DISASTER MONITORING AND MANAGEMENT USING DEEP LEARNING VGI

Abstract

Satellite imagery has fascinated the humankind from a long time. It has been used to study landscapes and view different regions of world where man could not reach and even provide surveillance for security purposes. Though the use of satellites has never been underestimated but the multimedia revolution has completely changed the scenario, for example the discovery of image processing helped us utilize the blurred images from the Hubble telescope and the use of object detection helped us train robots. The scope of image utilization is vast, and the areas of study are wide, for this research we try to understand and explain a similar topic that could potentially help us do good. The following study tries to explain a disaster monitoring model that can identify different landscapes based on input pictures using different deep learning algorithms and image processing techniques. To help build this project, images from various open sources have collected and stored into different categories to serve as dataset. Results from these different models have been then analysed and compared to gain the maximum possible accuracy. The study also suggests a potential future scope for the project and its disaster management capabilities.

Keywords: - Land, water and plain classification, convolutional neural network

I. Introduction

The amount of data available nowadays is varied and vast. This product of the modern word has proven itself to be a blessing in various fields of work. Using deep learning algorithms to train algorithms and machines has benefited all of Humankind and even saved lives. All of these was made possible by the advancement in technologies to handle, understand and utilize the data available.

Using graphic data to achieve similar goals has been on the rise from around a decade. The work done using such type of data ranges from detecting Urban Changes, humanitarian mapping to touching the lives of millions via its uses in google maps etc. While the uses of graphic information in corporate is a Billion Dollar industry, its benefits for humanitarian and disaster management work are an emerging idea.

Using Volunteer Graphic information to map out urban settlements has been widely discussed and researched upon in various formats. This paper proposes a way to use human settlement mapping along with disaster monitoring applications to make disaster management much more efficient.

The proposed model would use the concepts of CNN to classify different landscapes and human settlements using the study proposed by Kenji Ose under LULC classification, along with feature vectors from highly trained models of Alex Net, VGG-16, and inception models etc, to improve its accuracy for humanitarian and general purposes. The suggested model would ingest data collected from various sources like Open Street Map (Osm), Map Swipe, Google Earth etc. to learn and then later identify settlements according to their varying densities around the world.

II. Literature review

VGI has been around for a long time and has been studied and analysed by various individuals and teams to learn about various concepts and build models for various purposes.

Andrea Ballatore *et al* studied the semantic similarity in Open-Street Map and proposed the extraction of Geographic knowledge using OSM Semantic Network.[3] Later YING ZHANG *et al* came up with an Automatic approach to Extract Geographic Information from Internet using web search engines.[7] Moving forward Daniel Leung et al, suggested classification of land cover using geo-referenced photos with the help of SVMs to classify individual images.[8] Later Taili Wan *et al,* proposed the classification of High-Resolution Remote-Sensing Image using OpenStreetMap Information with the help of OSM RSS to implement road super pixel segmentation.[6] Further more Dino Ienco *et al* , combined Satellite Imagery and VGI for Urban LULC classification using simple CNN models which shows that VGI data (i.e. highway from OpenStreetMap) can be used as additional information source to cope with simple urban land-use mapping.[1] Similarly Fang *et al* , showed Urban Land-Use classification from Photographs using hierarchical street networks which eased the process to update urban maps with the help of simple deep learning models.[9] Srinivisan et al also classified overhead water tanks for humanitarian mapping, by using Transfer Learning in land use classification they combined Fast RCNN and inception version 3 models to achieve a higher mean average precision (mAP). [5] Deepank Verma used LULC classification method which is based on simple Convolutional Neural Network to map complex urban forms at finer scale using CNN model based upon LCZ schemes.[2] Ding qi Yang *et al* provided real time assistance in disaster relief by the means of crowdsourcing power for humanitarian purpose which resulted in CDSP, a crowd sourcing disaster support platform. [10] Jiao yan Chen et al, explained a Case Study of Humanitarian Mapping that automatically label satellite images for humanitarian mapping by learning deep CNNs from multiple heterogeneous VGI data, including OSM, Map Swipe, and OsmAnd GPX. The proposed deep learning framework called MC-CNN, which includes a customized loss calculation approach and an active learning-based sample fusion algorithm, resulted in fast efficient and improved result. [4]

III. Proposed Methodology

With the mentioned dataset we propose to build and develop a collection of varied models that help us achive our goal of Disaster Monitoring and Management. We plan to assemble and observe the model accurarcy for various methods like VGG16, Resnet 50, Inception-V3 etc in order to classify our dataset images into 3 categories with the best acquired accuracy. Various components like different activation functions and numerous libraries are comprised to help us achieve our goal. Activation function like Rectified Linear Unit (relu), softmax , sigmoid are used to train deep learning models.

a) Data Preprocessing

In order to make use of the data, initial preprocessing was done to help our models better understand and comprehend the information available.The proposed dataset comprised of images with varied shapes and sizes, in order to introduce uniformity the images of our dataset were resized in a consistent size of 512x512. To help categorize our data 3 different labels were introduced to denote Water bodies, Plain land and urban areas repectively. Along with that, to make our machines understand the different category labels, the defined labels were encoded into 0,1 and 2. To make our dataset more compatible with our aim, various Images processing techniques were applied on the images such as :-

1. Morphological Opening

The images in our dataset are manually classified by us, therefore the boundaries of the object which are to be classified may be inconsistent with the actual marked labels. Hence, morphological opening is employed to remove small objects from the images while preserving the shape and size of the large objects which is one of the main operations under the morphological operations of images. Morphological opening is achieved by first eroding the image and then dilating it. So it removes any narrow connections and lines between 2 regions. The mathematical equations for the opening operations is -

The opening operation of input image *S* by structural element *E* is defined by –

$$(So, Ex=(S \ominus Ex) \oplus Ex) \quad (1)$$

where Θ is the erosion operator, $\oplus$ is the dilation operator and *Ex* represents a structural element *E* with *x* as the origin.

2. Super Pixel Segmentation

Image segmentation technique makes the image more meaningful and easier to study. There are various ways for image segmentation, in this model we have used super pixel segmentation(SPS). Super pixels can be defined as group of pixels that share common characteristics. Super pixels carry more information and are less computationally intensive for deep learning models. There are 4 ways to apply super pixels, Felzenszwalb's method, SLIC, Quickshift and Compact watershed method. In this letter we have applied SLIC (Simple Linear Iterative Clustering) method on our images. This algorithm generates superpixels by clustering pixels based on their color similarity and closeness to the image plane. It performs K-means in the 5d space of color information and image location. It takes 3 inputs which are image, compactness and number of segments. The compactness parameter trades off color-similarity and proximity, while n_segments choose the number of centres for k means.

Algorithm 1 Efficient superpixel segmentation

Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling pixels at regular grid steps S.
Perturb cluster centers in an $n \times n$ neighborhood, to the lowest gradient position.
repeat
  for each cluster center C_k **do**
    Assign the best matching pixels from a $2S \times 2S$ square neighborhood around the cluster center according to the distance measure (Eq. 1).
  end for
  Compute new cluster centers and residual error E {$L1$ distance between previous centers and recomputed centers}
until $E \leq$ threshold
Enforce connectivity.

Fig 1 :- Super Pixel Segmentation Algorithm

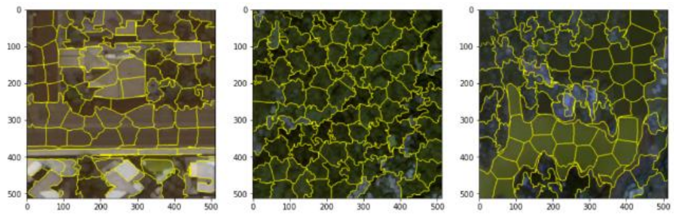

Fig 2 :- SPS Sample Output

b) Data Augmentation

To further increase the scope of our model several data augmentation techniques were applied. Where different aspects of the images were altered like the brightness of the image was increase/decreased according to its visibility, image width/height were changed according to the uniformity and original images were flipped/rotated to create a copy of images from different angles.

c) Model Building

To improve the feature extraction qualities of our models we have based our project on Transfer Learning Concept (TLC). To support TLC in our project, learnable parameters of a machine learning called "weights", are used. The weights of models used are trained on ImageNet Dataset. To inculcate the pre trained weights in our existing models the last layer i.e., the output layer of each weighted model is switched off and replaced by the new layers such as Dense, Dropout, Pooling etc, according to the needs of our project aim. Extensive hyperparameter tuning has been done for each weighted model to ensure our models achieves highest possible accuracy. To execute the above-mentioned methodology 4 different Deep Learning Models namely –

1. VGG 16
2. ResNet 50
3. Inception V3
4. InceptionResNet V2

Have been implemented. For further(Morphological)

1. VGG 16

Oxford Net or VGG 16 is a neural network based apon CNN architecture. It emerged in the ImageNet Large Scale Visual Recognition Challenge in 2014 as a winner and has about 16 layers with 138 million parameters that makes it a potential model for the solution that we aim to seek. Pre trained VGG 16 model with auto tuned parameters available from Keras library was altered and used according to the project need. The output layer of the imported VGG16 model was switched off and flattened to join 3 more layer of our dataset model which included a dense layer of 256 nodes followed by a dropout layer of step size of 0.2 and a second dense layer with 128 nodes. The input layer of the altered model was set with an input size of 512x512x3 to match the size of our dataset images and the output layer was fitted with SoftMax activation function and consisted of 3 nodes namely, water bodies, plain land and urban areas connected to the last dense layer. The model was compiled with categorical cross entropy loss function and Stochastic Gradient Descent optimizer. To retain results and compare the accuracy 2 case scenarios were executed with and without fine tuning on the VGG16 parameters, where the added weights of the models were updated in one case and original values were used on the other. The case scenarios were tested on the training dataset.

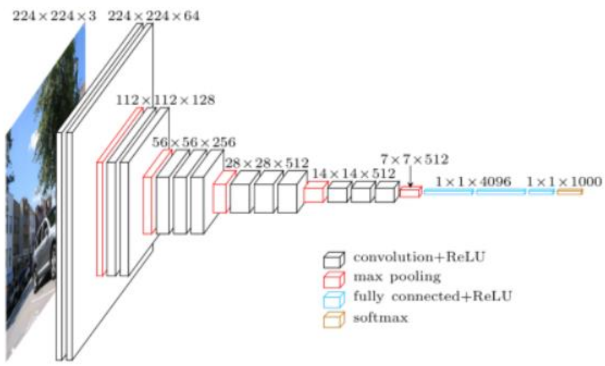

Fig3 :- Architecture of VGG 16

2. ResNet 50

After AlexNet won the Image Net 2012 competition, every model further built used more layers so as the error rate can be reduced. But with increasing layers another problem arose which was of vanishing and exploding gradients. To solve that problem Residual Networks were build which used the idea of skip connections. ResNet50 was one such model build using that idea. It is a variant of Res Net model which has 48 Convolution layers along with 1 Max Pool and 1 Average Pool layer, making it a total of 50 layers. A pre-trained version of the network is available on the internet which is trained on a million images of ImageNet dataset. This trained version and its weights were imported using Keras lib and the output layer was switched off to be replaced by new layers. Before adding the new layers, which are required to train the model, the input size is changed to 512x512x3 which is according to the images in our dataset and the output of the Resnet50 model is flattened. The output after flattening is given to a Dense layer with 256 nodes and is followed by a Dropout layer with a step size of 0.2 and another Dense layer of 128 nodes. Each of the Dense layer has the activation function as relu. After adding the required layers the final output layer is added which contains the number of nodes equal to the number of classes in our dataset i.e. 3, namely- Water bodies, Plain Lands and Urban areas. The activation function used in the output layer is set as SoftMax. Similar 2 scenarios are run in this model as the VGG 16, one with fine tuning and another without fine tuning. The model is then compiled with the categorical cross entropy as loss function and Adam optimizer. After compiling, the array obtained from the super pixel segmentation on training images is fitted on the model. The model is then evaluated using preprocessed validation data.

Fig 4 : - Architecture of ResNet 50

3. Inception V3

Inception V3 or I-V3 is a CNN architecture that made significant improvement in the field of CNN classifiers. I-V3 belongs to the constant evolution of the Inception models network that made several improvements within each version in terms of speed and accuracy like – Label smoothing, factorised 7x7 convolutions ,etc. Some of the better known version from the inception network are Inception V1,V2 ,V3 (used in this scenario) and Inception Resnet which is also used in the project later on. To begin using the features and benefits of IV3 for our project , the pre trained inception library is imported along with its node weights. The output layer is altered and flattened like the earlier models for the same reasons as before and additional customised layers are attached to gain desired results. To better understand the work of such a model different test case were implemented with and without fine tuning. To improve the accuracy and get fine detailed results in built features of IV3 like label smoothing which is a type of regularizing component added to the loss formula that prevents the network from becoming too confident about a class and prevents over fitting are also introduces along with RmsProp optimizer and categorical cross entropy loss function. The results of these particular observation are then noted to be compared in the later stages of this study.

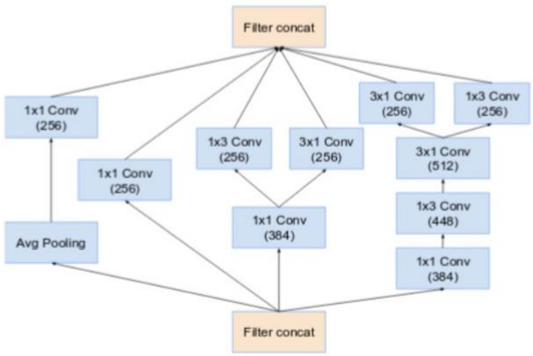

Fig 5 : -Architecture of Inception V3

4. InceptionResNet-V2

Inception and Resnet have been major turning points in the deep learning history. The pre-trained version of these models have achieved very good accuracy and have been computationally efficient. Inception-Resnet-V2 combines these 2 powerful architectures so that the extraordinary qualities of these models can be merged. It is a convolutional network which is also trained on the ImageNet dataset. The network is 164 layers deep and can classify images into 1000 categories. It is designed with a combination of concepts of inception and residual networks. Multiple convolutional blocks are combined with the residual connections. The usage of residual connections not only avoids the degradation problem caused by deep structures but also reduces the training time. The pre-trained version of the model is downloaded from keras for feature extraction and the output layer of the model is switched off to add new layers. The input size is given as 512x512x3 and the output of the model is flattened. New dense layers with nodes of 428 and 214 with activation functions as tanh are added. Finally an output layer is added with number of nodes as 3 and activation as softmax. Similar to previous 3 models, 2 scenarios of fine-tuning are considered and accuracy is compared between them. The model is compiled with loss function as categorical cross entropy and optimizer as Adam. After compiling, the model is fitted with the training images and is evaluated using the validation data.

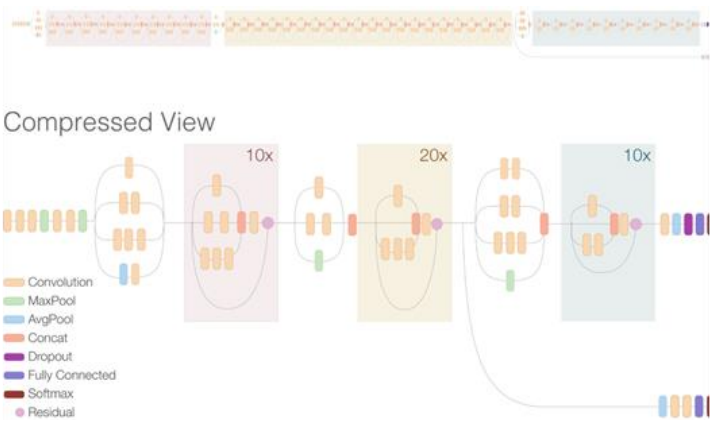

Fig 6:- Architecture of InceptionResNet – V2

IV. Experimental Setup

A self-made data set comprising of more than 1100 google earth satellite images from different demographics has been used to achieve the model goals of disaster monitoring and management. The dataset contains images from different parts of the world to increase the scope of our results and model accuracy. The combined dataset is then divided into 3 categories i.e., Urban areas, water bodies and plain lands to help our model identify all main landscapes. To make our model more time efficient all images in the dataset are saved in jpg format. The whole dataset is then again divided into test, train, and validation folders to feed our model in its different stages of development. The dataset is divided in the ratio of 9:1 where about 90% of the data which comprises of about 1000 images is used for training our model and 5% is used for testing and validation which sums up to be 50 images each.

The model is based on python programming language, which is built using various tools like google collaborator, anaconda, PyCharm etc. Many libraries were also used to implement and develop the model like Pandas, NumPy, cv2, keras, TensorFlow etc. To diversify our dataset even more we have applied various data augmentation techniques like alteration in brightness, width, height, image flip and rotation. To get the desired outcome different models like VGG16, Resnet 50, Inception- V3 and InceptionResNetV2 are used to obtain and compare the best accuracy and identify the best fit model .

Fig 7 :- Water Body

Fig 8 :- Plain Land Image

Fig 9 : - Urban Area

Fig 10 : - Forest

V Results

To assess the performance of the proposed models, numerous test cases were executed, observed, and compared that are discussed below. The results form these test cases are the conclusion of various outputs that helped in altering and improving our model according to the need of our project. The accuracy and loss value from training, validation, and testing from all the models with and without fine tuning is explained and stated as follows: -

VGG 16	
Without Fine Tuning	
Training Acc/Loss:	87.2%/0.38
Validation Acc/Loss:	92.6%/0.63
Testing Acc:	78.2%
With Fine Tuning	
Training Acc/Loss	87.2%/0.49
Validation Acc/Loss	85.2%/0.70
Testing Acc	68.9%
INCEPTION	
Without Fine Tuning	
Training Acc/Loss	63.09%/5.28
Validation Acc/Loss	59.26%/7.64
Testing Acc	55.42%
With Fine Tuning	
Training Acc/Loss	62.5%/5.58
Validation Acc/Loss	72.2%/4.52
Testing Acc	61.3%
RESNET 50	
Without Fine Tuning	
Training Acc/Loss	85.81%/0.79
Validation Acc/Loss	92.59%/0.53
Testing Acc	81.73%
With Fine Tuning	
Training Acc/Loss	88.5%/0.95
Validation Acc/Loss	88.9%/.032
Testing Acc	78.32%
INCEPTION RESNET V2	
Without Fine Tuning	
Training Acc/Loss	42.7%
Validation Acc/Loss	41.1%
Testing Acc	48.9%
With Fine Tuning	
Training Acc/Loss	42.4%
Validation Acc/Loss	44.4%
Testing Acc	44.7%

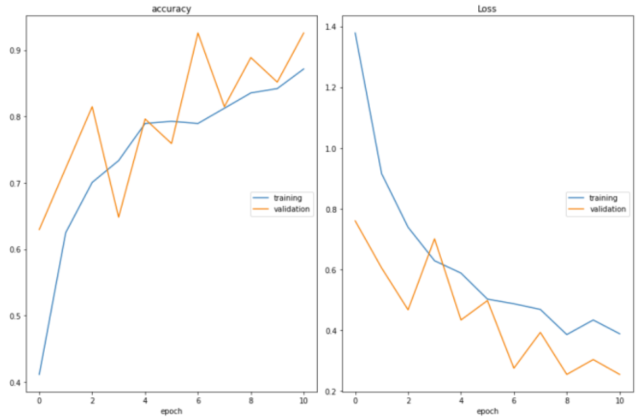

FIG 11 : - VGG16 Live Loss Plot without Fine Tuning

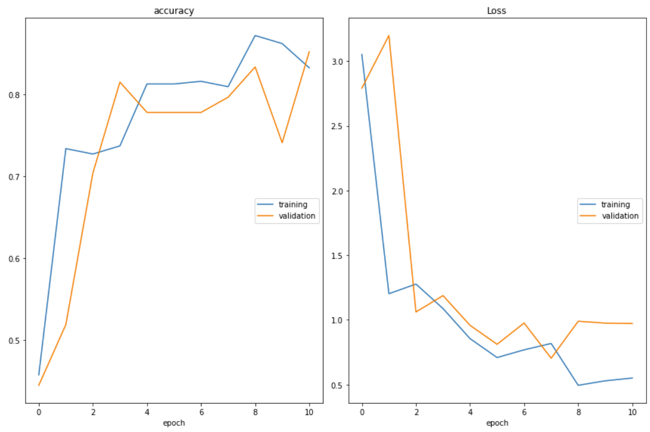

FIG 12 : - VGG16 Live Loss Plot with Fine Tuning

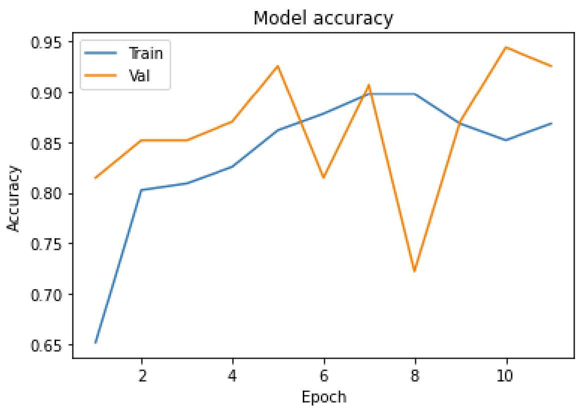

FIG 13:-Resnet 50 Live Acc Plot without Fine Tuning

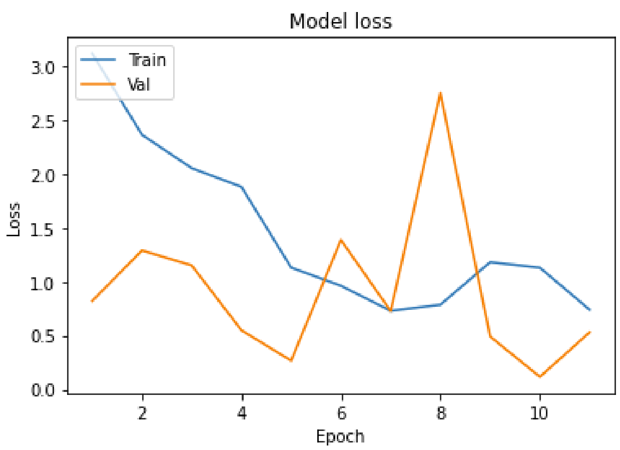

FIG 13:-Resnet 50 Live Loss Plot without Fine Tuning

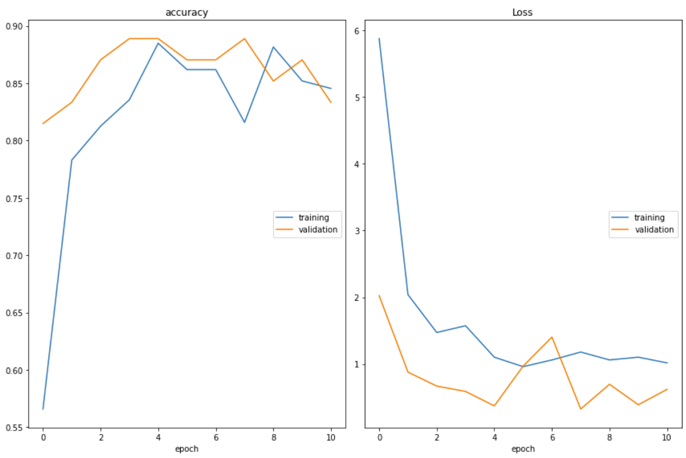

FIG 13:-Resnet 50 Live Loss Plot with Fine Tuning

The above stated results indicate that RESNET 50 without fine tuning performed the best amongst the models with an accuracy of 81.73% followed by VGG 16 with a model accuracy of 78.2%. The models in which fine tuning was not activated performed better as compared to the models with fine tuning. It can be concluded that the number of layers and parameters in our model are inversely proportional to the model accuracy. The result analysis shows that our model is capable of labelling images from different landscapes efficiently. It reduces the amount of human efforts required to label images manually, providing fast and accurate results.

VI. Conclusion

The above study presented a model to identify different landscapes on the user entered pictures using various image processing techniques and deep learning models. Various high state of the art convolutional neural networks was used to analyse and study the input data to label them correctly. The study used the data form various landscapes around the globe and included images from various majors' cities as well to increase the scope identify and label data form different sources at the same time. The project has successfully identified and labelled images with an accuracy above 80% which indicates its effectiveness. While image data is widely used for social media platforms or military use, the study conveys a way to use them for humanitarian work too and though the objective of this model is purely based upon humanitarian purposes, the strategy presented can also be implemented to various problems like disaster management, landscape mapping, urban planning etc. with simple modification in the implementation.

VII. Future Scope

Application of image analysis are immense and wide. This study merely presents a starting point for what could be achieved with the technology and data we have. For further development and advantages the project can be altered and fitted with a live time image sensing server that could help us in real time surveillance of high-risk areas. Ingesting such a large amount of data would require a more powerful model which could then self-learn using unsupervised learning and increase its results considerably using the ever-increasing data. The project apart from humanitarian needs can also be used in corporate sector like infrastructure, mining, power production, etc. providing the users with an upper edge in their field whilst being conscious of the natural habitats.

References

[1]. Ienco, Dino, Kenji Ose, and Christiane Weber. "Towards combining satellite imagery and VGI for urban LULC classification." *2019 Joint Urban Remote Sensing Event (JURSE)*. IEEE, 2019.

[2]. Verma, Deepank, and Arnab Jana. "LULC classification methodology based on simple Convolutional Neural Network to map complex urban forms at finer scale: Evidence from Mumbai." *arXiv preprint arXiv:1909.09774* (2019).

[3]. Ballatore, Andrea, Michela Bertolotto, and David C. Wilson. "Geographic knowledge extraction and semantic similarity in OpenStreetMap." *Knowledge and Information Systems* 37.1 (2013): 61-81.

[4]. Chen, J., Zhou, Y., Zipf, A. and Fan, H., 2018. Deep learning from multiple crowds: A case study of humanitarian mapping. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(3), pp.1713-1722.

[5]. Saini, I., Sharma, P., Dandabathula, G., Parikh, D., Khandelwal, S. and Rao, S.S., 2019. AUTOMATIC DETECTION OF OVERHEAD WATER TANKS FROM SATELLITE IMAGES USING FASTER-RCNN. International Journal of Advanced Research in Computer Science, 10(5).

[6]. Wan, T., Lu, H., Lu, Q. and Luo, N., 2017. Classification of high-resolution remote-sensing image using openstreetmap information. *IEEE Geoscience and Remote Sensing Letters*, *14*(12), pp.2305-2309.

[7]. Viji Amutha Mary, A., Konduru Sandeep Kumar, and Kesa Pavan Sri Sai. "An Automatic Approach to Extracting Geographic Information from Internet." *Journal of Computational and Theoretical Nanoscience* 16.8 (2019): 3216-3218.

[8]. Leung, Daniel, and Shawn Newsam. "Land cover classification using geo-referenced photos." *Multimedia Tools and Applications* 74.24 (2015): 11741-11761.

[9]. Fang, F., Yuan, X., Wang, L., Liu, Y. and Luo, Z., 2018. Urban land-use classification from photographs. *IEEE Geoscience and Remote Sensing Letters*, *15*(12), pp.1927-1931.

[10]. Yang, D., Zhang, D., Frank, K., Robertson, P., Jennings, E., Roddy,

M. and Lichtenstern, M., 2014. Providing real-time assistance in disaster relief by leveraging crowdsourcing power. *Personal and Ubiquitous Computing*, *18*(8), pp.2025-2034.

[11]. Guo, Z., Du, S. and Habib, A., 2016. An extended random walker approach for object extraction by integrating VGI data and VHR image. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *9*(5), pp.1854-1863.

[12]. Prakash, T., Comandur, B., Chang, T., Elfiky, N. and Kak, A., 2015. A generic road-Following framework for detecting markings and objects in satellite imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *8*(10), pp.4729-4741.

[13]. Yang, L., Wang, X., Zhang, C. and Zhai, J., 2020. Road Extraction Based on Level Set Approach From Very High-Resolution Images With Volunteered Geographic Information. *IEEE Access*, *8*, pp.178587-178599.

[14]. Yu, J., Zhao, Q. and Chin, C.S., 2019. Extracting Typhoon Disaster Information from VGI Based on Machine Learning. *Journal of Marine Science and Engineering*, *7*(9), p.318.

[15]. Geiß, C., Schauß, A., Riedl inger, T., Dech, S., Zelaya, C., Guzmán, N., Hube, M.A., Arsanjani, J.J. and Taubenböck, H., 2017. Joint use of remote sensing data and volunteered geographic information for exposure estimation: evidence from Valparaíso, Chile. *Natural Hazards*, *86*(1), pp.81-105.

[16]. Jiang, T.B., Xia, G.S., Lu, Q.K. and Shen, W.M., 2017. Retrieving aerial scene images with learned deep image-sketch features. *Journal of Computer Science and Technology*, *32*(4), pp.726-737.

[17]. Manandhar, P., Marpu, P.R. and Aung, Z., 2018, November. Deep Learning Approach To Update Road Network using VGI Data. In *2018 International Conference on Signal Processing and Information Security (ICSPIS)* (pp. 1-4). IEEE.

[18]. Han, J. and Yamana, H., 2019. Geographic Diversification of Recommended POIs in Frequently Visited Areas. *ACM Transactions on Information Systems (TOIS)*, *38*(1), pp.1-39.

[19]. Johnson, P.A. and Sieber, R.E., 2013. Situating the adoption of VGI by government. In *Crowdsourcing geographic knowledge* (pp. 65-81). Springer, Dordrecht.

[20]. Jiang, Z., Li, Y., Shekhar, S., Rampi, L. and Knight, J., 2017, November. Spatial ensemble learning for heterogeneous geographic data with class ambiguity: A summary of results. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (pp. 1-10).