Evaluation of Deep Learning - based Road Segmentation Methods for Satellite Images

***Abstract*—We demonstrate several approaches for dealing with road segmentation problems. Some rely on the encoder-decoder method, some on the Generative Adversarial Network (GAN) technique, and others on a fully convolutional network. The approaches utilizing Encoder-Decoder and GAN seem to have promise. Due to the great performance of Encoder-Decoder Deep Convolutional Neural Networks in many segmentation problems. Our aim is to apply all recent model architectures that use the DCEP network as a primary base model on two open-source data sets DeepGlope and Massachusetts. We choose the most common encoder-decoder models that proved great performance for different data sets of image segmentation. We choose Unet, FPN, PSPNet, Unet++, PAN, LinkNet, DeepLab- v3, DeepLab-v3+, and MA-Net for our experiments and we give a brief comparison based on the result. We show the results for each model we use, both with and without the bilateral filter, and we show how the IOU (Intersection Over Union) and Dice Loss of the results for all models on the Massachusetts data set are very similar. In an effort to improve model performance, we also use different data augmentation parameters, however, the results are the same for this data set. The Unet model has an excellent IOU for the Deepglobe data set, scoring 95.46% accuracy.**

***Index Terms*—Road segmentation, Encoder-Decoder, Deep learning, Remote sensing, EfficientNet**

I. INTRODUCTION

A. Motivation

The increase in Deep learning approaches that achieve state-of-the-art performance in many applications related to computer vision, the importance of segmentation problems in many real-world applications around the world, and the increasing demand for better performance and accuracy on Road Segmentation problems, all motivated us to start our work and present a robust approach for accurately segment the road from Satellite images.

The extraction of map components including roads, rivers, and buildings from high- resolution satellite data is a major engineering problem in many civilian and military applications. Remote sensing is heavily used in cartography. Autonomous road extraction from satellite imagery is crucial for maintaining maps current as transportation networks grow and change. Synthetic Aperture Radar (SAR) satellites may provide in-depth topographical mapping. Highways are difficult to separate in these statistics since they visually resemble objectives like rivers and railways. Most methods for extracting roads from SAR images still rely on many studies of deep learning's potential despite its successful applications in optical imaging.

Because of the overall urban growth during the preceding 20 years, there has been tremendous transportation network expansion. Because the infrastructure is always evolving, it is necessary to frequently update road maps. Numerous applications use this data, including monitoring urban growth, aiding in rescue operations after disasters, and automated geolocalization system data updates. A Synthetic Aperture Radar (SAR) equipped satellite can be used to determine the topography of a region. The information generated is less sensitive to changes in lighting and color than optical imaging. Since SAR sensors can operate in any weather, they are the best sensor for surveying areas damaged by weather-related disasters.

Extraction of land objects like buildings and roads from remotely sensed photos has several applications, including map updating, urban planning, navigation and route optimization. The majority of primary research for road extraction uses global optimization algorithms and unsupervised learning techniques like graph cutting (43). However, as all of these unsupervised methods rely solely on color features, they all have the same flaw of being color-sensitive. The road colors displayed in the remotely sensed images contain more than one color (for example, yellowish brown roads in the rural parts and cement-grayed roads. In fact, overcoming color sensitivity concerns has been one of the inspirations for our work.

B. Aim And Objectives

Due to the increase of applications that rely on road segmentation requiring high accuracy. We aim to implement different deep learning techniques for two different road extraction data sets, DeepGlope (19) and Massachusetts (40), and propose a robust approach that can achieve a reasonable performance of road segmentation problems by studying recent research papers in the field of image segmentation and studying how deep learning can increase the performance of road extraction problems. Examples of both DeepGlope and Massachusetts data sets are shown in fig. **??**, and **??** respectively.

The objectives and steps we tackle in our work are to provide state-of-the-art architectures that could be considered in the next development and research.

- To Choose one of the recent Deep learning approaches is a crucial step, We demonstrate several distinct methods for resolving issues with road segmentation. some rely on a generative adversarial network (GAN) method, some rely on the encoder-decoder approach others depend on a fully convolutional network (FCN).
- To build a robust approach by figuring out different data augmentation strategies, we choose different angle ranges for rotation, choose more than one data set, and apply 9 models that rely on different architectures to achieve the

best approach that could be considered to increase the accuracy of the road segmentation problem.
- To study multiple strategies based on various loss functions, such as Binary cross entropy (BCE), Loss of dice, and the Cross-Entropy Dice Loss function (CEDL), which combine together in one loss function with weight for each one them.

Due to the great performance of Encoder-Decoder Deep Convolutional Neural networks in many segmentation problems. Our aim is to apply all recent model architectures that use the DCEP network as a primary base model on two open-source data sets DeepGlope (19), and Massachusetts (40). We choose the most common encoder-decoder models that proved great performance for different data sets of image segmentation. We choose Unet(34), FPN(24), PSPNet(67), UnetPlus(68), PAN(57), LinkNet(9), Deeplabv3(15), Deeplabv3+(5), and MA-Net(35).

C. Research Questions

How Deep Learning (DL) can improve the performance of image segmentation? What is the best loss function for image segmentation? What are the best deep-learning architectures for this problem? All these questions lead us to compare different results obtained from different research papers on the Synthetic Aperture Radar (SAR) image segmentation problem. Some use Fully Convolutional Networks (FCN), others use Encoder-Decoder architectures, and others use GAN (Generative Adversarial Neural Networks).

Also, the loss function was an important choice some papers use the Cross-Entropy Loss function and others use Dice Loss, also there is some research combining the two-loss function together giving weight to each one of them. Our research questions listed in the following can comprehend our interest area of research and give intuition about what we tackle in the following sections.

- RQ1: What is the limitation of traditional segmentation techniques?
- RQ2: What is the effect of Data Augmentation on the regularization of the model?
- RQ3: How deep learning can improve the performance of image segmentation?
- RQ4: What are the best deep learning architectures for the SAR road segmentation problem?
- RQ5: What is the best loss function suitable to tackle the SAR road segmentation problem?

D. Contributions

We provide a proof of concept that Encoder-Decoder models achieve great results on Road Segmentation and that is by applying 9 different architectures that depend on the Encoder-Decoder baseline.

Encoder-Decoder segmentation techniques achieve robust performance in the two data sets we choose. The bullet points show the contributions of this thesis.

- First contribution, Encoder-Decoder Deep Convolutional Neural Networks have been proven to have great performance at many segmentation issues, which is why. Our contribution is to apply all current model architectures that use the DCEP network as their primary base model to two open-source data sets from DeepGlope and Massachusetts. We choose 9 different models Unet(34), FPN(24), PSPNet(67), Unet++(68), PAN(57), LinkNet(9), DeepLabv3(15), DeepLabv3+(5), and MA-Net(35).
- Second contribution, to ensure our approach is robust we use two different data sets DeepGlope and Massachusetts, and apply the same models to each one of them.
- Third contribution, we apply different data augmentation hyperparameters for angle rotation degrees we try [0:0], [-30:30], [-60,60], and [-90, 90], To make sure we pick up the most appropriate hyperparameters for all other models.
- Forth contribution, we evaluate all the models using Intersection Over Union (IOU) metric, and dice loss (DL) and give a brief comparison between each one of them and what is the most suitable model for each data set.

After the experiments, we found that the 9 models obtain a great performance metric for the 2 data sets, but the result was so close to each other. The IOU (Intersection Over Union) was around 91% for the Massachusetts data set, and around 0.95% for the Deepglobe data set. The Encoder-Decoder network seems to be a great choice for these data sets.

II. LITERATURE REVIEW

A. Introduction

Deep learning has advanced significantly in recent years. Deep neural network-based techniques have shown novel results in a range of computer vision applications, including scene identification and object detection. Deep neural networks, according to experts in the field (41; 42; 66), have the potential to significantly improve the interpretation and comprehension of remote sensing data. We intend to use a variety of recent deep-learning architectures for image segmentation to construct a robust strategy that is appropriate for our data sets.

In recent years, a number of techniques for extracting roads from satellite images have been suggested. The majority of these techniques fall into one of two categories: road centerline extraction or road area extraction. Road area extraction (28; 41) can produce pixel-level labeling of roads, whereas road centerline extraction (36; 53) try to find a road's bones. Additionally, there are techniques that simultaneously extract the centerline and the road areas (17). Depending on algorithms such as morphological thinning, road centerlines may be easily extracted from road areas. This study focuses on extracting road areas from high-resolution satellite images.

It is possible to think about road area extraction as a segmentation or pixel-level classification issue. For example, Song and Civco (51) suggested a technique for identifying

road regions that makes use of the form index feature and support vector machine (SVM). In order to extract roads from high-resolution multi-spectral images using probabilistic SVM, Das et al. (18) took advantage of two key characteristics of roads. Using hierarchical graph-based image segmentation. Also, Alshehhi and Marpu (4) suggested an unsupervised road extraction method that depends on image processing techniques.

Deep learning has advanced significantly in recent years. Deep neural network-based techniques have attained cutting-edge results in a range of computer vision applications, including scene identification and object detection. Deep neural networks have the potential to significantly improve the interpretation and comprehension of remote sensing data, according to researchers in the field (41; 42; 66). These strategies produce superior outcomes to conventional ones, demonstrating the huge potential of using deep learning algorithms to assess remote sensing jobs.

Mnih and Hinton conducted one of the earliest attempts to employ deep learning techniques in the field of road extraction (41). They suggested a technique using high-quality aerial photos and restricted Boltzmann machines (RBMs) to identify road regions. A preprocessing phase before the detection and a post-processing step after the detection were used to get better results. The pre-processing was used to make the input data's dimensions smaller. Post-processing was used to patch up the roads' holes and remove disconnected spots. method, in contrast to Mnih and Hinton's (41) approach, used convolutional neural networks (CNNs) to directly extract buildings and roads from unprocessed remote sensing data. On the Massachusetts highways data set, this strategy outperforms Mnih and Hinton's method (41).

B. Why Image Segmentation Is Required?

Segmentation is a crucial step in the image recognition process because it isolates the items that are of interest to us so that they can be processed further for recognition or description. For the classification of image pixels, segmentation of a picture is used (8). In order to analyze the item, segmentation techniques are utilized to separate the target object from the image. With the use of image segmentation techniques, for instance, a tumor, cancer, or obstruction in the blood flow can be easily identified from its background (22). There are numerous methods available for segmenting monochrome photos. Since each pixel in color photographs has a vector value, segmenting them is more difficult (3).

Many applications strongly rely on segmentation, including video surveillance, augmented reality, driverless cars, medical image analysis (such as tumor border extraction and measurement of tissue volumes), and autonomous automobiles (such as passable surface and pedestrian recognition)(21).

The expansion of transportation systems, including the introduction of autonomous road navigation and unmanned vehicles, as well as urban planning (65), which is critical for both ordinary life and business, are significantly reliant on the road network. As a result, the development of a unique approach for extracting road networks from high-resolution remote sensing pictures may improve geographic information systems (GIS) and intelligent transportation systems (ITS) (49; 69).

High-resolution photography has evolved into an important data source for real-time updating of the road network in the spatial database, and extracting road networks has become one of the key study areas in the field of image processing.(60).

C. Overview Of Image Segmentation

Computer vision and digital image processing have a section known as image segmentation that focuses on categorizing related areas or segments of an image.

Since the entire process is digital and it is possible to obtain a representation of the analog image in the form of pixels, the work required to create segments is equivalent to the operation of grouping pixels.

Using localization in addition to classification, picture segmentation is an extension of image classification. Because of this, image segmentation is a subset of image classification in which the model draws attention to an object's boundaries to show where it is present.

Most image segmentation models in computer vision employ an encoder-decoder network, as opposed to classifiers, which commonly use a single encoder network. The decoder creates segment maps or maps that reveal the positions of each item in the image after the encoder turns the input into a latent space representation.

1) Traditional Image Segmentation techniques: Digital image processing and optimization methods were the initial sources of image segmentation. These early algorithms used techniques like region expanding and the snake's algorithm, which involved setting up initial regions and comparing pixel values to get a notion of the segment map.

These techniques focused on pixel differences and gradients at the local level, taking a local perspective of an image's attributes. Much later, among the standard image processing techniques, methods like adaptive thresholding, Otsu's algorithm, and clustering algorithms were created. These algorithms took a global perspective of the input image, see fig. 1. All these methods depend on image processing and could be considered unsupervised techniques.

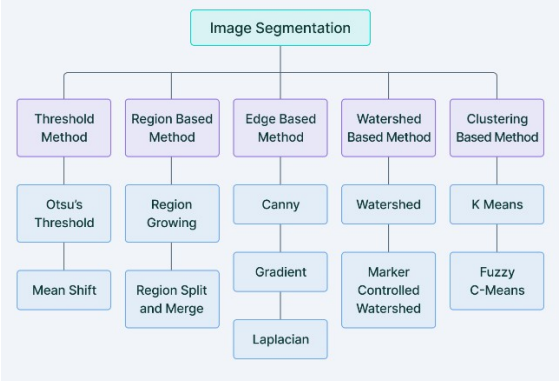

Fig. 1. Traditional Image Segmentation techniques (7).

2) Deep Learning-based methods: Segment maps are the outputs that semantic segmentation models produce in response to the inputs they are fed. The number of classes the model is designed to segment is represented by the number n in these segment maps. Each of these n-channels is binary in nature, with vacant regions made up of zeros and object locations ”filled” with ones. The ground truth map is an integer single-channel array with a range of ”n” segments that are ”filled” with the index values of the appropriate classes (classes are indexed from 0 to n-1). The ”n-channel” binary output of the model is also referred to as a two-dimensional one-hot encoded representation of the predictions.

As seen in fig. 2, neural networks often use an encoder, a bottleneck, and a decoder or upsampling layer that begins at the bottleneck as part of their encoder-decoder architecture.

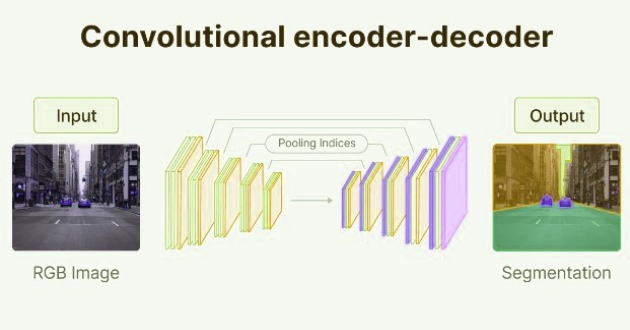

Fig. 2. Encoder-Decoder base architecture.

SegNet (6) encoder-decoder architectures for semantic segmentation gained popularity since the moment of its publication. SegNet suggests using downsampling and convolutional blocks combined to cram data into a bottleneck and create a representation of the input. The decoder then rebuilds the input data to produce a segment map, highlighting and categorizing input points. The output of the decoder is restricted to a specific range by a sigmoid activation in the final phase (0,1).

U-Net (46), which first introduced skip connections in Deep Learning as a remedy for the information loss noticed in downsampling layers of traditional encoder-decoder networks, was released concurrently with the publication of SegNet. Skip connections avoid the bottleneck entirely by going straight from the encoder to the decoder. In other words, several layers of encoded representations of feature maps are recorded and joined to form feature maps in the decoder. By aggressively downsampling and pooling data, as done in the encoder blocks of an encoder-decoder design, this helps to minimize data loss. Particularly in the field of medical imaging, Skip Connections were quite popular, with U-Net delivering cutting-edge outcomes in cell segmentation for illness identification.

Following UNet, DeepLab (12) by Facebook was a turning point in semantic segmentation research by delivering state-of-the-art results on semantic segmentation.

Atrous convolutions were used by DeepLab to replace straightforward pooling procedures and prevent severe information loss while downsampling. To help the network segment objects of all sizes, they also added multi-scale feature extraction with the aid of Atrous Spatial Pyramid Pooling.

They used completely connected conditional random fields to retrieve boundary information, one of the most crucial components of semantic and instance segmentation (CRFs).

DeepLab outperformed techniques like FCNs and SegNet by a significant margin when the recognition ability of CNNs and the fine-grained localization accuracy of CRFs were combined.

SegNet, U-Net, and DeepLab papers established the foundation for further work like Mask-RCNN (25), Facebook’s DeepLab series, and projects like PspNet (67) and GSCNN.

D. Studies On Road Segmentation

We are interested in two open-source data sets DeepGlope and Massachusetts. Automatically extracting roads and street networks from satellite images is a challenge posed by DeepGlobe. The Road Challenge training data includes 6226 RGB 1024x1024 satellite images. The data set includes 1101 test photos and 1243 validation images.

The Massachusetts Roads Data set contains 1171 aerial photos of Massachusetts. Each image has a dimension of 1500 by 1500 pixels and spans 2.25 square kilometers. We divided the data into three sets at random: an 1108-image training set, a 14-image validation set, and a 49-image test set. The data set spans more than 2600 square kilometers and includes a wide range of urban, suburban, and rural districts. Over 110 square kilometers are taken up only by the test set. Road centerlines from the OpenStreetMap project were rasterized to create the target maps. The labels were created using no smoothing and a line thickness of 7 pixels. One pixel per square meter resolution is applied to every imagery.

The authors of (1), construct a high-resolution road segmentation map and use the VNet model, a novel deep learning-based convolutional network. Furthermore, cross-entropy-dice-loss is a novel loss function known as dual loss (CEDL). It is a hybrid of cross-entropy (CE) and dice loss (DL). Dual loss considers both local and global information (CE and DL) to lessen the influence of class imbalance and improve the outcomes of route extraction. For the Massachusetts data set, the proposed VNET+CEDL strategy received an average f1 score of 90.64%. When compared to other cutting-edge deep learning-based frameworks like FCN, Segnet, and Unet, the suggested technique might enhance the outcomes by 1.09%, 2.45%, and 0.39% for the Massachusetts data set. The first VNet model was inspired by the U-Net architectural family (46), which blends lower-level and higher-level feature maps to achieve precise localization. The goal of this network design is to solve picture segmentation challenges effectively, particularly in the context of medical imaging.

In (2), the authors provide a deep learning technique for road segmentation from high-resolution aerial images based on generative adversarial networks (GANs). In the generative phase of the proposed GAN approach, they use a modified UNet model (MUNet) to generate a high-resolution segmentation map of the road network. When paired with simple pre-processing that incorporates edge-preserving filtering, the

proposed technique greatly improves road network segmentation when compared to previous methods. In testing on the Massachusetts road image data set, their technique obtains 91.54% precision and 92.92% recall, corresponding to a Mathews correlation coefficient (MCC) of 91.13%, a Mean intersection over union (MIOU) of 87.43%, and an F1-score of 92.20%.

In this study (23), they offer a technique for extracting roads from optical satellite pictures using a postprocessing step and a fine-tuned deep residual convolutional neural network (RDRCNN). The RDRCNN is composed of a dilated perception unit (DPU) and a residual connected unit (RCU). The symmetric RDRCNN structure produces outputs of the same size. Math morphology and tensor voting are used for RDRCNN postprocessing. The Massachusetts data set is utilized in research. The IOU they receive is 67.10%, and their f1-score is 80.31%.

In (55) conventional semantic segmentation networks can incorporate the spatial information inference structure (SIIS). The network with SIIS known as SII-Net can learn not only the local visual characteristics of the road but also the information about the global spatial organization (such as the continuity and trend of the road). As a result, it is able to successfully resolve the difficult occlusion problem in road detection and maintain the continuity of the extracted road. Two data sets were used in the experiments, and the findings demonstrate that the suggested strategy can enhance the overall performance of road extraction. They obtain an IOU of 68.22% and the f1-score is 83.58% for the DeepGlope data set.

In (61) to address the DS issue in this field, they suggest a brand-new stagewise domain adaptation paradigm called RoadDA. Through interdomain adaptation based on generative adversarial networks (GANs), RoadDA first aligns the target domain features with the source ones. A feature pyramid fusion module is specifically designed to prevent information loss due to lengthy and narrow roadways and to learn strong and discriminative features. In addition, they suggest an adversarial self-training strategy for the second stage to resolve the intradomain difference in the target domain. They obtain an IOU of 0.8521 and the f1-score is 0.9276 on the Massachusetts data set.

In (44) they suggest the ATD-LinkNet, a deep learning-based network with a number of specialized modules. In particular, they suggest an AT block, a replaceable module for ATD-LinkNet that uses multi-scale convolution and attention mechanisms as its building blocks. The AT block combines many scale features and efficiently makes use of the copious amounts of spatial and semantic data present in remote sensing photos. In the decoder portion of ATD-LinkNet, they employ the dense upsampling convolution to fine-tune the nonlinear bounds of interior objects in remote sensing images. The results show our ATD-LinkNet achieves 62.68% for mean Intersection over Union in the DeepGlobe Road Extraction data set.

In (20) they use three different models U-Net as a baseline model, Deep lab, and thirdly, describe a novel architecture they call Residual Inception Skip Net, which incorporates knowledge from a few well-known methods. Residual Inception Skip Net this model uses inception modules rather than conventional convolution blocks in its convolutional encoder-decoder architecture. The inception models are the ones that were first put forth, but with asymmetric convolutions. Since it uses fewer parameters and produces comparable performance, this is preferable. The He-norm is used to initialize all weights, and the batch norm is applied to all convolutions before activation. As our activation function, we employed a leaky RELU with a 0.1x slope. They experiment with the result on the Deep globe data set and get an MIOU for U-Net 59.0% over the test set, DeepLab gets 59.3%, and ResInceptionSkip gets 61.2%. ResInceptionSkip increases the performance over the test set by a 2 percent margin. The model's encoder is made up of five downsampling layers and a VGG network (50) with batch norm (54) added. Careful examination of the data set, task, and receptive field guided the decision of the network's depth (39). They made the choice to maintain 128 feature maps as a fixed quantity across the network. Based on the critical discovery that the network can afford to give up some representational power in the encoder half. This choice was taken because the model has access to low-level characteristics in the decoder portion via the skip connection. The decoder, like the encoder, employs deconvolution layers to upsample with a skip connection from the encoder, integrating the preceding decoder layer's deep representations with the more accurate spatial representations from the matching encoder layer. An activation function for the sigmoid is present in the final head. Following that, they apply the DeepLab model, which consists of a ResNet block with 5x5 convolutional layers and 3 x 3 kernels. In order to create a 1x1 convolutional layer, they employ a convolutional layer with stride 2 after 6 residual blocks. This creates four sections throughout the entire encoding network. In each of these portions, they employ 16 kernels, 32 kernels, 64 kernels, and 128 kernels. As a result, they are given a feature map with 128 dimensions and 1/8 of the original resolution. The decoder has three completely convolutional layers with a total of 64, 32, and 16 kernels, respectively. Each of these levels doubles the input resolution by upsampling. The feature map is scored in the last convolutional layer, which is followed by a sigmoid activation function. Accordingly, the entire network is made up of 5x5 convolutional layers for the encoder, 3 completely convolutional layers for the decoder, and a convolutional layer that outputs the class labels(12; 56). A stridden convolution is used for downsampling, and it is then followed by a batch norm and leaky RELU. Two by two upsampling convolutions are used for upsampling. Following the upsampling and downsampling layers is a 0.9 dropout. Similar to U-Net, they also include a skip connection connecting layers of the same size between the encoder and the decoder. The connections produced a residual block with a batch norm followed by a 1 x 1 convolution that was the sum of the input and the residual block. They tried scaling down the encoder layer as an experiment, but the performance was marginally lower. They experiment with the result on the Deep

globe data set and get an MIOU for U-Net 59.0% over the test set, DeepLab gets 59.3%, and ResInceptionSkip gets 61.2%. ResInceptionSkip increases the performance over the test set by a 2 percent margin.

Tables I, and III compromise the related work we discussed on both data sets Massachusetts, and DeepGlobe respectively.

TABLE I
RELATED WORK ON MASSACHUSETTS DATA SET.

Model	IOU	F1-score
FCN(50)	0.8197	0.9009
Segnet(6)	0.7983	0.8873
Unet(20)	0.8316	0.9079
Deep ResUnet(12)	0.8365	0.9102
DeepLabv3+(56)	0.8442	0.9196
SII-Net(55)	0.8521	0.9276
VNet_CEDL(1)	0.8382	0.9118
GAN+MUNet(2)	87.43	92.20
RDRCNN (23)	67.10	80.31

TABLE II
RELATED WORK ON DEEPGLOBE DATA SET.

Model	IOU	F1-score
RoadDA(61)	0.8535	0.9235
D-LinkNet50(44)	0.6112	0.6992
ATD-LinkNet50(44)	0.6268	0.7023
U-Net(20)	0.5900	–
DeepLab(20)	0.5930	–
ResInceptionSkip(20)	0.6120	–

E. Inference From The Studies

In this section, we deeply study each approach illustrated in the section above and give a brief inference of these studies, The best performance that could obtain from these studies, and The tricks used to increase the accuracy. Each approach has its own limitation, we give some examples of these approaches and study how we can increase the performance of some of them.

The proposed VNet (1) technique's left component has a compression route, while the right portion decompresses the input till it reaches its original size. All convolutions are performed with appropriate padding, with the purpose of using input features while decreasing resolution by applying the appropriate stride at the end of each step. The proposed VNet network design is similar to the commonly used Unet concept, but with notable modifications.

Researchers present a new dual objective loss function (CEDL) in this study (1) that combines both cross-entropy loss function (CE) and dice coefficient (DL) for reducing the impact of class imbalance issues because they have the same problem of imbalance classes, such as road pixels representing the foreground and non-road pixels representing the background. DL simply produces a scalar, but CE returns a tensor for each picture in the batch (Local Information) (Global Information). We merged global and local information to better extract the road network (CE). They train the VNet model with three different Loss functions VNet+CE, VNet+DL, and VNet+CEDL. They assessed the accurate measurements of the three approaches on the Massachusetts dataset. They achieved an IOU of 0.8018, 0.8159, and 0.8207 respectively. The presented approach Vnet+CEDL could achieve the highest accuracy because of the CEDL loss function which targets the imbalanced data problem. The limitation of this paper is that they used only the VNet model but the same approach of the CEDL loss function could be used with other models to obtain the best one.

Additional high-level semantic information is necessary to improve road detection performance and more effectively control occlusions in (2). The authors use a generative adversarial network (GAN) technique to handle road segmentation using remote sensing data. The GAN algorithm combines a generating network, which extracts road networks from an input satellite picture, with a discriminator network, which attempts to distinguish between road networks formed by the generator and those from ground truth labels. In a max-min configuration, the generator creates the most complicated route plan possible for the discriminator, which is striving to lower its mistake. There haven't been many studies on road semantic segmentation using the GAN model. They use a modified U-Net called MUNet as a generator network. The complete training and evaluation process for the suggested GAN+MUNet road network extraction strategy is divided into five main steps: The proposed method includes the following steps: (i) creation of training and test samples; (ii) local Laplacian filtering, LLF for short image quality enhancement; (iii) optimization of the GAN using the training samples; (iv) extraction of the road network using the generator from the optimized proposed GAN from images in the test set; and (v) performance quantification for the proposed method using a common metric. They also, compare their result on different GAN-based models which use other types of generator networks such as GAN+FCN architecture proposed by (62), GAN+SegNet presented by (48), Ensemble Wasserstein Generative Adversarial Network (E-WGAN) proposed by (59), Multi-supervised Generative Adversarial Network (MsGAN) performed by (64), and Multi-conditional Generative Adversarial Network (McGAN) implemented by (63). The MUNet achieves an F1-score of 90.18%, on the other hand, the GAN-MUNet archives a 92.20%. With compare the result of the GAN-MUnet with the other gan-based models mentioned above we found that the GAN-MUNed achieves the highest accuracy. For example GAN+FCN archives 87% f1-score, and GAN+Segnet achieves 89.63% f1-score.

The authors of this work (23) propose a model approach for extracting roads from high-resolution images. A postprocessing step and a refined deep residual convolutional neural network (RDRCNN) architecture are used in the method. The symmetric RDRCNN architecture is made up of the residual connected unit (RCU) and the dilated perception unit (DPU). In contrast to the prior techniques, the proposed method uses texture information to display high-level characteristics. Because a pretrained network can extract rich and distinctive high-level representations for visual objects, this information enhances extraction decisions without the need for any manual

specialised spectrum information approach. The RDRCNN architecture, an end-to-end symmetric training system, was inspired by ResNet, U Net, and Deeplab.The RCU and DPU are two of the basic components of RDRCNN, which is then followed by a complete convolution layer. In particular, at road intersections, RDRCNN recognizes road regions but does not guarantee continuous road areas. However, it may result in broken roads that were prevented from being traveled by trees or shadows in the RDRCNN outputs. A postprocessing step is utilized to improve topology expression and decrease broken sections in order to resolve this drawback. RDRCNN uses a Cross-entropy loss function which is not the perfect loss function for this problem due to, the imbalanced classes problem. Dice loss (DL) is better for capturing global information or using CEDL may lead to better results.

In (44) they implement ATD-LinkNet architecture. The encoder components of the overall network consist of 4 levels of downsampling blocks. And in this section, they swap out the Residual blocks that the baseline network employed as feature extractors with the AT building block. The AT building block they suggested, as opposed to the residual block, uses the attention module and multi-scale module to extract the context semantic data more thoroughly. A better feature extractor for the residual block is the AT building block. This is also the rationale for their suggested AT construction block, as it does not adversely influence the original D-LinkNet structure during the replacement of the remaining block. D-LinkNet is added after the encoder section of ATD-LinkNet, taking into account the complexity and coherence of the object in the remote-sensing image. It has been shown that dilated convolution is particularly good at preserving the spatial and semantic information of deep feature maps, so ATD-LinkNet adds more dilated convolutional. They employ a number of data augmentation techniques, such as random picture rotation, flipping, rescaling, and shift, among others, throughout the training phase to avoid the model from overfitting. Adam optimizer and the binary cross-entropy (BCE) loss function are used for training the ATD-LinkNet. BCE loss function is not the best loss function for this problem we employ another training strategy with different loss functions like Dice loss (DL), or Combined Cross-Entropy with Dice loss which is called CEDL. These may lead to better generalization and reduce the overfitting problem.

Encoder-Decoder Deep Convolutional Neural networks have been proven great performance at many segmentation issues. Our suggestion is to apply all current model architectures that use the DCEP network as their primary base model to two open-source data sets from DeepGlope and Massachusetts. We choose 9 different models Unet (34), FPN (24), PSPNet (67), Unet++ (68), PAN (57), LinkNet (9), DeepLabv3(15), DeepLabv3+ (5), and MA-Net (35). Also, from inference, we found that the Dice Loss function is the most appropriate loss function that could obtain robust training, and solve the imbalanced data problem. The Dice Loss is formulated as follows:

$$DL = 1 - \frac{2\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i^2 + \sum_{i=1}^{N} g_i^2} \quad (1)$$

Where $pi \in P$ is the i-th expected probability. pixel and $gi \in G$ is the i-th ground truth pixel.

F. Conclusion

We show different approaches for road segmentation problems some of them relying on the Encoder-Decoder approach, others relying on a Generative adversarial network (GAN), and others relying on a Fully convolution network. The GAN and Encoder-Decoder approaches seem to be promising approaches based on their results. We also study different training techniques based on different loss functions like Binary cross entropy (BCE), Dice loss, and Cross-Entropy Dice loss function (CEDL), which combines the two loss functions to improve the model's generalization and solve the problem of overfitting. Finally, we suggest our work depends on the inference found in these studies.

III. PROPOSED APPROACH

A. Introduction

Our objective is to apply to two open-source data sets from DeepGlope and Massachusetts all current model designs that employ the DCEP network as their principal base model. For our data sets, we choose 9 different models which are, Unet (34), fpn (24), PSPNet (67), UnetPlus (68), Pan (57), LinkNet (9), deeplabv3 (15), deeplabv3-plus (5), and MA-Net (35).

Fig. 3 illustrates our pipeline, and shows every module we apply to get our trained segmentation models. We apply the pipeline twice, one without the Bilateral filter and the other with the filter. So, for each data set, we use we train the 10 models twice one with the filter and the other without the filter, and compare the results to get the best models that represent our work.

A non-linear, noise-reduction, and edge-preserving image-smoothing filter are known as a bilateral filter. The intensity of each pixel is adjusted to be a weighted average of its surrounding pixels' intensities. This weight might have a Gaussian distribution. Importantly, the weights are determined by the Euclidean distance of the pixels as well as their radiometric inconsistencies (e.g., range differences, such as color intensity, depth distance, etc.). Sharp edges are so preserved.

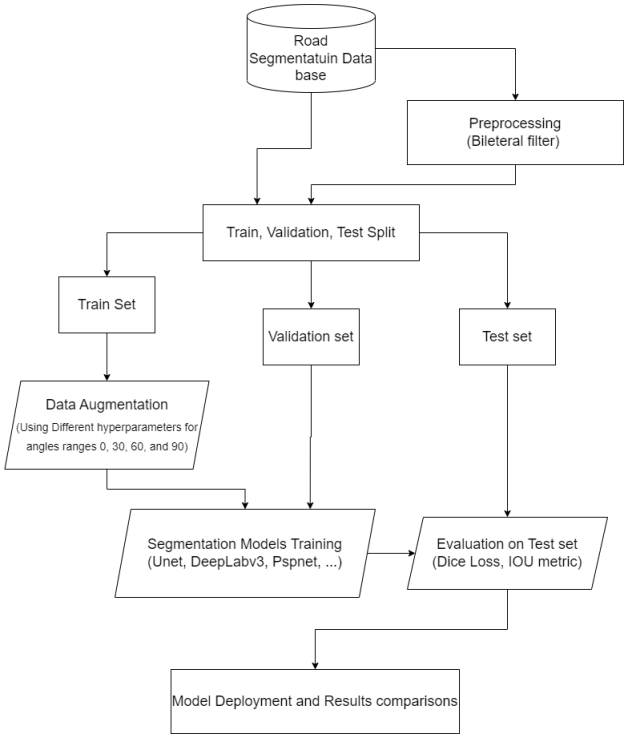

Fig. 3. The Proposed Framework.

B. Data Collection

Two open-source data sets, DeepGlope (19) and Massachusetts (40), caught our attention. Data set for DeepGlope Maps and accessibility information is essential in disaster zones, especially in poor nations, for crisis response. The issue of autonomously extracting roads and street networks from satellite images is posed by the DeepGlobe Road Extraction Challenge. The Road Challenge training data includes 6226 RGB 1024x1024 satellite images. The satellite imagery, which was acquired by DigitalGlobe, has a 50 cm pixel resolution. There are 1101 test photos and 1243 validation images in the dataset (but no masks).

The Massachusetts Roads Data collection contains 1171 aerial images. Each picture is 2.25 square kilometers in area and 1500 by 1500 pixels in quality. The data was divided into three sets at random: a training set of 1108, a validation set of 14, and a test set of 49 photos. The data collection includes about 2600 square kilometers of urban, suburban, and rural sectors. The test site alone spans approximately 110 square kilometers. The target maps were created using OpenStreetMap rasterized road centerlines. The labels were created with a 7-pixel line thickness and no smoothing. The resolution of each photograph is one pixel per square meter.

TABLE III
DATA SETS DESCRIPTION

Data	Image size	Num. of train img.	Num. of valid img.	Num. of test img.
DeepGlope	1024x1024	6226	1243	1101
Massachusetts	1500x1500	1108	14	49

C. Data Augmentation

We must artificially enlarge our data set to get around the overfitting issue. We can increase the size of the data set we already have. The goal is to replicate the differences observed when someone captures a photo or video by making small changes to the training data.

Data augmentation techniques are ways to change the training data while keeping the label and changing the array representation. There are numerous augmentations that are often utilized, including random crops, color hiccups, translations, rotations, and horizontal and vertical flips.

By making just a few of these changes to our training data, we can quickly double or quadruple the number of training examples and create a very strong model.

We apply several data augmentation processes that applied for both the input image and the mask. We apply vertical and horizontal flips, Random crop, and Rotation in the random range from 0 to 90 degrees. To set the parameters of augmentation that we used, we apply several searches for stable training, and this the best parameters that fit our needs.

D. Segmentation Algorithms

Here we give a summary of the Deep learning approach we are interested in, and what open-source data sets we will use in our research. The use of computer vision algorithms for comprehending satellite images has a long history (16; 29). In the past, satellite imagery was often of lesser resolution, with a variety of spectral bands, and a top-down perspective. Deep learning-based segmentation techniques have become more popular recently. Many studies (14; 58) have worked using the fully convolutional network (FCN) since it has demonstrated various improvements in semantic segmentation. The FCN is the foundation for the network model developed in this paper.

One of the most popular encoder-decoder networks is Unet(30) connects the network's encoder and decoder features, combining low-lever and high-level information (such as an hourglass shortcut connection structure known as U-shape), using Transposed-convolution (47) as its upsampling structure on the basis of FCN. The Unet model is one of the first architectures that employ Encoder-Decoder in its main structure. Other works that follow Unet model publication, try to improve the structure by improving the Encoder, and Decoder connectivity and structure, but they utilize the Encoder-Decoder architecture as it's the main structure of the proposed models.

We apply a variety of current model designs that employ the DCEP network as their principal base model. For our data sets we choose 9 different Unet(34), fpn(24), PSPNet(67), UnetPlus(68), Pan(57), LinkNet(9), DeepLabv3(15), deeplabv3-plus(5), and MA-Net(35). We illustrate each model of them in the next subsections.

1) Unet: A revolution in deep learning was brought about by the U-Net architecture (34), which was initially published in 2015. In a number of areas, the design easily defeated the competition during the 2015 International Symposium on Biomedical Imaging (ISBI) cell tracking challenge. Segmenting neural

structures in transmitted light microscopy images and electron microscopy stacks are a few of their accomplishments (34).

The U-Net(34) is a wonderful architecture that addresses the majority of problems that arise. This method makes advantage of the idea of fully convolutional networks. The purpose of the U-Net is to record both the localization and context features. The kind of architecture that was constructed successfully completes this procedure. The fundamental idea behind the implementation is to use upsampling operators directly after successive contracting layers in order to produce outputs with greater resolution on the input images.

We can see why the design in fig. 4 is likely referred to as U-Net architecture by taking a quick glance at it. The following term is derived from the shape of the so-formed architecture, which is in the shape of a "U." We can tell that the network produced is a fully convolutional network just by looking at the structure and the many components used in the building of this architecture. They did not employ any additional layers, such as dense, flat, or layers of a similar nature. The visual representation demonstrates a path that initially contracts before expanding.

The model's architecture demonstrates how an input image is processed before going through a few convolutional layers with the ReLU activation function. The image size decreases from 572X572 to 570X570 and then to 568X568 as can be seen. The usage of unpadded convolutions (which characterized the convolutions as "valid") led to a reduction in overall dimensionality, which is the cause of this reduction. In addition to the Convolution blocks, we can also notice the encoder block on the left and the decoder block on the right(30).

The max-pooling layers of stride 2 assist the encoder block in maintaining a steady reduction in image size. The encoder architecture also includes repeated convolutional layers with a growing number of filters. When we get to the decoder part, we see that the convolutional layers' number of filters starts to go down and that the subsequent layers gradually upsample until we get to the top. We also observe the application of skip connections, which link the decoder blocks' layers with earlier outputs.

This skip connection is a key concept for the loss from the prior levels to reflect more strongly on the overall values. Furthermore, they have been shown scientifically to improve results and faster model convergence. In the last convolution block, a few convolutional layers come after the final convolution layer. This layer contains a filter with the appropriate purpose for displaying the output. We can change this last layer based on the intended result of the project you're working on.

A multi-channel feature map that corresponds to each blue box is seen in fig. 4. A channel count indicator is located on top of the box. The x-y size is shown in the box's lower left corner. White boxes are used to indicate copied feature maps. The arrows represent the various operations.

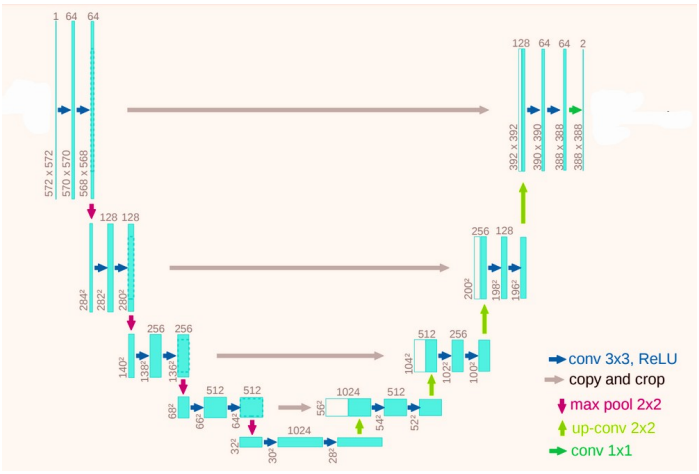

Fig. 4. The proposed UNet architecture.

2) Unet++: A high-level view of the suggested architecture is presented in fig. 5. As can be seen, UNet++ begins with a backbone or encoder sub-network before moving into a decoder sub-network. The redesigned skip paths (shown in green and blue) that connect the two sub-networks and the use of deep supervision set UNet++ apart from U-Net (the black components in fig. 5a) (shown red). (a) UNet++ is composed of a pair of encoders and decoders connected by nested convolutional blocks. UNet++'s primary purpose is to bridge the semantic gap between the encoder's feature maps. A dense convolution block with three convolution layers, for example, is employed prior to fusion and decoder to bridge the semantic gap between $(X0, 0)$ and $(X1, 3)$. The original U-Net is depicted in black, convolution blocks on skip pathways in green and blue, and deep supervision in red. The employment of red, green, and blue components distinguishes UNet++ from U-Net. (a) A thorough examination of UNet++'s first skip path. (c) If UNet++ is trained with close monitoring, it can be pruned at any time. (68).

The link between the encoder and decoder subnetworks is replaced by newly constructed skip routes. The encoder's feature maps are received directly by the decoder in the U-Net model, but in UNet++, they pass via convolution blocks, the number of which varies according to the pyramid level. Each convolution layer in the skip pathway, for example, between the nodes of $X0, 0$ and $X1, 3$ is followed by a concatenation layer that concatenates the output from the previous convolution layer of the same dense block with the equivalent up-sampled output of the lower dense block. There are three convolution layers in this convolution block. The dense convolution block effectively elevates the semantic level of the encoder feature maps to that of the feature maps awaiting decoding. When the incoming encoder feature maps and matching decoder feature maps are semantically equivalent, the optimizer has an easier time solving the optimization issue.(68).

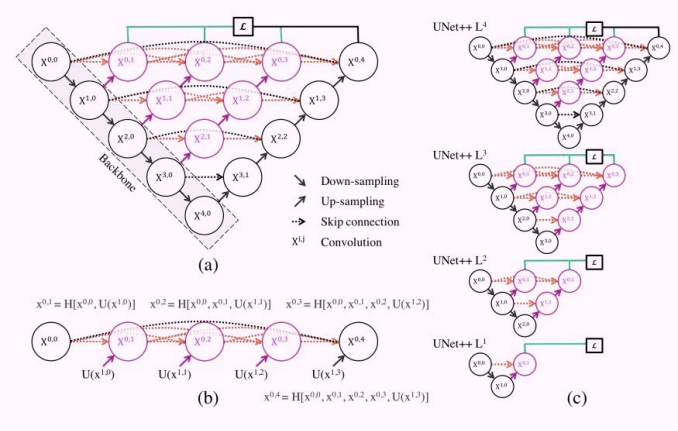

Fig. 5. Overview of the UNet++ architecture.

3) PSPNet: A better framework for pixel-level prediction is offered by PSPNet (67). On a variety of data sets, the suggested technique produces state-of-the-art performance. In the 2016 ImageNet scene parsing challenge, the 2012 PASCAL VOC benchmark, and the 2016 Cityscapes benchmark, it took first place.

Pyramid-pooled feature maps from several levels were finally flattened and combined to form a completely connected layer for classification. The goal of this global prior is to remove CNN's fixed-size constraint for picture classification. In order to further decrease context information loss between different sub-regions, they propose a hierarchical global prior that comprises information with varying scales and varies between unique sub-regions.

The authors of the main PSPNet architecture (67) suggest pyramid pooling module combines elements from four distinct pyramid scales. Firstly, the global pooling layer is responsible for creating a single bin output is the thickest level, which is highlighted in red as seen in fig. 6. The following pyramidal level separates the feature map into multiple sub-regions and generates pooled representations for distinct locations. The output of the pyramid pooling module's several phases includes a feature map of varying sizes. If the level size of the pyramid is N, the dimension of the context representation is decreased to 1/N of the original one while keeping the weight of the global feature, using a 1x1 convolution layer after each pyramid level. The low-dimension feature maps are then instantly upsampled using bilinear interpolation to achieve the same size feature as the original feature map. Concatenating several feature levels results in the final pyramid pooling global feature.

Fig. 6 shows an overview of the PSPNet(67) architecture. The authors begin by utilizing CNN to extract the feature map of the last convolutional layer in order to produce the final feature representation, which comprises both local and global context information (b). Following that, a pyramid parsing module is utilized to obtain multiple sub-region representations (c). The representation is then fed into a convolution layer to obtain the final per-pixel prediction (d).

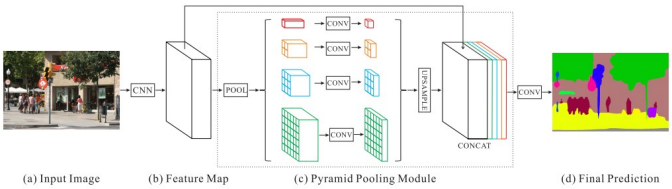

Fig. 6. Overview of the PSPNet architecture.

4) LinkNet: Fig. 7 shows the LinkNet (9) architecture. Conv and full-conv in this context refer to convolution and full convolution, respectively. Additionally, /2 indicates downsampling by a factor of 2, which is accomplished through stridden convolution, and 2 implies an upsampling by a factor of 2, which is accomplished through stridden convolution. We employ batch normalization between each convolutional layer, which is followed by ReLU non-linearity. The encoder is located on the left half of the network in fig. 7, while the decoder is located on the right. The encoder begins with a first block that convolutionally transforms the input image using a kernel with a size of 7 by 7 and a stride of 2. Additionally, this block conducts spatial max-pooling in a 3 by 3 area with a 2 stride. The remaining blocks that make up the later part of the encoder are referred to as encoder blocks (i) (10).

Similarly, fig. 7 provides layer information for decoder blocks. While m = n = 64 for the first block of encoder and decoder and $n_{encoder} = m_{decoder} = 64 * 2^i$ for the remaining blocks.

Networks with a large number of parameters and GFLOPs, such as ResNet101 (45 million parameters) and VGG16 (138 million parameters), are used as the encoder in modern segmentation techniques. ResNet18, a relatively lightweight network that LinkNet employs as an encoder, outperforms them, as shown in Section IV. In our decoder, we apply the full-convolution technique that was first suggested by. There are three parameters for each Conv(k x k)(im, om) and full-Conv(k x k)(im, om) operation. In this case, (k x k) stands for "kernel size" and (im, om), "input map," and "output map," respectively.

Their innovation comes from the method they connect each encoder and decoder, which is different from other neural network architectures that are currently being utilized for segmentation. Some spatial information is lost as a result of the encoder's successive downsampling procedures. It is challenging to restore this lost data using only the encoder's output that has been downscaled. Each encoder layer's input is also bypassed in this study and sent directly to the matching decoder's output. By doing this, we hope to retrieve lost spatial data that the decoder and its upsampling algorithms can employ. Additionally, the decoder can use fewer parameters because it is sharing the information that the encoder learned at each layer. When compared to the current state-of-the-art segmentation networks, this leads to an overall more efficient network and, consequently, real-time operation(10).

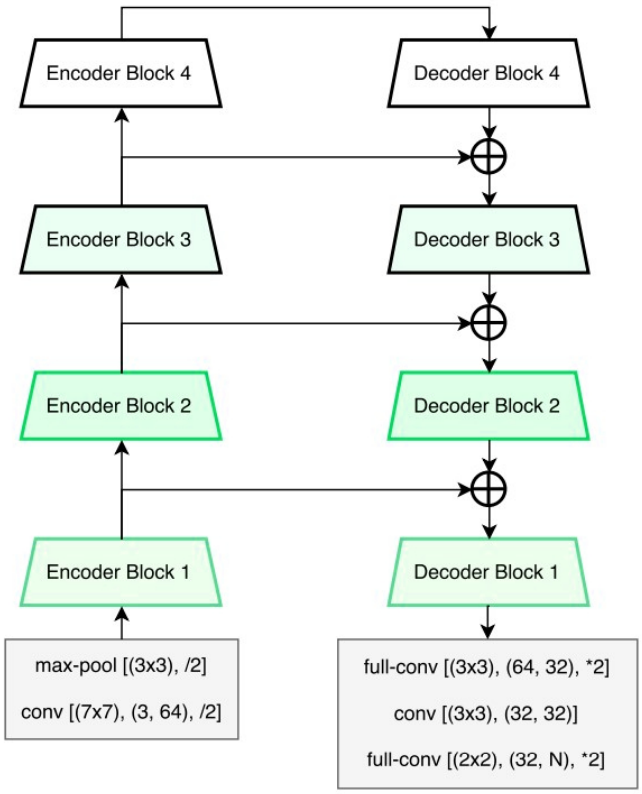

Fig. 7. Overview of LinkNet Architecture.

5) Deeplabv3: In Deeplabv3 (13), the authors review atrous convolution, a useful technique for controlling the resolution of feature responses calculated by Deep Convolutional Neural Networks as well as explicitly adjusting the filter's field of view, in the context of semantic picture segmentation. To overcome the difficulty of segmenting objects at different sizes, they create modules that capture multi-scale context using atrous convolution in parallel or cascade.

Fig. 8 shows atrous convolution with a 3 x 3 kernel and various rates. The atrous convolution at a rate of 1 corresponds to standard convolution. By using a high atrous rate, the model's range of view is expanded, allowing for object encoding at various scales.

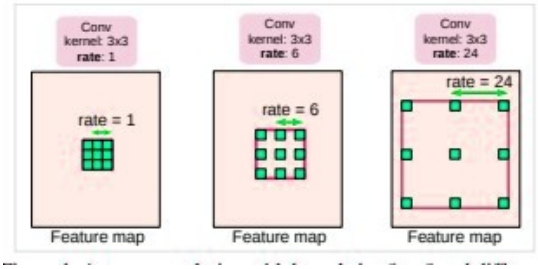

Fig. 8. Atrous convolution with a 3 x 3 kernel and various rates.

The authors first investigate creating modules with asymmetric convolution arranged in cascade. In order to illustrate, they create many copies of the last ResNet block, block4 in fig.9, and arrange them in cascade. In those blocks, there are three 3x3 convolutions, and, like the original ResNet, the last convolution contains stride 2 aside from the one in the last block.

This approach was developed because the added striding makes it simple to gather long-range information in the deeper blocks. For instance, the final small-resolution feature map, as shown in fig.9, might provide a summary of the entire image feature (a). Due to the reduction of detailed information, we learn that sequential striding is harmful to semantic segmentation. As a result, they apply atrous convolution at rates based on the desired output stride value, as illustrated in fig.10 (b), where output stride = 16.

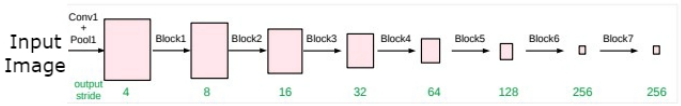

Fig. 9. (a) Going deeper without atrous convolution.

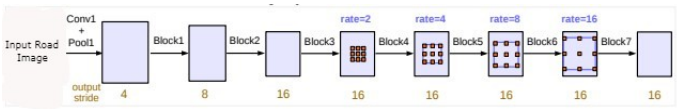

Fig. 10. (b) Arous convolution for deeper exploration. When the output stride equals 16, atrous convolution with a rate larger than 1 is used after block 3. Without and with arousal convolution, cascaded modules.

The authors reviewed the Atrous Spatial Pyramid Pooling method from (12), which applies the feature map to four parallel atrous convolutions with various atrous rates. The success of spatial pyramid pooling served as an inspiration for ASPP because it demonstrated that it is efficient and accurate to classify regions of any scale by resampling features at various stages. They incorporate batch normalization within ASPP in contrast to (12).

Information from several scales is efficiently captured by ASPP with various atrous rates. However, the number of genuine filter weights (i.e., the weights that are applied to the valid feature region, instead of padding zeros) decreases as the sampling rate increases. Applying a 3 x 3 filter with various atrous rates shows this impact. The 3 x 3 filter degenerates into a simple 1 x 1 filter in the extreme scenario where the rate value is close to the feature map size because only the center filter weight is useful, rather than collecting the entire image context.

The authors use image-level features, similar to those used by (37; 67), to get over this issue and add global context information into the model. The model's final feature map is pooled globally, and the resulting image-level features are put into a 1 x 1 convolution with 256 filters (along with batch normalization) before being bilinearly upsampled to the desired spatial dimension. With 256 filters and batch normalization, improved ASPP finally consists of (a) one 1x1

convolution and three 3x3 convolutions with rates = (6, 12, and 18) when outputting stride = 16, and (b) the image-level features, as illustrated in fig. 11. The output features from each branch are then merged and subjected to a second 1x1 convolution (again with 256 filters and batch normalizing) before the final 1x1 convolution.

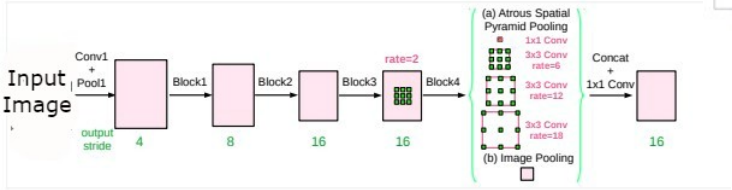

Fig. 11. Image-level features are added to parallel modules with atrous convolution (ASPP).

6) Deeplab-V3+: DeepLabv3 employs atrous convolution to get the features extracted by deep convolutional neural networks at any resolution. Deeplab-v3+ (15) the output stride is defined as the ratio of the input image's spatial resolution to the eventual output resolution (before global pooling or a fully-connected layer). The output stride for the image classification job is often 32 since the spatial resolution of the final feature maps is typically 32 times lower than the resolution of the input pictures. By removing the striding in the final one or two blocks and applying the appropriate atrous convolution, one can use output stride = 16, or 8 for denser feature extraction for the task of semantic segmentation. For example, for output stride = 8, the authors apply rate = 2, and rate = 4 to the final two blocks.

DeepLabv3 further improves the atrous Spatial Pyramid Pooling module using image-level features, which analyses convolutional features at different sizes by applying atrous convolution at different rates.

The authors employ the final feature map before logits from the original DeepLabv3 as the encoder output in the proposed encoder-decoder arrangement. Take notice that the encoder's feature map contains 256 channels of rich semantic data. Furthermore, depending on the computing budget, the atrous convolution may be used to extract features at any resolution.

Fig. 13 shows the architecture of the DeepLab-v3+. DeepLab-v3 is expanded by the further suggestion of DeepLabv3+ which employ an encoder-decoder structure. By using atrous convolution at various scales, the encoder module encodes multi-scale contextual information, and the straightforward but efficient decoder module enhances the segmentation outcomes at object boundaries (11).

The authors enhance DeepLab-v3 with the encoder-decoder structure, which uses the spatial pyramid pooling module shown in fig. 12(a), and (b) show that Rich semantic information from the encoder module is present in the suggested model, DeepLabv3+. They utilize a simple effective decoder module that recovers the precise object boundaries. They use atrous convolution on the encoder module to extract features at any resolution (11).

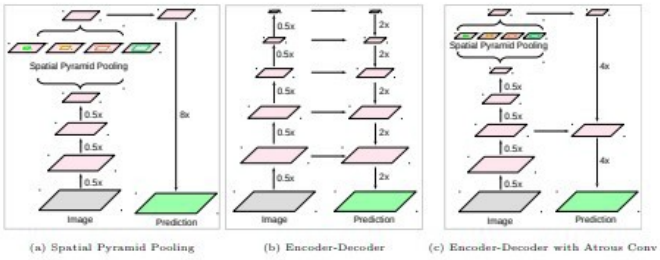

Fig. 12. Spatial Pyramid Pooling, and Encoder-Decoder structure.

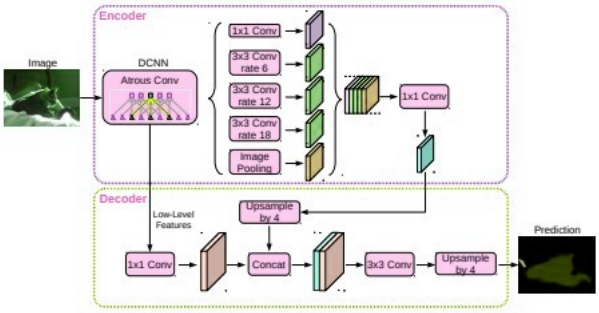

Fig. 13. The architecture of DeepLab-v3+ segmentation model.

7) FPN: This approach produces completely convolutional feature maps at various levels with appropriately sized pixels from a single-scale image of any size as the input. The results employing ResNets have presented in this study and the base FPN models are independent of the underlying convolutional designs. As shown in the subsequent sections, our pyramid is built using lateral connections, a top-down pathway, and a bottom-up method.

Bottom-up pathway The bottom-up feed-forward computation of the backbone ConvNet computes a feature hierarchy composed of feature mappings at various sizes with a scaling step of two. We refer to layers that are in the same network stage as those that frequently provide output maps of the same size. Each stage has its own pyramid level in our feature pyramid. Our reference collection of feature maps, which they will enlarge to form our pyramid, is the output of the final layer of each stage. The strongest traits should be found in each stage's deepest layer, therefore this choice makes sense.

For ResNets, they use the feature activations created by the final residual block of each step. We refer to the output of these last residual blocks as "C2, C3, C4, C5" for the conv2, conv3, conv4, and conv5 outputs. We also note that these outputs have strides of "4, 8, 16, 32" pixels relative to the input image. they omit conv1 from the pyramid because of its substantial memory requirements.

Top-down pathway and lateral connections.

The top-down pathway is responsible for upsampling spatially coarser but semantically stronger feature mappings from higher pyramid levels to provide higher-resolution features that are hallucinatory. Through lateral connections, these features are then improved with features from the bottom-up pathway.

Each lateral link combines feature maps from the top-down and bottom-up pathways that are the same spatial size. The bottom-up feature map has lower-level semantics, but because it was subsampled less frequently, its activations are more precisely localized.

The component that creates our top-down feature maps is depicted in fig. 14. We use the nearest neighbor upsampling to simply increase the spatial resolution of a coarser-resolution feature map by a factor of 2. The related bottom-up map, which has undergone an 11 convolutional layer to lower channel dimensions, is then combined with the upsampled map using element-wise addition. Up till the highest resolution map is produced, this process is iterated. We simply attach a 1x1 convolutional layer on C5 to begin the iteration in order to create the map with the lowest resolution. To create the final feature map and lessen the aliasing effect of upsampling, we add a 3x3 convolution to each of the merged maps. The designation of this last series of feature maps, P2, P3, P4, P5, corresponds to C2, C3, C4, C5 that are respectively of the same spatial sizes.

They fix the issue of feature dimension (numbers of channels, abbreviated as d) in all the feature maps since the shared classifiers/regressors used by all levels of the pyramid are the same as those used in a conventional pyramid of feature-rich images. In this study, they set d = 256, resulting in 256-channel outputs for all additional convolutional layers. These additional layers do not contain non-linearities, which we have empirically found to have negligible effects.

Their concept is robust to various design choices, and simplicity is a key component of our design. We have tried with more complex building blocks (using multilayer residual blocks as the links, for example), and we have seen slightly better outcomes. We choose the straightforward approach mentioned above because of improving connection module design.

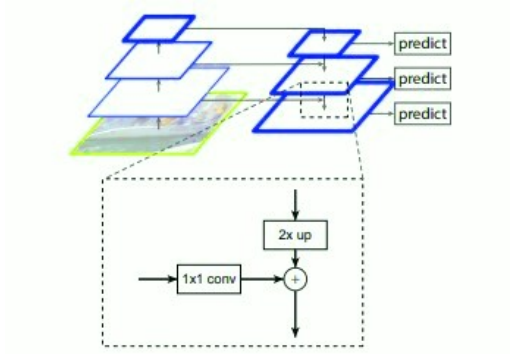

Fig. 14. A construction block that, by addition, demonstrates both the top-down pathway and the lateral link.

8) PAN: In this section, we first introduce the Global Attention Upsample (GAU) and Feature Pyramid Attention (FPA) modules that have been proposed. The Pyramid Attention Network (PAN) (57), a comprehensive encoder-decoder network architecture created for semantic segmentation tasks, is then described.

Fig. 15 Pyramid Attention Network overview In order to extract dense features, we use ResNet-101. To obtain precise pixel prediction and localization information, we then perform FPA and GAU. The downsample and upsample operators are shown by the blue and red lines, respectively.

they examine how to deliver accurate pixel-level attention for high-level characteristics retrieved from CNNs in the spirit of the Attention Mechanism. The pyramid structure fig. 16 in the current semantic segmentation architecture can effectively increase receptive field at the pixel level and extract different scales of feature information, but it lacks the global context-aware prior attention required to select features channel-wise as in SENet and EncNet. However, using a channel-wise attention vector alone is insufficient to successfully extract multi-scale features and is deficient in pixel-wise information.

The authors use a Global Attention Upsample (GAU) module to represent the global context as a direction for low-level characteristics to pick category localization details (fig. 17 computes global average pooling. They particularly perform 3 x 3 convolution on the low-level features to reduce channels of feature maps from CNN. For the objective of constructing a global context from high-level features, a 1 x 1 convolution with batch normalization and ReLU non-linearity is utilized, which is then multiplied by low-level features. Weighted low-level features are introduced after a steady upsampling and inclusion of high-level features. This module uses high-level features to simply direct low-level feature maps, enabling for more effective deployment of different-scale feature maps.

Fig. 16 shows the attention module structure of the feature pyramid. (a) The structure of the spatial pyramid pooling. (b) The attention module of the feature pyramid. The feature map's resolution is displayed as "4x4, 8x8, 16x16, and 32x32'. The branch for global pooling is shown by the dotted box. The downsample and upsample operators are shown by the blue and red lines, respectively. Keep in mind that batch normalization comes after every Convolution layer.

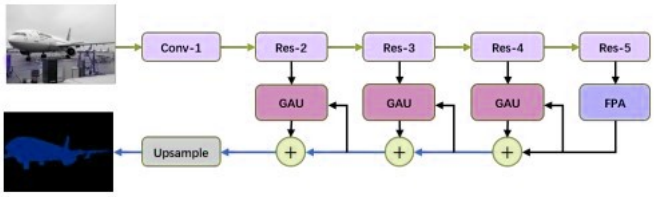

Fig. 15. Pyramid Attention Network (PAN) Architecture.

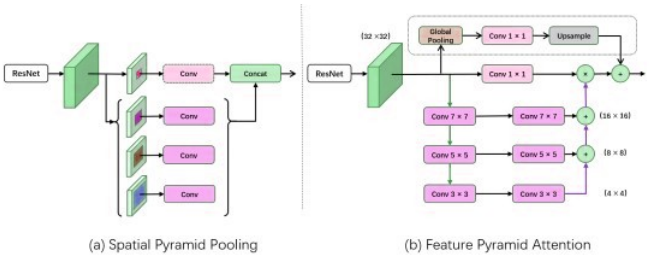

Fig. 16. The attention module structure of the feature pyramid. (a) The structure of the spatial pyramid pooling. (b) The attention module of the feature pyramid.

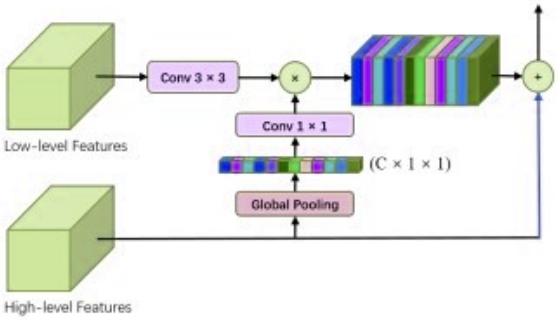

Fig. 17. Global Attention Upsample module structure.

9) MA-Net: The authors propose a new network dubbed a Multi-scale Attention Net (MA-Net) that employs a self-attention mechanism to adaptively integrate local information with global dependencies. The MA-Net, which is based on the attention mechanism, may record complex contextual interactions. The Position-wise Attention Block (PAB) and the Multi-scale Fusion Attention Block (MSFAB) are used by the writers (MFAB). The PAB is in charge of modeling the spatial dependencies between features, which correspond to the spatial dependencies between pixels in a global viewpoint. The MFAB separates and performs multi-scale semantic feature fusion to capture the channel relationships between each feature map (33).

The authors suggest the Multi-scale Attention-Net (MA-Net) for tumor and liver segmentation (shown in fig. 18. The MA-Net employs the self-attention mechanism. The authors particularly utilize two blocks based on self-attention techniques to record the spatial and channel correlations of feature maps. The first and second are the Position-wise Attention Block (PAB) and the Multi-scale Fusion Attention Block, respectively (MFAB). To gather the spatial relationships between pixels in feature maps, the PAB employs a self-attention approach. To capture the channel dependencies between any feature maps, the MFAB employs an attention approach. The MFAB considers the channel dependencies of low-level feature maps in addition to those of high-level feature maps. The channel dependencies of high-level and low-level feature maps are added together. in order to improve network performance and obtain rich Multi-scale semantic information of feature maps.

Position-wise Attention Block (PAB) and Multi-scale Fusion Attention Block are the first and second, respectively (MFAB) are shown in fig. 19 and fig. 20 respectively. The figures briefly show each module's design and its impact on the total network in fig. 18.

As observed from fig. 20, Multi-scale Fusion Attention Block (MFAB) employs two SE-Blocks as main blocks for capturing the Low-level and High-level feature maps respectively. The final channel attention feature map is obtained by using a concatenation layer.

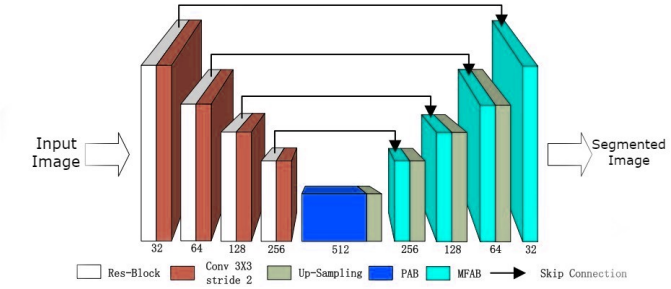

Fig. 18. The architecture of MA-Net.

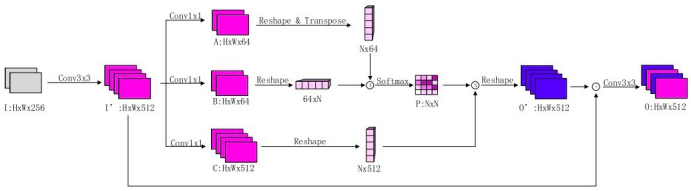

Fig. 19. Position-wise Attention Block (PAB). While the input image is HxWx256, the final image is HxWx512. Finally, the attention feature map is obtained using the softmax function.

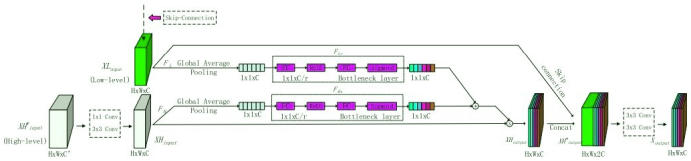

Fig. 20. The Multi-scale Fusion Attention Block (MFAB).

E. Evaluation Metrics

Working with these assessment metrics primarily serves to determine how well a machine learning model will perform on new data. This section focuses on two measures that provide a quick overview of how our model works. Intersection-Over-Union (IoU) and Dice Coefficient are the two measures. We quickly outline each of them, as well as other significant measures such as accuracy, precision, recall, and f1-score.

1) Intersection-Over-Union (IoU): One of the most used metrics in semantic segmentation is the Intersection-Over-Union (IoU), often known as the Jaccard Index, and with good

reason. The IoU is an exceedingly effective statistic that is relatively simple to use (45).

$$IOU = \frac{Area - Of - Overlap}{Area - Of - Union} \tag{2}$$

The IoU is the region that unites the ground truth and the projected segmentation, divided by the region where the two overlap. On a scale from 0 to 1, where 0 represents no overlap and 1 represents perfectly overlapping segmentation (0-100%), this statistic measures overlap.

By averaging the IoU of each class, the mean IoU of the image can be calculated for binary (two classes) or multi-class segmentation.

2) Dice Loss: To solve the issue of data imbalance, dice loss is frequently utilized in tasks involving segmenting medical images. However, it ignores another imbalance between easy and difficult instances that also negatively impacts a learning model's training process and solely addresses the problem of foreground and background imbalance (52).

The Dice Coefficient is equal to 2 * the Area of Overlap divided by the sum of the pixels in both images.

Due to the extreme scarcity of foreground instances in an image, medical image segmentation networks are forced to exhibit a substantial bias toward the background. The cross-entropy loss function has the aforementioned issue. Dice Loss, which is worded as follows, is suggested as a solution to this problem to balance the foreground and the background (52).

$$DL = 1 - \frac{2\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i^2 + \sum_{i=1}^{N} g_i^2} \tag{3}$$

where $pi \in P$ is defined as the predicted probability of the i-th pixel/voxel. While $g_i \in G$ is the ground truth of the i-th pixel/voxel.

3) Accuracy: The frequency that the classifier predicts correctly is how accuracy is calculated. Accuracy can be defined as the proportion of accurate predictions to all predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

You would assume that a model is working well when it reports an accuracy rate of 99%, but this isn't always the case and in some cases, it might be deceptive. I'll use an example to demonstrate how this works.

Precision is useful when the target class is balanced, but it is not a good choice for unbalanced classes. Imagine a scenario where 99% of the photos in our training data were of the dog but 1% were of the cat. Our model would then predict the dog with a 99.9% accuracy rate. As demonstrated by spam emails, credit card fraud, and inaccurate medical diagnoses, data is actually inherently uneven. To improve model evaluation and get a full view of the model evaluation, additional metrics like recall and precision should also be included(32).

Our data sets don't suffer imbalanced data but there is a slight difference between positive and negative target labels. So, we also consider precision, recall, f1-score, and AUC metrics to get better insight into how our models behave. Also, it may be better if we want to adjust the precision-recall trade-off. Increasing precision will decrease recall and vice versa. The precision/recall trade-off is used to describe this. Positive and negative predictions can be altered by adjusting the threshold value because the classifier behaves differently for various threshold values.

4) Precision: Precision reveals how many of the situations that were predicted with accuracy ended up being positive. When false positives are more problematic than false negatives, precision is helpful. Precision is essential for e-commerce websites, music or video recommendation systems, and other applications where inaccurate results could cause customers to leave, which would be bad for business (4).

The number of true positives divided by the number of predicted positives is the definition of precision for a label.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

5) Recall: Recall measures the proportion of real positive cases that our model correctly predicted. It is a useful indicator when a false negative is more significant than a false positive. In medical conditions, it is essential because even if we raise a false alarm, the genuine positive examples shouldn't go unnoticed (2).

The number of true positives divided by the total number of real positives is how to recall for a label is calculated.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

6) F1 score: The f1 score offers a summary of the Precision and Recall measures. When Precision and Recall are equal, it operates at its best. The F1 Score is the harmonic mean of recall and precision.(33).

$$F1 = 2 * \frac{PrecisionxRecall}{Precision + Recall} \tag{7}$$

More high values are penalized by the F1 score. F1 Score may function as a useful evaluation statistic in the following circumstances:

- When FP and FN are equally costly.
- Adding more data doesn't effectively change the outcome.
- True Negative is high.

IV. RESULTS AND DISCUSSION

A. Introduction

In the last section we have described each model of Unet(34), FPN(24), PSPNet(67), Unet++(68), PAN(57), LinkNet(9), DeepLab-v3(15), DeepLab-v3+(12), and MA-Net(33). Also, we have described each of the evaluation metrics we will depend on to select the best model. We will depend on the IOU (intersection over union) metric, and also we will use the dice loss as an evaluation metric.

In this part, we show the results of each model we have described in the last section. Also, we will show the best-selected model for each data set we used.

After selecting the best models, we will apply more evaluation using different hyperparameters for augmentation and the number of epochs, also we will show some examples from both data sets to get more intuition about how our model performs. We will compute the accuracy, f1-score, precision, recall, IOU, and dice loss as metrics for each model to get a better understanding of how our model performs from different perspectives.

B. Performance Scores of Segmentation Algorithms

Table IV shows the results of the 9 models for the Massachusetts data set. The result obtained from the evaluation of the models on the test set. We apply each model twice, one with the bilateral filter and the other without it. The gray cells of the table show the highest IOU models that get the best results on the test set.

All the results obtained for the Massachusetts data set seem to be close to each other and IOU differences between the models are very small. We can figure out that from the table IV.

The Linknet, Unet, Unet++, and Pan models obtained the highest IOU score they get the same results on the test set. They get an IOU score of 91.32%. All the other models obtain slightly less IOU on the test set. But for model deployment or the subsequent evaluation, we can explore these four models or choose one of them for deployment that depends on the inference time of each model of these four models.

Table V shows the results of the 9 models on the Deepglobe data set, also all the results obtained from the evaluation of the models on the test set. We train each model twice one with a bilateral filter and the other without a filter. The Unit model obtains the highest IOU score metric 95.46%.

The basic concept of transfer learning is to take a model trained on a large data set and transfer its knowledge to another one. Because of Transfer Learning, we could adjust the model parameters and train each model for just 3 epochs and this number was fair enough. Fig. 21 shows the training intersection over union (IOU) metric and fig. 22 shows Dice loss during the training of Unet model on the Deep globe data set which gets the best IOU metric on this data set.

To get more intuition about how Unet model performs on the DeepGlobe data set we show some samples with its ground truth mask and the predicted mask in fig. 25. The figure also shows the one hot encoded mask of the predicted mask. The Unet model seems to be efficient for this data set and could be considered for more evaluation and testing against different hyperparameters like the number of epochs and the augmentation method we will show in the next subsections.

Fig. 23 shows the training intersection over union (IOU) metric and fig. 24 shows Dice loss during the training of the Pan model on the Massachusetts data set. Also, fig. 26 shows some samples from the Massachusetts data set with its ground truth mask and the predicted mask.

In the next subsection, we employ different angles for data augmentation and we also, train the model for more epochs wishing we can get more robust accuracy.

TABLE IV
ALL MODELS RESULTS ON MASSACHUSETTS TEST SET.

Model	IOU	Dice Loss
LinkNET	0.9132	0.0474
LinkNET-bilateral	0.9115	0.0483
Unet	0.9131	0.0474
Unet-bilateral	0.9112	0.0485
UnetPlus	0.9132	0.0474
UnetPlus-bilateral	0.9117	0.0482
deeplabv3	0.9129	0.0475
deeplabv3-bilateral	0.9116	0.0482
deeplabv3plus	0.9125	0.0478
deeplabv3plus-bilateral	0.9115	0.0483
Fpn	0.9114	0.0483
fpn-bilateral	0.9114	0.0483
Manet	0.9127	0.0477
manet-bilateral	0.9119	0.0480
Pan	0.9132	0.0474
pan-bilateral	0.9114	0.0483
Pspnet	0.9131	0.04745
pspnet-bilateral	0.9113	0.0483

TABLE V
ALL MODELS RESULTS ON DEEPGLOBE TEST SET.

Model	IOU	Dice Loss
LinkNET	0.9129	0.04754
LinkNET-bilateral	0.948	0.0321
Unet	0.9546	0.02831
Unet-bilateral	0.9432	0.03477
UnetPlus	0.9496	0.03117
UnetPlus-bilateral	0.9473	0.03231
deeplabv3	0.9518	0.02987
deeplabv3-bilateral	0.9463	0.03307
deeplabv3plus	0.9525	0.02952
deeplabv3plus-bilateral	0.9454	0.03354
Fpn	0.9533	0.02917
fpn-bilateral	0.9454	0.03337
Manet	0.9522	0.02977
manet-bilateral	0.947	0.03253
Pan	0.9516	0.03004
pan-bilateral	0.9472	0.03256
Pspnet	0.9526	0.02953
pspnet-bilateral	0.9486	0.03157

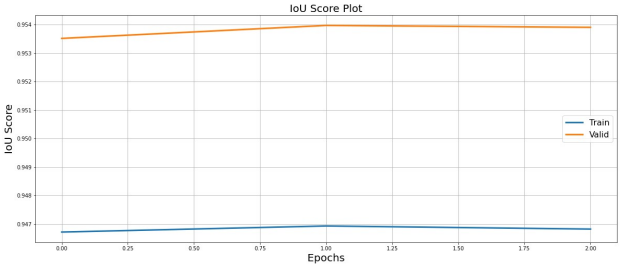

Fig. 21. Training and validation IOU for 3 epochs using Unet model on Deepglobe data set.

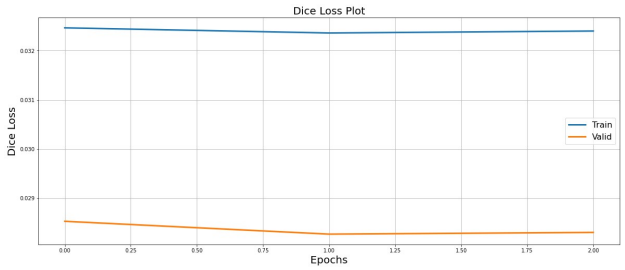

Fig. 22. Training and validation Dice Loss for 3 epochs using Unet model on Deepglobe data set.

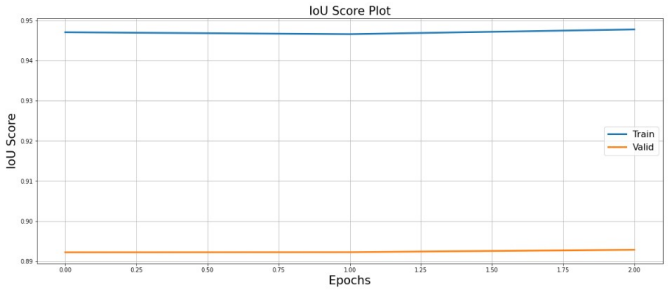

Fig. 23. Training and validation IOU for 3 epochs using Pan model on Deepglobe data set.

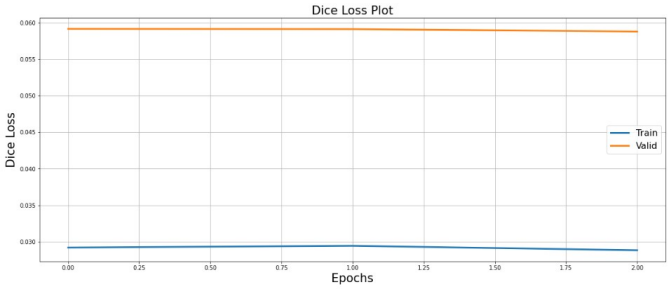

Fig. 24. Training and validation Dice Loss for 3 epochs using Pan model on Massachusetts data set.

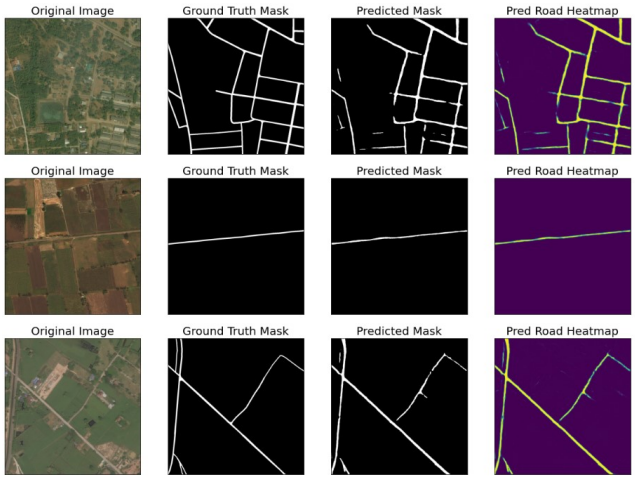

Fig. 25. Some predicted samples from Deepglobe data set using Unet model.

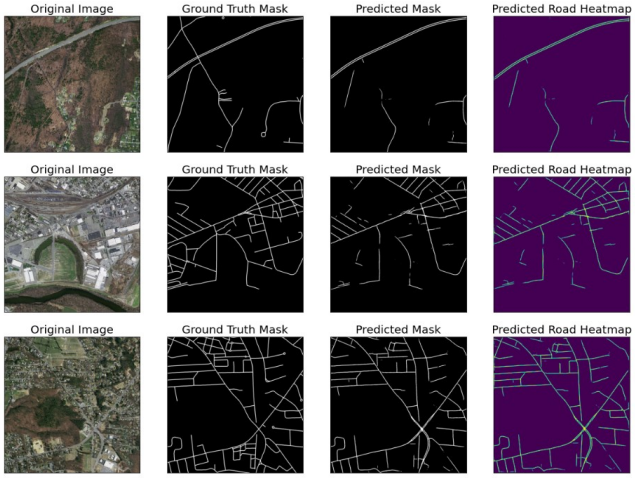

Fig. 26. Some predicted samples from Massachusetts data set using Pan model.

C. Efficiency of Data Augmentation Based On Different Angles

For the deep globe data set, we select the best model which was Unet model, and train it for 5 epochs and for different augmentation angles. The angles of rotation of input images we use it as a hyperparameter to try to understand what affects our training and if the angles of augmentation really matter or can change the result. Also, we will show how the training for more epochs does not affect and the 3 epochs are very enough for training our models.

From the table VI we can figure out that the changing of rotation angle does not affect the performance a lot also, increasing the number of epochs does not help, and the performance is still the same on the test set. By comparing the results obtained above and the results obtained from this step we found that the basic operation of augmentation which are horizontal flip, vertical flip, and the random crop is sufficient for training.

Fig. 27 shows the training and validation intersection over union (IOU) versus the 5 training epochs of the Unet model. The angle range used is [-90, 90] which obtains the best result. Also, fig. 28 shows the training and validation Dice Loss (DL) of the same model training.

TABLE VI
DIFFERENT AUGMENTATION ANGLES USING UNET ON DEEPGLOBE DATA SET.

Angle	IOU score	Dice Loss	Accuracy	Precision	Recall	F1-score
0	0.9498	0.0265	0.9737	0.9731	0.9743	0.9737
$[-30:30]$	0.9507	0.0260	0.9741	0.9736	0.9747	0.9742
$[-60:60]$	0.9489	0.0271	0.9731	0.9726	0.9737	0.9731
$[-90:90]$	0.952	0.0254	0.9748	0.9743	0.9754	0.9748

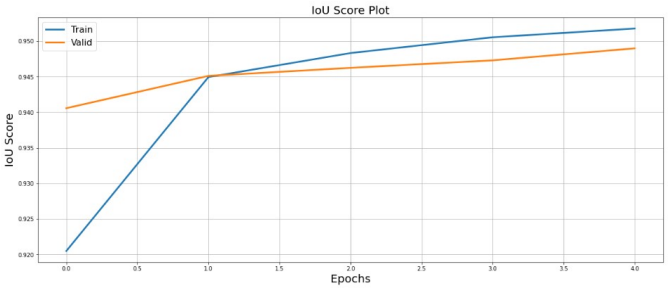

Fig. 27. Training and validation IOU for 5 epochs using Unet model on Deepglobe data set and with [-90:90] rotation angles.

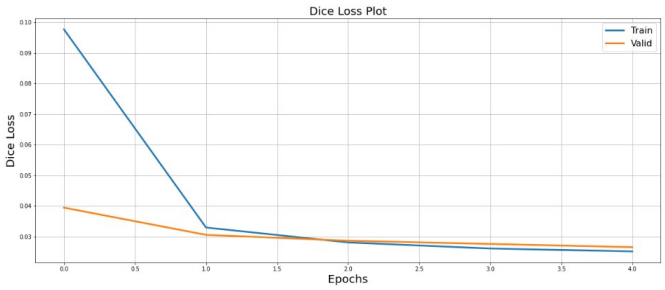

Fig. 28. Training and validation Dice Loss for 5 epochs using Unet model on Deepglobe data set and with [-90:90] rotation angles.

D. Robustness of the proposed Segmentation Approach

We use 10 different Encoder-Decoder-based models, so we can get a more robust model for our problem we could achieve a comparable accuracy with other researchers, by applying all state-of-the-art models of segmentation algorithms that be a proof of concept for future work to build robust encoder-decoder based on the problem of road segmentation from a satellite image. Our chosen models are robust and achieve great accuracy compared to other segmentation algorithms Tables VII and VIII give a comparison between our best results and other state-of-the-art results on both data sets Massachusetts and DeepGlobe respectively.

The segmentation algorithms are improved quickly over the last 8 years and there are a lot of shelf models that can be used, our choices could be a great baseline for other researchers to not consume time to try different segmentation algorithms.

Figure 29 shows a test example of the DeepGlobe dataset predicted using the 9 different segmentation models tested. Figure 30 shows a test example of the Massachusetts dataset predicted using the 9 different segmentation models tested. These figures provide visual representations of the performance of the different models in predicting road segmentation from satellite images.

TABLE VII
COMPARISON OF RESULTS WITH OTHER RELATED WORK ON MASSACHUSETTS DATA SET

Model	IOU	F1-score
FCN(50)	0.8197	0.9009
Segnet(6)	0.7983	0.8873
Deep ResUnet(12)	0.8365	0.9102
DeepLabv3+(56)	0.8442	0.9196
SII-Net(55)	0.8521	0.9276
VNet_CEDL(1)	0.8382	0.9118
GAN+MUNet(2)	87.43	92.20
RDRCNN (23)	67.10	80.31
LinkNET	0.9132	0.9398
UnetPlus	0.9132	0.9397
Pan	0.9132	0.9397
Unet	0.9131	0.9396

TABLE VIII
COMPARISON OF RESULTS WITH OTHER RELATED WORK ON DEEPGLOBE DATA SET

Model	IOU	F1-score
RoadDA(61)	0.8535	0.9235
D-LinkNet50(44)	0.6112	0.6992
ATD-LinkNet50(44)	0.6268	0.7023
DeepLab(20)	0.5930	–
ResInceptionSkip(20)	0.6120	–
Unet	0.9546	0.9768

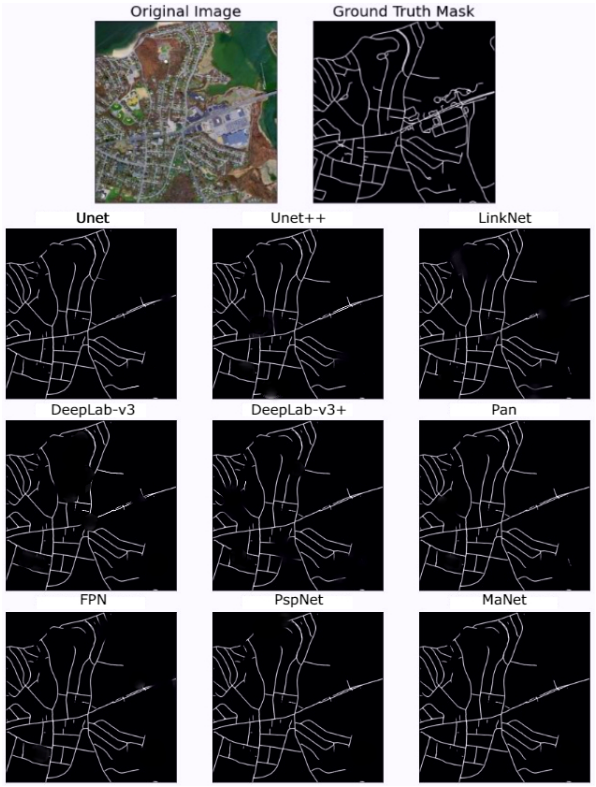

Fig. 29. A test example of the DeepGlobe dataset predicted using the 9 different segmentation models we tested.

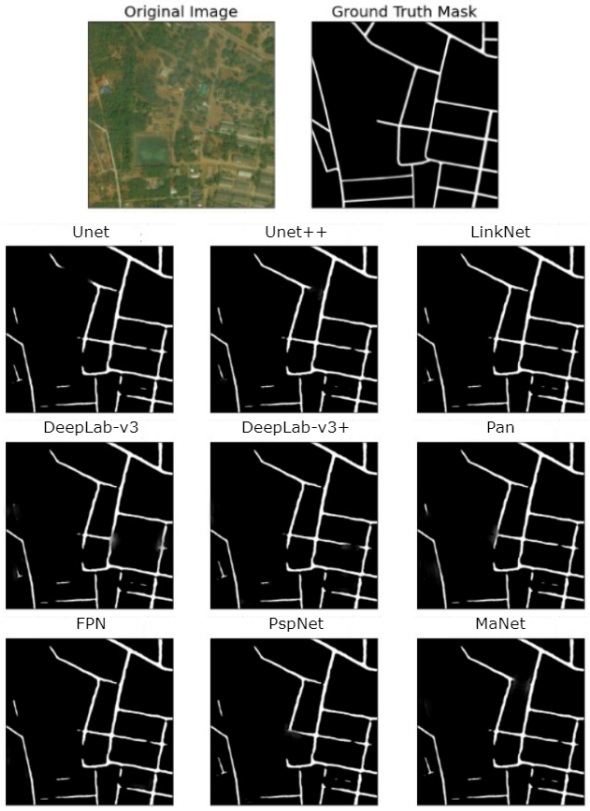

Fig. 30. A test example of Massachusetts dataset predicted using the 9 different segmentation models we tested.

E. Conclusion

In this part, we show the results of each model we use with and without the bilateral filter, and we show the results obtained for all models on the Massachusetts data set to obtain a close accuracy for each other.

We also, apply more data augmentation parameters to improve the performance of the models, but the results are the same for this data set. The Unit model obtained the best accuracy for the Deepglobe data set with an IOU score of 95.46%. Our chosen models are robust and achieve great accuracy compared to other segmentation algorithms. By comparing the results with other researchers' work on these data sets the Encoder-Decoder approach proves an excellent impact on the model performance the results of this approach could be considered the best model architecture for solving the Road Segmentation problem.

V. CONCLUSION AND FURTHER DIRECTION

A. Outline of the Contributions

In this part, we discuss and outline our work from different perspectives. We discuss what motivates us in the first place of increasing the need for many civilian and military applications. The extraction of map elements like roads, rivers, and buildings from high-resolution satellite data is a crucial undertaking problem. Cartography makes heavy use of remote sensing.

We also discuss the limitation of our work and how the backbone model can affect the results and could be considered for future work, and we give an overall conclusion about all we discussed from the literature review part and proposed approach and the best results we obtained. The main contribution we offer in this thesis:

- First contribution, we employ 9 different models of current model architectures that use the DCEP network as their primary base model on two open-source data sets from DeepGlope and Massachusetts. The models we employ are, Unet, FPN, PSPNet, Unet++, PAN, LinkNet, DeepLabv3, DeepLabv3+, and MA-Net.
- Second contribution, we apply different data augmentation hyperparameters for angle rotation degrees we try [0:0], [-30:30], [-60,60], and [-90, 90], To make sure we pick up the most appropriate hyperparameters for all other models.
- Third contribution, we evaluate all the models using Intersection Over Union (IOU) metric, and dice loss (DL) and give a brief comparison between each one of them and what is the most suitable model for each data set. we prove that Encoder-Decoder segmentation algorithms can achieve acceptable results in the field of road segmentation.

B. Limitations

The limitation of the work is the backbone model our work considers a proof of concept for the performance of deep learning specifically the Encoder-Decoder model architecture. So, we use the mobilenetv3 (27) as the backbone for all the segmentation models we use to give a brief comparison between all these models based on the same backbone architecture.

The backbone model is the model used to construct the encoder part of the segmentation architecture and used to encode the image into the latent space that is used later for the up-sampling part of the decoder.

Other backbone models could be considered to solve this limitation an example of these backbones are VGGNet(50), ResNet (26), and Inception-v3 (54) and there is always more and more backbone models could be considered.

Our work could be considered a starting point for other researchers to catch up with the best Encoder-Decoder architecture and start to try different backbone models for the encoder model. Also, consider transfer learning by using these backbones models trained on the ImageNet data set.

C. Overall Conclusion

In this thesis, we discussed crucial questions like how can image segmentation performance be enhanced by deep learning. What is the ideal loss function for segmenting images? What deep learning architectures work best for this issue? All these inquiries prompt us to compare various results on the SAR image segmentation issue from various research articles. Others utilize GAN, while still others use encoder-decoder designs and FCN (Generative Adversarial Neural network). The loss function was also a crucial decision. Some studies combine the two loss functions, giving weight to each one

of them. Some articles utilize the cross-entropy loss function while others use the dice loss.

By studying other researchers' work we choose one of the most effective recently created neural networks, the deep convolutional encoder-decoder (DCED) architecture, which has been suggested for object segmentation. The DCED network is intended to serve as the primary segmentation architecture for pixel-wise semantic segmentation. and has proved excellent performance in the experiments tested using PASCAL VOC 2012 data a well-known benchmark data set for image segmentation research (6; 31; 38).

we apply a variety of recent model architectures that use the Encoder-Decoder network as a primary base model on two open source data sets DeepGlope (19), and Massachusetts (40). We choose the most common encoder-decoder models that proved great performance for different data sets of image segmentation. We choose Unet, fpn, PSPNet, UnetPlus, Pan, LinkNet, deeplabv3, deeplab-v3+, and MA-Net.

Also, we describe each model architecture we used in detail and show the effect of the encoder-decoder architecture to increase the performance of the image segmentation problem.

We demonstrate the outcomes for each model we employ both with and without the bilateral filter, and we demonstrate how the outcomes for all models on the Massachusetts data set are highly comparable in terms of accuracy. We also apply additional data augmentation parameters in an effort to enhance model performance, but the outcomes are the same for this data set. With an IOU score of 95.46%, the Unet model has the highest IOU for the Deepglobe data set.

The Encoder-Decoder technique has a significant impact on the model performance when compared to other researchers' work on these data sets, and the results suggest that this is the best model architecture for the Road Segmentation problem.

REFERENCES

[1] Abolfazl Abdollahi, Biswajeet Pradhan, and Abdullah Alamri. Vnet: An end-to-end fully convolutional neural network for road extraction from high-resolution remote sensing data. *IEEE Access*, 8:179424–179436, 2020.

[2] Abolfazl Abdollahi, Biswajeet Pradhan, Gaurav Sharma, Khairul Nizam Abdul Maulud, and Abdullah Alamri. Improving road semantic segmentation using generative adversarial network. *IEEE Access*, 9:64381–64392, 2021.

[3] Preeti Aggarwal, Renu Vig, Sonali Bhadoria, and CG Dethe. Role of segmentation in medical imaging: A comparative study. *International Journal of Computer Applications*, 29(1):54–61, 2011.

[4] Rasha Alshehhi and Prashanth Reddy Marpu. Hierarchical graph-based segmentation for extracting road networks from high-resolution satellite images. *ISPRS journal of photogrammetry and remote sensing*, 126:245–260, 2017.

[5] Reza Azad, Maryam Asadi-Aghbolaghi, Mahmood Fathy, and Sergio Escalera. Attention deeplabv3+: Multi-level context attention mechanism for skin lesion segmentation. In *European conference on computer vision*, pages 251–266. Springer, 2020.

[6] Vijay Badrinarayanan, Ankur Handa, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *arXiv preprint arXiv:1505.07293*, 2015.

[7] Hmrishav Bandyopadhyay. Image segmentation: Deep learning vs traditional. available link https://www.v7labs.com/blog/image-segmentation-guide, 2022.

[8] Weiling Cai, Songcan Chen, and Daoqiang Zhang. Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation. *Pattern recognition*, 40(3):825–838, 2007.

[9] Abhishek Chaurasia and Eugenio Culurciello. Linknet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE, 2017.

[10] Abhishek Chaurasia and Eugenio Culurciello. Linknet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4, 2017.

[11] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[12] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 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, 2017.

[13] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[14] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016.

[15] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[16] Gong Cheng and Junwei Han. A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote sensing*, 117:11–28, 2016.

[17] Guangliang Cheng, Ying Wang, Shibiao Xu, Hongzhen Wang, Shiming Xiang, and Chunhong Pan. Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 55(6):3322–3337, 2017.

[18] Sukhendu Das, TT Mirnalinee, and Koshy Varghese. Use of salient features for the design of a multistage framework to extract roads from high-resolution multispectral satellite images. *IEEE transactions on Geoscience and Remote sensing*, 49(10):3906–3931, 2011.

[19] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[20] Jigar Doshi. Residual inception skip network for binary segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[21] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. prentice hall professional technical reference, 2002.

[22] Prachi Gadpayleand and PS Mahajani. Detection and classification of brain tumor in mri images. *International Journal of Emerging Trends in Electrical and Electronics, IJETEE–ISSN*, pages 2320–9569, 2013.

[23] Lin Gao, Weidong Song, Jiguang Dai, and Yang Chen. Road extraction from high-resolution remote sensing imagery using refined deep residual convolutional neural network. *Remote sensing*, 11(5):552, 2019.

[24] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7036–7045, 2019.

[25] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[27] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.

[28] Xin Huang and Liangpei Zhang. Road centreline extraction from high-resolution imagery based on multiscale structural features and support vector machines. *International Journal of Remote Sensing*, 30(8):1977–1987, 2009.

[29] Andres Huertas and Ramakant Nevatia. Detecting buildings in aerial images. *Computer vision, graphics, and image processing*, 41(2):131–152, 1988.

[30] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017.

[31] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015.

[32] Jake Lever. Classification evaluation: It is important to understand both what a classification metric expresses and what it hides. *Nature methods*, 13(8):603–605, 2016.

[33] Rui Li, Shunyi Zheng, Ce Zhang, Chenxi Duan, Jianlin Su, Libo Wang, and Peter M. Atkinson. Multiattention network for semantic segmentation of fine-resolution remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2022.

[34] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE transactions on medical imaging*, 37(12):2663–2674, 2018.

[35] Jingyun Liang, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Mutual affine network for spatially variant kernel estimation in blind image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4096–4105, 2021.

[36] Bo Liu, Huayi Wu, Yandong Wang, and Wenming Liu. Main road extraction from zy-3 grayscale imagery based on directional mathematical morphology and vgi prior knowledge in urban areas. *PloS one*, 10(9):e0138071, 2015.

[37] Wei Liu, Andrew Rabinovich, and Alexander C Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015.

[38] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen Change Loy, and Xiaoou Tang. Deep learning markov random field for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1814–1828, 2017.

[39] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29, 2016.

[40] Volodymyr Mnih. *Machine Learning for Aerial Image Labeling*. PhD thesis, University of Toronto, 2013.

[41] Volodymyr Mnih and Geoffrey E Hinton. Learning to detect roads in high-resolution aerial images. In *European conference on computer vision*, pages 210–223. Springer, 2010.

[42] Volodymyr Mnih and Geoffrey E Hinton. Learning to label aerial images from noisy data. In *Proceedings of the 29th International conference on machine learning (ICML-12)*, pages 567–574, 2012.

[43] Charalambos Poullis. Tensor-cuts: A simultaneous multi-type feature extractor and classifier and its application to road extraction from satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 95:93–108, 2014.

[44] Xingqun Qi, Kaiqi Li, Pengkun Liu, Xiaoguang Zhou, and Muyi Sun. Deep attention and multi-scale networks for accurate remote sensing image segmentation. *IEEE Access*, 8:146627–146639, 2020.

[45] Md Atiqur Rahman and Yang Wang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International symposium on visual computing*, pages 234–244. Springer, 2016.

[46] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[47] Christian J Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1439–1451, 2015.

[48] Qian Shi, Xiaoping Liu, and Xia Li. Road detection from remote sensing images by generative adversarial networks. *IEEE access*, 6:25486–25494, 2017.

[49] Wenzhong Shi, Zelang Miao, and Johan Debayle. An integrated method for urban main-road centerline extraction from optical remotely sensed imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 52(6):3359–3372, 2013.

[50] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[51] Mingjun Song and Daniel Civco. Road extraction using svm and image segmentation. *Photogrammetric Engineering & Remote Sensing*, 70(12):1365–1371, 2004.

[52] Toufique A Soomro, Ahmed J Afifi, Junbin Gao, Olaf Hellwich, Manoranjan Paul, and Lihong Zheng. Strided u-net model: Retinal vessels segmentation using dice loss. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE, 2018.

[53] Chinnathevar Sujatha and Dharmar Selvathi. Connected component-based technique for automatic extraction of road centerline in high resolution satellite images. *EURASIP Journal on Image and Video Processing*, 2015(1):1–16, 2015.

[54] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[55] Chao Tao, Ji Qi, Yansheng Li, Hao Wang, and Haifeng Li. Spatial information inference net: Road extraction using road-specific contextual information. *ISPRS Journal of Photogrammetry and Remote Sensing*, 158:155–166, 2019.

[56] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 1451–1460. Ieee, 2018.

[57] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8440–8449, 2019.

[58] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1568–1576, 2017.

[59] Chuan Yang and Zhenghong Wang. An ensemble wasserstein generative adversarial network method for road extraction from high-resolution remote sensing images in rural areas. *IEEE Access*, 8:174317–174324, 2020.

[60] Jing Zhang, Lu Chen, Chao Wang, Li Zhuo, Qi Tian, and Xi Liang. Road recognition from remote sensing imagery using incremental learning. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):2993–3005, 2017.

[61] Lefei Zhang, Meng Lan, Jing Zhang, and Dacheng Tao. Stagewise unsupervised domain adaptation with adversarial self-training for road segmentation of remote-sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2022.

[62] Xiangrong Zhang, Xiao Han, Chen Li, Xu Tang, Huiyu Zhou, and Licheng Jiao. Aerial image road extraction based on an improved generative adversarial network. *Remote Sensing*, 11(8):930, 2019.

[63] Yang Zhang, Xiang Li, and Qianyu Zhang. Road topology refinement via a multi-conditional generative adversarial network. *Sensors*, 19(5):1162, 2019.

[64] Yang Zhang, Zhangyue Xiong, Yu Zang, Cheng Wang, Jonathan Li, and Xiang Li. Topology-aware road network extraction via multi-supervised generative adversarial networks. *Remote Sensing*, 11(9):1017, 2019.

[65] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018.

[66] Zhengxin Zhang, Yunhong Wang, Qinjie Liu, Lingling Li, and Ping Wang. A cnn based functional zone classification method for aerial images. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5449–5452. IEEE, 2016.

[67] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[68] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 3–11. Springer, 2018.

[69] C Zhu, W Shi, M Pesaresi, L Liu, X Chen**, and B King. The recognition of road network from high-resolution satellite remotely sensed data using image morphological characteristics. *International Journal of Remote Sensing*, 26(24):5493–5508, 2005.