Design of a high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning for crop analysis

Akshay Dhande (akshaydhande126@gmail.com)
Lovely Professional University
Rahul Malik
Lovely Professional University

Research Article

Keywords: Earth, Satellite, crop, type, damage, classification, deep learning, incremental, transfer, accuracy, precision, recall

Posted Date: August 3rd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1884264/v1

Design of a high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning for crop analysis

Akshay Dhande[1], Rahul Malik[2]
[1]*Department of Electronics and Telecommunication Engineering, Lovely Professional University, Punjab, India, akshaydhande126@gmail.com,* https://orcid.org/0000-0002-7340-9806
[2]*Department of Computer Science and Engineering, Lovely Professional University, Punjab, India,* https://orcid.org/0000-0002-2566-2954

Abstract: Satellite image processing is a multidomain task which involves design of image capturing, denoising, segmentation, feature extraction, feature reduction, classification, and post-processing tasks. A wide variety of satellite image processing models are proposed by researchers, and each of them have different data & process requirements. For instance, image capturing module Might obtain Images in layered form, while feature extraction module might require data in 2D or 3D forms. Moreover, performance of these models also varies due to changes in internal process parameters & dataset parameters, which limits their accuracy and scalability when applied to real-time scenarios. To reduce the probability of these limitations, a novel high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning is proposed & discussed in this text. The model initially captures real-time satellite data using *Google's Earth Engine* and processes it using a transfer learning-based convolutional neural network (CNN) via backscatter coefficient analysis. These coefficients indicate average intensity value of Precision Image (PRI) when evaluated over a distributed target. Due to extraction of backscattering coefficients, the model is capable of representing crop images in VV (vertical transmit, vertical receive), and HV (horizontal transmit vertical receive) modes. Thereby assisting the CNN model to extract a wide variety of features from input satellite image, which classifies these datasets (original, VV, and VH) into different crop categories. The classified images are further processed via an incremental learning layer, which assists in visual identification of affected regions. Due to use of incremental learning and CNN for classification, the proposed TRSAITL model is capable of achieving an average accuracy of 97.8% for crop type & severity of damage detection, with an average PSNR of 29.6 dB for different image types. The model was tested on different regions around our local geographical area, and consistent performance was observed. This performance was also compared with various state-of-the-art approaches, and it was observed that the proposed TRSATL model has 5% better accuracy, 4.6% better precision, and 7.9% better recall when compared with them, which makes it highly useful for real-time satellite-based crop classification applications.

Keywords: Earth, Satellite, crop, type, damage, classification, deep learning, incremental, transfer, accuracy, precision, recall

1. Introduction

Satellite image classification requires design of multiple image processing tasks that should be able to work with different bands of image data. These tasks include band-based pre-processing, band fusion, segmentation, feature extraction, feature selection, classification, and post processing. A typical satellite image processing model is depicted in figure 1, wherein images from multiple bands are stacked, and sequenced for patch extraction & sequence estimation.The extracted patches are given to a windowing layer which extracts different smaller components, for extraction of large feature sets. These feature sets are classified using a deep learning (DL) model, which uses combination of different convolutional Neural Networks (CNNs) for generation of an initial clustered map. This initial map is given to a post-processing layer, which assists in formulation of a refined map, that can assist in identification of different components of the area under test.

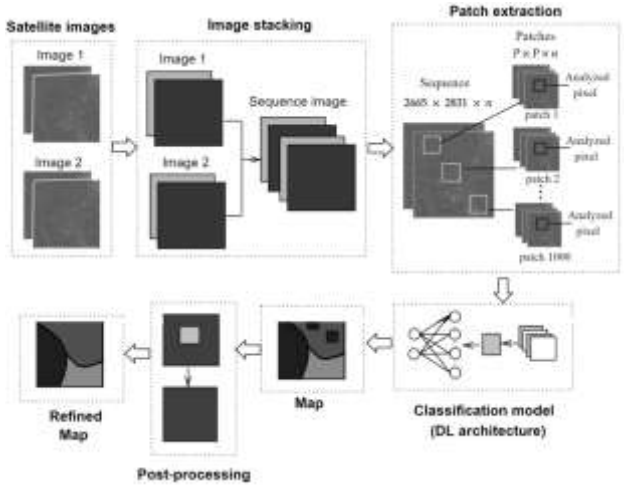

Figure 1. A typical satellite image processing model

These components can include, water cover, land cover, crop cover, type of crop, urban cover, etc. A wide variety of such classification models are proposed by researchers [1, 2, 3], and each of them vary in terms of classification accuracy, application, computational complexity, delay needed for classification, etc. Survey of such models, along with their nuances, advantages, limitations, and future research scopes is discussed in the next section of this text, which will assist readers to identify currently best performing models for stratification of satellite images. Based on this review, it is observed that these models are used for context-specific applications, which limits their accuracy & scalability for general purpose classification scenarios. To improve this performance, section 3 discusses design of a novel high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning for crop analysis. Performance of this model is evaluated in terms of delay, accuracy, precision, recall & area under the curve (AUC) measures, and compared in section 4 with various state-of-the-art approaches. Finally, this text concludes with some interesting observations about the proposed model, and recommends various methods to improve its performance.

2. Literature review

Multispectral image processing is one of the essential characteristics of satellite image processing systems. The work in [4, 5, 6] propose use of multiple sensors & multiple temporal image classification using hybrid CNN (HCNN) model, semantically enriched crop type classification models, and deep learning models for classification of different crop types. These models assist in improving efficiency of crop classification via redundancy reduction, and feature augmentation, which assists in enhancing classification performance. But the delay of these models is high, due to which they cannot be used for real-time classification applications. To reduce this delay, work in [7] proposes Pearson correlation (r) and Cohen's kappa coefficients for effective classification of crop types. This model has high accuracy, and can be used for real-time classification scenarios. Similar models are proposed in [8, 9, 10], wherein interferometric coherence with backscattering for timeseries classification, temporal stacking with vegetation indices, and 3D modelling for classification of unmanned aerial images are discussed. These models utilize statistical and temporal features from input imagery for training & validation of deep learning classifiers. They have good accuracy, but have higher delays due to requirement of continuous training & validation steps.

Models that assist in improving scalability while maintaining high accuracy are proposed in [11, 12, 13], wherein channel attention-based temporal convolutional Network, deep Neural Networks for multiple source remote and proximal sensing (MSRPS), and dense time series classification models are discussed. These models utilize various deep learning method which are suited for multiple band classifications, while reducing redundancies during feature extraction for different applications. Efficiency of these models is further improved via Convolutional Neural Networks with two sampling strategies (CNN TSS) [14], long-short-term memory (LSTM) [15], Multitemporal polarimetric deep learning [16], spatiotemporal deep learning Models [17], and Multitemporal deep learning models for Rice classification [18] are used. These models aim at improving feature representation via use of multiple layers of convolution, thereby maximizing feature variance, and minimizing error rate during classification operations. Extensions to these models are discussed in [19, 20, 21], wherein constrained distance-based clustering, synthetic aperture radar

(SAR) with optical remote sensing (ORS), and remote sensing observations are used for enhanced feature classification operations. These operations include crop classification, augmentation of features for damage estimation, and region marking for differential map observations. Accuracy of these models is high, but they have higher computational complexity, which can be reduced via the work in [22, 23, 24], wherein augmented deep learning, fusion of Geographic information with satellite images, and Machine learning Models for effective yield prediction are proposed. These models allow for lower complexity classifications via use of simplified feature extraction & reduction techniques. Similar Machine learning Models are discussed in [25, 26], for brown planthopper damage detection, and damaged vegetation areas classification applications. It is observed that these models are highly application specific, which limits their scalability for multiple application scenarios. To improve this scalability, next section proposes a high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning for crop analysis. Performance of the proposed model is compared with various reviewed models in terms of accuracy, precision, and recall parameters, for validation of the proposed method under different application scenarios.

3. Design of the proposed high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning for crop analysis

From the literature review, it can be observed that a wide variety of machine learning and deep learning models are available for satellite image classification. These models are applied to context-specific applications, which limits their scalability. To improve this scalability, a novel high-efficiency temporal engine for real-time satellite image classification using augmented incremental transfer learning for crop analysis is proposed in this section. The proposed TRSAITL model uses a combination of CNN with incremental learning to continuously improve classification accuracy. Overall flow of the proposed model is depicted in figure 2, wherein VV & VH components are extracted from Google Earth Engine, and then augmented via resizing, reshaping, rescaling, and sheering operations. The augmented images are classified via VGGNet-19 based CNN model, which assists in crop-type identification.

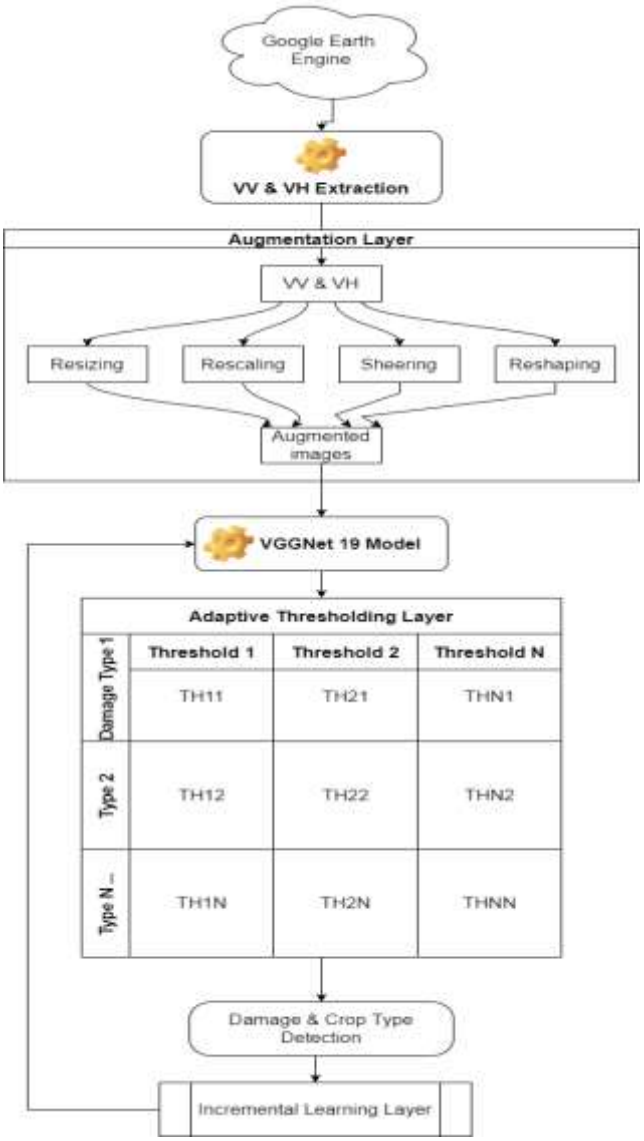

Figure 2. Overall flow of the proposed model

The crop-type identified tag, along with input image are given to a damage detection model, which assists in detection of damage severity via adaptive threshold engine. Results of the threshold engine are validated via an incremental learning layer, which assists in retuning of VGGNet model via correlation-based matching & hyperparameter tuning processes. To simplify the design discussions, the model design is divided into 3 different sub-parts, and each of these parts are discussed in different sub-sections of this text.

3.1. Design of VV & HV Extraction for creation of training set

The dataset is collected from a real-time Google Earth Engine which assists in accessing different temporal satellite sources. These sources were interpolated via multiple location-based data access, wherein current area was segregated into different sub-regions, and each region was processed via different transmitter receiver polarization. Extraction of these regions was done using the following process,

- Initialize a target location $Lat_{target}, Long_{target}$
- Extract images from this location via MODIS land cover collection, and COPERNICUS surface temperature collections.
- Extract VV and HV via equations 1 and 2 as follows,

$$VV_i = \frac{\sum_{i=1}^{NBands} B_i - B_{RGB}}{\sum_{i=1}^{NBands} B_i + B_{RGB}} \dots (1)$$

$$HV_i = \frac{\sum_{i=1}^{NBands} G_i - G_{RGB}}{\sum_{i=1}^{NBands} G_i + G_{RGB}} \dots (2)$$

Where, $NBands$ represents number of bands extracted from Google Earth Engine, G & B represents their green & blue components, and G_{RGB} & B_{RGB} represents green & blue components of RGB band.

- Once these values are extracted for each pixel, then VV and HV images are formed for current location.
- The current latitude and longitude are modified via equation 3, to extract a greater number of images from the current location,

$$New_{lat} = Lat_{target} \pm 0.1, New_{long} = Long_{target} \pm 0.1 \dots (3)$$

- For each of these latitude and longitude positions, VV and VH images are extracted to form the initial training set.

The training set is manually tagged, and different crop types & damage percentages are stored along with each image. This data was extracted from Indian Meteorological Dataset (https://mausam.imd.gov.in/), and can be scaled to any Global Geographical location for accurate development of training sets.

3.2. Augmentation layer with CNN model design

The collected dataset has limited number of images, due to which efficient training of CNN is not possible. To overcome this issue, an augmentation layer is activated, which rescales, resizes, sheers, and reshapes each of the input image. The rescaling process is performed via equation 4 as follows,

$$I_{scale} = \frac{I_{orig} - \min(I_{orig})}{\max(I_{orig}) - \min(I_{orig})} * S_F \dots (4)$$

Where, I_{orig}, I_{scale} represents original and scaled images, while S_F represents scaling factor, which is varied between 1 to 128 in steps of 2, for obtaining 64 different images per input image. The input image is further augmented using resizing operations, which is performed via equation 5 as follows,

$$I_{resize} = \bigcup_{i=1}^{N} \left| I_{orig_i} \right|_{i=i+r} \dots (5)$$

Where, r represents resizing factor, and is varied in the range of 0.1 to 10, in steps of 0.1 to obtain 100 different images per input image. Similarly, reshaping of image is performed via equation 6,

$$I_{reshape} = \bigvee_{r=1}^{R} \bigvee_{c=1}^{C} |I_{r+s_1, c+s_2}| \dots (6)$$

Where, $s_1, and\ s_2$ represents reshaping constants and are varied between 2 to 8 to obtain 64 different images from input image, while R, C represents number of rows, and columns in the input image. The sheering process is controlled via equation 7 as follows,

$$I_{sheer} = Q * \frac{I_{orig}}{C_{sheer}} \dots (7)$$

Where, Q, C_{sheer} represents quality factor & constant of sheering, which is varied between 1 to 8 for obtaining 64 different images for each input image. These augmented images are combined to obtain 292

images for each input image, and given to a VGGNet19 model. The extracted images are resized into 128x128 for initial convolution operations, which assist in augmented feature extraction. To assist in large-scale feature extraction, multiple layers of convolution, with different window sizes, stride sizes, and padding sizes are used. In this work, window sizes vary between 8x8 to 512x512, stride sizes vary between 3x3 to 5x5, and padding sizes vary between 3x3 to 5x5, which assists extraction of over 1 million features from each augmented image set. To select relevant features, the convolutional operations are controlled via an activation layer. For VGGNet-19, a leaky rectilinear unit (LReLU) is used, which enables variance-based feature extraction for better accuracy performance. The output features extracted by convolutional layers are controlled via equation 8 as follows,

$$F_{out} = Leaky_{ReLU}\left(\frac{r}{2}+a,\frac{c}{2}+b\right) * \sum_{a=0}^{\frac{r}{2}}\sum_{b=0}^{\frac{c}{2}} I_{aug}(i-a,j-b) \dots (8)$$

Where, I_{aug} is augmented satellite image for current layer, m,n are convolutional window sizes for given convolution layer, and a,b are sizes of strides that vary between 3x3, to 5x5 for each layer. The leaky RELU used in this case removes 5% of all low variance features, which assists in initial feature selection process. The Leaky ReLU is controlled via equation 9 as follows,

$$LReLU(x,y) = 0.05 * (x,y), when\ x < 0\ or\ y < 0$$
$$else, 1, when\ x \geq 0\ and\ y \geq 0 \dots (9)$$

Due to this variance evaluation, large-scale feature extraction with variance-based feature reduction is possible. Total number of features extracted & selected by this model are estimated via equation 10 as follows,

$$f_{conv} = \frac{2 * p - k + f_{prev}}{s} + 1 \dots (10)$$

Where, f_{prev} represents total number of features extracted by previous convolutional layer, p represents padding size of the leaky ReLU based convolutions, f_{conv} represents total number of convolutional features extracted by current convolutional layer, s is size of the stride, and k is the leaky ReLU kernel size for convolutions. But Leaky ReLU estimates variance based on a fixed threshold, which can be inefficient for a large sized dataset. To enhance feature selection capabilities, a maximum variance pooling (MaxPooling) layer is used. This layer estimates a variance-based threshold for each extracted feature set, and uses this variance threshold for selection of features. All features having intensity more than this threshold are passed to the next layer, else they are discarded at current layer. This variance threshold is evaluated via equation 11 as follows,

$$f_{th} = \sqrt[p_k]{\left(X_k^{-1}\sum_{x\in X_k} x^{p_k}\right)} \dots (11)$$

Where, X_k represents input image, and p_k represents the probability of pooling & controls number of features to be pooled at each layer. This probability of pooling is modified via hyperparameter tuning to achieve better feature selection performance. This process is repeated for different window sizes and stride sizes to estimate large number of features from input satellite images.

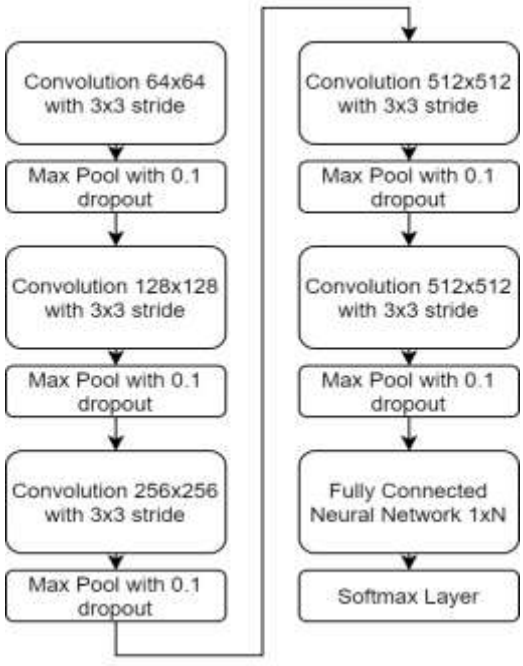

Figure 3. General purpose CNN Model used for classification of MRI data

These features are given to a fully connected Neural Network (FCNN) model as observed from figure 3 for classification of image into crop & damage severity types. The classification layer uses a Soft Max Activation Model for final class probability estimation, which assists in identification of crop & damage severity type. The Soft Max Activation Model has backpropagation capabilities which assists in continuous accuracy improvement. The probability of each output class is evaluated via equation 12 as follows,

$$c_{out} = SoftMax\left(\sum_{i=1}^{N_f} b + f_i * w_i\right) \dots (12)$$

Where, f_i represents extracted features by all previous convolutional and Max Pooling layers, w_i represents a tuned weight value for the given class, b represents bias value, and N_f represents number of features extracted via combination of different convolutional layers. These classes are used for adaptive threshold evaluation with parameter tuning as discussed in the next section of this text.

3.3. Adaptive thresholding layer with parameter tuning

Once the CNN Model evaluates damage severity & crop type, an adaptive thresholding layer is activated for final processing & continuous accuracy improvement.

Season -> Damage Type	Winter	Summer	Rainy	Autumn
Low Damage	$\frac{G}{R} > 0.6$ & $\frac{G}{B} > 0.5$	$\frac{G}{R} > 0.5$ & $\frac{G}{B} > 0.65$	$\frac{G}{R} > 0.6$ & $\frac{G}{B} > 0.6$	$\frac{G}{R} > 0.55$ & $\frac{G}{B} > 0.65$
Moderate Damage	$\frac{B}{R} > 0.2$ & $B > 0.5 * G$	$\frac{B}{R} > 0.6$ & $B > 0.6 * G$	$\frac{B}{R} > 0.8$ & $B > 0.85 * G$	$\frac{B}{R} > 0.75$ & $B > 0.85 * G$
High Damage	$R > 0.3 * B$ & $R > 0.35 * G$	$R > 0.45 * B$ & $R > 0.55 * G$	$R > 0.35 * B$ & $R > 0.65 * G$	$R > 0.75 * B$ & $R > 0.8 * G$

Table 1. Adaptive threshold evaluation for damage severity detection

The layer uses different thresholds for R, G, and B components for evaluation of different damage types depending upon type of season in which image is captured. These thresholds are tabulated in table 1. Once the images are classified into damage types, then a correlation between testing dataset, and their ground truth is evaluated via equation 13 as follows,

$$C_{train,test} = \frac{\sum I_{test} - I_{train}}{\sqrt{(\sum (I_{test} - I_{train})^2)}} \dots (13)$$

If this correlation value is above 0.999, then it indicates that the test set image closely matches one of the training set images. In such cases, this test set image is discarded, and not used for parameter tuning process. Other images are added to the training set, and the CNN model is retrained based on these images. Due to which the model is continuously upgraded, and its accuracy is incrementally improved. Evaluation of this accuracy performance, along with various other performance measures is discussed in the next section of this text.

4. Performance evaluation & comparison

The proposed TRSAITL model uses Google Earth Engine for collection of a wide variety of datasets from MODIS satellite data collections. From this data collection, the COPERNICUS subset is evaluated, and USGS SRTMGL1 ground elevations were used for dataset collection. This data was collected around the Amravati region (having latitude & longitude of 77.7523, 20.9320), for the years 2015 to 2021, and processed via the proposed model. A total of 2000 images were extracted, and divided into 70:15:15 ratio for training, testing and validation respectively. These images were extracted for the 4 crop types (Rice, Bajra, Cotton, and Wheat), and 3 different severity types. Based on this dataset collection, accuracy was evaluated via equation 14 as follows,

$$A = \frac{N_C}{N_T} * 100 \dots (14)$$

Where, N_c & N_T represents number of correctly classified images, and total number of images used for classification. This accuracy performance was compared with HCNN [4], MSRPS [12], & CNN TSS [14], and tabulated w.r.t. number of images used

for evaluation (NI) in table 2, wherein accuracy for different crop types was averaged to identify the final accuracy performance.

NI	A (%) HCNN [4]	A (%) MSRPS [12]	A (%) CNN TSS [14]	A (%) ADLRBSAS
130	73.23	65.56	83.94	96.67
260	73.90	66.17	84.72	97.22
400	74.06	66.30	84.90	97.52
530	74.37	66.58	85.25	97.81
660	74.51	66.71	85.42	97.97
800	74.60	66.78	85.51	98.02
930	74.60	66.78	85.51	98.03
1060	74.61	66.79	85.52	98.04
1200	74.61	66.80	85.53	98.04
1300	74.61	66.80	85.53	98.05
1500	74.62	66.80	85.54	98.05
1600	74.62	66.80	85.54	98.05
1730	74.62	66.80	85.54	98.06
1860	74.63	66.81	85.55	98.06
2000	74.63	66.81	85.55	98.06

Table 2. Accuracy of different satellite image processing models

From these results it can be observed that the proposed model is 12.5% better than CNN TSS [14], and at least 15% better than HCNN [4] & MSRPS [12] in terms of multiple domain accuracy performance. This is due to use of CNN with incremental learning, which combines different high-performance classification & adaptive thresholding to achieve better performance with lower error rates when compared to standard models. Similar observations are made for precision (P) values, and can be observed from the following table 3,

NI	P (%) HCNN [4]	P (%) MSRPS [12]	P (%) CNN TSS [14]	P (%) ADLRBSAS
130	80.42	71.03	84.03	85.44
260	81.15	71.67	84.81	85.92
400	81.32	71.82	84.99	86.18
530	81.66	72.12	85.34	86.44
660	81.83	72.27	85.52	86.58
800	81.92	72.35	85.61	86.62
930	81.92	72.35	85.61	86.63
1060	81.92	72.36	85.61	86.65
1200	81.93	72.36	85.62	86.65
1300	81.93	72.36	85.62	86.65
1500	81.93	72.38	85.62	86.65
1600	81.93	72.38	85.62	86.65
1730	81.93	72.38	85.62	86.66
1860	81.95	72.38	85.64	86.66
2000	81.95	72.38	85.64	86.84

Table 3. Average precision values for different algorithms

The proposed model is observed to be 4.9% more precise than HCNN [4], 14.2% more precise than MSRPS [12], and 1.5% more precise than CNN TSS [14], under different types of satellite images. Due to which, the proposed model is capable of being deployed for a wide variety of real-time high-precision applications. Similar observations are made for recall (R) values, and can be observed from the following table 4,

NI	R (%) HCNN [4]	R (%) MSRPS [12]	R (%) CNN TSS [14]	R (%) ADLRBSAS
130	79.41	70.14	83.00	90.86
260	80.15	70.79	83.76	91.38
400	80.31	70.94	83.93	91.66
530	80.64	71.22	84.27	91.94
660	80.79	71.37	84.44	92.08
800	80.88	71.45	84.54	92.12
930	80.88	71.45	84.54	92.13
1060	80.90	71.46	84.54	92.15
1200	80.90	71.46	84.56	92.15
1300	80.90	71.46	84.56	92.15
1500	80.91	71.48	84.56	92.15
1600	80.91	71.48	84.56	92.15
1730	80.91	71.48	84.56	92.16
1860	80.93	71.48	84.57	92.16
2000	80.93	71.48	84.57	92.36

Table 4. Average recall values for different algorithms

The proposed model is observed to be 10.2% efficient than HCNN [4], 20.5% efficient than MSRPS [12],

and 5.9% efficient than CNN TSS [14], under different types of satellite images. Due to which, the proposed model is capable of being deployed for a wide variety of real-time high-recall applications. Similar observations are made for area under the curve (AUC) values, and can be observed from the following table 5,

NI	AUC (%) HCNN [4]	AUC (%) MSRPS [12]	AUC (%) CNN TSS [14]	AUC (%) ADLRBSAS
130	70.88	63.45	81.24	93.57
260	71.53	64.04	82.00	94.09
400	71.68	64.17	82.17	94.38
530	71.97	64.44	82.51	94.67
660	72.12	64.56	82.67	94.82
800	72.20	64.64	82.76	94.87
930	72.20	64.64	82.76	94.88
1060	72.20	64.64	82.77	94.89
1200	72.21	64.65	82.78	94.89
1300	72.21	64.65	82.78	94.90
1500	72.22	64.65	82.78	94.90
1600	72.22	64.65	82.78	94.90
1730	72.22	64.65	82.78	94.91
1860	72.23	64.66	82.79	94.91
2000	72.23	64.66	82.79	95.62

Table 5. Average area under the curve (AUC) values for different algorithms

The proposed model has 22.5% more AUC than HCNN [4], 28.6% more AUC than MSRPS [12], and 12.4% more AUC than CNN TSS [14], under different types of satellite images. Due to which, the proposed model is capable of being deployed for a wide variety of real-time low-error applications. Similar observations are made for delay (D) values, and can be observed from the following table 6,

NI	D (ms) HCNN [4]	D (ms) MSRPS [12]	D (ms) CNN TSS [14]	D (ms) ADLRBSAS
130	9.39	10.63	8.98	6.83
260	9.30	10.53	8.90	6.77
400	9.28	10.51	8.88	6.76
530	9.24	10.47	8.84	6.73
660	9.22	10.44	8.83	6.71
800	9.22	10.43	8.82	6.72
930	9.22	10.43	8.82	6.93
1060	9.22	10.43	8.82	6.87
1200	9.22	10.43	8.82	7.13
1300	9.22	10.43	8.82	7.15
1500	9.22	10.43	8.82	7.11
1600	9.22	10.43	8.82	7.25
1730	9.22	10.43	8.82	7.27
1860	9.22	10.43	8.82	7.28
2000	9.22	10.43	8.82	7.29

Table 6. Average delay (D) values for different algorithms

The proposed model is 4.1% faster than HCNN [4], 5.9% faster than MSRPS [12], and 3.4% faster than CNN TSS [14], under different types of satellite images. Due to which, the proposed model is capable of being deployed for a wide variety of real-time high-speed applications. Figure 4 show the input satellite image and its segmentation output.

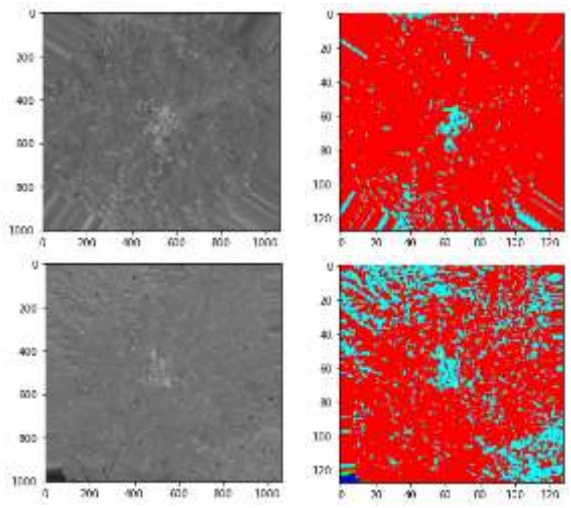

Figure 4. Input Satellite image and its segmented output

The reason for this delay improvement is use of incremental learning with effective feature extraction, which assists in better feature selection, thereby improving overall classification performance & speed for different satellite image classification applications.

5. Conclusion & Future work

The proposed model uses a combination of deep learning with augmented dataset collection & incremental learning in order to improve overall accuracy, precision, recall & AUC performance. Due to these characteristics, the proposed model was able to achieve 98.06% accuracy, 85.6% precision, 92.05% recall, and 95.6% AUC for different image types. The proposed model uses an incremental learning layer, which assists in continuous accuracy improvement via feature variance maximization. Because of this, the proposed model was observed to achieve a low delay of 6.85 ms across different image types, thereby making it useful for various high-speed application deployments. Upon comparison, it was observed that the proposed model was 12.5% better than CNN TSS [14], and at least 15% better than HCNN [4] & MSRPS [12] in terms of multiple domain accuracy performance. Similarly, proposed model is observed to be 4.9% more precise & better recall than HCNN [4], 14.2% more precise & better recall than MSRPS [12], and 1.5% more precise & better recall than CNN TSS [14], under different types of satellite images. Furthermore, the proposed model is 4.1% faster than HCNN [4], 5.9% faster than MSRPS [12], and 3.4% faster than CNN TSS [14], under different types of satellite images. In future, researchers can use Q-Learning, and adopt ensemble classification models for further improving accuracy & precision performance. Researchers can also validate performance of the proposed model on different datasets, and identify any lacunas in the current implementation. Once identified, these can be resolved via improved augmentation & deep learning classification models depending upon type of application scenarios.

Declarations

Conflict of interest

The authors declare that they have no conflict of interest.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Authorship Contributions

Akshay Dhande: Conceptualization, Methodology, Writing- Original draft preparation, Writing- Reviewing and Editing.

Rahul Malik: Investigation. Supervision, Validation.

Funding

Not Applicable

Availability of Data and Materials

Data available on request from the authors

6. References

[1] R. Luciani, G. Laneve and M. JahJah, "Agricultural Monitoring, an Automatic Procedure for Crop Mapping and Yield Estimation: The Great Rift Valley of Kenya Case," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 12, no. 7, pp. 2196-2208, July 2019, doi: 10.1109/JSTARS.2019.2921437.

[2] A. Shelestov et al., "Cloud Approach to Automated Crop Classification Using Sentinel-1 Imagery," in IEEE Transactions on Big Data, vol. 6, no. 3, pp. 572-582, 1 Sept. 2020, doi: 10.1109/TBDATA.2019.2940237.

[3] J. Jiang et al., "HISTIF: A New Spatiotemporal Image Fusion Method for High-Resolution Monitoring of Crops at the Subfield Level," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 4607-4626, 2020, doi: 10.1109/JSTARS.2020.3016135.

[4] Z. Li, G. Chen and T. Zhang, "A CNN-Transformer Hybrid Approach for Crop Classification Using Multitemporal Multisensor Images," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 847-858, 2020, doi: 10.1109/JSTARS.2020.2971763.

[5] M. Rousi et al., "Semantically Enriched Crop Type Classification and Linked Earth Observation Data to Support the Common

Agricultural Policy Monitoring," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 529-552, 2021, doi: 10.1109/JSTARS.2020.3038152.

[6] Z. Sun, L. Di, H. Fang and A. Burgess, "Deep Learning Classification for Crop Types in North Dakota," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 2200-2213, 2020, doi: 10.1109/JSTARS.2020.2990104.

[7] J. Quiros Vargas, L. R. Khot, R. T. Peters, A. K. Chandel and B. Molaei, "Low Orbiting Satellite and Small UAS-Based High-Resolution Imagery Data to Quantify Crop Lodging: A Case Study in Irrigated Spearmint," in IEEE Geoscience and Remote Sensing Letters, vol. 17, no. 5, pp. 755-759, May 2020, doi: 10.1109/LGRS.2019.2935830.

[8] A. Mestre-Quereda, J. M. Lopez-Sanchez, F. Vicente-Guijalba, A. W. Jacob and M. E. Engdahl, "Time-Series of Sentinel-1 Interferometric Coherence and Backscatter for Crop-Type Mapping," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 4070-4084, 2020, doi: 10.1109/JSTARS.2020.3008096.

[9] W. Khan et al., "On the Performance of Temporal Stacking and Vegetation Indices for Detection and Estimation of Tobacco Crop," in IEEE Access, vol. 8, pp. 103020-103033, 2020, doi: 10.1109/ACCESS.2020.2998079.

[10] P. W. Khan, Y. -C. Byun and M. A. Latif, "Clifford Geometric Algebra-Based Approach for 3D Modeling of Agricultural Images Acquired by UAVs," in IEEE Access, vol. 8, pp. 226297-226308, 2020, doi: 10.1109/ACCESS.2020.3045443.

[11] P. Tang, P. Du, J. Xia, P. Zhang and W. Zhang, "Channel Attention-Based Temporal Convolutional Network for Satellite Image Time Series Classification," in IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022, Art no. 8016505, doi: 10.1109/LGRS.2021.3095505.

[12] M. H. Asad and A. Bais, "Crop and Weed Leaf Area Index Mapping Using Multi-Source Remote and Proximal Sensing," in IEEE Access, vol. 8, pp. 138179-138190, 2020, doi: 10.1109/ACCESS.2020.3012125.

[13] Y. T. Solano-Correa, F. Bovolo, L. Bruzzone and D. Fernández-Prieto, "A Method for the Analysis of Small Crop Fields in Sentinel-2 Dense Time Series," in IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 3, pp. 2150-2164, March 2020, doi: 10.1109/TGRS.2019.2953652.

[14] S. Liu, Z. Zhou, H. Ding, Y. Zhong and Q. Shi, "Crop Mapping Using Sentinel Full-Year Dual-Polarized SAR Data and a CPU-Optimized Convolutional Neural Network With Two Sampling Strategies," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 7017-7031, 2021, doi: 10.1109/JSTARS.2021.3094973.

[15] M. M. G. de Macedo, A. B. Mattos and D. A. B. Oliveira, "Generalization of Convolutional LSTM Models for Crop Area Estimation," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 1134-1142, 2020, doi: 10.1109/JSTARS.2020.2973602.

[16] C. Silva-Perez, A. Marino, J. M. Lopez-Sanchez and I. Cameron, "Multitemporal Polarimetric SAR Change Detection for Crop Monitoring and Crop Type Classification," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 12361-12374, 2021, doi: 10.1109/JSTARS.2021.3130186.

[17] S. Yang, L. Gu, X. Li, F. Gao and T. Jiang, "Fully Automated Classification Method for Crops Based on Spatiotemporal Deep-Learning Fusion Technology," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-

16, 2022, Art no. 5405016, doi: 10.1109/TGRS.2021.3113014.

[18] H. -W. Jo et al., "Deep Learning Applications on Multitemporal SAR (Sentinel-1) Image Classification Using Confined Labeled Data: The Case of Detecting Rice Paddy in South Korea," in IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 11, pp. 7589-7601, Nov. 2020, doi: 10.1109/TGRS.2020.2981671.

[19] T. Lampert, B. Lafabregue, T. -B. -H. Dao, N. Serrette, C. Vrain and P. Gançarski, "Constrained Distance-Based Clustering for Satellite Image Time-Series," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 12, no. 11, pp. 4606-4621, Nov. 2019, doi: 10.1109/JSTARS.2019.2950406.

[20] J. Bell, E. Gebremichael, A. Molthan, L. Schultz, F. Meyer and S. Shrestha, "Synthetic Aperture Radar and Optical Remote Sensing of Crop Damage Attributed to Severe Weather in the Central United States," IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019, pp. 9938-9941, doi: 10.1109/IGARSS.2019.8899775.

[21] S. Sawant, J. Mohite, M. Sakkan and S. Pappula, "Near Real Time Crop Loss Estimation using Remote Sensing Observations," 2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics), 2019, pp. 1-5, doi: 10.1109/Agro-Geoinformatics.2019.8820217.

[22] S. Jones and J. Saniie, "Using Deep Learning and Satellite Imagery to Assess the Damage to Civil Structures After Natural Disasters," 2019 IEEE International Conference on Electro Information Technology (EIT), 2019, pp. 189-193, doi: 10.1109/EIT.2019.8833724.

[23] Y. Sofue, C. Hongo, N. Manago, G. Sigit, K. Homma and B. Barus, "Estimation of Normal Rice Yield Considering Heading Stage Based on Observation Data and Satellite Imagery," 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, 2021, pp. 6439-6442, doi: 10.1109/IGARSS47720.2021.9554679.

[24] G. Sravan Kumar, S. Venkatramaphanikumar and K. Venkata Krishna Kishore, "Smart Farming - A Flexible Approach to Improve Crop Yield and Profit using Machine Learning Techniques," 2021 2nd International Conference for Emerging Technology (INCET), 2021, pp. 1-6, doi: 10.1109/INCET51464.2021.9456433.

[25] D. Lakmal, K. Kugathasan, V. Nanayakkara, S. Jayasena, A. S. Perera and L. Fernando, "Brown Planthopper Damage Detection using Remote Sensing and Machine Learning," 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), 2019, pp. 97-104, doi: 10.1109/ICMLA.2019.00024.

[26] S. Haque, N. Rahman and M. Mostakim, "Classification Of Damaged Vegetation Areas Using Convolutional Neural Network Over Unlabelled Sentinel-2 Images," 2021 26th International Conference on Automation and Computing (ICAC), 2021, pp. 1-7, doi: 10.23919/ICAC50006.2021.9594269.