

# Deep Learning Approach using Patch-based Deep Belief Network for Road Extraction from Remote Sensing Imagery

Md. Abdul Alim Sheikh, *Member, IAENG*, Tanmoy Maity and Alok Kole

**Abstract**—In this paper, an automated Deep Learning Architecture (DLA) called patch-based Deep Belief Neural Networks (P-DBN) is designed, implemented and experimentally evaluated for extracting semantic maps of roads in Remote Sensing (RS) images. Representative features are extracted by unsupervised pre-training of DBN and supervised fine-tuning phase. A Logistic Regression (LR) is added to the end of feature learning system to constitute a P-DBN-LR architecture. This LR classifier is employed to fine-tune the whole pre-trained network in a supervised way and classifies the patches from RS images. The features extracted from the image patches are fed to the architecture as input and it produces the class labels as a probability matrix as either a positive sample (road) or a negative sample (non-road). A math morphology algorithm is used to improve P-DBN performance during post processing. Experiments are conducted on a dataset of 970 RS scene images of urban and suburban areas to demonstrate the performance of the proposed network architecture. The proposed deep model resulted in an Overall Accuracy (OA) of 95.57% and F1-score of 0.9588. When compared to other state-of-the-art deep learning-based models such as U-Net, Cascaded CNN, Roadtracer, FCN and Salient features-SVM, our proposed P-DBN-LR model outperforms by 3.22% (0.9550 vs. 0.9242), 4.56% (0.9550 vs. 0.9114), 3.30% (0.9550 vs. 0.9233), 5.54% (0.9550 vs. 0.9020) and 7.40% (0.9550 vs. 0.8843), for the dataset. Experimental results demonstrate the effective performance of the proposed method for extracting roads from a complex scene.

**Index Terms**— Road Network Extraction; Remote Sensing Imagery; Deep Learning; Deep Belief Network; Restricted Boltzmann Machine

## I. INTRODUCTION

WITH the development of different high-speed imaging sensors, RS imagery is becoming increasingly available nowadays [1,2]. Extraction of roads from RS images plays important roles for supporting several government activities and various Geographical Information System (GIS) applications such as map generation and update, urban planning, traffic management, automated vehicle navigation and guidance, emergency response, change detection, disaster management etc. [3, 4].

In recent years, road extraction has also been extensively studied for terrain classification and ground vehicle

navigation [5]. A stark increase in the amount of RS imagery available in recent years has made the interpretation and identification of terrestrial objects e.g., road in remote sensing images a challenging problem at scale. The traditional process of manually updating a road database is very tedious and time-consuming [3]. For this reason, automatic extraction of roads in RS imagery has involved a lot of attention in the photogrammetry and remote sensing field [1, 3].

Over the past few years, many different techniques have been proposed to deal with the road network extraction from RS imagery. Some comprehensive reviews on road objects extraction from RS imagery can be found in [6-8]. Most of the road extraction techniques found in the literature can be clustered in two groups: classical methods and DL methods [4,6]. DL uses much high-level and multi-scale information where classical methods use low-level features for road extraction [4,9]. Road extraction from RS imagery is actually a classification problem including supervised and unsupervised classification methods [8].

Based on recent advances, Deep Learning (DL) is proving to be a very successful set of tools for several image understanding tasks, segmentation, classification tasks and other applications including remote sensing (RS) image analysis [9,10]. DL methods have been used for road extraction tasks as well due to its superiority in modelling non-linear relationships among variables [11]. For example, Mnih and Hinton applied DL to detect roads in high-resolution aerial images [25]. Recently, in [4,11,12], DL techniques are used for road extraction task from RS images with promising results. The Convolutional Neural Network (CNN) [13] and the Fully Convolutional Neural (FCN) [14] network are widely used for road extraction purposes. Various CNN methods for road extraction have been reported in the literature in [6] [15-19]. Over the past few years, many FCN-based variants [4,9,12,20,14,21,22] have been proposed to achieve more accurate segmentation results. Though many DL techniques have been developed for image segmentation and road extraction, they have not fully exploited the spatial contexts of road images and may fail to extract in depth features. They are also harder to train, learning can be very slow with multiple hidden layers and overfitting can also be a serious problem for both generative models and discriminative models [23].

In this paper, we aim to tackle the aforementioned problems by designing a P-DBN-LR model which can be used to extract the in-depth features by efficient layer-by-layer learning strategy of the original image for extracting semantic maps of roads in RS images. Successful spatio-temporal mapping features can be extracted by using the proposed DBN-LR architecture, to improve the accuracy of

Manuscript received Jan 03, 2022; revised July 15, 2022.

Md. Abdul Alim Sheikh is an Assistant Professor of Electronics and Communication Engineering Department, Aliah University, Kolkata, India (phone: 033-23416570; e-mail: alim.sheikh16@gmail.com).

Dr. Tanmoy Maity is an Associate Professor of Mining Machinery Engineering Department, Indian Institute of Technology (Indian School of Mines), India (email: tanmoy@iitism.ac.in.).

Dr. Alok Kole is a Professor of Electrical Engineering Department, RCCIT, Kolkata, India (email: alok.kole@rccit.org.in).

the classification. Representative features are extracted by unsupervised pre-training of DBN using CD learning algorithm with 1 step of Gibbs sampling (CD-1) [23] and supervised fine-tuning phase with back-propagation algorithm.

A LR [24] is added to the end of feature learning system to constitute a P-DBN-LR architecture. This LR classifier is employed to fine-tune the whole pre-trained network in a supervised way and classifies the patches from RS images. The main advantage of training the model using DBN because of its unsupervised feature learning in pre-training which makes it prominent from other methods. The goodness of the proposed method on its computational simplicity and high accuracy. The proposed P-DBN technique does not required much hardware resources which is better than other deep learning methods. The contributions of this paper are as follows:

1. The major contribution of this work is the road extraction model based on the construction of patch-based deep network architecture called P-DBN-LR and the introduction of the RBM as the feature extractor.
2. Proposed method can learn based on patch instead of pixel-by-pixel learning, which can significantly save the memory and computational time.
3. At the same time, Logistic unit is designed in the expansive part of the model, as feature classifier.
4. Compare different neural network structures specified in existing literature.

The paper is organized as follows. Section I covers the introduction and briefly reviews the related works. Section II provides an overview of our approach with the DBN architecture stacked by RBM for roads extraction from RS imagery; experiments and discussions are reported in Section III. Section IV discusses the limitations of the proposed model. Finally, conclusion and future direction is covered in Section V.

## II. PROPOSED METHOD

The overall methodology of the proposed P-DBN-SR model for road extraction from RS imagery is illustrated in Fig.1. A patch-based deep learning model is proposed to extract road network from RS imagery. The network structure of the proposed scheme is shown in Fig. 5.

### A. Data Pre-processing

A pre-processing pipeline is adopted to cope with input images of varying quality, resolution, and channels to remove noises and undesirable objects. The input images are filtered with the nonlinear bilateral filter [5] to minimize false alarms. Bilateral filter performs nonlinear smoothing on images by keeping the edge information. Fig. 2 shows the input image of urban area and pre-processing result after bilateral filtering on the sample image. Next, the RGB colored images are converted from RGB color space to the grayscale as shown in Fig. 3(b).

As a pre-processing step, a rough threshold is applied to the dataset. The threshold is applied after conversion of RGB to Grayscale by manually analyzing the frequency values of both road regions and background regions. This is useful to differentiate the road regions from others before patch generation. The reason for applying threshold is that roads

have higher intensities than other objects in the images but are also similar to other man- made structure, such as buildings. In this way, the proposed model learns to differentiate between road positive samples from other man-made structure e.g., buildings as negative samples. A Thresholded image generated from Fig.3(b) is shown in Fig. 3(c).



Fig. 2. Input Image of urban area and Result of bilateral Filtering

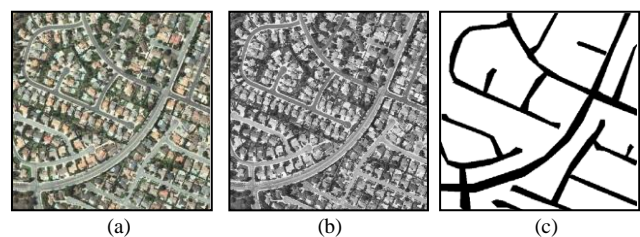


Fig. 3. (a) Input Image of urban area (b) Gray Image (c) Thresholded image

### B. Patch-based Training Data

Our proposed method can learn based on patches, which can significantly save the memory and computational time. The original image of size 512×512 are spilt into non-overlapping patches of size 32×32 which is converted into a one-dimensional vector of size 1024 as shown in Figure 3. These are applied as an input into the P-DBN model. All the patches having the same size with two class either road region (positive samples) or non-road region (negative sample). A patch is considered to be road patch if 75 percent pixels of the patch related to road region. The dataset became a binary class dataset as shown in Fig. 4. The whole dataset is divided into two class of road and non-road regions.

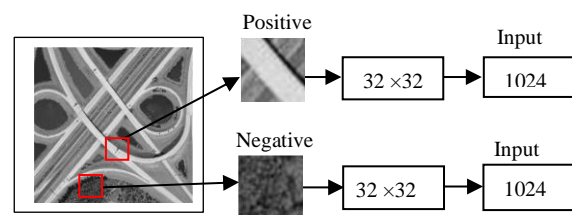


Fig. 4. Positive sample is a part of the road regions where more pixels belong to the Road and negative sample is the non-road regions which is not a part of road

### C. Patch-based Deep Belief Neural Network Model

DBN is composed by stacked of RBMs one by one each of which has its own visible layer and output layer [24]. The input image is divided into 32 ×32 patches and applied as input to the DBN. The proposed P-DBN is used to extract the in-depth features by efficient layer-by-layer learning strategy of the original image for road network extraction.

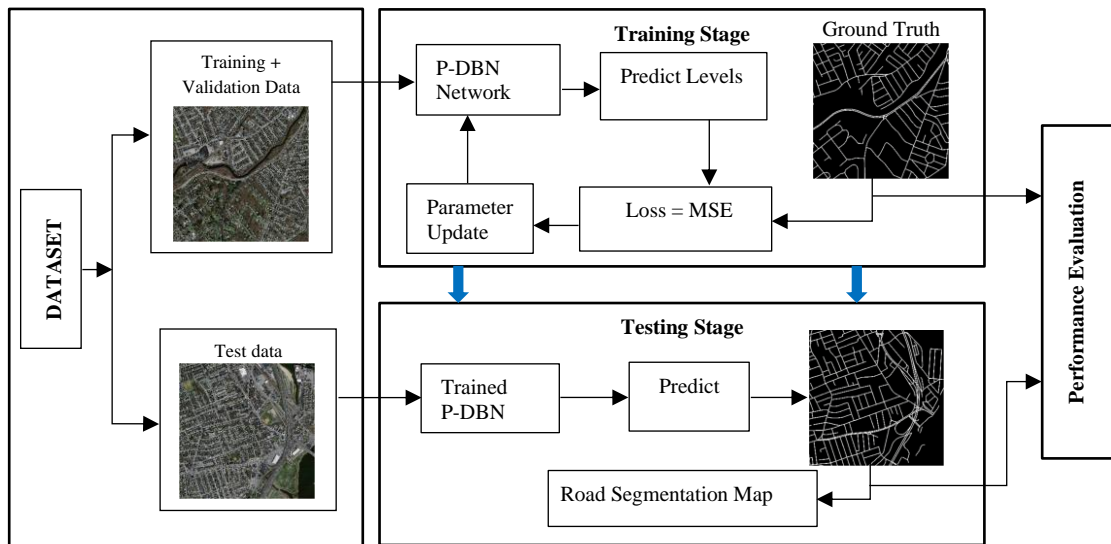


Fig.1. Overall Framework of the proposed road network extraction technique

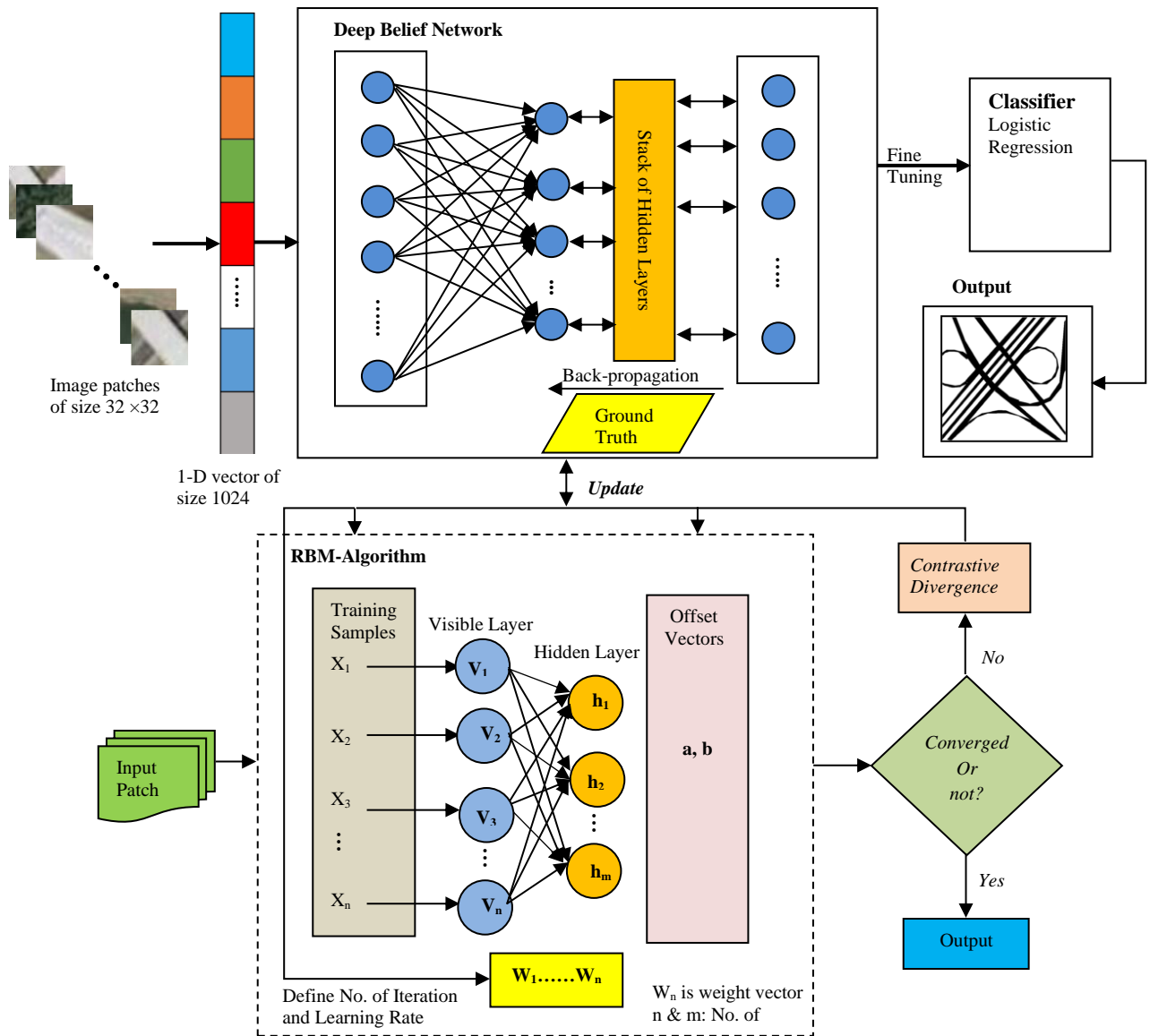


Fig. 5. Proposed Scheme for Road Extraction

An overview of RBM is discussed first in the following section followed by training and testing model of P-DBN-LR.

1) Overview of Restricted Boltzmann Machine

The topology of RBM is a two-layer stochastic graph which consists of one visible and one hidden layer [16]. The input is fed to the visible layer represented by  $\mathbf{v}$ , and the hidden layer represented by  $\mathbf{h}$  is used to reconstruct the input as close to as possible. The visible layer is connected to the hidden layer, while there is no connection between the neurons within the same layer. An illustration of an individual RBM is given in Fig. 6. Given a training set  $S=\{\mathbf{v}^1, \mathbf{v}^2, \mathbf{v}^3, \dots, \mathbf{v}^N\}$  that contains  $N$  samples, where  $\mathbf{v}^r = [v_1^r, v_2^r, v_3^r, \dots, v_n^r], r = 1, 2, \dots, N$  is the  $r$ -th training sample. Then, RBM can be considered as an energy model [7]. Given a set of network states  $(\mathbf{v}, \mathbf{h})$ , the energy of certain joint configuration of the two layers is given by

$$E_{\theta}(\mathbf{v}, \mathbf{h}) = - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m v_i h_j w_{ij} \quad (1)$$

Where  $\theta$  denotes the parameters (i.e.,  $\mathbf{W}, \mathbf{a}, \mathbf{b}$ );  $\mathbf{a} = [a_1, a_2, \dots, a_n]$  is the bias of visible layer,  $\mathbf{b} = [b_1, b_2, \dots, b_m]$  is the hidden layer bias, and  $\mathbf{W} = [w_{ij}]$  denotes the weights between visible unit  $i$  and hidden unit  $j$  respectively.

Then, the joint probability distribution of a set of visible and hidden states can be obtained by

$$p_{\theta}(\mathbf{v}, \mathbf{h}) = \frac{1}{Z_{\theta}} e^{-E_{\theta}(\mathbf{v}, \mathbf{h})} \quad (2)$$

Where  $Z_{\theta} = \sum_{\mathbf{v}, \mathbf{h}} e^{-E_{\theta}(\mathbf{v}, \mathbf{h})}$  is the partition function. Accordingly, the probability of the visible vector being assigned as  $\mathbf{v}$  can be obtained by summing over all possible hidden vectors as follows:

$$p(\mathbf{v}) = \frac{1}{Z_{\theta}} \sum_{\mathbf{h}} e^{-E_{\theta}(\mathbf{v}, \mathbf{h})} \quad (3)$$

Equations (1)-(3) form a statistical mechanics model and can be used in the training of RBM.

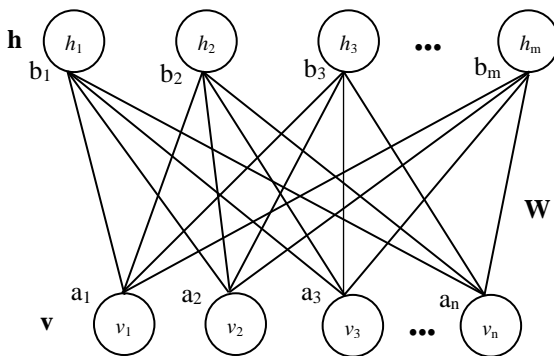


Fig. 6. Structure of Restricted Boltzmann Machine (RBM)

RBM learning algorithm is based on gradient ascent on the log-likelihood. The log-likelihood of the model (3) for all training vector  $\mathbf{v}$  is

$$\begin{aligned} \ln L_{\theta, \mathbf{v}} &= \ln \prod_{\mathbf{v}} p(\mathbf{v}) = \ln \frac{1}{Z_{\theta}} \sum_{\mathbf{h}} e^{-E_{\theta}(\mathbf{v}, \mathbf{h})} \\ &= \ln \sum_{\mathbf{h}} e^{-E_{\theta}(\mathbf{v}, \mathbf{h})} - \ln \sum_{\mathbf{v}, \mathbf{h}} e^{-E_{\theta}(\mathbf{v}, \mathbf{h})} \end{aligned} \quad (4)$$

The goal is to find a set of parameters which can make  $\ln \prod_{\mathbf{v}} p(\mathbf{v})$  to its maximum. The parameters of RBM network are adjusted according to the principle of maximum likelihood. Maximizing the likelihood is the same as maximizing the log-likelihood function for all vectors  $\mathbf{v}$  which is given by

$$\ln L_{\theta, S} = \ln \prod_{\mathbf{v}} p(\mathbf{v}^r) = \sum_{r=1}^N \ln p(\mathbf{v}^r) \quad (5)$$

The purpose of training RBM is to get the optimal value of parameter  $\theta$ , that is

$$\theta^* = \operatorname{argmax}_{\theta} (\ln L_{\theta, S}) \quad (6)$$

Where  $\theta^*$  is the optimal value that makes the free energy of RBM system be minimum. Here the gradient decent method is used to find the maximum value of gradient  $\ln L_{\theta, S}$  with respect to the parameter  $\theta$ , we have

$$\frac{\partial \ln L_{\theta, S}}{\partial \theta} = \sum_{r=1}^N \frac{\partial \ln p(\mathbf{v}^r)}{\partial \theta} \quad (7)$$

From Eq. (7), w.r.t. the parameters  $(\mathbf{W}, \mathbf{a}, \mathbf{b})$  we get

$$\frac{\partial \ln L_{\theta, S}}{\partial w_{i,j}} = \sum_{r=1}^N \left[ p(h_j = 1 | \mathbf{v}^r) v_i^r - \sum_{\mathbf{v}} p(\mathbf{v}) p(h_j = 1 | \mathbf{v}) v_i \right] \quad (8)$$

$$\frac{\partial \ln L_{\theta, S}}{\partial a_i} = \sum_{r=1}^N \left[ v_i^r - \sum_{\mathbf{v}} p(\mathbf{v}) v_i \right] \quad (9)$$

$$\frac{\partial \ln L_{\theta, \mathbf{v}}}{\partial b_j} = \sum_{r=1}^N \left[ p(h_j = 1 | \mathbf{v}^r) - \sum_{\mathbf{v}} p(\mathbf{v}) p(h_j = 1 | \mathbf{v}) \right] \quad (10)$$

Where  $i = 1, 2, 3, \dots, n$  (No. of visible neuron),  $j = 1, 2, 3, \dots, m$  (No. of hidden neurons)

Considering that there are no direct connections between the hidden units, the conditional probability of the binary state of unit  $h_j$  being set to 1 given visible vector  $\mathbf{v}$  can be calculated as

$$p(h_j = 1 | \mathbf{v}) = \operatorname{sigmoid} \left( b_j + \sum_{i=1}^n w_{i,j} v_i \right) \quad (11)$$

Also given a hidden vector  $\mathbf{h}$ , the probability of the visible unit being 1 could be obtained as

$$p(v_i = 1 | \mathbf{h}) = \operatorname{sigmoid} \left( a_i + \sum_{j=1}^m w_{i,j} h_j \right) \quad (12)$$

Due to the high computational complexity of  $\sum_{\mathbf{v}}$ , i.e.,  $O(2^n + 2^m)$  updating the parameters based on these gradient formulas is not feasible. An efficient approximation method CD algorithm proposed by Hinton [7] has been adopted here. The gradients of the log-likelihood w. r. t parameters  $\theta \{W, a, b\}$  for the training pattern  $\mathbf{v}$  is then approximated by CD algorithm as follows:

$$\frac{\partial \ln L_{\theta,S}}{\partial w_{i,j}} \approx \sum_{r=1}^N [p(h_j = 1|v^{(r,0)})v_i^{(r,0)} - p(h_j = 1|v^{(r,k)})v_i^{(r,k)}] \quad (13)$$

$$\frac{\partial \ln L_{\theta,S}}{\partial a_i} \approx \sum_{r=1}^N [v_i^{(r,0)} - v_i^{(r,k)}] \quad (14)$$

$$\frac{\partial \ln L_{\theta,v}}{\partial b_j} \approx \sum_{r=1}^N [p(h_j = 1|v^{(r,0)}) - p(h_j = 1|v^{(r,k)})] \quad (15)$$

Where K is the number of sampling times in CD algorithm equal to 1 and 0 represents the starting points of sampling. Now, the parameters  $\theta$  with  $(\mathbf{W}, \mathbf{a}, \mathbf{b})$  are updated using Eqns. (13) to (15) as outlined in algorithm 1. An individual RBM can be trained efficiently using the learning rules in eqns. (13)-(15). The procedures described above is the pre-training stage of the DBN and is performed unsupervised manner.

---

**Algorithm 1:** Parameter Updating Algorithm

---

**Input:** RBM; Training Set S

No. of Hidden Neurons m;

No. of Iteration: t;

**Output:** Gradient approximation  $\Delta w_{ij}, \Delta a_i, \Delta b_j$  for  $i=1,2,\dots,n$  and  $j=1,2,\dots,m$

Estimated parameters  $(\mathbf{W}, \mathbf{a}, \mathbf{b})$

1. Initialization of Weight Matrix  $\mathbf{W}$ , the visible layer bias  $\mathbf{a}$  and the hidden layer bias  $\mathbf{b}$  based on S and n
  2. **do**
  3.  $v^{(0)} \leftarrow v$
  4. **for** t=1 to k-1 **do**
  5.     **for** i = 1,2,3,.....n **do** sample  $h_j^{(t)} \approx p(h_j|v^{(t)})$
  6.     **for** j = 1,2,3,.....m **do** sample  $v_i^{(t+1)} \approx p(v_i|h^{(t)})$
  7.     **for** i=1,2,3,.....n; j=1,2,3,4,.....m **do**
  8.          $\Delta w_{ij} \leftarrow \Delta w_{ij} + p(h_j = 1|v^{(0,0)})v_i^{(0,0)} - p(h_j = 1|v^{(0,k)})v_i^{(0,k)}$
  9.          $\Delta a_i \leftarrow \Delta a_i + v_i^{(0,0)} - v_i^{(0,k)}$
  10.         $\Delta b_j \leftarrow \Delta b_j + p(h_j = 1|v^{(0,0)}) - p(h_j = 1|v^{(0,k)})$
  11.     **end**
  12. **Return** (updated value)
- 

## 2) Training of the P-DBN Model

Image patches of size 32×32 are converted into the vector of size 1024 which is given as an input to P-DBN architecture for training and testing phase. The proposed P-DBN-LR architecture is trained on the balanced class of features concerning its road regions (foreground) and non-road regions (background). We extracted 1,489,92 non-overlapping patches from the training dataset, where half of the patches represents the foreground and the remaining half belongs to the background.

DBN is composed by stacked of RBMs one by one each of which has its own visible layer and output layer [24]. DBN is formed by stacking the layers of RBM and initialize the network weights by using a greed learning method. Since each layer of DBN is made as RBM, training each layer of DBN is the same as training an RBM. Representative features are extracted by unsupervised pre-training of DBN using CD-1 [23] and supervised fine-tuning phase with standard back propagation algorithm. After the supervised training, a LR is added to the end of feature learning system to constitute a four-layer deep network P-DBN-LR architecture. This LR classifier is employed to fine-tune the whole pre-trained network in a supervised way and classifies the patches from

RS images. The fine-tuning stage includes two phases: only the output layer is trained in the first phase; in the second phase, all layers are fine tuned. Fine-tuning is conducted in a supervised way with labelled data and treated it as classification method. At the early stage of the fine-tuning, only the weights connected to the output layer were adjusted; and after certain number of epochs, the weights of all layers were tuned sequentially. The obtained parameters can be directly applied to the incoming new data, which enables efficient object extraction.

The three hidden layers are used to extract the features from RS images. As illustrated in Fig. 7, in P-DBN model, the input is fed to the lower RBM and the hidden layer output of each layer worked as input to the visible layer of the subsequent RBM stage. The joint probability distribution between the input data  $v$  and the layer hidden layer  $h^k$  in the visible layer is shown in Eqn. (16).

$$p(v, h^1, h^2, h^3 \dots, h^l) = \left( \prod_{k=0}^{l-2} p(h^{k+1}|h^k) \right) p(h^{l-1}, h^l) \quad (16)$$

Where  $p(h^{l-1}, h^l)$  is the joint probability distribution between the visible and hidden layers of the top most RBM.

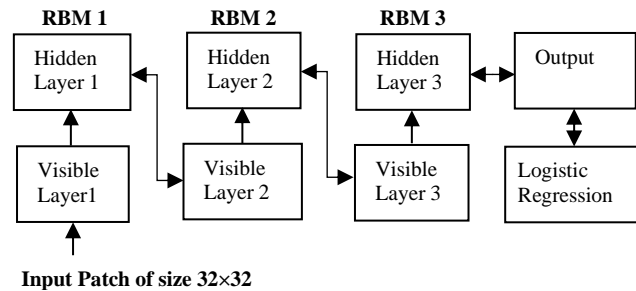


Fig. 7. Training Process

The training data were spilt into 1124 mini batches, and validation data were divided into 388 mini batches. The architecture of P-DBN was selected as 1024-100-200-200-2, where the number of input nodes is 1024. Given a test image, non-overlapping patches of size 32×32 are used as input to the trained architecture to produce prediction probability maps as either a positive sample (road) or a negative sample (background). For the classification purpose LR is considered as a proficient way. As our problem is a binary classification problem, Sigmoid function is used in the output layer to show the probability map of road or non-road objects. The sigmoid function helps in transforming the output of logistic regression into the probability values. For the non-overlapped image patch, the final probability maps will be the average probability map of non-overlapping patches that can be used for road segmentation semantic maps.

## 3) Testing Scheme

Once the network is trained and network parameters are learned, image-patch data of a test data can be fed into the model to directly predict the road and non-road class. For the testing stage, the same pre-processing steps are applied on the testing data which were used in training. Image patches are created in the same fashion as in the training and these patches are then passed to the trained model. The patches from the RS images used in the training stage are not used in

testing for the model. For the classification purpose LR is considered as a proficient way. As our problem is a binary classification problem, Sigmoid function is used in the output layer to show the probability map of road or non-road objects. The segmented image is obtained from the reconstruction of test patch probability map distribution. Since the predictions are processed in a patch-wise manner, after placing a distribution at each of the feature vectors, the final probability map is generated by adding all distributions.

However, some pixels of the background are segmented as road because of high contrast among road, buildings. Moreover, some road pixels are missing from foreground. To remove such flaws (red circled) from the segmented images, we apply different morphological operations [28] as a post-processing step to obtain the final segmentation.

*D. Post-Processing*

The objective of this refinement process is to eliminate the non-road regions or false segments which do not belong to roads. To eliminate these false segments, Connected Component Analysis (CCA) [28] is used at first to group pixels into different components based on pixel connectivity to extract the disjoint segments from the output of the previous stage. Components whose surface areas are smaller than a predefined threshold will be deleted. Region linking algorithm [28] is used to eliminate the discontinuities detected between road segments. Initially a dilation operation is performed on the CCA images to link the edge segments which are very close to each. A structural element of disk shape of radius 10 is used for the dilation operation followed by morphological thinning operation. Morphological closing is then applied to remove small holes and noise from the road surface, while an opening operation is used to eliminate small pathways with a structuring element size that is smaller than the main roads width but larger than those of the pathways, resulting in the extracted road network as shown in Fig. 8(c). The filtered image is shown in Fig. 8(c), it can be clearly seen that all misclassified objects unconnected to the main road network were removed.

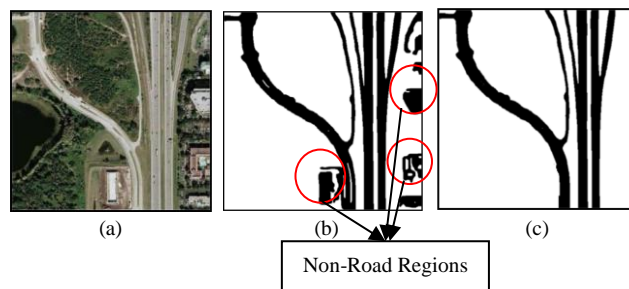


Fig. 8. (a) Input Image (b) Classified Image (c) Final result after refinement process

III. EXPERIMENT AND RESULTS

The experiments and performance evaluation are carried out on RS images having different types of urban and suburban areas with various classes such as roads, buildings, trees, vegetation, and shadow. The proposed approach is implemented with MATLAB and tested on a Core i7 2.67GHz PC with 8GB RAM.

A. Dataset Creation

To evaluate the proposed approach, the experiment is carried out on a dataset of 970 RS imagery of size 512x512 each among which 500 images of urban and 470 images of suburban regions. For creating the data set, we use the benchmark dataset created by Cheng et al. [12] which consists 240 RS images collected from Google Earth. The current work also makes use of a part of database used in [5], images from Massachusetts Roads Dataset [26]. The dataset used were augmented by introducing random flips and rotation. In the present work, only 3-Band RGB image are utilized of all the datasets to extract the road networks. Fig.9 shows few samples from our dataset. For each image in the data set, a ground truth (road) map was also obtained using a human operator as shown in Fig.9. These labelled datasets are randomly divided into 60% for training data sets, 20% for validation data sets, and 20% for test data sets as shown in Table I.

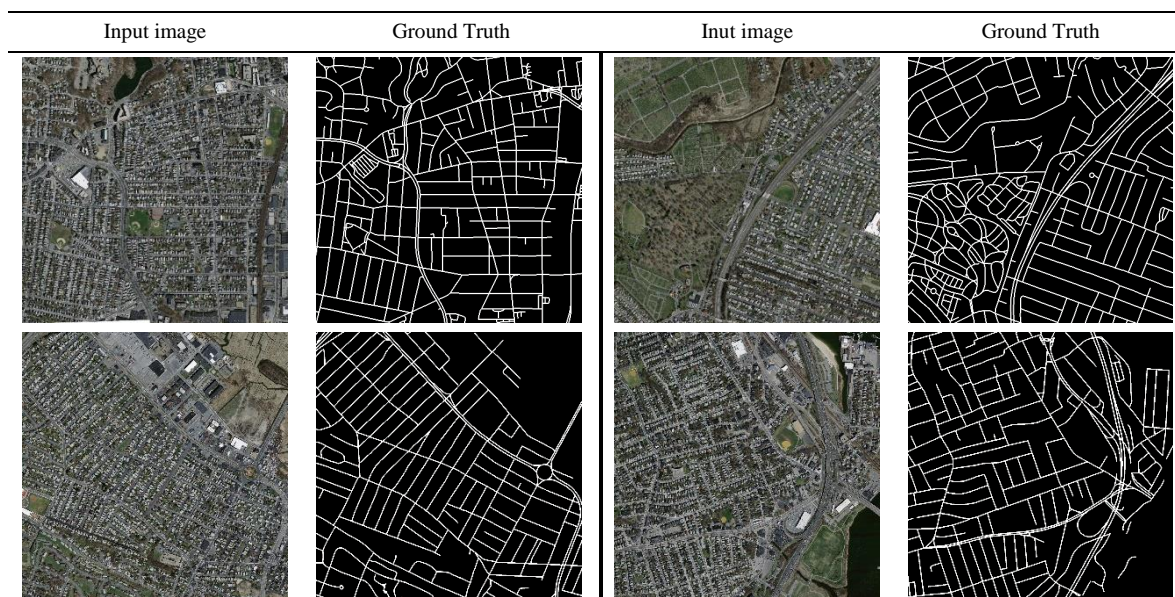


Fig. 9. Sample RS imagery from road dataset. The input imagers and corresponding ground truth maps.

TABLE I  
NUMBER OF SAMPLES PER CLASS FOR THE TRAINING,  
VALIDATION AND TEST SET

Land Cover	Data set (970 image patches)		
	Training (60%)	Testing (20%)	Validation (20%)
Road	302	120	100
Non-Road	280	74	94

All RGB images are converted into grayscale images. Image patches of size 32×32 are generated from the input images. Then, the image patches of size 32×32 are converted into the vector of size 1024 which is given as an input to P-DBN architecture for training and testing phase. We extracted 1,489,92 non-overlapping patches from the training dataset, where half of the patches represents the road regions i.e. foreground and the remaining half belongs to the non-road regions or background. Samples of 49664 patches of 1024 dimension are used for validation purpose. The training data were spilt into 1124 mini batches, and validation data were divided into 388 mini batches. The architecture of P-DBN was selected as 1024-100-200-200-2, where the number of input nodes is 1024.

**B. Parameters Settings**

After the preprocessing step biases and weights are initially set to zero. The P-DBN model has three hidden layers with nodes 100, 200 and 200, respectively. Two nodes in the output layer are used to classify features to represent each patch into road and non-road regions. Each RBM is trained using CD algorithm in a greedy manner. The learning rate was set to 0.001 during the pre-training stage, and 0.5 was the momentum with 100 epochs for RBM training. Mini-batch size is set to 128 in the pre-training and fine-tuning stages where sigmoid was used as an activation function. An image patch of size 32×32 was used to generate 1024 vector dimension which is given as an input to the visible layer of first RBM in training the model. The second RBM was trained on the output of first RBM. Total of 970 images were used for patches dataset creation in our experiment where patch dataset is separated with a ratio of 60:20:20 for training, validation and testing respectively. In the training phase, a matrix with 1,489,92×1024 dimensions is used and for validation, a matrix of 49664×1024 is used as input and the mini-batch size was 128. The learning rate was 0.005 with momentum with 0.7 using back-propagation neural network. For fine tuning we trained our model on 2000 epochs. The parameters setting is given in Table II.

TABLE II  
PARAMETERS SETTINGS

Parameter	Pre-training	Fine-tuning
Hidden layer	3	3
No. of neuron in each layer	100-200-200	100-200-200
No. of epochs	100	2000
Learning Rate	0.001	0.005
Momentum	0.5	0.7
Batch size	128	128

**C. Performance Evaluation**

The quantitative evaluation of the experimental results is achieved by comparing the automated (derived) results against a manually formed ground truth data. For assessing the performance of the proposed road extraction method, the ‘Overall Accuracy’ (OA), ‘Precision’, ‘Recall’, Quality and ‘F1-score’ are used as quality metrics [1,12,34].

The *recall* value indicates the percentage of the ground truth road pixels detected. The *Precision* indicates the percentage of the correctly classified road pixels among all predicted pixels of the classifier. Finally, the *quality* value shows the goodness of the result. The *F1-Score* indicates the harmonic average of *Precision* and *Recall*. The Classification Overall Accuracy (OA) is used to measure the rate of images that were correctly classified. The values of these metrics are in the range of 0 to 1, and higher values indicate better classification performance. The metrics are calculated as follows:

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

$$Precision = \frac{TP}{TP + FP} \tag{18}$$

$$Quality = \frac{TP}{TP + FP + FN} \tag{19}$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{20}$$

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{21}$$

where TP denotes the true positive; FP denotes the false positive, and FN denotes the false negative.

For evaluating the effectiveness of the proposed technique for road object extraction, the qualitative segmentation results are presented on the different dataset in Fig.10 to Fig.12. Road extraction results from the proposed technique on samples in the Massachusetts Road dataset are presented in Fig.10 for a qualitative assessment. The eight experimental results can highlight algorithm’s efficiency for road detection.

Fig.11 shows a series of results generated in the process of extracting roads, including sample regions of suburban category of datasets, the reference road maps and the final road extraction results. They contain all three bands (i.e., R, G, B) and rich information including roads, various buildings, vegetation etc. The images in first column are the original RS images of our database. Those in the third column indicate the automatic results of proposed framework. The final road extraction results can then be generated after post-processing, as presented in the images of the fourth column. The results are assessed using the reference road extracted by human operator manually illustrated in second column.

Fig.12 illustrates a series of results generated in the process of extracting roads from the dataset of urban category. All the images contain rich information, including buildings, roads, vegetation, parks etc.

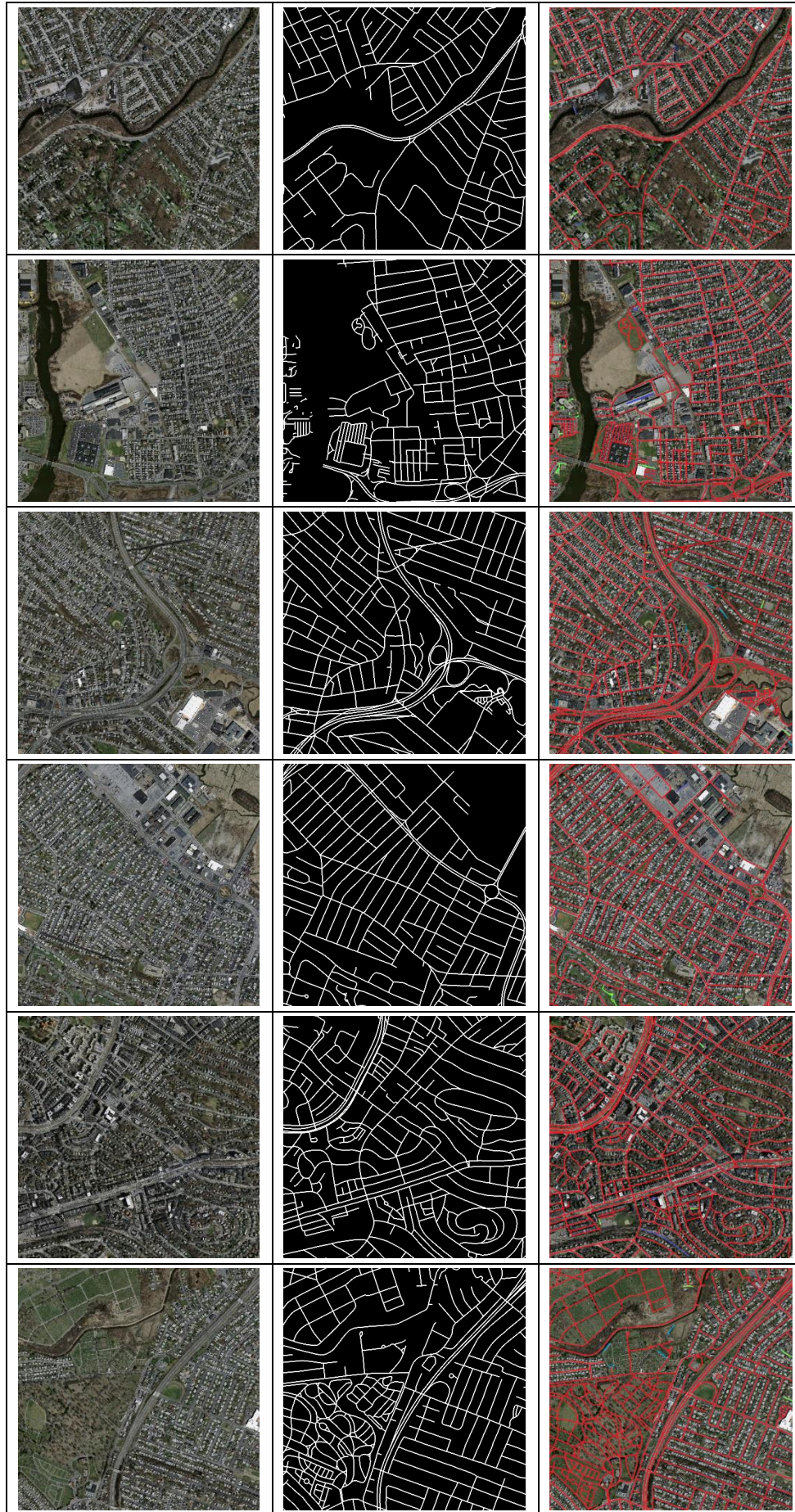


Fig. 10. Road extraction Results from Massachusetts Road Dataset achieved by proposed method. The First and second column depict the input images and their corresponding ground truth images from Massachusetts road dataset. The third column shows the building extraction results. The red, blue, and green colours represent TPs, FPs, and FNs, respectively.




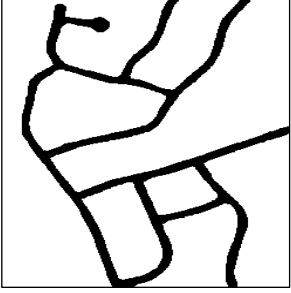
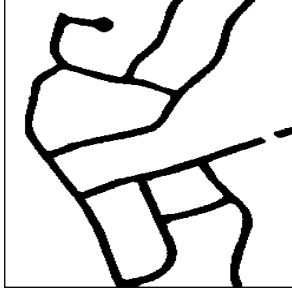
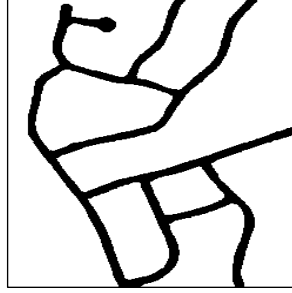


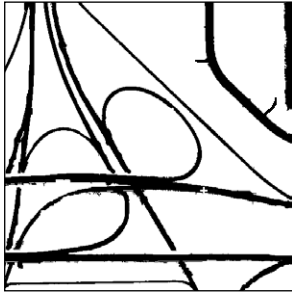
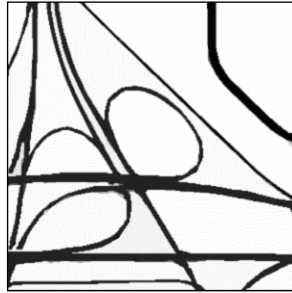

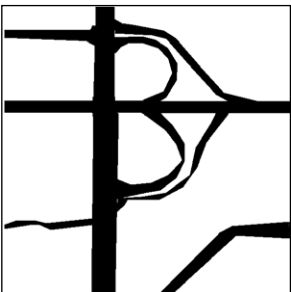
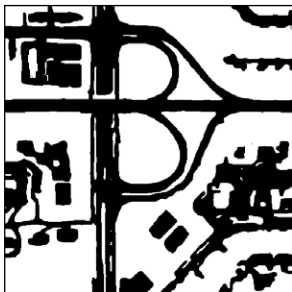
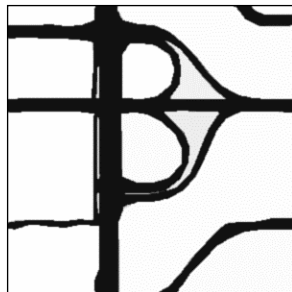

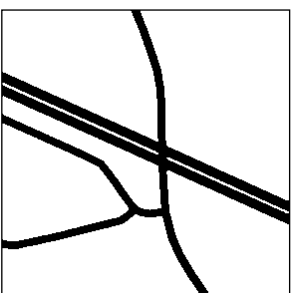
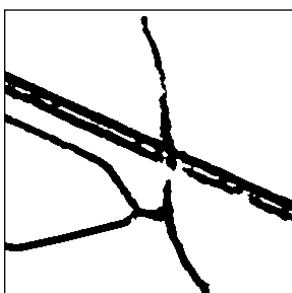
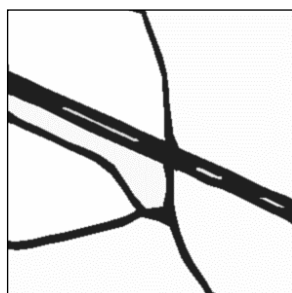
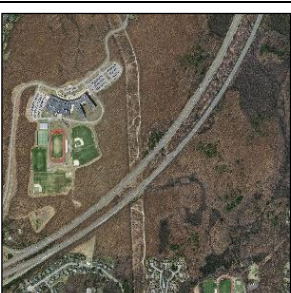
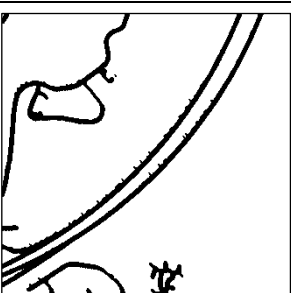


Input Image	Ground Truth	Initial Extraction Results	Final Extraction Results
			
			
			
			
			

Fig. 11. Road Extraction Results of Sub-urban areas from the data set achieved by the proposed method. First and second columns show the original images and corresponding ground truth. The third and fourth columns are results achieved before post-processing and after post-processing.

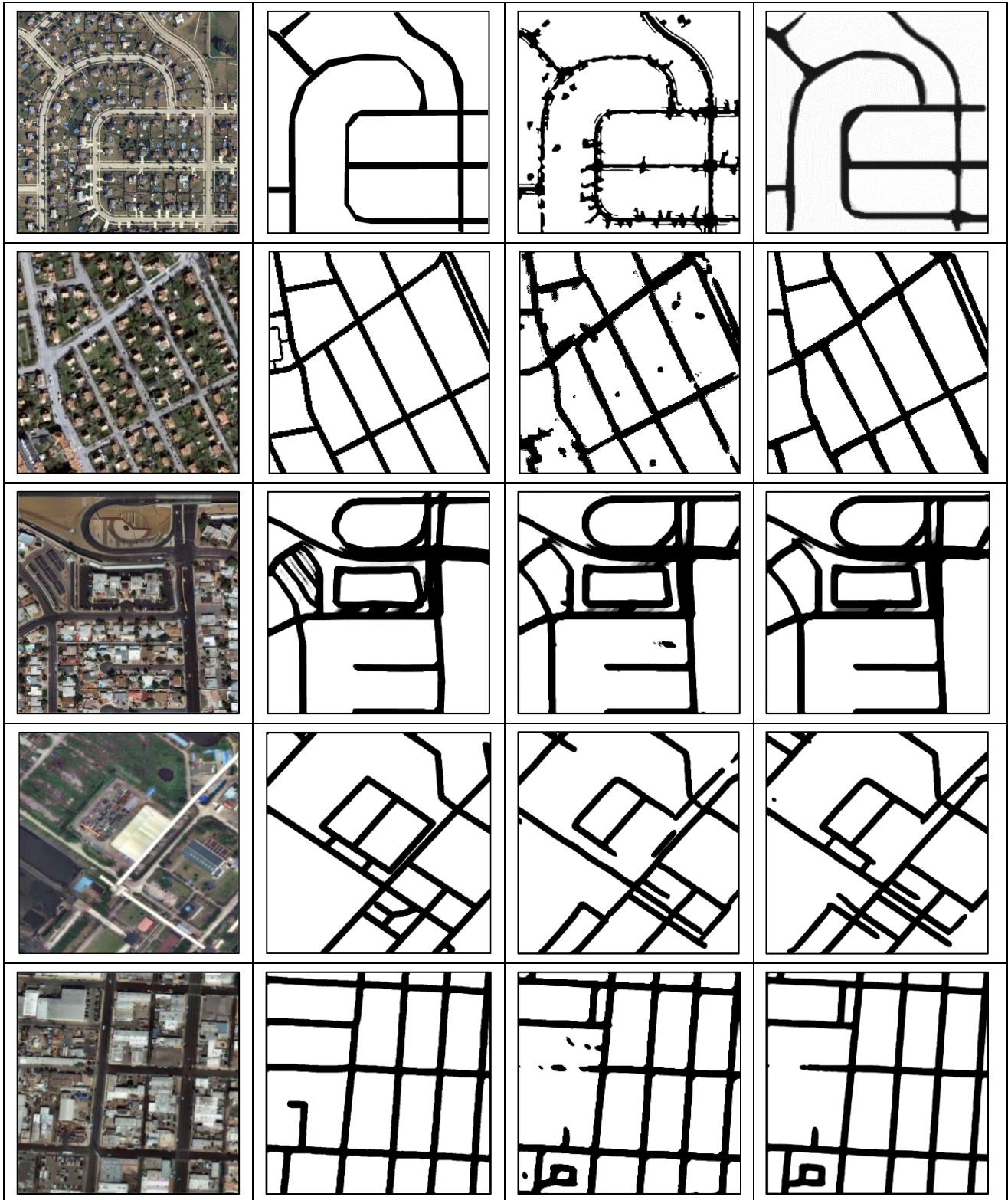


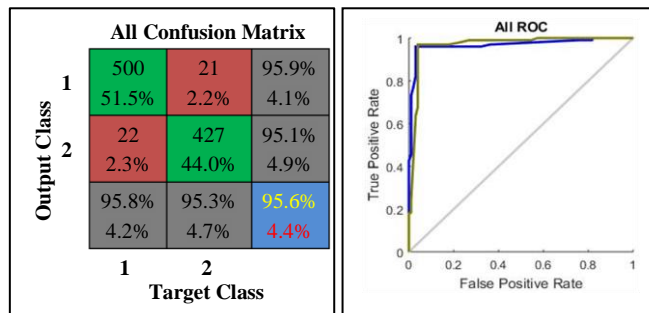
Fig. 12. Road Extraction Results of Urban areas from the data set achieved by the proposed method. First and second columns show the original images and corresponding ground truth. The third and fourth columns are results achieved before and after post-processing.

The performance of the proposed method is calculated by analysis of confusion matrix and the receiver operator characteristic curve (ROC). The confusion matrix and the ROC of all data are shown in Fig.13. Of total 970 samples, 927 samples were correctly classified and 43 samples were misclassified by this DLA as shown in 13(a). The proposed model resulted in an OA of 95.57%, recall 95.79%, precision 95.97%, quality 92.08% and F1-score 0.9588. ROC graph

depicted in Fig. 13(b) shows the plotting of true positive rate (recall) against false positive rate. ROC graph of this network for all data shows an excellent classification between the two categories as the curves lie between the diagonal and the upper-left corner but mainly towards the upper-left corner.

Tables III list the overall Recall, Precision, Quality and F1-Score of the proposed method of training, testing, validation and all data, respectively. As seen from the table, the

proposed system using deep learning technique incurs the acceptable level of performance with the mean values of no less than 95.5, 95.97, 92.08, 0.9588 and 95.60 for Recall, Precision, Quality, F1-Score, and OA respectively. The overall performance analysis is depicted graphically in Fig. 14 and Fig.15. The proposed method obtains satisfactory performance for all cases, in terms of both visual interpretation and quantitative assessment.



(a) (b)

Fig. 13. Confusion matrix and ROC plot of all data

TABLE III  
PERFORMANCE ANALYSIS OF PROPOSED TECHNIQUE OF TRAINING, TESTING DATA, VALIDATION AND ALL DATA

	P-DBN-LR DLA				
	Recall	Precision	Quality	F1-Score	OA
Training (%)	95.03	95.66	91.11	0.9535	96.1
Validation (%)	95.00	93.14	88.78	0.9406	94.3
Testing (%)	95.8	94.26	90.55	0.9504	94.3
All data (%)	95.79	95.97	92.08	0.9588	95.6

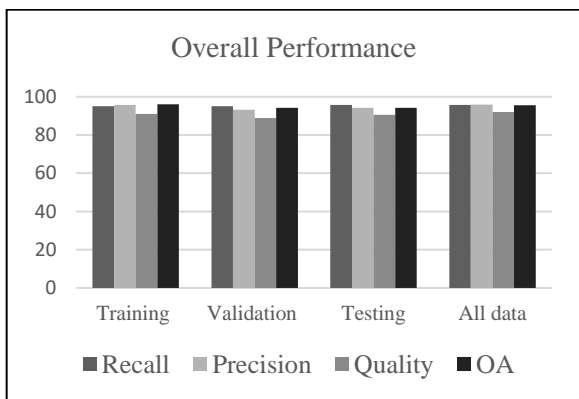


Fig. 14. Overall performance analysis of training, testing dataset, validation and all dataset

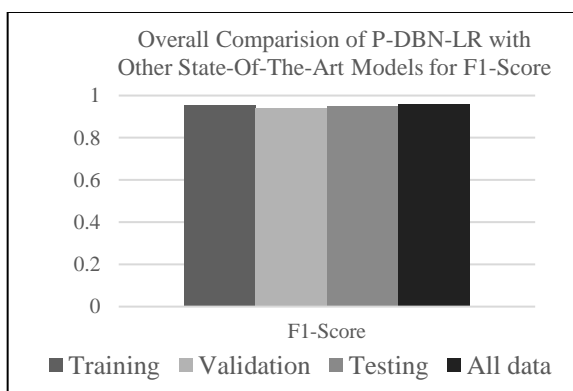


Fig. 15. Overall performance analysis of F1-Score

Mean squared error (MSE) [27] was used to measure the error during the back propagation neural network training. The training error was recorded 0.09 and validation error was 0.10 on 2000 epochs wherever no convergence was recorded after these iterations. The best error rate is achieved at this point and the experiment was stopped here. In Fig.16, two exemplary plots are provided depicting the changes in training and validation loss over epochs for P-DBN trained model. It is observed that both curves show a decreasing tendency and converge to a minimum. The gradual decrease in the MSE indicates the efficiency of the proposed architecture for learning suitable representations associated to the underlying mapping between the input and output.

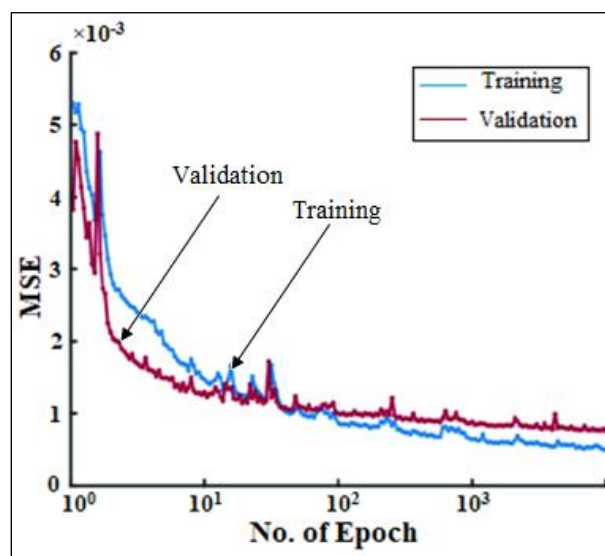
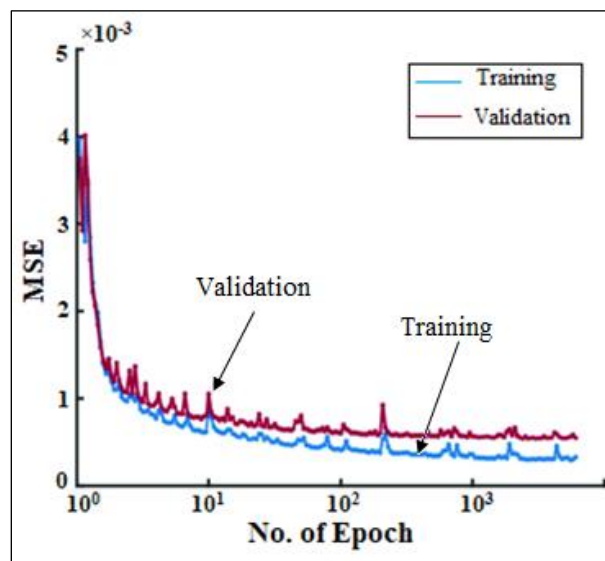


Fig. 16. The learning curve over epochs obtained by training the proposed P-DBN architecture

Table IV presents the elapsed time of each section. All test images took 67.54 minutes which corresponds to an average processing time of approximately 21 seconds for each image. On the other hand, 35% - 40% of the total processing time is spent on during the training (Sec. II.C(2)) and classification process (Sec. II(C(3))). A neglectable percentage of the total time is spent on the pre-processing step (Sec. II(A)) and post-

processing (Sec. II(D)). It is also noted that, the proposed approach is quite suitable for GPU programming.

TABLE IV  
ELAPSED TIME OF EACH SECTION OF THE PROPOSED ROAD EXTRACTION APPROACH

Process (Section)	Elapsed Times min: sec (%)
Pre-processing stage (Sec. II (A))	05:21 (7.72%)
Training Extraction (Sec. II (C (2)))	30:68 (45.43%)
Classification (Sec. II (C (3)))	27:52 (40.74%)
Post-processing (Sec. II (D))	04:13 (6.11 %)
<b>Total</b>	<b>67:54 (100%)</b>

D. Comparison of Road Extraction Algorithms

To assess the relative significance of our approach, a comparison with 5 state-of-the-art methods is undertaken as baseline which include U-Net [27], Cascaded CNN [12], Roadtracer [19], FCN [14] and Salient features-SVM [5]. Table V shows the comparative quantitative evaluation measured in terms of Recall, Precision, Quality, F1-score and OA. The best performance is marked in bold, and the next best is marked with underlines. We found that the proposed method can give the relatively high average performance better than the other methods.

The quantitative comparison of the different networks for the five test images is presented in the first five rows in Table V and the average performance is shown in the last row of the Table V. The proposed P-DBN-LR architecture delivers improvements on all evaluation metrics over the other models except for precision. The P-DBN-LR model achieved best result among all models on OA metric with an improvement of 3.22% over U-Net (95.50 vs 92.42). The proposed P-DBN-LR model achieved best result among all models on F1-Score metric with an improvement of 0.424% (0.9530 vs. 0.9490) over the next best model Roadtracer. As for Precision, the U-Net model holds the highest values and gains 0.04% over our P-DBN-LR model (94.898 vs. 94.936). For Recall, the cascadedCNN, Roadtracer, and P-DBN-LR methods scored significantly better performance over the other three methods while P-DBN-LR achieves the best value being 0.35% (94.75 vs. 94.42) ahead of the Roadtracer method. Similarly, P-DBN-LR achieves the best quality metric being 0.79% (91.20 vs. 90.48) ahead of the U-Net model where CascadedCNN and Roadtracer achieves the best model amongst the others. For the F1-Score metric, P-DBN-LR has scored the best value 6.92% ahead of U-Net (0.9530 vs. 0.8871) and even 2.4% ahead of CascadedCNN (0.9530 vs. 0.9302). Compared to the FCN, the proposed model yields a higher F1-score by 11.11% (0.9530 vs.0.8471) and by 14.99% (0.9530 vs. 0.8101) over salient-SVM method. When compared to other state-of-the-art deep learning-based models such as U-Net, Cascaded CNN, Roadtracer, FCN and Salient features-SVM, our proposed P-DBN-LR model outperforms by 3.22% (0.9550 vs. 0.9242), 4.56% (0.9550 vs. 0.9114), 3.30% (0.9550 vs. 0.9233), 5.54% (0.9550 vs. 0.9020) and 7.40% (0.9550 vs. 0.8843), for the dataset. Experimental results demonstrate the effective performance of the proposed method for extracting roads from a complex scene. The overall average quantitative

comparison of the different networks is depicted graphically in Fig. 17 and Fig. 18.

TABLE V  
COMPARISON AND QUANTITATIVE EVALUATION FOR COMPLEX URBAN AREAS SHOWN IN FIG. 11

		Proposed	U-Net [27]	Cascaded CNN [12]	Roadtracer [19]	FCN [14]	Salient Feature[5]
Test Image 1	Recall	<b>94.21</b>	92.27	<u>93.98</u>	93.97	88.20	86.10
	Precision	<u>94.19</u>	<b>94.86</b>	93.47	94.17	91.31	84.97
	Quality	<b>90.85</b>	<u>90.04</u>	89.69	89.15	88.90	75.08
	F1-Score	<b>0.9415</b>	0.8934	0.9304	<u>0.9401</u>	0.8701	0.8081
	OA	<b>95.36</b>	<u>92.45</u>	91.28	92.23	90.08	88.25
Test Image 2	Recall	<b>95.01</b>	93.27	94.08	<u>94.87</u>	91.20	85
	Precision	<b>95.57</b>	<u>95.26</u>	92.47	94.47	91.31	84.67
	Quality	<b>91.95</b>	<u>90.34</u>	88.89	89.05	88.36	74.32
	F1-Score	<b>0.9572</b>	0.8831	0.9314	<u>0.9531</u>	0.8501	0.8091
	OA	<b>95.62</b>	<u>92.42</u>	91.38	92.35	90.32	89.05
Test Image 3	Recall	<b>94.98</b>	93.47	93.91	<u>94.55</u>	93.20	86.01
	Precision	<b>95.05</b>	<u>95.46</u>	92.87	95.27	92.31	85.67
	Quality	<u>90.82</u>	<b>91.32</b>	89.31	89.25	88.75	76.12
	F1-Score	<b>0.9550</b>	0.8854	0.9294	<u>0.9541</u>	0.8201	0.8131
	OA	<b>95.56</b>	92.61	91.37	<u>92.78</u>	90.35	87.85
Test Image 4	Recall	<b>94.73</b>	93.01	93.92	<u>94.59</u>	91.86	85.70
	Precision	<b>94.93</b>	<u>94.61</u>	92.93	94.63	91.64	85.10
	Quality	<b>91.20</b>	<u>90.56</u>	89.29	89.15	88.67	75.17
	F1-Score	<b>0.9565</b>	0.8873	0.9304	<u>0.9491</u>	0.8467	0.8101
	OA	<b>95.43</b>	<u>92.42</u>	91.35	92.23	90.12	88.74
Test Image 5	Recall	<b>94.83</b>	92.42	93.85	<u>94.32</u>	91.66	85.69
	Precision	<b>94.95</b>	94.49	92.89	<u>94.58</u>	91.74	85.14
	Quality	<b>91.18</b>	<u>90.15</u>	89.12	89.11	88.77	75.11
	F1-Score	<b>0.9548</b>	0.8864	0.9296	<u>0.9487</u>	0.8485	0.8102
	OA	<b>95.51</b>	<u>92.21</u>	90.32	92.15	90.15	88.25
Average	Recall	<b>94.75</b>	92.88	93.95	<u>94.42</u>	91.22	85.70
	Precision	<u>94.90</u>	<b>94.94</b>	92.926	94.62	91.66	85.11
	Quality	<b>91.2</b>	<u>90.48</u>	89.26	89.142	88.69	75.16
	F1-Score	<b>0.9530</b>	0.8871	0.9302	<u>0.9490</u>	0.8471	0.8101
	OA	<b>95.50</b>	<u>92.42</u>	91.14	92.35	90.20	88.43

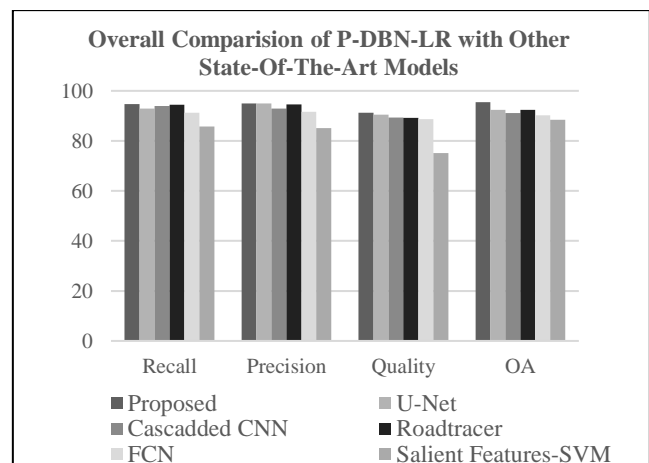


Fig. 17. The comparative quantitative evaluation measured in terms of Recall, Precision, Quality and OA with 6 baseline methods U-Net [27], Cascaded CNN [12], RoadCNN [19], FCN [14] and Salient features and SVM [5]

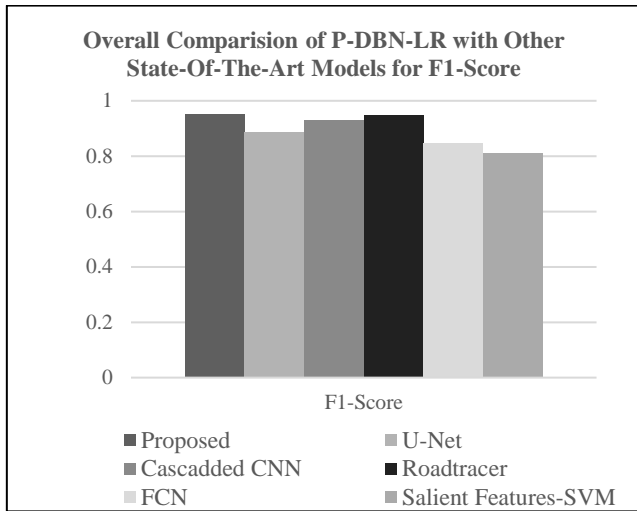


Fig. 18. The comparative quantitative evaluation measured in terms of F1-score with 6 baseline methods U-Net [27], Cascaded CNN [12], RoadCNN [19], FCN [14] and Salient features and SVM [5]

For evaluating the effectiveness of the proposed P-DBN-LR architecture for road extraction, the qualitative segmentation results are presented for all six models on the dataset in Fig. 19. The proposed P-DBN-LR architecture generally perform better than the baseline methods. For FCN

and U-Net, we find that it misses road regions i.e., the false negative (blue) part is large, especially in the corner parts of roads, or the occlusion parts caused by trees and buildings. Moreover, a few discontinues and false positive parts are observed in the results.

From Fig. 19, it is observed that Cascaded CNN and RoadCNN produce spikes in some points in roads and spurs, false positive parts are seen in results. They also miss some road regions i.e., the false negative (blue) part. We can see from the figure that salient features-based method performs the worst, which misses many ground truth road segments. Moreover, many false positive parts are observed in the results produced by salient features-based method. Though, some false positive parts still appear in the results produced by our proposed DLA but it does not miss out any road regions i.e., the false negative (blue) parts. The road networks detected by the proposed P-DBN-LR are the most consistent results.

Fig.20 shows another qualitative result of the proposed approach and other state-of-the approaches which demonstrates the effectiveness of the proposed model.

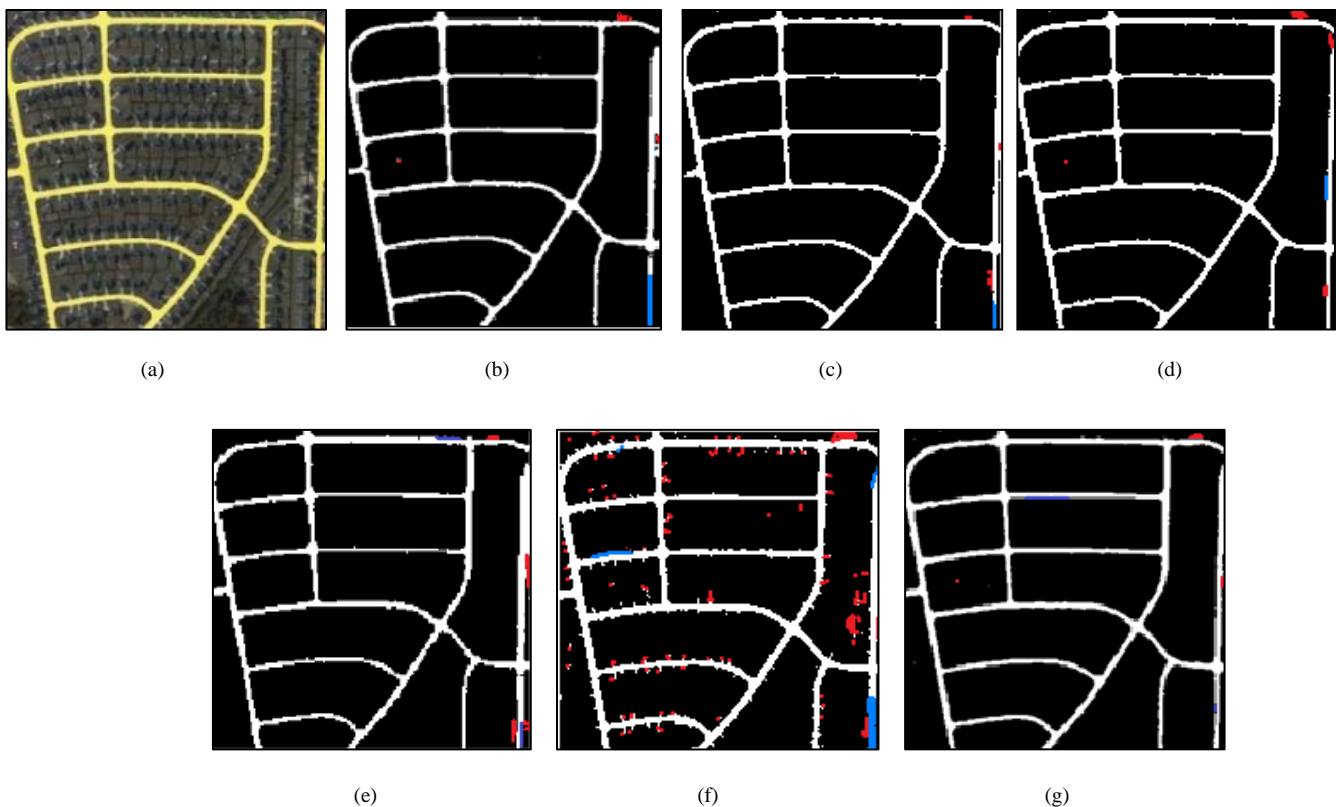


Fig. 19. Visual comparison of road extraction results achieved by different methods (a) Input Image from dataset (Road networks masked with yellow colour) (b) FCN (c) U-Net (d) Cascade CNN (e) RoadCNN (f) Salient Features-SVM and (g) proposed method based. The white colour denotes the road parts (true positive), red and blue colour denote false positive and the false negative parts.



(a) Input Image



(b) Ground Truth



(c) Proposed



(d) FCN



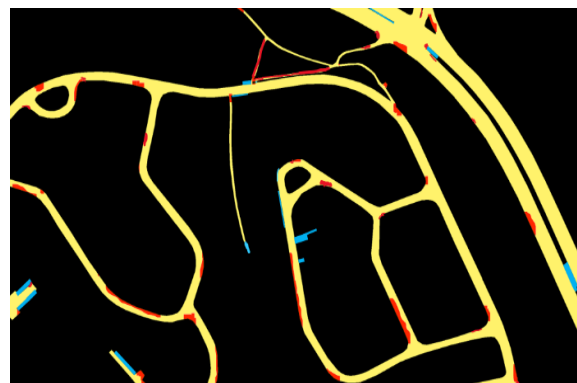
(e) Cascaded CNN



(f) Roadtracer



(g) U-Net



(h) Salient Feature

Fig. 20. Road extraction Results from Massachusetts Building Dataset achieved by proposed model and other state-of-the-art methods. First row shows the original images and corresponding ground truth. The second, third, and fourth rows are the results achieved by the proposed architecture, FCN [14], Cascaded CNN [12], Roadtracer [19], U-Net [27] and Salient Features and SVM [5], and the, respectively. The black (background), yellow, red and blue colours represent TNs, TPs, FNs, and FPs respectively.

## IV. LIMITATION

Despite the improvements in semantic segmentation of roads from RS images by the proposed P-DBN-LR architecture, some issues remain to be considered. With the rapid development of RS technology, the availability of high-resolution RS imagery with abundant features and spectral information is significantly increased [33]. Extraction of roads from RS images plays a vital role in a wide range of RS applications but poses a major challenge for computer vision and image processing researchers. The proposed P-DBN-LR model is able to help to improve the accuracy of semantic segmentation.

However, this model may fail to generalize to areas with complex and heterogeneous roads because the datasets used in this research do not cover images from different sensors, such as hyperspectral images, DSMs or Light Detection and Ranging (LiDAR) DSM and SAR images. Spectral information is not enough since roads and building roofs can have similar texture. Moreover, in observing only two-dimensional images, we lose the third dimension—height. As a result, accuracy and robustness of the extraction results could be improved by integration of different data sources. Therefore, fusing data sources, such as multi-spectral images with either stereo DSMs or LiDAR DSM rather than the use of only a single data source can be used for solving these problems and as a result, improve image interpretation.

However, these data provide information complementary to the data in the visual spectrum, and therefore there is the potential that training models with these additional data may lead to better segmentation results.

## V. CONCLUSION

In this paper, a DL based feature-extraction and classification model called P-DBN-LR has been proposed to extract road networks from RS images. The proposed model is capable of learning features automatically from the image patches instead of pixel-by-pixel learning, which can significantly save the memory and computational time. The proposed P-DBN-LR deep model resulted in an OA of 95.58% and F1-score of 0.9552. Compared with few state-of-the-art road extraction network architectures, our proposed architecture proves that the patched-based DBN model has superior performance in most of the cases. The proposed method in this work can obtain improvements in terms of the comprehensive evaluation metric, the F1-score, Recall, Precision and Quality for road extraction.

However, the proposed deep model faces challenges to accurately detect roads under the bridges and in presence of large occlusion. We also witness the false positive due to its visual similarity with buildings and unpaved roads. Although our results are encouraging, the proposed method can be improved further by fusing deep features with structural ones in future studies. In the future work, we will use graphics processing unit to accelerate the feature learning process. It is observed that the road detection process produces a high degree of accuracy especially for the images of urban regions. The presence of buildings and other features like roads made the extraction process somewhat more difficult compared to the suburban case. Road junction detection and modelling of shadows are issues to be addressed in future work. In addition, vectorization of the extracted road networks can also be a good extension of this work for GIS applications.

## REFERENCES

- [1] Md. Abdul Alim Sheikh, Alok Kole, Tanmoy Maity, "A Multi-level Approach for Change Detection of Buildings using Satellite Imagery," *International Journal of Artificial Intelligence Tools*, vol. 27, no. 8, pp. 1850031, 2018, DOI: 10.1142/S0218213018500318.
- [2] Abdul Alim Sheikh, S. Mukhopadhyay, "Noise Tolerant Classification of Aerial Images into Manmade Structures and Natural- Scene Images based on Statistical Dispersion Measures," 2012 Annual IEEE Conference (INDICON), 653-658, 2012. DOI: 10.1109/INDCON.2012.6420699
- [3] Zhongbin Li, Wenzhong Shi, Qunming Wang, and Zelang Miao, "Extracting Man-Made Objects from Remote Sensing Images via Fast Level Set Evolutions," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 883–899, 2015.
- [4] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual U-net," *IEEE Geosci. Remote Sens. Lett.*, vol., no. 5, pp. 749-753, 2018.
- [5] Sukhendu Das, T. T. Mirmalinee, and Koshy Varghese, "Use of Salient Features for the Design of a Multistage Framework to Extract Roads from High-Resolution Multispectral Satellite Images, IEEE Transactions on Geoscience and Remote Sensing, vol.49, no. 10, pp. 3906-3931.
- [6] Zhaoli Hong, Dongping Ming, Keqi Zhou, Ya Guo, and Tingting Lu, "Road Extraction from a High Spatial Resolution Remote Sensing Image Based on Richer Convolutional Features," *IEEE Access*, vol. 6, pp.46988 – 47000, 2018. DOI: 10.1109/ACCESS.2018.2867210.
- [7] A. Abdullah, B. Pradhan, N. Shukla, S. Chakraborty and A. Alarming, "Deep Learning Approaches to Remote Sensing Datasets for Road Extraction: A State-of-The-Art Review," *Remote Sensing*, vol.12, 1444, 2020, doi:10.3390/rs12091444.
- [8] Weixing Wang, Nan Yang, Yi Zhang, Fengping Wang, Ting Cao, Patrik Eklund, "A review of road extraction from remote sensing images," *Journal of traffic and transportation engineering (english edition)*, vol., 3, no. 3, pp. 271-282, 2016.
- [9] T. Panboonyuen, K. Jitkajornwanich, S. Lawawirojwong, P. Srestasathien, and P. Vateekul, "Road segmentation of remotely-sensed images using Deep Convolutional Neural networks with landscape metrics and conditional random fields," *Remote Sens.*, vol. 9, no. 7, pp 680-699, 2017.
- [10] Ma, L.; Liu, Y.; Zhang, X.L.; Ye, Y.X.; Yin, G.F.; Johnson, B.A "Deep learning in remote sensing applications: A meta-analysis and review," *ISPRS J. Photogramm. Remote Sens.* vol. 152, pp. 166–177, 2019.
- [11] Xiaofei Yang, Xutao Li, Yunming Ye and Raymond Y. K. Lau, "Road Detection and Centerline Extraction Via Deep Recurrent Convolutional Neural Network U-Net," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 7209-7220, 2019 DOI: 10.1109/TGRS.2019.2912301PP(99).
- [12] G. Cheng, Y. Wang, S. Xu, H. Wang, S. Xiang, and C. Pan, "Automatic Road detection and centerline extraction via cascaded end-to-end convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 6, pp.3322 -3337, 2017.
- [13] Alshehhi, R., Marpu, P. R., Woon, W. L., & Mura, M. D., Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 130 pp. 139-149. 2017, <https://doi.org/10.1016/j.isprsjprs.2017.05.002>.
- [14] Long, J.; Shelhamer, E.; Darrell, T., "Fully convolutional networks for semantic segmentation," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 1337–1342, 2015.
- [15] S. Saito, T. Yamashita, and Y. Aoki, "Multiple object extraction from aerial imagery with convolutional neural networks," *Electron. Imag.*, vol. 60, no.1, pp. 1-9, 2016.
- [16] W. Zhao, S. Du, and W. J. Emery, "Object-based Convolutional neural network for high-resolution imagery classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 7, pp. 3386-3396, 2017.
- [17] Ruyi Liu, Qiguang Miao, Jianfeng Song, Yining Quan, Yunan Li, Pengfei Xu and Jing Dai, "Multiscale road center lines extraction from high-resolution aerial imagery," *Neurocomputing*, vol. 329, pp. 384-396, 2019.
- [18] Jun Wang, Jingwei Song, Mingquan Chen & Zhi Yang, "Road network extraction, A neural-dynamic framework based on deep learning and a finite state machine, *International Journal of Remote Sensing*," vol. 36, no. 12, pp. 3144-3169, 2015, DOI: 10.1080/01431161.2015.1054049.
- [19] F. Bastani, S. He, S. Abbar, M. Alizadeh, H. Balakrishnan, S. Chawla, S. Madden, and D. DeWitt, Roadtracer, "Automatic extraction of road networks from aerial images," *Computer Vision and Pattern Recognition (CVPR)*, 2018, arXiv:1802.03680

- [20] G. Fu, C. Liu, R. Zhou, T. Sun, and Q. Zhang, "Classification for high resolution remote sensing imagery using a fully convolutional network" *Remote Sens.*, vol.9, no.5, pp. 498-519, 2017.
- [21] Z. Zhong, J. Li, W. Cui, and H. Jiang, "Fully convolutional networks for building and road extraction: Preliminary results," In *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, pp. 1591-1594, 2016, DOI: 10.1109/IGARSS.2016.7729406.
- [22] Kestur, R.; Farooq, S.; Abdal, R.; Mehraj, E.; Narasipura, O.; Mudigere, M. "UFCN, A fully convolutional neural network for road extraction in RGB imagery acquired by remote sensing from an unmanned aerial vehicle," *J. Appl. Remote Sens.*, vol. 12, 016020, 2018.
- [23] G.E. Hinton, S. Osindero, and Y.W.Teh., "A Fast Learning Algorithm for Deep Belief Nets," *Neural Comput.*, vol.18, no.7, pp. 1527-1554, 2006.
- [24] Xi-Zhao Wang, T. Shang, and R.Wang, "Noniterative Deep Learning: Incorporating Restricted Boltzman Machine Into Multilayer Random Weight Neural Networks," *IEEE Trans. on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 7, pp. 1299-1308, 2019.
- [25] Mnih, V.; Hinton, G.E. "Learning to detect roads in high-resolution aerial images," In *Proceedings of the European Conference on Computer Vision (ECCV)*, Heraklion, Greece, 5–11 September, 210–223, 2010.
- [26] V.Mnih, "Machine learning for aerial image labeling" Ph.D. dissertation, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2013.
- [27] O. Ronneberger, P. Fischer, and T. Brox., "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Springer, pp. 234-241, 2015.
- [28] Kang, Y.; Engelke, K.; Kalender, W.A., "A new accurate and precise 3-D segmentation method for skeletal structures in volumetric CT data," *IEEE Trans. Med. Imaging* vol. 22, pp. 586–598, 2003.
- [29] Jieling Wu, Noriaki Endo, and Mitsugu Saito, "Cluster Analysis for Investigating Road Recovery in Iwate Prefecture Following the 2011 Tohoku Earthquake, Proceedings of the International MultiConference of Engineers and Computer Scientists 2021 IMECS 2021, October 20-22, 2021, Hong Kong
- [30] Yinyin Hu, Xiaoxia Zhang, Jiao Yang, and Shuai Fu, "A Hybrid Convolutional Neural Network Model Based on Different Evolution for Medical Image Classification," *Engineering Letters*, vol. 30, no.1, pp168-177, 2022
- [31] Chunfeng Wang, Pengpeng Shang, and Lixia Liu, "Improved Artificial Bee Colony Algorithm Guided by Experience," *Engineering Letters*, vol. 30, no.1, pp261-265, 2022
- [32] Rotimi-Williams Bello, Ahmad Sufril Azlan Mohamed, Abdullah Zawawi Talib, Daniel A. Olubummo, and O. Charles Enuma, "Enhanced Deep Learning Framework for Cow Image Segmentation," *IAENG International Journal of Computer Science*, vol. 48, no.4, pp1182-1191, 2021.
- [33] J. Hui, M. Du, X.Ye, Q. Qin, and J. Sui, "Effective building extraction from high-resolution remote sensing images with multitask driven deep neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 786-790, May 2018.
- [34] Md. Abdul Alim Sheikh, Tanmoy Maity, Alok Kole "IRU-Net: An Efficient End-to-End Network for Automatic Building Extraction from Remote Sensing Images," *IEEE Access*, vol. 10, pp. 37811-37828, 2022.