OBJECT EXTRACTION FROM REMOTELY SENSED AERIAL IMAGES

Thesis

Submitted in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

by
KARUNA KUMARI EERAPU

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING,
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA,
SURATHKAL, MANGALORE -575025

January, 2021

DECLARATION

I hereby ***declare*** that the research Thesis entitled **Object Extraction from Remotely Sensed Aerial Images** which is being submitted to the ***National Institute of Technology Karnataka, Surathkal*** in partial fulfillment of the requirement for the award of the Degree of ***Doctor of Philosophy*** in **Department of Electronics and Communication Engineering** is a ***bonafide report of the research work carried out by me***. The material contained in this Research Thesis has not been submitted to any University or Institution for the award of any degree.

25/01/2021

Karuna Kumari Eerapu,
Reg.No: 148049EC14FV11
Department of Electronics and Communication Engineering.

Place: NITK-Surathkal.
Date:

CERTIFICATE

This is to certify that the Research Thesis entitled **Object Extraction from Remotely Sensed Aerial Images** submitted by **KARUNA KUMARI EERAPU**(Register Number: 148049EC14FV11) as the record of the research work carried out by her, is accepted as the *Research Thesis submission* in partial fulfilment of the requirements for the award of degree of **Doctor of Philosophy**.

Dr. Shyam Lal
Research Guide,
Assistant Professor,
Dept. of Electronics and Communication Engg.,
NITK Surathkal-575025.

25/01/21

Dr. A V Narasimhadhan
Research Guide,
Assistant Professor,
Dept. of Electronics and Communication Engg.,
NITK Surathkal-575025.

25/01/21

I/C **Chairman-DRPC**
(Signature with Date and Seal)

प्राध्यापक एवं विभागाध्यक्ष / PROF & HEAD
ई एवं सी विभाग / E & C Department
एन आई टी के, सुरतकल/N I T K, Surathkal
मंगलूर / MANGALORE - 575 025

Acknowledgements

This thesis is the result of the effort and unwavering support of many people to whom I am incredibly thankful. Most importantly, I am very grateful to my supervisors Dr. Shyam Lal and Dr. A.V Narasimhadhan, for providing me a platform for learning and helping me emotionally and morally throughout this journey. It is my great privilege to work with you all. I am willing to work with both of you in the near future.

Dr. Shyam Lal, I always feel thankful for your close guidance, insightful discussions while carrying out research, and having confidence in me. I am very much impressed with your positive energy and understanding attitude. You made me realize that "nothing can stop from learning if you are determined." "the importance of consistency, and tuning to recent advancements in the research." One uttered word by you, "Do not worry," acted as the power booster during my research journey. I feel blessed to be under the shower of your grace. I am deeply expressing my appreciation towards you for everything, and I promise that I will always be responsible for you.

Dr. Narasimhadhan, thanks for your inspiration and supervision throughout the journey to finish the thesis. I feel grateful for your attention and warm support when I needed the most. I learned a lot from you and am deeply indebted to you for providing valuable life lessons that made me go further professionally and personally. I am fortunate to have your esteemed guidance and for showing more faith in me than I deserved.

I sincerely thank Dr. Raghavendra B.S. and Dr. Jidesh P., Research Progress Assessment Committee (RPAC) members, for their evaluation and motivation. Both have contributed significantly by providing valuable suggestions to uplift the content and quality of the research. I pay special gratitude to the former Head, Prof. U. Shripathi Acharya, and the present Head of the Department of Electronics and Communication Engineering, Prof. T. Laxminidhi, for their precious advice and administrative support.

Thanks to Balraj Ashwath for collaborative work and fruitful discussions. It was an excellent opportunity to gain knowledge by working with a young mind like him. I wish to acknowledge the useful critics and reviews provided by Prof. Fabio Dell' Acqua.

Special thanks to Sravani K and her family members for excellent care, providing shelter at times, and offering enormous help when I badly needed. Hats off to them for having a generous heart to treat me as their family member. I thank all my fellow research scholars (Asha C S, Jayaram Reddy M K, Shareef Babu Kalluri, Shilpa Suresh, Ragesh Rajan M, Sruthakirthi Godkhindi, Ramavath Prasad Naik, and Shara Mathew) at NITK for their help, support, motivation, and friendship.

My mother, Sugunamma Eerapu, and my father, Subbarayudu Eerapu, are the visible Gods to me. Nothing is a substitute for their love, care, support in all my pursuits. I pay the highest regard to my parents for their unconditional affection, sacrifices, and support. I thank my sister, Nipuna sri Eerapu, brother-in-law, Rajagopal, and my brother, Teja for their support during this journey.

I express my warm gratitude to my loving husband, Lakshmaiah Medampudi, who is so understanding, loving, encouraging, and have tonnes of patience. I recognize your sacrifice of staying far from us and faithful support to me during this journey. I thank God for blessing me with your presence. My son, Sai Pradhyumn Medampudi, is a great joy and cheerfulness to me in this journey. I appreciate his understanding and the sacrifice of going to school at an early age. Words can not express how grateful I am to my mother-in-law, Nagendramma, who had come from a long way to stay with us during my Ph. D. journey. I thank Sumathi aunty, who cooked and served food for us with a pure heart. I sincerely acknowledge Sahana and her colleagues for taking care of my son during the daytime, which is the primary source for my studies.

I thank my God, Sri Sai Ram, for arranging everything timely and providing me with a good support system both academically and personally. Without His will and grace, I am nothing. I thank Him for offering me the courage to go through this journey and answering my prayers. I bow my head for everything you have done to me, and I say thank you.

Dedicated to

Lord Sri Sai Ram.

&

My Family, My Husband (Lakshmaiah Medampudi), and My Son (Sai Pradhyumn Medampudi)

Abstract

The topographical map of the Earth is recorded by capturing high spatial resolution (HSR) aerial images from higher altitudes using Aircraft, Helicopters, and Unmanned Aerial Vehicles (UAVs). The object extraction from HSR remotely sensed aerial imagery data is a prerequisite in a large number of applications such as planning urban cities, accessing disasters, managing traffic congestion, and providing up-to-date road maps. The development of automated methods to extract objects accurately is highly required for the applications, as mentioned above. However, this is a notoriously challenging task in the field of remote sensing. The deep learning field has gained massive interest due to its ability to learn after the availability of high-volume data and computational resources. This thesis investigates evolutionary optimization based framework for quality enhancement of remotely sensed aerial images and various Convolutional Neural Network (CNN) based approaches of deep learning and proposes an enhancement and object segmentation techniques for HSR remotely sensed aerial imagery data.

In the first part of the thesis, the contrast enhancement technique to improve the visual quality of remotely sensed aerial images is presented. The visual quality of captured aerial images is impaired due to the atmospheric effects and limitations of sensors. The visual quality of aerial images needs to improve to extract the hidden object details by increasing the pixel intensity ranges. Most of the techniques in the literature do not consider multi-objective function optimization, either computationally complex or less stable. There is a high demand to introduce a framework to find stable optimum values for multi-objective function with reduced computational complexity. The new framework is introduced to restore the visual quality of images by adjusting saturation, color values, and finally enhanced the contrast of the images through Particle Swarm Optimization (PSO). The experimental and visual quality results showed that the proposed framework outperformed over other state-of-the-art quality restoration techniques.

Next, in the thesis, two deep learning-based semantic segmentation architectures are introduced to extract diversified objects from aerial images. The deep learning-based approaches achieved better results as compared with conventional techniques. The conventional techniques involve manual feature extraction in multiple stages, which demands to maintain the accuracy of each stage to get overall high accuracy. On the other hand, the Convolutional Neural Network (CNN) based architectures in deep learning field provide object segmentation by learning the image features from a higher volume of imagery data. Plentiful of semantic segmentation architectures are present in the literature to perform object segmentation. However, there is still scope to improve accuracy while segmenting small objects in high-resolution aerial images.

In aerial images, objects appear irregularly shaped and tiny in size and also present in dominant background scenarios. Further, class-wise pixels and background pixels are in the ratio of ones-to-tens, which leads to class imbalance problem. Therefore it is very challenging to obtain better prediction accuracy and completeness by preserving the connectivity without any gap between successive objects. In this thesis, a Dense Refinement Residual Network (DRR Net) is proposed for road extraction from aerial imagery data. The DRR Net is introduced based on dense convolutions for feature learning, residual connections to guide the learning path, and provides refinement through the stacking of DRR modules in the network. In the final part of the thesis, the robust encoder and decoder architecture, namely O-SegNet for objects segmentation from high-resolution aerial imagery data, is introduced. The O-SegNet provides emphasis to relevant object details and extracts global context through the self-attention mechanism and multi-level pooling. Both of these proposed architectures are trained with composite loss function to focus more on small object instances. The proposed semantic segmentation architectures have achieved significant quantitative and qualitative results compared with the other existing semantic segmentation architectures.

Keywords: Aerial images, Contrast enhancement, PSO algorithm, Framework, Segmentation, Dense convolutions, Dense blocks, DRR Net, IOU, Loss function, Residual connections, Self-Attention, and GA blocks.

Contents

List of Figures

List of Tables

Nomenclature

Symbol	Meaning	Symbol	Meaning
I	Image	K	Kernel
f	filter size	W, H	Width and height of the image
p	amount of padding	L	Number of layers in network
N	Number of pixels in the image	Y_i	pixel values in the ground
Y_i	predicted probabilities of the class-wise pixels	$log(p)$	logarithmic of probability
L_{BCE}	Binary Cross Entropy loss	L_{LZS}	Lovasz softmax loss
$L_{Composite}$	Composite loss function	Q_L	local quality measure
Q_G	global quality measure	γ	weight factor
L_{max}	Maximum luminance	L_{min}	minimum luminance
p_i	probabilities of outcomes	k_1, k_2	image blocks
I_{max}	Maximum pixel intensities	I_{min}	Minimum pixel intensities
M	gray scale range	P_R	predicted road pixels
P_B	predicted building pixels	G_R	road pixels in the ground truth
G_B	building pixels in the ground truth	R_{Accu}	Road Accuracy
B_{Accu}	Building Accuracy	N_C	Number of classes
C_{ii}	Class-wise predicted pixels	T_i	pixels in the ground truth
P_i	Number of pixels whose predictions are i	T_P	True positives
F_P	False positives	F_N	False negatives
(i, j)	pixel indices	S_M^C	mean brightness value of the image
S_M	minimum mean brightness value	k^C	exponent of each color
min	minimum value	max	maximum value
β	minimum color value of pixels	χ	medium color value of pixels
δ	maximum color value of pixels	μ	scaling parameter
ζ	kernal	a	filter element
$\triangle$	$3 * 3$ neighbourhood	s	parameter
T_j	Number of particles in PSO	G	Number of iterations
T_g	Global best solution	v	Velocity vector
η	inertia constant	c_g, c_q	acceleration constants
C_{max}	maximum constant value	f_{obj}	Objective function
ϵ	Entropy	σ	Number of over ranged pixels
X	Edge signal	F	number of filters used in ICU
X_i	Initial features	$X_{i,j}$	Features from Dense blocks
Y_i	predictions	$X'_{i,j}$	Upsampled features in DRR module
H	Sequence of BN, ReLU, CONV operations	F	Transposed convolution operation
F'	Non linear operation due to $1 * 1$ convolutions	β_1	First order momentum
β_2	second order momentum	E_c	Vector of errors
$\nabla \bar{J}_c$	Loss Surrogate of Jaccard index		
f	resolution	C	Number of $1 * 1$ convolutions
F	factor	T	Transpose operation
σ	softmax operation	$\otimes$	Element-wise matrix operation
γ	attention scores or weights	X_k	Formulated features
X_{jA}	output feature representation of j^{th} GA block	X_{jA}^i	Features learned at i^{th} convolutional layer of j^{th} GA block
v_1	Value features	$/$	pooling operation
X_{3AP}	Features from PPN network	Y_i	Upsampled features
Y_{jA}	output feature representation of j^{th} GA block of the decoder	$:$	concatenation operation

Abbreviations

Abbreviation	Expansion
HSR	High Spatial Resolution
UAV	Unmanned Aerial Vehicle
GPS	Global Positioning System
CONV	Convolution
ReLU	Rectified Linear Unit
CNN	Convolutional Neural Network
PSO	Particle Swarm Optimization
BCE	Binary Cross Entropy
LZS	Lovasz Softmax loss
DRR	Dense Refinement Residual
IOU	Intersection Over Union
GA	Genetic Algorithm
DWT	Discrete Wavelet Transform
SVD	Singular Value Decomposition
HE	Histogram Equalization
SA	Simulated Annealing
NSA	Negative Selection Algorithm
MDE	Modified Differential Evolution
GMSR	Gradient Magnitude based Support Regions
FAST	Features from Accelerated Segment Test
PDF	Probabilistic Density Function
CRF	Conditional Random Field
LSE	Level Set Evolution
ZLC	Zero Level Curve
VGG	Visual Geometry Group
SPP	Spatial Pyramid Pooling
ASPP	Atrous Spatial Pyramid Pooling
FoV	Field of View
D2S	Depth to Space
FCN	Fully Convolutional Network
DSM	Digital Surface Model
NMT	Neural Machine Translation
CT	Computed Tomography
PAM	Position Attention Module
CAM	Channel Attention Module
PSA	Point-wise Spatial Attention
SE	Squeeze and Excitation
AG	Attention Gated
GAN	Generative Adversarial Network

Abbreviation	Expansion
NIQMC	No reference image quality metric for contrast distortion
BIQMC	Blind image quality measure of enhanced images
DE	Discrete Entropy
EME	Measure of Enhancement
PIXDIST	Pixel Distance
CS	Color Saturation
HSI	Hue Saturation Intensity
HSV	Hue Saturation Value
UMF	Unsharp Masking Filter
DB	Dense Block
ICU	Initial Convolution Unit
FLOP	Floating Point Operation
PPL	Pyramid Pooling Layer
GCN	Global Convolutional Network
GA	Guided Attention
SA	Self-Attention
BN	Batch Normalization
PPN	Pyramid Pooling Network
RC	Residual Connection
AD	Attention between Encoder and Decoder

Chapter 1

INTRODUCTION

The Earth's topographical map is obtained by capturing high-resolution aerial images using Aircraft, Helicopters, Unmanned Aerial Vehicles (UAVs), etc. Due to turbulence in the atmosphere, the acquired aerial image's visual quality is impaired. These atmospheric effects should be eliminated to enhance the visual quality of high-resolution aerial images. Locating the presence of the Earth's geographical features in the enhanced aerial images is essential for a variety of geospatial applications. Among all features, roads and buildings are useful for applications like forecasting urban growth, disaster management, traffic management, and map updating. Moreover, the information regarding roads and buildings serves as a basis to update maps in Google Earth and Global Positioning System (GPS)-based navigation devices. In high-resolution aerial images, roads and buildings do not possess a continuous regular shape, and also appear tiny as they occupy the small number of pixels in the image. Further, inter and intra class variations are present in aerial imagery data. Due to this, segmentation ambiguities are produced while performing segmentation. Hence, objects-segmentation from aerial imagery data is a challenging problem in the field of computer vision.

In the literature, plenty of conventional and machine learning-based approaches are present to segment objects from aerial images. In machine learning based techniques, features are determined manually in multiple stages. However, the techniques based on Convolutional Neural Networks (CNN) learn the features automatically from the high volume of input data. The CNN based approaches are utilized to segment roads and buildings through the semantic segmentation task in which every pixel of the image is classified according to respective labels.

The visual quality of aerial images is enhanced in the first work by introducing a framework using Particle Swarm Optimization (PSO). In the second work, the semantic segmentation architecture, namely Dense Refinement Residual (DRR) network, is proposed to segment roads. The DRR network is introduced based on dense convolutions to focus on small details of the aerial image. Further, in order to extract multiple objects from the aerial images, novel O-SegNet architecture is proposed. In this architecture, the attention mechanism is incorporated in the proposed O-SegNet architecture to provide emphasis on the intended parts of the image during the segmentation of roads and buildings.

1.1 Motivations

With the recent advancements in remote-sensing technology and data processing approaches, remote sensing data can provide accurate ground information for various applications like monitoring roads, surveying, transportation, mapping, etc. However, remote-sensing data usually contain heterogeneous objects of varying shapes with inter and intra-class variations. Therefore, it is very challenging to obtain high-precision object-related information from high spatial resolution remotely sensed images with object extraction techniques. The methods to segment objects in remotely sensed aerial images are mainly divided into two broad categories: Traditional/ Conventional methods and Deep learning approaches.

In traditional techniques, objects were extracted in more than two stages by calculating image features in an unsupervised way. Further, the image features are computed from the smaller context of input aerial imagery data. Consequently, the accuracy of the conventional object extraction technique depends on the previous stage outputs, and the accuracy of each stage should be maintained at a reasonable value to get overall high accuracy. Otherwise, the final predictions are not satisfactory to the real-time applications of remote sensing.

On the other hand, by incorporating CNN based approaches for object segmentation from aerial images, the image features are learned from the broad context of the large volume of the data to provide correct predictions. However, techniques based on CNN's involve spatial resolution reduction operations that tend to lose fine details in the aerial images. The other approaches demand post-processing operations for filling holes formed in the predictions and are not well suited for the segmentation of small object instances.

Though numerous semantic segmentation architectures are present in the literature for object extraction from remote-sensing aerial images, segmentation accuracy is still not promising, necessitates the accurate object extraction technique. The efficiency of different types of convolutions, connections for learning of distinct image features, benefits of operating at full image resolution, and attention mechanisms to focus on relevant image context for semantic segmentation motivate the exploration of new techniques for object extraction from aerial imagery data.

1.2 Extraction of Objects from Aerial Imagery Data

The extraction of objects from high-resolution aerial images is obtained by considering processed images. These processed images, along with labels, are fed into the deep Convolutional Neural Network (CNN) model, where the model has been trained to produce the segmented result. Once the training of the model finishes, it has been tested for its effectiveness in the testing phase. The functional diagram of aerial image segmentation is depicted in Figure. 1.1.

1.2.1 Functional Diagram of Aerial Image Segmentation

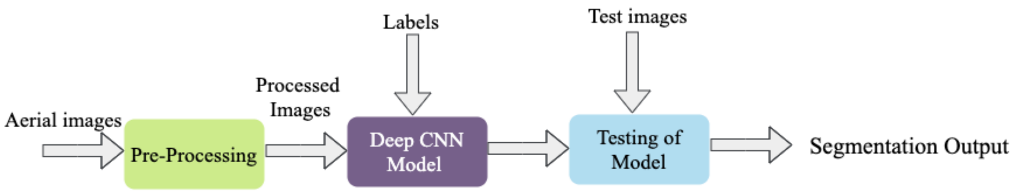


Figure 1.1: Functional Diagram for Objects Segmentation from Remotely Sensed Aerial Images

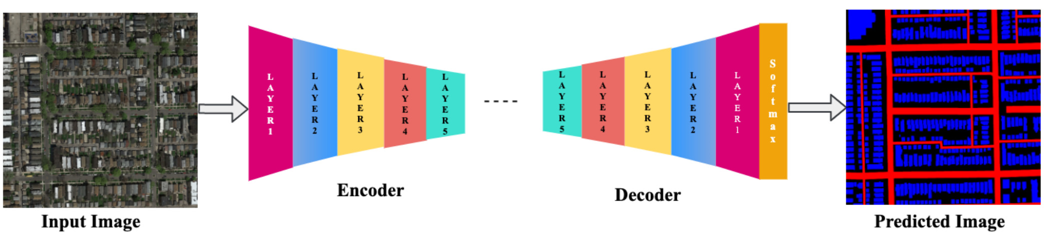


Figure 1.2: General Schematic Architecture for Semantic Segmentation.

1.2.2 Pre-processing of Aerial Imagery

The appearance of observed aerial images is not satisfactory because of limitations in sensors and the thermal noise during image capturing. Further, the aerial images acquired during bad atmospheric and low light conditions suffer from poor contrast problem. So the contrast of aerial images need to be improved to enhance the perceptibility before applying any segmentation method. Hence, the primary objective of pre-processing is to enhance the visual quality of aerial imagery data for image analysis tasks like classification, object segmentation, and object recognition, etc.

1.2.3 Deep CNN Model

The pre processed aerial images are provided as input to the deep CNN model, where the model learns to produce segmentation maps through a process of semantic segmentation in a supervised way. In this process, intended objects are segmented from others by considering pre-processed aerial images along with target labels. The deep CNN based model/architecture is utilized to provide segmentation, and its structural diagram is presented in Figure 1.2. An increased number of convolutional layers are utilized in encoder and decoder of the deep CNN model to produce segmentation maps with higher levels of accuracy. The image features that are necessary to obtain the pixel-wise predictions for each class of input image are learned during encoding-decoding processes. The detailed explanation about the semantic segmentation and its structural diagram is presented in the Section 1.2.5.

1.2.4 Testing of the Model

In this Section, the effectiveness of the deep CNN model is evaluated by providing unseen test images as input. During testing, the learned weights of the trained deep CNN model are loaded to produce segmentation maps of the provided test aerial images.

1.2.5 Semantic Segmentation

Semantic segmentation is the process of classifying every pixel in the image as belonging to one of several classes (roads, buildings, or background). In general, the semantic segmentation architecture can be broadly thought of as an encoder network followed by a decoder network, as shown in Figure 1.2. The encoder structure resembles a classification network. The encoder consists of several layers in which each layer comprises of Convolution (CONV), Batch Normalization(BN), Rectified Linear Unit (ReLU) activation, and max-pooling layers [LeCun *et al.* (1995)]. Different filters incorporated at each convolutional layer of the encoder to learn various distinct features (edges, corners, etc.) from an input aerial image. The learned discriminative image features from the encoder are of lower resolution. In the deep layers of the network, the higher-level features are learned from lower-level and mid-level features. These higher-level features are required to obtain rich semantic information to distinguish various classes in the image. The decoder contains the same layers as the encoder, but max-pooling layers are replaced with up-sampling layers. The higher-level features of the encoder are employed to produce predicted image through the reconstruction process by projecting into the pixel space. The features are reconstructed to produce predicted image through transposed convolutions in the up-sampling path of the decoder. The loss between the predicted image and ground truth and derivative of loss value with respect to filter weights are determined. These values are back propagated to update the parameters of the deep CNN architecture. In this way, the semantic segmentation architecture learns to produce predictions from the input aerial imagery data irrespective of location and translation variance.

1.3 Types of Convolutions Used in Encoder

Lecun *et al.* introduced Convolutional Neural Networks (CNNs) for processing of different types of data such as time-series data and image data [LeCun *et al.* (1995)]. CNN's are a special kind of Neural Networks which employ a convolution operation. CNN's are utilized in numerous applications; some of them are classification, detection, and extraction of objects in images, automatic handwritten recognition, text and image caption generation, etc. There are different types of convolutions present in the literature, which are discussed below.

1.3.1 Normal Convolutions

The convolution operation is performed by considering image data (I) and kernel (K) or filter as inputs and produces a output referred as feature maps [LeCun *et al.* (1995), Alpaydin (2020)]. The kernel has learnable weights and biases and its size can be variable.
The mathematical expression to represent $2D$ convolution operation is expressed in Eq.

(1.1)

$$y(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) = \sum_m \sum_n K(m,n)I(i-m, j-n) \qquad (1.1)$$

The diagram to represent convolution operation is show in Figure 1.3

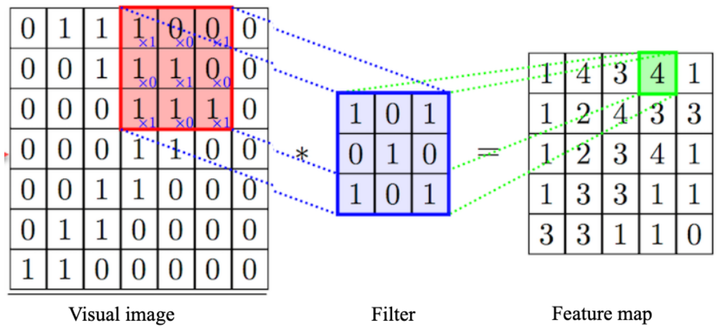


Figure 1.3: Diagram to represent Convolution Operation

The CNN's typically provide *sparse connectivity* between image(s) and kernel(s) as it is performed by considering only some pixels of image and weights of filter. Each member of kernel is used at every other position of image means convolutional neural networks possesses the ability of *parameter sharing*. The size of feature maps after convolution operation is $\frac{W-f+2p}{s} + 1$, where W represents height or width of image, f represents filter size, p refers to amount of padding used and s is stride , the rate at which filter is stride over the input image. In deep learning libraries, there are two types of convolution operations namely *valid* and *same* convolutions. In same convolutions type, image is padded with zeros to produce feature map sizes same as input. The number of zeros padded (p) in the case of *same* convolutions is equal to $(filter\_size(f) - 1)/2$. when f is odd [LeCun *et al.* (1995), Alpaydin (2020)]. In each layer of Convolutional Neural Networks consists of three stages. In first stage, several convolution layers are applied to produce image features. A set of non linear activation functions such as Rectified Linear Unit (ReLU) and its variations are applied after convolution operation. The different pooling operations are employed to produce *invariant* feature maps by extracting minimum or average values from activation maps.

1.3.2 Dense Convolutions

Huang *et al.* (2017) introduced a new type of convolutions named as Dense convolutions based on observation of *ease of training the model if shorter connections are provided between the layers close to input and output.*. Each convolutional layer operates on all of its subsequent layer outputs, totally having $\frac{L\,(L+1)}{2}$ number of connections in a L layer Network. The variation in inputs of particular layer is increased by concatenating its subsequent feature maps which leads to improved efficiency.
The major strengths of Dense Convolutions are :

1. Increased potentiality of network by re using of feature maps.

2. There is no scope for gradient vanishing/exploding problem.

3. The strength of feature maps is increased through repetitive utilization.

4. Reduced number of model parameters.

1.4 Up Sampling Techniques

The spatial resolution of input is recovered via up-sampling techniques which is a very useful phenomenon in dense prediction. The prediction of class labels for every pixels from higher level feature representation is done after retaining the spatial resolution of input. So the way of up sampling the coarse feature maps plays an important role in semantic segmentation. The below mentioned are the various up-sampling approaches present in the literature are presented in the following Section.

1.4.1 Bilinear Up-sampling

Bilinear up-sampling assigns the pixel intensity values when exact pixel matching is not possible. It calculates the up sampled pixel values by considering 4 diagonal pixel values in 2×2 neighbourhood. The operation of Bilinear up-sampling is represented in the Figure1.4 [$source : https : //en.wikipedia.org/wiki/Bilinear\_interpolation$] The coordinates of the unknown point (P) are interpolated from the known values which are represented in Red

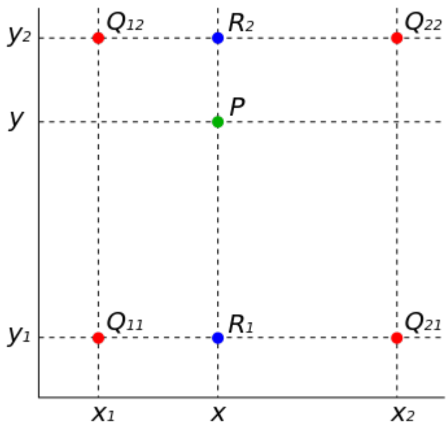


Figure 1.4: Bilinear Up-sampling/interpolation Operation

color using Eq. (1.2).

$$\left.\begin{array}{lll}(x, y_1) & = \frac{x_2-x}{x_2-x_1}\,(x_1, y_1) + \frac{x-x_1}{x_2-x_1}\,(x_2, y_1) \\ (x, y_2) & = \frac{x_2-x}{x_2-x_1}\,(x_1, y_2) + \frac{x-x_1}{x_2-x_1}\,(x_2, y_2) \\ (x, y) & = \frac{y_2-y}{y_2-y_1}\,(x, y_1) + \frac{y-y_1}{y_2-y_1}\,(x, y_2)\end{array}\right\} \tag{1.2}$$

The drawbacks of Bilinear Upsampling are :

1. The way of up sampling of feature maps is not a learnable process.
2. It is both memory and computational intensive process.

1.4.2 Transposed Convolutions

Transposed convolutions are the most commonly used approach to up-sample the feature maps to a higher spatial resolution [Odena *et al.* (2016)]. It maintains one to many relation ship between input and output feature maps. In this, the input features are interleave with *holes* gets convoluted with weights of filters by a standard convolution. The weights of filters are learned and gets updated during back propagation in training. The diagram to represent transposed convolution operation is presented in Figure 1.5.

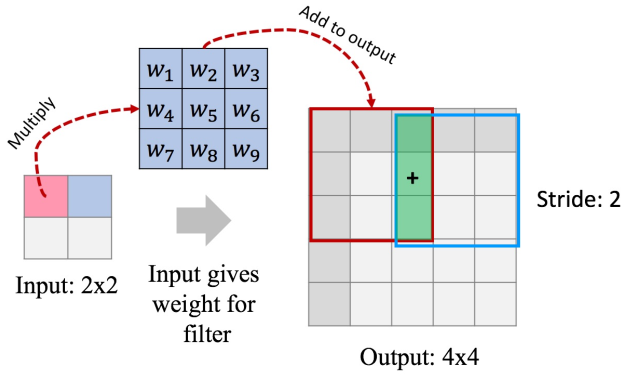


Figure 1.5: Transpose Convolution Operation

1.5 Different Loss Functions

The effect of false predictions in semantic segmentation architecture is quantified through loss functions. The model utilizes the values of loss functions to learn from its mistakes. So a proper choice of loss function is a necessary input to the semantic segmentation architecture to provide segmented image which is indistinguishable from its corresponding ground truth image. Various loss functions are presented in the literature, a detailed explanation about some of them are mentioned below.

1.5.1 Binary Cross Entropy Loss Function

Binary cross entropy loss (BCE) calculates the loss by comparing the class prediction probabilities of each pixel and its one hot encoded labels [Alpaydin (2020)]. BCE loss value is high for false predictions and low for true predictions. BCE loss provides equal priority to pixels of all classes means it cannot be able to discriminate pixels of most occurring classes and less frequent classes and the mathematical expression for BCE loss function is given in the Eq. (1.3).

$$L_{BCE} = \frac{-1}{N} \sum_{i=1}^{N} [y^{(i)}\ log(\widetilde{y^{(i)}}) + (1 - y^{(i)})\ log(1 - \widetilde{y^{(i)}})] \tag{1.3}$$

where $y^{(i)}$ are the pixel values of ground truth images, $\widetilde{y^{(i)}}$ are the predicted probabilities for the pixels of corresponding classes, and N represents the number of classes.

1.5.2 Lovasz Softmax Loss Function

Jaccard index or Intersection over Union (IOU) is evaluated by considering the overlap of pixels between predicted and its ground truth image. It reduces the bias towards the most frequent classes which is a useful metric for evaluating the semantic segmentation of small object instances. Lovasz Softmax loss (LZS) proposed by Berman *et al.* (2018) was intended to optimize mean Intersection over Union by considering a collection of pixel predictions.
The mathematical expression for L_{LZS} is given in Eq. (1.4)

$$L_{LZS} = \frac{1}{|C|} \sum_{c \in C} \triangle \overline{Jc} E(c) \tag{1.4}$$

Here $\triangle \overline{Jc}$ is the loss surrogate to the Jaccard index of class c, $E(c)$ is the vector of errors $[0, 1]^p$ and $|C|$ represents the number of classes.

1.5.3 Composite Loss Function (Proposed)

The combination of Binary cross entropy and Lovasz softmax Loss is utilized in our experiments to provide more emphasis on small object instances to improve pixel wise classification accuracy and it is given in Eq. (1.5)

$$L_{composite} = L_{BCE} + L_{LZS} \tag{1.5}$$

where L_{BCE} is Binary Cross Entropy loss and L_{LZS} is Lovasz Softmax loss.

1.6 Gradient Descent Optimization

It is an efficient adaptive stochastic optimization approach for first-order gradients [Kingma and Ba (2014)]. This is derived based on the advantages Adagrad [Duchi *et al.* (2011)] and RMSProp [Tieleman and Hinton (2012)] optimizers, which calculates adaptive learning rates for all parameters from estimates of first and second-order momentums. The update rule in Adam Optimizer is given in Eq. (1.6)

$$W = W - \alpha \, \frac{V^{c}_{dw}}{\sqrt{S^{c}_{dw}} + \epsilon} \tag{1.6}$$

where α represents the learning rate, V^{c}_{dw}, S^{c}_{dw} are the exponentially weighted average of gradients and square of the gradients after bias correction, ϵ is a small number to take care of division by zero operation, and W represents weights.

1.7 Problem Statement

To propose an efficient object extraction technique from remote sensed aerial images and its evaluation.

1.8 Main Contributions of the Thesis

The contributions of the research work presented in the thesis are listed below.

- A robust framework for contrast enhancement of remotely sensed aerial images is proposed. In this framework, color values are balanced to remove color cast, followed by saturation adjustment. As a final step, the contrast of images is enhanced by finding optimum values of unsharp masking filter through PSO algorithm.

- The novel semantic segmentation architecture named DRR Net for road extraction is proposed, which incorporates dense convolutions and residual connections. The proposed DRR Net is composed of multiple DRR modules operating at full resolution, and these modules are constrained to refine the predictions by stacking one over another. Further, each DRR module obtained predictions by extracting rich semantics.

- The O-SegNet, a robust encoder-decoder architecture, is presented for multi-object segmentation from HSR aerial imagery data. The Guided-Attention blocks are utilized in O-SegNet to model inter-relationship between features, and also to guide the successive GA blocks. The aggregation of encoder features extracted the global context through multi-level pooling. Further, attention between encoder and decoder is provided to focus on relevant encoder context while producing predictions.

1.9 Organization of the Thesis

The thesis is arranged into six Chapters, as follows.

Chapter 2 provides an exhaustive literature survey of aerial image enhancement and segmentation by relating progress and problems, which lead to the formulation of research gaps, and problem statement.

Chapter 3 discusses the need for contrast enhancement of captured aerial images. The detailed explanation of the newly proposed contrast enhancement algorithm is included in this Chapter, followed by a summarization of the work done to fulfil the first objective.

Chapter 4 includes the details about proposed semantic segmentation architecture DRR Net for road extraction from HSR aerial imagery data. The training details, computational complexity, and simulation results of the proposed DRR Net is specified in this Chapter. The summary of the work done for the semantic segmentation of aerial images is also presented.

Chapter 5 provides information about the another architecture named as O-SegNet for multi object extraction from HSR aerial imagery data. This Chapter details about the training, computational complexity, simulation results and summary of DRR Net.

Chapter 6 reports the conclusion and summaries of the overall work done to accomplish the research objectives. The Chapter also indicates the future scope for this research work.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

Nowadays, due to the ability of new sensors, the fine-resolution remotely sensed aerial imagery data of urban areas become increasingly available. Therefore, automatic interpretation and analysis of aerial images to extract and identify the intended objects is a prerequisite in majority of applications. The development of techniques for image enhancement and segmentation are considered as a major challenging problem in the field of remote sensing. The major contributions achieved so far in the field of aerial image enhancement and segmentation are provided in the below Sections 2.2 and 2.3, respectively. The research gap analysis in the field of remotely aerial image enhancement is discussed in the Section 2.5. Finally, the information regarding quality metrics of enhancement techniques and semantic segmentation approaches is presented in the Section 2.4.

2.2 Literature Survey on Enhancement Techniques

The high-resolution aerial images which are captured from higher altitudes possess a narrow range of brightness values. Consequently, the acquired images suffer from poor contrast. Therefore, it is necessary to improve the visual perception of such images by improving the contrast and brightness to facilitate further image analysis tasks like classification, segmentation, feature extraction, etc. Contrast enhancement improves the visual quality of aerial images by expanding a range of pixel intensities to unveil hidden object details. Plentiful literature has emerged in the field of aerial image enhancement, which is based on spatial domain and transformed domain. Some of the important contributions by distinguished authors across the globe in the field of image enhancement are discussed in this Section. Braik *et al.* introduced the automatic method using the Particle Swarm Optimization (PSO) algorithm for the enhancement of images [Braik *et al.* (2007*b*)]. In this method, the PSO algorithm is utilized to enhance the contrast and details of images by maximizing the defined objective function. During the maximization of an objective function, parameters of the technique are fine-tuned appropriately. This method was computationally less complex compared to the Genetic Algorithm (GA) based enhancement method. Despite less complexity, the proposed technique requires a few modifications to reduce the number of particles, iterations, and for application of local parameters to neighbourhood regions. Kwok *et al.* utilized the multi objective-based method for contrast enhancement of grayscale images by preserving the mean image intensity [Kwok *et al.* (2008)]. In this method, authors have considered discrete entropy as the first objective function, in which the main objective is maximization and gamma correction as a second objective function for intensity preservation. In this approach, the multi-objective PSO approach is incorporated to resolve the trade-off between requirements of contrast enhancement and preservation of mean image intensity. Authors claim that their method provides better experimental results for grayscale images but also demonstrated discrepancies in the output images. Demirel *et al.* proposed an image enhancement technique in the transform domain by employing a Discrete Wavelet Transform (DWT) and Singular Value Decomposition(SVD) [Demirel *et al.* (2009)]. In this technique, by incorporating DWT, the input image is divided into four frequency bands. Out of four sub bands, the low-low sub band image is estimated using the SVD method. Later, the enhanced image is reconstructed by employing inverse DWT. The quantitative performance of this approach is estimated through the Gaussian distribution of contrast-enhanced images.

Shanmugavadivu *et al.* proposed a new enhancement method, which is based on histogram equalization (HE) and PSO for contrast enhancement of images [Shanmugavadivu and Balasubramanian (2014)]. In this method, authors have used Otsu's method for segmenting the histogram of the original image into two sub-bands and equalizing both of them independently through optimized weighing constraints using the PSO algorithm. This technique was less stable and computationally complex than other existing methods.

Later in the literature, authors have proposed methods based on the hybridization of meta-heuristics optimization algorithms for contrast enhancement of images by [Hoseini and Shayesteh (2013), Coelho *et al.* (2009), and Gogna and Tayal (2013)]. Hoseini *et* *al.* proposed a hybrid algorithm for the global transformation of the input intensities. In this introduced framework, Ant Colony Optimization, Simulated Annealing (SA), and GA techniques are incorporated. The contrast of images is enhanced by generating and modifying transfer functions using Ant Colony Optimization and SA, respectively. Later, the GA algorithm is employed to operate the fitness function by finding optimal values. The experimental results of method [Hoseini and Shayesteh (2013)] are better but this method is computationally complex. Coelho *et al.* (2009) implemented a differential evolution scheme using chaotic sequences to enhance the contrast by maximizing fitness criterion while avoiding the local state . In this proposed framework, a fast convergence rate has obtained, and also population diversity is maintained by escaping from local minima. Multi-objective optimization has not considered in this approach. Mahapatra *et al.* proposed a hybrid method to improve the perception in images for the applications where brightness is of critical priority by combining PSO with the Negative Selection Algorithm (NSA) [Mahapatra *et al.* (2015)]. Suresh *et al.*introduced a Modified Differential Evolution (MDE) with a cuckoo search algorithm for contrast enhancement of satellite images [Suresh and Lal (2017)]. The developed algorithms worked with the exploration phase using MDE approach and exploitation phase by the cuckoo search algorithm. In this algorithm, the fitness function is employed to enhance the image perception in all ways by adjusting the set of parameters. The simulation results of this approach were promising, but it is a computationally complex method. The summary of image quality enhancement techniques are listed in Table 2.1.

Table 2.1: Summary of Enhancement Techniques

Method	Advantages	Disadvantages
PSO algorithm Braik *et al.* (2007*b*)	• Maximizes the number of pixels in the image	• Need to reduce number of particles and iterations. • Local parameters should be coded to apply to the neighbourhood.
Multi objecitve PSO approach Kwok *et al.* (2008)	• Achieves better results for gray scale images.	• Obtained discrepancies in the output images.
DWT+ SVD Demirel *et al.* (2009)	• Covers a wide range of gray values	• Estimated quantitative performance is presented.
DE with Choatic sequences dos Santos Coelho *et al.*	• Avoided local minimum state.	• Multi objective optimization is not considered.
Multi objective HE model Shanmugavadivu and Balasubramanian (2014)	• Preserved the brightness	• Computationally complex and less stable.
Hybrid algorithm based on GA Hoseini and Shayesteh (2013)	• Balanced the contrast and naturalness.	• Computationally complex.

Continued on next page

Table 2.1 – continued from previous page

Method	Advantages	Disadvantages
PSO + Negative selection algorithm Mahapatra *et al.* (2015)	• Brightness of images is retained.	• Need more number of iterations.
MDE + Cuckoo Search algorithm Suresh and Lal (2017)	• Entropy, standard deviation, and edge details are greatly improved.	• Computational complexity is high.

2.3 Literature Survey on Segmentation Techniques

The extraction of topographical features(roads, buildings, land, trees, etc.) present in the high-resolution aerial images is usually obtained by the segmentation of intended objects from others. The segmentation is the process of dividing the aerial image into distinct parts, with extent precisely according to the object's shape. Various segmentation techniques are present in the literature, which are mainly fall into two broad categories. The first approaches are based on manual feature extraction, and the second ones are automatic feature extraction approaches for segmentation. The significant contributions in both ways of segmentation of roads and buildings are presented in the Sections 2.3.1 and 2.3.2, respectively.

2.3.1 Literature Survey of Conventional Segmentation Techniques

Stoica *et al.* constructed the Gibbs point process for the detection of road pixels along the line segment [Stoica *et al.* (2004)]. During the process of detection, the lined-up road pixels are considered, whereas mixed road pixels are penalized using the candy model. In this probabilistic based Gibbs point process, the road pixels are estimated through the minimization of the energy function. The local minima state is avoided by incorporating a strategy which is based on Monte Carlo dynamics. However, this road detection approach is based on the assumption that roads are narrow in shape during the formation of the road network in the image. Amo *et al.* extracted road pixels irrespective of shape using a region growing technique by considering initial points and clues regarding the placement of points [Amo *et al.* (2006)]. With these, a rough road estimate is obtained initially. At later stages, more points are added to the algorithm after analysing the formation of the image. The obtained road approximation is then refined by applying region competition techniques. The major limitation associated with this method was the requirement of the user to select seeds for the region growing technique.

Hu *et al.* presented an automated method for road extraction in two stages. In the first stage, the road footprints are tracked after detection, and a road tree is grown using a road seeding approach after the classification of road footprints [Hu *et al.* (2007)]. The Bayes decision rule is employed in the pruning stage to remove paths that exude into the nearby regions. The main drawback of this work is the over-segmentation of multi-directional roads. Ortner *et al.* attempted to extract building outlines correctly using a framework derived from spatial point process [Ortner *et al.* (2007)]. The buildings are estimated by minimizing the energy function in which low-level information and geometric knowledge are considered. This method extracts only rectangular-shaped buildings and elementary shapes of other objects in the image. Inglada *et al.* introduced a Support Vector Machine (SVM) classifier to extract objects of several classes by incorporating the highest number of geometric features into it [Inglada (2007)]. The developed model learns generic features to recognize independent object classes. In this approach, the computational complexity related to an exhaustive search of parameters in SVM and processing time of images is high. Kluckner *et al.* introduced a low-complexity Random Forest (RF) classifier for classification of multi-class objects in aerial images [Kluckner *et al.* (2009)]. In this framework, conditional random fields are utilized to embed semantic contextual information into the classifier. However, this method provides high classification scores only by considering different image features like color, edges, and information related to height.

Mnih *et al.* introduced a feature learning approach that is based on a neural network for the automatic extraction of roads [Mnih and Hinton (2010)]. The proposed neural network contains millions of trainable weights to focus on the broad input context for the automatic extraction of roads. Further, the prediction performance of this approach is improved by utilizing the local spatial relationship of the output labels. Nevertheless, the disadvantages of this approach are: The road predictions contain gaps, and yields disconnected irregular patches in predictions. Hence, there is a need for further post-processing operations to fill gaps and also to produce connected patches in the predictions. Sirmacek *et al.* extracted buildings without any holes by formulating the detection problem as a probabilistic framework [Sirmacek and Unsalan (2010)]. In this framework, four different feature vectors are obtained and then fed into the probabilistic density function (pdf). The obtained feature vectors are Harris-corner-Based vectors [Harris *et al.* (1988)], Gradient-Magnitude-based Support Regions (GMSR) vectors [Unsalan (2006)], Gabor-Filtering-Based vectors [Vetterli and Kovacevic (1995)], and Features from Accelerated Segment Test (FAST) vectors [Rosten *et al.* (2008)]. The pdf is estimated using a variable kernel density estimation [Silverman (1986)] by representing building locations as joint random variables. However, this framework produces lower performance when buildings are tall, dense, and are not visible.

Das *et al.* implemented a framework for road extraction by exploiting the spectral contrast and linear trajectory features using probabilistic support vector machines [Das *et al.* (2011)]. From these salient features, road regions were segmented without the need for parameter tuning. This process could extract roads of greater width. However, narrow roads hidden under shadows and roads at junction regions were not extracted properly. Unsalan *et al.* developed a three stage approach to extract all kinds of roads by incorporating graph-based approaches in a probabilistic way [Unsalan and Sirmacek (2012)]. In this approach, road centres are detected at the first stage, road shapes are identified during the second stage, and finally road networks are extracted. The order of these stages is varied accordingly with the type of application. The disadvantage of this developed system is that it extracted roads only from images of predefined spatial resolution. Wegner *et al.* extracted roads from images of any spatial resolution by expressing the road extraction problem using Conditional Random Fields(CRF's) and graph cuts [Wegner *et al.* (2013)]. The prior information about the road structure is utilized at CRF formulation to improve the likelihood of predicted pixels along thin roads. However, the model parameters for this method need to determine manually, which made it as a semi-automatic method. Montoyo *et al.* automatically produced topologically accurate road maps in two stages [Montoya-Zegarra *et al.* (2014)]. The context and road plan are learned by embedding multi-scale appearance information into the per-pixel classifier. Next, the promising road structure is detected using the shortest path search algorithm based on the results of the pixel classifier. Finally, the road pixels are selected from these paths through energy minimization in a CRF. During the prediction of road pixels, unnecessary short connections are formed while bridging gaps.

Li *et al.* proposed an object extraction technique using Level Set Evolution (LSE) which does not require bridging of gaps in predictions [Li *et al.* (2014)]. The computations involved in the LSE approach are expedited with including the Gaussian kernel rather than the traditional mean curvature-based regularization term. By doing so, less number of parameters are involved while maintaining the performance and speed. The major pitfalls of this technique are twofold. Firstly, in this approach, there is a requirement of human intervention to initialize the zero level curve (ZLC). Hence it is a semi-automatic technique. Secondly, it is very challenging to estimate the involved parameter values of LSE. The summary of the above-discussed approaches is provided in the Table 2.2.

In techniques [Stoica *et al.* (2004),Li *et al.* (2014)], the objects are extracted by computing features in more than two stages from the smaller context of training data. During the object extraction, the spatial correlation of output labels is not utilized, which means these are unsupervised techniques. Further, the output of one stage is dependent on the precedent stage outputs. Hence, to get an overall high accuracy value, the accuracy of each stage should be maintained. However, techniques based on deep Convolutional Neural Networks (CNNs) extract the objects in a supervised way by considering the larger context of input data. The major benefit of deep CNNs is their ability to calculate features by learning from the high volume of input data. The deep-CNN-based approaches extract the objects through semantic segmentation to perceive *what is in the image and where it is located.* The following Section 2.3.2 discusses the important contributions to segment objects through the semantic segmentation in the field of deep learning.

Table 2.2: Summary of Conventional Segmentation Techniques

Method	Advantages	Disadvantages
Gibbs Point process Stoica *et al.* (2004)	• Extracted thin road properly. • Avoided false alarms.	• Assumed that roads are narrow in shape
Region growing and competition techniques Amo *et al.* (2006)	• Recovers road sides and also intersections.	• Requirement of user for seed selection.
Bayes decision rule Hu *et al.* (2007)	• Trims paths that leak into road surroundings.	• Over segmentation of roads.
Continued on next page		

Table 2.2 – continued from previous page

Method	Advantages	Disadvantages
Spatial Point process Ortner *et al.* (2007)	• No need of initial conditions. • Interaction with operator is not required.	• segments only rectangular-shaped buildings. • It is a slow algorithm.
SVM classifier Inglada (2007)	• System is adopted to different applications.	• Computational complexity is more.
RF classifier Kluckner *et al.* (2009)	• Involves less computational cost	• Need various image features to get high accuracy.
Based on Neural Networks Mnih and Hinton (2010)	• First approach to work on large scale real world data.	• Predictions contain gaps and disconnected patches.
Based on Probabilistic framework Sirmacek and Unsalan (2010)	• Time required to detect buildings is relatively short.	• Produces lower performance when buildings are tall and dense.
Probabilistic SVM Das *et al.* (2011)	• Roads are extracted with few obstacles.	• Failed to segment narrow roads at junctions.
Graph based approach Unsalan and Sirmacek (2012)	• Attempted to produce refined and improved road predictions.	• Extracted roads from images of predefined spatial resolutions.

Continued on next page

Table 2.2 – continued from previous page

Method	Advantages	Disadvantages
CRF's and graph cuts Wegner *et al.* (2013)	• Improved the likelihood of predicted pixels.	• Determination of model parameters manually.
Pixel classifier +Shortest path search Montoya-Zegarra *et al.* (2014)	• Extracted topologically correct roads.	• Short connections are formed.
LSE Li *et al.* (2014)	• Require few input parameters. • Computationally efficient approach.	• Human intervention is needed.

2.3.2 Literature Survey on CNN based Segmentation Approaches

Badrinarayan *et al.* introduced a fully convolutional encoder-decoder based architecture for semantic segmentation [Badrinarayanan *et al.* (2015)]. The encoder architecture resembles the Visual Geometry Group(VGG) network [Simonyan and Zisserman (2014)] with 13 convolutional layers. The decoder also contains the same number of layers as the encoder, with the purpose of mapping encoder features, which are of lower resolution to the full resolution. The max-pooled encoder feature maps are transferred to the decoder through pooling indices. However, by considering only the maximum values of encoder feature maps, there might be a possibility of losing fine details associated with small objects. The utilization of pooling indices may result in inefficient segmentation of small objects, especially in the case of high-resolution images. Ronneberger *et al.* presented an architecture named U-Net, with contracting and expanding paths [Ronneberger *et al.* (2015)]. The contracting path focuses on capturing the input context while expanding path attempts for precise object localization in the images. Skip connections are introduced as an alternative to pooling indices for transferring the learned features to the corresponding resolution level of the decoder. This results in a higher number of feature maps; hence, the complexity of the decoder increases. The architectures in [Badrinarayanan *et al.* (2015) , Ronneberger *et al.* (2015)] share a common point of using convolutional filters for feature learning and pooling layers to exploit semantics. The use of pooling layers reduces the spatial resolution of feature maps. Preserving spatial resolution is important for retaining the fine details of objects.

Yu *et al.* proposed another type of convolutions called as dilated/atrous convolutions for dense prediction problems [Yu and Koltun (2015)]. This type of convolutions aggregates the multi-scale contextual information by preserving the spatial resolution of the input. The corresponding authors claim that, with atrous convolutions, the receptive field increases because these convolutions support the exponential expansion of the receptive field without losing coverage. Saito *et al.* automatically segmented terrestrial objects from aerial images by constructing both feature extractors and classifiers using Convolutional Neural Networks [Saito *et al.* (2016)]. Chen *et al.* introduced architecture to perform semantic segmentation by utilizing distinct dilation filters in spatial pyramid pooling (SPP) [He *et al.* (2014)] for aggregation of multi-scale context [Chen *et al.* (2017*b*)]. The resulting architecture produced a segmentation map with one-eighth input resolution. Chen *et al.* presented the extended architecture by placing an additional decoder module to maintain the spatial resolution of the architecture mentioned above [Chen *et al.* (2018)]. This additional decoder refines the segmentation results across the object boundaries.

Yang *et al.* attempted to encode multi-scale information of objects, which have large scale variations by connecting the Atrous Spatial Pyramid Pooling (ASPP) network densely [Yang *et al.* (2018)]. The resulting Dense ASPP semantic segmentation architecture generates features that densely cover a large scale range without an increase in model size. However, it is observed that the obtained receptive field due to incorporation of dilation filters in [Yu and Koltun (2015) , Yang *et al.* (2018)] are not sufficient to preserve the spatial connectivity of objects in high resolution aerial images during extraction.

Huang *et al.* proposed the idea of dense convolutions that iteratively reuses the learned features at later resolutions by connecting each layer in a network to every other layer. [Huang *et al.* (2017)]. The idea is put forward for image classification based on the fact that shorter connections close to the input ease the training of the network. Jegou *et al.* extended the concept of dense convolutions to semantic segmentation by utilizing them in the paths of encoder and decoder. The resulting network has fewer parameters and achieves comparable performance without any post-processing module. However, due to the usage of dense convolutions together with skip connections in the up-sampling path, the model demands more memory during training [Jégou *et al.* (2017)].

Pohlen *et al.* utilized two streams in semantic segmentation architecture for combining of multi-scale context with pixel-level accuracy [Pohlen *et al.* (2017)]. Here, the first stream operates at full resolution to adhere to the segmentation of object boundaries precisely. The second stream comprises a sequence of pooling operations to extract the features for object identification. Both streams operate at a full image resolution and are combined using residual connections. Samy *et al.* obtained semantic segmentation of aerial images with multiple Field of View (FoV) models connected one above other sequentially to improve the predictions of precedent ones [Samy *et al.* (2018)]. Each designed FoV module operates at full resolution, contains a sequence of convolutions, pooling, and up-sampling layers. The approaches mentioned above increase both localization and classification accuracy. However, these techniques are computationally intensive as they operate at full resolution. Zhang *et al.* proposed a model named as ResUNet by incorporating residual connections in U-Net [Zhang *et al.* (2018*b*)]. The strengths of ResUNet are ease of training process due to residual units and information propagation from the encoder to the decoder. Despite its strengths, the resulting model failed to segment small roads in parking lots. Filin *et al.* attempted to refine the predictions of the ResUNet model by further processing the result [Filin *et al.* (2018)]. For refinement, road vectors are derived from the network probabilities. In this approach, the processing operations are mainly intended for the determination of road width, continuation of disconnected roads, and removal of bad roads. Sun *et al.* introduced a model for the generation of road maps by stacking two U-Nets. Recall values are improved by incorporating road vectors and shortest path search in the approach [Sun *et al.* (2018)]. The introduced models by [Filin *et al.* (2018)] and [Sun *et al.* (2018)] need further post-processing operations to extract road center lines and to connect disjoint roads.

Kim *et al.* placed SPP at the end of the encoder of U-Net to aggregate multi-scale contextual information [Kim *et al.* (2019)]. Distinct pooling layers are used in the SPP module to learn the fine-grained classification maps from the input data. The major limitation with this approach is the increased depth of feature maps due to the usage of a more number of filters. Hence, the computational complexity of an architecture is high. Aich *et al.* introduced a technique called Depth to Space (D2S) to reduce the computational complexity by excluding the decoder in the architecture [Aich *et al.* (2018)]. The encoders employed are ResNet50 [He *et al.* (2016*a*)] and VGG 16 [Simonyan and Zisserman (2014)]. In the D2S approach, the features along the depth dimension are reordered along the spatial dimension. This approach is, however, not well suited for the segmentation of small objects as there is no learning path for up-sampling.

Sevo *et al.* performed the patch wise classification of pixels in the aerial images using GoogleNet architecture [Szegedy *et al.* (2015)]in two stages [Ševo and Avramović (2016)]. In the first stage, the pre-trained weights of the GoogleNet, except the fully connected layer, are copied. At the second stage of training, weights of the convolutional filters are fine tuned according to the corresponding dataset. The authors attempted to obtain a high prediction rate; however, with this approach, the better classification is achieved if the patching step size is small. Wei *et al.* introduced a refined convolutional network originated from the Visual Geometry Group (VGG) network [Simonyan and Zisserman (2014)] to refine the road structure [Wei *et al.* (2017)]. The information about the road's geometric structure is incorporated in the cross-entropy loss function to impose the penalty to the loss values based on the proximity of the pixels from road regions. The adaptation of road structure is achieved after utilizing extra deconvolutional and fusion layers than the VGG network. Kaiser *et al.* introduced a variant of Fully Convolutional Network (FCN) [Long *et al.* (2015)] to generate segmentation maps for roads and buildings [Kaiser *et al.* (2017)]. However, the introduced architecture is computationally complex due to the presence of more number of filters.

Sun *et al.* provides semantic segmentation of aerial images with the low complexity variant of FCN and Digital Surface Model (DSM) [Sun and Wang (2018)]. This method aimed to restore the information which is lost due to down sampling operation by introducing a strategy to combine semantic and detailed information from deep layers and shallow layers, respectively. The overall accuracy is improved if network architecture is modified. Eerapu *et al.* implemented a dense refinement residual (DRR) network for road extraction in all circumstances by utilizing dense convolutions only in the encoder [Eerapu *et al.* (2019)]. It contains four different DRR modules that are stacked together to refine the predictions of the previous module(s). Each DRR module has contracting, expanding paths, and also residual connections to transfer the predictions subsequent module(s) to provide further refinement. The time required to train the model is significantly high as compared to other approaches despite having fewer model parameters. The above-discussed architectures produce segmentation maps by extracting fine-grained features through a set of convolutions in the encoder and also recovers the original resolution via a sequence of transposed convolutions in the decoder. However, these architectures do not provide emphasis on essential regions of intended objects during the learning and reconstruction stages.

The attention techniques are widely employed in the field of Neural Machine Translation (NMT) to focus on salient features during translation and alignment stages in [Bahdanau *et al.* (2014)], [Wu *et al.* (2016)], [Vaswani *et al.* (2017)], and [Gehring *et al.* (2017)]. The concept of attention is adopted to perform classification and semantic segmentation in [Wang *et al.* (2017)], [Oktay *et al.* (2018)], [Fu *et al.* (2019)], and [Schlemper *et al.* (2019)], respectively.

Oktay *et al.* proposed a model based on the attention gate that learns to concentrate on diversified target structures present in medical images [Oktay *et al.* (2018)]. The attention gate is incorporated into standard U-Net architecture, so the resulting model is intended to suppress irrelevant regions while focusing on essential features in images. The computational overhead is high as compared with U-Net. Schlemper *et al.* utilized the above architecture for the segmentation of multiple organs in computed tomography (CT) images of the abdomen. Fu *et al.* appended position attention module (PAM) and channel attention module (CAM) on top of dilated FCN to model semantic dependencies present in spatial and channel dimensions Fu *et al.* (2019)]. The PAM aggregates the features by a weighted combination of features across all positions irrespective of distance. In the same way, CAM integrates feature maps among all channels. Finally, the PAM and CAM features are added to improve the segmentation results precisely. However, networks in [Wang *et al.* (2017)- Schlemper *et al.* (2019)] applied attention mechanism either at encoder or decoder. Pan *et al.* introduced the network to produce refined segmentation results in an adversarial way, which includes generator and discriminator [Pan *et al.* (2019)]. The generator in the proposed framework is U-Net with spatial and channel attention mechanisms, while the discriminator is an adversarial network to differentiate between generator output and ground truth. Due to the utilization of adversarial mechanism in the framework, leads to additional complexity. The milestones of CNN based approaches, which are discussed above, are summarised in the Table 2.4.

Table 2.3: Summary of Semantic Segmentation Techniques

Method	Advantages	Disadvantages
Encoder-Decoder Badrinarayanan *et al.* (2015)	• Efficient memory computation time during validation.	• Loss of fine details due to pooling operation.
U-Net Ronneberger *et al.* (2015)	• Transfers learned features to decoder.	• Higher number of feature maps.
		Continued on next page

Table 2.3 – continued from previous page

Method	Advantages	Limitations
Dilated Convolutions + SPP Chen *et al.* (2017*b*)	• Aggregation of multi-scale context.	• Produces lower resolution output.
Dilated Convolutions + SPP+Decoder Chen *et al.* (2018)	• Spatial resolution is maintained.	• Spatial connectivity of small objects is not preserved.
Based on Dense convolutions Jégou *et al.* (2017)	• Have fewer parameters. • Eases training process.	• Demands more memory during training.
Full resolution residual network Pohlen *et al.* (2017)	• Increases object localization and recognition accuracy.	• Computationally intensive.
Mutiple FoV modules	• Increases object localization and recognition accuracy.	• Computationally intensive.
ResUNet Zhang *et al.* (2018*b*)	• Eases training process. • Flow of information from encoder to decoder.	• Failed to segment roads present in parking lots.
ResUNet+ processing operations Filin *et al.* (2018)	• Attempted to improve network output.	• Need for further processing operations.
Stacked UNet Sun *et al.* (2018)	• Taken care of unbalanced classes in the training data.	• Demands post-processing operations.

Continued on next page

Table 2.3 – continued from previous page

Method	Advantages	Limitations
UNet+PPL Kim *et al.* (2018)	• Detailed object classification	• Computational complexity is high.
D2S approach	• Computational complexity is reduced greatly.	• Not suitable for segmentation of small object instances.
Based on GoogleNet Ševo and Avramović (2016)	• Attempted to achieve high prediction rate.	• Patching step need to reduce to achieve high classification accuracy.
Refined road structure convolutional neural network Wei *et al.* (2017)	• The road structure is refined.	• Need more deconvolutional and fusion layers.
Variant of FCN Kaiser *et al.* (2017)	• Avoid annotating of large volume of training data.	• Computationally complex.
FCN + DSM Sun and Wang (2018)	• Detailed information is restored	• Need to improve network architecture to improve accuracy.
DRR network Sun and Wang (2018)	• Extracted roads in all circumstances. • Fewer model parameters.	• Time required for training and inference is high.

Continued on next page

Table 2.3 – continued from previous page

Method	Advantages	Limitations
Attention gates +U_Net Oktay *et al.* (2018)	• Increased model sensitivity and prediction accuracy.	• Increased computational overhead than U-Net.
Dual attention network Fu *et al.* (2019)	• Improved segmentation results.	• Need to decrease the computational complexity. • Robustness of the model should be enhanced.
GAN based network Pan *et al.* (2019)	• Enhances the utilization of features to produce improved predictions.	• Additional complexity due to adversarial way.

2.3.3 Literature Survey on Attention Mechanism in the field of Segmentation

This Section discusses various approaches present in the literature to provide attention to relevant image parts while performing segmentation. The attention techniques are widely employed in the field of Neural Machine Translation (NMT) to focus on salient features during translation and alignment stages in [Bahdanau *et al.* (2014), Wu *et al.* (2016), Vaswani *et al.* (2017), and Gehring *et al.* (2017)]. The concept of attention mechanism is adopted to perform classification and semantic segmentation. Chen *et al.* obtained predictions by training DeeLab architecture [Chen *et al.* (2017*a*)] and attention mechanism jointly [Chen *et al.* (2016)]. In this approach, the attention mechanism is incorporated to weight each pixel location of the multi-scale features learned using DeepLab architecture. This method failed to segment small object instances and to handle imbalance classes. Wang *et al.* performed image classification with Residual Attention Network, which is a stack of attention modules and introduced attention residual learning to ease the training process [Wang *et al.* (2017)].

Zhao *et al.* attempted to avoid local neighborhood constraints while performing semantic segmentation by introducing a Point-wise Spatial Attention (PSA) network [Zhao *et al.*

(2018)]. In this network, bi-directional information flow is allowed by connecting every position of feature map to all positions through a learned adaptive mask. Hu *et al.* introduced Squeeze and Excitation (SE) block to model interdependencies between the channels of features. With these blocks, SE architecture is constructed to perform semantic segmentation [Hu *et al.* (2018)]. In this technique, a trade-off between model complexity and performance is present. Oktay *et al.* proposed attention gates to focus on irregular shaped objects explicitly [Oktay *et al.* (2018)]. These gates can be incorporated into any semantic segmentation architecture with less computational overhead. Further, the performance of this network is improved only by employing fine resolution inputs.

Schlemper *et al.* introduced Attention Gated (AG) Network to learn the object's structural information while performing segmentation [Schlemper *et al.* (2019)]. However, optimal performance is obtained by training the network at different scales, followed by fine-tuning. Zhang *et al.* presented an Edge-Attention Network to focus on edge information explicitly while performing segmentation [Zhang *et al.* (2019)]. This learned edge representation and weighted aggregation modules are fused and utilized at the expanding path of a network. Huang *et al.* introduced a new technique to factorize the dense affinity matrix of the self-attention mechanism into two matrices of long and short-range spatial intervals [Huang *et al.* (2019*a*)]. Due to this factorization, the computational and memory complexity required for the processing of high-resolution feature maps is substantially reduced. Zhu *et al.* introduced a pyramid sampling module in the self-attention module and fusion block in the semantic segmentation architecture [Zhu *et al.* (2019)]. With this approach, the authors claim that semantic segmentation efficiency is improved by reducing the computational overhead.

Pan *et al.* introduced Generative Adversarial Network (GAN) based semantic segmentation architecture [Pan *et al.* (2019)]. In this architecture, U-Net is utilized as a generator network with spatial and channel attention mechanisms to enhance the useful features selectively. The discriminator is embedded only with a channel attention mechanism to differentiate predictions and ground truth images. This resulting architecture produced predictions in an adversarial way, which leads to additional complexity. Huang *et al.* presented a Criss-Cross Network to obtain contextual information of pixels while performing segmentation [Huang *et al.* (2019*b*)]. The contextual information is extracted by attaching a criss-cross module to every pixel of the image. Fu *et al.* introduced the Dual Attention Network to model semantic interdependencies by appending attention modules across spatial and channel dimensions on top of dilated FCN architecture [Fu *et al.* (2019)]. Further, the rich contextual dependencies are captured by integrating local and global contexts. The computational complexity of this approach needs to reduce while preserving the robustness. Han *et al.* proposed a real-time semantic segmentation architecture, EdgeNet, to increase the inference speed. The EdgeNet is composed of channel attention and class-aware edge loss module to uplift the segmentation accuracy [Han *et al.* (2020)]. It improves the classification scores near to the edges. Li *et al.* introduced a SCattNet by proposing lightweight attention modules across spatial and channel dimensions to refine the features [Li *et al.* (2020)]. In this approach, new attention modules are required to capture the discriminative features of objects of similar semantics. Zhang *et al.* presented a 2D semantic segmentation architecture by incorporating residual modules combined with the attention gates in the U-Net [Zhang *et al.* (2020)]. There is a loss of contextual information and local details due to the slicing of images.

Table 2.4: Summary of Semantic Segmentation Techniques

Method	Advantages	Disadvantages
DeepLab + attention mechanism Chen *et al.* (2016)	• Multi-scale features are extracted	• Cannot handle imbalanced classes. • Failed to segment small objects.
PSA network Zhao *et al.* (2018)	• Avoids neighbourhood constraint	• Provides low attention at the current position of feature maps.
SE Block Hu *et al.* (2018)	• Channel interdependencies are modelled	• Trade-off between model complexity and performance.
Attention Gated Networks Schlemper *et al.* (2019)	• False positives are reduced.	• Fine tuning is required to get optimal performance.
Edge attention network Zhang *et al.* (2019)	• Edge representations learned explicitly	• Edge details are preserved in the early encoding layers.

Continued on next page

Table 2.4 – continued from previous page

Method	Advantages	Disadvantages
Interlaced Sparse Attention Huang *et al.* (2019*a*)	• Reduced memory and computation complexity.	• Multi-scale testing need to incorporate to boost performance.
Asymmetric Non-Local technique Zhu *et al.* (2019)	• Improved segmentation performance with less computational cost	• Trade off between efficiency and efficacy.
Based on GANs Pan *et al.* (2019)	• Enhances features selectively.	• Computational complexity is increased.
Criss-Cross Network Huang *et al.* (2019*b*)	• Requires less computational cost and memory.	• Adapts alternative way to capture image dependencies rather than Global pooling.
Dual Attention Network Fu *et al.* (2019)	• Captures semantic inter dependencies. • Computational complexity is not increased greatly.	• Robustness need to increase by decreasing the number of parameters.
EdgeNet Han *et al.* (2020)	• Improves classification results across edges	• Slower frame rates in real time application.
SCattNet Li *et al.* (2020)	• Features are refined adaptively	• Need to capture the discriminative features of similar objects.
ResUNet with attention gates Zhang *et al.* (2020)	• Pays attention to small scale objects	• Loss of context and fine details.

2.4 Quality Metrics for Enhancement and Semantic Segmentation of Aerial Images

The different quality metrics to measure the effectiveness of enhancement method and semantic segmentation approaches are presented here.

2.4.1 Quality Metrics for Enhancement of Aerial Images

2.4.1.1 No reference Image Quality Metric for Contrast Distortion(NIQMC)

To measure the image quality after performing image enhancement, NIQMC [Gu *et al.* (2016)] calculated without need of reference image. This quality metric is calculated based on local details and global details and it is presented in the Eq. (2.1).

$$NIQMC = \frac{Q_L + \gamma Q_G}{1 + \gamma} \tag{2.1}$$

where, Q_L denotes the local quality measure calculated from the entropy of local neighborhood regions, Q_G indicates the global quality measure, and γ represents the weight factor in controlling the significance of local and global information.

2.4.1.2 Blind Image Quality Measure of Enhanced Images (BIQMC)

In this quality metric, different image features like contrast, brightness, sharpness, etc. are calculated. These calculated measures are fed to the regression-based model, which is trained with big-data to evaluate the visual quality of enhanced images[Gu *et al.* (2017)].

2.4.1.3 Michelson Contrast (MICHELSON)

The Michelson contrast introduced by [Michelson (1995)] for images where larger area of uniform luminance is not present. The mathematical equation to measure Michelson contrast is given in Eq. (2.2).

$$Michelson = \frac{L_{max} - L_{min}}{L_{max} + L_{min}} \tag{2.2}$$

Where L_{max} and L_{min} represents the maximum and minimum luminance of the image.

2.4.1.4 Discrete Entropy (DE)

Discrete Entropy measures the average information content by summing the products of outcome probabilitiesp_i and log of the inverse outcome probabilities. The mathematical expression is given in Eq. (2.3).

$$DE = -\sum p_i \ log \ p_i \tag{2.3}$$

2.4.1.5 Measure of Enhancement (EME)

The measure of enhancement presented by [Agaian *et al.* (2007)] calculates the average ratio of maximum and minimum intensities in the transform domain. The mathematical expression to calculate EME is given in Eq. (2.4).

$$EME = \frac{1}{k_1 k_2} \sum_{l=1}^{k_1} \sum_{k=1}^{k_2} 20 \ \ln \frac{I_{max}}{I_{min}} \tag{2.4}$$

where k_1 and k_2 represents image blocks, I_{max} and I_{min} denote the maximum and minimum pixel intensities in the image blocks.

2.4.1.6 Pixel Distance (PIXDIST)

The distance between pixels in grayscale is used to measure the visual quality of the contrast-enhanced image. The mathematical expression to calculate pixel distance is given below in Eq. (2.5) according to [Chen *et al.* (2006)].

$$PIXDIST = \frac{1}{N(N-1)} \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} H(i)H(j)(j-i) \ \ for \ \ i, j \ \epsilon \ [0, \ M-1] \tag{2.5}$$

Here, M represents the grayscale range, N is the total number of pixels in the image, and H denotes the histogram of the contrast-enhanced image.

2.4.2 Quality Metrics for Semantic Segmentation of Aerial Images

2.4.2.1 Per-class Accuracy

Per-class accuracy calculates the proportion of correctly predicted pixels of corresponding classes over pixels in the labelled image. The road (R_{Accu}) and building accuracy (B_{Accu}) values are calculated from the Eq. (2.6) and (2.7).

$$R_{Accu} = \frac{1}{N} \sum_{I \in N} \frac{P_R}{G_R} \tag{2.6}$$

$$B_{Accu} = \frac{1}{N} \sum_{I \in N} \frac{P_B}{G_B} \tag{2.7}$$

where P_R, P_B, G_R, and G_B represents the number of predicted road, building pixels, the total number of road, building pixels in the ground truth, respectively. Here I denotes the image, which is the subset of the total number of images (N) in the dataset.

2.4.2.2 Intersection Over Union (IOU)

IOU provides the percentage of overlap between the predicted image and its corresponding ground truth image. So, the average intersection over the union of the predicted image of each class with the labeled image is measured using IOU values, and these values are calculated using the Eq. in (2.8), according to [Long *et al.* (2015)].

$$\frac{1}{N_c} \sum_{i=1}^{N_c} \frac{C_{ii}}{T_i + P_i - C_{ii}} \tag{2.8}$$

where N_c denotes the number of classes present, C_{ii} represents the class-wise predicted pixels, T_i represents the total number of corresponding class-wise pixels in the ground truth, and P_i, denote the total number of pixels, whose predictions are i.

2.4.2.3 Precision and Recall

Precision (P) denotes the ratio of true positives (T_P) to the sum of true positives and false positives (($T_P + F_P$)), which is given in Eq. (2.9)

$$Precision = \frac{T_{(P)}}{T_{(P)} + F_{(P)}} \tag{2.9}$$

Recall (R) values are calculated as true positives over the combination of true positives and false negatives (($T_P + F_N$)) and is given in Eq. (2.10).

$$Recall = \frac{T_{(P)}}{T_{(P)} + F_{(N)}} \tag{2.10}$$

The high recall and low precision values mean that the model segments the majority of pixels, but most of them are incorrect. As a contrast, low recall and high precision values indicate that the model predicts very few correct pixels. The accurate semantic segmentation architecture must have higher values of precision and recall.

2.5 Research Gap Analysis

2.5.1 Gap Analysis for Aerial Image Enhancement

- The stability of the enhancement technique needs to maintain by avoiding discrepancies and blocking effects present in the output image. Only a few researchers address the above objectives in contrast enhancement of aerial images.

- The multi-objective function needs to formulate to improve the various image properties without much increasing the computational complexity of the enhancement technique.

- There is a requirement in nature-inspired optimization algorithms to reduce the number of particles, and iterations to increase speed while maintaining the accuracy.

2.5.2 Gap Analysis for Aerial Image Segmentation

- The majority of aerial image segmentation techniques are either semi-automatic or not automatic: need user intervention and determination of model parameters manually.

- Hence, there is a scope to develop an automatic multi-object extraction to segment diverse shaped objects with topological completeness while avoiding over-segmentation.

2.5.2.1 Building Extraction Techniques

Building extraction techniques presented in the literature failed in following scenarios.

- If there is geometric similarity between the background objects and roofs of the building.

- When the number of building are dense in the environments like urban cities.
- The appearance of buildings and nearby objects look similar.

2.5.2.2 Road Extraction Techniques

Road extraction techniques presented in the literature have the following flaws.

- Failed to segment roads with very high curvature and variable width.
- Not efficient to extract roads without gaps while maintaining the spatial connectivity.

Therefore, there is a need to propose a multi-object extraction technique which considers above-mentioned circumstances.

2.6 Research Objectives

- To propose a reliable enhancement technique and evaluate its performance on aerial images.
- To propose an efficient semantic segmentation architecture and validate its performance on aerial images.
- To develop a reliable objects segmentation technique and investigate its performance on aerial imagery data.

Chapter 3

Quality Enhancement of Aerial Images

3.1 Introduction

This Chapter discusses the necessity for aerial image enhancement prior to object extraction. Further, the details of the newly proposed framework for aerial image enhancement are presented. The aerial remotely sensed images are captured from higher altitudes using Air crafts, resulting in the atmospheric degradation source, high geometric fidelity, and possess a narrow range of brightness values. Therefore, there is a need to improve the contrast of the image to preserve the details that aid in further image processing operations. The objective of the image contrast enhancement process is to improve the visual human perception by minimizing unwanted artifacts. The enhancement process involves defining a kernel transformation function to map pixel intensities to a different set of values, with a criterion/fitness function to quantify the enhanced image quality. The detailed information about the proposed aerial image enhancement technique is explained in the Section 3.2. The experimental results of proposed technique are discussed in the Section 3.3, and summarized this work in the Section 3.4.

3.2 Proposed Aerial Image Enhancement Technique

The details about the proposed aerial image enhancement technique are presented in this Section. The proposed enhancement technique is represented in three stages: (1) Efficient color balancing and saturation adjustment, (2) Efficient color restoration, (3) Modified contrast enhancement using PSO algorithm. The flow diagram for proposed aerial image en-

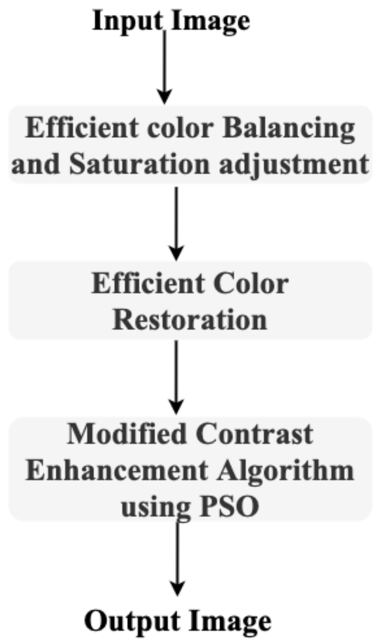


Figure 3.1: Flow diagram of proposed aerial image enhancement technique

hancement technique is presented in Figure 3.1.

3.2.1 Efficient Color Balancing and Saturation Adjustment

A RGB aerial remote sensing image I is given as input with red (R), green (G), and blue (B) as it's channels and then it is normalized between 0 and 1 which is presented in Eq. (3.1).

$$S(i, j) = R(i, j), G(i.j), B(i, j) \in [0, 1] \tag{3.1}$$

where (i, j) denotes the pixel indices and [0, 1] is the range of normalized magnitudes. The indices may be removed henceforth as it is understood from the context. According to grey-world assumption, the averaged image color is grey thereby eliminating color cast. For this assumption to hold there should be exponential alignment of the mean color values of the channels according to [Kwok and Shi (2015)]. Moreover the use of exponent assures that all pixel values are within the prescribed range [0, 1]. First we calculated the minimum value of mean brightness for each input color channel based on method presented by [Kwok and

Shi (2015)] and is given in Eq. (3.2).

$$S_M = min_c \left\{ \frac{1}{N} \sum_{W,H} S^C \right\} \tag{3.2}$$

Where S^C denotes the pixel color magnitude for $C \in R, G, B$ and N depicts the total number of pixels which is actually the product of width (W) by height (H) of the input image, that is $N = W \times H$. The summation is performed for all three color components for all pixels. The average values belonging to each channel are defined and is given in Eq. (3.3).

$$S_M^C = \frac{1}{N} \sum_{W,H} S^C \tag{3.3}$$

Then, there is further alignment of each color channel to the mean image brightness and this is done by each pixel raised to an exponent. It is given in Eq. (3.4).

$$k^C = \frac{log(S_M)}{log(S_M^C)} \tag{3.4}$$

where k^Cdenotes the exponent for each individual color.
Removal of color cast is followed by saturation adjustment of image. This is impelled by scrutinizing the definition of color saturation (CS) in both hue saturation intensity (HSI) and hue saturation value (HSV) color spaces and is given in Eq. (3.5) and Eq. (3.6) respectively.

$$CS_{HSI} = 1 - \frac{3 \times min\{R, G, B\}}{R + G + B} \tag{3.5}$$

$$CS_{HSV} = 1 - \frac{min\{R, G, B\}}{max\{R, G, B\}} \tag{3.6}$$

From here it can be easily deduced that saturation can be enhanced by compressing $min\ R, G, B$ or amplifying. So saturation adjustment comprises of two stages, which are described below. First, there is global alignment of all the pixel magnitudes to cover [0, 1] and is given in Eq. (3.7).

$$S^C = \frac{S^C - min_{C,WH}\{S\}}{max_{C,WH}\{S\} - min_{C,WH}\{S\}} \tag{3.7}$$

where $minx_{C,WH}\{S\}$ represents the minimum values calculated over all color channels and pixels and $max_{C,WH}\{S\}$ represents the maximum values calculated over all colour channels and pixels. After this process, at least one pixel with minimum color would be at zero and at least one pixel with maximum color would be at unity. Hence, as per the definition, there would be partial saturation enhancement of pixels.

Second, colors of each pixel would be sorted into three elements defined as minimum (β) , middle (χ) and maximum (δ) which are colour independent and satisfies the condition $\beta < \chi < \delta$. Further, for post normalization we have defined a magnitude variable (μ) and is given in Eq. (3.8).

$$\mu = \frac{\chi - \beta}{\delta - \beta} \tag{3.8}$$

This is the ratio of middle element to the min-max range. This is succeeded by a shift-and-scale process or in simpler term, compression and expansion of the minimum and maximum elements respectively depending on the parameter μ, $(0 < \mu < 1)$ as depicted below in Eq. (3.9) and Eq. (3.10) respectively.

$$\beta = (1 - \mu)\beta \tag{3.9}$$

$$\delta = \mu + (1 - \mu)\delta \tag{3.10}$$

This operation is indeed needed for keeping the resultant elements within the range [0, 1]. The change in magnitudes of minimum and maximum element leads to color shift which is reduced by restoring original ratio of the middle element between the minimum and maximum elements and it is presented in Eq (3.11).

$$\chi = \mu \times (\delta - \beta) + \beta \tag{3.11}$$

After this stage, the elements owing to their respective sorting index are remapped to their respective color channels and an image possessing improved saturation is obtained.

3.2.2 Efficient Color Restoration

The RGB saturated image from the prior step is processed further to overwhelm any sort of color violation. Here, a color restoration technique presented by [Jmal *et al.* (2017)] is employed to enhance the non-uniform illuminated regions. This technique is basically an optimized search procedure for computing and evaluating optimal parameters of the image. This processed image is passed for a modified contrast enhancement process.

3.2.3 Modified Contrast Enhancement using PSO Algorithm

After enrichment of non-uniform illuminated regions, a contrast enhancement process is applied to improve the brightness of the image. This process is a modified unsharp masking filter (UMF) introduced by [Kwok and Shi (2014)]. The enhancement operation is carried out on the brightness space. For this, the processed RGB image is converted to hue-saturation-value (HSV) space according to [Kwok *et al.* (2012)], where color is denoted by hue (H), the richness of the color is represented by saturation (S) and brightness is denoted by value (V). Here, the unsharp masking enhancement is performed on V- channel. The operation of UMF is given in Eq. (3.12) [Mohamed *et al.* (2010)].

$$Z = V + sX \tag{3.12}$$

where Z is the filtered V-channel pixel, V is the given input V-channel pixel, s is the control factor deciding the strength of enhancement and X is the edge signal from the kernel. This is done using two steps.

3.2.3.1 Kernel design

The performance of UMF is largely dependent on the filter kernel used and the proper setting of gain factors. The improper setting of gain factor would make the output either under-enhanced or over-enhanced according to [Mohamed *et al.* (2010)]. So, here a kernel ζ of 3×3 size is taken. The chosen kernel is used to extract local edge thereby giving thinner edges and improved sharpness in the enhanced image and is given in Eq. (3.13).

$$\zeta = \frac{1}{16} \times \begin{bmatrix} -a & -2 & -a \\ -2 & 12 & -2 \\ -a & -2 & -a \end{bmatrix} \tag{3.13}$$

and the constraint of kernel element is given by $a > 0$. The edge signal output from kernel is depicted in Eq. (3.14).

$$X = \zeta \otimes \Delta \tag{3.14}$$

where $\otimes$ represents the convolution operation and Δ is the 3×3 neighborhood pixels positioned around pixel V.

The convolved output is scaled with control parameter s and is added to original image in order to obtain enhanced sharpness. Hence, to achieve desirable enhancement results, parameters a and s are required to be optimized. This is done using PSO algorithm.

3.2.3.2 Particle Swarm Optimization (PSO) Algorithm

The PSO algorithm developed by the authors [Braik *et al.* (2007*a*), Mai *et al.* (2011), Faria *et al.* (2013), Yang (2014) and Delice *et al.* (2017)] belongs to the class of meta-heuristic algorithm. It is inspired from the swarm behaviour of living species in nature, such as fish and bird schooling, while searching for food. The reason for choosing PSO algorithm for optimizing kernel parameters is that because it mainly consists of mutation and selection parameters. Wan *et al.* (2018) introduced a PSO algorithm where there is no crossover phase, means it provides high mobility in particles with a high degree of exploration. Hence, it is suitable for finding optimal solutions to arduous optimization problems. The PSO algorithm has many advantages observed by [Ab Wahab *et al.* (2015)], which are: (1) Implementation of PSO algorithm is simple because it requires to set only few parameters. (2) PSO algorithm is an effective in global search and also insensitive to scaling of design variables. (3) PSO algorithm is easily parallelized for simultaneous processing, it has propensity to result in a fast and early convergence in mid optimum points.

Here, in this optimization problem, the parameters are going to be optimized are kernel parameter element a, and parameter s. For this, a particle in PSO algorithm is encoded with these parameters and is given in Eq. (3.15).

$$T = [t_1\,,\, t_2] = [a\,,\, s] \tag{3.15}$$

The swarm in PSO algorithm, consists of q particles, i.e., T_j where $j = 1, .., q$ (q set as 2 in our designed problem). In the beginning, the particles are assigned its initial positions in a random manner, in the potential solution space. The population size (P) of the solution space is initialized as 30 and the maximum number of generations (G) is fixed as 30 according to [Mai *et al.* (2011)]. Then, the particles are updated according to their defined objective function and during a number of time steps $i = 1, ...G$, they are guided to optimal solutions. The particles wander in the solution space and are attracted to global best solution T_g ascertained so far. The motion of the particle is controlled by its current best solution T_i^q. The velocity of motion and the new position of the particle based on its original position is calculated according to [Mai *et al.* (2011)]. The new velocity vector presented by [Mai *et al.* (2011)] is given in the Eq. (3.16).

$$v_{j,i+1} = \eta_j v_{j,i} + c_g(T_i^g - T_{j,i}) + c_q(T_{j,i}^q - T_{j,i}) \tag{3.16}$$

where velocity vector is denoted by v and inertia constant is denoted as η. The acceleration constants are denoted as c_g and c_q, which are random numbers in $[0, c_{max}]$ and c_{max} is the maximum value taken between 1.7 and 2.0.
According to new velocity, the new position of the particle can be updated as given in the Eq. (3.17).

$$T_{j,i+1} = T_{j,i} + v_{j,i+1} \tag{3.17}$$

In this problem, $a > 0$and $s > 0$, are the required constraints. Hence, both the constraints are satisfied by employing the given step by Eq. (3.18).

$$T_{j,i} \leftarrow |T_{j,i}| \tag{3.18}$$

The information content is determined by maximizing the objective function which is defined in the Eq. (3.19).

$$f_{obj} = \epsilon \times \left(1 - \frac{\sigma}{W \times H}\right) \tag{3.19}$$

where W, H represent the width and height of the image, ϵ is the entropy, σ denotes the number of over-ranged pixels. After maximizing this, output edge signal X is updated and after scaling with parameter s, it is superimposed onto the given V- channel image V and a higher contrast enhanced output V- channel image Z is obtained and it is again converted back to RGB channel to give the final output image. This framework is tested on a collection of different aerial remote sensing images.

3.3 Simulation Results and Discussion

In this Section, description about aerial image dataset used in the experimentation is presented in the Section 3.3.1. The proposed technique quality metrics and its visual enhancement results are evaluated and compared with other existing enhancement methods, like UM- FKG [Kwok and Shi (2014)] , RHE- DCT [Fu *et al.* (2015)], IFAIR [Kwok and Shi (2015)],LSCN [Zhan *et al.* (2017)], a method of JEI [Jmal *et al.* (2017)] and MDE algorithm [Suresh and Lal (2017)] in Section 3.3.2.

Table 3.1: Parameters used in PSO

Parameter name	Meaning	Default value
Population size(P)	Total number of candidate	30
No.of particles in the swarm(q)	Dimension of the problem/ No.of parameters to be optimized	2
Total no.of generations(G)	No.of iterations	30
Inertia Constant(n), acceleration constants(cg,cq)	Velocity update parameters	0.8,1.7,1.7
Initial velocity value(vmin)	Lower boundary limit of particle velocity	0

Table 3.2: Step-wise enhancement results of proposed framework for Aerial Image 1.

Metrics						
Each Stage	**NIQMC**[9]	**BIQME**[10]	**MICHELSON**[11]	**DE**[12]	**EME**[13]	**PIXDIST**[14]
Input Image	3.8518	0.4056	0.0008	6.9188	7.4905	20.4753
stage 1	4.7004	0.5073	0.0219	7.2976	15.4438	21.9989
stage 2	5.2305	0.6105	0.1513	7.6709	25.2839	29.5533
stage 3	5.4136	0.6456	0.3817	7.8024	55.0252	33.2337

3.3.1 Dataset Used

In the simulation, the different test aerial remote sensing images were used from dataset1 and dataset2. These aerial datasets were procured from USC-SIPI Image database (source of dataset1: **http://sipi. usc.edu/database/database.php?volume=aerials)** and SZTAKI air change benchmark set (source of dataset2: **http://web.eee.sztaki.hu/remotesensing/airchange_benchmark.html)**. First dataset contains total 37 aerial remote sensing images which

[9] No reference Image Quality Metric for Contrast Distortion
[10] Blind Image Quality Measure of Enhanced Images
[11] Michelson Contrast
[12] Discrete Entrophy
[13] Measure of Entrophy
[14] Pixel Distance

were originally stored in the TIFF color format. In this dataset, twelve aerial images were 512×512 and twenty-five aerial images were 1024×1024. Second dataset contain total 12 aerial change detection remote sensing image which were originally stored in the BMP color format and each aerial image are of size 952×640. The proposed technique is tested and validated against a number of existing image quality restoration mehods on remotely sensed aerial image datasets. The image quality restoration methods included for simulation and experimental results comparison are UMFKG [Kwok and Shi (2014)] , RHE- DCT [Fu *et al.* (2015)], IFAIR [Kwok and Shi (2015)], LSCN [Zhan *et al.* (2017)], a method of JEI [Jmal *et al.* (2017)] and MDE algorithm [Suresh and Lal (2017)].

3.3.2 Result Evaluation and Comparison with other Enhancement Methods

In order to the simulation and experimental results assessment, performance evaluation and visual results comparison of proposed framework with other image quality restoration methods on different aerial image datasets are presented. The proposed framework and other existing image quality restoration methods are tested on 49 aerial remote sensing images from the above mentioned databases, but visual results of only 06 aerial images are presented in this Section. The numerical performance comparison of different image quality restoration methods is presented in terms of image quality metrics such as NIQMC [Gu *et al.* (2016)], BIQME [Gu *et al.* (2017)], MICHELSON [Michelson (1995)], DE [Shannon (1948)], [Shin and Park (2015)], EME [Suresh *et al.* (2018)] and PIXDIST [Chen *et al.* (2006)]. The stage wise results of the proposed framework is presented in Table 3.2. Higher its numerical value better is the quality restoration method. The values for different parameters utilized in the PSO are listed in the Table 3.1. Table 3.3 and Table 3.4 depict the

Table 3.3: Average performance comparison of different techniques on Aerial Image Dataset 1

	Quality Metrics					
Algorithms	**NIQMC**[9]	**BIQME**[10]	**MICHELSON**[11]	**DE**[12]	**EME**[13]	**PIXDIST**[14]
UMFKG	4.5450	0.5307	0.0588	7.5166	16.9545	28.2191
RHE-DCT	4.9611	0.5835	0.0642	7.4685	18.3882	25.3302
IFAIR	4.8026	0.5208	0.1237	7.2823	20.583	22.2855
LSCN	4.9651	0.6072	0.0964	7.4228	24.1191	24.2948
JEI	3.9011	0.4492	0.0013	7.0743	7.9222	20.2816
MDE	3.9587	0.4209	0.0009	6.9014	5.9131	22.4380
Proposed Technique	**5.3429**	**0.6139**	**0.2537**	**7.7080**	**41.12225**	**30.4094**

average value of evaluated quality metric values for all the image quality restoration methods compared on aerial image dataset1 and dataset2. From the Table 3.3 and Table 3.4, it can be seen that performance metrics of proposed framework on aerial image dataset1 and dataset2 are better as compared to other existing image quality restoration methods on aerial image dataset1 and dataset2. Further, simulation and experimental qualitative results of the

Table 3.4: Average performance comparison of different techniques on Aerial Image Dataset 2

	Quality Metrics					
Algorithms	**NIQMC**[9]	**BIQME**[10]	**MICHELSON**[11]	**DE**[12]	**EME**[13]	**PIXDIST**[14]
UMFKG	5.1461	0.5847	0.2617	7.4714	35.6825	25.2336
RHE-DCT	5.1419	0.5977	0.1169	7.458	22.1714	25.4332
IFAIR	4.8884	0.5676	0.1518	7.2917	22.7625	22.4501
LSCN	4.7361	0.5045	0.0759	7.2692	17.1117	21.8005
JEI	5.2434	0.5898	0.0659	7.6116	17.0655	28.5676
MDE	4.1863	0.4270	0.0158	7.0601	9.2292	20.6029
Proposed Technique	**5.4479**	**0.6256**	**0.2756**	**7.7492**	**35.888**	**30.889**

proposed framework are analyzed against other existing image quality restoration methods like UMFKG, RHE-DCT, IFAIR, LSCN, JEI, and MDE on 06 aerial images from two aerial image datasets.

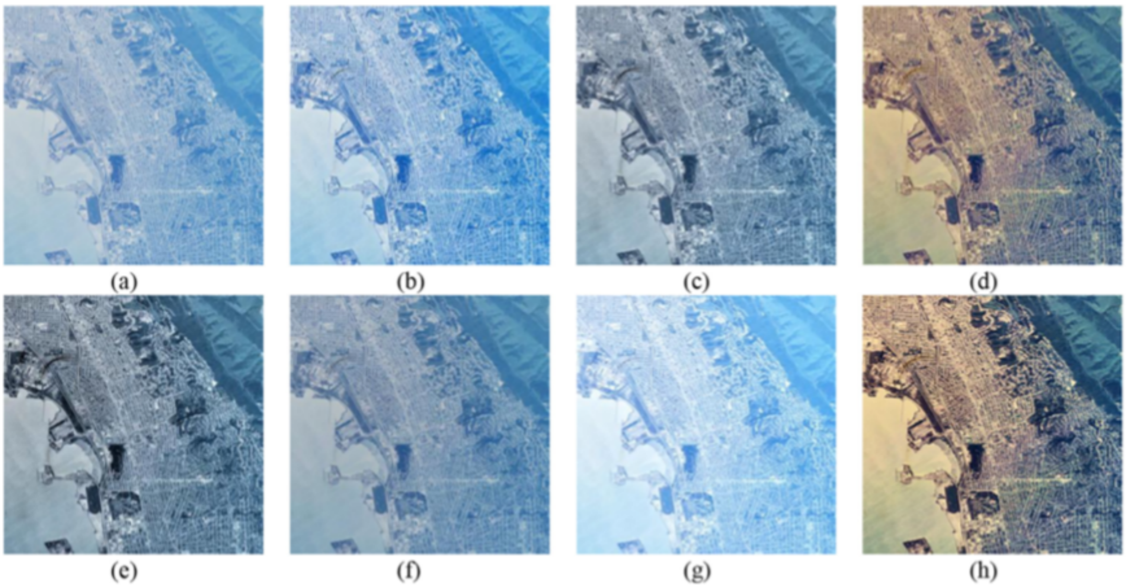


Figure 3.2: Visual results of different quality restoration methods for aerial image 3.2(a). (a) Original image, (b) UMFKG, (c) RHE-DCT, (d) IFAIR. (e) LSCN, (f) JEI, (g) MDE, (h)Proposed Technique.

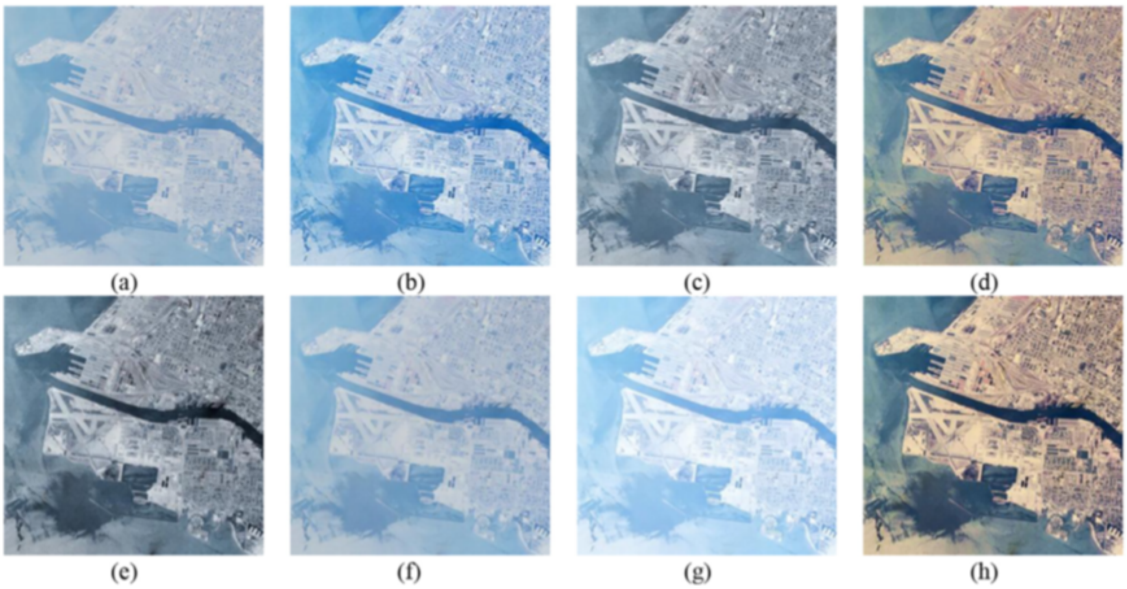


Figure 3.3: Visual results of different quality restoration methods for aerial image 3.3(a). (a) Original image, (b) UMFKG, (c) RHE-DCT, (d) IFAIR. (e) LSCN, (f) JEI, (g) MDE, (h)Proposed Technique.

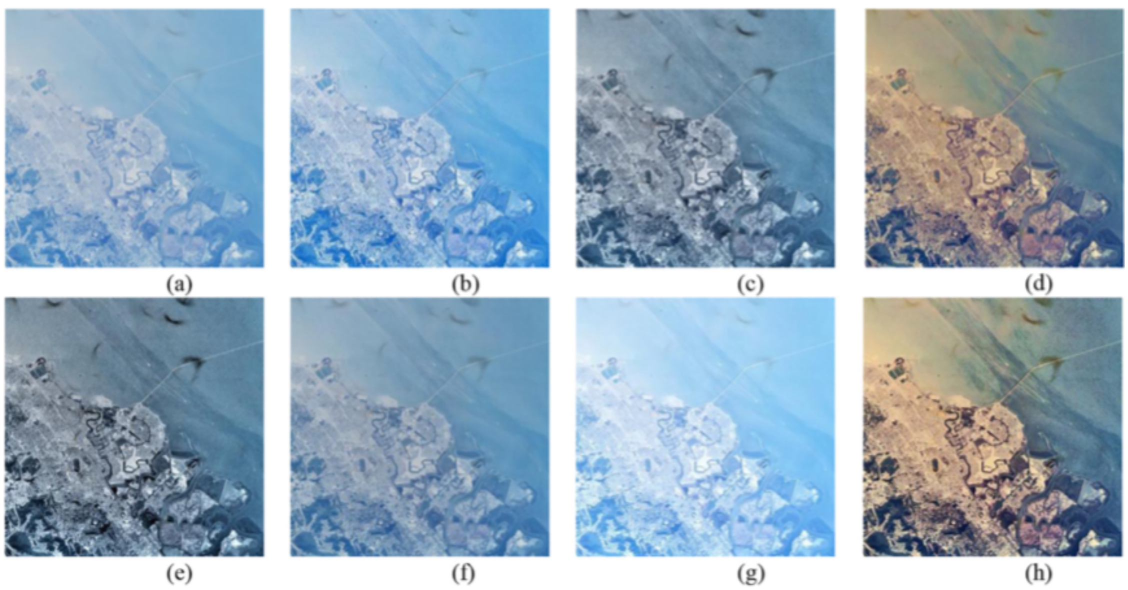


Figure 3.4: Visual results of different quality restoration methods for aerial image 3.4(a). (a) Original image, (b) UMFKG, (c) RHE-DCT, (d) IFAIR. (e) LSCN, (f) JEI, (g) MDE, (h)Proposed Technique.

The visual restoration results of proposed technique and other existing image quality restoration methods are given in Figures. [3.2 – 3.7].

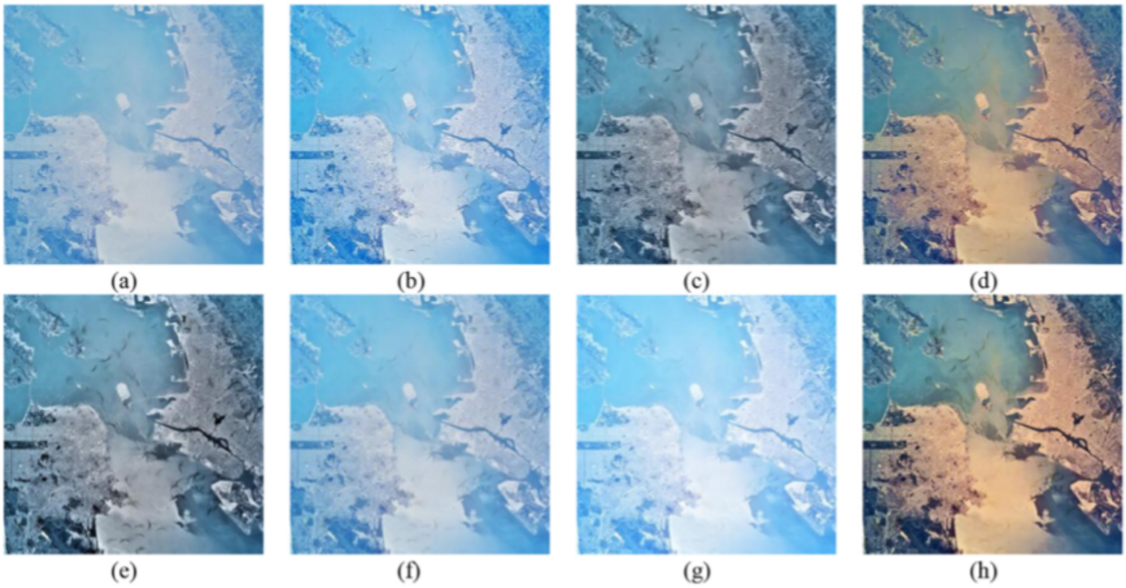


Figure 3.5: Visual results of different quality restoration methods for aerial image 3.5(a). (a) Original image, (b) UMFKG, (c) RHE-DCT, (d) IFAIR. (e) LSCN, (f) JEI, (g) MDE, (h)Proposed Technique.

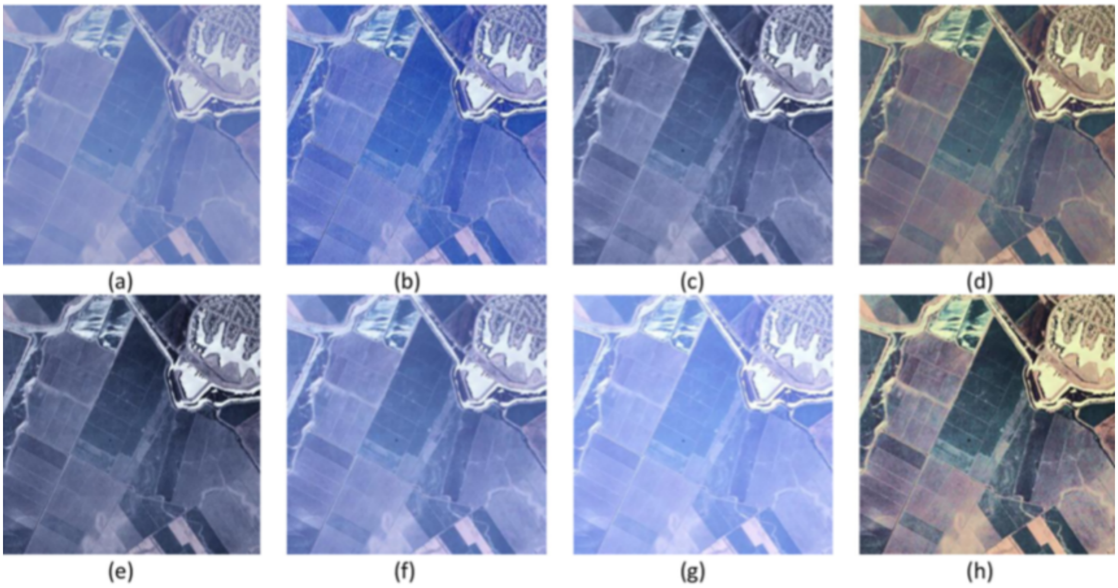


Figure 3.6: Visual results of different quality restoration methods for aerial image 3.6(a). (a) Original image, (b) UMFKG, (c) RHE-DCT, (d) IFAIR. (e) LSCN, (f) JEI, (g) MDE, (h)Proposed Technique.

From the Figures. [3.2 – 3.7], it can be seen that visual restoration result of UMFKG method does not provide good result. Whereas visual quality restoration results of RHE-DCT

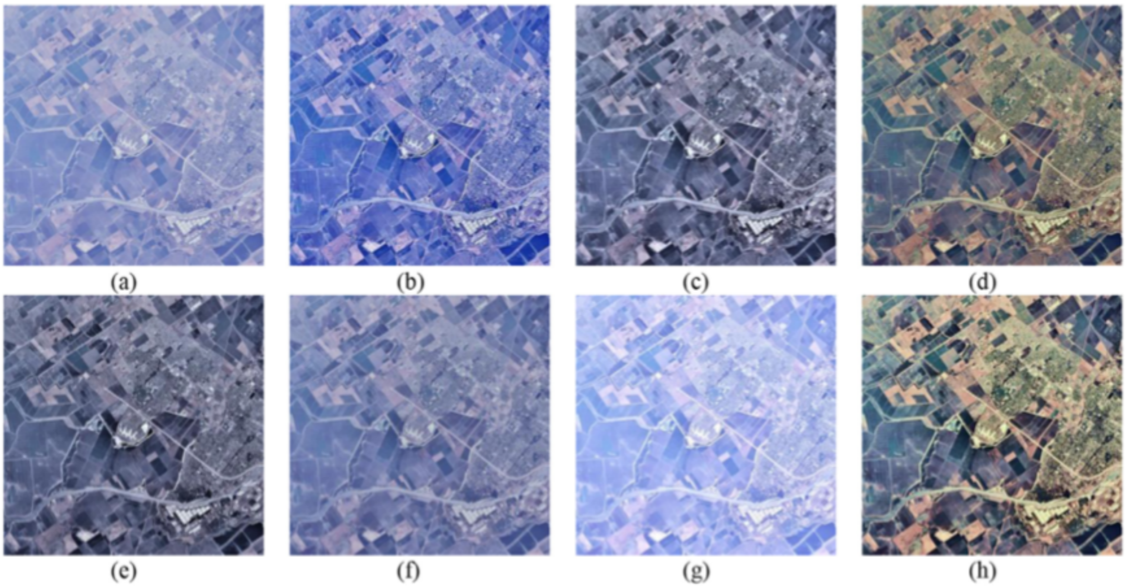


Figure 3.7: Visual results of different quality restoration methods for aerial image 3.7(a). (a) Original image, (b) UMFKG, (c) RHE-DCT, (d) IFAIR. (e) LSCN, (f) JEI, (g) MDE, (h)Proposed Technique.

method provided better result as compared to UMFKG method but still other details of the image are not very lucid. However, visual quality restoration of IFAIR method provided little improved visual results as compared to UMFKG and RHE-DCT methods but still naturalness is missing in the output image. Further, visual quality restoration of LSCN method provided little improved visual results with cost of information loss at the edges and it also provided faded colors in the output image. The LSCN method also amplifying noisy pixels which introducing small ringing effect because of the use of high pass filter. The visual quality restoration of JEI method has provided better output results but still naturalness is missing in the output image. Whereas, the visual quality restoration of MDE method provided an unnatural output image. Therefore, based on the comparison of simulation and experimental results with the other existing image quality restoration methods, the proposed technique has provided better visual and quantitative results.

3.4 Summary

This work highlighted proposed framework for contrast enhancement of aerial images. In this proposed method, firstly, the color cast of the images is removed by exponential alignment of the color values to the mean values. Followed by color balancing, the saturation values of the images are adjusted globally. Then to the resultant images, the PSO algorithm is applied to enhance the contrast of the images with the objective of increasing the entropy of the image and reducing the number of over ranged pixels. To demonstrate the effectiveness of the proposed framework, the different performance quality parameters were evaluated on different aerial image datasets. The simulation and experimental results were also evaluated and compared with other existing image quality restoration methods. Based on experiment results conducted on various aerial images datasets, suggested that proposed restoration framework provided better numerical value of NIQMC, BIQME, MICHELSON, DE, EME and PIXDIST as compared to other state-of-the-art quality restoration methods. Visual enhancement results comparison proved that the proposed framework provided better quality restoration results as compared to other state-of-the-art enhancement methods. Comparison of CPU processing time also revealed that the proposed restoration framework was computationally efficient as compared evolutionary based enhancement algorithms such as UMFKG and MDE algorithms. Hence, the proposed restoration framework can be used in the pre-processing stage of various applications of image processing.

The detailed fucntionality of the proposed semantic segmentation architectures for objects extraction from aerial images is presented in the Chapter 4 and 5.

Chapter 4

Road Extraction from Aerial Imagery Data

4.1 Introduction

In this Chapter, an efficient architecture is proposed which is inspired by the effectiveness of dense convolutions for feature learning [Huang *et al.* (2017), Jégou *et al.* (2017)] and residuals to achieve progress in learning ability of network [He *et al.* (2016*b*), He *et al.* (2016*a*)] at full resolution. The organization of this Chapter is as follows: The detailed explanation of proposed architecture along with its internal modules is given in the Section 4.2. The description of the dataset used for training of all models, including the particulars of hyper parameters utilized is presented under Section 4.3. An elaborate discussion about simulation results of all architectures are described in the Section 4.4. Finally Section 4.5 summarizes this work.

4.2 Proposed DRR-Net Architecture

The proposed dense refinement residual network for semantic segmentation of aerial images is presented in Figure. 4.1. The DRR Net is primarily composed of dense refinement residual (DRR) module(s), and the structure of DRR module(s) is presented in Figure. 4.2. In the proposed DRR architecture, each DRR module inherently contains down-sampling (encoder) and up-sampling (decoder) paths. In the encoder of the DRR module, features are extracted at different resolutions by utilizing dense convolutions. Similarly, in the decoder transposed convolutions are used at multiple scales to learn the up-sampling of feature maps together with learned features of the encoder. Residual connections are employed in each DRR module to provide a supervisory signal to successive DRR modules. This supervisory signal is formed by adding initial features on which each DRR module operates and its corresponding learned features. The successive DRR modules of the proposed architecture attempt to improve the predictions of antecedent(s) by operating on their features. The architecture effectively reuses the features through dense, residual connections and also by stacking of individual DRR modules. This leads to an increase in longevity of feature propagation. The resolution at which each DRR module of proposed architecture operates is given by $H \times W \times F$ [1]. The final stage utilises 1×1 convolutions as softmax in order to produce individual class probabilities. The detailed functionality of DRR module utilised in the proposed architecture is described in the following Section.

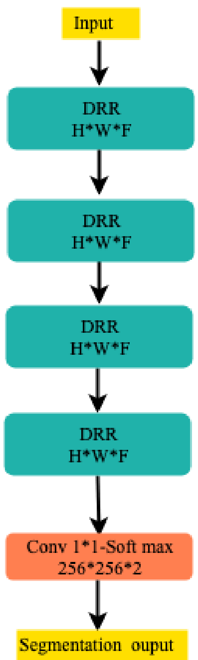


Figure 4.1: The proposed architecture for semantic segmentation of aerial imagery data

[1] H, W are height and width of the input image respectively and F represents the number of filters used in initial convolution unit of DRR module

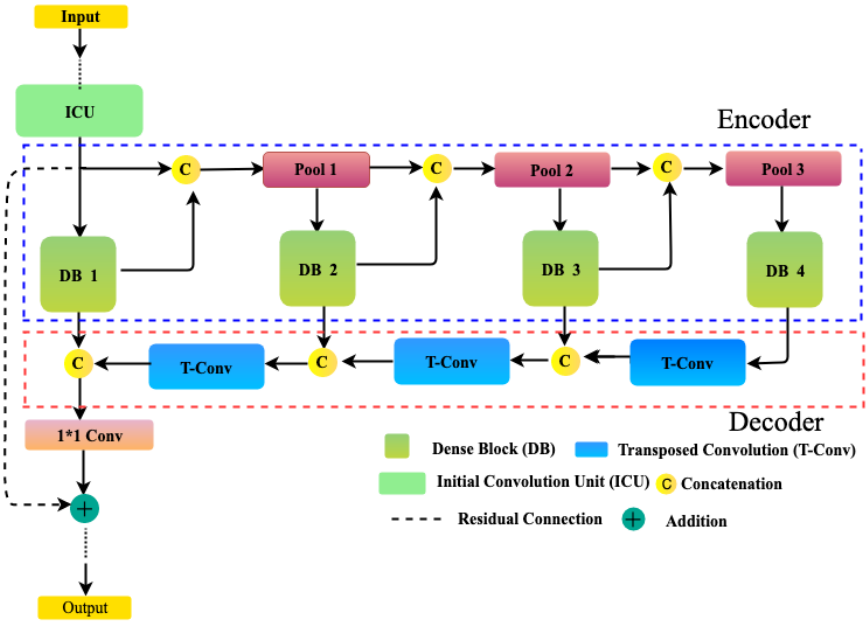


Figure 4.2: Dense Refinement Residual (DRR) module

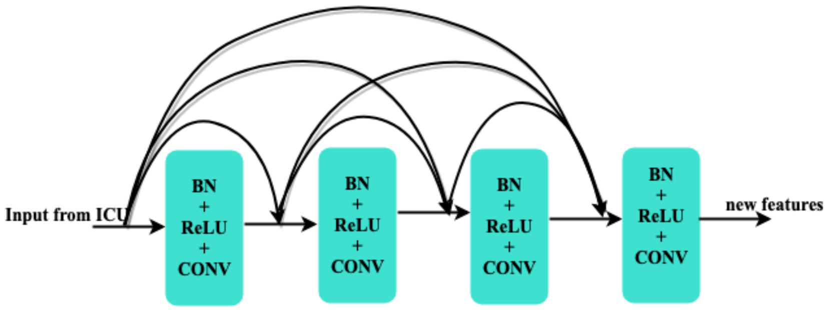


Figure 4.3: Dense Block (DB)

4.2.1 Dense Refinement Residual (DRR) module

The dense refinement residual module of the proposed architecture extracts and up-samples the fine-grained features from the input data. The initial convolution unit (ICU) of first DRR module attempts to learn the initial features from input by applying a sequence of normal convolutions. In the later DRR modules, ICU learns the intermediate feature maps from its preceding DRR module. The structural diagram of dense blocks (DBs) employed in DRR module(s) is represented in Figure. 4.3. This structural module of DBs function on the features of ICU. In each layer of DB, batch normalization (BN)]Ioffe and Szegedy (2015)], rectified linear unit (ReLU) and convolution (CONV) operations are performed by taking all the possible direct connections from its preceding layers. The number of such layers used is 4 with each layer having a growth rate (The number of convolutional filters used) of 16. The learned features of DBs are then passed to successive DBs after pooling. After each level of learning at dense blocks, the feature maps are concatenated with preceding learned features and are also spatially reduced by a factor of two. DB1 predominantly focuses on initial features, and its output is linked with them. The resulting feature maps are max pooled before feeding them to the successive dense blocks. DB2 attempts to learn a different set of feature maps based on DB1 output and initial features. Features extracted out of DB3 are based on the cumulative knowledge of the outputs of DB2, DB1 and initial features. Finally, DB4 extracts high-level features by making use of the collective knowledge accumulated by DRR module up to that point. Feature maps at this level are down-sampled by a factor of eight. From Figure 4.2, it can be seen that the up-sampling process begins at the higher level features of DB4. The up-sampling of feature maps for remaining resolutions is achieved by considering feature maps of preceding dense blocks and learned features of corresponding dense blocks. Residual connections in DRR modules provide a deep supervision to subsequent modules by transferring the combined initial and learned features. Thus, the strength of feature propagation increases due to the effective utilization of feature maps in the encoder and decoder of DRR module.

To summarize, the highlights of the proposed DRR Net are given as follows:

1. Each DRR module of the proposed architecture learns diverse features at various scales with the help of dense blocks.
2. In DRR module, dense blocks at consecutive pool path learn new features based on collective knowledge accumulated by the network.
3. In an up-sampling path of DRR module, transposed convolutions are used instead of dense blocks which results in a great reduction in the number of parameters without comprising the prediction accuracy.
4. Predictions are refined by stacking multi-scale context successively at full resolution.
5. The proposed DRR Net provides a guided learning path to successive DRR modules with establishment of residual connections in each module.
6. The depth of feature maps remain constant, though multiple DRR modules are appended sequentially. Thus avoiding feature map explosion.

7. The proposed architecture provides competitive results with a tenfold reduction in the number of parameters as compared to other existing semantic segmentation architectures.

8. The proposed architecture provides increased flexibility to append or efface number of DRR modules based on computational budget and accuracy.

4.2.2 Refinement Stage in DRR-Net

Let N denotes number of DRR modules, X_i are initial features extracted out of Initial Convolution Unit (ICU),and X_{ij} are the features of Dense blocks at different resolutions and Y_i are predictions or segmentation maps of i DRR module respectively, where $i \in [1, N]$ and $j \in [0, 3]$
Similarly X'_{ij} denote the up sampled features learned at i^{th} DRR module at different resolutions, where $i \in [1, N]$ and $j \in [1, 3]$, X_{11}, X_{12}, X_{13} are the features learned at Pool 1, Pool 2 and Pool 3 respectively in first DRR module, and X'_{11}, X'_{12}, X'_{13} are the features up sampled at Pool 1, Pool 2 and Pool 3 respectively in first DRR module.
Learned features from DB 1, DB 2, DB 3 and DB 4 are given in Eq. (4.1)

$$\left.\begin{aligned} X_{10} &= H\{X_i\} \\ X_{11} &= H\{X_{10}, X_i\} \\ X_{12} &= H\{X_{11}, X_{10}, X_i\} \\ X_{13} &= H\{X_{12}, X_{11}, X_{10}, X_i\} \end{aligned}\right\} \tag{4.1}$$

Where H represents sequence of Batch Normalization, ReLU, convolution operations performed in layers of Dense Blocks at different scales.
Further, up-sampled feature maps at Pool 3, Pool 2, and Pool 1 respectively are given in Eq. (4.2).

$$\left.\begin{aligned} X'_{13} &= F\{X_{13}\} \\ X'_{12} &= F\{X'_{13}, X_{12}\} \\ X'_{11} &= F\{X'_{12}, X_{11}\} \end{aligned}\right\} \tag{4.2}$$

Where F represents transposed convolution operation for up sampling of feature maps.
The output from first module DRR is given in the Eq (4.3).

$$Y_1 = F^{'}\{X^{'}_{11}, X_{10}\} + Xi \tag{4.3}$$

Here $F^{'}$ define the non linearity applied due to 1×1 convolutional filters.
In the same way if multiple DRR modules (consider number of modules (N)as 4) are connected consecutively its corresponding outputs are given in the Eq. (4.4).

$$\left.\begin{aligned} Y_2 &= F^{'}\{X^{'}_{21}, X_{20}\} + Y_1 \\ Y_3 &= F^{'}\{X^{'}_{31}, X_{30}\} + Y_2 \\ Y_4 &= F^{'}\{X^{'}_{41}, X_{40}\} + Y_3 \end{aligned}\right\} \tag{4.4}$$

Finally, Y_4 can be re-written as by substituting Y_3, Y_2, Y_1 values recursively, which is presented in Eq. (4.5).

$$Y_4 = F^{'}\{X^{'}_{41}, X_{40}\} + F^{'}\{X^{'}_{31}, X_{30}\} + F^{'}\{X^{'}_{21}, X_{20}\} + F^{'}\{X^{'}_{11}, X_{10}\} + Xi \tag{4.5}$$

Here the successive DRR module operates on output previous DRR module(s) and also on initial features at which it operated. From Eq. (4.5) it concludes that learned features and predictions are effectively reused in the path of encoder, decoder and also at various modules.

Without Residual: If multiple DRR modules (N=4) are connected consecutively without residual connections, its corresponding outputs are given in the Eq. (4.6).

$$\left.\begin{aligned} Y_1 &= F^{'}\{X^{'}_{11}, X_{10}\} \\ Y_2 &= F^{'}\{X^{'}_{21}, X_{20}\} \\ Y_3 &= F^{'}\{X^{'}_{31}, X_{30}\} \\ Y_4 &= F^{'}\{X^{'}_{41}, X_{40}\} \end{aligned}\right\} \tag{4.6}$$

The successive DRR module has no information of initial features at which previous DRR module being operated.

4.3 Training and Implementation

The proposed DRR architecture has been trained and evaluated by utilizing the Massachusetts roads dataset published in [Mnih and Hinton (2010)]. Each image is composed of 1500x1500 pixels covering an area of 500 square km at a resolution of 1.2 m/pixel. The details about the dataset utilized for road extraction is presented in Section 4.3.1 and the mathematical description of the loss function is presented in Section 4.3.2. The details regarding the ablation study of DRR Net are presented in Section 4.3.3.

4.3.1 Image Dataset

In this work, we consider aerial images that contain less than 50 per cent of white noise. Each resulting image is divided into thirty-six patches of size 256×256 pixels by padding with zeros instead of taking random crops. Thus, we generated 49,680 training, 1008 validation and 3528 test images including masks. The dataset was enlarged by applying horizontal, vertical flips and also brightness variations of different degrees at the time of training. The proposed DRR Net and state-of-the-art architectures were trained using TensorFlow [Abadi *et al.* (2016)] as a deep learning framework with an NVIDIA Tesla k80 GPU with 11GB on-board memory. The initial learning rate was set to 0.0002 and decayed exponentially by a factor of 0.994. The weights of convolution filters were initialized with Xavier initialization [Glorot and Bengio (2010)] . The optimal weights of filters are calculated during backpropagation by using the Adam optimizer [Kingma and Ba (2014)] . The optimizer has an exponential decay rate value of 0.99 for first-order momentum ($\beta 1$) and 0.999 for second-order momentum ($\beta 2$) respectively. All models are trained for 24,8400 iterations with a batch size of 2. The inference of all trained models is performed using an Intel central processing unit (CPU) [2].

4.3.2 Proposed Composite Loss Function

A binary cross entropy loss function (BCE) calculates the loss based on prediction probabilities of each pixel. The BCE loss value is high for false predictions and low for true predictions. Since the data set is highly skewed (it contains $\sim$ 96% background pixels and $\sim$ 4 % road pixels) the model bias towards background pixels frequently results into higher loss values. Hence, during the training phase, the semantic segmentation architec-

[2] Intel Xeon Processor E5-2650 v4@2.20 GHz

tures take a long time to learn and also to converge. The jaccard index or Intersection over Union (IOU) for semantic segmentation is evaluated by considering the overlap of pixels between the predicted image and its mask. This reduces the bias towards the most frequent classes and it is also a useful metric for evaluating the performance of semantic segmentation. The Lovasz softmax loss (LZS) is proposed by Berman *et al.* (2018) as a mean to optimize the mean Intersection over Union by considering a collection of pixel predictions. The combination of binary cross entropy and Lovasz softmax loss is utilized in experiments to improve the pixel-wise classification accuracy of intended objects. The mathematical expression of composite loss function is represented in Eq. (4.7).

$$L_{composite} = L_{BCE} + L_{LZS} \tag{4.7}$$

where L_{BCE} is binary cross entropy loss and L_{LZS} is Lovasz softmax loss. Following the definition of cross entropy mathematical expression for L_{BCE} is written in Eq. (4.8).

$$L_{BCE} = \frac{-1}{N} \sum_{i=1}^{N} [y^{(i)} \ log(\widetilde{y^{(i)}}) + (1 - y^{(i)}) \ log(1 - \widetilde{y^{(i)}})] \tag{4.8}$$

Here, $y^{(i)}$ represents the actual class label values, $\widetilde{y^{(i)}}$ denotes the predicted class probabilities after applying the soft max layer, and N denotes the total number of training samples in the dataset. Following [Berman *et al.* (2018)], the L_{LZS} is given in Eq. (4.9).

$$L_{LZS} = \frac{1}{|C|} \sum_{c \in C} \triangle \overline{Jc} E(c) \tag{4.9}$$

Here $\triangle \overline{Jc}$ is the loss surrogate to the Jaccard index of class c, $E(c)$ is the vector of errors $[0, 1]^p$ and $|C|$ represents the number of classes.

The proposed model has been trained separately with binary cross entropy loss function, Lovasz softmax loss function and also with composite loss function to observe its combination effect. When trained with BCE only, the proposed model took a long time before showing an improvement. When trained with Lovasz softmax loss function, the proposed model showed better performance at earlier iterations but did not maintain the same at later iterations. However, the model trained with combination of loss functions maintained its progress over the iterations. The IOU values of DRR Net when trained with individual loss functions and composite loss function is reported in Figure. 4.4. It can be observed that the proposed architecture trained with composite loss function ($BCE + LZS$) yields better IOU values as compared to other two loss functions.

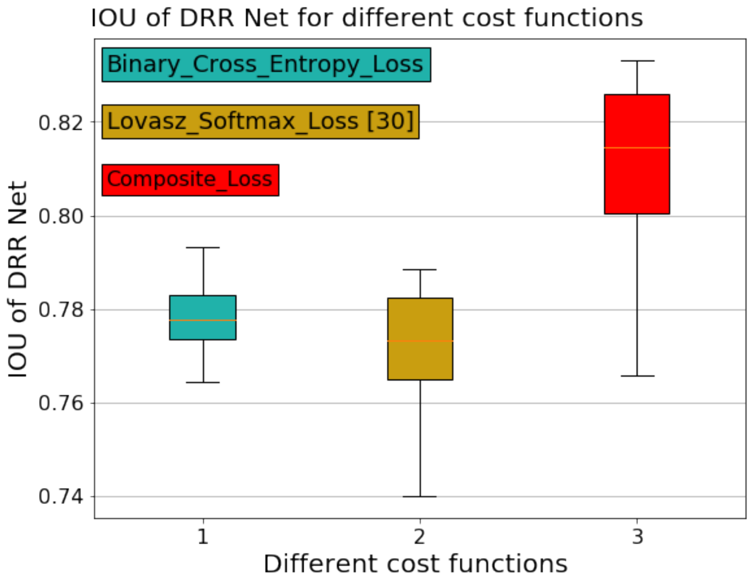


Figure 4.4: Box plot of IOU of proposed model for different loss functions

4.3.3 Ablation Study

To show the importance of residual connections in the proposed DRR Net the model has been trained by removing the residual connection. Due to the removal of residual connections, there is no sharing of initial features of each module to successive DRR modules of the proposed architecture. This leads to a reduction in the learning ability of network. The prediction results of the proposed architecture with and without residual connections are presented in Figures 4.5 and 4.6 along with input and ground truth images. From these predicted images it can be seen that the DRR Net clearly distinguished the road pixels better than the DRR Net without residual connections. The quality metrics of the proposed Net with and without residual are quantified in Table 4.1. These values reveal that the residual connections play a vital role in producing better IOU, road accuracy, precision and recall values. The number of parameters of DRR Net and the one without residual connection remain the same. After observing all predicted images of DRR Net, it is clear that the model precisely differentiated pixels of smaller, curved and parallel roads from background pixels. In addition to this the proposed architecture provided good separation of roads when background pixels are the majority in number.

(a) Input Test Aerial Image **(b)** GroundTruth

(c) DRRNet(W/o Residual) **(d)** DRRNet(Proposed)

Figure 4.5: Predicted images of DRR Net with and without residual

(a) Input Test Aerial Image **(b)** GroundTruth

(c) DRRNet(w/o Residual) **(d)** DRRNet(Proposed)

Figure 4.6: Predicted images of DRR Net with and without residual

4.4 Simulation Results and Discussion

The performance of DRR Net is compared with other existing objects segmentation networks in Section 4.4.1. The details regarding the computational complexity of DRR Net and other compared segmentation architectures are presented in the below Section 4.4.2.

4.4.1 Results Evaluation and Comparison with Other Methods

The proposed model and some of the semantic segmentation architectures are trained with the same hyper parameters and the loss function is considered as the composite loss function. The number of training iterations is the same for all models. The proposed DRR Net does not depend on any pre-trained weight set and it is instead trained end-to-end. To perform comparative analysis, quality metrics such as IOU, Road accuracy, Precision and Recall values, are evaluated at the end of every group of 12,420 iterations and also at the end of the training phase. The quality metrics are obtained by considering test aerial images as input. Road accuracy and mean IOU values are considered to measure the variability of these performance metrics. Figures. 4.7 and 4.8 represent the box plots of road accuracy and mean IOU values of semantic segmentation architectures.

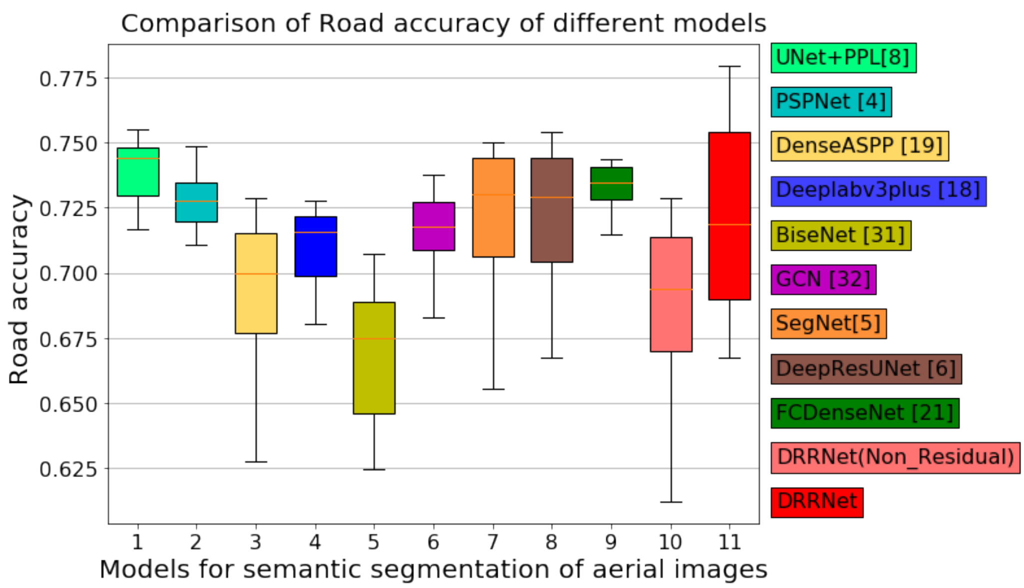


Figure 4.7: Box plot of Road accuracy of models

From Figure. 4.7 one can observe that the proposed DRR Net produces a wide range of

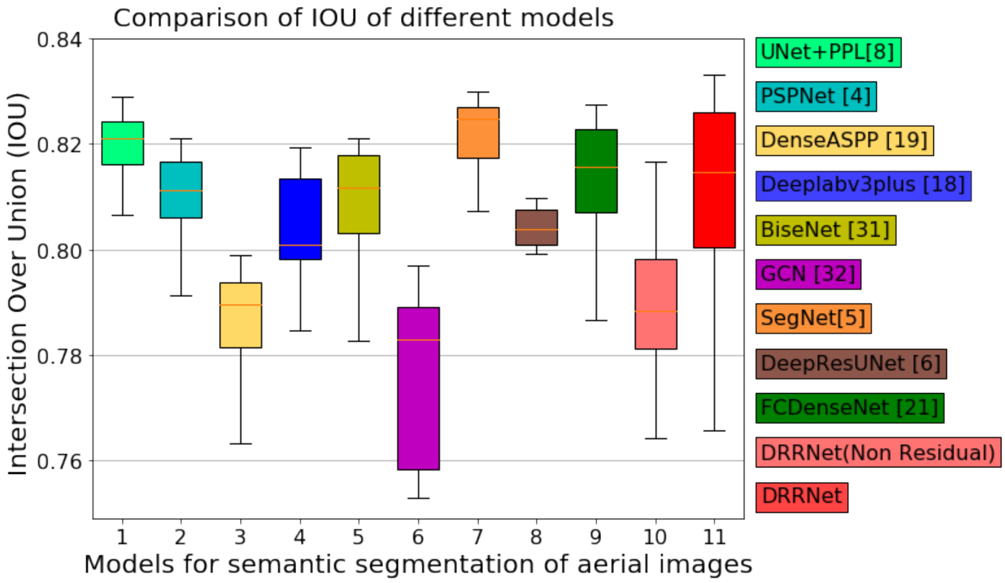


Figure 4.8: Box plot of Intersection Over Union of different models

road accuracy values. Additionally, it can be observed that the proposed model provides a 11.19% improvement over its initial value to reach a maximum value. This is comparable with other models and implies that the proposed model has good learning ability when compared with other architectures. Another measure to quantify a semantic segmentation technique is IOU or Jaccard index. IOU estimates the percentage of pixel overlap between semantic map and its corresponding ground truth. Figure. 4.8 reports that the proposed DRR model and the model presented by Peng *et al.* (2017) exhibit the same higher level of IOU variability. The proposed model reaches a maximum IOU value from an initial overlap of 76.57 per cent between predicted and ground truth image. In addition to this one can observe that the models Deep LabV3+ [Chen *et al.* (2018)], FC-DenseNet [Jégou *et al.* (2017)] and BiseNet [Yu *et al.* (2018)] possess a narrow range of IOU values. Further, box lengths of the remaining models is observed to be smaller. Table 4.1 lists the parameters of models and their corresponding performance metrics. The performance metric are evaluated by inferring the models at the end of the training. From Table 4.1, considering the number of trainable parameters the descending order of models is given by UNet+PPL [3] [Kim *et al.* (2019)], PSPNet [He *et al.* (2014)], DeepLabV3+ [Chen *et al.* (2018)], BiseNet [Yu *et al.* (2018)], DenseASPP [Yang *et al.* (2018)], GCN [Peng *et al.* (2017)], SegNet [Badrinarayanan *et al.* (2015)], FC-DenseNet [Jégou *et al.* (2017)], Deep ResUNet [Zhang

[3]Modified version of original architecture

Table 4.1: Quality metrics comparison of semantic segmentation techniques for aerial images

Mode	**PreTrained**	**Mean IOU**	**Road Accuracy**	**Precision**	**Recall**	**# Parameters** (in million)
PSPNet He *et al.* (2014)	√	0.820	0.7483	0.9827	0.9795	55.90
DeepLabV3Plus Chen *et al.* (2018)	√	0.819	0.7276	0.9839	0.9800	47.95
BiseNet Yu *et al.* (2018)	×	0.797	0.7073	0.9815	0.9765	47.41
Dense ASPP Yang *et al.* (2018)	√	0.799	0.7384	0.9820	0.9771	43.39
GCN Peng *et al.* (2017)	×	0.821	0.7376	0.9844	0.9801	42.99
FC-DenseNet Jégou *et al.* (2017)	×	0.829	0.7434	0.9846	0.9812	9.26
DeepResUNet Zhang *et al.* (2018*b*)	×	0.8304	0.7520	0.9841	0.9812	7.85
UNet+PPL Kim *et al.* (2019)	×	0.829	0.7434	0.9846	0.9812	56.49
SegNet Badrinarayanan *et al.* (2015)	×	0.822	0.7320	0.9841	0.9812	34.96
DRR Net(Non-Residual (N=4)) (Proposed)	×	0.816	0.7287	0.9848	0.9808	2.63
DRR Net (N=4) (Proposed)	×	**0.833**	**0.7794**	**0.9805**	**0.9792**	**2.63**

et al. (2018*b*)] and DRR Net. The order implies that UNet+PPL [Kim *et al.* (2019)] model requires maximum number of trainable parameters while the proposed model has least number of trainable parameters. Thus, the road accuracy of DRR Net is significantly superior to other models which also showed discrimination in the corresponding Precision and Recall values. The parameters of different models together with proposed DRR Net are represented in Bar graph which is shown in Figure. 4.9 and it reveals that the proposed DRR Net have far fewer parameters (2.63 million) compared to other models.

Few of the considered test aerial images are shown in Figures. 4.10a, 4.11a, 4.12a and 4.13a. For the test aerial images in Figure. 4.10a, 4.11a, 4.12a and 4.13a, the segmentation maps produced by proposed and the state-of-the-art architectures along with ground truth images are presented in Figure. [4.10b - 4.10l], [4.11b - 4.11l], [4.12b - 4.12l] and [4.13b - 4.13l] respectively. To highlight the performance of the DRR Net, its predicted images are compared with state-of-the-art-models by highlighting some parts of the image with red

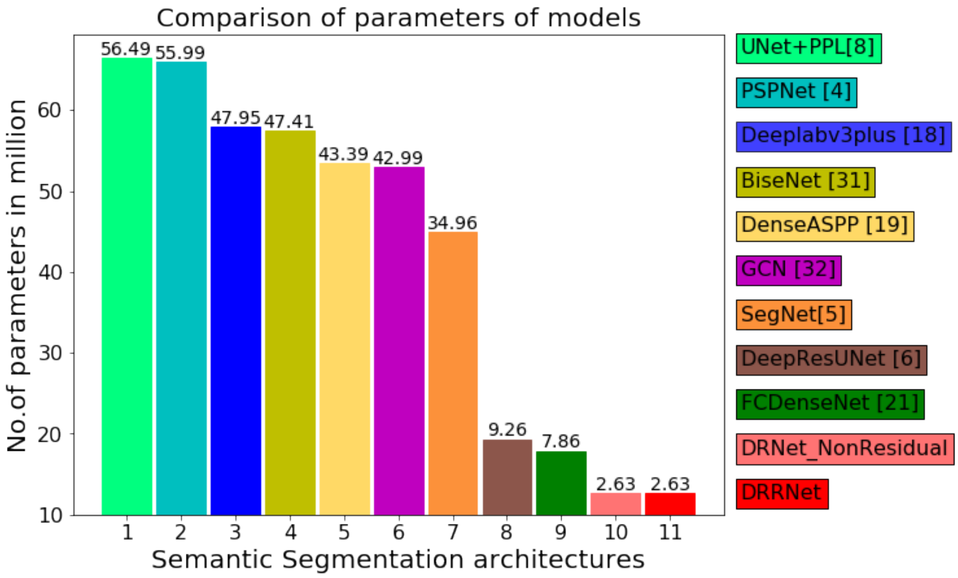
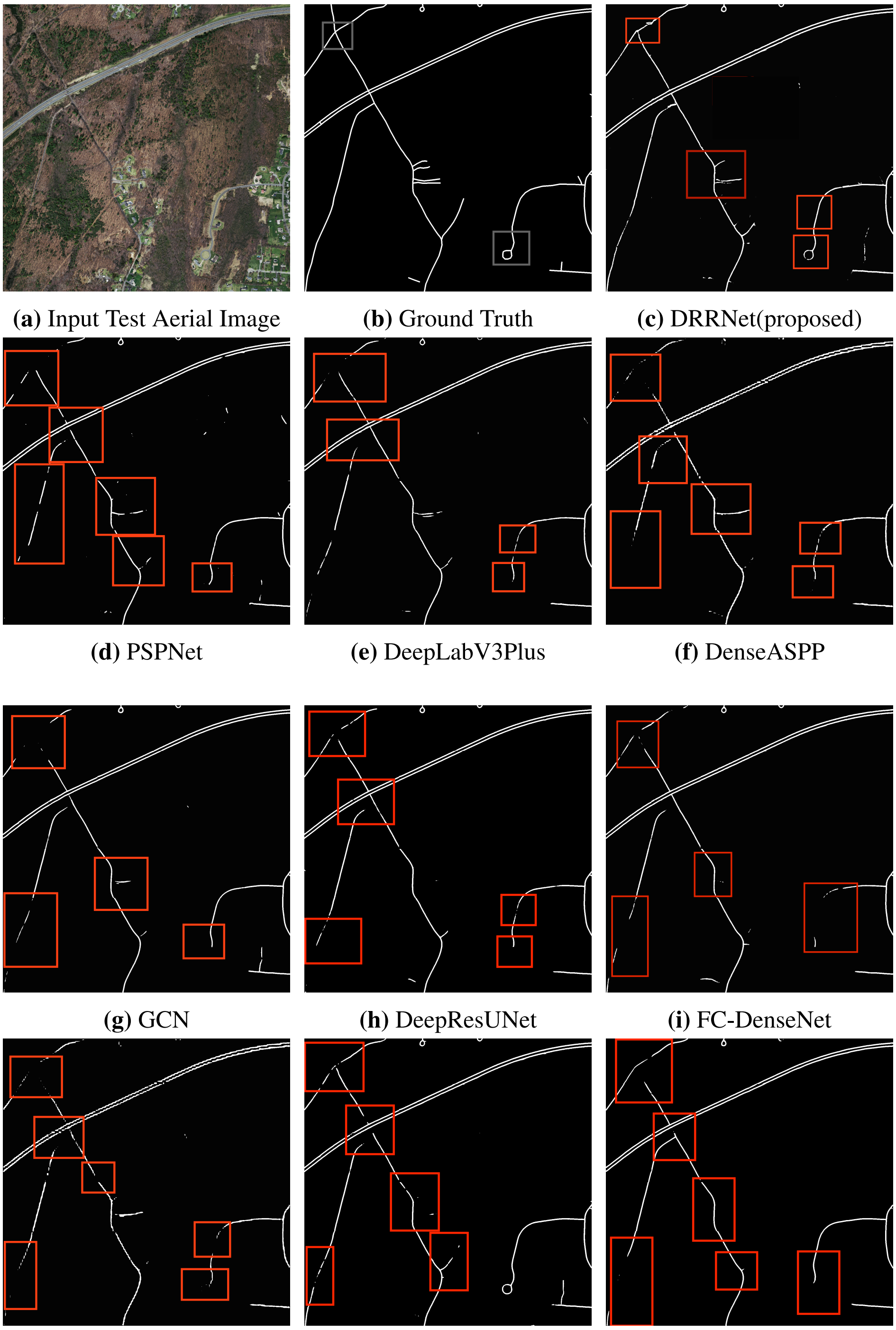


Figure 4.9: Bar graph for parameters of different models

colour boxes. Figure. [4.10b - 4.10l] represents the segmentation output of all models for the test input aerial image of Figure. 4.10a. From these predicted images, it can be seen that the DRR Net extracts round-shaped roads and also the intersections of roads without any gap. The predicted images of the test aerial input images Figures 4.11a, 4.12a and 4.13a are shown in Figure. [4.11b - 4.11l], [4.12b - 4.12l] and [4.13b - 4.13l]. These images reveal that the proposed DRR Net differentiates the parallel, smaller and diverse-shaped road regions clearly from other regions.



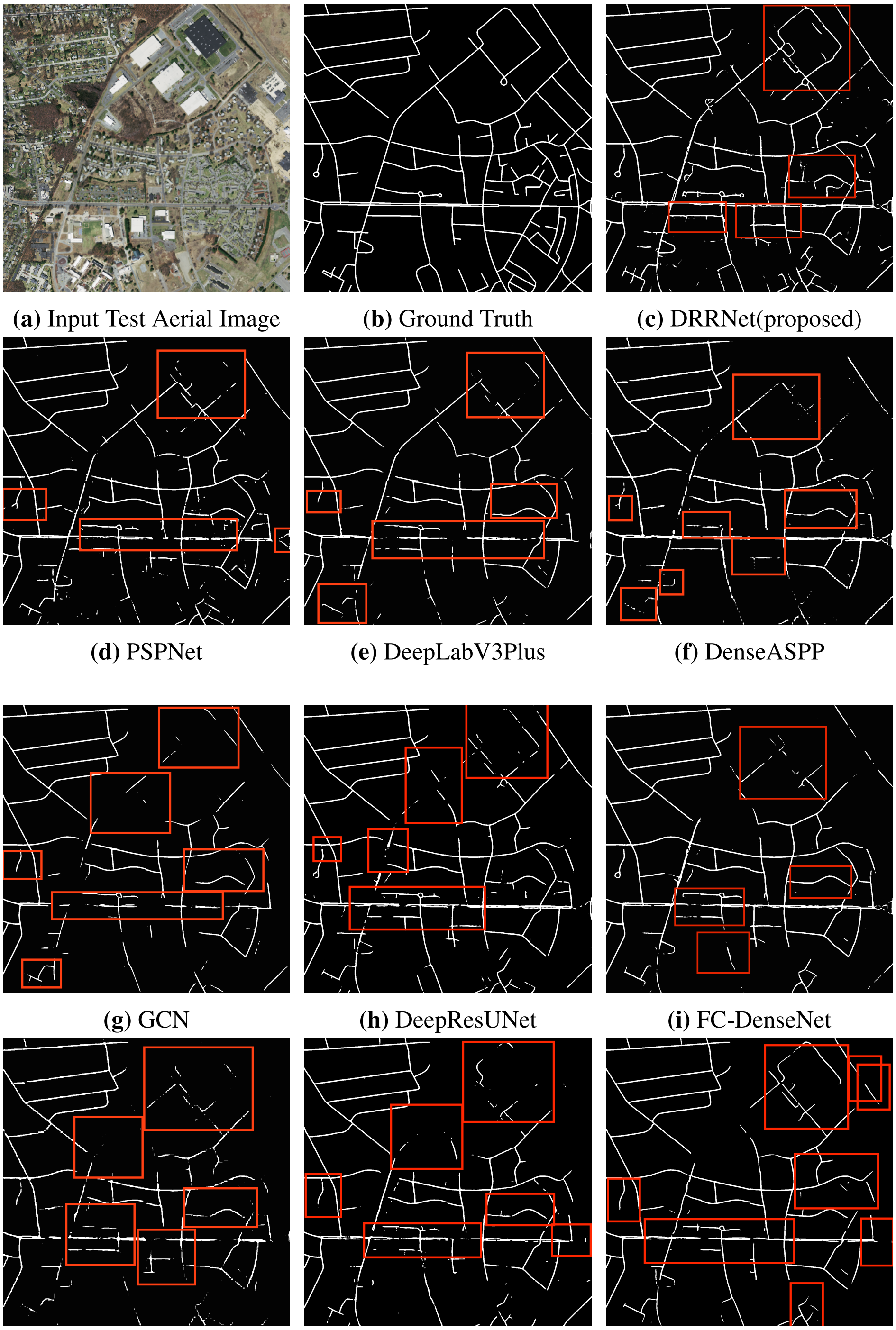
(a) Input Test Aerial Image **(b)** Ground Truth **(c)** DRRNet(proposed)

(d) PSPNet **(e)** DeepLabV3Plus **(f)** DenseASPP

(g) GCN **(h)** DeepResUNet **(i)** FC-DenseNet

(j) BiseNet **(k)** UNet+PPL **(l)** SegNet

Figure 4.10: Predicted images of semantic segmentation models of Figure 4.10a



(a) Input Test Aerial Image (b) Ground Truth (c) DRRNet(proposed)

(d) PSPNet (e) DeepLabV3Plus (f) DenseASPP

(g) GCN (h) DeepResUNet (i) FC-DenseNet

(j) BiseNet (k) UNet+PPL (l) SegNet

Figure 4.11: Predicted images of semantic segmentation models of Figure 4.11a

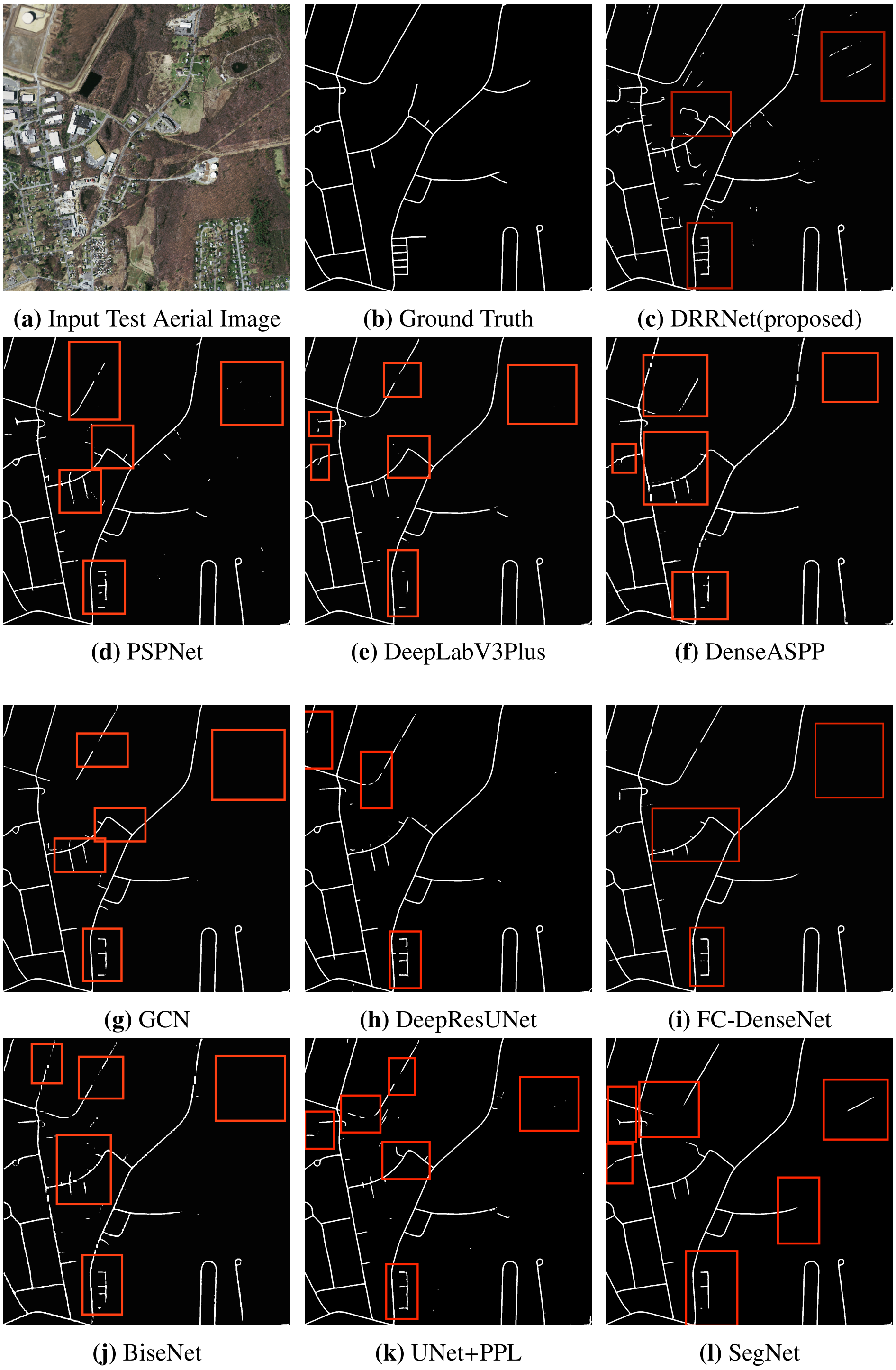


Figure 4.12: Predicted images of semantic segmentation models of Figure 4.12a

(a) Input Test Aerial Image

(b) Ground Truth

(c) DRRNet(proposed)

(d) PSPNet

(e) DeepLabV3Plus

(f) DenseASPP

(g) GCN

(h) DeepResUNet

(i) FC-DenseNet

(j) BiseNet

(k) UNet+PPL

(l) SegNet

Figure 4.13: Predicted images of semantic segmentation models of Figure 4.13a

4.4.2 Computational Complexity Analysis

In this Section, an elaborate discussion of computational complexity of all architectures including the proposed architecture is presented.

Table 4.2: Comparison of FLOPs, Training and average Test run time of all models

Model	Training time per image(sec)	Total Training time(Hours)	Average Test run time(sec)	FLOPs(in billion)
PSPNet He *et al.* (2014)	0.32	48	0.41	62.5
DeepLabV3Plus Chen *et al.* (2018)	0.26	37	0.31	32.8
BiseNet Yu *et al.* (2018)	0.34	46.35	0.32	20.4
Dense ASPP Yang *et al.* (2018)	0.19	28	0.15	22.1
GCN Peng *et al.* (2017)	0.375	52	0.395	20.9
FC-DenseNet Jégou *et al.* (2017)	0.51	70.63	4.17	52.3
DeepResUNet Zhang *et al.* (2018*b*)	0.39	57.81	0.094	77.6
UNet+PPL Kim *et al.* (2019)	0.65	89.5	1.30	144.1
SegNet Badrinarayanan *et al.* (2015)	0.54	75.4	0.916	90.0
DRR Net(Non-Residual (N=4))	0.63	86.8	4.66	80.1
DRR Net (N=4)	**0.63**	**86.8**	**4.66**	**80.1**

In Table 4.2, the total training time (per-image and also for all images of the dataset), the average test run time and the number Floating point operations (FLOPs) of all models are presented. The total training time is defined as the time taken to train individual architectures. The average test run time is defined as the average time required to infer the trained model over the total number of test images. It can be observed that, pretrained network architecture based models such as Dense ASSP [Yang *et al.* (2018)], Deeplabv3+ [Chen *et al.* (2018)] , PSPNet [He *et al.* (2014)], BiseNet [Yu *et al.* (2018)] and GCN [Peng *et al.* (2017)], require comparatively less training time than other models. The proposed DRR model and the model presented by [Jégou *et al.* (2017)] are built with dense convolutions. Because of the concatenation of features from the specified number of convolutional layers, these models need longer training time as contrary to other models. In the DeepResUNet model proposed by Zhang *et al.* (2018*b*), the training time is considerably reduced due to presence of residual connections. Due to the concatenation of feature maps after pooling with different scales, the UNet+PPL [Kim *et al.* (2019)] model demands increased training time as contrary to other models. The total time allocated to train all models is 678 hours. Referring to Table 4.2, from the average test run time of all models, FC-DenseNet [Jégou *et al.* (2017)], DRR Net requires a longer time to load the learned weights during a forward pass from dense convolutions of the trained model. The increasing order of computational complexity of models (in terms of FLOPs) is BiseNet [Yu *et al.* (2018)], GCN [Peng *et al.* (2017)] , DenseASSP [Yang *et al.* (2018)], DeepLabV3+ [Chen *et al.* (2018)], FC-DenseNet [Jégou *et al.* (2017)], PSPNet [He *et al.* (2014)], DeepResUNet [Zhang *et al.* (2018*b*)] , DRR Net (proposed) , SegNet [Badrinarayanan *et al.* (2015)] and U-NetPPL [Kim *et al.* (2019)]. The proposed DRR Net ranks third in increasing computational complexity order. The time complexity of DRR Net is high as compared to other models, as it is built with dense convolutions which involves concatenation of features from all possible previous layers. Due to this, the model need higher training and test times to load the weights from multiple paths of the proposed DRR Net. In addition, the computational complexity of the two models (DRR Net , DRR Net with out residual connection) remains the same in terms of training time, test time and FLOPs, which are presented in Table 4.2.

4.5 Summary

A semantic segmentation architecture named DRR Net was presented to segment roads in high resolution aerial imagery data. The proposed DRR Net is composed of multiple DRR modules to learn new features based on collective knowledge and to refine the predictions. Each DRR module of the proposed DRR Net provides guidance to its successive modules without increasing the depth of its feature maps.The proposed architecture was able to precisely segment roads and achieved prominent results on Massachusetts roads dataset as compared with state-of-art semantic segmentation architectures. The qualitative and quantitative results showed that the DRR Net could segment all kinds of roads including variable extent roads and also the non labelled roads. Comparison of proposed model has been done with the diversified semantic segmentation models based on normal convolutions, atrous convolutions, Global convolutions and Dense convolutions. Among all models the proposed model showed remarkable performance in all aspects including background dominant scenarios.The distinctive performance of proposed architecture can be attributed to the iterative reuse of collective knowledge acquired at various scales through dense, residual connections and the connectivity of DRR modules. The proposed architecture achieved ~2.74% increase in road accuracy with tenfold reduction in number of parameters. Further, the model provided well discrimination of roads in all scenarios.

Chapter 5

Objects Segmentation from Aerial Imagery Data

5.1 INTRODUCTION

In this Chapter, a robust encoder and decoder semantic segmentation architecture, namely O-SegNet for multi object segmentation from aerial imagery data is presented. This architecture utilizes attention mechanism [Zhang *et al.* (2018*a*)] at feature learning and reconstruction stages to produce dense predictions without an increase in complexity. The predictions are improved by modeling the relationship between pixels in the global view by incorporating attention-aware features and precedent features.

The organization of this Chapter is as follows: The functionality of the proposed O-SegNet architecture, and it's internal blocks are presented in Section 5.2. The mathematical analysis of proposed O-SegNet architecture and its variations is presented in Section 5.3. The dataset description, pre-processing operations, and training details are provided in Section 5.4. The simulation results of different variations of proposed architecture are presented in Section 5.5. The details about the simulation results and computational complexity of proposed and other compared semantic segmentation architectures are described in Section 5.6. Finally, summary of this work is presented in Section 5.7.

5.2 Proposed O-SegNet Architecture

The proposed O-SegNet (Objects Segmentation Network) architecture for Objects Segmentation through semantic segmentation task from Aerial Imagery Data is presented in Figure 5.1. The proposed O-SegNet architecture is composed of feature learning (encoder),

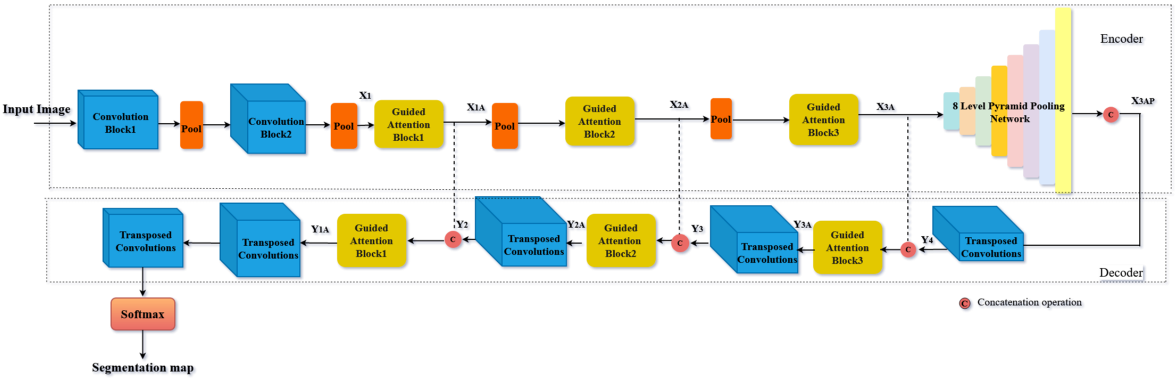


Figure 5.1: The proposed O-SegNet architecture for road extraction from aerial imagery data

context aggregation, and reconstruction stages (decoder) to extract diverse shaped objects present in the aerial images. The feature learning path contains convolution blocks, Guided Attention (GA) blocks, and max-pooling layers. The structural diagram of the GA blocks employed in the proposed architecture is shown in Figure 5.2. The intermediate features are learned through two sets of convolution blocks, and these features are then allowed to pass through GA blocks. The attention mechanism is applied to these features through Self-Attention (SA) module present in the first GA block. The structural diagram of the SA module employed in GA blocks is shown in Figure 5.3. These obtained attentive-aware features are combined with previous layer features. These resultant features are provided as input to successive GA block after reducing spatially by a factor of two. This process repeats for the other two resolutions.

The encoder GA blocks produce discriminative feature representation by accumulating attentive-aware and precedent learned features. In addition to this, the GA block at one resolution scale guides the following GA blocks of the encoder to model the inter-relationship between the pixels. The learned feature representation from the encoder is then allowed to pass through an 8-Level Pyramid Pooling Network (PPN). In 8-Level PPN, after performing multi-level pooling, features are aggregated along the depth dimension. Due to the concatenation of different pooled versions of encoder features, the global context of the objects is extracted. Thus the extracted global context is beneficial to provide correct predictions irrespective of shape variations of objects.

The aggregated features from an 8-Level PPN are provided as input to the reconstruction path. The transposed convolutions are applied to recover the original resolution from the lower-dimensional features obtained from the encoder. During each stage of reconstruction, the up-sampled features at a given resolution are concatenated with the corresponding encoder features. These combined features are then allowed to pass through GA blocks at the decoder. Here, the attention mechanism is incorporated between up-sampled and encoder features. The combination of attentive-aware features and precedent up-sampled features are provided as an input to following GA block of the decoder. This process of up-sampling and providing attention between encoder and decoder features repeats further for two resolutions. Due to the attention mechanism applied between up-sampling and down-sampling paths, the decoder produces the correct segmentation map by providing emphasis to the required features of the encoder. The different number of convolutional filters are used in each GA blocks of the encoder, specifically 256 number of filters at $f/4$, $f/8$ resolutions, and 512 number of filters at $f/16$ resolution. Similarly, the same pattern is followed for transposed convolutional filters at the decoder.

Finally, 1x1 convolutions are applied as a softmax to extract the class probabilities of the intended classes.

The novelties of the proposed architecture are as follows.

- The proposed O-SegNet semantic segmentation architecture effectively utilizes a Self-Attention mechanism at the encoder and decoder to model spatial dependencies present in the input aerial imagery data.
- The introduced Guided Attention (GA) blocks are utilized to establish inter-relationship between features at different scales.
- Each GA block guides the successive GA blocks at later resolutions in the paths of encoder and decoder via residual connections.
- The encoder features are aggregated through an 8-Level Pyramid Pooling Network to extract the global context.
- The proposed network introduces an attention between encoder and decoder to focus on relevant contextual information while producing predictions.

The detailed functionalities of GA, SA modules, and an 8-Level PPN are presented in Sections 5.2.1, 5.2.2, and 5.2.3, respectively.

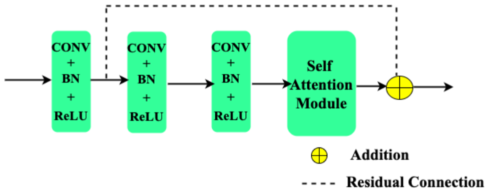


Figure 5.2: Guided Attention Block

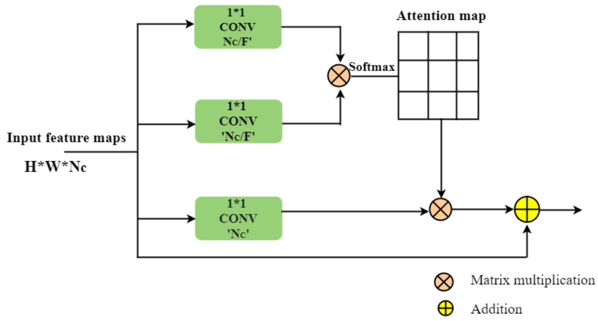


Figure 5.3: Self-Attention Module

5.2.1 Guided Attention Block

The GA block employed in the proposed O-SegNet architecture contains three convolutional layers, a Self-Attention module, and a residual connection. In each convolutional layer, the sequence of operations are performed namely, Convolution, Batch Normalization (BN) , and Rectified Linear Unit (ReLU) . The learned features from three convolutional layers are fed to the SA module to represent these features as a weighted combination of the entire neighbourhood. This weighted representation from the SA module and precedent layer features are combined through residual connection. These combined features are provided as an input to the following GA block at later resolution to explore the relationship between features of different resolutions. Hence, the rich contextual information is extracted through modeling of inter-relationship by considering features of GA blocks at previous resolutions.

5.2.2 Self-Attention Module

The Self-Attention (SA) module is incorporated in the GA block to encode a wide range of contextual information by re-expressing of input features. The input to a SA module is learned features from convolutional layers in the GA block. These input features are re-formulated as a weighted combination of the neighborhood spatial regions by summing with the same considered input features. This summation is required to keep track of the positional information of features that are considered. Weights for the summation are calcu-

lated via element-wise matrix multiplication between two feature spaces. These two feature spaces are obtained after applying 1x1 convolutions to input features. The total number of 1x1 convolutions applied is expressed as C/F, where C indicates the depth of input features and [1] F represents the factor. The values of C and F are different for each SA module of the GA block at various resolutions in the encoder and decoder. The detailed mathematical explanation for the functionality of the SA module is presented in the below Se ction.
Let f, g and v are the features obtained after applying C/F and C number of $1x1$ convolutions. The input features are denoted as X_i, the dimension of input features are H, W, C , where H, W, C represents height, width, and depth of features. The attention scores or weights (γ) are calculated by applying element-wise matrix multiplication between f, g. These scores are normalized through soft max operation, it is given in Eq. (5.1).

$$\gamma = \sigma[f^T \otimes g] \tag{5.1}$$

where T, σ, $\otimes$, and γ indicate transpose, softmax, element-wise matrix multiplication operation, and attention scores/weights, respectively.
The re-formulation of input features X_i as a weighted combination of neighborhood spatial regions is expressed in Eq. (5.2).

$$X_k = \gamma \otimes v + X_i \tag{5.2}$$

where X_K are re-formulated features of X_i.

5.2.3 8-Level Pyramid Pooling Network

The output features from SAR block 3 of the encoder are considered as input to an 8-Level PPN. The spatial resolution of these features is $\frac{1}{16}^{th}$ of input resolution. These features are allowed to pass through eight different pooling layers, with the pooling sizes of 1, 3, 5, 8, 13, 14, 15, and 16 respectively. The finest and coarsest pooling levels are 1 and 16 respectively. These pooled features are concatenated along the depth dimension and provided as an input to the up-sampling path. With 8 different pooling sizes, features from various ranges are aggregated to obtain the global context.

[1] F values for SA module of GA block 1, 2, and 3 of encoder and decoder are 4, 4, and 8, respectively

5.3 Mathematical Analysis of the Proposed Architecture

5.3.1 O-SegNet Architecture

The mathematical representation of GA blocks present at the encoder and decoder of Figure 5.1 is presented in the below Section.

Encoder: The input features provided as input to the GA block 1 of encoder is denoted as X_1.

Let X^i_{jA} are features learned at i^{th} convolutional layer of j^{th} GA block and X_{jA} denote the output feature representation of j^{th} GA block at the encoder, where $i, j \epsilon \{1, 2, 3\}$.

The attention scores/weights (γ_k) are calculated from the SA module of each GA block using the Eq. 5.1, where $k \, \epsilon \, \{1 \; to \; 6\}$.

The features extracted from j^{th} GA block of encoder are calculated from Eq. 5.2, and represented using the below mentioned Eq. (5.3), (5.4) and (5.5).

$$X_{1A} = \gamma_1 \otimes v_1 + X^3_1 + X^1_1 \tag{5.3}$$

$$X_{2A} = \gamma_2 \otimes v_2 + X^3_{1A/2} + X^1_{1A/2} \tag{5.4}$$

$$X_{3A} = \gamma_3 \otimes v_3 + X^3_{2A/2} + X^1_{2A/2} \tag{5.5}$$

The values of γ_1, γ_2, and γ_3 are calculated by applying Eq. (5.1), and v_1, v_2, and v_3 are obtained after applying $1x1$ convolutions to features X^3_1, X^3_{1A}, and X^3_{2A} respectively. Here $X_{1A/2}$ and $X_{2A/2}$ are inputs to GA block 2 and GA block 3, respectively. The pooling operation is denoted as $/2$.

The features extracted from 8-Level PPN are represented as X_{3AP}, which are drawn from X_{3A}.

Decoder: The up-sampled features using transposed convolutions are denoted as Y_i. Y_{jA} denotes the output feature representation of j^{th} GA block and Y^i_j are learned features at i^{th} convolutional layer of j^{th} GA block, where $j \epsilon \; \{3, 2, 1\}$, *and* $i \epsilon \; \{4, 3, 2, 1\}$.

Let Y_4, Y_3, Y_2, and Y_1 are up-sampled features of X_{3AP}, Y_{3A}, Y_{2A}, and Y_{1A}, respectively.

The input features to GA block 3, 2 and 1 are Y_4, Y_3, and Y_2 respectively.

The symbol : represents the concatenation operation between two features.

At the decoder, features of j^{th} GA block are calculated from Eq. 5.2, and represented in Eq. (5.6) to (5.8).

$$Y_{3A} = \gamma_4 \otimes v_4 + [Y_4 : X_{3A}]^3 + [Y_4 : X_{3A}]^1 \tag{5.6}$$

$$Y_{2A} = \gamma_5 \otimes v_5 + [Y_3 : X_{2A}]^3 + [Y_3 : X_{2A}]^1 \quad (5.7)$$

$$Y_{1A} = \gamma_6 \otimes v_6 + [Y_2 : X_{1A}]^3 + [Y_2 : X_{1A}]^1 \quad (5.8)$$

The values γ_4, γ_5, and γ_6 are obtained from Eq. (5.1), and v_4, v_5, and v_6 are calculated after applying 1×1 convolutions to input features $[Y_4 : X_{3AP}]^3$, $[Y_3 : X_{2A}]^3$, and $[Y_2 : X_{1A}]^3$, respectively.
It is to say that from Eq. (5.3) to (5.5), at each level of learning, attentive-aware features are combined with previous learned features in the GA blocks, and then these features are utilized at subsequent stages of learning.
The up-sampling process starts by considering aggregated features X_{3AP}. From Eq. (5.6), (5.7), and (5.8) the GA blocks at decoder consider encoder features and reconstructed features of particular resolution.

5.3.2 O-SegNet Variation 4

In this model variation, instead of the multi-level pooling operation, one pooling layer is utilized after GA block 3 of the encoder. These resultant features from GA block 3 of the encoder are supplied for up-sampling path. The input features to the first set of transposed convolutions in the up-sampling path are X_{3A}. So Y_4 is obtained after applying transposed convolutions to X_{3A}. The mathematical expression for GA blocks of the encoder and the decoder are same as in Eq. (5.3) to (5.5), and Eq. (5.6) to (5.8).

5.3.3 O-SegNet Variation 3

During reconstruction path, GA blocks without residual connections are utilized, and at the encoder GA blocks are kept in place. The mathematical representation of the j^{th} GA block of the decoder are obtained from Eq. (5.2), and are given below in Eq. (5.9) to (5.11).

$$Y_{3A} = \gamma_7 \otimes v_4 + [Y_4 : X_{3A}]^3 \quad (5.9)$$

$$Y_{2A} = \gamma_8 \otimes v_5 + [Y_3 : X_{2A}]^3 \quad (5.10)$$

$$Y_{1A} = \gamma_9 \otimes v_6 + [Y_2 : X_{1A}]^3 \quad (5.11)$$

γ_7, γ_8, and γ_9 are calculated from concatenation of encoder and up-sampled features, $[Y_4 : X_{3A}]$, $[Y_3 : X_{2A}]$, and $[Y_2 : X_{1A}]$, and v_4, v_5, and v_6 are calculated from the same features by

applying Eq. (5.1) and 1×1 convolutions respectively.
The Eq. (5.9) to (5.11) indicate that, attention scores/weights are calculated from up-sampled and encoder features. These scores are then passed to successive stages of reconstruction.

5.3.4 O-SegNet Variation 2

The mathematical representation for GA blocks of encoder is same as the O-SegNet architecture. However, encoder features are not considered at GA blocks of the decoder. The feature representation of the j^{th} GA block at decoder are derived from Eq. (5.2), and presented in Eq's. (5.12) to (5.14).

$$Y_{3A} = \gamma_{10} \otimes v_4 + Y_4^3 + Y_4^1 \tag{5.12}$$

$$Y_{2A} = \gamma_{11} \otimes v_5 + Y_3^3 + Y_3^1 \tag{5.13}$$

$$Y_{1A} = \gamma_{12} \otimes v_6 + Y_2^3 + Y_2^1 \tag{5.14}$$

where γ_{10}, γ_{11}, and γ_{12} are calculated by applying Eq. (5.1), and v_4, v_5, and v_6 are derived after 1×1 convolutions have applied to features Y_4, Y_3, and Y_2 respectively.
The feature representation from the encoder is not utilized in the up-sampling path from Eq's. (5.12) to (5.14).

5.3.5 O-SegNet Variation 1

The GA blocks without residual connections are utilized at paths of feature learning and reconstruction in the proposed O-SegNet architecture. The mathematical representation of j^{th} GA block of the encoder and decoder are obtained from Eq. (5.2), and are given below in Eq's. (5.15) to (5.20).

$$X_{1A} = \gamma_{13} \otimes v_1 + X_1^3 \tag{5.15}$$

$$X_{2A} = \gamma_{14} \otimes v_2 + X_{1A/2}^3 \tag{5.16}$$

$$X_{3A} = \gamma_{15} \otimes v_3 + X_{2A/2}^3 \tag{5.17}$$

The values of γ_{13}, γ_{14}, and γ_{15} are calculated by applying Eq. (5.1), and v_1, v_2, and v_3 are obtained after applying 1×1 convolutions to features X_1^3, X_{1A}^3, and X_{2A}^3 respectively.
Finally, the mathematical expression of the j^{th} GA block of the decoder are given in Eq's.

(5.18), (5.19), and (5.20) respectively.

$$Y_{3A} = \gamma_{16} \otimes v_4 + Y_4^3 \tag{5.18}$$

$$Y_{2A} = \gamma_{17} \otimes v_5 + Y_3^3 \tag{5.19}$$

$$Y_{1A} = \gamma_{18} \otimes v_6 + Y_2^3 \tag{5.20}$$

γ_{16}, γ_{17}, and γ_{18} are calculated up-sampled features, Y_4, Y_3, and Y_2, and v_4, v_5, and v_6 are calculated from the same features by applying Eq. (5.1) and 1×1 convolutions respectively.

5.4 Training and Implementation

In this Section, description about Dataset and Pre-processing are presented in Section 5.4.1. The mathematical description of the proposed composite loss function are explained in Section 5.4.2. Further, training details are presented in Section 5.4.3.

5.4.1 Dataset and Pre-processing

The proposed O-SegNet architecture and the other existing semantic segmentation architectures are evaluated by considering the dataset presented in Kaiser *et al.* (2017). The dataset contain high-resolution aerial images of five cities, including the labels for roads and buildings. The number of images in cities of the dataset, as mentioned earlier, is 200 (Berlin), 457 (Chicago), 625 (Paris), 364 (Zurich), 24 (Postdam), and 1 (Tokyo), with labels. The spatial resolution of each city image is different from other cities. Each image of the provided dataset, is divided uniformly into patches of spatial resolution 256×256 without any overlapping between the adjacent patches. The total number of generated images are 1,95,073 for training, 24,853 for validation, and 17,200 for testing. Out of the above generated images of all cities, 35,000 images are selected for training and 4,700 for validation randomly.

5.4.2 Composite Loss Function

The proposed O-SegNet architecture and compared semantic segmentation architectures are trained with the composite loss function in Eerapu *et al.* (2019), which is the sum of multi-class entropy function (L_{MCE}), and Lovasz Softmax function (L_{LS}) [Berman *et al.* (2018)]. The mathematical expression for L_{MCE} and L_{LS} are given in Eq's. (5.21) and (5.22). The composite loss function is utilized to increase pixel-wise accuracy values and IOU values simultaneously.

$$L_{MCE} = -\sum_{i=1}^{N}\sum_{j=1}^{N_c}[y_{ij}\ log(p(y'_{ij})) + (1 - y_{ij})\ log(1 - p(y'_{ij}))] \tag{5.21}$$

where, y_{ij}, y'_{ij}represent label values and predictions of j^{th} class, $p(y'_{ij})$ are predicted class probabilities of j^{th} class after applying the softmax layer, and N, N_c denote the total number of training samples in the dataset, the number of classes, respectively.

$$L_{LS} = \frac{1}{N_c}\sum_{c\epsilon N_c}\triangle\overline{J_c}\,E(c) \tag{5.22}$$

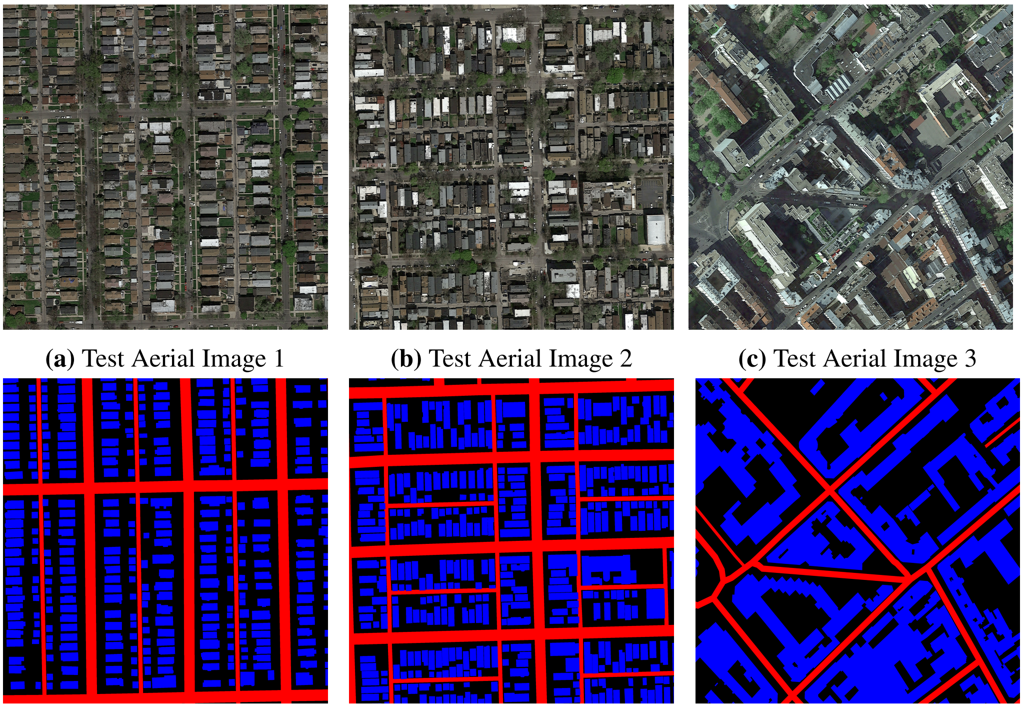
where N_c denote the total number of classes, $\triangle J_c$ provides loss surrogate to the Jaccard index of class c, and the vector of errors are represented as $E(c)$.

5.4.3 Training Setup

The provided road and building dataset have been augmented by employing horizontal, vertical, and brightness variation during training. The proposed O-SegNet architecture and comparison models are simulated by utilizing TensorFlow [Abadi *et al.* (2016)] with an NVIDIA Quadro K2200 GPU having on-board memory of 4GB. The learning rate is initially assumed as 0.001 and decayed exponentially with a rate of 0.994 over the iterations. At beginning of the training, convolutional filter weights are initialized with Xavier initialization [Glorot and Bengio (2010)]. These weights are optimized during back propagation using Adam optimizer [Kingma and Ba (2014)]. The values adapted for Adam optimizer are 0.99 for the exponential decay rate of first-order momentum (β_1) and 0.999 for second-order momentum (β_2). The proposed and the other compared semantic segmentation architectures are trained for 350000 number of iterations with a batch size of 2. At the end of every $17,500$ number of training iterations, validation of the models is performed. For testing, the Intel Central Processing Unit (CPU) is utilized. The composite loss function used to calculate the error between predictions and ground truth is similar to the one presented in [Eerapu *et al.* (2019)]. This loss function is a combination of Binary cross-entropy loss function and Lovasz loss function [Berman *et al.* (2018)].

5.5 Ablation Study

To measure the effectiveness of proposed O-SegNet architecture, its main modules are detached and formed different variants from the architecture. The main modules detached from the proposed architecture are the self-attention module, residual connections in GA blocks of the encoder and decoder, attention mechanism between encoder context, and upsampled features at the decoder, and finally PPN network at the end of the feature learning path. These variants are trained end-to-end with the same training details for the same number of training iterations as the proposed O-SegNet architecture. The quality metrics of proposed architecture variants are evaluated and listed in Table 5.1. This Table also includes the number of parameters and Floating Point Operations (FLOPs) of proposed O-SegNet and its variations to measure the computational complexity. The test aerial images along with ground truth are presented in Figure 5.4. The predicted images of proposed O-SegNet and its variations are shown in Figures 5.5, 5.6, and 5.7.



(a) Test Aerial Image 1 **(b)** Test Aerial Image 2 **(c)** Test Aerial Image 3

(d) Ground Truth 1 **(e)** Ground Truth 2 **(f)** Ground Truth 3

Figure 5.4: Test images with their ground truth

5.5.1 Effect of Attention Mechanism

The GA blocks without residual connections are incorporated in the paths of encoder and decoder in the O-SegNet variation1. The quality metrics of this model variant is compared with the SegNet architecture [Badrinarayanan *et al.* (2017)] to evaluate the effectiveness of attention mechanism. The O-SegNet variation 1 model produced a higher value of 0.8530 for road accuracy but produced a mere identical mean IOU value as compared with the SegNet architecture as shown in Table 5.1. From segmentation maps shown in Figures 5.5f, 5.6f, and 5.7f, it is clear that the O-SegNet variation 1 missed to segment narrow, continuous roads and also produced building pixels in the place of road pixels.

5.5.2 Residual Connections

Here, quantitative and qualitative results of O-SegNet variation 1 and 2 are considered to evaluate the effect of residual connections. In O-SegNet variation 2, GA blocks are utilized in feature learning and reconstruction paths. From the performance metrics of considered model variations as presented in Table 5.1, the residual connections play a vital role in producing a higher value of building accuracy (0.7679) and mean IOU value (0.5547) than the one without residual connection.

5.5.3 Encoder-Decoder Attention

The encoder and decoder features are combined and applied self-attention mechanism to the resultant features in the GA blocks of the decoder. The attention mechanism is applied to the combined features in the decoder of O-SegNet variation 3. However, in variation 2 of proposed architecture, the encoder features are not considered in the up-sampling path. Table 5.1 indicate that, the model with encoder-decoder attention produced a significantly higher value of road accuracy(0.8590), building accuracy(0.7728), and mean IOU value(0.5643), without adding much computational complexity as compared with O-SegNet variation 2.

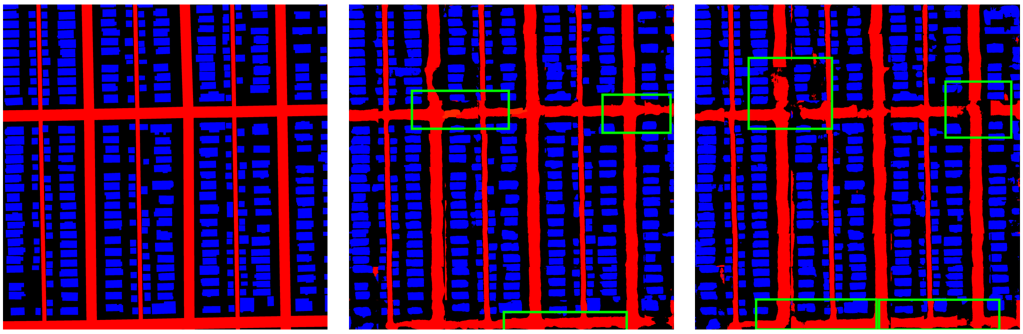
5.5.4 With and With Out PPN

The Pyramid Pooling Network at the feature learning path has dropped from the O-SegNet variation 4 and compared with the results of proposed O-SegNet architecture. The quality metrics presented in Table 5.1 show that the proposed O-SegNet achieved significant gain in class-wise accuracy and IOU as compared with O-SegNet variation 4. The segmentation

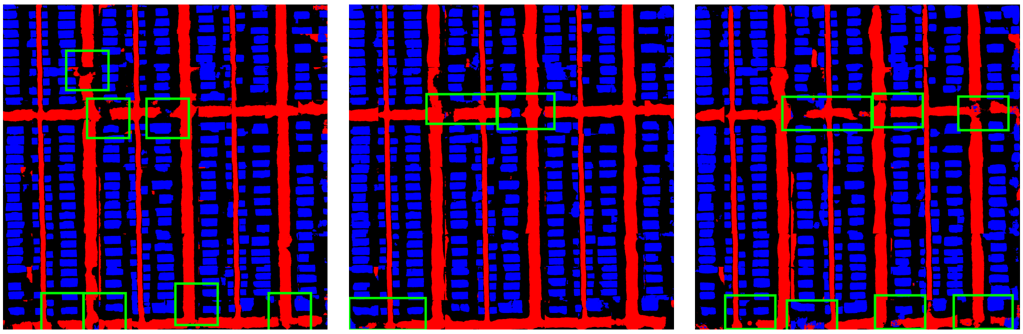
Table 5.1: Average quality metrics comparison of O-SegNet architecture and its variations for Test images.
PPN : Pyramid Pooling Network, **A**: Attention, **AD** : Attention between Encoder and Decoder, **RC**: Residual Connection

Model	PPN	A	AD	RC	Average Test Accuracy	Road Accuracy	Building Accuracy	Mean IOU	Parameters (in million)	FLOPs (in billion)	Total Training time (in hours)
SegNet	×	×	×	×	0.7092	0.8241	0.7337	0.5335	34.96	90.0	126
O-SegNet variation 1	×	✓	×	×	0.7077	0.8530	0.7412	0.5374	22.46	81.5	121
O-SegNet variation 2	×	✓	×	✓	0.6997	0.8583	0.7679	0.5547	26.70	82.4	115
O-SegNet variation 3	×	✓	✓	✓(at encoder)	0.7108	0.8590	0.7728	0.5643	22.16	83.3	121
O-SegNet variation 4	×	✓	✓	✓	0.7125	0.8742	0.7842	0.5713	23.16	84.6	123
O-SegNet (Proposed)	✓	✓	✓	✓	**0.7263**	**0.8880**	**0.8108**	**0.5861**	**42.40**	**83.9**	**126**

maps are presented in Figures 5.5c, 5.6c, and 5.7c, it reveal that O-SegNet architecture segmented roads, and irregular shaped buildings accurately, than compared with the one without PPN. Therefore, PPN contributed a significant role in producing better class-wise accuracies, IOU values and predictions by extracting the global context through different pooling levels. The number of parameters of O-SegNet is significantly high as compared with the other variants. However, the total training time and the number of FLOPs are merely the same as the O-SegNet variation 4. Hence, the computational complexity of the proposed architecture is the same as its model variations.

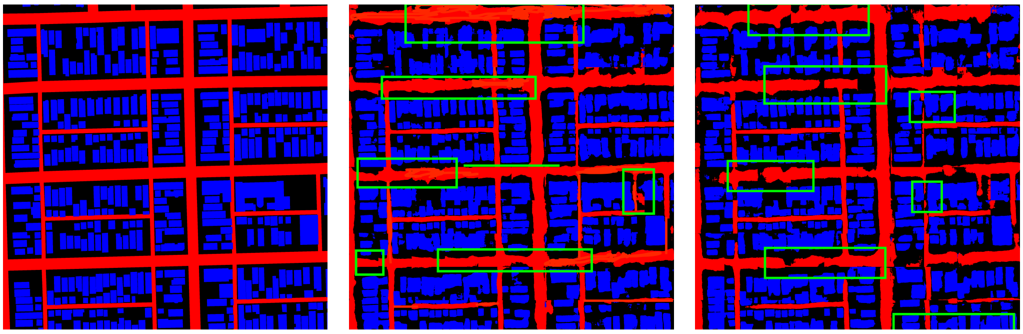


(a) Ground Truth 1 **(b)** O-SegNet(proposed) **(c)** O-SegNet(Variation 4)

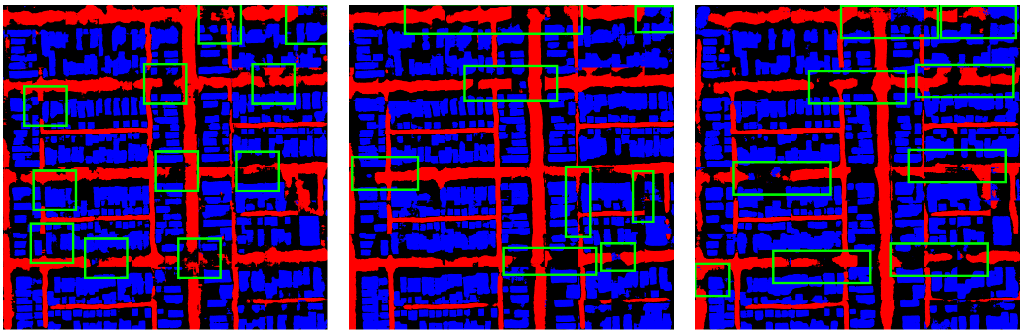


(d) O-SegNet(Variation 3) **(e)** O-SegNet(Variation 2) **(f)** O-SegNet(Variation 1)

Figure 5.5: Predicted image of O-SegNet and its variations for Test Aerial Image 1

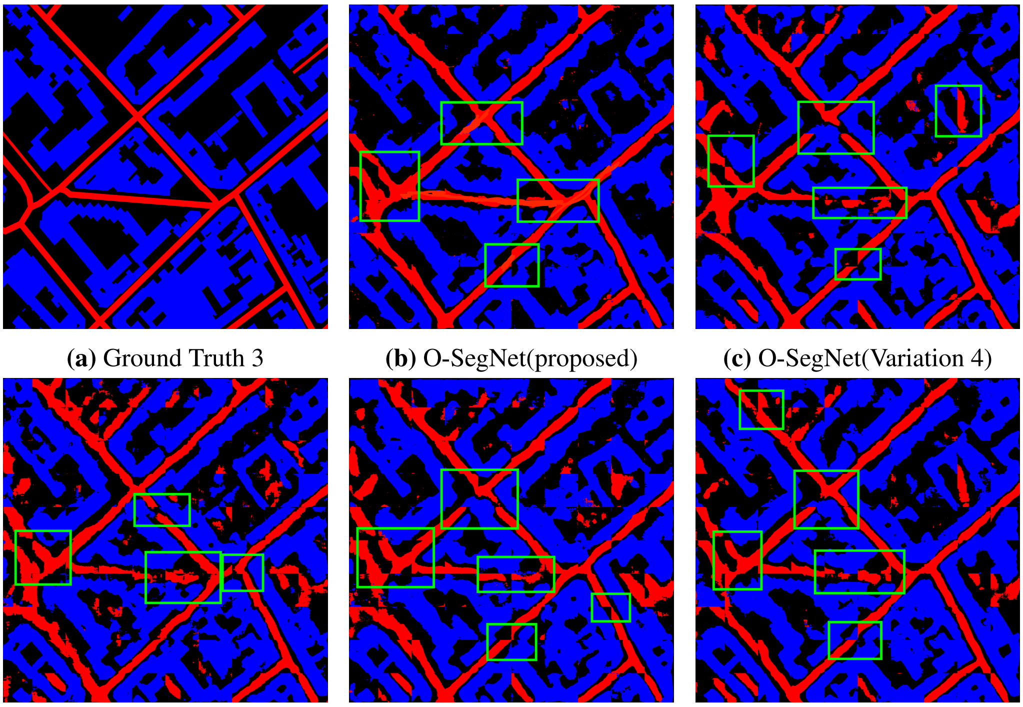


(a) Ground Truth 2 **(b)** O-SegNet(proposed) **(c)** O-SegNet(Variation 4)



(d) O-SegNet(Variation 3) **(e)** O-SegNet(Variation 2) **(f)** O-SegNet(Variation 1)

Figure 5.6: Predicted image of O-SegNet and its variations for Test Aerial Image 2



(a) Ground Truth 3 **(b)** O-SegNet(proposed) **(c)** O-SegNet(Variation 4)

(d) O-SegNet(Variation 3) **(e)** O-SegNet(Variation 2) **(f)** O-SegNet(Variation 1)

Figure 5.7: Predicted image of O-SegNet and its variations for Test Aerial Image 3

5.6 Simulation Results and Discussion

The proposed O-SegNet architecture quality metrics and its predictions are compared with other existing objects segmentation networks, like SCattNet, MiniNet, DeepLabV3Plus, DenseASSP, SegNet, and FCN Variant in Section 5.6.1. The discussions about the computational complexity of proposed architecture and compared segmentation techniques are presented in the below Section 5.6.2.

5.6.1 Results Evaluation and Comparison with other Segmentation Methods

The newly introduced O-SegNet architecture does not consider any pre-trained weights, and it is trained an end-to-end fashion to produce predictions. The performance of models is inferred during training and also at the end of the training. The quality metrics that are measured to evaluate the performance of models are mean Intersection Over Union (IOU) , road accuracy, building accuracy, precision, recall, and average test accuracy. These quality

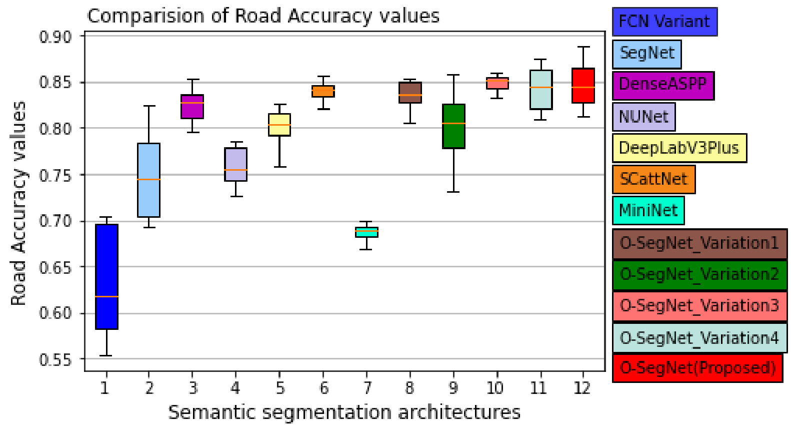


Figure 5.8: Comparison of Road Accuracy values

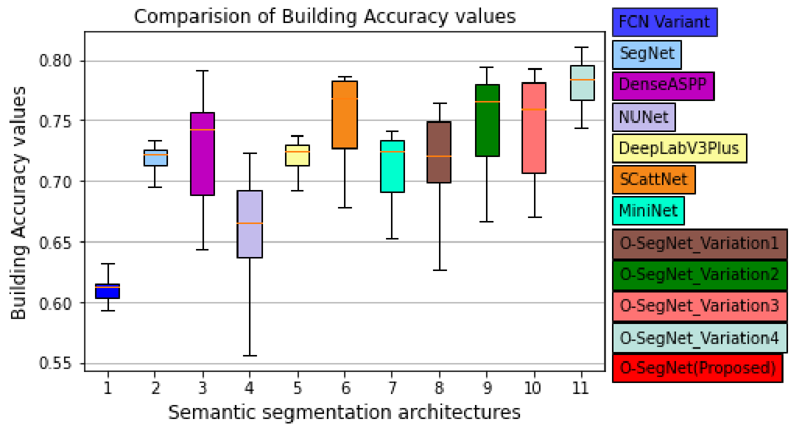


Figure 5.9: Comparison of Building Accuracy values

metrics are computed by considering test images as input to trained semantic segmentation architectures, and these values are listed in Table 5.2. To measure the performance variability of models, the road accuracy, building accuracy, and mean IOU values are represented in box plots. These box plots are shown in Figures 5.8, 5.9, and 5.10, respectively. The road and building accuracy values measure the ratio of corresponding class pixels in the predicted image and pixels in the ground truth. From Figures 5.8 and 5.9, it reveals that the proposed O-SegNet architecture discriminated around 88 percent of road pixels and 82 percent of building pixels. The FCN variant introduced by Kaiser *et al.* (2017) [2] produced a broad range of road accuracies and narrow range of building accuracies. However, maximum and minimum class-wise accuracies produced by FCN variant [Kaiser *et al.* (2017)]

[2]Modified version of original architecture

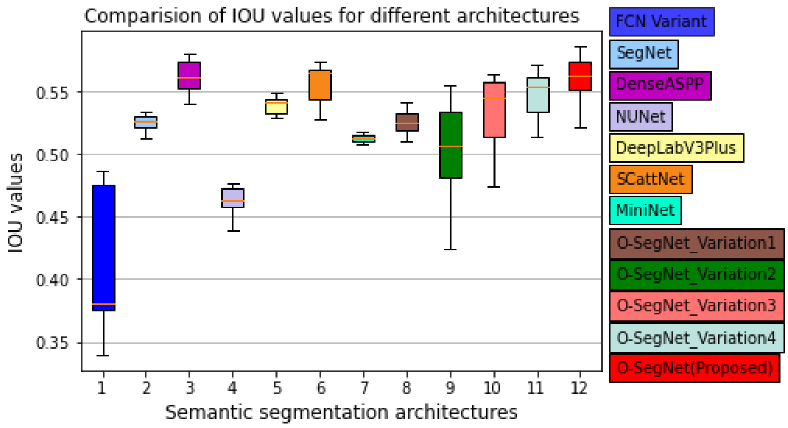


Figure 5.10: Comparison of IOU values

Table 5.2: Average quality metrics comparison of semantic segmentation architectures for Test images

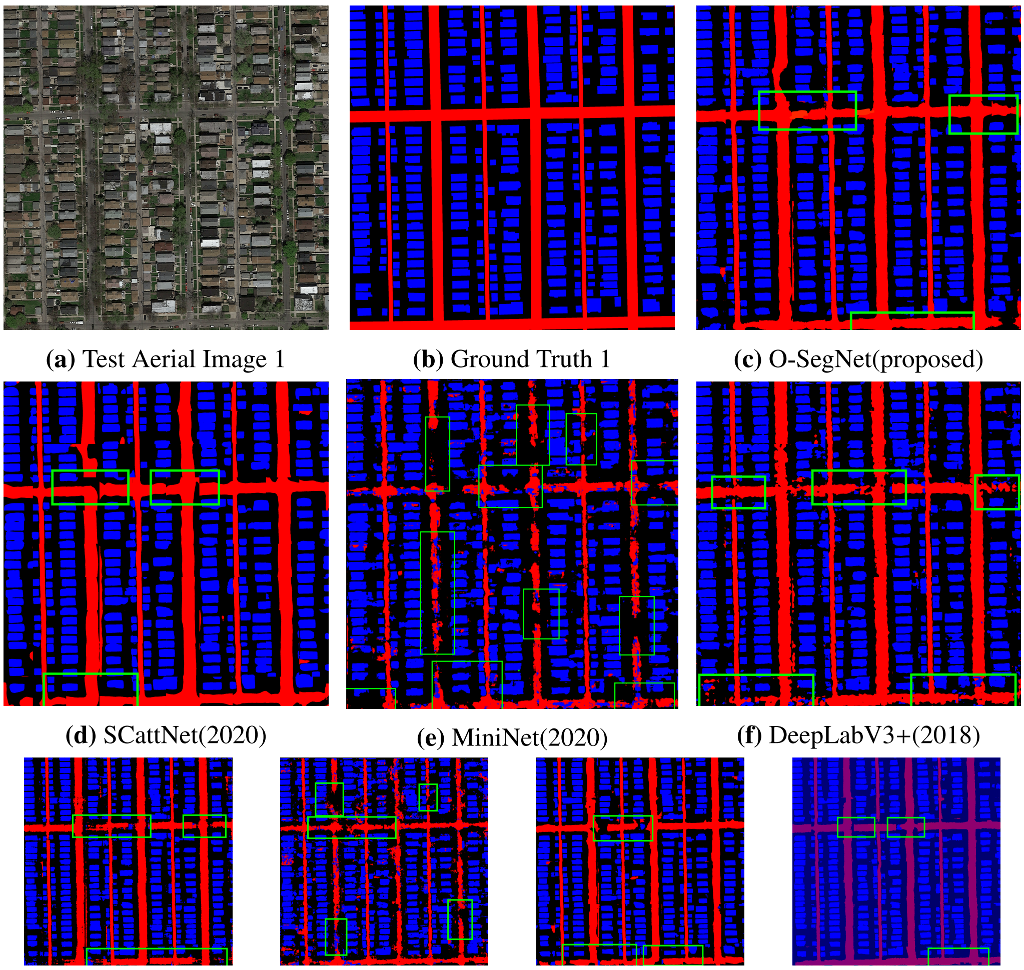
Model	Average Test Accuracy	Road Accuracy	Building Accuracy	Precision	Recall	Mean IOU	Parameters (in million)
FCN Variant [Kaiser *et al.* (2017)]	0.7045	0.4961	0.5186	0.8974	0.6077	0.3527	18.62
SegNet [Badrinarayanan *et al.* (2017)]	0.7092	0.8241	0.7337	0.6891	0.7092	0.5335	34.96
DenseASPP [Yang *et al.* (2018)]	0.7057	0.8296	0.7505	0.7241	0.7357	0.5804	43.39
NUNet [Samy *et al.* (2018)]	0.6367	0.7566	0.7232	0.6044	0.6367	0.4762	0.90
DeepLabV3Plus [Chen *et al.* (2018)]	0.7068	0.8257	0.7383	0.6857	0.7068	0.5489	47.95
SCattNet [Li *et al.* (2020)]	0.7158	0.8559	0.7732	0.7020	0.7158	0.5676	34.96
MiniNet(2020) Alonso *et al.* (2020)	0.6870	0.7767	0.6992	0.6930	0.6870	0.5171	0.52
O-SegNet (Proposed)	**0.7263**	**0.8880**	**0.8108**	**0.7042**	**0.7263**	**0.5861**	**42.40**

are far less than the other models. Therefore, initial and final class-wise accuracies of the proposed architecture are significantly higher compared to other architectures. Moreover, class-wise accuracies of O-SegNet variants are also comparable with the other existing semantic segmentation architectures.

The IOU value quantifies the percentage of predicted image pixels that are overlapped with the ones in the corresponding ground truth. From Figure 5.10, the proposed O-SegNet architecture reaches a maximum IOU value of around 0.59 from an initial overlap as compared to the other architectures. The model FCN variant [Kaiser *et al.* (2017)] possess higher box lengths, but final obtained IOU value is far less than other architectures. From variants of the O-SegNet architecture, variation 2 and 3 possess a higher IOU variability.

The model Dense ASPP [Yang *et al.* (2018)] and the O-SegNet architecture displayed nearly identical maximum IOU values. However, the proposed model show greater improvement in IOU variance from its initial values than the Dense ASPP model. Table 5.2 lists the values of quality metrics and also trainable parameters of the proposed and some of the existing semantic segmentation architectures. From these quality metrics, it can be observed that the proposed network exhibited a significantly higher values for the class-wise accuracies, average test accuracy, and the mean IOU values. The lowest number of trainable parameters is contributed by the model MiniNet. In contrast, the highest number of the trainable parameter is obtained for the model in Chen *et al.* (2018), and the third-highest number of parameters is attained for the proposed O-SegNet architecture.

The test input aerial images, ground truths, and the corresponding segmentation maps produced by proposed, and other existing semantic segmentation architectures are displayed in Figures 5.11, 5.12, and 5.13 The predicted images of models are highlighted with green color boxes to reflect their performance. From Figure 5.11c, one can observe that narrow, broad, and junction road regions are well-segmented by the proposed O-SegNet architecture without any gaps. Further, rectangular-shaped buildings are detected more accurately as compared to other architecture results. The segmentation maps produced by SCattNet, SegNet and MiniNet model contained a gap in between straight and junction roads, which are displayed in Figures 5.11d and 5.11i, and 5.11e respectively. In predicted images of NUNet shown in Figure 5.11h, road and building pixels are mixed, which means that the model failed to differentiate between road and building pixels. The FCN Variant model predictions are blurred, and also the segmented image quality is significantly reduced, and can be seen in Figure 5.11j. Further, holes are observed in DeepLabV3Plus, and DenseASPP model segmentation maps, and also predicted road pixels in the place of building pixels. These model results are shown in Figures 5.11f and 5.11g. The predicted images shown in Figures 5.12c, 5.13c, and 5.14c, reveal that the proposed O-SegNet architecture extracted all kinds of roads and buildings accurately by maintaining separation. Therefore, the proposed semantic segmentation architecture segmented parallel, cross, and diverse shaped roads and buildings without any gaps between them.

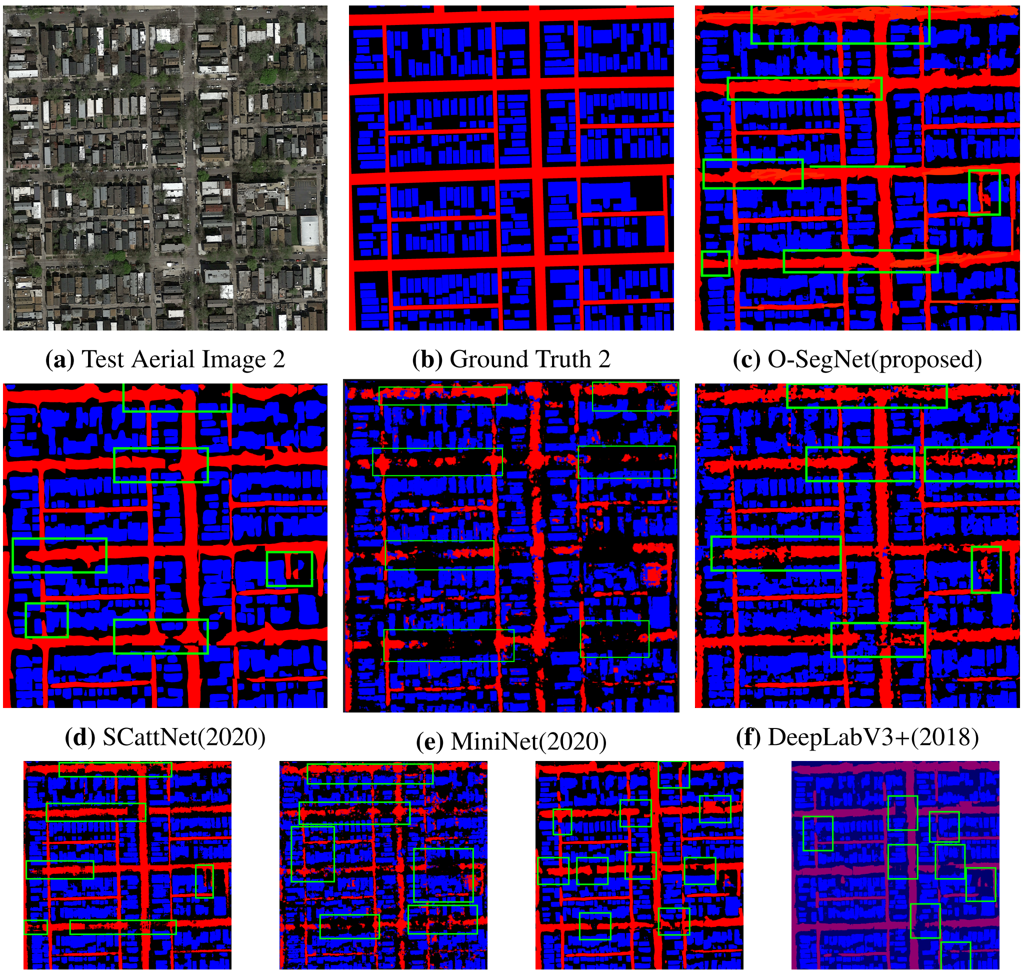


(**a**) Test Aerial Image 1 (**b**) Ground Truth 1 (**c**) O-SegNet(proposed)

(**d**) SCattNet(2020) (**e**) MiniNet(2020) (**f**) DeepLabV3+(2018)

(**g**) DenseASPP(2018) (**h**) NUNet(2018) (**i**) SegNet(2017) (**j**) FCN Variant(2017)

Figure 5.11: Predicted images of proposed O-SegNet and other existing deep learning segmentation architectures for the Test Aerial Image 1.

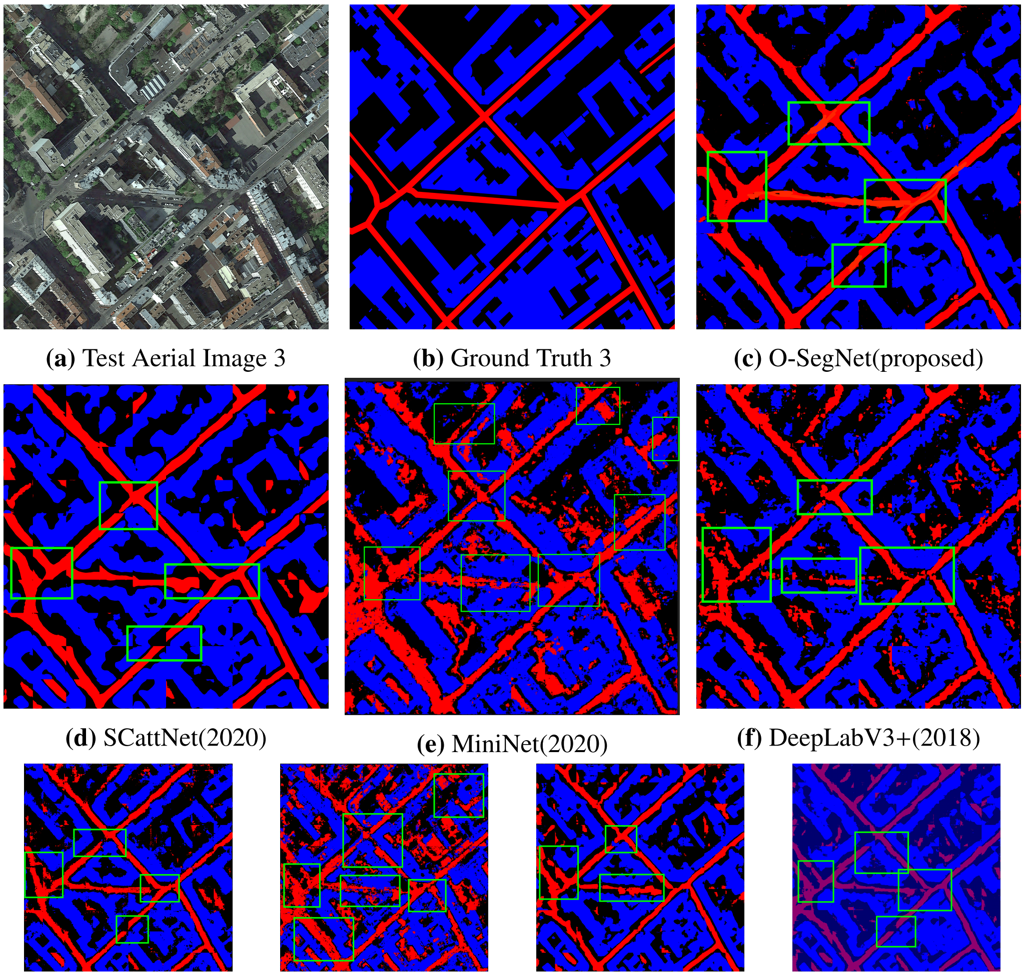


(a) Test Aerial Image 2 (b) Ground Truth 2 (c) O-SegNet(proposed)

(d) SCattNet(2020) (e) MiniNet(2020) (f) DeepLabV3+(2018)

(g) DenseASPP(2018) (h) NUNet(2018) (i) SegNet(2017) (j) FCN Variant(2017)

Figure 5.12: Predicted images of proposed O-SegNet and other existing deep learning segmentation architectures for the Test Aerial Image 2.

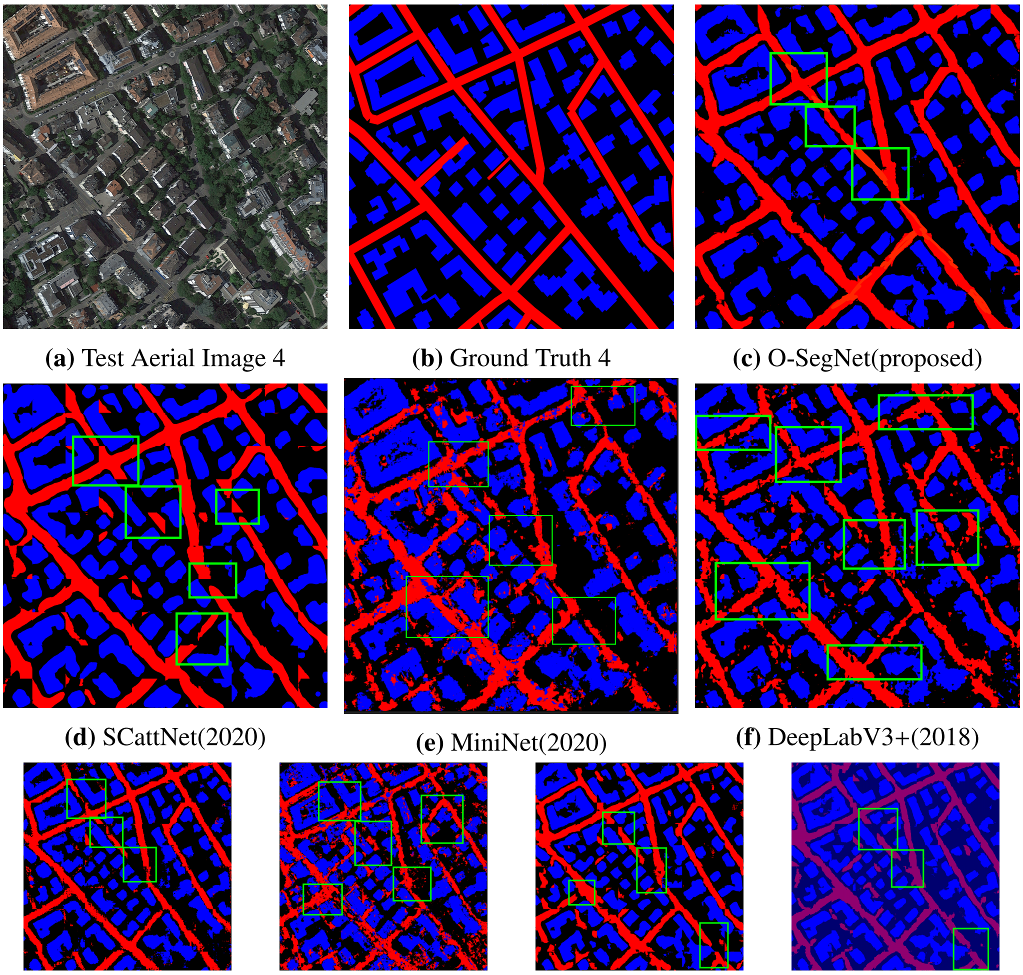


(a) Test Aerial Image 3 **(b)** Ground Truth 3 **(c)** O-SegNet(proposed)

(d) SCattNet(2020) **(e)** MiniNet(2020) **(f)** DeepLabV3+(2018)

(g) DenseASPP(2018) **(h)** NUNet(2018) **(i)** SegNet(2017) **(j)** FCN Variant(2017)

Figure 5.13: Predicted images of proposed O-SegNet and other existing deep learning segmentation architectures for the Test Aerial Image 3.



(a) Test Aerial Image 4 **(b)** Ground Truth 4 **(c)** O-SegNet(proposed)

(d) SCattNet(2020) **(e)** MiniNet(2020) **(f)** DeepLabV3+(2018)

(g) DenseASPP(2018) **(h)** NUNet(2018) **(i)** SegNet(2017) **(j)** FCN Variant(2017)

Figure 5.14: Predicted images of proposed O-SegNet and other existing deep learning segmentation architectures for the Test Aerial Image 4.

5.6.2 Computational Complexity Analysis

The computational complexity of the proposed O-SegNet and existing semantic segmentation architectures is measured by calculating training time (per image, per total dataset), average test time, and the number of FLOPs. These calculated values are listed in Table 5.3. Training time per image indicates the amount of time required for one image to pass through forward and back propagations of the model during the training phase. Similarly, the time required to train the individual model for a specified number of iterations is given by the total training time. Higher values of training time per image lead to more significant values of training and test run times. Average test time provides time required for the trained model to produce segmentation maps of the test images. It can be observed that the

Table 5.3: Comparison of FLOPs, Training and average Test run time of all models

Model	Training time per image(sec)	Total Training time(Hours)	Average Test (run time(sec))	FLOPs(in billion)
FCN Variant[Kaiser *et al.* (2017)]	0.2	46	0.35	42.3
SegNet [Badrinarayanan *et al.* (2017)]	0.54	126	0.49	90.0
Dense ASPP [Yang *et al.* (2018)]	0.07	16	0.11	20.8
NUNet (N=3) [Samy *et al.* (2018)]	0.525	120	1.29	57.8
DeepLabV3Plus [Chen *et al.* (2018)]	0.15	41.5	0.23	32.8
SCattNet [Li *et al.* (2020)]	0.54	126	0.49	90.2
MiniNet(2020) Alonso *et al.* (2020)	0.41	85	0.07	12.89
O-SegNet (Proposed)	**0.53**	**126**	**1.03**	**83.9**

proposed O-SegNet, SCattNet, and SegNet architectures require slightly higher than 120 hours of the training time. The model FCN Variant required less training and test run times due to the utilization of transposed convolutions with high stride factors in the up-sampling path. The training and test times of the models DenseASPP and DeepLabV3Plus are significantly lower since they are based on the pre-trained network. The number of FLOPs for the proposed network, ScattNet, and SegNet is considerably high as compared to the other semantic segmentation architectures. However, the average test run time of O-SegNet is significantly increased as it required more time to load trained weights due to multiple paths. Among all models, the number of FLOPs and test run times of MiniNet is considerably less because it is based on dilated convolutions with different dilation factors. The total training time required to train and validate the proposed model, its variations and compared semantic segmentation architectures is 972 hours. The proposed O-SegNet require moderately high training time as it involves 1×1 convolutions, matrix multiplications, and multiple paths. However, the average test time is nearly the same as compared with the architectures which do not consider pre trained weights.

5.7 Summary

In this work, O-SegNet architecture for semantic segmentation of roads and buildings from aerial imagery data was proposed. The O-SegNet contains Guided Attention blocks to model the inter-relationship between features and also 8 Level PPN module to extract global context. Further, in this architecture, relevant encoder context is considered at the decoder through attention mechanism. The quality metrics of the proposed O-SegNet architecture and existing semantic segmentation architecture revealed that the proposed architecture segmented nearly 89 percent of road pixels and 82 percent of building pixels accurately. Moreover, the average test accuracy, and mean IOU values of the O-SegNet architecture was significantly higher than the other architectures. From the prediction results of models, it was observed that the proposed O-SegNet architecture achieved the detailed segmentation of diversified roads and irregularly shaped buildings as compared to other architectures. Additionally, variants of the proposed network also provided comparable qualitative and quantitative results with the other semantic segmentation architectures.

The distinguishing performance of the proposed network was attributed to the effective utilization of an 8-Level Pyramid Pooling Network, attention mechanism, and also the residual connections. Further, the global context of the objects was extracted and also provided an emphasis on relevant encoder features. Therefore, the connectivity along roads and the interval between consecutive buildings were maintained in predictions produced by the O-SegNet architecture.

The summary of the work presented in this thesis and future scope will be discussed in the Chapter 6.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

The summary of the results achieved using the proposed approaches, conclusion, and future work scope is presented in this Chapter. The various methods for quality enhancement of aerial remotely sensed images were reviewed and introduced a new framework for contrast enhancement. Further, methods based on deep CNN's were effectively exploited and proposed two new architectures for semantic segmentation of HSR remotely sensed aerial imagery data. These architectures help in automating the segmentation process by avoiding manual intervention in all stages. A prior experimental study was conducted to assess the performance of existing semantic segmentation architectures for object extraction from aerial imagery data.

The framework for contrast enhancement of aerial images was proposed by employing Particle Swarm Optimization (PSO) after color balancing, restoration, and saturation adjustment. The experimental results of the proposed and the other quality restoration methods were evaluated on various aerial image datasets by considering different performance metrics. The visually enhanced results and performance metrics of the proposed aerial image restoration framework were better than the other state-of-the-art quality restoration methods. Hence, this quality restoration framework can be adapted to the pre-processing of HSR remote sensing images.

The objects present in HSR remotely sensed aerial images were segmented through semantic segmentation by classifying and localizing every pixel of the image. Two architectures were introduced to provide objects segmentation from high-resolution aerial images. The architecture named DRR Net based on dense convolutions was proposed to segment roads from aerial imagery data. This architecture incorporated the idea of iterative reuse of features for effective utilization through dense convolutions, residual connections, and the stacking mechanism of DRR modules present in the DRR Net. The proposed DRR Net achieved a significant increase in road accuracy compared with the state-of-the-art semantic segmentation architectures with a tenfold reduction in the number of parameters. The different roads present in all aspects, including dominant background environments, were precisely segmented by the proposed DRR Net.

Another semantic segmentation architecture O-SegNet was proposed to segment roads and buildings simultaneously from aerial images. In this architecture, the spatial dependencies present in the features of different resolution was modeled through guided attention. Further, attention between the encoder context and decoder of O-SegNet was introduced to emphasize relevant encoder context during prediction. The different variants were also introduced to evaluate the effectiveness of the proposed O-SegNet architecture. From quality metrics, it revealed that the O-SegNet outperformed significantly than the other compared semantic segmentation architectures. The O-SegNet achieved a detailed segmentation of roads and buildings by preserving the road connectivity and gap between neighboring buildings. Both the proposed architectures were trained with the composite loss function, a combination of the cross-entropy loss function and Lovasz loss function to improve classification accuracy and IOU values simultaneously. The information about the future scope of the research is presented in the Section 6.2.

6.2 Future Work

In future, there is a need to examine the possible combination of the loss functions present in the literature to improve the prediction accuracy of proposed semantic segmentation architectures for objects segmentation from aerial imagery data. Later, the segmentation approaches based on Generative Adversarial Networks (GAN) need to explore. In these types of architectures, the segmentation maps for aerial images are produced in an adversarial way. Another future direction is introducing modules to enhance the semantic ability of the network and their effective utilization in the reconstruction path of the network. Finally, semantic segmentation networks based on different types of convolution viz global convolution, grouped convolutions, and dense convolutions can be devised further.

Bibliography

Ab Wahab, M. N., **S. Nefti-Meziani**, and **A. Atyabi** (2015). A comprehensive review of swarm optimization algorithms. *PloS one*, **10**(5), e0122827.

Abadi, M., **P. Barham**, **J. Chen**, **Z. Chen**, **A. Davis**, **J. Dean**, **M. Devin**, **S. Ghemawat**, **G. Irving**, **M. Isard**, *et al.*, Tensorflow: A system for large-scale machine learning. *In 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. 2016.

Agaian, S. S., **B. Silver**, and **K. A. Panetta** (2007). Transform coefficient histogram-based image enhancement algorithms using contrast entropy. *IEEE transactions on image processing*, **16**(3), 741–758.

Aich, S., **W. van der Kamp**, and **I. Stavness**, Semantic binary segmentation using convolutional networks without decoders. *In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2018.

Alonso, I., **L. Riazuelo**, and **A. C. Murillo** (2020). Mininet: An efficient semantic segmentation convnet for real-time robotic applications. *IEEE Transactions on Robotics*.

Alpaydin, E., *Introduction to machine learning*. MIT press, 2020.

Amo, M., **F. Martínez**, and **M. Torre** (2006). Road extraction from aerial images using a region competition algorithm. *IEEE transactions on image processing*, **15**(5), 1192–1201.

Badrinarayanan, V., **A. Kendall**, and **R. Cipolla** (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*.

Badrinarayanan, V., **A. Kendall**, and **R. Cipolla** (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, **39**(12), 2481–2495.

Bahdanau, D., **K. Cho**, and **Y. Bengio** (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Berman, M., **A. Rannen Triki**, and **M. B. Blaschko**, The lovász-softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

Braik, M., **A. Sheta**, and **A. Ayesh** (2007*a*). Particle swarm optimisation enhancement approach for improving image quality. *International Journal of Innovative Computing and Applications*, **1**(2), 138.

Braik, M., **A. F. Sheta**, and **A. Ayesh**, Image enhancement using particle swarm optimization. *In World congress on engineering*, volume 1. 2007*b*.

Chen, L.-C., **G. Papandreou**, **I. Kokkinos**, **K. Murphy**, and **A. L. Yuille** (2017*a*). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, **40**(4), 834–848.

Chen, L.-C., **G. Papandreou**, **F. Schroff**, and **H. Adam** (2017*b*). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*.

Chen, L.-C., **Y. Yang**, **J. Wang**, **W. Xu**, and **A. L. Yuille**, Attention to scale: Scale-aware semantic image segmentation. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

Chen, L.-C., **Y. Zhu**, **G. Papandreou**, **F. Schroff**, and **H. Adam** (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611*.

Chen, Z., **B. R. Abidi**, **D. L. Page**, and **M. A. Abidi** (2006). Gray-level grouping (glg): an automatic method for optimized image contrast enhancement-part i: the basic method. *IEEE transactions on image processing*, **15**(8), 2290–2302.

Das, S., **T. Mirnalinee**, and **K. Varghese** (2011). 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*, **49**(10), 3906–3931.

Delice, Y., **E. K. Aydoğan**, **U. Özcan**, and **M. S. İlkay** (2017). A modified particle swarm optimization algorithm to mixed-model two-sided assembly line balancing. *Journal of Intelligent Manufacturing*, **28**(1), 23–36.

Demirel, H., **C. Ozcinar**, and **G. Anbarjafari** (2009). Satellite image contrast enhancement using discrete wavelet transform and singular value decomposition. *IEEE Geoscience and remote sensing letters*, **7**(2), 333–337.

dos Santos Coelho, L., **J. G. Sauer**, and **M. Rudek** (). Differential evolution optimization combined with chaotic sequences for image contrast enhancement.

Duchi, J., **E. Hazan**, and **Y. Singer** (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, **12**(7).

Eerapu, K. K., **B. Ashwath**, **S. Lal**, **F. Dell'Acqua**, and **A. N. Dhan** (2019). Dense refinement residual network for road extraction from aerial imagery data. *IEEE Access*, **7**, 151764–151782.

Faria, P., **J. Soares**, **Z. Vale**, **H. Morais**, and **T. Sousa** (2013). Modified particle swarm optimization applied to integrated demand response and dg resources scheduling. *IEEE Transactions on smart grid*, **4**(1), 606–616.

Filin, O., **E. D. Analytics**, **A. Zapara**, and **S. Panchenko**, Road detection with eosresunet and post vectorizing algorithm. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2018.

Fu, J., **J. Liu**, **H. Tian**, **Y. Li**, **Y. Bao**, **Z. Fang**, and **H. Lu**, Dual attention network for scene segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019.

Fu, X., **J. Wang**, **D. Zeng**, **Y. Huang**, and **X. Ding** (2015). Remote sensing image enhancement using regularized-histogram equalization and dct. *IEEE Geoscience and Remote Sensing Letters*, **12**(11), 2301–2305.

Gehring, J., **M. Auli**, **D. Grangier**, **D. Yarats**, and **Y. N. Dauphin**, Convolutional sequence to sequence learning. *In Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017.

Glorot, X. and **Y. Bengio**, Understanding the difficulty of training deep feedforward neural networks. *In Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 2010.

Gogna, A. and **A. Tayal** (2013). Metaheuristics: review and application. *Journal of Experimental & Theoretical Artificial Intelligence*, **25**(4), 503–526.

Gu, K., **W. Lin**, **G. Zhai**, **X. Yang**, **W. Zhang**, and **C. W. Chen** (2016). No-reference quality metric of contrast-distorted images based on information maximization. *IEEE transactions on cybernetics*, **47**(12), 4559–4565.

Gu, K., **D. Tao**, **J.-F. Qiao**, and **W. Lin** (2017). Learning a no-reference quality assessment model of enhanced images with big data. *IEEE transactions on neural networks and learning systems*, **29**(4), 1301–1313.

Han, H.-Y., **Y.-C. Chen**, **P.-Y. Hsiao**, and **L.-C. Fu** (2020). Using channel-wise attention for deep cnn based real-time semantic segmentation with class-aware edge information. *IEEE Transactions on Intelligent Transportation Systems*.

Harris, C. G., **M. Stephens**, *et al.*, A combined corner and edge detector. *In Alvey vision conference*, volume 15. Citeseer, 1988.

He, K., **X. Zhang**, **S. Ren**, and **J. Sun**, Spatial pyramid pooling in deep convolutional networks for visual recognition. *In European conference on computer vision*. Springer, 2014.

He, K., **X. Zhang**, **S. Ren**, and **J. Sun**, Deep residual learning for image recognition. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016*a*.

He, K., **X. Zhang**, **S. Ren**, and **J. Sun**, Identity mappings in deep residual networks. *In European conference on computer vision*. Springer, 2016*b*.

Hoseini, P. and **M. G. Shayesteh** (2013). Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing. *Digital Signal Processing*, **23**(3), 879–893.

Hu, J., **A. Razdan**, **J. C. Femiani**, **M. Cui**, and **P. Wonka** (2007). Road network extraction and intersection detection from aerial images by tracking road footprints. *IEEE Transactions on Geoscience and Remote Sensing*, **45**(12), 4144–4157.

Hu, J., **L. Shen**, and **G. Sun**, Squeeze-and-excitation networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

Huang, G., **Z. Liu**, **L. Van Der Maaten**, and **K. Q. Weinberger**, Densely connected convolutional networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

Huang, L., **Y. Yuan**, **J. Guo**, **C. Zhang**, **X. Chen**, and **J. Wang** (2019*a*). Interlaced sparse self-attention for semantic segmentation. *arXiv preprint arXiv:1907.12273*.

Huang, Z., **X. Wang**, **L. Huang**, **C. Huang**, **Y. Wei**, and **W. Liu**, Ccnet: Criss-cross attention for semantic segmentation. *In Proceedings of the IEEE International Conference on Computer Vision*. 2019*b*.

Inglada, J. (2007). Automatic recognition of man-made objects in high resolution optical remote sensing images by svm classification of geometric image features. *ISPRS journal of photogrammetry and remote sensing*, **62**(3), 236–248.

Ioffe, S. and **C. Szegedy** (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Jégou, S., **M. Drozdzal**, **D. Vazquez**, **A. Romero**, and **Y. Bengio**, The One Hundred Layers Tiramisu: Fully Convolutional Densenets for Semantic Segmentation. *In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2017.

Jmal, M., **W. Souidene**, and **R. Attia** (2017). Efficient cultural heritage image restoration with nonuniform illumination enhancement. *Journal of Electronic Imaging*, **26**(1), 011020.

Kaiser, P., **J. D. Wegner**, **A. Lucchi**, **M. Jaggi**, **T. Hofmann**, and **K. Schindler** (2017). Learning aerial image segmentation from online maps. *IEEE Transactions on Geoscience and Remote Sensing*, **55**(11), 6054–6068.

Kim, J. H., **H. Lee**, **S. J. Hong**, **S. Kim**, **J. Park**, **J. Y. H**, and **J. P. Choi** (2018). Objects segmentation from high-resolution aerial images using u-net with pyramid pooling layers. *IEEE Geoscience and Remote Sensing Letters*, **16**(1), 115–119.

Kim, J. H., **H. Lee**, **S. J. Hong**, **S. Kim**, **J. Park**, **J. Y. Hwang**, and **J. P. Choi** (2019). Objects segmentation from high-resolution aerial images using u-net with pyramid pooling layers. *IEEE Geoscience and Remote Sensing Letters*, **16**(1), 115–119.

Kingma, D. P. and **J. Ba** (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kluckner, S., **T. Mauthner**, **P. M. Roth**, and **H. Bischof**, Semantic classification in aerial imagery by integrating appearance and height information. *In Asian Conference on Computer Vision*. Springer, 2009.

Kwok, N. and **H. Shi**, Design of unsharp masking filter kernel and gain using particle swarm optimization. *In 2014 7th International Congress on Image and Signal Processing*. IEEE, 2014.

Kwok, N. and **H. Shi**, An integrated framework for aerial image restoration. *In 2015 International Conference on Machine Learning and Cybernetics (ICMLC)*, volume 1. IEEE, 2015.

Kwok, N. M., **Q. P. Ha**, **D. Liu**, and **G. Fang** (2008). Contrast enhancement and intensity preservation for gray-level images using multiobjective particle swarm optimization. *IEEE Transactions on Automation Science and Engineering*, **6**(1), 145–155.

Kwok, N. M., **H. Shi**, **G. Fang**, and **Q. P. Ha**, Intensity-based gain adaptive unsharp masking for image contrast enhancement. *In 2012 5th International Congress on Image and Signal Processing*. IEEE, 2012.

LeCun, Y., **Y. Bengio**, *et al.* (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, **3361**(10), 1995.

Li, H., **K. Qiu**, **L. Chen**, **X. Mei**, **L. Hong**, and **C. Tao** (2020). Scattnet: Semantic segmentation network with spatial and channel attention mechanism for high-resolution remote sensing images. *IEEE Geoscience and Remote Sensing Letters*.

Li, Z., **W. Shi**, **Q. Wang**, and **Z. Miao** (2014). Extracting man-made objects from high spatial resolution remote sensing images via fast level set evolutions. *IEEE Transactions on Geoscience and Remote Sensing*, **53**(2), 883–899.

Long, J., **E. Shelhamer**, and **T. Darrell**, Fully convolutional networks for semantic segmentation. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Mahapatra, P. K., **S. Ganguli**, and **A. Kumar** (2015). A hybrid particle swarm optimization and artificial immune system algorithm for image enhancement. *Soft computing*, **19**(8), 2101–2109.

Mai, C., **M. Nguyen**, and **N. Kwok**, A modified unsharp masking method using particle swarm optimization. *In 2011 4th International Congress on Image and Signal Processing*, volume 2. IEEE, 2011.

Michelson, A. A., *Studies in optics*. Courier Corporation, 1995.

Mnih, V. and **G. E. Hinton**, Learning to detect roads in high-resolution aerial images. *In European Conference on Computer Vision*. Springer, 2010.

Mohamed, S., **R. J. Priya**, **S. Rojan**, and **S. Y. Arafath**, Particle swarm based unsharp masking. *In Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing*. ACM, 2010.

Montoya-Zegarra, J. A., **J. D. Wegner**, **L. Ladickỳ**, and **K. Schindler**, Mind the gap: modeling local and global context in (road) networks. *In German Conference on Pattern Recognition*. Springer, 2014.

Odena, A., **V. Dumoulin**, and **C. Olah** (2016). Deconvolution and checkerboard artifacts. *Distill*, **1**(10), e3.

Oktay, O., **J. Schlemper**, **L. L. Folgoc**, **M. Lee**, **M. Heinrich**, **K. Misawa**, **K. Mori**, **S. McDonagh**, **N. Y. Hammerla**, **B. Kainz**, *et al.* (2018). Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*.

Ortner, M., **X. Descombes**, and **J. Zerubia** (2007). Building outline extraction from digital elevation models using marked point processes. *International Journal of Computer Vision*, **72**(2), 107–132.

Pan, X., **F. Yang**, **L. Gao**, **Z. Chen**, **B. Zhang**, **H. Fan**, and **J. Ren** (2019). Building extraction from high-resolution aerial imagery using a generative adversarial network with spatial and channel attention mechanisms. *Remote Sensing*, **11**(8), 917.

Peng, C., **X. Zhang**, **G. Yu**, **G. Luo**, and **J. Sun**, Large kernel matters–improve semantic segmentation by global convolutional network. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.

Pohlen, T., **A. Hermans**, **M. Mathias**, and **B. Leibe** (2017). Fullresolution residual networks for semantic segmentation in street scenes. *arXiv preprint*.

Ronneberger, O., **P. Fischer**, and **T. Brox**, U-net: Convolutional networks for biomedical image segmentation. *In International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015.

Rosten, E., **R. Porter**, and **T. Drummond** (2008). Faster and better: A machine learning approach to corner detection. *IEEE transactions on pattern analysis and machine intelligence*, **32**(1), 105–119.

Saito, S., **T. Yamashita**, and **Y. Aoki** (2016). Multiple object extraction from aerial imagery with convolutional neural networks. *Electronic Imaging*, **2016**(10), 1–9.

Samy, M., **K. Amer**, **K. Eissa**, **M. Shaker**, and **M. ElHelw**, Nu-net: Deep residual wide field of view convolutional neural network for semantic segmentation. *In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2018.

Schlemper, J., **O. Oktay**, **M. Schaap**, **M. Heinrich**, **B. Kainz**, **B. Glocker**, and **D. Rueckert** (2019). Attention gated networks: Learning to leverage salient regions in medical images. *Medical image analysis*, **53**, 197–207.

Ševo, I. and **A. Avramović** (2016). Convolutional neural network based automatic object detection on aerial images. *IEEE geoscience and remote sensing letters*, **13**(5), 740–744.

Shanmugavadivu, P. and **K. Balasubramanian** (2014). Particle swarm optimized multi-objective histogram equalization for image enhancement. *Optics & Laser Technology*, **57**, 243–251.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell system technical journal*, **27**(3), 379–423.

Shin, J. and **R.-H. Park** (2015). Histogram-based locality-preserving contrast enhancement. *IEEE Signal Processing Letters*, **22**(9), 1293–1296.

Silverman, B. W., *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

Simonyan, K. and **A. Zisserman** (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sirmacek, B. and **C. Unsalan** (2010). A probabilistic framework to detect buildings in aerial and satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, **49**(1), 211–221.

Stoica, R., **X. Descombes**, and **J. Zerubia** (2004). A gibbs point process for road extraction from remotely sensed images. *International Journal of Computer Vision*, **57**(2), 121–136.

Sun, T., **Z. Chen**, **W. Yang**, and **Y. Wang**, Stacked u-nets with multi-output for road extraction. *In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2018.

Sun, W. and **R. Wang** (2018). Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with dsm. *IEEE Geoscience and Remote Sensing Letters*, **15**(3), 474–478.

Suresh, S., **D. Das**, **S. Lal**, and **D. Gupta** (2018). Image quality restoration framework for contrast enhancement of satellite remote sensing images. *Remote Sensing Applications: Society and Environment*, **10**, 104–119.

Suresh, S. and **S. Lal** (2017). Modified differential evolution algorithm for contrast and brightness enhancement of satellite images. *Applied Soft Computing*, **61**, 622–641.

Szegedy, C., **W. Liu**, **Y. Jia**, **P. Sermanet**, **S. Reed**, **D. Anguelov**, **D. Erhan**, **V. Vanhoucke**, and **A. Rabinovich**, Going deeper with convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

Tieleman, T. and **G. Hinton** (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, **4**(2), 26–31.

Unsalan, C. (2006). Gradient-magnitude-based support regions in structural land use classification. *IEEE Geoscience and Remote Sensing Letters*, **3**(4), 546–550.

Unsalan, C. and **B. Sirmacek** (2012). Road network detection using probabilistic and graph theoretical methods. *IEEE Transactions on Geoscience and Remote Sensing*, **50**(11), 4441–4453.

Vaswani, A., **N. Shazeer**, **N. Parmar**, **J. Uszkoreit**, **L. Jones**, **A. N. Gomez**, **Ł. Kaiser**, and **I. Polosukhin**, Attention is all you need. *In Advances in neural information processing systems*. 2017.

Vetterli, M. and **J. Kovacevic**, *Wavelets and subband coding*. BOOK. Prentice-hall, 1995.

Wan, M., **G. Gu**, **W. Qian**, **K. Ren**, **Q. Chen**, and **X. Maldague** (2018). Particle swarm optimization-based local entropy weighted histogram equalization for infrared image enhancement. *Infrared Physics & Technology*, **91**, 164–181.

Wang, F., **M. Jiang**, **C. Qian**, **S. Yang**, **C. Li**, **H. Zhang**, **X. Wang**, and **X. Tang**, Residual attention network for image classification. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

Wegner, J. D., **J. A. Montoya-Zegarra**, and **K. Schindler**, A higher-order crf model for road network extraction. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013.

Wei, Y., **Z. Wang**, and **M. Xu** (2017). Road structure refined cnn for road extraction in aerial image. *IEEE Geoscience and Remote Sensing Letters*, **14**(5), 709–713.

Wu, Y., **M. Schuster**, **Z. Chen**, **Q. V. Le**, **M. Norouzi**, **W. Macherey**, **M. Krikun**, **Y. Cao**, **Q. Gao**, **K. Macherey**, *et al.* (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Yang, M., **K. Yu**, **C. Zhang**, **Z. Li**, and **K. Yang**, Denseaspp for semantic segmentation in street scenes. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

Yang, X.-S., *Nature-inspired optimization algorithms*. Elsevier, 2014.

Yu, C., **J. Wang**, **C. Peng**, **C. Gao**, **G. Yu**, and **N. Sang**, Bisenet: Bilateral segmentation network for real-time semantic segmentation. *In Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

Yu, F. and **V. Koltun** (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Zhan, K., **J. Shi**, **J. Teng**, **Q. Li**, **M. Wang**, and **F. Lu** (2017). Linking synaptic computation for image enhancement. *Neurocomputing*, **238**, 1–12.

Zhang, H., **I. Goodfellow**, **D. Metaxas**, and **A. Odena** (2018*a*). Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*.

Zhang, J., **Z. Jiang**, **J. Dong**, **Y. Hou**, and **B. Liu** (2020). Attention gate resu-net for automatic mri brain tumor segmentation. *IEEE Access*, **8**, 58533–58545.

Zhang, Z., **H. Fu**, **H. Dai**, **J. Shen**, **Y. Pang**, and **L. Shao**, Et-net: A generic edge-attention guidance network for medical image segmentation. *In International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019.

Zhang, Z., **Q. Liu**, and **Y. Wang** (2018*b*). Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*.

Zhao, H., **Y. Zhang**, **S. Liu**, **J. Shi**, **C. Change Loy**, **D. Lin**, and **J. Jia**, Psanet: Point-wise spatial attention network for scene parsing. *In Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

Zhu, Z., **M. Xu**, **S. Bai**, **T. Huang**, and **X. Bai**, Asymmetric non-local neural networks for semantic segmentation. *In Proceedings of the IEEE International Conference on Computer Vision*. 2019.

Publications based on the Thesis

- Karuna Kumari E., Devikalyan Das, Shilpa Suresh, Shyam Lal, et al., “A Robust Framework for Quality Enhancement of Aerial Remote Sensing Images”, Infrared Physics and Technology, vol.93, 2018, pp. 362-364, 2018, Elsevier Publisher. Indexed by SCI, Thomson ISI, Scopus (Elsevier), JCR (2017) Impact Factor: 1.851.
- Karuna Kumari Eerapu, Balraj Ashwath, Shyam Lal, et al.,“Dense Refinement Residual Network for Road Extraction from Aerial Imagery Data, 2019, IEEE Access Journal, IEEE publisher. Indexed by SCIE, Thomson ISI, Scopus (Elsevier), JCR (2017) Impact Factor:4.05.
- Karuna Kumari Eerapu, Shyam Lal, and A V Narasimhadhan. "O-SegNet : Robust Encoder and Decoder Architecture for Objects Segmentation from AerialImagery Data" IEEE Transactions on Emerging Topics in Computational Intelligence, IEEE Publisher. (Online Published) Indexed by Scopus (Elsevier).

Bio-data (FOR PHD)

Name	:	KARUNA KUMARI EERAPU
Address	:	Dept.of E&C, NITK, Surathkal, Mangalore, Karnataka - 575025, India. Ph: +917899805584.
Email	:	karuna.eerapu5023@gmail.com

Educational Qualification	:	**M.Tech** in Electronics and Communication Engineering, Jawaharlal Nehru Technological University(JNTU), Kakinada, Andhra Pradesh **B.Tech** in Electronics&Communication Engineering, Jawaharlal Nehru Technological University(JNTU), Kakinada, Andhra Pradesh
Teaching Experience	:	3 years