# DeepExt: A Convolution Neural Network for Road Extraction using RGB images captured by UAV

Neelanshi Varia
*Dhirubhai Ambani Institute of Information and Communication Technology*
*Gandhinagar, India 382007*
*neelanshiV2@gmail.com*

Akanksha Dokania
*Department of Electronics and Electrical Engineering,*
*IIT Guwahati*
*Guwahati, India*
*akankshadokania@gmail.com*

J. Senthilnath
*Institute for Infocomm Research*
*A*STAR*
*Singapore-138632*
*senthil.iiscb@gmail.com*

***Abstract*— In this paper, we propose automatic road extraction using Unmanned Aerial Vehicle (UAV) based Remote Sensing data. Road extraction using UAV data is very useful in traffic management, city planning, GPS based applications, etc. Deep learning techniques namely, Fully Convolutional Network (FCN) and conditional Generative Adversarial Networks (GAN) are used to extract roads from a UAV dataset available in the literature. FCN performs semantic segmentation on the image whereas the GAN generates output images from the model it learns. The results demonstrate the efficiency of the deep learning methods for the task of road extraction.**

***Keywords*— *road extraction, convolutional neural network, generative adversarial networks, semantic segmentation***

## I. INTRODUCTION

Ever since the advent of remote sensing technology, numerous areas of applications have been emerging with the help of data provided. Hyperspectral data, Very High Resolution (VHR) satellite images, Light Detection, and Ranging (LiDAR) data, etc. have increased the amount of information with which one can experiment and produce results which weren't possible initially. The imagery obtained from the satellites is not only in the visible spectrum but also available in the infrared, multispectral and hyperspectral range. Especially with the availability of remote sensing images using Unmanned Aerial Vehicle (UAV) and satellites, techniques of image processing, computer vision, machine learning and various statistics can be easily applied. Remote sensing has a wide area of applications including traffic management, environmental monitoring, agricultural and forest mapping, urban area change detection, etc.

In this paper, we are exploring one such application; road extraction using remote sensing satellite images. All the GPS utilities including e-maps, traffic control systems, urban area monitoring, construction planning have a study of roads as the most essential part. Road extraction has been researched since the early years of remote sensing technology using various classical statistical methods to current Deep Learning methods. Road extraction can be considered as a pixel-level classification task into 2 categories, namely, road and non-road. One of the earliest methods proposed a model for the road as a network of intersections to extract road area from images [1]. Road extraction based on geometrical, structural and spectral characteristics of a road [2], locally adaptive spectral-spatial classification along with post-processing based on road features have also been implemented. [3] To extract road centerlines, SVM and multi-scale features [4], [5], [6], [7] and line segmentation [8] have been widely used. Mathematical morphology-based methods along with others have also been explored [9], [10], [11]. As features play a very important role in extraction, methods using neural networks have also been proposed. Some of the earliest methods use Back Propagation [12] which gives better results than some statistical methods. Other Machine Learning methods that use backpropagation [13], Bayesian filtering [14], online learning [15] and Mean-Shift clustering [16] have also been proposed in the past.

Computational power has very well allowed us to explore techniques using Deep Learning to get precise results like never before. A paper implementing GPU based method clearly shows how computational power can improve the net time taken for processing images [17]. Variants and applications of Convolutional Neural Networks (CNNs) are almost impossible to list but some of the major breakthroughs have been made by AlexNet [18], ResNet [19], VGGNet [20], GoogLeNet [21], RCNN [22], FCN [23], etc. DeepLab has set the benchmark for semantic segmentation. [24] These nets use a large number of layers which are basically representative of the low-level and high-level features of the images and hence result in better object extraction. In the past, few deep networks have been implemented for road extraction from satellite images. In the recent years, even Generative Adversarial Networks [25] have gained a lot of popularity. They are a type of Neural Networks that allows one to generate images which are similar to the ones with which the network is trained. And hence it is not just able to generate segmented images but is also able to generate original looking images from the outlines or segmentation provided without any details.

It becomes difficult to extract the exact shape of roads due to occlusions and different types of backgrounds, noise in the form of vehicles on road, etc. For removing such obstacles many methods suggest some sort of pre-processing before implementing the actual method. With neural networks, especially Deep Networks, this task of learning is automated up to a great extent and one majorly has to worry only about providing the right quality and amount of the dataset. A recently proposed method implements Residual layers on the U-Net for road extraction which outperforms just U-net applied for the task. [26] Recently proposed CasNet uses a cascaded end-to-end neural network which is 25 times faster than many of the comparable networks. It consists of two networks – one to extract roads with complex backgrounds and features and the second net focuses on a good centerline extraction. Finally, a thinning algorithm is used to obtain a single pixel width road centerline. [27] RSRCNN has been proposed to extract road based on spatial correlation as well as geometric features of a road. Here they propose a new loss function to incorporate the geometric structure and features of the road. They include

978-1-5386-9276-9/18/$31.00 ©2018 IEEE

convolutional and fusion layers for better feature extraction. [28]

In this paper, we have implemented two deep learning techniques, namely, Fully Convolutional Network (FCN) and Generative Adversarial Network (GAN) for road extraction in diverse types of backgrounds. The FCN-32 variant of the FCN and Conditional GAN was implemented to test UAV remote sensing dataset. The algorithms captured the curvature as well as road intersection efficiently. The performance of the methods is analyzed using different measures and time taken to extract road segment is also analyzed.

Section II discusses the methodologies and architectures implemented for this experiment. Section III gives us the details of the performance measures. In Section IV we discuss the results obtained on various images of the dataset and analysis of road extraction using CNN. The paper is concluded in Section V.

## II. Methodology

Convolutional neural networks and its variants are formed by a pile of layers. Majority of which does feature extraction from the layer immediately before it and hence a greater number of layers helps in much better feature extraction of the data. The weights of these layers are calculated via supervised or semi-supervised learning through neurons. Like the neural networks, we pass it through activation functions and calculate loss for the whole network. Filters are applied over different layers which stack up the layers and reduce the size of an individual layer while down-sampling. Similarly, the learned features are projected during the up-sampling to reach the final output. The aim of the experiment/task was to extract the area of road present in an RGB image obtained. The task was accomplished by training the images on two different convolutional neural networks, namely, Fully Convolutional Network (FCN) and Conditional Generative Adversarial Network (pix2pix) [29].

We evaluated our method on the UAV dataset created by Zhou [30,31,32], which is a large UAV remote sensing dataset composed of RGB imagery with different sizes. The dataset includes training and testing dataset. Dataset was mainly acquired using a UAV that flew in different sessions in Australia. The other image sequences downloaded from the internet were also included in this dataset for more evaluation on different scenarios. In the test dataset, there were 2760 images from six videos with different resolutions, including 1280 × 720, 1244 × 748, 1024 × 576, and 848 × 480. In this study, the number of training dataset used was 189 and 23 images in the test set for evaluation. The images had a variety in terms of shapes, color, orientation, the angle of capture and complexity of the network. It also had a lot of occlusions and noise and hence a good mix of data to train the network on.

### A. Fully Convolutional Network (FCN)

Fully convolutional neural networks were the first to be implemented on images for semantic segmentation and for tasks like these where binary classification has to be done in the semantic segmentation, FCN networks very well capture the features and segment the data. FCNs do not have fully-connected layers at the end but possess a convolutional layer to perform the classification. For this task, we had implemented an FCN-32 network variant. [23] An FCN consists of three types of components – convolution, pooling and activation function. FCNs give an output of the dimension $h \times w \times d$ where $h \times w$ is the dimension of image and $d$ is the number of channels. If the input data point is denoted by $x_{i,j}$ then the output of these layers is given by,

$$y_{i,j} = f_{k,s}(\{X_{si+\delta i,ij+\delta j}\} 0 \leq \delta i, \delta j < k) \quad (1)$$

where $f_{k,s}$ determines the type of the layer. It does matrix multiplication for average pooling layer, takes spatial maximum for max pooling layer and elementwise output for the activation function. Fig. 1 shows the encoder architecture of FCN.

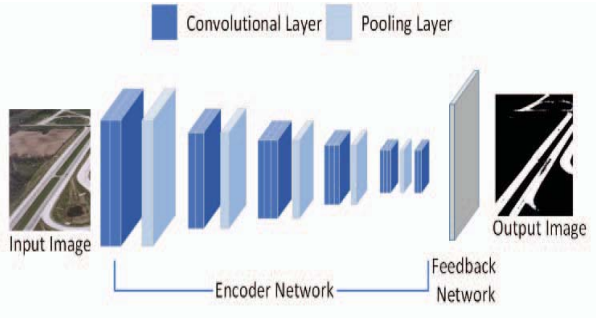



**Fig. 1:** Encoder network of FCN

We used convolution filters of size 3 X 3 with stride 1. Rectified Linear Unit (ReLU) and Leaky ReLU functions were used alternatively for all the layers.

$$\text{ReLU: } f(x) = \max(0, x) \quad (2)$$

$$\text{Leaky ReLU: } f(x) = \begin{cases} 0.01x, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (3)$$

Here the training was done via back propagation to minimize the softmax loss function for individual pixel classification. Skip layers were also added for the task of semantic segmentation to join the coarse output along with local one.

### B. Conditional Generative Adversarial Network (pix2pix)

Generative adversarial networks [25] are models with two frameworks – a generator net and a discriminator net which generate and evaluate an image respectively. The main advantage of GANs over auto-encoders is that they can be moulded the way we want and generate images as per our need. We applied pix2pix method which uses GAN in a conditional setup. Here, the generator part of the model used U-net based architecture. This net uses the concept of skip-layers which can help in avoiding the passing of the low-level information through all the layers of net and directly pass it to the corresponding decoding network. This network when compared with a normal encoder-decoder network without skip layers, gives a better output. A down-sampling stack consists of 2 convolutional layers with filters of size 3 × 3, a ReLU layer followed by a max pooling layer with stride 2; such a stack of layers is repeated. Fig. 2 shows the Generator architecture used in the implementation.

The discriminator network uses "PatchGAN" for discriminating between a real and a fake (generated) image. PatchGAN runs the discriminator at a patch level and hence penalizes only a part of the image if it is found fake.

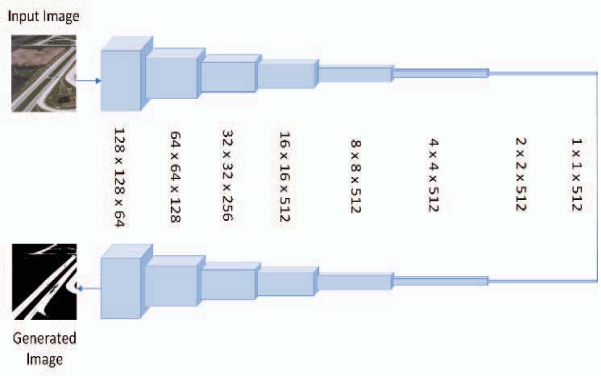



**Fig. 2:** Generator network of conditional GAN

GANs learn a mapping $G: z \rightarrow y$, where $z$ is random noise and $y$ is the output image. The objective function of the Conditional GAN is given by,

$$\mathcal{L}cGAN(G,D) = E_{x,y}[logD(x,y)] + E_{x,y}[log(1 - D(x,G(x,z))] \quad (4)$$

where $x$ is the target output image introduced through which the generator learns to map in cGAN. Fig. 3 shows the flow of GAN.

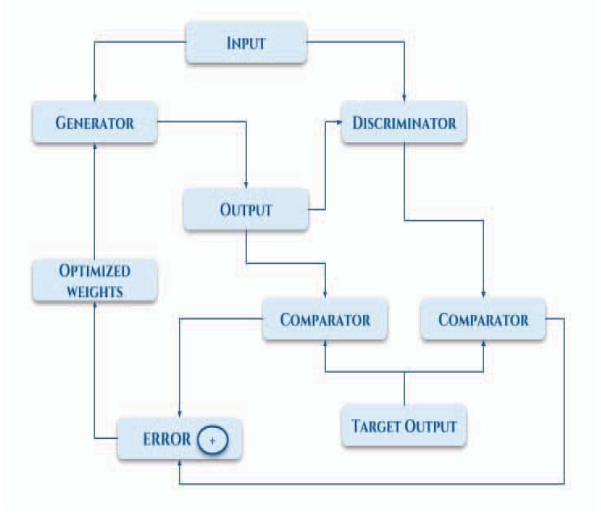



**Fig. 3:** Flow of conditional GAN

Instead of *L2* distance, the method uses *L1* distance as it blurs the image lesser.

$$\mathcal{L}_{L1}(G) = E_{x,y,z}[\|y - G(x,z)\|] \quad (5)$$

That makes the final objective function –

$$G^* = \arg\min_G \max_D \mathcal{L}cGAN(G,D) + \lambda \mathcal{L}_{L1}(G) \quad (6)$$

The generator $G$ was trained to maximize $logD(x,G(x,z))$. Adam solver algorithm was applied to optimize the weights. The momentum parameters for the function were chosen to be 0.5. The learning rate was set to be 0.0002 which decayed after half the number of total epochs (total epochs being 300). The discriminator in this experiment used ResNet architecture [19] with 9 blocks. For both, the Generator and Discriminator network, 64 filters were used and the batch size for training was 2.

## III. Performance Measures

Unlike the regular evaluation matrices which are usually dependent on confusion matrix parameters of the whole image, we used the evaluation metric which focused on the length of the road of the data extracted and not on the area. The confusion matrix evaluation represents the area of the road which is correctly or incorrectly extracted whereas with the completeness, correctness and quality dimensions of the output, one can know the results in terms of length of the road.

*A.* ***Completeness****:* Completeness is the percentage of the correctly extracted data in terms of the reference data. The ideal value for completeness is 1 (or 100% efficiency). The formula for completeness is given by,

$$Completeness = \frac{TP}{(TP + FN)} \quad (7)$$

where True Positive (TP) is the length of the road extracted correctly and the term (TP + FN) represents the actual reference length of the road, where FN (False Negative) represents the length of the actual road which couldn't get extracted.

*B.* ***Correctness****:* Correctness is the percentage of the correctly extracted data with respect to the actual extracted data. The idea value for correctness is 1 (or 100% efficiency).

$$Correctness = \frac{TP}{(TP + FP)} \quad (8)$$

where TP is the length of the road extracted correctly and the term (TP + FP) represents the length of the road extracted by the algorithm, where FP (False Positive) denotes the length of the road extracted which is not road in the given data.

*C.* ***Quality:*** It is the measure of fineness of a data obtained by combining both - completeness and correctness. Again, the ideal value for Quality is 1 (or 100% efficiency). It is defined as,

$$Quality = \frac{TP}{(TP+FP+FN)} \quad (9)$$

## IV. Results and Discussions

This section discusses the results obtained on 5 test images with different complexities. The first image (Fig. 4 (a)) chosen is the simplest one as it has a linear road. In both FCN and pix2pix CNN methods the road segment was extracted quite accurately and the results also reflect them. The completeness, correctness and quality measure for the first image is 1 for both the algorithms.

The second image chosen has a simple curvature. (Fig. 4 (e)) The surrounding of the road segment is in sharp contrast to it and hence it is easier to detect the road. The completeness, correctness and quality measure for the second image is also 1.

The third image (Fig. 4 (i)) has a lesser complexity in terms of the network but has roads of varying width, curvature and angle of elevation. As it can be seen, FCN misclassifies some area of non-road segment to be road segment, whereas pix2pix does not misclassify road segment as a non-road segment. For the same reason, correctness of pix2pix is 98.4% and that of FCN is 90.9%. (Table I)

The fourth image chosen (Fig. 4 (m)) is a mix of linear and curved roads with not so contrasting color and intensity. The orientation of the image and angle makes some of the lanes very thin to be detected. This resulted in large False Negative for both the methods and hence low completeness and low quality of the output.

The fifth image (Fig. 4 (q)) has a very complex road network. The shadow of the bridges is similar to the roads which makes it relatively a complex network to be extracted. The occlusion by trees and vehicles further adds to the complexity. Yet both the algorithms are able to extract it better than the fourth image (Fig. 4 (o, p)).

The images were captured in different areas with various combinations of shapes and surroundings. For validating the results obtained by the models, a completeness, correctness and validity [28] check of the data was performed. The images were resized to 128 X 128 X 3 for training the FCN network. For the Conditional GAN, the images had varied sizes as in the dataset. 189 images, chosen randomly, were used for training and the remaining 23 were used for testing. The experiments were performed on consumer computer with Intel i7 (6th Gen Skylake processor) cores, 4GB AMD Radeon graphic card and 16GBs of RAM. All the experiments were implemented in Python. The training was done without any GPU and took 370 seconds and 300 seconds per image respectively for FCN-32 and pix2pix algorithm. Table II shows the time taken by the algorithms for training.

TABLE I. EVALUATION OF PERFORMANCE MEASURE ON IMAGES

| Image | Method | Completeness | Correctness | Quality |
|---|---|---|---|---|
| 1 | FCN-32 | 1.000 | 1.000 | 1.000 |
| | pix2pix | 1.000 | 1.000 | 1.000 |
| 2 | FCN-32 | 1.000 | 1.000 | 1.000 |
| | pix2pix | 1.000 | 1.000 | 1.000 |
| 3 | FCN-32 | 0.968 | 0.856 | 0.909 |
| | pix2pix | 0.968 | 1.000 | 0.984 |
| 4 | FCN-32 | 0.742 | 0.987 | 0.858 |
| | pix2pix | 0.763 | 1.000 | 0.866 |
| 5 | FCN-32 | 0.922 | 0.989 | 0.954 |
| | pix2pix | 0.892 | 1.000 | 0.943 |

TABLE II. EXECUTION TIME FOR TRAINING DATASET

| Method | Avg. training time per image | Total number of images | Total training time |
|---|---|---|---|
| FCN-32 | ~370 s | 189 | ~20 hrs |
| pix2pix | ~300 s | 189 | ~16 hrs |

## V CONCLUSION

In this paper, we addressed automatic road extraction using Deep Learning techniques, namely, FCN and pix2pix. On an average, as it was observed that pix2pix performed better than FCN-32 in terms of Quality for extracting intersected roads. Also, the completeness of road obtained by pix2pix is 100% for all the images whereas for FCN it varies from 85.6% to 100%. It can be observed that FCN extracts extra areas (which include both road and non-road features) which pix2pix does not. It is helpful to use FCN when the network of the roads is sort of scattered as it usually does not miss those parts of the road. We can further extend this work by including a greater variety and number of images so as to better train the FCN as well as GAN. Both of the algorithms are highly dependent on the types and quantity of images fed to the network. Along with augmentation, the number of images can be increased as a part of future work. Images have occlusion of roads by trees, vehicles, etc. can be added in a larger number as well as images having other objects visually more salient than roads to train a better network. The FCN-8 variant is proven to give better segmentation results than FCN-32 [23]. It gives more precise segmentation than FCN-32 but for this task, the centerline of a road is more important and hence FCN-32 suffices. To sum, GAN (pix2pix) and CNN (FCN-32) were implemented for extraction of road areas from the images. The output's efficiency was measured using completeness, correctness, and quality.

## ACKNOWLEDGEMENT

For training the conditional GAN, we used the code available on https://github.com/phillipi/pix2pix [29]. We used Image Labeler app of MATLAB for labeling the training data.

## REFERENCES

[1] A. Baumgartner, C. Steger, H. Mayer, W. Eckstein, & H. Ebner, "Automatic road extraction based on multi-scale, grouping, and context." *Photogrammetric Engineering and Remote Sensing*, vol. 65, 777-786, 1999.

[2] J. Senthilnath, M. Rajeshwari, and S.N. Omkar, S.N. "Automatic road extraction using high resolution satellite image based on texture progressive analysis and normalized cut method." *Journal of the Indian Society of Remote Sensing,* vol. 37, no. 3, pp.351-361, 2009.

[3] W. Shi, Z. Miao, & J. Debayle. "An integrated method for urban main-road centerline extraction from optical remotely sensed imagery." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 6, 3359-3372, 2014.

[4] X. Huang, & L. Zhang, "Road centreline extraction from high resolution imagery based on multiscale structural features and support vector machines." *International Journal of Remote Sensing*, vol. 30, no. 8, 1977-1987, 2009.

[5] M. Song, & D. Civco, "Road extraction using SVM and image segmentation." *Photogrammetric Engineering & Remote Sensing*, vol. 70, no. 12, 1365-1371, 2004.

[6] H. Zhou, H. Kong, L. Wei, D. Creighton, & S. Nahavandi, "On detecting road regions in a single UAV image." *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, 1713-1722, 2017.

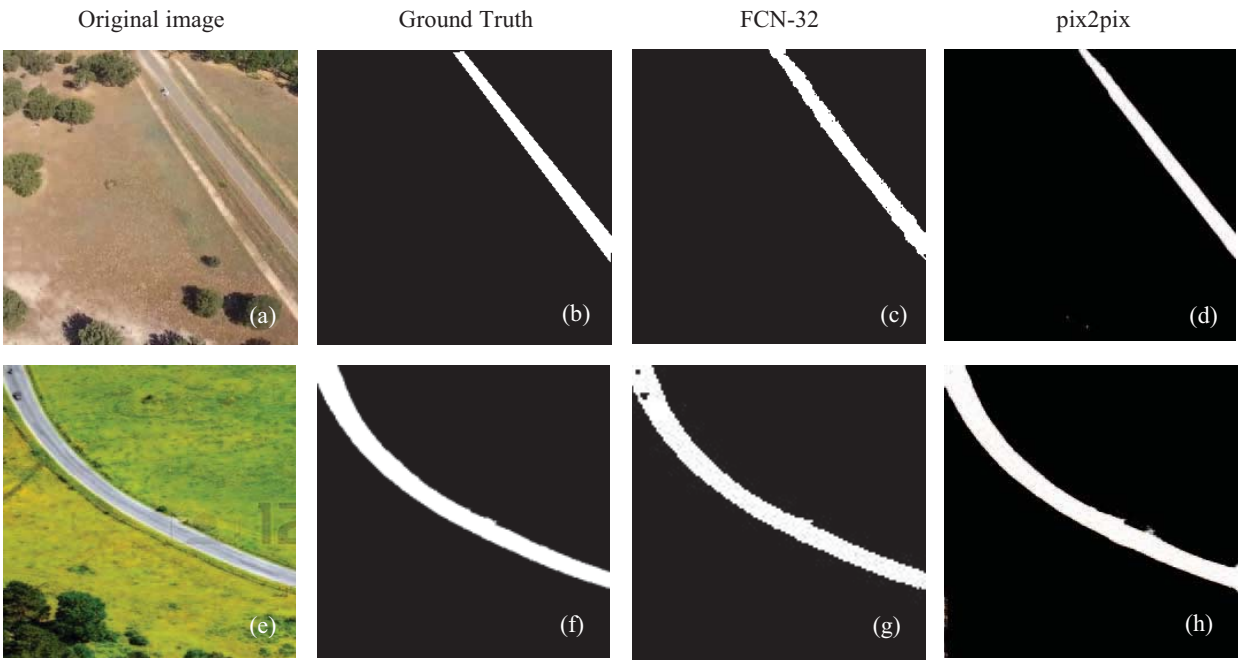
[7] N. Yager, & A. Sowmya, "Support vector machines for road extraction from remotely sensed images." In International Conference on Computer Analysis of Images and Patterns (pp. 285-292). Springer, Berlin, Heidelberg, Aug 2003.

[8] W. Shi, & C. Zhu, "The line segment match method for extracting road network from high-resolution satellite images." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 2, pp.511-514, 2002.

[9] C. Zhang, S. Murai, & E.P. Baltsavias. "Road network detection by mathematical morphology." In ISPRS Workshop" 3D Geospatial Data Production: Meeting Application Requirements. Institute of Geodesy and Photogrammetry, ETH-Hoenggerberg, 1991.

[10] A. Mohammadzadeh, A. Tavakoli, & M.J. Valadan Zoej, "Road extraction based on fuzzy logic and mathematical morphology from pan

sharpened ikonos images." *The photogrammetric record*, vol. 21, no. 113, pp.44-60, 2006.
[11] S. Valero, J. Chanussot, J.A. Benediktsson, H. Talbot, & B. Waske, "Advanced directional mathematical morphology for the detection of the road network in very high resolution remote sensing images." *Pattern Recognition Letters*, vol. 31, no. 10, pp.1120-1127, 2010.
[12] P.D. Heermann, & N. Khazenie, "Classification of multispectral remote sensing data using a back-propagation neural network." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 30, no. 1, pp.81-88, 1992.
[13] M. Mokhtarzade, & M.V. Zoej, " Road detection from high-resolution satellite images using artificial neural networks." *International journal of applied earth observation and geoinformation,* vol. 9, no. 1, pp.32-40, 2007.
[14] J. Zhou, W.F. Bischof, & T. Caelli. "Road tracking in aerial images based on human–computer interaction and Bayesian filtering." *ISPRS journal of photogrammetry and remote sensing*, vol. 61, no. 2, pp.108-124, 2006.
[15] S. Zhou, J. Gong, G. Xiong, H. Chen, & K. Iagnemma, "Road detection using support vector machine based on online learning and evaluation." In Intelligent Vehicles Symposium (IV), 2010 IEEE (pp. 256-261). IEEE, Jun 2006.
[16] M. Rajeswari, K.S. Gurumurthy, S.N. Omkar, J. Senthilnath, and L.P. Reddy, "Automatic road extraction using high resolution satellite images based on level set and mean shift methods." In Electronics Computer Technology (ICECT), 2011 3rd International Conference on (Vol. 2, pp. 424-428). IEEE, Apr 2011.
[17] J. Senthilnath, S. Sindhu, and S.N. Omkar, " GPU-based normalized cuts for road extraction using satellite imagery." *journal of earth system science*, vol. 123, no. 8, pp.1759-1769, 2014.
[18] A. Krizhevsky, I. Sutskever, & G.E. Hinton, " Imagenet classification with deep convolutional neural networks." *In Advances in neural information processing systems (*pp. 1097-1105), 2012.
[19] K. He, X. Zhang, S. Ren, & J. Sun, "Deep residual learning for image recognition." *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778), 2016.
[20] K. Simonyan, & A. Zisserman, "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556, 2014.
[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, ... & A. Rabinovich, "Going deeper with convolutions." In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9), 2015.
[22] R. Girshick, J. Donahue, T. Darrell, & J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587), 2014.
[23] J. Long, E. Shelhamer, & T. Darrell, "Fully convolutional networks for semantic segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440), 2015.
[24] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, & A.L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp.834-848, 2018.
[25] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, ... & Y. Bengio, "Generative adversarial nets." *In Advances in neural information processing systems* (pp. 2672-2680), 2014.
[26] Z. Zhang, Q. Liu, & Y. Wang, (2018). "Road extraction by deep residual u-net." *IEEE Geoscience and Remote Sensing Letters.*
[27] G. Cheng, Y. Wang, S. Xu, H. Wang, S. Xiang, & C. Pan, "Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network." *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 6, 3322-3337, 2017.
[28] Y. Wei, Z. Wang, & M. Xu, "Road Structure Refined CNN for Road Extraction in Aerial Image." *IEEE Geosci. Remote Sensing Lett.,* vol. 14, no. 5, 709-713, 2017.
[29] P. Isola, J.Y. Zhu, T. Zhou, & A.A. Efros, "Image-to-image translation with conditional adversarial networks." arXiv preprint, 2017.
[30] H. Zhou, H. Kong, L. Wei, D. Creighton, and S. Nahavandi, "Efficient road detection and tracking for unmanned aerial vehicle." *IEEE transactions on intelligent transportation systems*, *16*(1), pp.297-309, 2015.
[31]https://www.dropbox.com/sh/w8e3a8j5eksfi7o/AADIqsM8Uy7XrceceR6x8NFoa?dl=0 (accessed on 8/7/2018)
[32]https://www.dropbox.com/sh/99cbjs6v73211fk/AABlmOeaPY6NAKUykqAc_E2ra (accessed on 8/7/2018)



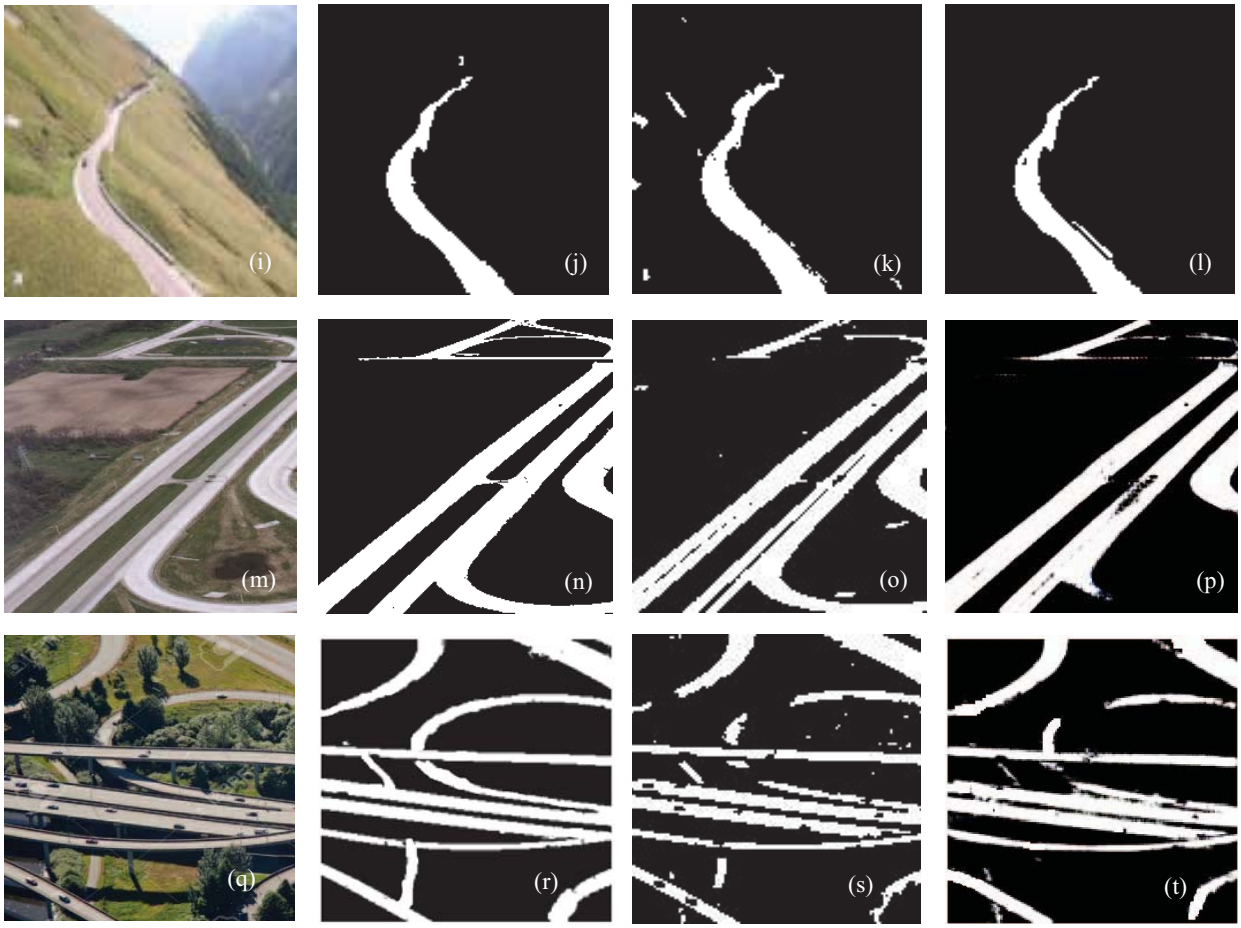



**Fig. 4**: Results of FCN-32 and pix2pix on 5 images taken from the dataset