*International Journal of Computational and Experimental Science and ENgineering (IJCESEN)*

*Vol. 11-No.2 (2025) pp. 1887-1900*
*http://www.ijcesen.com*

***ISSN: 2149-9144***



**Research Article**

# UMV2: Deep Learning Model for Semantic Segmentation of Satellite Imagery

**Babitha Lokula[1], P.V.V. Kishore[2], L. V. Narasimha Prasad[3]**

[1]Department of ECE, KLEF Educational Foundation, Vaddeshwaram
* **Corresponding Author Email:** babithakodishala@gmail.com- **ORCID:** 0000-0003-0841-2395

[2]Department of ECE, KLEF Educational Foundation, Vaddeshwaram
**Email:** pvvkishore@gmail.com - **ORCID:** 0000-0002-3247-3043

[3]Department of ECE, Institute of Aeronautical Engineering
**Email:** lvnprasad@iare.ac.in- **ORCID:** 0000-0001-6514-1064

**Article Info:**

**DOI:** 10.22399/ijcesen.1362
**Received :** 05 January 2025
**Accepted :** 10 March 2025

**Keywords** (must be 3-5)

Satellite image segmentation,
Unet++,
MobileNetV2 encoder,
deep learning model,
land cover classification,
environmental monitoring.

**Abstract:**

Semantic segmentation is a computer vision task that categorizes each pixel in an image into a class or object. Although a number of relevant architectures have been proposed in recent years, they incur the problems like computational cost,large amounts of training data, class imbalance, edge uncertainty, varying sizes of objects, shadow and lighting variations. Such a more number of drawbacks degrades the semantic segmentation performance in terms of accuracy, efficiency and generalization capability. In this work, comprehensive architecture UMV2 for satellite image semantic segmentation is proposed. The UMV2 utilizing a fusion of Unet++ architecture and the lightweight MobileNetV2 encoder deep learning model. The Unet++ architecture, an extension of the widely adopted Unet, is employed for its ability to capture hierarchical features and enhance segmentation performance. Integrating MobileNetV2 as the encoder provides computational efficiency, making the model well-suited for resource-constrained environments, such as satellite image analysis on edge devices. The proposed model leverages the strengths of both architectures, combining the expressive power of Unet++ with the efficiency of MobileNetV2. Extensive experiments are conducted on a diverse satellite image dataset, evaluating the model's segmentation accuracy of 0.89, mean IOU of 0.52, precision of 0.80, recall of 0.83 and F1-score of 0.82 with the state of art methods. The results demonstrate the effectiveness of the proposed approach in achieving accurate and efficient satellite image segmentation, making it a promising solution for real-world applications in remote sensing and geospatial analysis.

## 1. Introduction

Image segmentation is the computer vision task, it is the process of dividing a digital image into multiple image segments, also known as objects or regions. Image segmentation can be performed with conventional segmentation techniques and various deep learning models. Conventional image segmentation methods like thresholding, edge based, region based, clustering and graph based methods requires manual intervention and suffers with sensitive to light conditions, fragmented edges, difficulty with complex boundaries, parameter selection and scalability respectively. On the other hand, in the last few years image segmentation achieved tremendous performance using deep learning techniques like semantic segmentation, instance segmentation and panoptic segmentation. Semantic segmentation is a computer vision task in which the goal is to categorize each pixel in an image into a class or object. The goal is to produce a dense pixel-wise segmentation map of an image, where each pixel is assigned to a specific class or object. Semantic segmentation of satellite imagery is useful in remote sensing[1,2], agriculture[3,4], medical field[5], autonomous vehicles[6], robotics[7,8], video surveillance[9,10] as well as quality control.Even though various approaches have been proposed for semantic segmentation, they are still facing the problems like high computational cost, requirement of large amount of data, class imbalance, boundary delineation, dependence on annotation quality. Satellite imaging is useful in many different fields, including environmental

monitoring, agriculture, climate change, urban planning, national security and surveillance, risk management, disaster monitoring, traffic signal analysis and environmental studies, to name a few. It is very necessary to be able to derive relevant information from satellite photos in order to effectively make decisions, manage resources, and respond to natural disasters [11]. The process of segmentation, which includes identifying and classifying all of the objects and land cover elements included within the picture, is one of the most important steps in satellite image processing. Significant problems for accurate and effective segmentation are presented by the complexity of satellite imagery, which is characterized by huge geographical extents, varied resolutions, and diverse landscapes[12]. Traditional approaches often fail to keep up with all of these different obstacles, which results in solutions that are not ideal. In addition, the human interpretation of training data in supervised learning is not only labor-intensive but also time-consuming, which limits the scalability of these approaches[13]. As a consequence of this, there is an urgent need for robust and automated segmentation algorithms that are capable of adjusting to the varied and ever-changing characteristics of satellite data. Even though deep learning has made great strides and is more successful than previous methods, satellite image segmentation using deep learning models still has certain limitations.The enormous amount of processing power that deep learning algorithms frequently demand is one of the most prominent disadvantages[14]. These models' intricate designs need a large amount of processing power and memory, which makes them computationally demanding and sometimes unsuitable for applications that have restricted resources. Another disadvantage is that it requires a substantial quantity of labeled training data to function properly. To train a deep learning model successfully large datasets are required. Obtaining and annotating such information may be a process that is both time-consuming and resource-intensive. This is especially true for satellite images that include a variety of different landscapes and features[15]. This dependence on large amounts of labeled data might provide difficulties in circumstances in which acquiring such data is difficult from a budgetary or logistical perspective.

The most recent investigations into the use of deep learning models for image segmentation in satellite imagery are at an advanced stage of developments in remote sensing and earth observation[16]. The complex and ever-changing nature of satellite imagery presents a number of issues, which are now being actively addressed by researchers by exploring creative ways[17]. There is work being carried out to improve previously developed deep learning architectures and to create new models that are specifically suited for the segmentation of satellite pictures. The optimization of computational efficiency to manage the huge volumes of data associated with high-resolution satellite imagery[18] is one of the primary focuses of this research. This optimization is done to ensure that these models can function well in real-time or near-real-time applications. In this research, the difficult problem of segmenting satellite images is tackled, and a unique method based on self-supervised learning and semantic segmentation is presented as a potential solution. The process of segmenting satellite images lends itself especially well to the use of self-supervised learning, which is a promising method in the area of computer vision. Self-supervised learning is a method of machine learning that enables a model to acquire meaningful representations without the need for vast human-labeled datasets. This method works by exploiting the innate patterns and connections that are present within satellite images. Because of this modification in the approach of learning, the model is now able to recognize and differentiate between objects and aspects of land cover even when there is no pre-labeled training data available. Furthermore, the method goes beyond standard segmentation methods by including semantic segmentation techniques. This enables the detection and distinction of individual instances of the same object class included inside the picture. This is very helpful in situations in which accurate and comprehensive segmentation is of the utmost importance, such as when one is tasked with counting trees in a forest or monitoring automobiles in a parking lot.

In this research, a novel Supervised Learning Based Semantic Segmentation Method is presented. It is designed to particularly handle the one-of-a-kind difficulties that are inherent in the processing of satellite images. By using the full potential of supervised learning, this approach is able to make efficient use of the vast amounts of information that are included inside satellite photos. Because of its decreased dependence on manually annotated training data, this strategy mitigates a constraint that is often experienced when using more standard supervised approaches, which is one of the methodology's significant advantages.

The contributions of this paper can be summarised as:

An efficient UMV2 model is proposed that integrates MobileNetV2 as the encoder in Unet++ architecture, to address the accuracy problem, computational efficiency and generalization capability.

The Unet++ architecture and MobileNetV2 architecture are explored to capture the features such as edges, textures, and other relevant information that are crucial for distinguishing different segments in complex and high resolution satellite images.

To test the performance of proposed model, we evaluated our model over the traditional models using various performance metrics. The results show the strength of the demonstrated model against other baseline models.

The image patching technique is considered to get all image patches of size 256X256 for efficient way of training and testing processes of dataset.

The remainder of this paper is organised as follows. Section 2 reviews the related work of various authors. Section 3 details our proposed architecture. Section 4 illustrates the methodology includes data preprocessing, environmental setup, training and testing process and also evaluation metrics. Section 5 details the qualitative and quantitative comparative analysis of experimental results with the various state of the art techniques. Section 6 provides the conclusion of this work.

## 2. Related Works

Tareque Bashar Ovi et al[19] introduced a novel tri-level attention-based DeepLabv3+ architecture, referred to as DeepTriNet, for the purpose of semantic segmentation of satellite images. The hybrid technique under consideration integrates squeeze-and-excitation networks (SENets) and tri-level attention units (TAUs) into the existing DeepLabv3+ architecture. The TAUs are used to address the semantic feature disparity between the output of encoders, while the SENets are utilized to assign more importance to pertinent features. The DeepTriNet model, as presented, determines the most significant characteristics in a more generic manner via self-supervision, rather than relying on manual annotation.

Yann Fabel et al[20] presented a novel approach of self-supervised learning to effectively use a much bigger dataset in comparison to traditional supervised training methods, resulting in improved performance of the model. The first stage of the study entails using over 300,000 Atmospheric State Indices (ASIs) in two separate pretext assignments as part of the pretraining process. One of the objectives focuses on the process of reconstructing images, while the other job is concentrated on the utilization of the DeepCluster model. The DeepCluster model is an iterative procedure that includes grouping and categorizing the neural network's output. Following that, the model is subjected to a process of fine-tuning using a rather modest labeled dataset consisting of 770 ASIs. Out of these, 616 ASIs are allocated for training purposes, while the remaining 154 ASIs are reserved for validation. Every Artificial Intelligence System (ASI) is linked to a ground truth mask that classifies individual pixels into several categories, such as clear sky, low-layer clouds, mid-layer clouds, or high-layer clouds. In order to evaluate the efficacy of self-supervised pretraining, a comparison study is undertaken, whereby this methodology is contrasted with models that are started with random weights and those that are pretrained using ImageNet data. All models are trained and validated using identical datasets.

Fabien H.Wagner et al[21] presented the k-textures technique, which offers a self-supervised approach for segmenting a 4-band picture (consisting of RGB and NIR bands) into k distinct classes. An example of its use using high-resolution Planet satellite images is shown. According to the algorithmic analysis, it has been determined that the use of convolutional neural networks (CNN) in conjunction with gradient descent renders discrete search a viable approach. The model is capable of identifying k distinct clustering classes within the data. These classes are represented by k discrete binary masks and their corresponding separately produced textures. When merged, these masks and textures simulate the initial picture. The similarity loss refers to the average squared error between the features of the actual picture and the simulated image. These features are obtained from the penultimate convolutional block of two different models: The Keras "imagenet" pre-trained VGG-16 model and a custom feature extractor created using Planet data. The primary advancements of the k-textures model include the acquisition of k discrete binary masks inside the model via the use of gradient descent. The proposed model facilitates the production of discrete binary masks via the use of a unique approach including a hard sigmoid activation function. Furthermore, the algorithm offers hard clustering classes, where each pixel is assigned to a single class. In contrast to the k-means algorithm, which treats each pixel as an independent entity, the approach discussed here incorporates contextual information and associates each class not just with comparable color channel values, but also with texture. The proposed methodology aims to facilitate the generation of training samples for satellite image segmentation. Additionally, the k-textures architecture may be modified to accommodate varying numbers of bands and to address more intricate self-segmentation problems, such as object self-segmentation.

Wadii Boulila et al[22] introduced a hybrid strategy for object categorization in very-high-resolution

satellite images, using the PPDL framework. The encryption technique under consideration involves the integration of Paillier homomorphic encryption (PHE) and slightly homomorphic encryption (SHE). The objective of this combination is to augment the encryption of satellite images while simultaneously maintaining optimal runtime and achieving a high level of accuracy in object categorization. The encryption technique used for pictures is supported by the utilization of the public keys associated with Partially Homomorphic Encryption (PHE) and Somewhat Homomorphic Encryption (SHE). The researchers performed experiments utilizing high-resolution satellite images obtained from the SPOT6 and SPOT7 satellites in real-world scenarios. This study examined four distinct convolutional neural network (CNN) architectures, namely ResNet50, InceptionV3, DenseNet169, and MobileNetV2.

Wenyuan Li et al[23] proposed a self-supervised multitask methodology for acquiring representations in remote sensing images that effectively captures visual aspects. The proposed approach entails the development of three separate pretext tasks and the use of a triplet Siamese network to simultaneously capture both high-level and low-level visual features. The training process of this network does not need the use of labeled data. However, the resulting model may be further refined by the use of annotated segmentation datasets during the fine-tuning phase. The efficacy of their methodology is validated by empirical investigations carried out on several datasets, including Potsdam, Vaihingen, and the Levir_CS dataset, which focuses on cloud and snow identification. The trial's results demonstrate that the suggested approach effectively lowers the reliance on labeled datasets and improves the performance of remote sensing semantic segmentation. When comparing their method to recent state-of-the-art self-supervised representation learning methods and commonly employed initialization methods such as random initialization and ImageNet pretraining, it is observed that their method consistently outperforms the others in the majority of experiments, particularly in situations where there is a scarcity of training data. Surprisingly, their strategy demonstrates equivalent performance to randomly initialized models with a little 10 to 50 labeled data.

Haifeng Li et al [24] presented a new network called the Global Style and Local Matching Contrastive Learning Network (GLCNet) for the task of semantic segmentation in Remote Sensing Images (RSIs). The GLCNet has been designed with a unique structure to improve the segmentation of Remote Sensing Images (RSIs). During the first stage, the use of the Global Style Contrastive Learning module is implemented to enhance the process of acquiring image-level representations. This premise is based on the notion that stylistic attributes have the capacity to accurately encapsulate the holistic qualities of a picture. The subsequent module, known as the Local Features Matching Contrastive Learning module, has been carefully developed to acquire representations of local areas, which play a critical role in semantic segmentation tasks. The authors of the study conducted a thorough evaluation of their technique by using four separate datasets for RSI semantic segmentation. The experimental findings repeatedly demonstrate that their approach significantly outperforms both contemporary self-supervised approaches and the ImageNet pretraining method in terms of performance.

Wenbo Sun et al [25] proposed a novel approach aimed at enhancing the accuracy of picture segmentation by including depth estimation techniques into the analysis of RGB images. Subsequently, the obtained depth map is utilized as input for a convolutional neural network (CNN) to facilitate the process of semantic segmentation. Moreover, for the purpose of concurrently parsing the depth map and RGB pictures, An encoder-decoder network with several branches is designed, and the RGB and depth characteristics are progressively integrated. The results of the extensive experimental assessment on four baseline networks indicate that the suggested technique significantly improves the quality of segmentation and achieves superior performance when compared to other segmentation networks.

Jannik Zurn et al [26] proposed a novel framework for terrain categorization that leverages an unsupervised proprioceptive classifier. The classifier in question acquires knowledge from the auditory signals generated during the interactions between vehicles and the terrain. This allows for the autonomous training of a classifier that can perform pixel-wise semantic segmentation of pictures, based on external sensory information. The methodology initiates by creating a discriminative embedding space for the sounds produced during vehicle-terrain interaction. This is achieved by using triplets of audio clips, which are constructed by combining the visual attributes of the respective terrain patches. The produced embeddings are further subjected to clustering, whereby these clusters are used as labels for the visual terrain patches. The assignment of these labels is accomplished by projecting the pathways walked by the robot onto the camera pictures. The use of poorly labeled pictures for training the semantic segmentation network is achieved by the application of weak supervision. The study provides a thorough collection of quantitative and qualitative results, illustrating the

superiority of their proprioceptive terrain classifier over current unsupervised approaches. Furthermore, the self-supervised exteroceptive semantic segmentation model developed by the researchers demonstrates performance levels that are equivalent to those reached by supervised learning using manually annotated data.

Huihui Dong et al[27] proposed an innovative approach to self-supervised representation learning for remote sensing picture change detection. This technique is centered on temporal prediction. The primary objective is to enhance the consistency of feature representations in two satellite pictures via a self-supervised process, without relying on semantic supervision or requiring extra computations. By using the modified feature representations, it is possible to produce an improved difference image (DI) that effectively minimizes the error transmitted by the DI in the end result of detection. In the framework of self-supervision, the neural network is tasked with discerning distinct sample patches within a pair of temporal pictures, hence engaging in temporal prediction. By using a network architecture that emulates the discriminator component of generative adversarial networks, the temporal prediction task is able to capture distribution$-$aware feature representations, leading to a resultant model that exhibits strong resilience.

## 3. Proposed Unet++ model

The Unet++ framework is an expansion of the Unet design, characterized by the inclusion of an encoder-decoder structure that incorporates skip connections. The process of feature extraction is performed by the encoder on the input image, while the decoder utilizes these extracted features to build the segmentation mask. The Unet++ architecture incorporates nested skip connections to enhance the transmission of information between the encoder and decoder components. This methodology facilitates the comprehensive capture of both low-level and high-level features, hence enhancing the quality of segmentation outcomes. Figure 1 shows architecture of Unet++ with integration of mobileNetV2 back bone. The encoder component of the Unet++ design has the potential to be substituted with a pre-trained MobileNetV2 backbone. This enables the model to use the efficiency of MobileNetV2 in terms of its computational speed and resource utilization. The MobileNetV2 encoder is responsible for extracting hierarchical features from the input image, while the Unet++ decoder further refines these features in order to generate the final segmentation mask. The inclusion of skip links between the encoder and decoder components of the model facilitates acquisition of both low-level and high-level information, hence enhancing the performance of segmentation. Figure 2 shows the Internal architecture of proposed Unet++ model with MobileNetV2.

MobileNetV2 is used as an encoder for Unet++ in the proposed model. MobileNetV2 is a convolutional neural network (CNN) architecture specifically developed to cater to the computational constraints of mobile and edge devices. The approach employs depthwise separable convolutions and linear bottlenecks to effectively decrease the quantity of parameters and computational burden, while still achieving satisfactory performance. MobileNetV2 is renowned for its efficacy in terms of velocity and resource consumption, rendering it well-suited for real-time applications on devices with constrained processing capabilities.

The model is trained using a dataset that contains segmentation masks that have been labeled. During the training process, the model acquires the ability to establish a mapping between input images and their related segmentation masks. The commonly employed loss functions for semantic segmentation encompass loss of cross-entropy, which calculates the difference between the ground truth and the prediction pixel-wise labels.

### A. MobileNetV2 Encoder

Convolutional layers, batch normalisation, ReLU activation, and other components make up the encoder and a series of 17 Inverted Residual (IR) blocks, followed by additional convolutional layers. Figure 3 shows the MobileNetV2 architecture.The encoder begins with a convolutional layer, then came batch normalization and Rectified Linear Unit (ReLU) activation. The initial module is accountable for the processing of the raw input image and producing a collection of feature maps.

The core component of the MobileNetV2 architecture is the Inverted Residual block. In the proposed model, there are a total of 17 blocks that have been arranged in a sequential manner. The Inverted Residual block effectively retains and enhances the features extracted from the preceding layer. The architecture is comprised of depth-wise separable convolutions, linear bottlenecks, and skip connections.

After the series of Inverted Residual blocks, a final set of convolutional layers batch normalization and ReLU activation layers are employed. The approach employs depthwise separable convolutions and linear bottlenecks to effectively decrease the quantity of parameters and computational burden, while still achieving satisfactory performance. MobileNetV2 is renowned for its efficacy in terms of velocity and resource consumption, rendering it well-suited for real-time applications on devices with constrained processing capabilities. The last

stage of this block involves enhancing the acquired features by the network and readying the output for subsequent processing.

The initial layers and the first Inverted Residual block capture low-level features from the input image. The subsequent Inverted Residual blocks progressively capture more abstract and high-level features through their skip connections and hierarchical processing. The final layers refine these features and prepare them for the transition to the decoder part of the Unet++ architecture.

The convolutional layer is an essential component in convolutional neural networks (CNNs), specifically intended for the purpose of analyzing 2D structures, such as images. The input undergoes a convolution operation, wherein learnable filters or kernels are employed to extract features from the input data.

Filters: Filters are small matrices that may be moved horizontally to cover the input data. Every filter acquires the ability to identify particular patterns or characteristics within the given input.

Parameters: The learnable parameters of the filters in a convolutional layer are subject to adjustment throughout the training process via back propagation. These settings facilitate the network in autonomously acquiring hierarchical properties from the input.

Stride: Stride is responsible for determining the magnitude of the step taken by the filter as it traverses the input data. Increasing the stride size leads to a decrease in the spatial dimensions of the resulting feature map.

Padding: Padding is a technique that entails the addition of additional border pixels to the input in order to mitigate the risk of information loss occurring at the edges.

Batch Normalization

Batch Normalization serves to enhance the stability of training and expedite the process of convergence. The process of normalizing involves the adjustment and scaling of activations during the training phase. The fundamental components of the Batch Normalization layer are outlined as follows:

Normalization: During the training process, Normalization is applied to each mini-batch by normalizing the input through the subtraction of the mean and division by the standard deviation. This procedure effectively mitigates the issue of internal covariate shift by maintaining a relatively consistent distribution of inputs to a layer throughout successive batches.

Parameters: Batch Normalization incorporates a pair of trainable parameters for each feature (or channel) within the layer, namely scale (gamma) and shift (beta). The utilization of these parameters enables the network to dynamically adjust the normalized output, hence offering adaptability and maintaining the layer's ability to effectively represent information.

The Rectified Linear Unit (ReLU) is an activation function that is frequently employed in artificial neural networks, specifically in deep learning architectures. The incorporation of the function within the network introduces a non-linear element, hence facilitating The capacity of the network to acquire knowledge of complex patterns, additionally correlations inherent within the data.

The Inverted Residual block is a fundamental component used in lightweight convolutional neural network structures, notably in topologies like MobileNetV2. Figure 4 shows the inverted residual block.The present block comprises a series of convolutional layers enclosed within a Sequential container. The initial ConvBNReLU sub module consists of convolutional layer that modifies the input channels, succeeded by batch normalization and ReLU6 activation. The second sub module of ConvBNReLU consists of depth-wise separable convolutional layer with a stride, which enhances the efficiency of the block. The concluding component comprises convolutional layer and batch normalization.

Skip connections are utilized in order to include the initial input into the final output, hence facilitating the acquisition of residual mappings. In general, the Inverted Residual block has been specifically devised to effectively capture and process features, all the while preserving a lightweight architecture that is well-suited for deployment on mobile and edge devices.

**B. UNet++ Decoder**

The decoder block is designed to up-sample and refine feature maps in the decoding part of a neural network. Figure 5 shows the Unet++ decoder architecture. The block consists of two convolutional layers (conv1 and conv2) each followed by batch normalization and ReLU activation, aiming to capture and enhance spatial features. The attention mechanisms (attention1 and attention2) within the block are implemented using an identity function, which effectively serves as a placeholder for an attention mechanism.

The attention mechanisms can be later replaced or modified to incorporate attention mechanisms that dynamically adjust the importance of different parts of the feature maps.

The attention layer is a component commonly used in neural network architectures, especially in natural language processing and computer vision tasks, to selectively focus on specific parts of the input or feature maps. The goal of attention mechanisms is to assign varying degrees of importance to different elements in the input, allowing the network to weigh and consider certain information more prominently.

## 4. Methodology

The dataset called "Semantic segmentation of aerial imagery" is downloaded from kaggle. This Dataset consists of aerial imagery of Dubai obtained by Mohammad Bin Rashid Space Centre(MBRSC) satellites. This dataset is annotated with pixel-wise semantic segmentation into 6 classes. The total 72 images in the dataset are grouped into 8 larger tiles. The classes of dataset are building, land, road, vegetation, water and unlabeled. Each tile consists of 2 sub-folders i.e., images and masks. Image sub-folder consists of 9 images and masks sub-folder consists of corresponding masks for those images. The images which are present in dataset are of many different sizes like 797x644, 509x544, 682x658, 1099x846, 1126x1058, 859x838, 1817x2061, 2149x1479 in each tile respectively. To process the pictures for testing and training,

the dimensions of all pictures should be of equal size. To achieve this, dataset need to be preprocessed. The preprocessing is carried out by cropping each image and masks into size divisible by 256. Further these images and masks are patchified to the size of 256x256. The sample images and their patchifying images are shown in figure 6.

For example, tile1 consists of 797x644 size of images and masks. So choose the nearest size with divisible by 256, we can get 768x512 size, from this total 6 patches are appearing. Similarly tile 2, 3, 4, 5, 6,7 and 8 has 2,4,12,16,9,56 and 40 patches respectively. Each tile consists of 9 images. So, a total of 1305 patches are available for both images and masks after patchifying. Masks are in RGB form and information is in the form of hexadecimal color code. So we need to convert hexadecimal to RGB values and then convert RGB labels to integer values and then to one hot encoding. Segmented images need to convert back into original RGB colors, otherwise the colors of image and its mask will be different and we could not identify the corresponding mask of each image. Predicted tiles need to be merged into a large image by minimizing blending artifacts or edge effects.

### A. Environmental Setup

This section is a description of the results obtained from the simulations conducted using the proposed methodology.

The programming language used is python. The execution is performed in Google Colab work environment with Python 3 Google Compute Engine backend, T4 GPU, 12.7GB RAM and 78 GB Disk space. A training : testing ratio of 70 : 30 is used for all experiments.

### B. Training and Testing

Figure 7 shows loss of training and validation loss of proposed method. The ideas of training loss and validation loss have significant importance within the field of machine learning, specifically in the context of training and evaluating models such as neural networks. The training loss is a statistic used to assess the discrepancy between the anticipated output and the actual target values for the training dataset during the training phase. In essence, it measures the extent of the model's deviation from the data it is undergoing training on. During the training phase, the model iteratively modifies its internal parameters, such as weights and biases in the context of neural networks, in order to minimize the training loss. The primary objective is to educate the model in a manner that enables it to exhibit strong generalization capabilities when presented with new, unfamiliar data. However, the concept of validation loss is significant since it serves as an autonomous metric for evaluating the effectiveness of a model . The measure is obtained by the assessment of the model's performance on a distinct dataset, known as the validation dataset, which has not been previously encountered by the model during the training process. In contrast to the training data, the validation dataset does not have any impact on the updates of the model's parameters. On the other hand, the validation loss offers valuable information about the model's potential performance on entirely novel and unknown data. The use of this technique is crucial in the identification of over-fitting, which refers to a situation where a model demonstrates exceptional performance on the training dataset but fails to properly generalize to new, unseen data.The training loss and validation loss are recorded for every epoch, acting as crucial metrics for evaluating the model's performance. The training loss, which measures the discrepancy between predicted and actual values on the training data, shows a progressive decline from 0.83 in the tenth epoch to 0.65 in the hundred epoch. In contrast, the validation loss, which evaluates the model's efficacy on a distinct dataset that was not used throughout the training process, exhibits an early decline from 0.83 to 0.72. However, it then undergoes a marginal rise during the hundred epoch. Figure 8 shows accuracy of training and validation of proposed method. The evaluation of machine learning models, particularly in supervised learning settings, relies heavily on the basic statistic of training accuracy. The performance of the model on the training dataset is assessed by quantifying it via the calculation of the ratio between the number of properly predicted occurrences and the total number of examples in the training dataset. The main objective of training accuracy is to assess the model's proficiency in acquiring the underlying patterns and connections present within the training data. A high training accuracy is indicative of the model's ability to effectively remember the training

examples and generate precise predictions on the data it was trained on. Nevertheless, it is essential to acknowledge that achieving a high training accuracy does not automatically ensure favorable performance when applied to novel, unobserved data. Overfitting is a potential concern, since the model may inadvertently include irrelevant noise or idiosyncratic features that hinder its ability to effectively generalize to other datasets. The validation accuracy serves as a complementary metric to the training accuracy, since it evaluates the performance of the model on an independent dataset referred to as the validation dataset. The dataset in question has unique characteristics that distinguish it apart from the training data, rendering it unsuitable for use during the model training phase. The accuracy of validation functions as an indicator of the model's generalizability well to novel, unseen data. Throughout the training phase, models undergo evaluation on both the training and validation datasets. When a model possesses

a elevated level of accuracy during training but a low level of accuracy during validation, it is possible that the model is over-fitting. Overfitting occurs when the model becomes too specialized to the training data and encounters difficulties in generalizing its predictions to new instances. The incorporation of a validation dataset is crucial in the process of model selection, which aims to identify a model that exhibits satisfactory performance not just on the training dataset but also on unobserved data.

During the tenth epoch, the validation accuracy was recorded as 0.54, whereas the training accuracy exhibited a higher value of 0.83. In the subsequent epochs, the validation accuracy demonstrates an improvement, reaching a value of 0.62. Conversely, the training accuracy experiences a tiny reduction, reaching a value of 0.77. As the training process advances, the validation accuracy demonstrates a consistent upward trend, ultimately attaining a value of 0.77 during the sixty epoch. However, it is noteworthy that the training accuracy experiences a decline, reaching a value of 0.69. During the ninety epoch, there is a significant rise in the validation accuracy, reaching a value of 0.80. The maximum validation accuracy attained is 0.83, observed on the ninety epoch. This section describes the evaluation metrics used and gives the extensive description about the results.The most extensively utilised metric for segmentation performance is the accuracy(A). Recall (R) and precision (P) are often used metrics for evaluating how well image classification systems work. Accuracy(Ac): is described as the total number of accurately located and separated instances (images) in the dataset under investigation[28]. The mathematical expression for accuracy is

$$\mathrm{Ac} = \frac{\mathrm{tp} + \mathrm{tn}}{\mathrm{tp} + \mathrm{tn} + \mathrm{fp} + \mathrm{fn}}$$

where the terms true positives (tp), true negatives (tn), false positives (fp), and false negatives (fn) are used.

**Precision(P)**:The equivalency of the proportion of accurately classified photos to all classified images is known as precision.

$$\mathrm{P} = \frac{tp}{tp + fp}$$

Here, tp denotes the appropriately categorised image and fp denotes false positives, or inaccurately classified photos.

**Recall:**The percentage of correctly classified photographs to all linked images in the database is known as recall. The recall as mathematically represented as

$$\mathrm{R}=\frac{tp}{tp+fn}$$

false negatives (fn) are photos that belonged to the right class but were incorrectly labelled by the classifier.

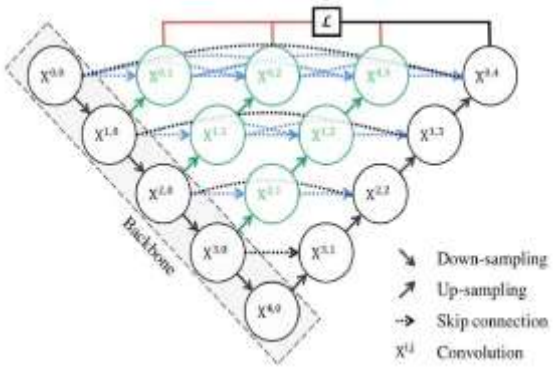



***Figure 1.*** *Architecture of Unet++ with MobileNetV2 as encoder*

**F1 score:** An increased F1-score, which is the result of multiplying the harmonic mean of recall and precision, indicates that the system has more predictive power. Evaluation of a system's performance requires more than just precision or recall. Mathematically speaking, the F1-score is expressed as

$$\mathrm{F1\ score} = 2\left(\frac{P.R}{P+R}\right)$$

In this case, P and R stand for recall and precision, respectively.

**Mean IoU**: determines the ratio of area overlapped by the two bounding boxes to the area of their union.

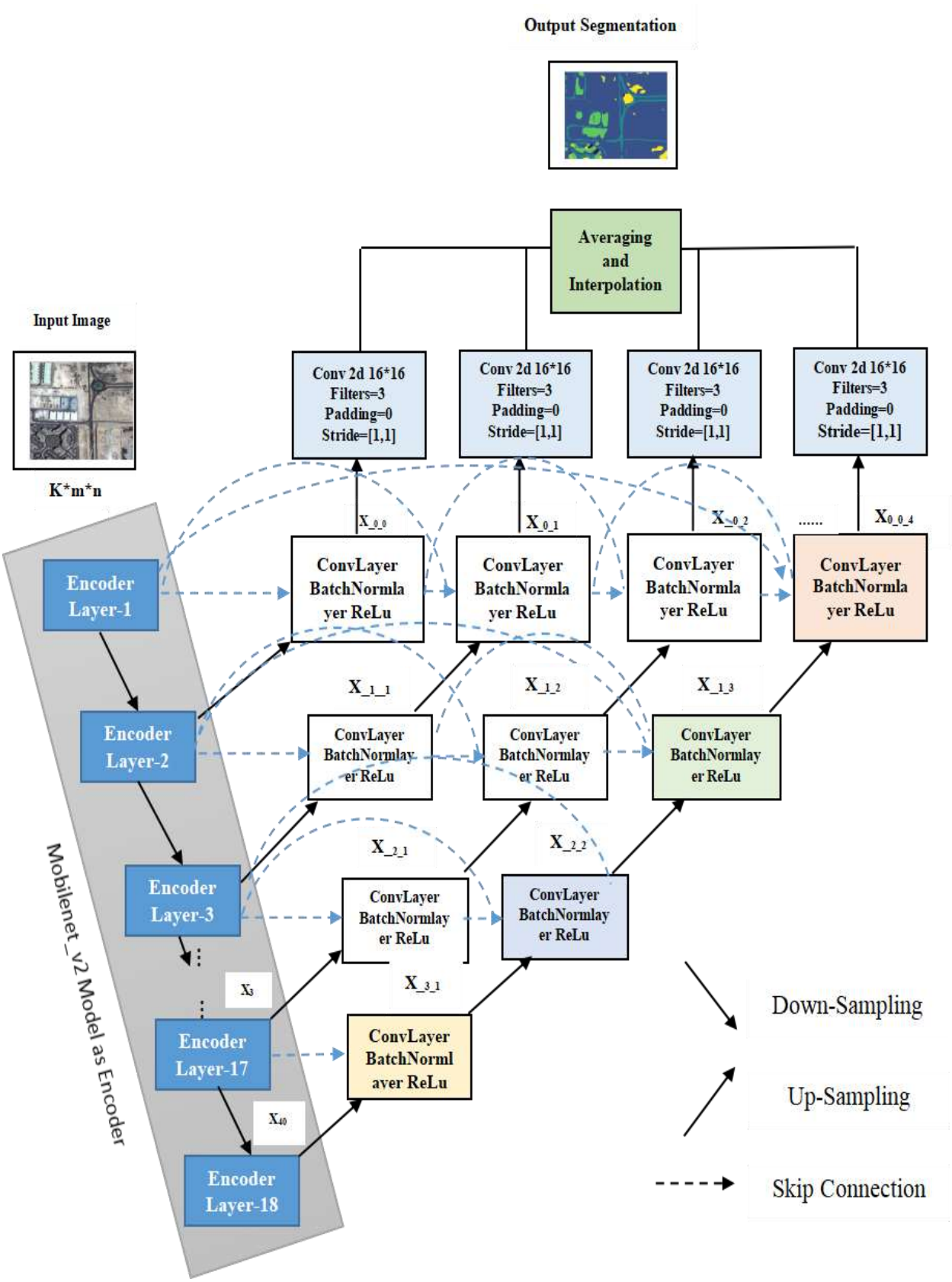



***Figure 2.*** *Internal architecture of Unet++ model with MobileNetV2*

The 2 bounding boxes are the real observation and prediction. The mathematical formula is:

$$J(A,B)=\frac{|A\cap B|}{|A\cup B|}=\frac{|A\cap B|}{|A|+|B|-|A\cap B|}$$

Where,the total area covered by both the bounding boxes (union). The area common between the bounding boxes (intersection).Table 1 shows the evaluation metrics of proposed method Unet++ with MobileNetV2. The metrics such as accuracy, precision, mean IOU, F1 score, and recall are determined for four different images of corresponding data set.

## 5. Comparative analysis of Experimental Results

Table 2 shows a comparative examination of several methodologies, with a focus on their respective levels of accuracy, mean IOU, precision, recall and F1-score.The performance of state of the art methods such as Unet, RUnet(Residual Unet), SE-UNet(Squeeze and Excitation-Unet), SE-RUNet(Squeeze and Excitation-Residual Unet), Unet++ with Efficientnet-b0, Unet++ with VGG19, Unet++ with ResNet50 are comparedwith UMV2 model. The proposed method achieved 1.01% accuracy, 3.12% Mean IoU, 0.21% precision, 1% recall, 0.58% F1-score high compared to the respective metrics of state of the art methods. The table presents a concise overview of the performance of the various approaches, revealing that MobileNetV2 attained the greatest level of performance compared to the other methods under consideration. This finding demonstrates a significant degree of proficiency in handling unfamiliar data, implying that the model has strong applicability capabilities.

The qualitative analysis of proposed model is performed by considering four different sample images from the Semantic segmentation of aerial imagery' data set. The sample images, ground truth mask of these images, prediction masks obtained from Unet++ with EfficientNetb0, Unet++ withVGG19, Unet++ with ResNet50 and Unet++ with mobileNetV2 are depicted in table 3. The Unet++ with Efficientnetbo able to predict most of the classes, but couldn't able to detect the object boundaries and assigning class labels at a pixel level due to reduction in spatial dimensions of the feature maps as the network deepens.The Unet++ with VGG19 predict most of the classes, but it struggle with segmenting objects with complex structures or in cluttered scenes due to lack of mechanisms to aggregate contextual information from larger receptive fields effectively, which are crucial for understanding scene layout and object relationships in segmentation.The Unet++ with ResNet50 predict most of the classes, but struggle with segmenting objects that vary widely in size or capturing contextual relationships in complex scenes as objects to be segmented can vary significantly in size. On the other hand, our proposed Unet++ with mobileNetV2 able to detect all the classes including very minure details efficiently and semantic segmentation results for all sample images are achieved efficiently. In this way our proposed model outperforms compared with other models.

## 4. Conclusions

The proposed satellite image segmentation model UMV2, which combines the Unet++ architecture with the lightweight MobileNetV2 encoder, has demonstrated noteworthy performance in accurately delineating features within satellite images. Through extensive experiments conducted on a diverse satellite image dataset, the model achieved an


***Figure 3.*** *MobileNetV2 Architecture*

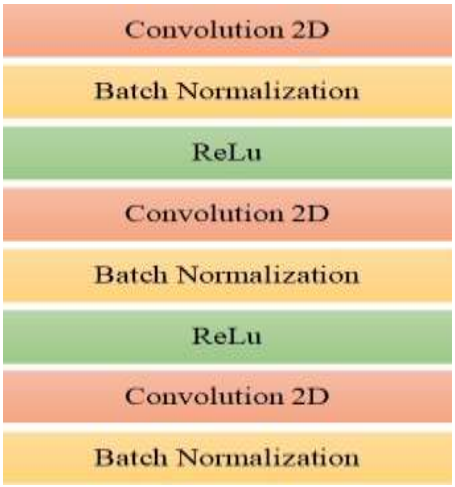
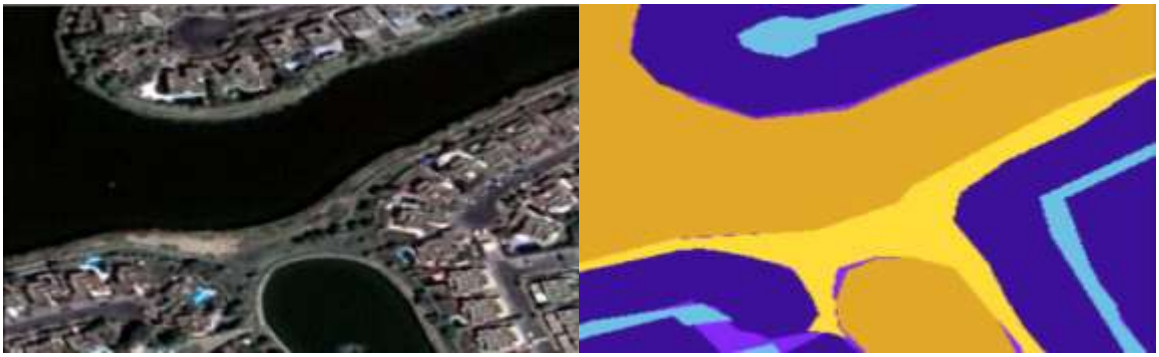



***Figure 4.*** *Inverted Residual Block*


***Figure 5.*** *Unet++ Decoder Architecture*



(a):Image 1 before patchify (b): Image 1 after patchify

(c):Image 1 before patchify (d): Image 1 after patchify

***Figure 6.*** *Image 1 and 2 before and after patchify*


***Figure 7.*** *Training and Validation loss of Proposed method*

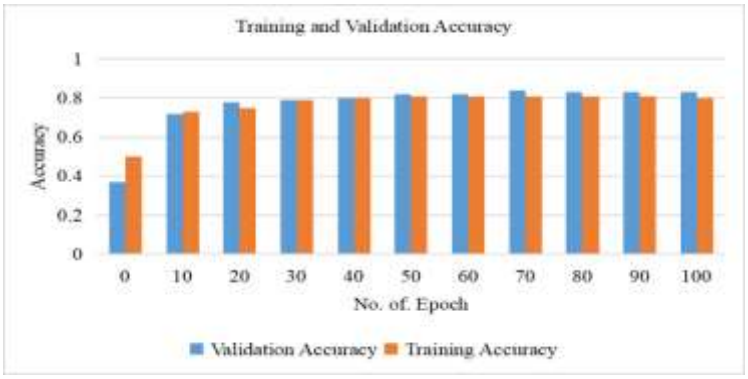



***Figure 8.*** *Training and Validation accuracy of Proposed method*

***Table 1.*** *The evaluation metrics of proposed method Unet++ with MobileNetV2.*

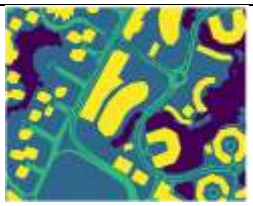
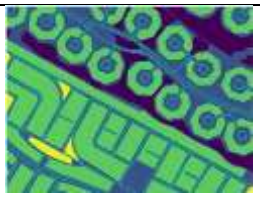
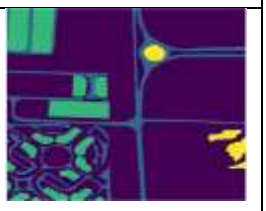
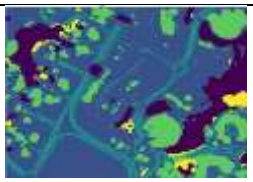
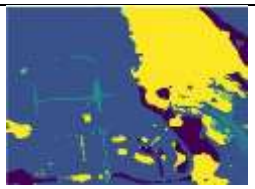
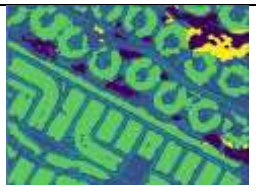
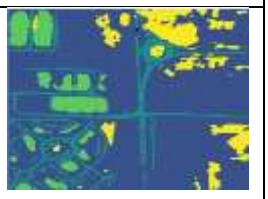
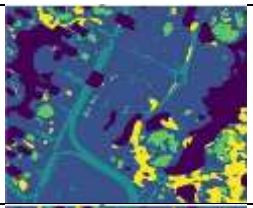
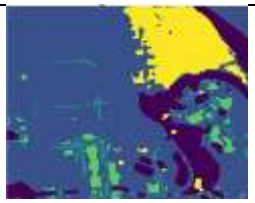
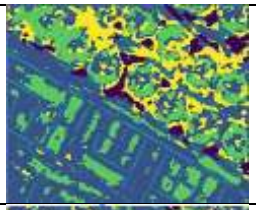
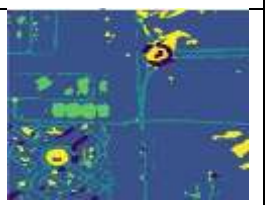
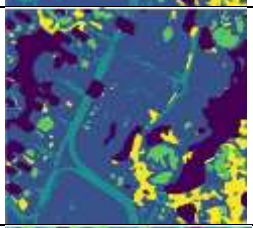
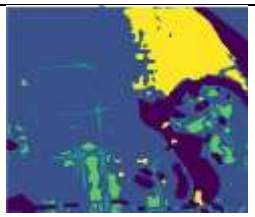
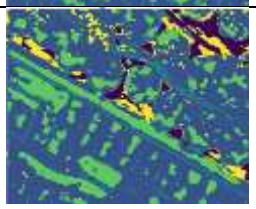
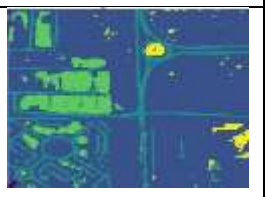
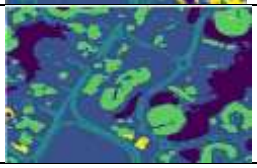
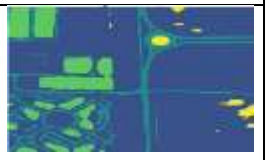
| **Images** | **Image 1** | **Image 2** | **Image 3** | **Image 4** |
|---|---|---|---|---|
| **Original input image** |  |  |  |  |
| **Ground truth mask** |  |  |  |  |
| **Unet++ with Efficientnet - b0** |  |  |  |  |
| **Unet++ with VGG19** |  |  |  |  |
| **Unet++ with Resnet50** |  |  |  |  |
| **Unet++ with Mobilenetv2** |  |  |  |  |

***Table 2.*** *Evaluation Metrics of Proposed UMV2 model*

| UMV2 Model | Accuracy | Precision | Recall | F1 Score | Mean IOU |
|---|---|---|---|---|---|
| **Image 1** | 0.8904 | 0.807041 | 0.807023 | 0.806393 | 0.482 |
| **Image 2** | 0.8912 | 0.819062 | 0.796282 | 0.807511 | 0.417 |
| **Image 3** | 0.8911 | 0.790811 | 0.783484 | 0.772145 | 0.411 |
| **Image 4** | 0.8917 | 0.828221 | 0.821937 | 0.825067 | 0.508 |

***Table 3.*** *The sample images, ground truth mask of these images, prediction masks.*

| Method | Accuracy | Mean IoU | Precision | Recall | F1 score |
|---|---|---|---|---|---|
| **Unet[29]** | 0.8806 | 0.4559 | 0.8854 | 0.8769 | 0.8812 |
| **RUnet[29]** | 0.8668 | 0.4842 | 0.8705 | 0.8642 | 0.8673 |
| **SE-Unet[29]** | 0.8849 | 0.4900 | 0.8891 | 0.8819 | 0.8855 |
| **SE-RUnet[29]** | 0.8706 | 0.4781 | 0.8736 | 0.8685 | 0.8711 |
| **Unet++ with Efficientnet-b0** | 0.8706 | 0.4788 | 0.8619 | 0.8782 | 0.8712 |
| **Unet++ with Efficientnet-b1** | 0.77 | 0.378 | 0.7807 | 0.689 | 0000 |
| **Unet++ with VGG19** | 0.8816 | 0.4813 | 0.8414 | 0.8412 | 0.8714 |
| **Unet++ with REsNet50** | 0.8512 | 0.4617 | 0.8515 | 0.8617 | 0.8618 |
| **UMV2 Model** | 0.8917 | 0.5212 | 0.8912 | 0.8919 | 0.8913 |

impressive accuracy of 0.8917, Mean IoU as 0.5212, precision of 0.8912, recall of 0.8919 and F1 score of 0.8913. This level of accuracy is particularly promising for real-world applications, like disaster management,land cover classification, environmental monitoring. The integration of Unet++ for its hierarchical feature capturing capabilities and MobileNetV2 for computaional efficiency has proven to be successful fusion, striking balance between accuracy and resource efficiency. The achieved accuracy of 89 percent underscores the model's effectiveness in extracting valuable information from satellite imagery, making it a compelling solution for remote sensing tasks in resource-constrained environments.

## Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

## References

[1]Li, X., Li, J (2024). Mfca-net: a deep learning method for semantic segmentation of remote sensing images. *Scientific Reports* 14(1), 5745

[2]Jia, P., Chen, C., Zhang, D., Sang, Y., Zhang, L.: (2024). Semantic segmentation of deep learning remote sensing images based on band combination principle: *Application in urban planning and land use. Computer Communications* 217, 97–106

[3]Marrewijk, B.M., Dandjinou, C., Rustia, D.J.A., Gonzalez, N.F., Diallo, B., Dias, J., Melki, P., Blok, P.M. (2024) Active learning for efficient annotation in preci- sion agriculture: a use-case on crop-weed semantic segmentation. *arXiv preprint arXiv:2404.02580*

[4]Wang, Z., Li, Z., Yu, X., Jia, Z., Xu, X., Schuller, B.W. (2024) Cross-scene semantic segmentation for medical surgical instruments using structural similarity based partial activation networks. *IEEE Transactions on Medical Robotics and Bionics*

[5]Do˘gan, G., Ergen, B. (2024) A new cnn-based semantic object segmentation for au- tonomous vehicles in urban traffic scenes. *International Journal of Multimedia Information Retrieval* 13(1), 11

[6] Ibrahem, ., Salem, A., Kang, H.-S.: Seg2depth (2024) Semi-supervised depth estima- tion for autonomous vehicles using semantic segmentation

and single vanishing point fusion. *IEEE Transactions on Intelligent Vehicles*

[7] Liao, Y., Kang, S., Li, J., Liu, Y., Liu, Y., Dong, Z., Yang, B., Chen, X. (2024) Mobile- seed: Joint semantic segmentation and boundary detection for mobile robots. *IEEE Robotics and Automation Letters*

[8]Lee, C., Soedarmadji, S., Anderson, M., Clark, A.J., Chung, S.-J. (2024) Semantics from space: Satellite-guided thermal semantic segmentation annotation for aerial field robots. *arXiv preprint arXiv:2403.14056*

[9] Zahra, A., Ghafoor, M., Munir, K., Ullah, A., Ul Abideen, Z. (2024) Application of region-based video surveillance in smart cities using deep learning. *Multimedia Tools and Applications* 83(5), 15313–15338

[10]Sun, G., Liu, Y., Ding, H., Wu, M., Van Gool, L. (2024) Learning local and global tem- poral contexts for video semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*

[11]Yuan, K., Zhuang, X., Schaefer, G., Feng, J., Guan, L., Fang, H. (2021) Deep-learning- based multispectral satellite image segmentation for water body detection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14, 7422–7434

[12]Jia, H., Lang, C., Oliva, D., Song, W., Peng, X. (2019) Dynamic harris hawks optimiza- tion with mutation mechanism for satellite image segmentation. *Remote sensing* 11(12), 1421

[13]Ghassemi, S., Fiandrotti, A., Francini, G., Magli, E. (2019) Learning and adapting ro- bust features for satellite image segmentation on heterogeneous data sets. *IEEE Transactions on Geoscience and Remote Sensing* 57(9), 6517–6529

[14]Rahaman, J., Sing, M. (2021) An efficient multilevel thresholding based satellite image segmentation approach using a new adaptive cuckoo search algorithm. *Expert Systems with Applications* 174, 114633

[15]Kotaridis, I., Lazaridou, M. (2021) Remote sensing image segmentation advances: A meta-analysis. *ISPRS Journal of Photogrammetry and Remote Sensing* 173, 309– 322

[16]Pare, S., Kumar, A., Singh, G.K., Bajaj, V. (2020) Image segmentation using multi- level thresholding: a research review. *Iranian Journal of Science and Technology, Transactions of Electrical* Engineering 44, 1–29

[17]Gupta, A., Watson, S., Yin, H.(2021) Deep learning-based aerial image segmenta- tion with open data for disaster impact assessment. *Neurocomputing* 439, 22–33

[18]Iqbal, J., Ali, M. (2020) Weakly-supervised domain adaptation for built-up region seg- mentation in aerial and satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* 167, 263–275

[19]Ovi, T.B., Mosharrof, S., Bashree, N., Islam, M.N., Islam, M.S.: Deeptrinet (2023).A tri-level attention-based deeplabv3+ architecture for semantic segmenta- tion of satellite images. In: *International Conference on Human-Centric Smart Computing,* pp. 373–384

[20]Fabel, Y., Nouri, B., Wilbert, S., Blum, N., Triebel, R., Hasenbalg, M., Kuhn, P., Zarzalejo, L.F., Pitz-Paal, R. (2022) Applying self-supervised learning for seman- tic cloud segmentation of all-sky images. *Atmospheric Measurement Techniques* 15(3), 797–809

[21]Wagner, F.H., Dalagnol, R., S´anchez, A.H., Hirye, M., Favrichon, S., Lee, J.H., Mauceri, S., Yang, Y., Saatchi, S. (2022) K-textures, a self-supervised hard clus- tering deep learning algorithm for satellite image segmentation. *Frontiers in Environmental Science* 10, 946729

[22]Boulila, W., Khlifi, M.K., Ammar, A., Koubaa, A., Benjdira, B., Farah, I.R. (2022) A hybrid privacy-preserving deep learning approach for object classification in very high-resolution satellite images. *Remote Sensing* 14(18), 4631

[23]Li, W., Chen, H., Shi, Z.(2021) Semantic segmentation of remote sensing images with self-supervised multitask representation learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14, 6438–6450

[24]Li, H., Li, Y., Zhang, G., Liu, R., Huang, H., Zhu, Q., Tao, C. (2022) Global and local contrastive self-supervised learning for semantic segmentation of hr remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing* 60, 1–14

[25]Sun, W., Gao, Z., Cui, J., Ramesh, B., Zhang, B., Li, Z. (2021) Semantic segmentation leveraging simultaneous depth estimation. Sensors 21(3), 690

[26]Zu¨rn, J., Burgard, W., Valada, A. (2020) Self-supervised visual terrain classification from unsupervised acoustic feature learning. *IEEE Transactions on Robotics* 37(2), 466–481

[27]Dong, H., Ma, W., Wu, Y., Zhang, J., Jiao, L.(2020) Self-supervised representation learning for remote sensing image change detection based on temporal prediction. *Remote Sensing* 12(11), 1868

[28]Shabbir, A., Ali, N., Ahmed, J., Zafar, B., Rasheed, A., Sajid, M., Ahmed, A., Dar, S.H. (2021) Satellite and scene image classification based on transfer learning and fine tuning of resnet50. *Mathematical Problems in Engineering* 2021, 1–18

[29]Aburaed, N., Al-Saad, M., Alkhatib, M., Zitouni, M., Almansoori, S., Al-Ahmad, H. (2023). Semantic segmentation of remote sensing imagery using an enhanced encoder- decoder architecture. ISPRS Annals of the Photogrammetry, *Remote Sensing and Spatial Information Sciences* 10, 1015–1020