HUMAN DETECTION IN NATURAL DISASTERS USING YOLO v3

Using YOLO Object Detection

Developed by: Sheetal P, Meghana H D, Vidyashree M.B, Smruthi S Kiran

Smart Bridge-Remote Summer Internship Program

**ABSTRACT**

A natural disaster is the effect of a natural hazard such as flood, tornado, hurricane, volcanic eruption, earthquake or landslide on human beings. The aftermath of a major natural disaster leads to financial, environmental or human losses. The risks of losing life due to such calamities has all the more increased and added to the chaos. Many people get killed instantly due to these natural and manmade disasters when they hit a region. Many others get trapped under debris for hours and days because their presence there cannot be detected by the rescue teams easily. Natural disasters cannot be prevented, **but they can be**detected**.** Hence, the method to fix this issue by automating the detection. The study focuses on collecting and analyzing the dataset of humans in disasters to train a convolutional neural network. The object detection system tiny YOLOv3 is used for detecting the humans. The design of a system is identified which can be used for developing a mobile application for detection and presenting a visualized view of the humans.

1. INTRODUCTION

Natural disasters do occur and they are unstoppable. But humans are becoming increasingly aware in the concept of intelligent rescue operations in such calamities so that precious life and material can be saved though calamities cannot be stopped. Still there are lots of disasters that occur all of a sudden and Earthquake is one such thing. Earthquakes produce a devastating effect and they see no difference between human and material. Hence a lot of times humans are buried among the debris and it become impossible to detect them. A timely rescue can only save the people who are buried and wounded. Detection by rescue workers becomes time consuming and due to the vast area that gets affected it becomes more difficult. So the project proposes an automatic way of detecting humans in such critical conditions, so that the rescue team can go and save lives within time. For these reasons and others in this paper, we are discussing about a live human detection using opencv and yolo algorithms.

* 1. OVERVIEW

A natural disaster is a major adverse event resulting from natural processes of the Earth. Examples are floods, hurricanes, tornadoes, volcanic eruptions, earthquakes, tsunamis, storms and other geologic processes. A natural disaster can cause loss of life or damage property, and typically leaves some economic damage in its wake, the severity of which depends on the affected population's resilience (ability to recover) and also on the infrastructure available. Here the risk factor is so many of human beings suffer without the basic needs so we are trying to implement that by detecting human in natural disaster areas by using yolo algorithm(you only look once) with their location and we can try to help humans by providing basic needs and bringing them into safer place.

To solve this problem we are going to build a model using YOLOv3 which helps us in identifying the humans in natural disasters. For instance, we will be giving a video feed to our model such that it will be in a position to identify human and store its images.

* 1. PURPOSE

Natural Disasters happen almost every year in some part of the world so we have to detect how the places and the human beings living in that place and surroundings have been affected. Yolo algorithm is used for object detection, which helps for instant identification of humans struck in those disasters. By which many lives can be saved on time.Object detection is a field of Computer Vision and Image Processing that deals with detecting instances of various classes of objects in a digitally captured Image or Video.

1. LITERATURESURVEY

In natural disasters are like Floods, storms, earthquakes, droughts, forest fires and volcanic eruptions are among the most devastating types of natural catastrophe. But some disasters are man-made. These include explosions, major fires, aviation, shipping and railway accidents, and the release of toxic substances into the environment. At the American Geophysical Union meeting held last month, Benoit Mandelbrot, a professor of mathematical sciences at Yale University who is considered to be the father of fractals, described how he has been using fractals to find order within complex systems in nature, such as the natural shape of a coastline. As a result of his research, earth scientists are taking Mandelbrot’s fractal approach one step further and are measuring past events and making probability forecasts about the size, location, and timing of future natural disasters database for the period from 1900 to 2017. Keywords used in the search strategy were obtained from the classifications of natural disasters presented by the Centre for Research on the Epidemiology of Disasters. The health component was determined by selecting the health-related subject areas in Scopus.

Natural and man-made hazards include, for instance, droughts, desertification, floods, fires, earthquakes and dispersion of radioactive gases in the atmosphere. They have significant social, environmental and economic impacts. The JRC carries out extensive work to continuously monitor the situation, assess risks and potential impacts, and forecast future events as accurately as possible in order to help prevent these phenomena from happening or to limit their impact.

Various algorithms and techniques used in it. As our proposed system uses tiny YOLOv3, the basic understanding of the model was studied from this paper. This model makes use anchor boxes with a score for each object identiﬁed with the help of logistic regression.

1. PROPOSEDSOLUTION
   1. About the Dataset

The dataset for training constitutes 246 images for training and 29 images for testing the model developed. This dataset consists of repetitive images which needs to be ﬁltered out such that images in the ﬁnal dataset is unique and of different features. But, certain factors in this dataset is constant such as humans in natural disasters like floods earthquake and also humans not detected images. Using python and Open CV, we also reduced the resolution of images to remove the constant part of images in the dataset like dashboard and the sky.

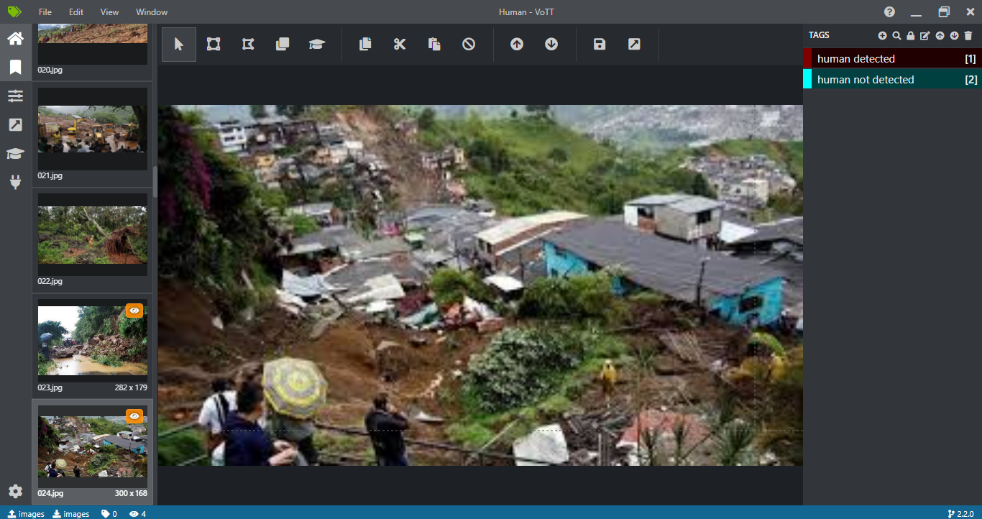


fig: Loading of dataset in vott.

* 1. Description of Dataset

There are 2 categories in the dataset namely human detected and human not detected. human detected denotes the fact that human is found in that particular place of captured image, whereas human not detected denotes that the image doesn’t contain any person. But training a model in YOLO does not require partition of images in the dataset and hence a 70-30 ratio of images from 2 classes were taken from the ﬁles into the ﬁnal dataset.

* 1. Analyzing the Dataset

Python is used to collect and process the ﬁnal human detection dataset for our system. The scraping of images from the web is done as the images were in different ﬁle formats. A common ﬁle format is used to convert all the scraped images into same extension. Since, the dataset had scraped images from web it consisted of multiple repetitive images in a row, images in the ﬁnal dataset have been taken from an interval of 17 images. This ensures that the ﬁnal dataset is free from redundancy. The images selected has some common characteristic. So, to ﬁlter out the other unwanted parts of the image, cropping of images is done using a simple python program.

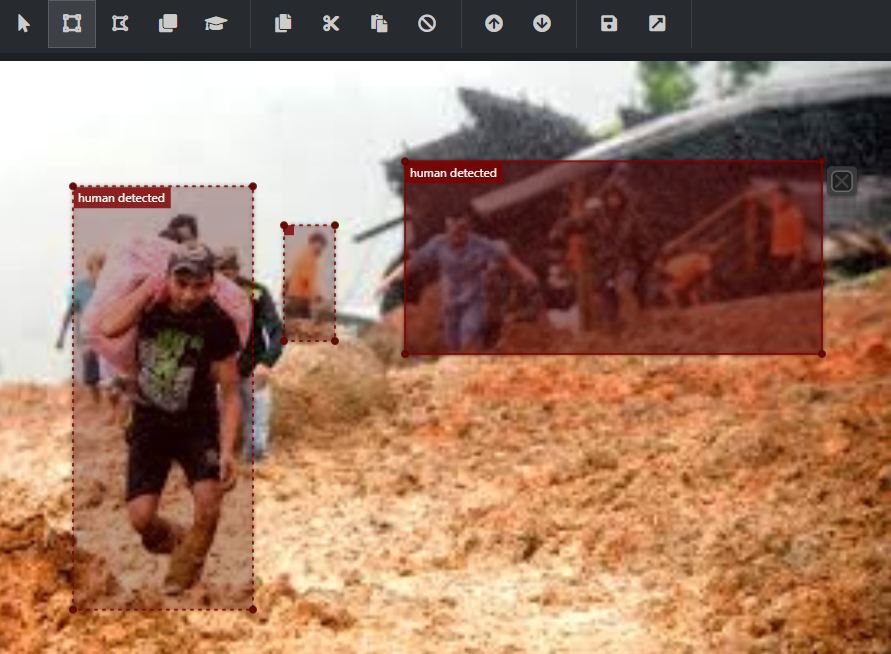


fig: Annoting image as human detected

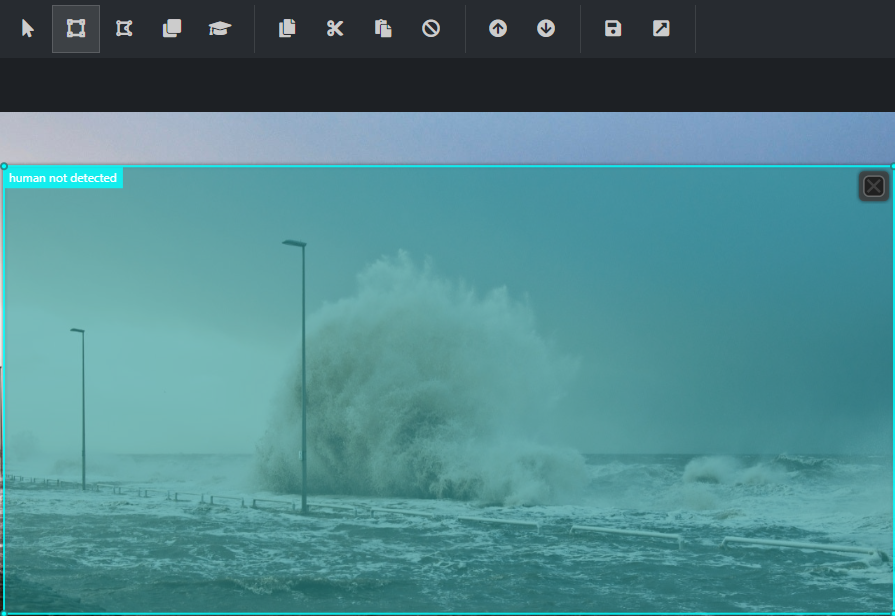


fig: Annoting image as human not detected

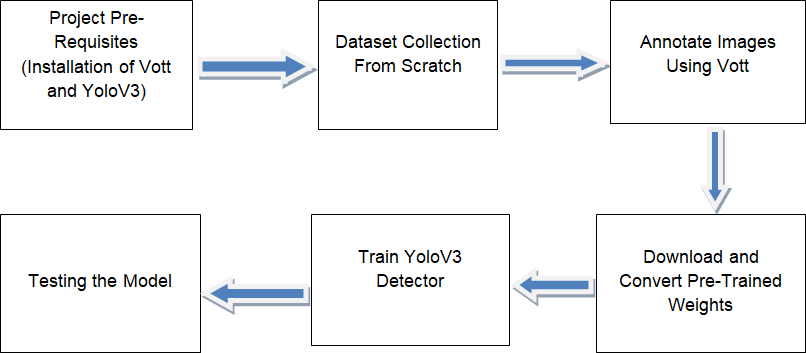
1. THEORITICALANALYSIS

YOLO treats the problem of detection as a regression problem and not as classiﬁcation problem. And it uses convolutional layers for all the tasks which are done in Fast R-CNN and R-CNN. YOLO Version1 (YOLO V1) is evaluated on the Pascal voc dataset. It consists of 2 fully connected layers along with the 24 convolutional layers. To produce the probabilities of each label. The ﬁrst 20 convolutional layers followed by the pooling layer and a fully connected layer is trained on the ImageNet of 1000 class classiﬁcation dataset. The resolution of each image is 224x224. And 2 fully connected layers along with the 4 convolutional layers which are present in the last are used for object detection. The resolution of the dataset was changed to 448x448 for better detection of the objects. The drawbacks of yolo v1 encounter the background errors (false positives) and also it ﬁnds it diﬃcult to localize small objects. YOLO Version 2 (yolo v2) is faster and better. The variation in this model is, ﬁrstly, Batch normalization is done which is used to decrease the problem of overﬁtting and also helps in improving the stability of the neural network. Secondly, the high resolution classiﬁer means that the input size of the image is increased from 224x224 to 448x448 which helps in increasing the mean average precision. Anchor boxes is another thing which makes v1 differ from the v2. YOLO v2 uses darknet-19 architecture. It contains 19 convolutional layers, 5 max pooling layers and a softmax layer. Even though it has many improvements which make it separate from YOLO V1. Still the yolo v2 is producing less accuracy. To improve the accuracy, not only in salient object detection, this model focuses on smaller object detection in pothole images.

YOLOv3 uses DarkNet-53 as base network with blocks made up of successive convolution layers of 3 × 3 and 1 × 1 ﬁlter sizes and residual connections.22 Authors show that DarkNet-53 achieves classiﬁcation similar to that of ResNet-152 on ImageNet dataset but with twice the speed. YOLOv3 uses 9 anchor boxes generated using K-means clustering where k = 3 for each of the 3 scales. To predict the objectness and class of each box, the authors have used multi-label classiﬁcation, using independent logistic classiﬁers for each class instead of softmax and used binary cross-entropy loss for training. We note that loss functions are squared errors instead of cross-entropy values in YOLOv2. Several convolutional layers are added to the base network (DarkNet-53) for detection thus making the YOLOv3 a 106 layered fully convolutional architecture. The last layer predicts a 3-d tensor encoding bounding box, objectness and class predictions S × S × (B × (4 + 1 + C)) using logistic regression. YOLOv3 predicts B = 3 boxes at 3 different scales by extracting features from different scales like feature pyramid networks.23 These feature maps at three different scales are extracted from layers having strides 32, 16, 8 respectively which implies with an input of 416 × 416,

detections are done on scales 13 × 13, 26 × 26 and 52 × 52. Features are extracted from the last two layers of these additional layers, upsampled two times and concatenated with the feature map from the base network. The former upsampled features give semantic information while the latter from backbone gives ﬁner-grained information about the objects. YOLOv3 achieves comparable results with the other detection methods for COCO dataset considering 0.5 IOU and at a speed that is three times higher than RetinaNet.24 YOLOv3 with DarkNet-53 being a 106 layered architecture, is slower than the YOLOv2 but has higher accuracy and is still faster than other state-of-art methods of object detection. At the ﬁrst level, image is down sampled by the network till 81 layers which has a stride of 32. So, the 416 × 416 image becomes 13 × 13 × 255. One detection is made at this level. Feature map from layer 79 is connected to convolutional layer 84 and then upsampled by 2 times resulting into 26 ×26 feature map. This feature map is then depth concatenated with the feature map from layer 61. This concatenated feature map is again passed through some stacked 1 × 1 convolutional layers and at layer 94 the feature map of 26 × 26 × 255 is used for detection. The features from layer 91 are passed through other convolutional layers till layer 97 and depth concatenated with the feature map from layer 36 of the classiﬁer network. This concatenated feature map is passed through the convolutional layers till the 106th layer and then the detection is performed on this 52 × 52 × 255 feature map. This way, the detection is performed at 3 different scales reducing the image using strides of 32, 16 and 8 on the output of the convolutional layers.

* 1. BLOCKDIAGRAM

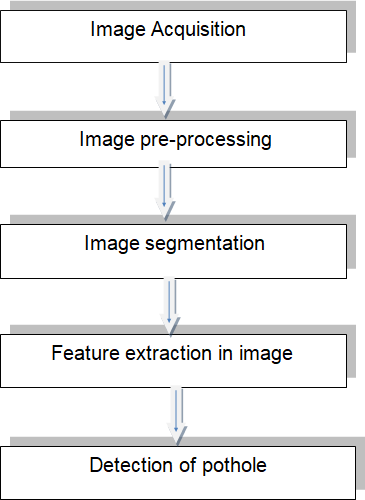


1. **SOFTWARE DESIGNING**
2. Microsoft VOTT
3. Machine Learning Algorithms
4. Python Programming Language
5. YOLOv3structure

With multiple images being captured, it becomes essential that the object detection model detects humans in the least amount of time. YOLOv3 (You Only Look Once ver.3) improves operation speed which meets real time requirements for detection. Darknet is a framework for YOLO implemented using C/CUDA. To make the object detection model mobile friendly, the model is converted to a TensorFlow lite model, which compresses the neurons in the neural network. This will make the model run eﬃciently in the mobile device.

1. EXPERIMENTALINVESTIGATION

In our project, we have used Image dataset. The dataset contains two folders: Test\_images and Training\_images. The Training\_images folder consists of 218 images of natural disaster with and without humans. Similarly, the Test\_images consists of 48 images.

**7. FLOWCHART**

**8.PROCEDURE**

After the annotations of images, yolov3 model should be trained. Before getting started we need to download the pre-trained dark-net weights and convert them to YOLO format. To do this

* navigate to TrainYourOwnYOLO/2\_Training and run: python Download\_and\_Convert\_YOLO\_weights.py
* Once finished, train the detector by running:python Train\_YOLO.py
* The final weights are saved in TrainYourOwnYOLO/Data/Model\_Weights. This concludes the training step and you are now ready to detect objects in new images

**Prediction**

* To test our object detector, navigate to TrainYourOwnYOLO/3\_Inference and run: python Detector.py
* This will apply your freshly trained YOLOv3 object detector on test images located in TrainYourOwnYOLO/Data/Source\_Images/Test\_Images. In our example, we detect human faces in natural disaters . To test the detector on your own images, populate the folder with your own images.

**9. RESULT:**



fig: final output predicted wit percentage of detection

Various deep convolutional networks with different architectures and improvements in the accuracy and speed as a result have been proposed for image classiﬁcation. The classiﬁcation models are evaluated in terms of the number of ﬂoating point operations, depth, number of parameters, sparsity inﬂuence the computational speed and memory requirement. DarkNet-19 and DarkNet-53 are fully convolutional networks, proposed with YOLOv2 and YOLOv3 respectively and have achieved reasonable classiﬁcation accuracy and speed. We have evaluated these backbone networks for their accuracy in detecting the humans. We trained and tested our algorithms on the complete data set to start with. Later we randomly separated the data set into training data and test data so that we had samples from each class. The model was able to classify more than 80% of the images. The testing accuracy of the system is about 80%. Depending on the classiﬁcation, the message and details will be sent over to produce sound or alarm to the person who hands over that department. Thus effectively detecting the humans.

10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

1. Except from the improvements in precision observed in the classiﬁcation/prediction problems at the surveyed works, there are some other important advantages of using YOLO image processing. Previously, traditional approaches for image classiﬁcation tasks were based on hand-engineered features, whose performance and accuracy greatly affected the overall results. Feature engineering (FE) is a complex, time-consuming process which needs to be altered whenever the problem or the data set changes. Thus, FE constitutes an expensive effort that depends on experts’ knowledge and does not generalize well.
2. It seems to generalize well and they are quite robust even under challenging conditions such as illumination, complex background, size and orientation of the images, and different resolution. They can take preventive measures.

DISADVANTAGES:

1. The main disadvantage is that sometimes it can take much longer to train. However, after training, their testing time eﬃciency is much faster than other methods
2. Other disadvantages include problems that might occur when using pre-trained models on similar and smaller data sets, optimization issues because of the models’ complexity, as well as hardware restrictions.
3. APPLICATIONS
4. We are developing a system that will detect the humans through a camera.
5. Once the frame matches our data then it will detect the exact pattern of the humans by framing it.
6. This ensures prior detection of humans in natural disasters.
7. Early and accurate identiﬁcation of humans is essential to ensure reduce death rate.
8. CONCLUSION

Natural disasters reflect people's vulnerability or their susceptibility to be affected when confronted with floods, cyclones, volcanic eruptions, landslides.The paper showcases our work performed to train an object detection model capable of detecting the humans in natural disasters. The collection of images for dataset to train the model consisting images of natural disasters like floods,uraicane, earthquake .Tiny YOLOv3, a CNN which is capable to identify the humans in natural disasters, which helps to identify humans who are in danger as early as possible. These capacities have to be utilized to reduce the risk of disaster.

1. FUTURESCOPE

During the emergency situations and especially in urban disasters, this project will be a great requirement. The disasters can be sensed in a quicker time and the rescue operation will be there for the stake to help the victims.

The invention of this device will be user friendly and advanced in technology. This

circuit is mainly using in land rovers by making its movement more effective on trough surface .More optimization which include image processing through the using of yoloV3 object detection it recognize the images for humans detection in natural disasters.

14.REFERENCES

[1] M. B. Blaschko and C. H. Lampert. Learning to localize objects with structured output regression. In Computer Vision– ECCV 2008, pages 2–15. Springer, 2008.

[2] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In International Conference on Computer Vision (ICCV), 2009.

[3] H. Cai, Q. Wu, T. Corradi, and P. Hall. The crossdepiction problem: Computer vision algorithms for recognising objects in artwork and in photographs. arXiv preprint arXiv:1505.00110, 2015.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 1, pages 886–893. IEEE, 2005.

[5] T. Dean, M. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, J. Yagnik, et al. Fast, accurate detection of 100,000 object classes on a single machine. In Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pages 1814–1821. IEEE, 2013.

[6] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. arXiv preprint arXiv:1310.1531, 2013.

[7] J. Dong, Q. Chen, S. Yan, and A. Yuille. Towards unified object detection and semantic segmentation. In Computer Vision–ECCV 2014, pages 299–314. Springer, 2014.

[8] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pages 2155–2162. IEEE, 2014.

[9] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision, 111(1):98–136, Jan. 2015.

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(9):1627–1645, 2010.

[11] S. Gidaris and N. Komodakis. Object detection via a multiregion & semantic segmentation-aware CNN model. CoRR, abs/1505.01749, 2015.

[12] S. Ginosar, D. Haas, T. Brown, and J. Malik. Detecting people in cubist art. In Computer Vision-ECCV 2014 Workshops, pages 101–116. Springer, 2014.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pages 580–587. IEEE, 2014.

[14] R. B. Girshick. Fast R-CNN. CoRR, abs/1504.08083, 2015.

[15] S. Gould, T. Gao, and D. Koller. Region-based segmentation and object detection. In Advances in neural information processing systems, pages 655–663, 2009.