2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) | 978-1-7281-8337-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICACCCN51052.2020.9362776

Computer Vision Based Campus Surveillance

Mahesh Mahajan
Department of Information Technology
Pillai College of Engineering, New Panvel
Navi Mumbai, India
mahajanmgit16e@student.mes.ac.in

Raj RajeshwariBajre
Department of Information Technology
Pillai College of Engineering, New Panvel
Navi Mumbai, India
bajrerrit16e@student.mes.ac.in

Lovkesh Sharma
Department of Information Technology
Pillai College of Engineering, New Panvel
Navi Mumbai, India
sharmalos17dse@student.mes.ac.in

Satishkumar L Varma
Department of Information Technology
Pillai College of Engineering, New Panvel
Navi Mumbai, India
vsat2k@mes.ac.in

***Abstract*—Security is an important aspect for all of the human race and the organizations that are run by them. In this paper, an automated multipurpose security and surveillance system is proposed which is useful at highly critical places such as borders or highly restricted areas where tracking of each and every object is important. This system works on real time video footage captured using drone cameras or Closed Circuit Television (CCTV) systems and with the use deep learning object detection techniques detects buildings, trees, vehicles, water bodies, playground and slums. The system uses Faster R-CNN architecture and with the help of transfer learning the top layers of the architecture are fine tuned to detect system specific objects at a high accuracy. The video frames are given as input to convolution neural network (CNN) layers for classification of live footage that reveals the count of all objects detected. It approximately gives the percentage of each object class identified.**

Keywords—Surveillance; Campus survey; Real-time; Tensorflow;* Convolution neural network; *Region proposal network

I. Introduction

Surveillance means keeping track of behavior, activities, or information for the purpose of influencing, managing or directing. It is required to prevent theft and monitor the activities happening around the campus. This is done by using the footage from Closed Circuit Television (CCTV) or Drone. It is proposed to set up drone and interface cameras to develop a machine learning based application for automatic detection of land cover like buildings, trees, playground, vehicles, pedestrians, etc. using images/video taken form Unmanned Aerial Vehicles (UAV) or drone. The application will help to classify land cover like buildings, trees, playground, vehicles, etc in real-time by integrating a camera on a flying UAV. It will help to detect, locate and sometimes count the pedestrian and vehicles. The counting visitors in marine or other protected areas is important. Object identification and feature extraction from detected moving pedestrians and vehicles is carried out using deep learning models. The algorithms for pre-processing, feature extraction, pedestrian identification and detection are proposed to be implemented and tested in Python using the Google Colab environment.

II. Related work

The ML techniques [1] like nearest neighbor algorithm, support vector machine (SVM), decision tree (DT), random forest, and Naïve Bayes classifier have already been used for land cover prediction. In such cases, normally the input features are collected from satellite images with the help of time-series data after normalized difference vegetation index. There are six classes as output into impervious, forest, grass, water, orchard and farm. Here, to balance the data, in each class synthetic minority techniques are applied using oversampling. Python code is used to carry out operations. The k-NN gives highest accuracy.

The images are high spatial resolution taken from Google Earth (GE) [2]. These freely available images taken from GE are used for generating land cover thematic maps for a typical urban scene. The Euclidean distance along with average pixel intensity is used with k-NN classification for 5 different land objects (Building, Water Body, Vegetation, Road Network, Bare Land). It uses the study area as Bangalore city, India. Both methods exhibit classification errors due to the properties of poor spectral reflectance of GE imagery. GE maps downloader helps to download the google earth imagery. Erd helps to mosaic the variou tiles of GE imagery. Also the Arc map helps in geo referencing.

The different image classification methods are used for classifying the land usage and cover map of districts in [3]. Overall, the classification techniques for maximum likelihood produces better accuracy with the kappa coefficient as 0.8216. The Mahalanobis classifiers as well as minimum classifier result in overall accuracy with lowest kappa value. There are 5 features used for classification of land cover and these features

ISBN: 978-1-7281-8337-4/20/$31.00 ©2020 IEEE

are water bodies, forest, agriculture, urban and open land. It results in detailed information of the classified map for high resolution images.. These classified images are very useful for planning, developing and managing natural resources.

The minimum distance and support vector machine helps to comparatively analyse as supervised classifiers and the maximum likelihood and parallelepiped system is used in [4]. Such experiments indicate better results with respect to kappa coefficient and overall accuracy as compared to minimum distance and parallelepiped classifier. Here 88% is overall accuracy. The kappa value is 0.82 with respect to maximum likelihood classifier. Similarly, the combined result of kernels.Sigmoid and radial basis function with SVM gives 92% overall accuracy.

Table 1 - Experimentation Details

Paper	Machine learning techniques				
	SVM	KNN	Naive Bayes	Maximum likelihood	Decision Tree
Panda, A., et al. (2018)	✓		✓		✓
Sowmya, et al. (2017)		✓			
Mahmon, et al. (2015)				✓	
Jog, S., et al. (2016)	✓			✓	

III. Convolutional Neural Networks

CNN image classification captures image processing, processing and categorizing it under the categories desired (Eg, Human, vehicle, Dog, Road sign). The images are recognized based on the list of pixels and the image resolution. Here, h is height, w is width and d is dimensions. The image in Figure 1 is a 5x5 with 3 planes namely Red, Green and Blue which generally results in a 3x3 gray image.

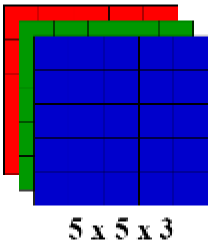


Figure 1 Image matrix of 5x5 pixels with 3 color planes

The CNN models are used to train and then test respective input images by passing it through a sequence of convolution layers. It also uses filters which are called kernels such as pooling, fully connected layers. It also uses the Softmax function to distinguish an object with values with the help of ones and zeros. Figure 2 shows a sequence of complete flow of deep CNN to classify the objects.

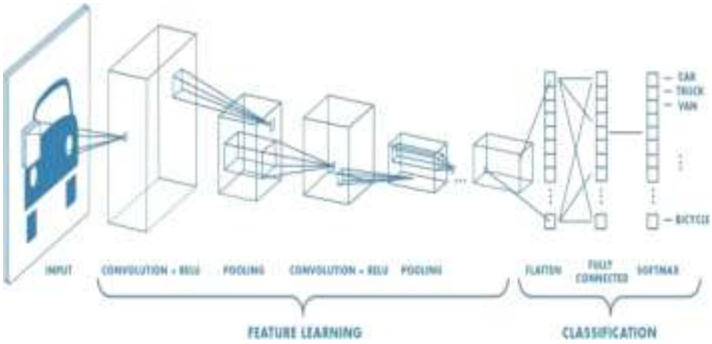


Figure 2 Neural network with many convolutional layers [9]

Convolution Layer: This is an initial layer. It helps to draw out features from an input image. This layer takes care of the association between pixels by learning image features using small boxes of input image files. This operation takes 2 inputs that is a picture matrix and a filter. An image matrix of dimension h x w x d is shown in Figure 3. A filter with value fh x fw x d give the output a volume dimensions as (h - fh + 1) * (w - fw + 1) * 1.

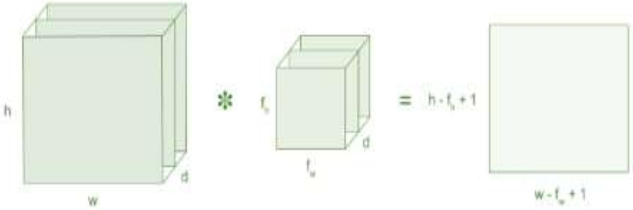
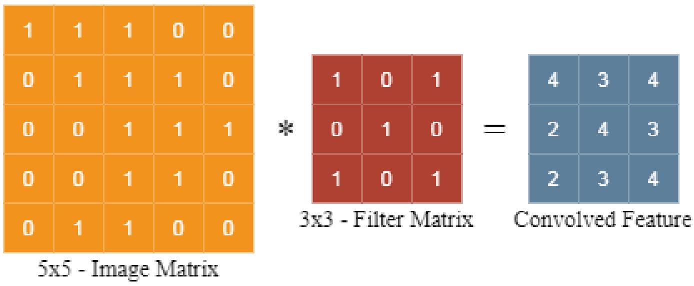


Figure 3 Image matrix multiplies kernel or filter matrix [9]

Let us take a 5 x 5 image pixel with values ranging from 0 to 1. Also take a filter matrix of size 3 x 3 as shown here in Figure 4.



$$\begin{matrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{matrix} * \begin{matrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{matrix} = \begin{matrix} 4 & 3 & 4 \\ 2 & 4 & 3 \\ 2 & 3 & 4 \end{matrix}$$

Figure 4 Image matrix multiplies kernel or filter matrix

Later, the convolution of a size of 5 x 5 image matrix is multiplied with a filter of size 3 x 3. It is also termed as a feature map. This feature map is called output as shown here in Figure 4. The convolution is performed on this picture with varying filters that can perform operations like blurring, sharping, edge detection.

A. Strides and Process

The stride is where the number of pixels normally change above the input matrix. When 1 step is then we move the filter to 1 pixel at a time. When step is two then at a time, the filter is moved to 2 pixels. Figure 5 shows the process of convolution with a filter of size 3x3 filled with ones and with a stride of 2.

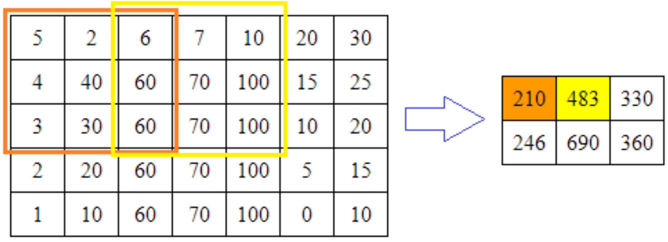


Figure 5: Process of convolution with a filter of size 3x3 filled with ones and with a 2 stride.

B. Padding

A given filter does not match sometime with the input image. In such cases there are 2 options. The first option is padding the picture with 0. It is called zero-padding. It is done to fit with the given size. The second option is to drop the part of the image where the filter does not fit with it. Only the valid part of the image is kept so it is known as valid padding.

C. Non Linearity (ReLU)

ReLU stands for Rectified Linear Unit for a non-linear operation. The output is given in Equation 1.

$$f(x) = max(0, x) \qquad (1)$$

The ReLU function is important as it helps to introduce offline content on our ConvNet. As, real-world data would require our ConvNet to read even at equal unbalanced values. Figure 6 shows the change in negative values based on ReLU operation. ReLU acts as an activation function.

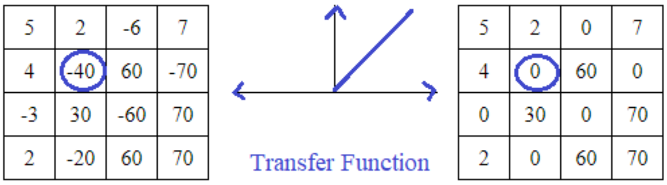


Figure 6 ReLU operation

There are other offline activities which we can do such as tanh. The sigmoid is another option that can be applied in place of ReLU function. Most data scientists use ReLU because ReLU's smart performance is better than the remaining two options.

D. Pooling Layer

The table layout section helps to reduce parameters used in case of very large images. Area integration is also called sampling or ground sampling. This helps to reduce the size of each map. This also helps to retain important details. Normally, the spatial pooling is 3 types:

- Ma
- Average
- Sum

The max takes the largest element from the feature map. Taking such elements too big sometimes results in a normal reunion. Total number of items in feature map call like merge. Figure 7 shows the max pooling operation with 2x2 matrix and stride = 2

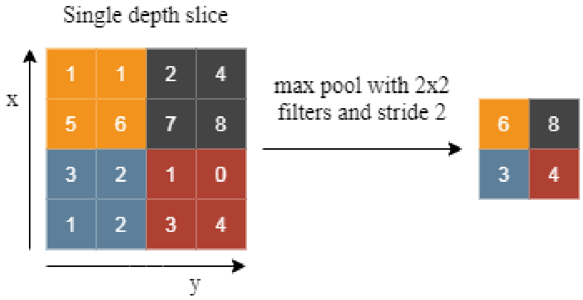


Figure 7 Process of Max Pooling with example over 4 x 4 image

E. Fully Connected Layer

The layer normally called FC layer. Here the matrix is flattened into vectors. It fed it into a fully connected layer something similar to a typical neural network.

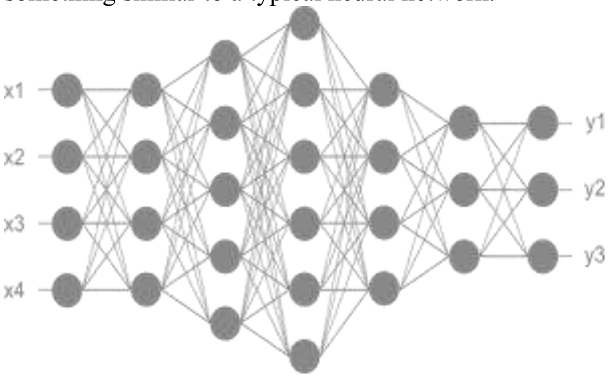


Figure 8 Layer, flattened as FC layer

The map element matrix is converted as a vector x1, x2, x3,… as shown in Figure 8. With fully integrated layers. These features are combined to create a model. Finally, there is a function applied to make it work like softmax or sigmoid to separate effects like cat, dog, car, truck etc. The input images can be Aerial, SAR, VHR , these images will be provided to the Deep learning algorithm for model training and feature identification. Then the model will provide classified images which can be further used in other applications in real time. The model is trained and the accuracy is based on the quality of the dataset. There are optimizer and activation functions used in the layers of the convolution neural network.

IV. FASTER R-CNN ALGORITHM

R-CNN fast-paced network has two networks: a regional redistribution network (RPN) for making a network. This network uses these suggestions to find objects. The Fast R-CNN uses selected searches to generate regional suggestions. The cost of time to make regional proposals is much lower for RPN than the preferred search, where RPN shares multiple computers with the acquisition network. In short, the RPN sets the circuit boxes. This is called anchors. It suggests to contain have some material.

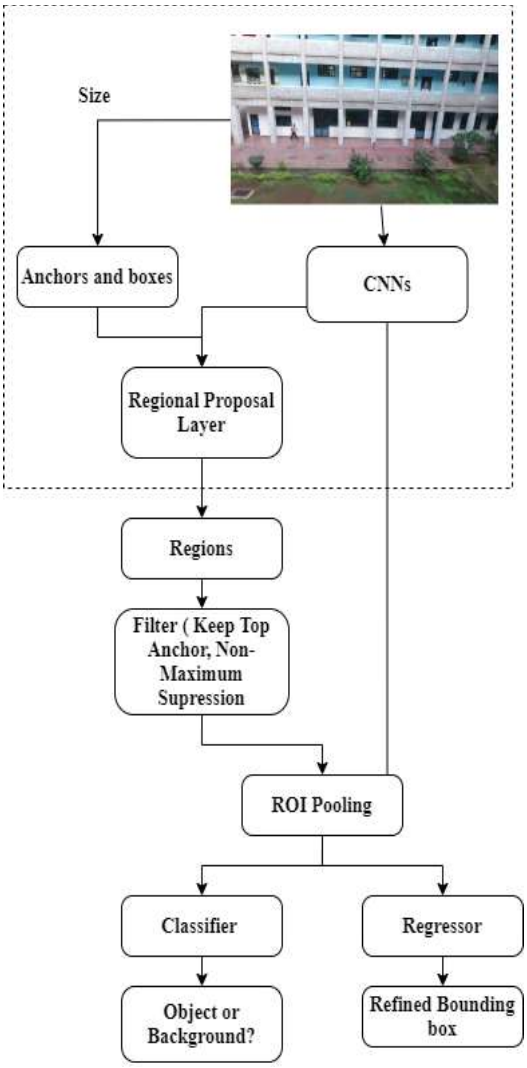


Figure 9 The architecture of the Faster R-CNN

A. Region Proposal Network

The release of a regional proposal network (RPN) is a bunch of boxes / suggestions that will be reviewed by the planned editor and postponed to finally assess the feasibility. To be more precise, the RPN predicts that the anchor may be in the background or in the front, and then filter the anchor.

B. The Regressor of Bounding Box

If you follow the labeling process, you can also select anchors based on the same regressor redesign process. One point here is that anchors with a label as a domain should not be included in the backlash, as we do not have the true boxes for their land. Feature map depth is 32 (9 anchors x 4 positions). The architecture of the Faster R-CNN is as shown in Figure 9. In this paper, a smooth L1 loss in the upper left (x, y) area of the left box, as well as a logarithm of high and wide areas, such as Fast R-CNN is used.

$$L_{loc}(t^u, v) = \sum_{i\epsilon\{x,y,w,h\}} \quad smooth_{L_1}(t_i^u - v_i) \quad (2)$$

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 if & |x| < 1 \\ |x|-0.5 & otherwise \end{cases}$$

Loss of Regressors Job is the total loss of RPN is a combination of phase loss and setback loss.

V. EXPERIMENTAL ANALYSIS

There are few experiments conducted to evaluate the performance of the proposed approach over multiple datasets. There are 3 well-known datasets that are used namely the UC Merced Land Use dataset which include aerial optical images. These images are low level characteristics similar to the Imagenet where the images are of size 256*256 pixels. There are a total 11 classes used to classify images. The Stanford Aerial Pedestrian Dataset, instead, includes aerial images of pedestrians and bikers by using a drone camera, hence less similar to general purpose images. The images were trained on only two classes for this experiment. For the third dataset we have used our own Pillai College drone videos, we annotated the videos and created a dataset with 9 classes which include Trees, Coconut Trees, Play Ground, Vehicle, etc. A notebook is used to conduct experiments with an NVIDIA GeForce GTX 1660TI 6,114 MB GPU. In each training method and dataset used, there are many number of training repetitions established by reducing the loss to 0.02 to 0.05.

A. UC-Merced Land Use Data

The UC Merced Land Use Dataset consists of 256*256 pixelated images from 11 classes (airplane, tennis court, bridges, highways, parking lot etc). These images were hand labelled using LabelIng software. These images due to their very less resolution size did not produce good results. The images were highly misclassified. The number of images used for training was 560. Each class was either misclassified or predicted with an accuracy of over less than percent.

B. Stanford Aerial Pedestrian Dataset

The Stanford Aerial Pedestrian Dataset consist of 2019*1147 pixelated images from 2 classes (pedestrian , biker). These images were annotated by Stanford. The resolution of these images was too high. It took 11 hours to train the model and decrease the training loss to 0.05 on 100 images. It gave an average accuracy with some misclassifications. So the model was trained again on 200 images and took 13 hours to bring the loss down to 0.04. It gave an accuracy of 90% on pedestrians and 85% on bikers. To get better overall accuracy of the model different images were given from the same dataset of different locations, 500 images were trained for 24 hours due to less GPU it took extra time to train this much of data. It gave an accuracy of 96% on both pedestrians and bikers.

C. College Drone Footage

College Drone Footage consisted of many videos of Full HD resolution. The videos were converted into frames and then hand labelled by us. 28,000 were hand labelled and the 1500 images were given to the model of the size 1280*720. The dataset consisted of 9 classes (Trees, Coconut Tree, Vehicles, Playground,etc.). It was trained for 8 hours.

Table 2 - Experimentation Details

Dataset	**Time (hrs)**	**Error**	**Accuracy**
UCMerced_Land Use (560 images)	12	0.04 to 0.001 (highly misclassifying)	50% of each class
Stanford Aerial Pedestrian Dataset (10 images)	11	0.1 to 0.09	80% pedestrian 70% biker 30-40% N/I
Stanford Aerial Pedestrian Dataset (20 images)	13	0.06 to 0.02	90% pedestrian 86% biker 10-20% N/I
Stanford Aerial Pedestrian Dataset (100 images)	1 day	0.05 to 0.02	97% pedestrian 93% biker 5-10% N/I
College Drone Footage	8	0.07	70% of each class
College Drone Footage	14	0.4	70-80% of each class

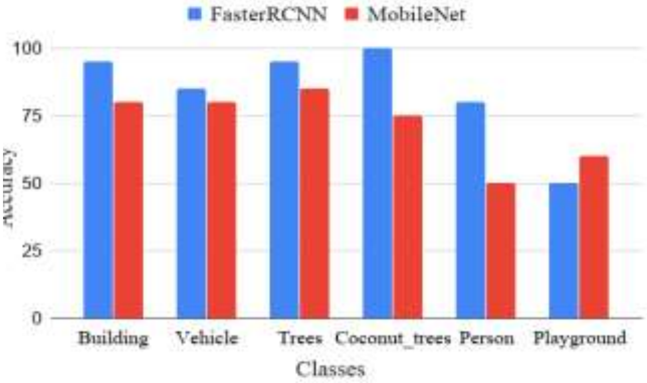


Figure 10 comparison between accuracy of classes

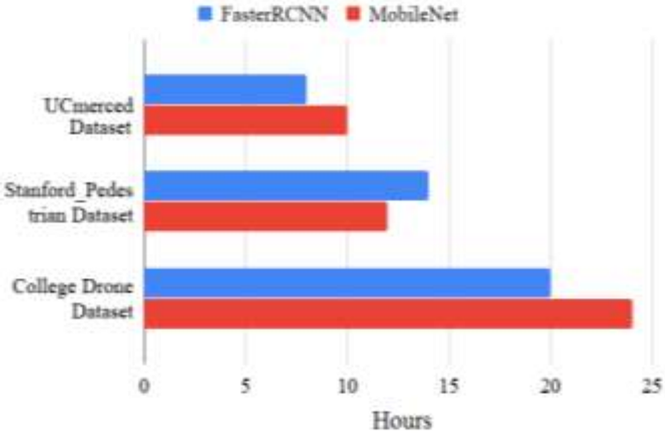


Figure 11 comparison between time taken to train

VI. RESULTS AND CONCLUSION

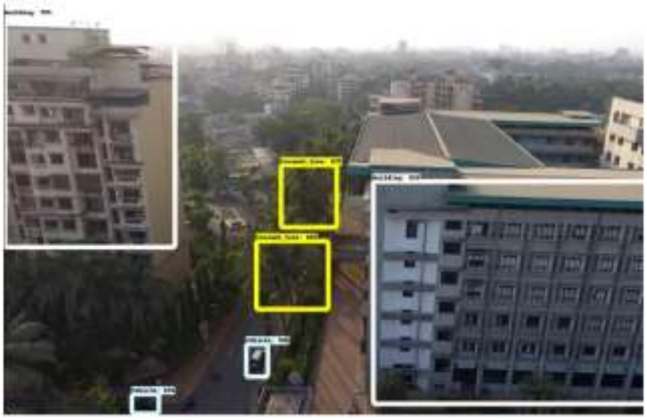


Figure 12 Object classification

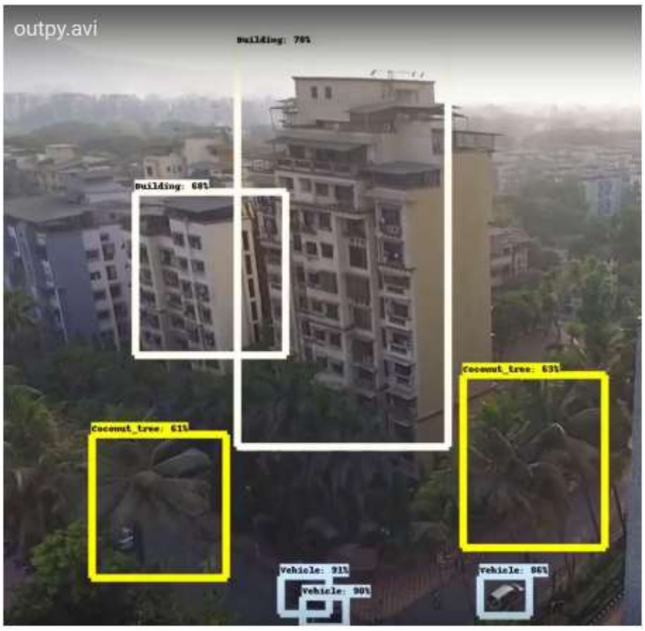


Figure 13 Object Classification

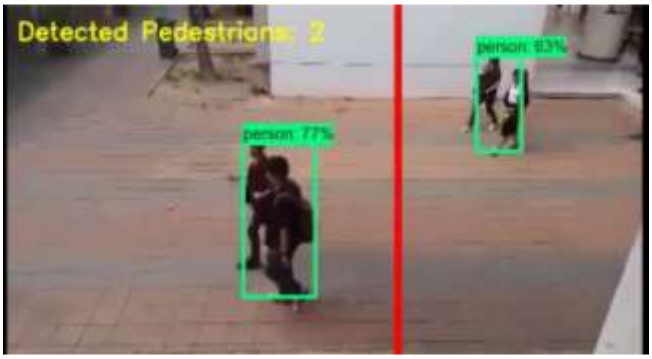


Figure 14 Pedestrian count

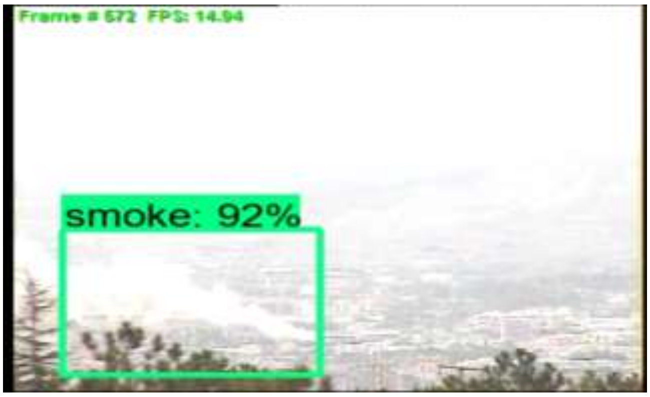


Figure 15 Smoke detection

CONCLUSION

The surveillance task by resorting to convolutional neural networks is addressed here. The architecture is experimentedwith over 3 different datasets with different properties that provide insightful information.

The training of CNN is carried out with a limited-size datasets available to validate and pretrain CNN. It helps to adapt the feature vectors generator for classification. With the fine-tuning along with several layers of the architecture, it provides good results. Overall, the experimental results are encouraging as shown in Section VI. The proposed approach helps as a better referencing method over 3 datasets. The College Drone dataset was also used in the experiment and it helped to also study the behavior of this approach during the pre-training. The aerial images used were challenging to train and analyse results..

ACKNOWLEDGMENT

We are thankful to the Pillai College of Engineering, New Panvel and University of Mumbai for providing us an opportunity to work on such projects. It helps us learn to learn and explore various components in this domain.

REFERENCES

[1] Panda, A., Singh, A., Kumar, K., Kumar, A., Uddeshya, &Swetapadma, A. (2018). Land Cover Prediction from Satellite Imagery Using Machine Learning Techniques. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT).doi:10.1109/icicct.2018.8473241

[2] Sowmya, D. R., Hegde, V. S., Suhas, J., Hegdekatte, R. V., Shenoy, P. D., &Venugopal, K. R. (2017). Land Use/ Land Cover Classification of Google Earth Imagery. 2017 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE). doi:10.1109/wiecon-ece.2017.8468898

[3] Mahmon, N. A., Ya'acob, N., &Yusof, A. L. (2015). Differences of image classification techniques for land use and land cover classification. 2015 IEEE 11th International Colloquium on Signal Processing & Its Applications (CSPA). doi:10.1109/cspa.2015.7225624

[4] Jog, S., & Dixit, M. (2016). Supervised classification of satellite images. 2016 Conference on Advances in Signal Processing (CASP).doi:10.1109/casp.2016.7746144

[5] Pritt, M., &Chern, G. (2017). Satellite Image Classification with Deep Learning. 2017 IEEE Applied Imagery Pattern Recognition Workshop (AIPR).doi:10.1109/aipr.2017.8457969

[6] Helber, P., Bischke, B., Dengel, A., &Borth, D. (2018). Introducing Eurosat: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium.doi:10.1109/igarss.2018.8519248

[7] Kaiser, P., Wegner, J. D., Lucchi, A., Jaggi, M., Hofmann, T., & Schindler, K. (2017). Learning Aerial Image Segmentation From Online Maps. IEEE Transactions on Geoscience and Remote Sensing, 55(11), 6054–6068.doi:10.1109/tgrs.2017.2719738

[8] Sophia S. Rwanga, J. M. Ndambuki,"Accuracy Assessment of Land Use/Land Cover Classification Using Remote Sensing and GIS",International Journal of Geosciences, 2017, 8, 611-622 http://www.scirp.org/journal/ijg ISSN Online: 2156-8367 ISSN Print: 2156-8359

[9] Prabhu, "Understanding of Convolutional Neural Network (CNN)— Deep Learning." [Online].Available: https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148 [Mar 4, 2018].