Segmentation of Satellite Images of Solar Panels Using Fast Deep Learning Model

M. Arif Wani*‡, Tahir Mujtaba*

*Department of Computer Science, University of Kashmir, Srinagar, J&K, India -190006

(awani@uok.edu.in, mjtbatahir@gmail.com)

‡ Corresponding Author; Department of Computer Science, University of Kashmir, Srinagar, J&K, India -190006

Received: 01.12.2020 Accepted:22.12.2020

Abstract- Segmenting satellite images provides an easy and cost-effective solution to detect solar arrays installed on building tops and on ground over a region. Solar panel detection is the first step towards image based estimation of energy generation from the distributed solar arrays connected to a conventional electric grid. Segmentation models for small devices require light weight procedures in terms of computational effort. State-of-the-art deep learning segmentation models have the disadvantage that these require long training times, large number of floating-point operations (FLOPS) and tens of millions of parameters which make these models less suitable for devices with limited computational power. This paper proposes a deep learning segmentation architecture that is suitable for small devices. The proposed architecture combines features of Mobilenet classification architecture and Unet architecture in such a way such that it is efficient in terms of computational effort and produces segmentation results with good accuracy. The results of the proposed model are compared with the results obtained by various state-of-the-art segmentation models. The results demonstrate that the proposed model is computationally efficient as it requires less number of model parameters, less training time, consumes less number of FLOPS and produces good segmentation results with competitive accuracy.

Keywords Deep learning, semantic segmentation, convolutional neural networks, depthwise separable convolution, satellite imagery, solar panel arrays.

1. Introduction

Deep learning (DL) has been very effective in computer vision tasks such as image classification, object detection and semantic segmentation. Semantic segmentation finds its application in various domains like analysis of medical images, natural images, and satellite images. Supervised semantic segmentation obtains different segments in an image based on the predefined classes into which an image is to be segmented. It classifies each pixel into one of the classes dividing the image into segments where pixels of one segment belong to the same class. DL algorithms have shown promise both in supervised and unsupervised image segmentation over traditional machine learning (ML) algorithms. Although traditional ML algorithms have been used widely but these algorithms require extraction of features from an image first and which are then used in a classifier to classify each pixel int one of the predefined classes. DL algorithms are end-to-end algorithms where feature extraction and classification are done by a single algorithm. Convolutional neural networks (CNNs) is a DL approach that has been used in domains like medical image analysis, object detection, street view segmentation, and detection of objects from satellite images. Examples of CNN application in satellite image analysis include building detection, road extraction, vehicle detection, population estimation, poverty estimation, and identification of urban patterns. However, very little work has been done in detection of solar panels from satellite images.

Detecting solar panels from satellite images is challenging due to their varied shapes, sizes and colors and installations on roof tops can be at different angles. Further, using a device with limited computational power can make the task more challenging. This paper proposes a new segmentation architecture for solar panel detection that is suitable for devices with limited computational power. The paper is structured as follows: The state-of-the-art DL segmentation models are summarized in Section 3. The proposed model is described in Section 4. Section 5 presents

experimental results and discussion. The conclusion and future work is finally given in Section 6.

2. Related Work

Traditional ML models like Support vector machines (SVM), random forests have been used by researchers for a number of applications. This includes use of SVM for multiclass classification applications [17] and for solar panel detection, and extraction of rules [16] for classification applications and detect solar panels.

SVM has been used in [9] to count and detect the solar panels from the satellite images of Lemoore, California. It uses maximally stable external regions (MSER) to identify the elliptical regions and then applies features on these elliptical regions to determine the presence of solar panels in these elliptical regions. Although the method helps in counting the number of solar panels but the exact shapes cannot be determined with this approach. Hand crafted features and a random forest classifier have been used in [10] to detect solar panels from the Fresno city dataset [11]. It produces various false positives, therefore a post processing step is used to identify high confidence pixels. However, post processing may sometimes miss less bright pixels which may actually belong to solar panels. Further, the success of such an approach highly depends on quality of the designed features.

Wıth the emergence of DL in the recent years, DL models have been very successful for various computer vision tasks, which includes image classification, object detection and segmentation tasks. Traditional ML approaches like subspace grid based aproach is described in [23,24], and cluster based described in [27] for classification task produce less accuracy results than DL approach. The same is true even with the hybrid approach described in [28] for classification task. The authors in [12] have used a DL based VggNet classification model for Imagenet classification. A deep residual learning model consisting of residual blocks has been used by authors [13] for image classification and produced better results than the VggNet. The residual structure helps to alleviate the vanishing gradient problem and thus helps deep layered models to get trained easily. The authors in [14] use a densely connected model called as DenseNet for image classification where all layers within a block are densely connected to the following layers. The dense connections eases the flow of feature propagation through the layers. Image classification results from DenseNet surpassed those of ResNet model. The results show that CNN produces better classification than the ML based models like random forest classifier, SVM, subspace grid, cluster, hybrid approaches.

DL CNNs have also been used for segmentation tasks which includes detecting solar panels from the satellite images. Authors in [1] used Convolution Neural Network (CNN) for segmentation f finger print iages. Authors in [18] used a CNN classifier for detecting solar panels from the satellite images. Each pixel is classified to determine whether it belongs to the solar panel or not. The authors compare results with random forest approach and it has been shown that CNN model produces better results than random forest approach. The work in [19] uses a pretrained Vggnet classifier. It used the basic architecture of Vggnet classifier with 6 convolutional layers and 2 fully connected ones. Its classification model classifies each pixel to determine whether it belongs to the solar panel or not. A post processing method is applied to connect contagious detected pixels to form regions. Though such a model can be used to detect the regions containing solar panels but being a classification model exact shape of solar panel arrays cannot be acquired. A fully convolutional network model has been used by authors in [26] for large scale solar panel array mapping on the aerial RGB images of Boston and San Francisco. The authors report a precision of 0.855 and recall of 0.873 for the dataset of San Francisco and a precision score of 0.812 and recall of 0.840 for the dataset of Boston. However no segmentation metric like dice coefficient or Intersection over union has been reported.

In addition to basic CNN models, a number of state-of-the-art deep learning (DL) segmentation models like FCN, Unet, Segnet, DeepLab v3, Dilated Net, dilated Resnet and PSPNet have been proposed in the literature. FCN [2] is one of the first DL segmentation models that is designed by converting the classification network like Vggnet, Googlenet and Alexnet into segmentation models. The Unet segmentaion model was proposed in [3] which uses a fully encoder decoder structure for semantic segmentation. A similar encoder decoder architecture called Segnet was proposed in [4] but it differs from Unet in the way upsampling process is done. Segnet uses index based upsampling process to upsample the feature maps in its decoder. A multi context aggregation based segmentation model was proposed in [6] which uses a context module for aggregating the context of the objects in an image.

Some work has been reported in the literature on the use of the state-of-the-art deep learning segmentation models for detection of solar panels. DL segmentation network Segnet has been used in [20] on the satellite images of Fresno and it was shown that the Segnet model performs better than classification models like Vggnet model. But no specific accuracy or segmentation performance measures have been reported and the models have been trained on image patches

of size 41x41. However, small image patches may contain less diverse background details which may restrict the model to perform well on images with diverse and dense backgrounds. Unet has been used in [21] to detect solar panels from the aerial images of Switzerland. Fast RCNN based on Resnet-50 has been used in [25] to detect solar panels by placing bounding boxes around panels. However, such an approach cannot be used to determine the exact shape and boundary of solar panels which may not result in accurate energy estimation. A solar mapper was introduced in [29] to determine the size, location and capacity of solar panels using satellite images. It reported segmentation accuracy with aggregate f1 score of 0.76 over the cities of Fresno, Modesto and Stockon. It also reported an aggregate f1 score of 0.85 over the satellite images of Connecticut. With the advancement in Deep Learning, there is a scope to improve the solar panel segmentation accuracy further by developing new DL segmentation architectures.

3. State-of-the-art Deep Learning Segmentation Models

The most prominent and improved segmentation models proposed by researchers include Unet, Segnet, Dilatednet, Pspnet, Deeplab v3+, Dilated residual network which have been used for different segmentation tasks. In this paper these models are explored for the problem of solar panel detection and are compared with the proposed models. These most prominent segmentation models are summarized below.

a. Unet:

Unet was proposed for medical image segmentation and it uses an encoder and a decoder structure. Encoder consists of encoder blocks which usually consist of series of convolutional and a pooling layer like in any classification architecture, however no fully connected layer is present. It extracts features and downsamples the input image resolution. Encoder is followed by a decoder which is responsible for restoring the image resolution and perform the pixel wise classification. It consist of various decoders where each decoder usually consist of upsampling layers for feature map upsampling followed by a concatenation layer which concatenates the upsampled features with those of corresponding encoder features transferred via skip connections. The concatenation layer is followed by a pair of convolutional layers. The skip connections which are from every encoder block to the corresponding decoder block having same feature map resolution help in localization and densify the segmentation. The last decoder block is followed by 1x1 convolution and a sigmoid function for pixel wise classification.

b. Segnet:

Segnet was proposed for scene understanding application, it also uses an encoder decoder structure but does not use skip connections. For image upsampling it uses index based max pooling technique where indices of pixels with maximum values in max pooling are recorded and later the saved index and the value are used to upsample the image features in the decoder. So it does not use any transpose convolution for upsampling. Main reason for this is to decrease the parameter count used in transpose convolution and thereby reducing training time.

c. Dilatednet:

One of the first CNNs model that used the concept of dilated convolutions is Dilatednet. Dilated convolutions are the convolutions with widened filters that have zeros in between the various weight values. These convolutions have been used for dense segmentation, context aggregation and parameter reduction and enlarging the field of view. Dilatednet uses a front end module for feature extraction and dilation module for context aggregation. The front end is like a normal Vggnet with dilated convolution used in 4^{th} and 5^{th} block. It uses two types of context aggregation modules- a basic and large containing seven layers with different dilation rates. Basic module applies dilation of 1, 1, 2, 4, 8, 16, 1 and 1. It then upsamples the feature maps by using bilinear upsampling and calculates the probability segmentation map.

d. Pspnet:

Pspnet was proposed for scene parsing and uses a pyramid pooling module (PPM) for context aggregation and detection of small objects in a scene. The PPM uses different pooling rates in parallel for generating feature maps of different sub regions and forms pooled representation of different positions. These pooling operations form feature maps of varying sizes which are then upsampled using bilinear upsampling and segmentation maps are produced.

Deeplab v3+*:*

Deeplab v3+ is an improvised version of its predecessor Deeplab v3 and uses an encoder decoder structure with a skip connection from encoder to decoder. It uses dilated context aggregation module known as *atrous spatial pyramid pooling module* with 1x1 convolution and dilation rates of 1x1, 6, 12 and 18 to refine its segmentation results. The encoder uses Resnet-50 and xception network with dilated convolutions. Deeplab v3+ uses a relatively better decoder than its predecessor by using a skip connection from one of its encoder to its corresponding decoder for better localization. The upsampling used is bilinear upsampling.

e. Dilated Resnet:

Dilated Resnet utilizes Residual network for image classification and segmentation by incorporating dilated convolutions in its encoder and dilation module. It utilizes Resnet-18 and its variants as encoder and utilizes a different dilation module with smaller dilation rates ad alleviates the degridding pattern problem that occurs with high dilation rates. It upsamples the feature maps so obtained from dilation unit by using bilinear interpolation.

4. Proposed Deep Learning Segmentation Architecture

A new DL segmentation architecture is proposed here which is based on Mobilenet architecture [15] and Unet architecture. It uses depthwise separable convolution unit in its encoder blocks instead of using the standard convolution operations. This allows the proposed architecture to get trained in less time, use less parameters and perform with less floating point operations (FLOPS) as compared to the state-of-the-art models described in the last section. A detailed comparison with experimental results of the proposed model and state-of-the-art models is provided in the next section. The model can be termed as light weight model from computational point of view and is suitable for devices with limited computational power. The architecture has full encoder decoder structure with skip connections at all levels for dense segmentation and fine boundary refinement. The components of the proposed architecture are discussed below:

4.1 Encoder

The encoder for our proposed architecture is based on Mobilenet architecture which uses depthwise separable (DWS) convolution instead of standard convolution. MobileNet is one of the real time classification architectures with very less parameters and requires less FLOPS. The use of Mobilenet architecture (with fully connected layers removed) as encoder reduces the number of parameters without any significant drop in the segmentation accuracy. The encoder of our proposed model consist of various encoder blocks where each block consists of depthwise separable (DWS) convolution units shown in Fig. 1. The DWS convolution unit implements the following operations: depthwise (DW) convolution, batch normalization (BN) layer, ReLU activation, a pointwise (PW) convolution, BN layer, and ReLU activation as shown in Fig 1. The number of

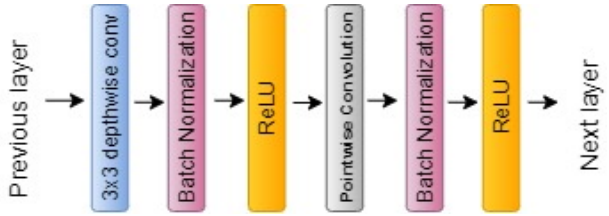


Fig 1. Encoder block: DWS convolution unit

DWS units and filters used with each convolution operation varies with each encoder block in the encoder.

4.2 Decoder (Udec-BL-BN)

A decoder is essentially meant for upsampling the feature maps extracted from the encoder. Different state-of-the-art architectures use different upsampling techniques and with skip connections at different scales or with no skip connections at all. For example, Unet uses skip connections at levels 1/2, 1/4, 1/8 and 1/16, Segnet uses a different decoder with index based maxpooling. DeepLabv3 and DilatedNet use naïve decoders with no skip connections. Similarly, PSPNet and DeepLab v3+ use only one skip connection at scale 1/8 and 1/4 respectively. Despite having rich semantic information present in the last layer of the encoder, the models with few or no skip connections are unable to get detailed boundary information.

The decoder of the proposed architecture uses skip connections at all levels i.e. 1/2, 1/4, 1/8, 1/16 which is similar to that of the Unet architecture. The use of skip connections leverages the decoder with rich semantic

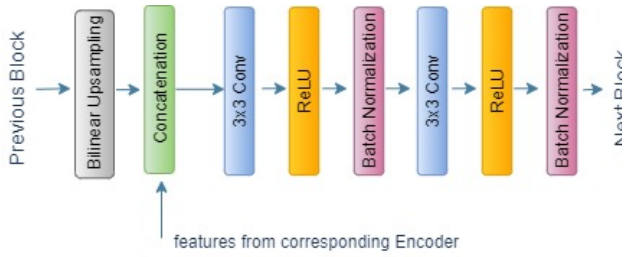


Fig 2. Proposed decoder block: *Udec-BL-BN*

information for accurate localization and boundary recovery. The proposed decoder uses a series of blocks which is similar to the decoder of the Unet model but the difference is in the manner upsampling is done and the number of filters used per decoder block. Here upsampling is done by bilinear upsampling which calculates a pixel value by interpolating the values from the nearest pixels which are known. The ratio of contribution from each nearby pixel matters here and is inversely proportional to the ratio of their corresponding

distance. We have managed to get good accuracy with lesser number of filters in our decoder blocks than present in the decoder blocks of the original Unet model. By using Bilinear upsampling and lesser number of filters in decoder blocks we are able to decrease the number of parameters of the model. The proposed decoder is shown in Fig. 2.

4.3 Architecture of the proposed model

The block diagrams of the proposed architecture is shown in Fig. 3 and we call it as Unet-Mobilenet architecture as it is based on Unet and Mobilenet architectures. It has a full encoder decoder structures with skip connections used at all levels. The left side of the block diagram represents the encoder and right side represents the decoder and in between is the bridge block. The Encoder blocks consists of Conv and DWS units with first encoder consisting of a Conv unit and all other encoder blocks consist of DWS units. The Second, third, fourth and the bridge block conatin two DWS units each where as fifth encder block contains six DWS units. The number of flters used in the DW and PW convolutions of each DWS unit is shown along after the '/' symbol respectively. Conv unit is a simple block comprising of standard convolution, batch normalization and ReLU activation. The 3x3 DW convolution, batch normalization, ReLU activation, 1x1 PW convolution, batch normalization and ReLU activation of the DWS unit is shown to the right side of the model in Fig. 3. The output from each encoder is downsampled by convolution with a stride factor of 2. In decoder, five blocks have been used. The composition of each decoder block is shown in Fig. 2 and in Fig 3 alongside the model. It consists of bilinear upsampling layer, followed by concatenation layer which concatenates the encoder block features with the corresponding decoder block features having the same resolution shown by dotted arrows. The concatenation layer is followed by two standard convolutions having different filters at different blocks. Each convolution is followed by batch normalization and ReLU activation. The last decoder block is followed by 1x1 convolution and a sigmoid layer for pixel classification. The output of the model is 224x224x1 predicted segmentation map showing the locations of the solar panels. The detailed layered architecture of the model is given in Table 1. It gives the composition of each encoder, bridge and decoder blocks. A DWS/Conv column indicates whether a DWS or Conv unit has been used. No DWS units have been used in the decoder. The Layer column indicates the different types of layers used: Conv-dw indicates a depthwise (DW) convolution, Conv-pw indicates a pointwise (PW) standard convoution, Conv indicates a standard convolution, Bi-Upsam indicates bilinear upsampling, Concat indicates a concatenation layer which concatenates features from the corresponding encoder and does the job of skip connections.

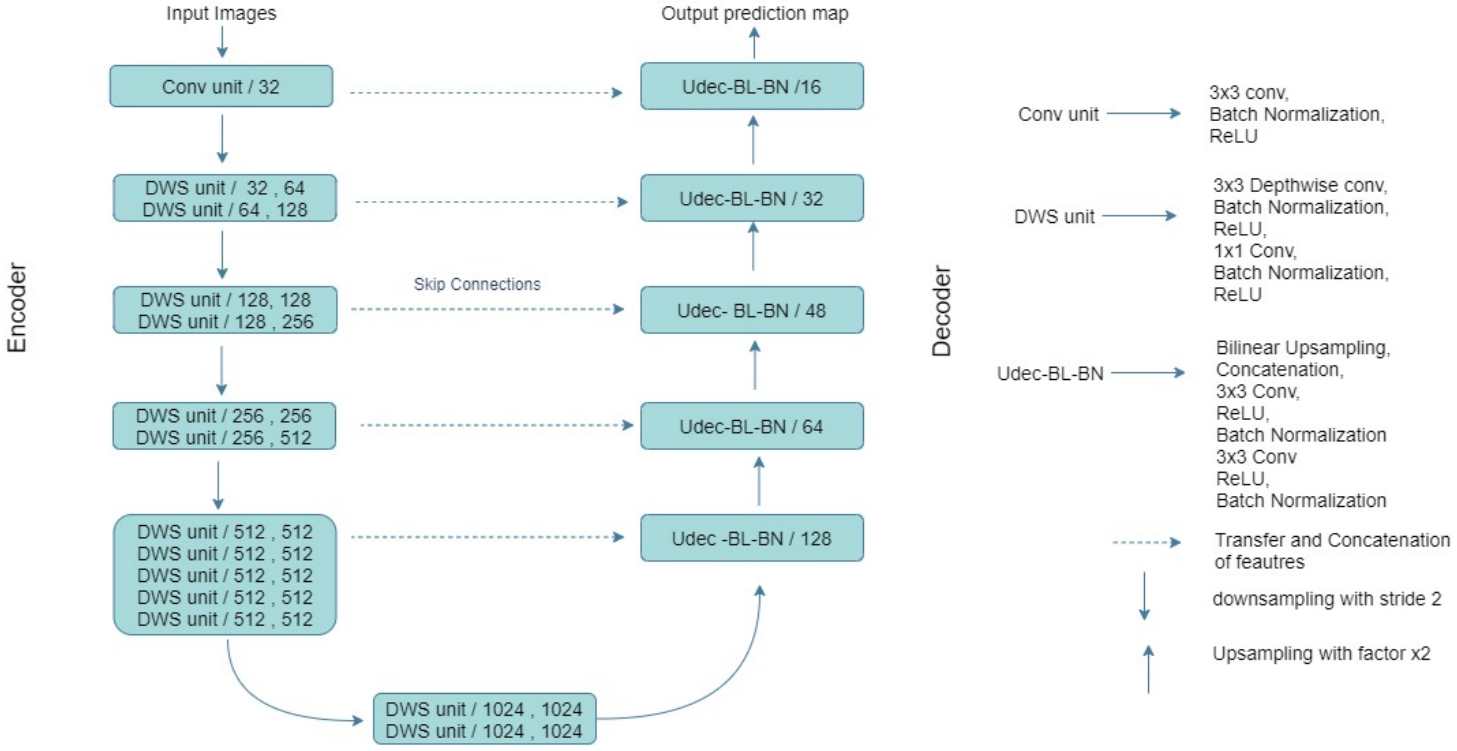


Fig 3. Proposed Unet-Mobilenet segmentation model

Sigmoid layer classifies the pixels into solar panel pixels or non-solar panel pixels. Input column indicates the size of the input (height and width) that the layer is recieving. Filter size indicates size of the kernel used with each convolution operation. A filter size of 1x1 is used with a PW convolution and 3x3 filter size is used with DW and standard convolution. Each Conv-dw, Conv-PW, and Conv layer is followed by Batch Normalization and ReLU function. A DWS unit comprise of depthwsie convolution followed by pointwise convolution. The number of filters used in each convolution operation is specified in 'number of filters' coulmn. The value in the 'Stride' column shows the number of strides used in the convolution operation. A stride of 1 does not change the dimensions of the feature maps while as a stride value of 2 halves the dimensions of feature maps. The last column 'Output size' indicates the ouput dimensions (height and width) of the feature maps that a particular layer outputs.

5. Experimental results and discussion

The proposed architecture is evaluated and compared with various state-of-the-art segmentation models: FCN, Unet, Segnet, DeepLab v3, Dilated Net, dilated Resnet and PSPNet. The results are also compared with the segmentation architectures previously proposed by the authors: Vggnet-Unet, Resnet-Unet and fully Resnet-Unet. The comparision is done in terms of classification and segmentation accuracy measured through dice coefficient, precision and recall. Other performance metrics used for fast training, computational efficiency and light weightedness are training time, number of floating-point operations and number of parameters required respectively. The models are also compared by using shapes of solar panels detected.

5.1 Dataset used:

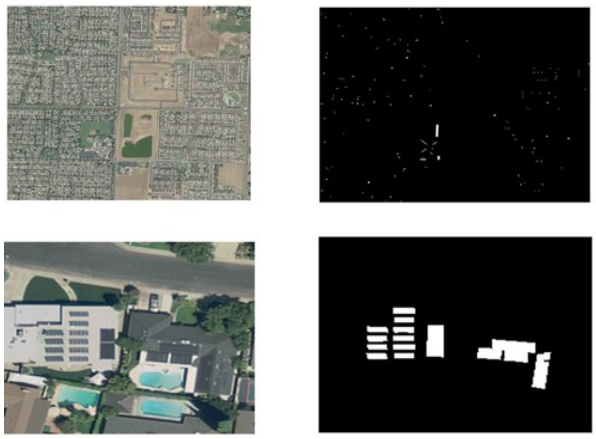


Fig. 4. First row shows an image and its mask from dataset of 5000x5000 dimensions. Second row shows an image with its mask of 224x224 dimensions.

Table 1. Layered Architecture of the Proposed Model

Encoder Blocks	DWS / Conv Unit	Layer	Input	Filter Size	Number of Filters	Stride	Output size
Encoder Block 1	Conv unit	Conv	224x224	3x3	32	2	112x112
Encoder Block 2	DWS	Conv-dw	112x112	3x3	32	1	112x112
		Conv-pw	112x112	1x1	64	1	112x112
	DWS	Conv-dw	112x112	3x3	64	1	112x112
		Conv-pw	112x112	1x1	128	1	112x112
Encoder Block 3	DWS	Conv-dw	112x112	3x3	128	2	56x56
		Conv-pw	56x56	1x1	128	1	56x56
	DWS	Conv-dw	56x56	3x3	128	1	56x56
		Conv-pw	56x56	1x1	256	1	56x56
Encoder Block 4	DWS	Conv-dw	56x56	3x3	256	2	28x28
		Conv-pw	28x28	1x1	256	1	28x28
	DWS	Conv-dw	28x28	3x3	256	1	28x28
		Conv-pw	28x28	1x1	512	1	28x28
Encoder Block 5	5 * DWS	Conv-dw	28x28	3x3	512	2	14x14
		Conv-pw	14x14	1x1	512	1	14x14
Bridge Block	DWS	Conv-dw	14x14	3x3	1024	2	7x7
		Conv-pw	7x7	1x1	1024	1	7x7
	DWS	Conv-dw	7x7	3x3	1024	1	7x7
		Conv-pw	7x7	1x1	1024	1	7x7
Decoder Block 5	-	Bi-Upsam	7x7	-	-	-	14x14
	-	Concat	14x14	-	-	-	14x14
	-	Conv	14x14	3x3	128	1	14x14
	-	Conv	14x14	3x3	128	1	14x14
Decoder Block 4	-	Bi-Upsam	14x14	-	-	-	28x28
	-	Concat	28x28	-	-	-	28x28
	-	Conv	28x28	3x3	64	1	28x28
	-	Conv	28x28	3x3	64	1	28x28
Decoder Block 3	-	Bi-Upsam	28x28	-	-	-	56x56
	-	Concat	56x56	-	-	-	56x56
	-	Conv	56x56	3x3	48	1	56x56
	-	Conv	56x56	3x3	48	1	56x56
Decoder Block 2	-	Bi-Upsam	56x56	-	-	-	112x112
	-	Concat	112x112	-	-	-	112x112
	-	Conv	112x112	3x3	32	1	112x112
	-	Conv	112x112	3x3	32	1	112x112
Decoder Block 1	-	Bi-Upsam	112x112	-	-	-	224x224
	-	Concat	224x224	-	-	-	224x224
	-	Conv	224x224	3x3	16	1	224x224
	-	Conv	224x224	3x3	16	1	224x224
		Conv	224x224	1x1	1	1	224x224
		Sigmoid	224x224	-	-	-	224x224

The most commonly used and publicly accessible solar panel dataset is Duke California Solar array dataset (DCSA) [11] . The dataset has been used in several studies and has satellite images of four California cities: Modesto, Oxnard, Stockon and Fresno. It has 601 high resolution RGB images of size 5000x5000 containing approximately 19,000 solar panels. The images are dense with a resolution of $\leq$ 0.3 m and represent diverse background of urban, suburban and rural landscape. The training of the model on diverse backgrounds makes it more reliable. Geometric and pixel coordinates of the vertices of solar panels present in the images are also given in the dataset. Satellite images and their corresponding masks (ground truth) are required for the detection of solar panels using the segmentation process. But the DCSA dataset does not include the ground truth masks with its satellite images. The DCSA dataset images were manually annotated with a labelling tool that used the specified pixel coordinates of solar panels given in the dataset. The corresponding masks

were prepared by changing the pixels of solar panels to white and all other pixels to black as depicted in Fig. 4. For training purposes, images were cropped to the size of 224x224. Fig. 4 displays sample image of size 5000x5000 and a cropped image of size 224x224 with their corresponding masks.

5.2 Experimental setup and training details

All models have been implemented in Python using Keras with TensorFlow as backend and trained on the Google Colab platform with 12 GB Tesla K80 GPU and 13 GB RAM. A total of 958 images from the cities of Oxnard and Fresno have been used for training purposes. The training and validation split value of 0.2 has been used. The Adam training algorithm has been used and all the models have been trained for 200 epochs, using a commonly used learning rate of 1e-4 and 1e-5.

5.2.1 Performance metric used

The performance measuring metrics of Dice similarity coefficient, precision and recall have been used in this work. Training time, number of parameters used and number of floating-point computations required have been used for training speed, light weightedness, and computational efficiency comparison purposes. The performance metrics are briefly summarized below:

i. Precision

It is the measure of correctness and is defined as number of true positives (TP) to the total number of true positives and false positives (FP).

$$Precision = TP / TP + FP$$

ii. Recall

It is the measure of completeness and is calculated as the number of true positives divided by the total number of true positives and false negatives.

$$Recall = TP / TP + FN$$

iii. Dice similarity coefficient (DSC)

Also known as f1 score and is the most commonly used performance measure for segmentation. This metric is used to quantify how similar the ground truth annotated segmentation matches with the predicted segmentation of the model. Given two set of pixels X (predicted) and G (ground truth) the DSC can be defined as:

$$DSC = \frac{2\sum_1^N x_i y_i}{\sum_1^N x_i^2 + \sum_i^N g_i^2}$$

The value of dice similarity coefficient ranges from 0 to 1. Value close to 1 means more overlap and similarity between the between the two regions, hence more accurate predicted segmentation from the model and value of 0 means no overlap and no similarity between two regions.

iv. Gigaflops

Flops stands for floating-point operations per seconds and is usually calculated in Giga units for a model. GFlops are used to determine the computational requirements of a model. Less number of GFlops means less operations and less computation requirements of the model.

5.2.2 Dice Loss Layer

The reason behind the use of dice loss function is its ability to deal with unbalanced datasets where background pixels are far more than the foreground pixel like solar panels. Dice loss function does not need to reweight the foreground pixels. Dice loss (DL) is defined as:

$$DL = 1 - \left(\frac{2\sum_1^N x_i y_i}{\sum_1^N x_i^2 + \sum_i^N g_i^2}\right)$$

5.3 Performance Comparison

The experimental results of the proposed architecture are compared with those obtained from the various state-of-the-art DL segmentation architectures.

5.3.1 Comparison with State-of-the-art DL Segmentation Models

The proposed model has been compared with the state-of-the-art DL models using recall, precision, dice similarity coefficient (DSC), loss, number of parameters and training time and GFLOPS. The results have been categorized according to the different classes of the metrics as under. The encoders for all models are same in all comparisons as mentioned in Table 2.

5.4 Empirical Comparison

This section gives empirical comparison of the proposed model with various models pointed out above.

5.4.1 Comparison based on precision and recall scores

The comparison of models using precision and recall is shown in Table 2. As can be seen from the Table 2, the proposed model produces the best precision measure value. The best recall value is produced by authors' previous models. High precision means less false positive rate but a tradeoff between precision and recall is necessary for a

Table 2. Comparison on the basis of classification metrics

Model		Encoder	Recall	Precision
Unet		Vggnet 16	0.8695	0.8773
Segnet		Vggnet 16	0.816	0.8252
DilatedNet		Vggnet 16	0.6427	0.6813
PspNet		Resnet 50	0.7002	0.6113
DeepLab v3+		Resnet 50	0.7536	0.7319
Dilated Resnet		Resnet 18	0.6615	0.6557
Our other Implementations	Unet-Vggnet-BN	Vggnet 16 (with batch normalization)	**0.9266**	0.9227
	Unet-Vggnet-DWS	Vggnet 16 (with DWS convolution)	0.9115	0.9045
	Unet_Resnet	Resnet Blocks	0.9078	0.9198
	Fully Unet-Resnet	Resnet Blocks	0.8994	0.9421
Proposed	Unet-Mobilenet	Encoder (Mobilenet)	0.8498	**0.9595**

model to minimize both the false positives and false negatives. Increasing the training time of the proposed model can improve its recall score.

5.4.2 Comparison based on Segmention metrics

Dice similarity coefficient (DSC) and dice loss have been used for measuring the segmentation accuracy. A DSC score close to 1 means more overlap of ground truth and predicted segmentation maps and hence better segmentation accuracy. Lower values for dice means higher values for DSC. The comparison is shown in Table 3. The proposed model has about 90% of segmentation accuracy, which is much better than the state-of-the-art models.

Table 3. Comparison on the basis of Segmentation metrics.

Model		DSC	Dice Loss
Unet		0.8780	0.1277
Segnet		0.7698	0.2301
DilatedNet		0.6698	0.3399
PspNet		0.6555	0.3477
DeepLab v3+		0.7420	0.2607
Dilated Resnet		0.6608	0.3421
Our other Implementations	Unet-Vggnet-BN	**0.9284**	**0.0757**
	Unet-Vggnet-DWS	0.9132	0.11
	Unet_Resnet	0.9184	0.0868
	Fully Unet-Resnet	0.9192	0.0825
Proposed	Unet- Mobilenet	0.9094	0.1

5.4.3 Comparison based on speed

The time taken to train the proposed model and its comparison with other models is shown in Table 4. The proposed model takes 0.91 hours for its training which is much better improvement over the previous models proposed by authors and the state-of-the-art models. This feature of the proposed model makes them suitable for small devices that offer low computational power.

5.4.4 Comparison based on number of model parameters and number of floating-point operations

A model is a light weight model if the number of model parameters and number of floating-point operations (GFLOPS) required are less. This metrics gives the light weight measure of a model. The comparison of light weight measure of the proposed model and various other models is shown in Table 5. The proposed model uses 5.53 million parameters and requires 4.74 GFLOPS which is a better than the authors previous models and state-of-the-art models. The proposed model has less values for this metrics which makes

Table 4. Comparison on basis of fastness.

Model		**Training time (in hours)**
Unet		4.7
Segnet		5.46
DilatedNet		1.83
PspNet		7.85
DeepLab v3+		6.68
Dilated Resnet		1
Our other Implementations	Unet-Vggnet-BN	3.5
	Unet-Vggnet-DWS	2.2
	Unet_Resnet	3.33
	Fully Unet-Resnet	8.5
Proposed	Unet- Mobilenet	**0.91**

it ideal for smaller devices with limited computational power. The file size of model parameters of Unet-Mobilenet is 64 MB which is much smaller than the file size of state-of-the-art models which is hundreds of megabytes.

5.4.5 Comparison based on predicted segmentation maps

A comparison of predicted segmentation maps obtained by the proposed model with those of authors' earlier models and state-of-the-art models is given in Fig 5. As can be seen from Fig. 5, the proposed model gives correct predictions with fine and crisp boundaries with less false positives and false negatives. This is mainly due to the reason that the proposed model uses full encoder decoder structure and transfers features using skip connections at all scales.

5.5 Comparison with the Existing Deep Learning Models Used for Solar Panel Detection

Table 6 provides a comparison of the proposed model with those reported by the researchers in the literature on solar panel detection. The proposed model has better precision and DSC scores.

Table 5. Comparison on the basis of model lightweightedness and Computational efficiency.

Model		Number of parameters (in millions)	GFlops (in giga units)
Unet		43.41	84.9
Segnet		29.43	68.5
DilatedNet		23.59	54.3
PspNet		27.72	45.4
DeepLab v3+		41.43	40.2
Dilated Resnet		11.53	19.8
Our other Implementations	Unet-Vggnet-BN	25.86	78.1
	Unet-Vggnet-DWS	3.47	14.8
	Unet_Resnet	30.12	177
	Fully Unet-Resnet	31.26	188
Proposed	Unet-Mobilenet	**5.53**	**4.74**

In summary, the proposed Unet-Mobilenet segmentation model, which is computationally efficient requiring less number of model parameters, has managed to produce good segmentation results without significant drop in accuracy.

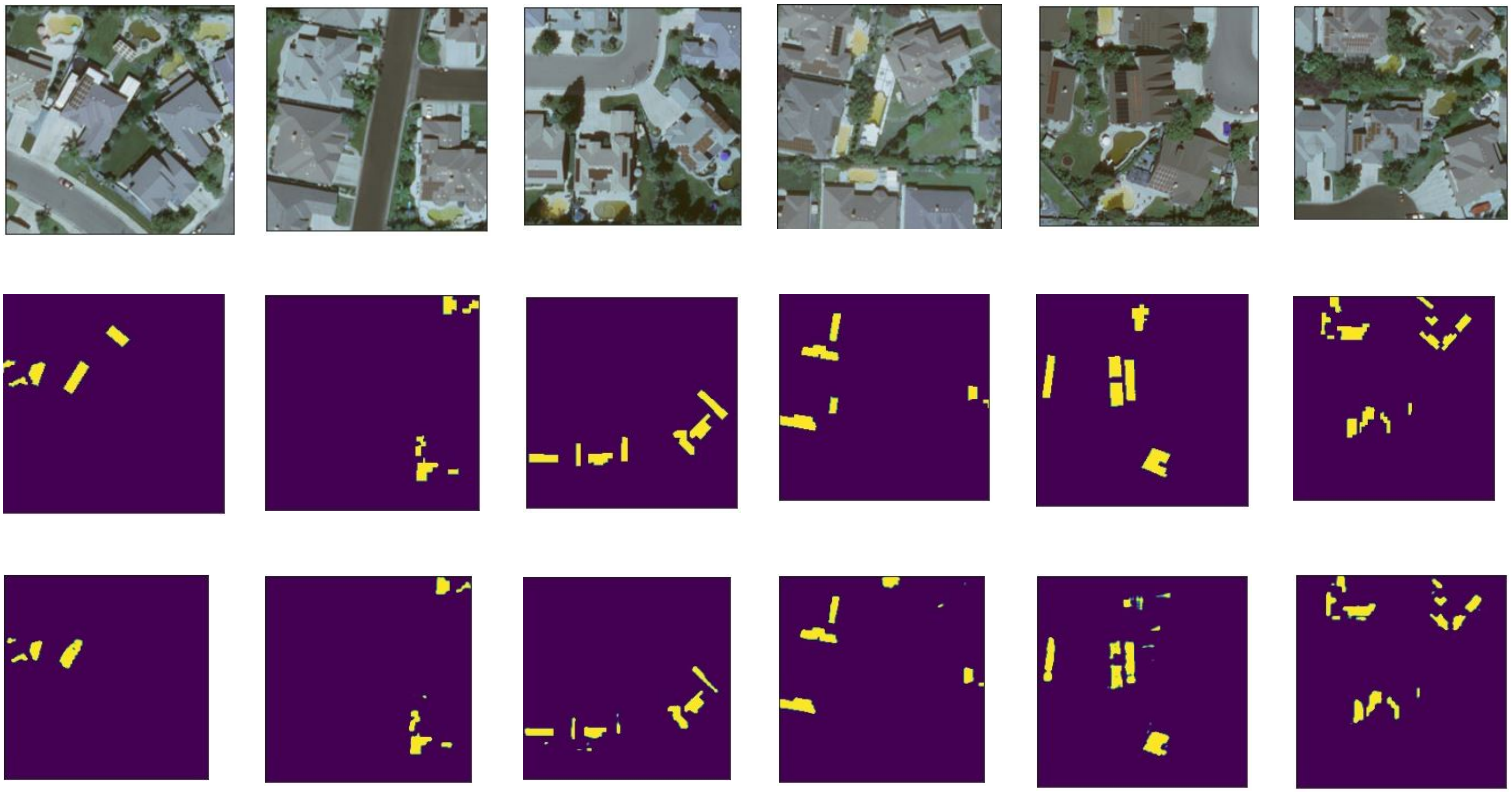
6. Conclusion

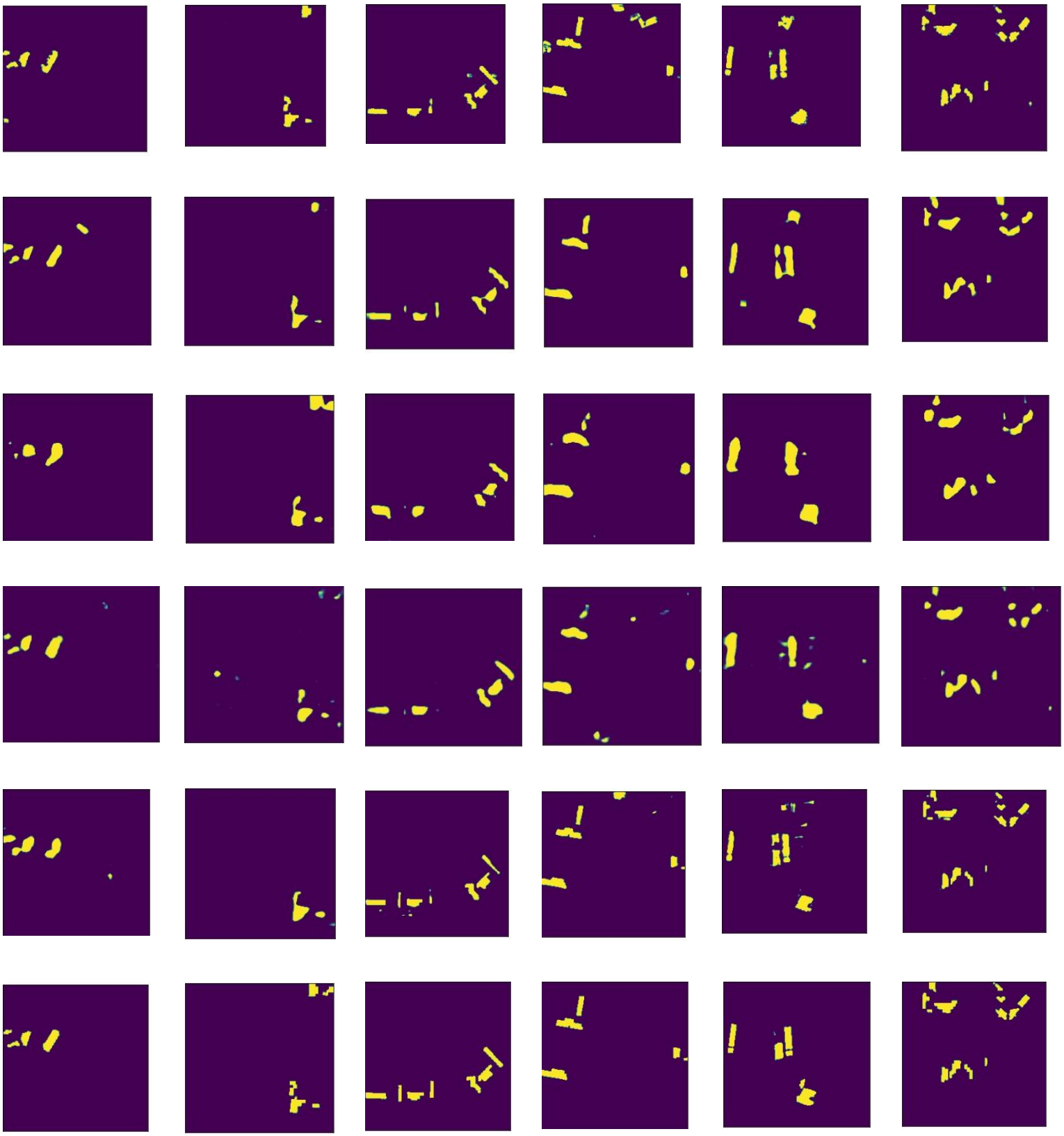
This paper presented an efficient deep learning segmentation architecture for detection of solar panels. The model was tested on satellite images of solar panels mounted on ground, buildings and roof tops. The training and testing datasets, and ground truth images were prepared by using the DCSA dataset. The proposed architecture was based on Unet and Mobilenet architectures. The use of depthwise separable

convolutions made the proposed model efficient in terms of training time, number of model parameters, and floating-point operations required while maintaining good segmentation accuracy. The proposed model produced accurate segmentation maps with accurate shapes and fine boundaries and had less false positives and false negatives. These features make the proposed model suitable for devices with limited computational power and for real time applications. Future work can include improving the architecture of the model for improving the results further.

Table 6. Comparison with Models used in the literature.

Study	Model Used	Dataset used	Recall	Precision	DSC
Yaun et al [24]	FCN	Satellite images of Boston [24]	0.840	0.812	*n.r**
		Satellite images of San Francisco [24]	**0.873**	0.855	*n.r**
Malof et al [25]	*n.r*	DCSA dataset [11]	0.77	0.76	0.76
		Satellite images of Connecticut [25]	0.83	0.88	0.85
Castello et al [21]	Unet	Satellite images of Switzerland [21]	*n.r**	*n.r**	0.8
Proposed Model	Unet-Mobilenet	DCSA dataset [11]	0.8498	**0.9595**	**0.9094**





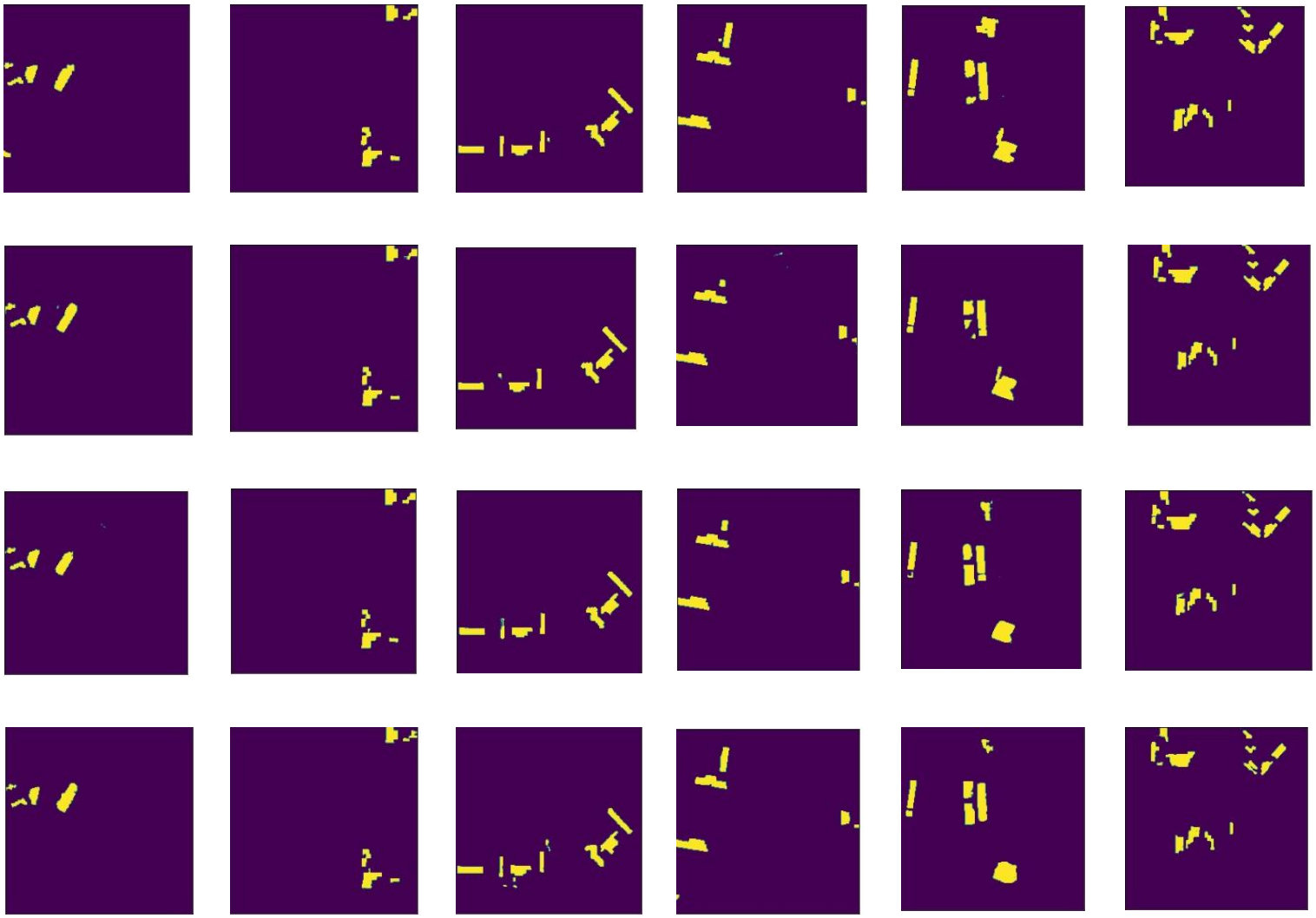


Fig. 5. The first row consists of RGB images from the test dataset, the second row consists of the corresponding masks for RGB images. Each row after that shows the solar panel shapes detected by different models in the following sequence: Unet, Segnet, Dilatednet, Pspnet, Deeplab v3+, Dilated Resnet, Unet-Vggnet-BN, Unet-Vggnet-DWS, Unet-Resnet, Fully Unet-Resnet, and Unet-Mobilenet.

References

[1] Asif Iqbal Khan, M. Arif Wani, "Patch-based segmentation of latent fingerprint images using convolutional neural network", Applied Artificial Intelligence, Vol. 33 (1), pp. 87-100, 2019.

[2] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, 2017, doi: 10.1109/TPAMI.2016.2572683.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9351, pp. 234–241, 2015, doi: 10.1007/978-3-319-24574-4_28.

[4] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017.

[5] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11211 LNCS, pp. 833–851, 2018, doi: 10.1007/978-3-030-01234-2_49.

[6] F. Yu and V. Koltun, "Multi-scale context

aggregation by dilated convolutions," *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, 2016.

[7] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 636–644, 2017, doi: 10.1109/CVPR.2017.75.

[8] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 6230–6239, 2017, doi: 10.1109/CVPR.2017.660.

[9] J. M. Malof, R. Hou, L. M. Collins, K. Bradbury, and R. Newell, "Automatic solar photovoltaic panel detection in satellite imagery," *2015 Int. Conf. Renew. Energy Res. Appl. ICRERA 2015*, vol. 5, pp. 1428–1431, 2015, doi: 10.1109/ICRERA.2015.7418643.

[10] J. M. Malof, K. Bradbury, L. M. Collins, and R. G. Newell, "Automatic detection of solar photovoltaic arrays in high resolution aerial imagery," *Appl. Energy*, vol. 183, pp. 229–240, 2016, doi: 10.1016/j.apenergy.2016.08.191.

[11] K. Bradbury *et al.*, "Distributed solar photovoltaic array location and extent dataset for remote sensing object identification," *Sci. Data*, vol. 3, no. December, pp. 1–9, 2016, doi: 10.1038/sdata.2016.106.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[14] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2261–2269, 2017, doi: 10.1109/CVPR.2017.243.

[15] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv Prepr. arXiv1704.04861*, 2017, [Online]. Available: http://arxiv.org/abs/1704.04861.

[16] M A Wani, "SAFARI: A structured approach for automatic rule induction", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 31(4), pp. 650-657, 2001.

[17] M. Arif Wani, Heena F. Bhat, "Multiclass SVM algorithms for wind speed prediction" Renewable Energy Research and Applications (ICRERA), 2017 IEEE 6th International Conference on, pp. 1139-1143, 2017.

[18] J. M. Malof, L. M. ; Collins, K. Bradbury, and R. G. Newell, "A deep convolutional neural network and a random forest classifier for solar photovoltaic array detection in aerial imagery," *2016 IEEE Int. Conf. Renew. Energy Res. Appl.*, pp. 650--654, 2016.

[19] J. M. Malof, L. M. Collins, and K. Bradbury, "A deep convolutional neural network, with pre-training, for solar photovoltaic array detection in aerial imagery," *Int. Geosci. Remote Sens. Symp.*, vol. 2017-July, pp. 874–877, 2017, doi: 10.1109/IGARSS.2017.8127092.

[20] J. Camilo, R. Wang, L. M. Collins, K. Bradbury, and J. M. Malof, "Application of a semantic segmentation convolutional neural network for accurate automatic detection and mapping of solar photovoltaic arrays in aerial imagery," 2018, [Online]. Available: http://arxiv.org/abs/1801.04018.

[21] R. Castello, S. Roquette, M. Esguerra, A. Guerra, and J. L. Scartezzini, "Deep learning in the built environment: Automatic detection of rooftop solar panels using Convolutional Neural Networks," *J. Phys. Conf. Ser.*, vol. 1343, no. 1, 2019, doi: 10.1088/1742-6596/1343/1/012034.

[22] J. Yu, Z. Wang, A. Majumdar, and R. Rajagopal, "DeepSolar: A Machine Learning Framework to Efficiently Construct a Solar Deployment Database in the United States," *Joule*, vol. 2, no. 12, pp. 2605–2617, 2018, doi: 10.1016/j.joule.2018.11.021.

[23] M. Arif Wani, "Introducing Subspace Grids to Recognise Patterns in Multidimensinal Data", International Conference on Machine Learning and Applications, Boca Raton, USA, IEEE publication, Volume 1, pp. 33-39, 2012.

[24] M. Arif Wani, "Microarray Classification using Sub-Space Grids", Proceedings of the Tenth International Conference on Machine Learning and Applications, Hawaii, USA, IEEE publication, Volume 1, pp. 389-394 , December, 2011.

[25] V. Golovko, A. Kroshchanka, S. Bezobrazov, A. Sachenko, M. Komar, and O. Novosad, "Development of Solar Panels Detector," *2018 Int. Sci. Conf. Probl. Infocommunications. Sci. Technol. (PIC S&T)*, pp. 761–764, 2018.

[26] J. Yuan, H. H. L. Yang, O. A. Omitaomu, and B. L. Bhaduri, "Large-scale solar panel mapping from aerial images using deep convolutional networks," *Proc. - 2016 IEEE Int. Conf. Big Data, Big Data 2016*, pp. 2703–2708, 2016, doi: 10.1109/BigData.2016.7840915.

[27] Mohd Rouf Wani, M. Arif Wani, and Romana Riyaz, "Cluster Based Approach For Mining Patterns To Predict Wind Speed", 5 th International Conference on Renewable Energy and Applications, Birmingham, U.K., pp. 1046-1050, 2016

[28] M. A. Wani, "Incremental Hybrid Approach for Microarray Classification", Proceedings of the Seventh International Conference on Machine Learning and Applications, San Diego, USA, IEEE publication, pp. 514-520 , December, 2008.

[29] J. M. Malof, B. Li, B. Huang, K. Bradbury, and A. Stretslov, "Mapping solar array location , size , and capacity using deep learning and overhead imagery," pp. 1–6, 2015.