Mokhtarzade *et al.* [49], as well as a free commercial tool for feature extraction (FeatureObjeX) [51].

FeatureObjeX [51] is a semiautomatic system, where it allows the user to select the training samples. Once the seed is created, intensity distributions are computed for a set of pixels around the seed, which are then used to fit a multivariate normal distribution. Each seed region is modeled by a Naive Bayes classifier [39]. Then, the likelihood of a given pixel is computed with respect to each of the seed distributions. If the likelihood of a particular pixel is the same or greater than the likelihood of the seed, then that pixel is classified as a target class. For the sake of comparative study, FeatureObjeX was used to segment the image into road and nonroad classes using color features only. Several configuration changes were made in FeatureObjeX before the tests to make it more efficient and closer to our requirement for working in road scenes over urban and suburban environments.

A. Data Set Description and Performance Measures

We created a database for satellite images with 1-m/pixel resolution from Wikimapia [79]. The quality of the image depends on the sensor used and the weather conditions. We screen captured 100 images of developed countries and 100 images of emerging countries, which are useful for our work. A part of the database is available in [80]. In our case, the place and date were not very critical and the only characteristic that we were looking for was the content of the images which had views of highways and roads. For creating the data set, we consider selected sections (512 × 512 pixels) of scenes from satellite images of 1-m/pixel resolution acquired from Wikimapia [79], which includes: 1) suburban and 2) urban areas from developed and emerging countries. Figs. 10(a) and 12(a) show four examples of multispectral images from suburban areas in developed and emerging countries, respectively, whereas Figs. 11(a) and 13(a) show four examples of images from urban areas in developed and emerging countries. For each image in the data set, a ground truth (road) map was also obtained using a human operator. The categorization of the images was done with the advice (used visual observation and geolocation) of a GIS expert. As the data were distributed in four groups of 50 images each, we trained four different P-SVMs with a distinct set of 25 images from each respective group. The rest (25 images) were used for the testing and performance analysis of the proposed multistage framework.

Positive samples representing the road class and negative samples for the nonroad class were manually chosen only from the training images. The negative samples were manually identified from the locations (regions) of an image which did not

Fig. 8. Results of MAT-based hypothesis verification shown on satellite images of four different categories: (a) Input image of four categories. (b) Results of our proposed method. (c) Reference map.

contain any roads and vice versa for positive samples. In total, 7200 negative samples and 5000 positive samples (windows of 21×21 size each) were manually identified from 100 images (25 distinct for each category) and used for the learning process (for four P-SVMs). For each window representing a road or (nonroad) sample, the histogram-based measures, such as mean, standard deviation, skew, energy, and entropy, were computed to form a 15-D feature vector for training the P-SVM. One thousand eight hundred (7200/4) feature vectors for negative samples and 1250 (5000/4) feature vectors for positive samples were used to train a single P-SVM, belonging to a particular category (with 25 images) of road network.

To assess the performance of the road extraction system, the length of the extracted road network (parameter obtained after morphological thinning) that falls within a prespecified range with respect to the reference road network was used for the calculation of accuracy measures. The road segments in the test sites were manually digitized to form the reference road network. This subjectively obtained reference road network covers all roads present in the image. Hence, this is used as a ground truth to estimate the accuracy measures for road extraction. Two measures used to evaluate the accuracy of the extracted road network [53] are defined as follows.

Completeness is defined as the percentage of the reference data which was detected during road extraction

$$\mathbf{c}ompleteness = \frac{length\ of\ matched\ reference}{length\ of\ reference}. \quad (9)$$

Correctness represents the percentage of the extracted road data which is correct

$$\mathbf{c}orrectness = \frac{length\ of\ matched\ extraction}{length\ of\ extraction}. \quad (10)$$

B. Results and Discussion

A separate training of the P-SVM was necessary for the four categories of image samples, as the spectral characteristics exhibited for roads were different for the four categories (areawise) of images in our study. The road intensity and contrast also vary between the four different types of image samples. Edge maps are obtained using the method discussed in Section IV-A. The CSNN-CII algorithm requires the probability values for all the pixels corresponding to each class in an image. The initial probabilistic values and segmented maps are obtained using the method discussed in Section IV-B. The proposed CSNN-CII-based algorithm iteratively shuttles between adding new and removing redundant edge pixels and hence inherently produces a correction mechanism to the process of fusion. The optimal values of the parameters of all the stages of our proposed approach are obtained empirically and are shown in Table III. In the future, any changes in the data sensor will cause the data and, hence, processing stages to be empirically tuned for optimal parameter selection. Fig. 9 shows the results of the intermediate stages of our proposed framework, using four images (one for each category).

To compare our approach with the recently published results in [26] and [49], we show the results with a few examples from the testing data set acquired from Wikimapia [79] in Figs. 10–13. Figs. 10 and 11 show the results obtained using the proposed methodology on satellite images of developed countries, whereas Figs. 12 and 13 show the results for emerging countries. Figs. 10–13(b) show the results of feature extraction using the FeatureObjeX tool for the images in Figs. 10–13(a), respectively. Figs. 10–13(c) show the results for the algorithm proposed in [26] for the images in Figs. 10–13(a). Figs. 10–13(d) show the extracted road segments from the input satellite images in Figs. 10–13(a), using the technique presented in [49]. Figs. 10–13(e) show manually plotted reference road layouts from the respective input images considered as ground

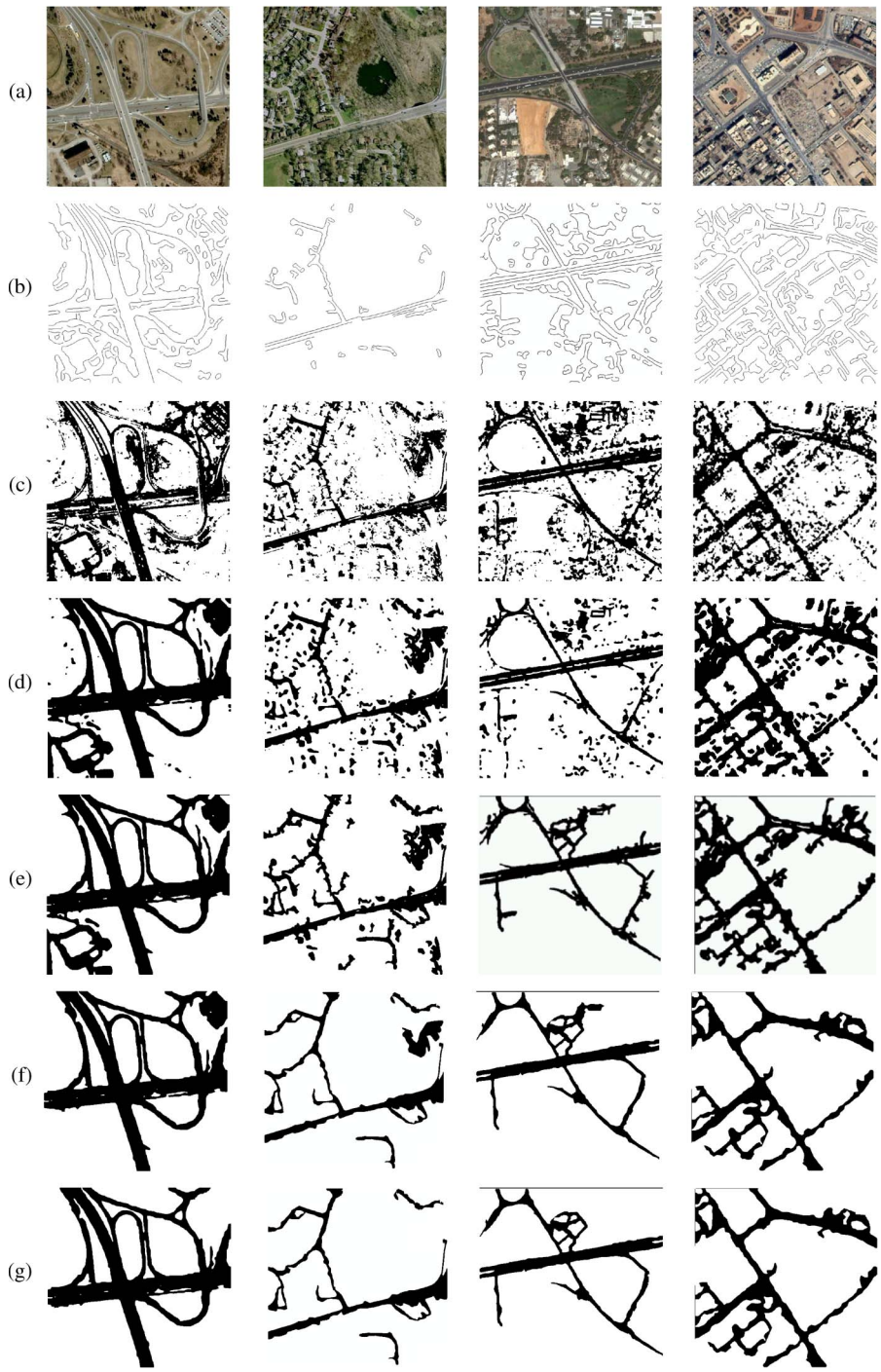


Fig. 9. Results of the intermediate stages of our proposed framework using four images (one for each category): (a) Satellite images of size (512 × 512), corresponding output (b) DSM, (c) results of P-SVM, (d) segmented map obtained using CSNN-CII, (e) results of postprocessing and segment linking, (f) RPS, and (g) MAT.

truth. It can be observed that the results of our proposed method, given in Figs. 10–13(f), are significantly better than those of other approaches and are quite close to the ground truth, as given in Figs. 10–13(e). Our system outperforms FeatureObjeX [51] and other state-of-the-art methods in all the cases.

Table IV gives the comparison of accuracy measures for the results shown in Figs. 10–13, using the completeness and correctness measures. The values given in bold font (four in each row) indicate the maximum performance measures for each of the four image samples (in a category) used in the corresponding figure, as specified in the leftmost column of Table IV. The performance measures given in Table IV show that our proposed method outperforms the other techniques in almost all the cases. In only two cases, the completeness measure of the FeatureObjeX tool is marginally better than our method. Tables V–VIII show the average classification accuracy obtained by analyzing images using the proposed method, FeatureObjeX [51], and the two state-of-the-art techniques in [26] and [49] over 25 test cases (images) of four different categories of image samples, respectively.

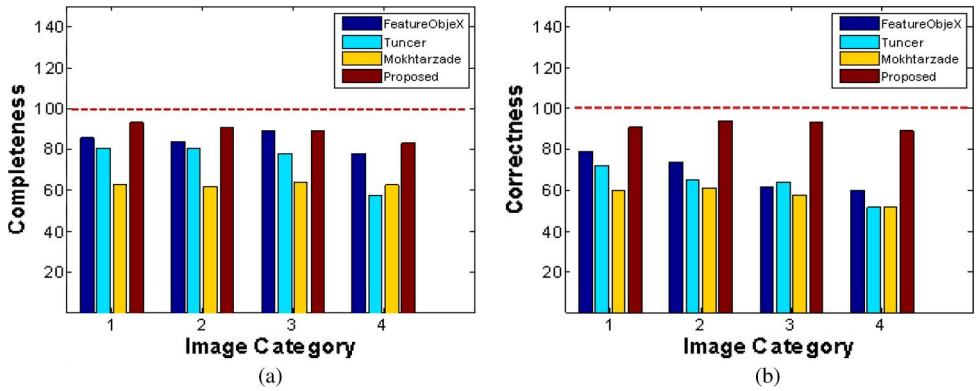


Fig. 10. Bar chart showing (a) completeness and (b) correctness measures averaged over 25 images for four categories.

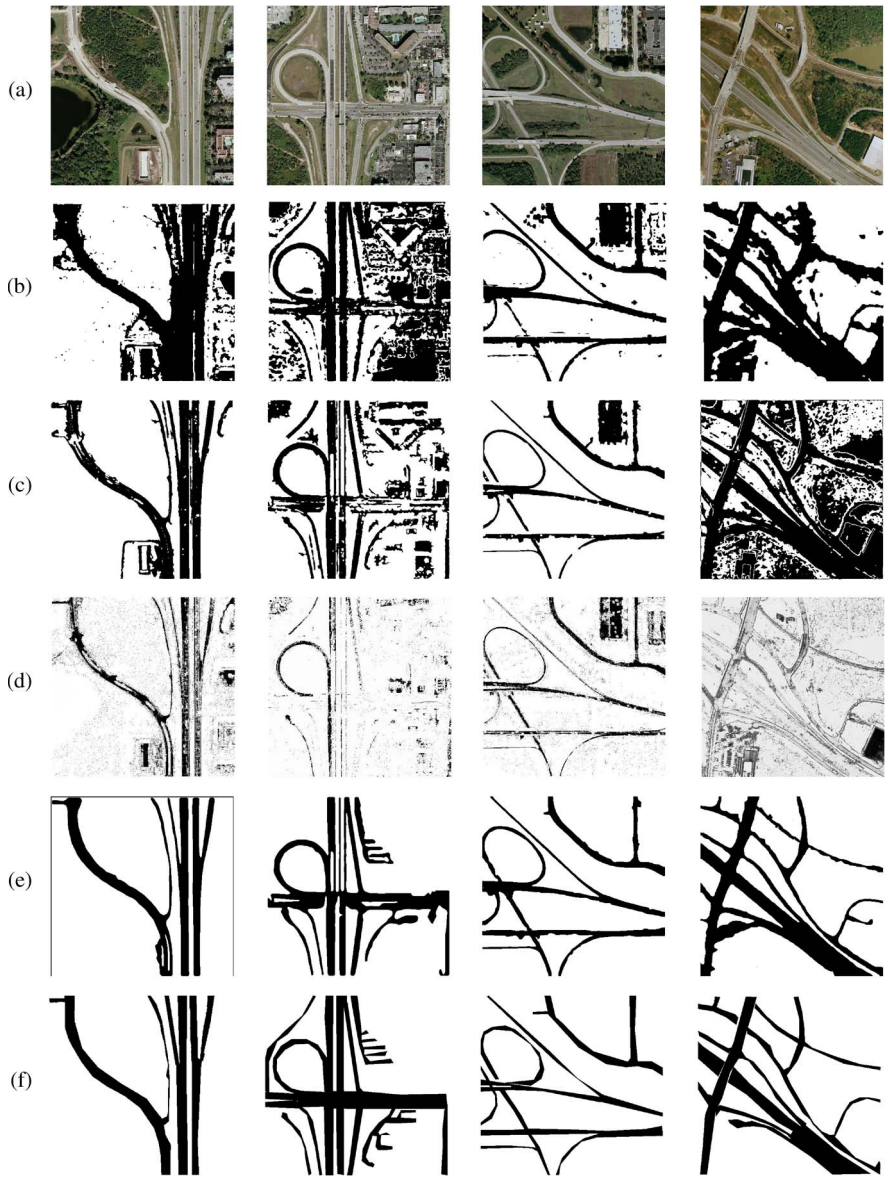


Fig. 11. (a) Four satellite images of size (512 × 512) from a suburban area of a developed region. (b) Results from FeatureObjeX [51]. (c) Results of the method proposed in [26]. (d) Results of the method proposed in [49]. (e) Results of our proposed method. (f) Hand-drawn (manual) road map.

It is observed from the results shown in Figs. 10–13 and Tables IV–VIII that the performance measure for our proposed algorithm is superior than those of the other methods. The results obtained using the proposed methodology are much superior to the methods presented in [26] and [49] and are very close to the manually drawn reference road network. Compared

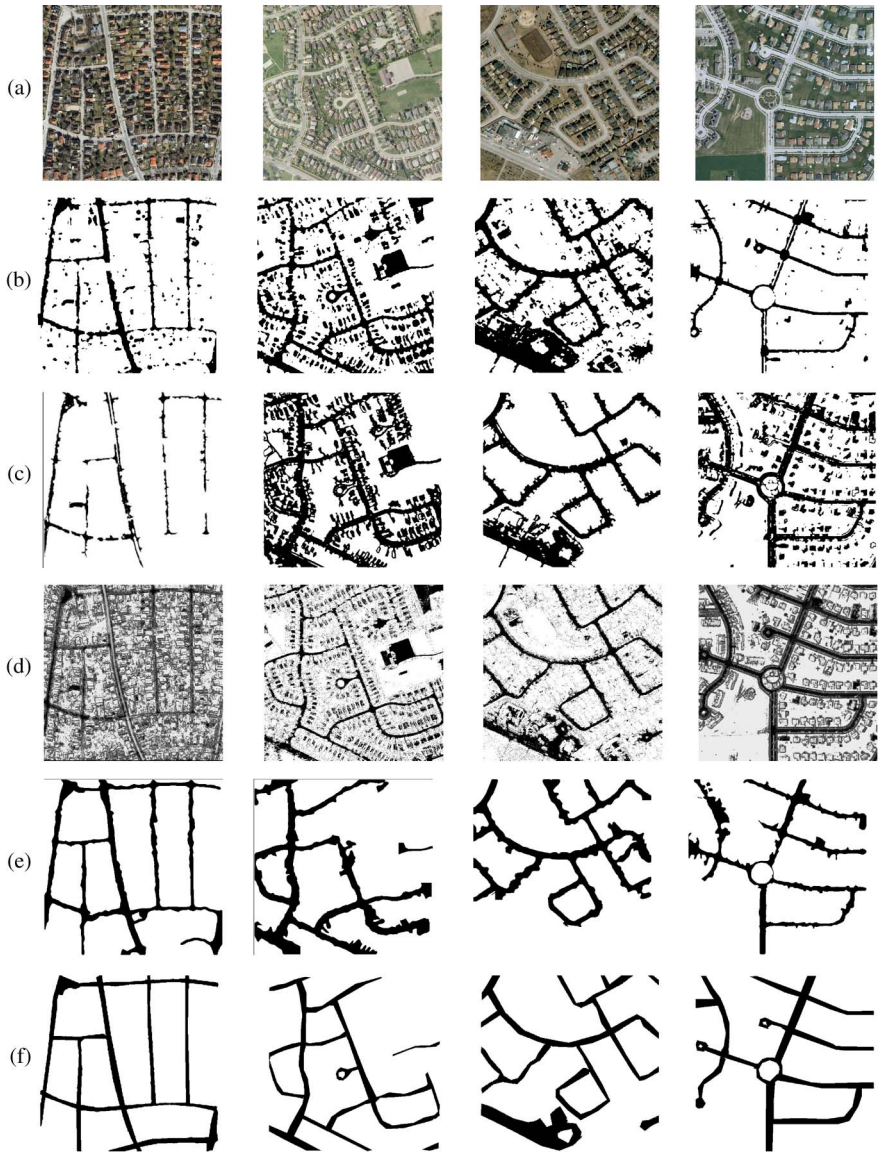


Fig. 12. (a) Four satellite images of size (512 × 512) from an urban area of a developed region. (b) Results from FeatureObjeX [51]. (c) Results of the method proposed in [26]. (d) Results of the method proposed in [49]. (e) Results of our proposed method. (f) Hand-drawn (manual) road map.

with our preliminary investigation in [62], the performance in terms of completeness and correctness measures has enhanced significantly. The region linking algorithm improves the completeness measure, whereas the RPS together with MAT-based hypothesis verification improves the correctness measure.

The correctness and completeness measures obtained for scenes from emerging countries (see Tables VII and VIII) are, in most cases, less compared with the scenes from the developed countries (see Tables V and VI). This decrease in accuracy for the scenes of emerging countries is expected due to the large numbers of linear nonroad features, four-way crossings, nonlinear road structures, and unplanned layouts. Comparing the results of developed urban and suburban scenes (see Tables V and VI), the performance of urban scenes is low, because of the distortions due to occlusion and shadows. It is obvious that images of urban areas exhibit a more complex structure than scenes of suburban areas, as the number of different objects and their heterogeneity are much higher in urban scenes. Some of the roads comprise several lanes that are linked by complex road crossings. Generally, as shown in Fig. 10, the extraction results for open landscape areas are nearly complete and correct. Suburban scenes of emerging countries are covered by vegetation. On the other hand, the spectral response of roads in urban areas are occasionally similar to the spectral response of open parks and rooftops, which, in turn, increases the false positives, thereby reducing the correctness measure. Overall, our proposed method outperforms FeatureObjeX [51] and the two state-of-the-art methods [26], [49] for observations averaged over images of 25 developed and 25 emerging areas.

The performances in terms of completeness and correctness measures at the different stages of our proposed approach are shown in Table IX. This shows the gradual stagewise improvement in the accuracy of the proposed framework. The average completeness and correctness measures are around 80% for the output of Stage I. Fig. 14 shows a bar chart of performance measures using the proposed method, FeatureObjeX [51], and the two state-of-the-art techniques in [26] and [49] over 25 images in four different categories. It can be observed

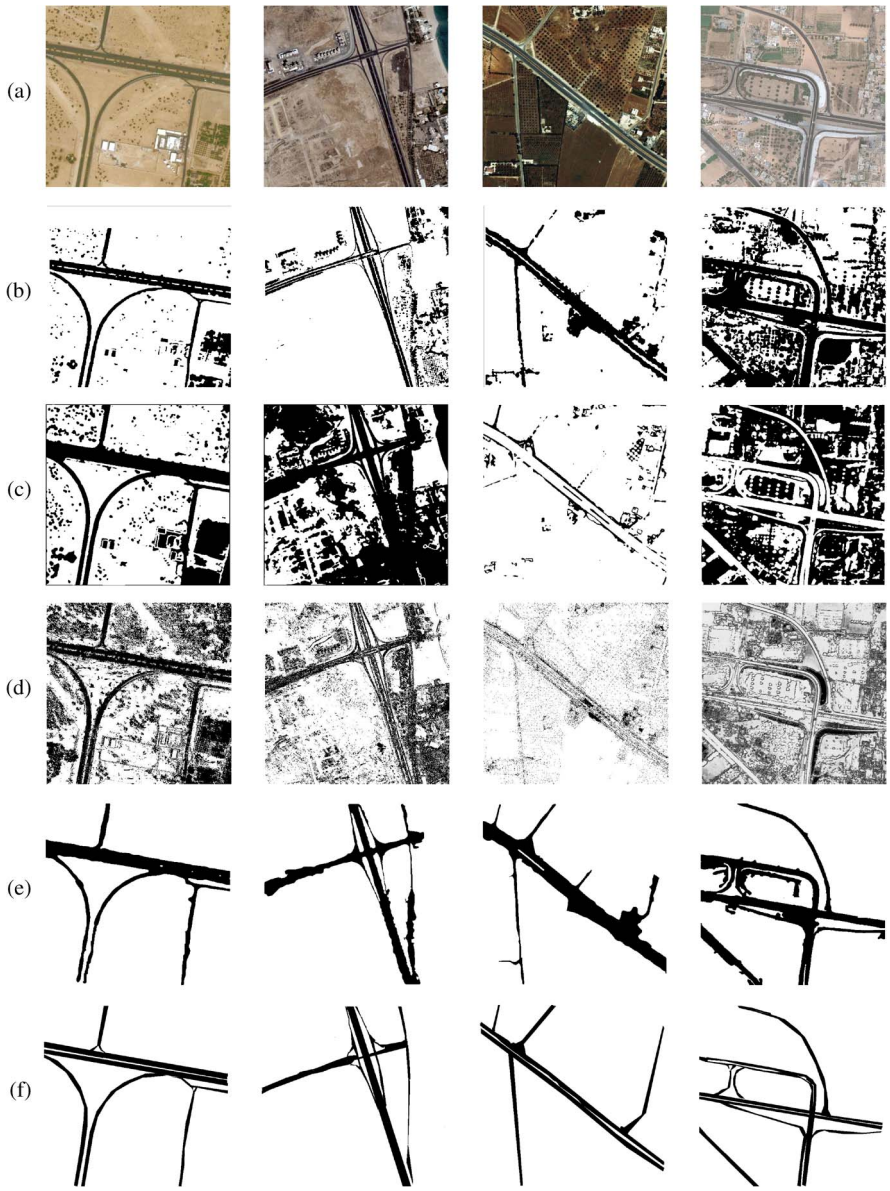


Fig. 13. (a) Four satellite images of size (512 × 512) from a suburban area of an emerging region. (b) Results from FeatureObjeX [51]. (c) Results of the method proposed in [26]. (d) Results of the method proposed in [49]. (e) Results of our proposed method. (f) Hand-drawn (manual) road map.

TABLE IV
PERFORMANCE OF THE SYSTEM FOR THE IMAGES SHOWN IN FIGS. 11–14. (A) COMPLETENESS. (B) CORRECTNESS

Road image type		FeatureObjeX [53]				Tuncer [26]				Mokh et al. [51]				Proposed			
		I	*II*	*III*	*IV*	*I*	*II*	*III*	*IV*	*I*	*II*	*III*	*IV*	*I*	*II*	*III*	*IV*
Developed suburban (Fig. 11)	A	100	91	100	100	100	88	98	97	100	63	98	69	**100**	**98**	**100**	**100**
	B	88	72	85	82	91	77	91	68	87	57	83	61	**100**	**96**	**100**	**99**
Developed urban (Fig. 12)	A	85	97	98	100	75	98	98	100	65	95	98	100	**98**	**98**	**99**	**100**
	B	79	78	76	100	83	80	86	91	52	98	73	81	**96**	**100**	**98**	**100**
Emerging suburban (Fig. 13)	A	100	90	**99**	100	100	89	68	91	100	94	54	81	**100**	**100**	96	**100**
	B	92	84	82	86	81	62	56	78	63	61	52	71	**100**	**100**	**100**	**95**
Emerging urban (Fig. 14)	A	91	87	83	**94**	62	52	74	91	71	64	59	71	**92**	**92**	**85**	91
	B	74	76	96	82	72	63	64	72	57	48	50	56	**83**	**100**	**98**	**83**

that our proposed method outperforms all the three methods. The values of these accuracy measures are obtained from Tables V–VIII. The performance of the proposed road detection framework is high for the suburban and urban scenes of developed countries. It is marginally lower for the same in emerging countries. The main reason for this is the fact that most emerging countries have unregulated growth plan. They employ nonuniform means of constructing roads with differing material constituents, based on demand and economic considerations. Main arterial roads have been detected properly over all four categories of suburban and urban scenes, whereas for secondary roads (lanes, bylanes, and links), the system occasionally fails to provide a high degree of accuracy. Compared with the main roads, the secondary roads are much more difficult to deal

TABLE V
PERFORMANCE OF THE SYSTEM AVERAGED OVER 25 IMAGES OF SUBURBAN SCENES OF DEVELOPED COUNTRIES

Methods	Completeness	Correctness
FeatureObjeX [53]	86%	79%
Tuncer [26]	83%	72%
Mokhtarzade et al. [51]	68%	60%
Proposed Method	**93%**	**97%**

TABLE VI
PERFORMANCE OF THE SYSTEM AVERAGED OVER 25 IMAGES OF URBAN SCENES OF DEVELOPED COUNTRIES

Methods	Completeness	Correctness
FeatureObjeX [53]	87%	74%
Tuncer [26]	85%	65%
Mokhtarzade et al. [51]	71%	61%
Proposed Method	**91%**	**94%**

TABLE VII
PERFORMANCE OF THE SYSTEM AVERAGED OVER 25 IMAGES OF SUBURBAN SCENES OF EMERGING COUNTRIES

Methods	Completeness	Correctness
FeatureObjeX [53]	82%	62%
Tuncer [26]	83%	64%
Mokhtarzade et al. [51]	68%	58%
Proposed Method	**87%**	**92%**

TABLE VIII
PERFORMANCE OF THE SYSTEM AVERAGED OVER 25 IMAGES OF URBAN SCENES OF EMERGING COUNTRIES

Methods	Completeness	Correctness
FeatureObjeX [53]	81%	60%
Tuncer [26]	73%	52%
Mokhtarzade et al. [51]	68%	52%
Proposed Method	**86%**	**90%**

with, because of the low discriminative power of the gray-level distributions of road regions with respect to background and greater effect of occlusions and noise on narrower roads. This is the main difficulty with secondary roads. The primary concern in urban areas is the misclassification between roads and parking lots. Road and surfaces of parking lots often use the same construction material and thus have similar spectral signatures, making it very difficult to automatically separate them in remotely sensed images. Roads and parking lots usually exhibit the same spectral and (occasionally) geometrical features, and both will be usually occupied by vehicles. In the test images, the sizes and shapes of roads, their proximity, environment, and contrast with respect to background all differ to a great extent. These test images are specifically selected to represent a wide and diverse area with varying road characteristics.

Fig. 15 shows the roads extracted from two large-size images obtained for the southern part of the city of Chennai, India, and a part of Toronto City in Canada. We see that most of the highways are detected with good accuracy. Fig. 15(a) shows the input images. A larger size image has been obtained by manually cropping adjacent areas from the screen and then stiching/cascading them as tiles, using an image registration [81] process. Fig. 15(b) shows the roadmap detected by the GIS tool FeatureObjeX [51], Fig. 15(c) shows the (hand-drawn) ground truth, and Fig. 15(d) shows the roadmap detected by our algorithm. Our algorithm fails to detect only the narrow roads and one large road structure having heterogenous spectral response in such complex situations due to the presence of strong background clutter.

Our proposed algorithm fails in discriminating road structures in highly complex and cluttered scenes. Fig. 16 shows the results for a few failure cases. Fig. 16(a) shows the input satellite images, with roads occluded by shadows, trees, and buildings. The outputs obtained by our proposed algorithm for the images shown in Fig. 16(a) are shown in Fig. 16(c). The manual hand-drawn road networks for the input images are shown in Fig. 16(b). Overall, the percentages of failure in terms of completeness and correctness measures are 54% and 35%, respectively, as observed for eight samples. These eight samples are not included in the set of test (or training) cases used for performance analysis consisting of 100 samples in each set. The results in these cases do not have a high accuracy due to the reasons specified earlier.

Table X shows the computational time required for the different stages of processing proposed for our approach as well as the other approaches used for comparative study. The time required by our proposed method is the mean of computational time, required by [26] and [49], which are used for comparative study. FeatureObjeX [51] needs the least time, as it is a simple supervised segmentation technique based on Bayes criteria, producing a poorer performance. Observing the average of the values of performance of FeatureObjeX in Tables V–VIII, the completeness measure is marginally better, while the correctness measure is significantly lower than that of the Stage I output. However, if computational time is a major constraint in operation, rather than performance, FeatureObjeX may be used in place of Stage I. If we replace Stage I of our proposed framework by FeaureObjeX [51], we reduce the computational time by 160 s, by compromising the accuracy (completeness by 5% and correctness by 19% approximately) to a large extent.

VII. CONCLUSION

A multistage strategy for automatically extracting roads from high-resolution multispectral satellite images based on salient features has been introduced and demonstrated. This new method incorporates the salient features of roads in satellite images using P-SVM and DSM in such a way that it preserves the strong discriminative ability of SVM while simultaneously exploiting the linear like characteristics in the features derived using DSM. For the determination of the discontinuity and elimination of nonroad parts, multistage postprocessing approaches were used. A segment linking module solves the problem of discontinuity to some extent, thereby increasing the completeness. RPS together with MAT-based hypothesis verification eliminates nonroad parts and increases the correctness. The results prove that the proposed system is able to effectively extract major sections of the road network, a few junctions, and curved roads from high-resolution multispectral satellite images. A method based on pipelining and thread-based parallelism will speed up the computation. MAT and many other postprocessing stages are of SIMD type, providing scope for the same.

TABLE IX
PERFORMANCE AT THE END OF THE DIFFERENT STAGES OF OUR PROPOSED APPROACH AVERAGED OVER $25 \times 4 (= 100)$ images

Stages	Methods	Completeness	Correctness
I	DSM	76%	71%
	SVM	74%	76%
	CSNN-CII	79%	80%
II	Post-processing	84%	82%
	Segment linking	88%	84%
	Region part segmentation	89%	88%
	Medial Axis Transform (final output of proposed method)	89%	93%

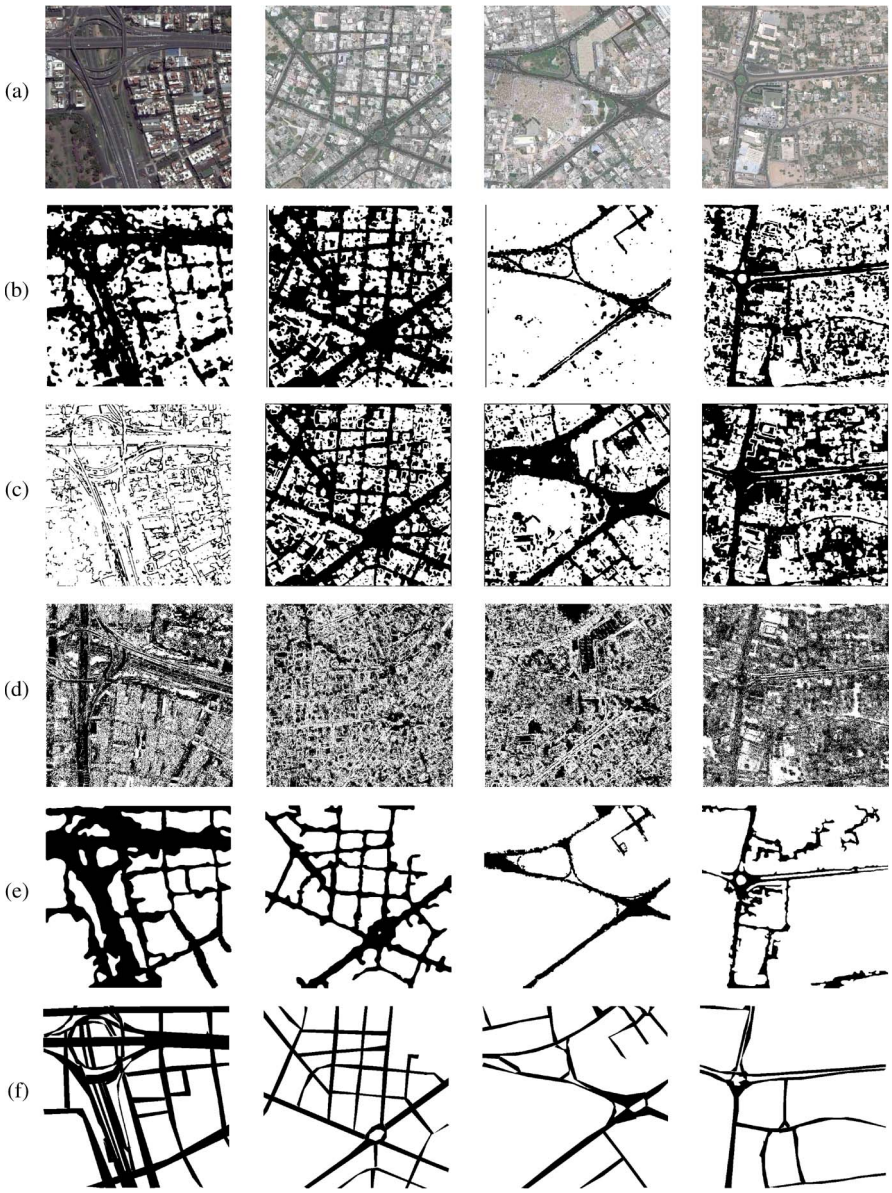


Fig. 14. (a) Four satellite images of size (512×512) from an urban area of an emerging region. (b) Results from FeatureObjeX [51]. (c) Results of the method proposed in [26]. (d) Results of the method proposed in [49]. (e) Results of our proposed method. (f) Hand-drawn (manual) road map.

We did not consider the use of SAR images as they are not easily available in public domain, free of cost. Most of the database samples have been acquired from Wikimapia [77], which does not provide any SAR data for use. Hence, we have restricted our use to multiband (visible) high-resolution optical images only. Designing a road detection algorithm for a wide range of sensors could be a tedious task, which could form a nice scope of work in the future.

It is observed that the road detection process produces a high degree of accuracy particularly for the scenes of developed countries. In urban areas, however, only major roads with larger pixel widths have been detected. Moreover, the presence of

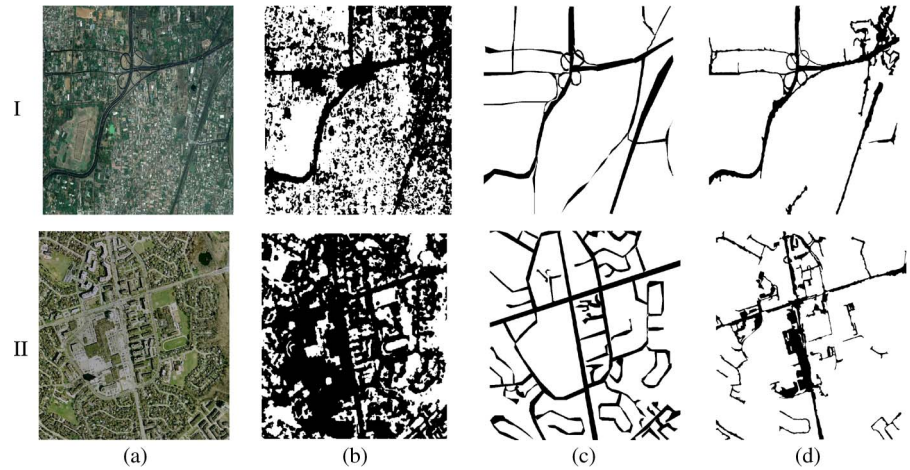


Fig. 15. (a) Input satellite images of (I) a southern part of Chennai City, India, and (II) a part of the city of Toronto, Canada, each of size 1000 × 1000. (b) Outputs using [51]. (c) Manual (hand-drawn) extraction. (d) Our proposed method.

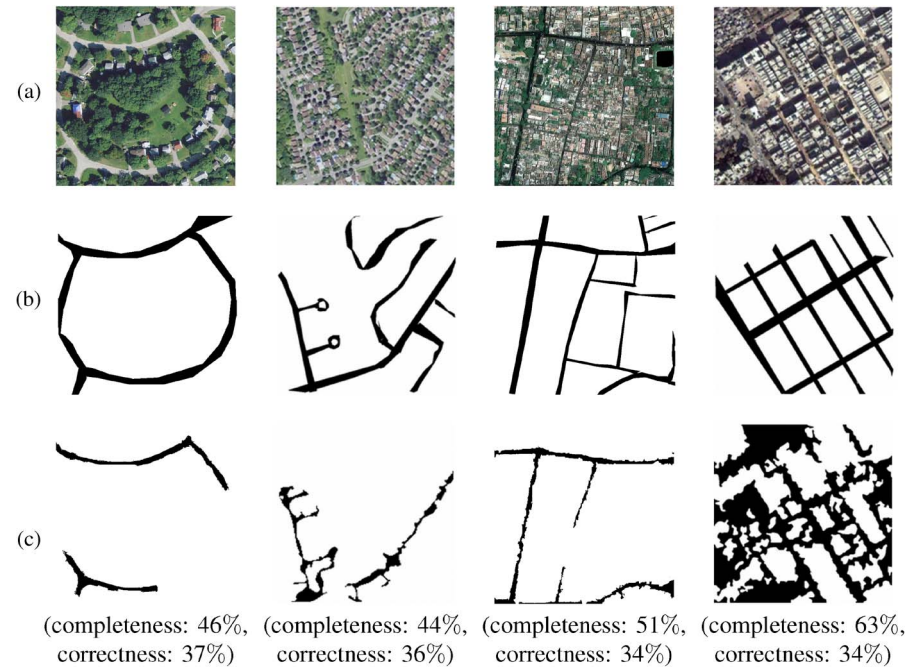


Fig. 16. Results of failure cases with satellite images of four different categories. (a) Input images. (b) Reference map (hand-drawn). (c) Results of our proposed method.

TABLE X
COMPARISON OF COMPUTATIONAL TIME (IN SECONDS) OF DIFFERENT METHODS IN A PENTIUM IV 3-GHZ 4-GB QUAD-CORE MACHINE

Techniques	DSM	PSVM	CSNN-CII	Post-processing	Segment linking	RPS	MAT	Proposed	FeatureObjeX [53]	Tuncer[26]	Mokhtarzade[51]
Time	40	180	240	5	20	45	12	**502**	**260**	**430**	**610**

buildings and other features similar to roads made the extraction process somewhat more difficult compared with the suburban scenes. Road junction detection and modeling of shadows are issues to be addressed in the future scope of work for this problem. Vectorization of the extracted road segments (using MAT) can also form a nice extension of this work for GIS updates and cartography.

APPENDIX A
SVM

The main idea of SVM is to separate the classes with a hyperplane surface so as to maximize the margin among them. In this paper, SVMs are used to classify roads from satellite imagery. In SVM, the input vectors are mapped nonlinearly to a very high dimensional feature space [66]. Considering a two-class pattern classification problem, let the training set of size N be $(X_i, d_i)_{i=1}^{N}$ where $X_i \in R^n$ is the input pattern for the ith example and $d_i \in [-1, +1]$ is the corresponding desired response. The classifier is represented by the function $\mathbf{f}(x; \alpha) \rightarrow \mathbf{y}$, with $\boldsymbol{\alpha}$ as the parameter of the classifier. The SVM method involves finding the optimum separating hyperplane so that the following are satisfied.

1) Samples with labels $\mathbf{y} = \pm 1$ are located on each side of the hyperplane.

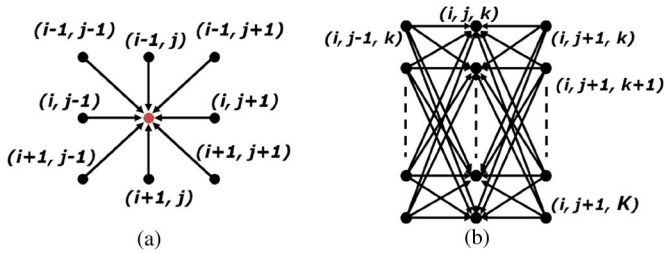


Fig. 17. (a) Top view of a neuron in the CSNN network, showing connection in a single layer. (b) Side view of the network, showing connections across layers.

2) The distances of the closest vectors to the hyperplane on each side are maximum. These are called support vectors, and the distance is the optimal margin.

The membership decision rule is based on the function $\mathbf{f_s}(x)$, which represents the discriminant function associated with the hyperplane in the transformed space, defined as

$$\mathbf{f_s}(x) = w^*.\phi(x) + w_0 \tag{11}$$

where w^* is the weight vector, w_0 is the bias, and $\phi(x) \in R^{d'}$ $(d' \gg d)$. SVM is used to classify every pixel into either road or nonroad groups based on the sign of the discriminant function $(\mathbf{y} = \mathrm{sgn}(\mathbf{f_s}(x)))$. Pixels belonging to roads are assigned as group 1 and the others to group 2 from training sample images. Since SVM has good generalization ability, this decision function can be applied to extract road structures from satellite images. Through training, we obtain the decision function. The feature vectors are fed into the SVM classifier initially for training (to learn the pattern) from known examples and then for predicting the labels of unknown samples once the training is complete.

APPENDIX B
CSNN ARCHITECTURE

CSNN is a feedback (recurrent) neural network which optimizes a set of constraints based on the design used to solve an optimization problem. As proposed in [82], CSNN consists of K layers of $I \times J$ neurons, each corresponding to a pixel, where K is the number of target segments (classes). Neurons in each layer hold the probabilities, corresponding to the pixels, for the segment (class) represented by the corresponding layer. As shown in Fig. 17, each neuron synapses to a set of 3-D neighboring neurons. The connectivities of neurons within the same layer and across different layers of neurons are shown in Fig. 17(a) and (b). The assignment of weights is done, based on a set of nonparametric (generally) constraints for the optimization problem being solved. Synaptic weights are set to update the probabilities toward convergence by inhibitory and excitatory actions [83].

The label probabilities of each neuron are updated after every iteration in a winner-take-all style [84]. The final segmented map is obtained by taking the label corresponding to the winner layer for each pixel in an image. The kth class probability at pixel (i, j), O^t_{ijk}, at iteration step t, is updated by a small amount δ if it is a winner class; otherwise, it is decremented by $-\delta$. Initial value of δ is 0.04. Weights in the CSNN can be interpreted as constraints. H_{ijk} represents the contribution from the neighborhood and participates in adjusting the activation level of the neuron, which is given as

$$H^t_{ijk} = \sum_{O_{qrl} \in N_{ij}} W_{ij,qr,k,l} O^t_{qrl} \tag{12}$$

where O_{ijk} represents the output of the (i, j)th neuron in the kth layer and N_{ij} shows the neighborhood of the (i, j)th neuron. The weight between the kth layer's (i, j)th neuron and lth layer's (q, r)th neuron is computed as

$$W_{ij,qr,k,l} = \frac{1}{P}\left(1 - \frac{2|k-l|}{K}\right) \tag{13}$$

with K and P representing the total number of layers (the number of segments) and the size of 2-D neighborhood centered at (i, j), respectively. Weights are determined based on the heuristic that a neuron excites those neighbors which represent similar labels of intensities/features and inhibits other neurons representing different labels of intensities/features [82]. Weights in the CSNN can be interpreted as constraints. Weights are determined based on the heuristic that a neuron excites neighboring neurons representing the labels of similar intensities and inhibits other neurons representing labels of different intensities [82]. The kth layer's (i, j)th neuron is updated at each stage of iteration during the process of relaxation, using a nonlinear updating rule, as follows:

$$O^{t+1}_{ijk} = \frac{\mathrm{Pos}\left(O^t_{ijk} + \Delta O^t_{ijk}\right)}{\sum_{l=1}^{K} \mathrm{Pos}\left(O^t_{ijl} + \Delta O^t_{ijl}\right)} \tag{14}$$

$$\Delta O^t_{ijk} = \begin{cases} \delta, & \text{if } H^t_{ijk} = \max_l \left(H^t_{ijl}\right) \\ -\delta, & \text{otherwise} \end{cases} \tag{15}$$

$$\mathrm{Pos}(X) = \begin{cases} X, & \text{if } X \geq slant0 \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

The algorithm iteratively shuttles between adding new and removing redundant edge pixels and hence inherently produces a correction mechanism to the process of fusion (for details, see [52]). The algorithm requires the probability values for all the image pixels corresponding to each class and edge maps at two different thresholds as input. The process of iteration converges when the local constraints (based on probabilities of class assignment of pixels and edge map) are satisfied. The final solution after convergence of CSNN-CII will yield the class assignment for each pixel (one of the classes—road or nonroad). This information will be available at the output of the CSNN neurons (similar to that done during initialization of the probability values for each of the classes, using P-SVM output). $O_{i,j,k}$ represents the probability for a pixel (i, j) belonging to one of the two classes: road $k = 1$ and nonroad $k = 2$ with the constraint $O_{(i,j,1)} + O_{(i,j,2)} = 1$. Based on the "maximum" value of probability between the two classes, a pixel is decided to be road or nonroad.

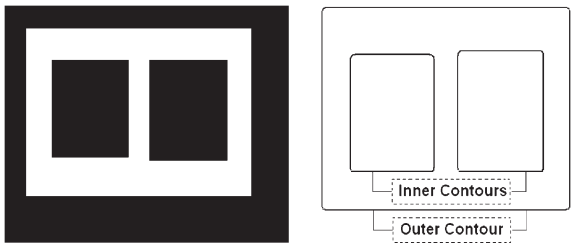


Fig. 18. (a) Input image. (b) Inner and outer contours.

APPENDIX C
STEPS IN RPS

A. Estimation of Curvature

A curve is represented in parametric form, where t is the path length and x and y are the coordinates of the contour

$$r(t) = (x(t), y(t)). \tag{17}$$

If there is more than one object, then the outer contour is traced for each object. If there is a child object inside an object, we have to then trace the outer contour for the child object as well. Inner boundary pixels are extracted by tracing the pixels at the inner contour in an object.

A smoothing of the contour with a Gaussian kernel is then needed prior to the computation of the curvature to overcome the problem of discontinuities in derivatives needed for curvature calculation [85]. The smoothed contour is represented

$$x_s(t) = x(t) * G, \quad y_s(t) = y(t) * G \tag{18}$$

where G is defined in (2). Fig. 18(a) shows an image having one object with two holes. The outermost pixels of the object are traced to extract the outer contour, and the boundary of the holes gives the inner contours as shown in Fig. 18(b). Curvature is defined as the rate of change of slope as a function of arc length t

$$K(t) = \frac{d\theta(t)}{dt} \tag{19}$$

where $\theta(t)$ is the tangent to the curve at "t." The curvature is computed as [72]

$$K_s(t) = \frac{\dot{x_s}\ddot{y_s} - \dot{y_s}\ddot{x_s}}{\left(\dot{x_s}^2 + \dot{y_s}^2\right)^{3/2}}. \tag{20}$$

The curvature obtained from (20) is smoothed with a Gaussian kernel [(2)] to obtain a smoothed curvature, as given by the following:

$$K_s(t) = K(t) * G. \tag{21}$$

B. Detection of Dominant Points

From the viewpoint of the human visual system [86], it has been suggested that the dominant points have high curvature or the rate of change of slope along the path length is high. In this paper, we detect these points and use them to decompose the object to remove the nonroad structures. Dominant points are points having a curvature value greater than a threshold. Local extrema are defined by the points at which the derivative of the curvature equals zero [85] as

$$K_s'(t) = \frac{dK_s(t)}{dt} = 0 \tag{22}$$

which is equivalent to convolving the curvature with the derivative of Gaussian and taking the zero crossings of this operation. Convex dominant points on the outer contour is combined with local extrema using AND operation to give the effective CDP_{cx}. Similarly, convex dominant points on the inner contour is combined with the local extrema to get effective CDP_{ce}.

These points are then used to segment the nonroad parts from the given image. The CDP_{cx} are moved inward along the direction of its normal, whereas CDP_{ce} are moved outward along the direction of the normal. For a particular contour, all the CDPs (both CDP_{cx} and CDP_{ce}) are allowed to move simultaneously, and a CDP freezes only when it touches another moving CDP in the same contour or a point in the contour itself, which is within a specified path length. The specified path length of the moving CDP dictates the maximum perimeter of the nonroad region for the purpose of elimination. All the frozen CDPs are traced back to their origins, and the corresponding CDPs or the CDP and the contour point are joined using a line segment. Unlike Bennamoun algorithm [72], there is no necessity to freeze all the CDPs, and we only move the CDPs for a particular number of iterations. Unfrozen CDPs are not taken into account for segmentation. Now, the regions fitted with the new line segments are isolated as separate components. Setting an area threshold, small noisy nonroad structures are eliminated.

Fig. 19 shows an illustration using a synthetic example to explain the processing in the "RPS" algorithm. The synthetic example shown in Fig. 19(a) may be visualized to be a short fragment of a road segment, having a nonlinear protrusion attached to the road segment. In this synthetic example, we need to keep the elongated (horizontal) structure and eliminate the curved section, detected erroneously as a part of the road. Fig. 19(e) shows the local extrema on the curvature plot of the object contour for the input image in Fig. 19(a). Convex dominant points (CDPs) on the contour are combined with local extrema using an AND operation to obtain the effective convex dominant points CDP_{cx}. The effective dominant points computed for the image in Fig. 19(a) are shown in Fig. 19(f). Fig. 19(g) shows the effective dominant points plotted on the object contour. Curvature normals are drawn from the pair of CDPs, as given in Fig. 19(h). If they meet, then the corresponding pair of CDPs are connected to yield a region boundary. Hence, the segmented map in Fig. 19(i) is obtained by joining the pair of CDPs. The small region (segment) thus formed [shown filled in red color, in Fig. 19(i)] is eliminated as a nonroad part of the given image (see step 11 in the RPS algorithm in Section V-C).

APPENDIX D
MULTISCALE GRADIENT (EDGE) ESTIMATION PROCESS

Edge candidates are detected for individual channels by the process as described in [60]. The gradients obtained from the

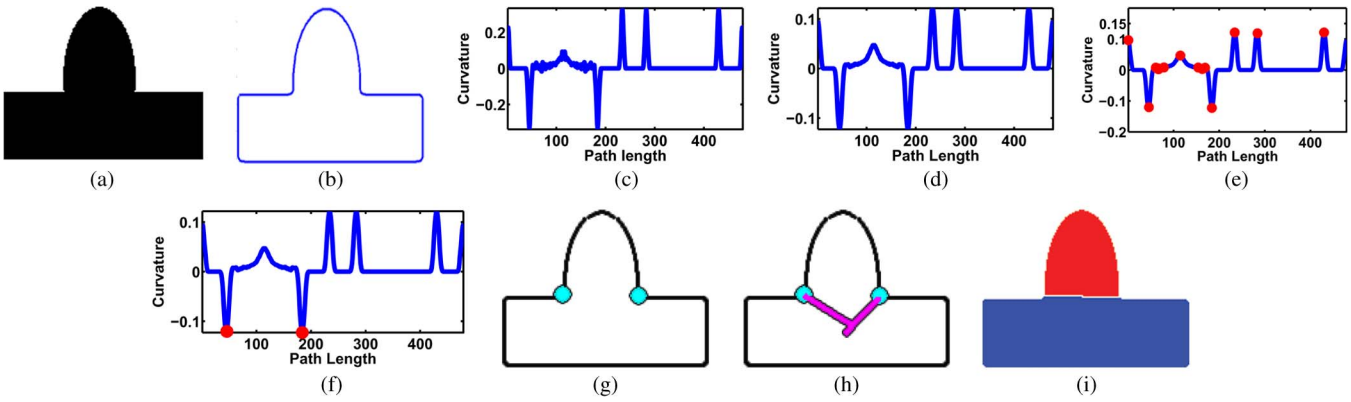


Fig. 19. (a) Input synthetic image. (b) Smoothed contour. (c) Curvature plot of (b). (d) Smoothed curvature from (c). (e) Extrema marked (using red dots) on the smoothed curvature plot in (d). (f) Effective CDPs (CDP_{cx}) extracted from the extrema in (e). (g) CDPs plotted on the object contour in (b). (h) Curvature normals drawn from the CDPs and connected. (i) Segmented map by joining the pair of CDPs.

individual channels [f is a vector in (1)] are used to obtain the strength of an edge along a particular direction. Edges can occur over a wide range of strengths as well as scales in real images. Selecting an optimal scale for all edges in an image is difficult. Combining the estimates from different scales, we get a final edge map using the method as described in [59].

The basic idea involves combining the estimates from different scales to get a final edge map. Edges can occur over a wide range of strengths as well as scales in real images. Selecting a single scale of smoothing which is optimal for all edges in an image is very difficult. One filter size may not be good enough to remove noise while keeping good localization. Multiscale edge detection offers an alternative. This involves applying smoothing operators of different sizes to an image, extracting the edges at each scale, and combining the recovered edge information to form a more complete picture of the actual edge representation of the image. Edge candidates are detected for individual channels, as described in [60]. Then, the Euclidean (vectored) sum of the gradients of the three channels produces the response to obtain the final edge map. A simple color edge detection method discussed in [87] is used in our work. To remove noisy fragments, we have performed the saliency test as described in [59]. Our edge detection scheme generates a final edge map by combining the detected salient edge candidates from different scale channels. Since smaller scale of the detection functional always gives better edge localization accuracy and only salient edges with high signal-to-noise ratio survive through the salience test, the final edge map always incorporates all the detected salient edge candidates from the smallest scale. The combination procedure can then continue to check if there are new salient edges in the detection results from larger scales. The Canny operator is efficient and produces better gradient vectors which are orthogonal to the dominant orientation of the image pattern.

A road is expected to be minimally five pixels wide. This is based on the resolution (in meters per pixel) of the image data available with us. We have adjusted the minimal scale (sigma) of the Canny [88] edge detector, based on this minimal expected width (as a heuristic, again obtained empirically) of roads. Most established local orientation estimation techniques are based on the analysis of the local gradient field of the image. However, the local gradients are very sensitive to noise, thus making the estimate of local orientation from these images unreliable. The Canny edge detector (standard in imaging literature) works well in the presence of moderate noise, but it does not process the image signal to filter (as in image restoration or enhancement operations) the same. Hence, it performs the best (automatically) in the absence of noise. We do not expect large levels of speckle noise which is present in SAR images. The Canny operator has the power to detect edges in small to moderate levels of noise, which is additive Gaussian in nature.

REFERENCES

[1] F. Wang and R. Newkirk, "A knowledge-based system for highway network extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 26, no. 5, pp. 525–531, Sep. 1988.

[2] F. Tupin, H. Maitre, J. F. Mangin, J. M. Nicolas, and E. Pechersky, "Detection of linear features in SAR images: Application to road network extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 2, pp. 434–453, Mar. 1998.

[3] B. K. Jeon, J. H. Jang, and K. S. Hong, "Road detection in spaceborne SAR images using a genetic algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 1, pp. 22–29, Jan. 2002.

[4] W. Shi and C. Zhu, "The line segment match method for extracting road network from high-resolution satellite images," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 2, pp. 511–514, Feb. 2002.

[5] F. Tupin, B. Houshmand, and M. Datcu, "Road detection in dense urban areas using SAR imagery and the usefulness of multiple views," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 11, pp. 2405–2414, Nov. 2002.

[6] L. Bentabet, S. Jodouin, D. Ziou, and J. Vaillancourt, "Road vectors update using SAR imagery: A snake-based method," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 8, pp. 1785–1803, Aug. 2003.

[7] M. Negri, P. Gamba, G. Lisini, and F. Tupin, "Junction-aware extraction and regularization of urban road networks in high-resolution SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2962–2971, Oct. 2006.

[8] J. Hu, A. Razdan, J. Femiani, M. Cui, and P. Wonka, "Road network extraction and intersection detection from aerial images by tracking road footprints," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4144–4157, Dec. 2007.

[9] D. Guo, A. Weeks, and H. Klee, "Robust approach for suburban road segmentation in high-resolution aerial images," *Int. J. Remote Sens.*, vol. 28, no. 2, pp. 307–318, 2007.

[10] J. Yang and R. S. Wang, "Classified road detection from satellite images based on perceptual organization," *Int. J. Remote Sens.*, vol. 28, no. 20, pp. 4653–4669, Oct. 2007.

[11] T. Kavzoglu, Y. E. Sen, and M. Cetin, "Mapping urban road infrastructure using remotely sensed images," *Int. J. Remote Sens.*, vol. 30, no. 7, pp. 1759–1769, Jan. 2009.

[12] C. Zhu, W. Shi, M. Pesaresi, L. Liu, X. Chen, and B. King, "The recognition of road network from high-resolution satellite remotely sensed data using image morphological characteristics," *Int. J. Remote Sens.*, vol. 26, no. 24, pp. 5493–5508, Dec. 2005.

[13] X. Hu and C. V. Tao, "A reliable and fast ribbon road detector using profile analysis and model based verification," *Int. J. Remote Sens.*, vol. 26, no. 5, pp. 887–902, Mar. 2005.

[14] X. Huang and L. Zhang, "Road centreline extraction from high-resolution imagery based on multiscale structural features and support vector machines," *Int. J. Remote Sens.*, vol. 30, no. 8, pp. 1977–1987, Apr. 2009.

[15] H. Long and Z. Zhao, "Urban road extraction from high-resolution optical satellite images," *Int. J. Remote Sens.*, vol. 26, no. 22, pp. 4907–4921, Jan. 2005.

[16] J. Trinder and Y. Wang, "Knowledge-based road interpretation in aerial images," *Int. Arch. Photogramm. Remote Sens.*, vol. 32, no. 4, pp. 635–640, 1998.

[17] A. Katartzis, H. Sahli, V. Pizurica, and J. Cornelis, "A model-based approach to the automatic extraction of linear features from airborne images," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 9, pp. 2073–2079, Sep. 2001.

[18] A. Baumgartner, C. Steger, C. Wiedemann, H. Mayer, W. Eckstein, and H. Ebner, "Update of roads in GIS from aerial imagery: Verification and multi-resolution extraction," in *Proc. Int. Arch. Photogramm. Remote Sens.*, 1996, vol. XXXI, pp. 53–58.

[19] H. Mayer, I. Laptev, A. Baumgartner, and C. Steger, "Automatic road extraction based on multi-scale modelling, context and snakes," in *Proc. Int. Arch. Photogramm. Remote Sens.*, 1997, pp. 106–113.

[20] H. Mayer, I. Laptev, and A. Baumgartner, "Multi-scale and snakes for automatic road extraction," in *Proc. 5th Eur. Conf. Comput. Vis.*, 1998, pp. 720–733.

[21] 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.

[22] A. Baumgartner, S. Hinz, and C. Wiedemann, "Efficient methods and interfaces for road tracking," in *Proc. Int. Soc. Photogramm. Remote Sens.*, 2002, pp. 28–31.

[23] N. Yager and A. Sowmya, "Support vector machines for road extraction from remotely sensed images," in *Proc. Comput. Anal. Images Patterns*, vol. 2756, *LNCS*, 2003, pp. 285–292.

[24] M. Song and D. Civco, "Road extraction using SVM and image segmentation," *Photogramm. Eng. Remote Sens.*, vol. 70, no. 12, pp. 1365–1371, Dec. 2004.

[25] Q. Zhang and I. Couloigner, "Benefit of the angular texture signature for the separation of parking lots and roads on high resolution multispectral imagery," *Pattern Recog. Lett.*, vol. 27, no. 9, pp. 937–946, Jul. 2006.

[26] O. Tuncer, "Fully automatic road network extraction from satellite images," in *Proc. Recent Adv. Space Technol.*, Jun. 2007, pp. 708–714.

[27] C. Steger, "An unbiased detector of curvilinear structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 2, pp. 113–125, Feb. 1998.

[28] M. Amo, F. Martinez, and M. Torre, "Road extraction from aerial images using a region competition algorithm," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1192–1201, May 2006.

[29] D. Geman and B. Jedynak, "An active testing model for tracking roads in satellite images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 1, pp. 1–14, Jan. 1996.

[30] P. Gamba, F. Dell'Acqua, and G. Lisini, "Improving urban road extraction in high-resolution images exploiting directional filtering, perceptual grouping, and simple topological concepts," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 3, pp. 387–391, Jul. 2006.

[31] M. Barzohar and D. B. Cooper, "Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 7, pp. 707–721, Jul. 1996.

[32] J. B. Mena, "State of the art on automatic road extraction for GIS update: A novel classification," *Pattern Recognit. Lett.*, vol. 24, no. 16, pp. 3037–3058, Dec. 2003.

[33] M. F. A. Fortier, D. Ziou, C. Armenakis, and S. Wang, "Survey of work on road extraction in aerial and satellite images," Univ. Sherbrooke, Sherbrooke, QC, Canada, Tech. Rep. 241, pp. 3037–3058, 2003, vol. 24, no. 16.

[34] J. Shen, X. Lin, Y. Shi, and C. Wong, "Knowledge-based road extraction from high resolution remotely sensed imagery," in *Proc. Congr. Image Signal Process.*, 2008, vol. 4, pp. 608–612.

[35] C. Zhang, S. Murai, and E. P. Baltsavias, "Road network detection by mathematical morphology," in *Proc. ISPRS Workshop 3D Geospatial Data Prod.*, 1999, pp. 185–200.

[36] R. Marikhu, M. Dailey, S. Makhanov, and K. Honda, "A family of quadratic snakes for road extraction," in *Proc. Asian Conf. Comput. Vis.*, 2007, pp. 85–94.

[37] P. Doucette, P. Agouris, A. Stefanidis, and M. Musavi, "Self-organized clustering for road extraction in classified imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 55, no. 5/6, pp. 347–358, Mar. 2001.

[38] P. Mantero, G. Moser, and S. B. Serpico, "Partially supervised classification of remote sensing images through SVM-based probability density estimation," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 559–570, Mar. 2005.

[39] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley-Interscience, 2000.

[40] G. Miliaresisa and N. Kokkasb, "Segmentation and object-based classification for the extraction of the building class from LIDAR DEMs," *Comput. Geosci.*, vol. 33, no. 8, pp. 1076–1087, Aug. 2007.

[41] W. Wei and Y. Xin, "Feature extraction for man-made objects segmentation in aerial images," *Mach. Vis. Appl.*, vol. 19, no. 1, pp. 57–64, Jan. 2008.

[42] D. Mumford and J. Shah, "Optimal approximation by piece wise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, Jul. 1989.

[43] J. Boggess, "Identification of roads in satellite imagery using artificial neural networks," Mississippi State University, Mississippi State, MS, Tech. Rep. 930815, 1993.

[44] K. Tu-Ko, "A hybrid road identification system using image processing techniques and backpropagation neural network," Mississippi State Univ., Mississippi State, MS, Tech. Report MS 39 762, 2003.

[45] X. Hu and V. Tao, "Automatic extraction of main road centerlines from high resolution satellite imagery using hierarchial grouping," *Photogramm. Eng. Remote Sens.*, vol. 73, no. 9, pp. 1049–1056, Sep. 2007.

[46] S. Movaghati and A. Moghaddamjoo, "Road extraction from satellite images using particle filtering and extended Kalman filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 7, pp. 2807–2817, Jul. 2010.

[47] Y. Wang, Y. Tian, X. Tai, and L. Shu, "Extraction of main urban roads from high resolution satellite images by machine learning," in *Proc. Asian Conf. Comput. Vis.*, 2006, pp. 236–245.

[48] M. Mokhtarzade, M. Zoej, and H. Ebadi, "Automatic road extraction from high resolution satellite images using neural networks, texture analysis, fuzzy clustering and genetic algorithms," *Int. Soc. Photogramm. Remote Sens.*, vol. 37, no. Pt. B3b, pp. 549–556, 2008.

[49] M. Mokhtarzade and M. J. V. Zoej, "Road detection from high-resolution satellite images using artificial neural networks," *Int. J. Appl. Earth Observ. Geoinform.*, vol. 9, no. 1, pp. 32–40, Feb. 2007.

[50] X. Jin and C. H. Davis, "An integrated system for automatic road mapping from high-resolution multi-spectral satellite imagery by information fusion," *Inform. Fusion*, vol. 6, no. 4, pp. 257–273, Dec. 2005.

[51] Geomatics, FeatureObjeX. [Online]. Available: www.pcigeomatics.com/geomatica

[52] G. Lalit, U. G. Mangai, and S. Das, "Integrating region and edge information for texture segmentation using a modified constraint satisfaction neural network," *Image Vis. Comput.*, vol. 26, no. 8, pp. 1106–1117, Aug. 2008.

[53] C. Heipke, H. Mayer, C. Wiedemann, and O. Jamet, "Evaluation of automatic road extraction," in *Proc. Int. Arch. Photogramm. Remote Sens.*, 1997, pp. 47–56.

[54] G. Cooper and D. Cowan, "Enhancing linear features in image data using horizontal orthogonal gradient ratios," *Computers and Geosciences*, vol. 33, no. 7, pp. 981–984, Jul. 2007.

[55] G. Granlund and H. Knutson, *Signal Processing for Computer Vision*. Norwell, MA: Kluwer, 1995.

[56] E. P. Lyvers and O. R. Mitchell, "Precision edge contrast and orientation estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 6, pp. 927–937, Nov. 1988.

[57] B. Majidi and A. Bab-Hadiashar, "Aerial tracking of elongated objects in rural environments," *Mach. Vis. Appl.*, vol. 20, no. 1, pp. 23–34, Jan. 2009.

[58] M. Basu, "Gaussian-based edge-detection methods—A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 3, pp. 252–260, Aug. 2002.

[59] R. J. Qian and T. S. Huang, "Optimal edge detection in two-dimensional images," *IEEE Trans. Image Process.*, vol. 5, no. 7, pp. 1215–1220, Jul. 1996.

[60] P. Kumar, S. Das, and B. Yegnanarayana, "One-dimensional processing of images," in *Proc. Int. Conf. Multimedia Process. Syst., IIT*, Madras, India, Dec. 2000, pp. 451–454.

[61] S. Ramesh, S. Das, and B. Yegnanarayana, "Eigenedginess vs. eigenhill, eigenface and eigenedge," in *Proc. XI Eur. Signal Process. Conf.*, Toulouse, France, 2002, pp. 559–562.

[62] T. Mirnalinee, S. Das, and K. Varghese, "Integration of region and edge-based information for efficient road extraction from high resolution satellite imagery," in *Proc. IEEE ICAPR*, Kolkata, India, 2009, pp. 373–376.
[63] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1998.
[64] G. Strang, *Linear Algebra and Its Application*. Belmont, CA: Thomson Brooks, 2005.
[65] A. Gruen and H. Li, "Road extraction from aerial and satellite images by dynamic programming," *ISPRS J. Photogramm. Remote Sens.*, vol. 50, no. 4, pp. 11–20, Aug. 1995.
[66] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.
[67] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. Cambridge, MA: MIT Press, 1999, pp. 61–74.
[68] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Reading, MA: Addison-Wesley, 1992.
[69] R. Mukundan and K. R. Ramakrishnan, *Moment Functions in Image Analysis Theory and Applications*. Singapore: World Scientific, 1998.
[70] N. B. Rizvandi, A. Pizurica, W. Philips, and D. Ochoa, "Edge linking based method to detect and separate individual elegans worms in culture," in *Proc. DICTA*, 2008, pp. 65–70.
[71] P. Soille, *Morphological Image Analysis: Principles and Applications*. New York: Springer-Verlag, 1998.
[72] M. Bennamoun and G. Mamic, *Object Recognition Fundamentals and Case Studies*. New York: Springer, 2002.
[73] A. Baumgartner, C. Steger, H. Mayer, W. Eckstein, and H. Ebner, "Automatic road extraction based on multi-scale, grouping, and context," *Photogramm. Eng. Remote Sens.*, vol. 65, no. 7, pp. 777–785, Jul. 1999.
[74] P. Soille and M. Pesaresi, "Advances in mathematical morphology applied to geoscience and remote sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 9, pp. 2042–2055, Sep. 2002.
[75] M. P. Martínez-Pérez, J. Jiménez, and J. L. Navalón, "A thinning algorithm based on contours," *Comput. Vis. Graph. Image Process.*, vol. 39, no. 2, pp. 186–201, Aug. 1987.
[76] H. Breu, J. Gil, D. Kirkpatrick, and M. Werman, "Linear time euclidean distance transform algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 5, pp. 529–533, May 1995.
[77] Y. Xia, "Skeletonization via the realization of the fire front's propagation and extinction in digital binary shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 10, pp. 1076–1086, Oct. 1989.
[78] F. Leymarie and M. D. Levine, "Simulating the grassfire transform using an active contour model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 1, pp. 56–75, Jan. 1992.
[79] A. Koriakine and E. Saveliev, Data, 2006. [Online]. Available: http://www.wikimapia.org/
[80] VPLab, Downloads, 2011. [Online]. Available: http://www.cse.iitm.ac.in/~sdas/vplab/satellite.html
[81] B. Zitova and J. Flusser, "Image registration methods: A survey," *Image Vis. Comput.*, vol. 21, no. 11, pp. 977–1000, Oct. 2003.
[82] W. C. Lin, E. C. Kuo, and C. T. Chen, "Constraint satisfaction neural networks for image segmentation," *Pattern Recognit.*, vol. 25, no. 7, pp. 679–693, Jul. 1992.
[83] F. Kurugollu and B. Sankur, "Map segmentation of color images using constraint satisfaction neural network," in *Proc. Int. Conf. Image Process.*, 1999, pp. 236–239.
[84] X. Munoz, J. Freixenet, X. Cufi, and J. Marti, "Strategies for image segmentation combining region and boundary information," *Pattern Recognit. Lett.*, vol. 24, no. 1–3, pp. 375–392, Jan. 2003.
[85] S. Pei and C. Lin, "The detection of dominant points on digital curves by scale space filtering," *Pattern Recognit.*, vol. 25, no. 11, pp. 1307–1314, Nov. 1992.
[86] M. Bennamoun, "A contour-based part segmentation algorithm," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 1994, pp. 41–44.
[87] A. Koschan, "A comparative study on color edge detection," in *Proc. ACCV*, 1995, pp. 574–578.
[88] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1996.

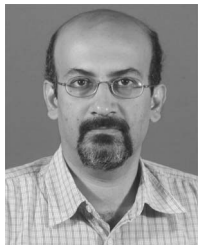
Sukhendu Das received the B.Tech. degree from the Department of Electrical Engineering, Indian Institute of Technology (IIT) Kharagpur, Kharagpur, India, in 1985, the M.Tech. degree in computer technology from IIT Delhi, New Delhi, India, in 1987, and the Ph.D. degree from IIT Kharagpur in 1993.

He has been a member of faculty with the Department of Computer Science and Engineering, IIT Madras, Chennai, India, since 1989, where he has been a Professor since June 2010. He has guided four (currently guiding two) Ph.D. students and several M.S. (currently guiding four), M.Tech. and B.Tech. students. He had completed several international and national sponsored projects and consultancies, both as Principal Investigator and Coinvestigator. He is currently involved in three sponsored projects in IIT Madras. He has published close to 100 technical papers in international and national journals and conferences. His current areas of research interest are visual perception, computational intelligence, computer vision, digital image processing and pattern recognition, computer graphics, computational science and engineering, and soft computing.

Dr. Das has received two best paper awards and a best design contest award.

T. T. Mirnalinee received the B.E. degree in computer engineering from Bharathidasan University, Trichy, India, the M.E. degree in software engineering from the College of Engineering, Guindy, Anna University, Chennai, India, and the Ph.D. degree from the Indian Institute of Technology (IIT) Madras, Chennai, India, in 2010.

She has about 16 years of teaching and research experience. She is currently a faculty with the Department of Computer Science and Engineering, Anna University of Technology, Chennai, India.



Koshy Varghese received the Ph.D. degree from the University of Texas, Austin, in 1992.

He is currently a Professor with the Department of Civil Engineering, Indian Institute of Technology (IIT) Madras, Chennai, India. He was a Postdoctoral Fellow with the Construction Industry Institute, U.S., for a year, before he joined IIT Madras. He works on basic and applied research in the area of computer-integrated construction project delivery.