International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Detection of Camouflaged Objects Using Convolutional Neural Network

Khandave Vaishali Ramdas[1] , Dr. Shyamrao Gumaste[2]

Submitted: 25/05/2023 **Revised**: 06/07/2023 **Accepted**: 24/07/2023

Abstract: The goal of camouflage is to blend a foreground object's texture into the background image frame's texture. Camouflage detection methods, also known as decamouflaging methods, are typically used to locate foreground objects that are concealed in the background image. One of the key uses of machine vision in this field is the identification of camouflaged objects in photographs for both military and non-military applications. The camouflage detection technique put forth in this study can be utilized to find one or more target objects in camouflaged pictures. In this study, a neural network is employed to recognize the item in the image of camouflage. Images with naturally occurring and man-made camouflaged objects are used in experiments. Naturally camouflaged objects are equivalent to animals and artificially, that is, man-made camouflaged objects are equivalent to people in the actual world. The performance of the proposed technique is validated using precision and recall.

Keywords: *Camouflage detection, decamouflaging*

1. Introduction

The word "camouflage" in English comes from the French verb "camoufler," which means to conceal. The natural colours of animals and man-made camouflaged images are used in camouflage to create concealment and obscurity. The term "camouflage" dates back to the early days of the animal kingdom, when creatures used to conceal themselves from predators by altering the pattern, texture, and pigment of their bodies to blend in with their surroundings.

Decamouflaging, also known as camouflaging identification system, is a technique used to distinguish foreground objects from camouflaged images in order to separate the target object from the background. The Camouflage Identification System has a wide range of possible uses, including the ability to identify foes in the field of battle, find manufacturing flaws in items, and identify duplicate goods during logistics. Therefore, the idea behind decamouflaging is how to recognize a certain texture from the background that is provided. To identify the camouflaged areas, some models have been proposed in the literature survey. Some of them detect the motion camouflaged objects means detecting an object that gets camouflaged while moving. Opposite to motion camouflaged objects, some of the models detect the static camouflaged object in an image.

[1]*Research Scholar,*
MET's Institute of Engineering
Nashik, India,
Savitribai Phule Pune University, Pune
vaishali7187@gmail.com
[2]*Professor,*
MET's Institute of Engineering
Nashik, India
Savitriba iPhule Pune University, Pune
svgumaste@gmail.com

2. Literature Survey

In this paper, the author offered a technique for extracting the camouflaged area from a given image's background [4]. Author claim that decamouflaging or camouflaging is done in an unsupervised manner, which means that neither the disguised portion of the image nor the characteristics of the background are known. First, they have transformed the input image into a grayscale version. Then, they divided the image into LXL equal blocks. Further the GLCM value for each block of the image frame is calculated. They then computed the mean for every block. A mark is placed on the dendrogram's largest individual block, and the adjacent blocks are then combined after the dendrogram has been plotted for the mean values of each block. This technique has a 70% success rate in identifying the area covered in camouflage. This method is not feasible.

Using multiscale aggregation of filter response and shape element, the author in this work explains how to discriminate between texture segments in images [1]. Texture characteristics are computed at various scales and their features are used to determine bigger texture qualities. This method also establishes the structure of the texture components and categorizes them according to their size, brightness, aspect ratio, and orientation before using other statistical criteria to differentiate between various textures. The identical texture segment can be recognized and retrieved using this method. The challenge here is figuring out how to combine several statistical variables into a unique weight. Additional statistics can be used to improve this approach.

To locate a hidden object in a picture, the author used the local HSV stands for Hue-Saturation-Value color model and gray-level co-occurrence matrix (GLCM) texture features [7]. The given input image is separated into equal-sized subblocks, and the texture and color characteristics of the subblocks are then computed. Hue-Saturation-Value (HSV) Color space is quantified from each sub-block to describe the color feature of an image using a cumulative histogram. Each sub-block's texture feature is calculated using a gray-level co-occurrence matrix. The adjacency matrix of a bipartite graph is created using the integrated matching technique of query and target image sub blocks. Based on the idea of Most Similar Highest Priority on images, this adjacency matrix is used to match the query sub block image and the target sub block image.

Author described a background subtraction technique based on an image's colour and edge properties [6]. This paper mainly concentrated on the removal of the foreground from the background in visual surveillance. The author created a model based on the features of color, edge, and intensity to address the camouflage problem that arises whenever the foreground and background colours are the same. This method uses low contrast and shadowed images to distinguish between the foreground and background. This technique resolves the issue of the shadow effect in detecting P. Sengottuvelan's camouflaged image, i.e., paper of Exploratory Image Analysis for Decamouflaging Performance.

Author discussed identifying camouflage defects and employed co-occurrence matrix-based texture extraction inside a tiny block of an image [2]. Here, the entire image frame is first separated into tiny, equal disjoint chunks, and invariant central moments up to Kth order are then calculated for each block (10th order is adequate). By using the cluster analysis approach and the watershed segmentation methodology, camouflaged areas of a picture are finally detected. When the camouflaged area is present in close proximity to the image blocks and the overall percentage of camouflage is less than 4%, this strategy produces good results. When there is a significant quantity of camouflaged imagery in the input image frame or when the regular texture is not regular, this technique would not be practical. Additionally, it is unable to distinguish between different types of defects when they coexist in a single image frame.

Author suggested a method for detecting camouflage based on the color and intensity properties of an image [3]. The authors in this article confronted the issue of camouflage where the foreground object's pixel had the same intensity as the background object. Also they covered two types of camouflage, namely dark camouflage and bright camouflage. When a pixel loses intensity and transforms into a shadow, a dark camouflage is produced. The appearance of Light Camouflage follows when the brightness of the foreground pixels is greater than that of the background pixels. Foreground identification is accomplished with the use of normalized chromaticity measures and normalized intensities from the color-intensity model, and the camouflage component is then extracted using the pixel classification technique. With strong shadows and light, it does not function well. It might be improved by taking into account the image's corner, cue, and edges.

Author proposed a technique to remove the background in videos for surveillance. The method uses Bayes classification and a Gaussian mixture model. However, this method faces a problem when the foreground and background have similar colours. It is difficult to choose a threshold to separate them [5]. The problem of camouflage typically arises in visual surveillance applications when the foreground object's colour characteristics are similar to those of the background image frame. They therefore put forth a solution that involves averaging video frames in order to lessen differences in background image frames. They decreased the likelihood of camouflage, but more investigation is needed.

3. Algorithm

Step 1: Load the image file that will provide as input

Step 2: Scale each pixel value into a 0...1 range

Step 3: Feeding the pre-trained YOLO neural network

Step 4: Obtaining the detection result

Step 5: measuring and printing how many objects were detected and how fast the detection phase

You Only Look Once is referred to as YOLO. YOLO is a technique that uses convolutional neural networks to give real-time object detection. As the name implies, the technique employs a single forward propagation to detect objects in a neural network. This suggests that a single algorithm run is utilized to carry out prediction throughout the entire image. The convolutional neural network (CNN) is used to forecast several bounding boxes and class probabilities at once.

The image is first separated into grid cells. Forecasting B bounding boxes and providing confidence scores are done in each grid cell. The cells forecast the class probability and use that information to identify each object's class. Only one convolutional neural network is used to make all of the predictions at once. It is ensured via intersection over union that the projected bounding boxes match the actual boxes of the items. As a result, the bounding boxes that don't fit the dimensions of the objects (such as height and width) are eliminated. The unique bounding box that precisely fits the objects, will make up the final detection.

Initially, this work consists of 20 classes only to find a camouflage object from an image: bicycle, boat, sofa, cow, cat, bottle, person, TV monitor, bus, dining table, bird, horse, motorbike, dog, chair, sheep, car, train, aeroplane, and potted plant.

4. Experimentation Analysis

An experimentation analysis is performed on the image dataset. The source of this image dataset is the article "Anabranch network for camouflaged object segmentation" [13].

The dataset contains 1250 images. There are two categories of concealed images that means camouflaged items: those that are camouflaged naturally and those that are created, which typically correlate to humans and animals in the actual world, respectively. Animals that are camouflaged include reptiles, insects, amphibians, mammals, birds, and submerged creatures in a variety of habitats, including the ground, ocean, desert, forest, mountain, and snow. Soldiers on the battlefield and human body painting artists both use camouflaged humans. 20% of the dataset is utilized for testing, while 80% of the dataset is used for training.

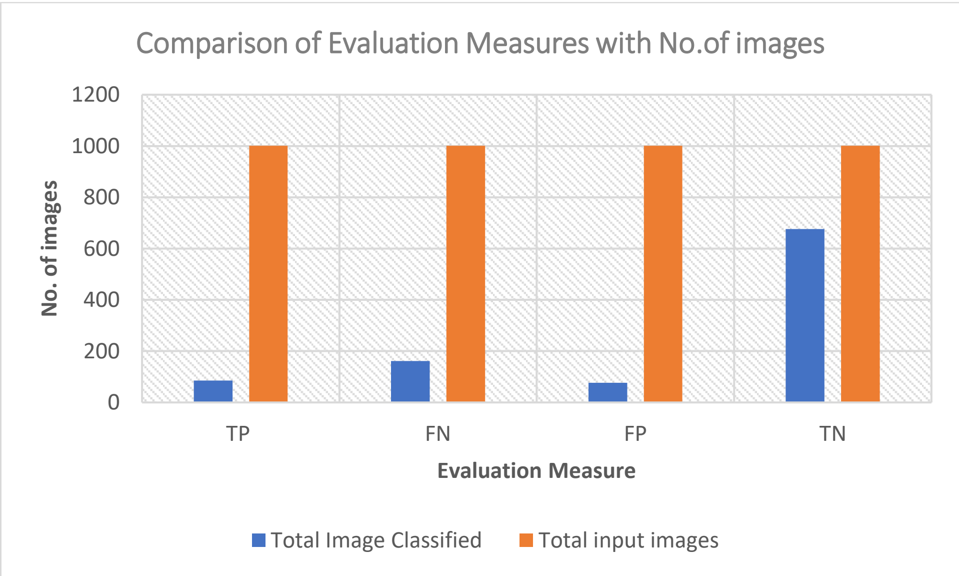
Therefore, by giving all the images from the dataset as input to the system, the following results are obtained.

Firstly, the results on the evaluation measure of the training dataset images are calculated.

Table 1: Results on Evaluation Measure (Training Dataset)

	TP	**FN**	**FP**	**TN**
Total Image Classified	86	162	77	676
Total Input Images	1001	1001	1001	1001
Percentage	9%	16%	8%	68%

Graphically the results on Evaluation Measure are shown as below.



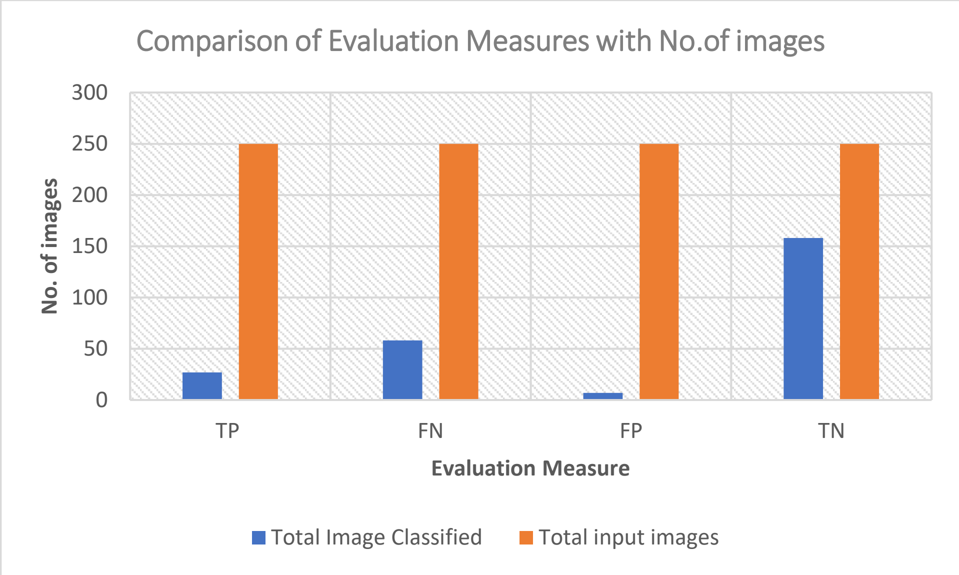
Graph 1: The Results on Evaluation Measure (Training Dataset)

Now, the results on the evaluation measure of the testing dataset images are calculated.

Table 2: Results on Evaluation Measure (Testing Dataset)

	TP	**FN**	**FP**	**TN**
Total Image Classified	27	58	7	158
Total Input Images	250	250	250	250
Percentage	11%	23%	3%	63%

Graphically the results on Evaluation Measure are shown as below.



Graph 2: The Results on Evaluation Measure (Testing Dataset)

From the values of the above evaluation measure, the results of the performance parameter are obtained as follows:

i. Performance parameter values for the training dataset, i.e., from Table 1

specificity $= \frac{TN}{TN+FP}$

$= 0.90$

$= 0.35$

accuracy$= \frac{TP+TN}{TP+TN+FP+FN}$

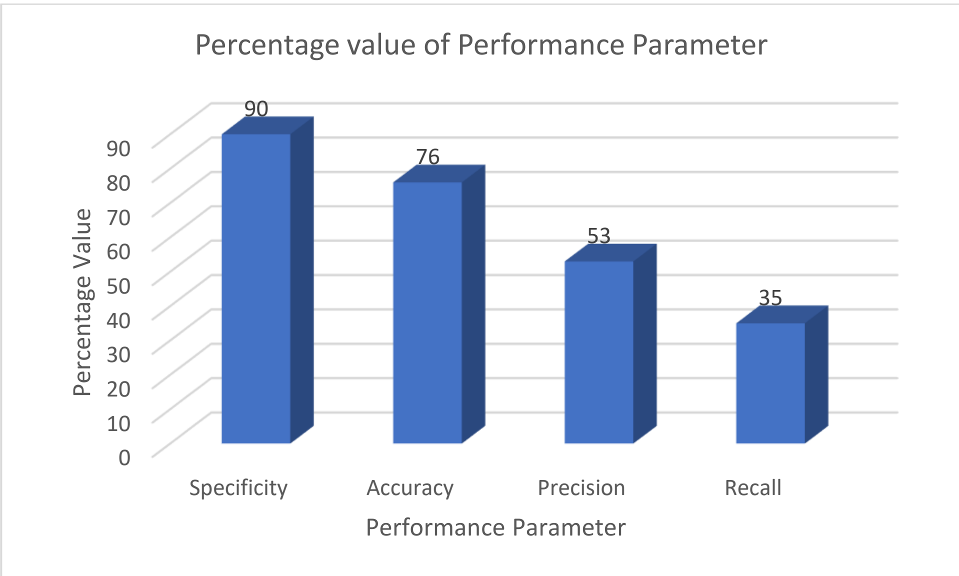
$= 0.76$

precision $= \frac{TP}{TP+FP}$

$= 0.53$

recall $= \frac{TP}{TP+FN}$

Graphically the results of the performance parameter are shown as below.



Graph 3: Performance Parameter

ii. Performance parameter values for the testing dataset, i.e., from Table 2

specificity $= \frac{TN}{TN+FP}$

accuracy$= \frac{TP+TN}{TP+TN+FP+FN}$

$= 0.96$

$= 0.74$

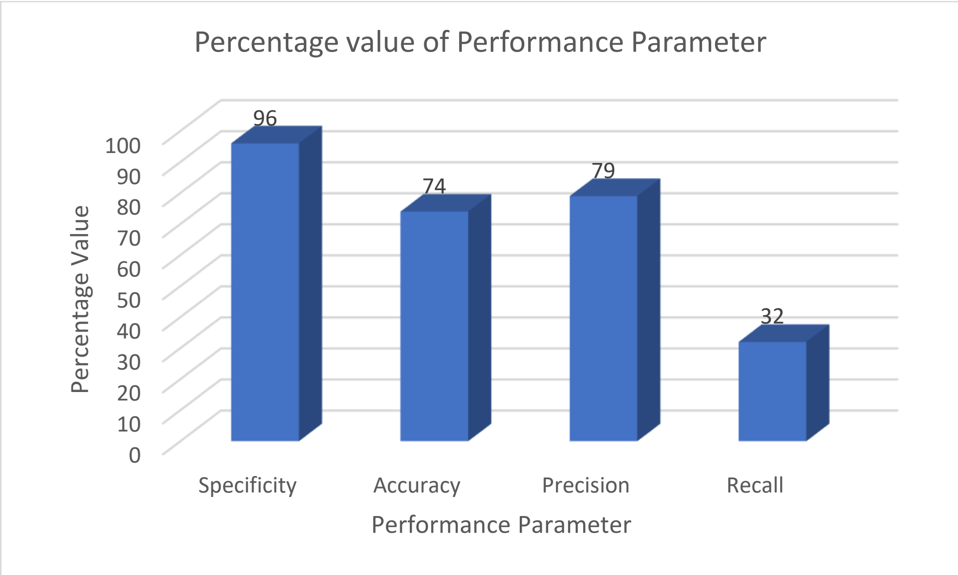
precision $= \frac{TP}{TP+FP}$

$= 0.79$

recall $= \frac{TP}{TP+FN}$

$= 0.32$

Graphically the results of the performance parameter are shown as below.



Graph 4: Performance Parameter

By comparing the results of both the training and testing dataset, it is observed that accuracy lies in between 74% to 76%

Then, the K-fold cross-validation method is applied, and the following results are obtained. Taken the value of K as 5.

Table 3: 5-fold results on the training dataset

Fold	Accuracy	Specificity	Precision	Recall
1	0.76	0.90	0.53	0.35
2	0.75	0.90	0.56	0.33
3	0.76	0.91	0.58	0.34
4	0.75	0.92	0.65	0.34
5	0.77	0.91	0.56	0.35
Average	0.76	0.91	0.58	0.34

Table 4: 5-fold results on the testing dataset

Fold	Accuracy	Specificity	Precision	Recall
1	0.74	0.96	0.79	0.32
2	0.79	0.93	0.64	0.40
3	0.76	0.90	0.56	0.35
4	0.78	0.86	0.32	0.37
5	0.72	0.91	0.62	0.30
Average	0.76	0.91	0.59	0.35

Therefore, from the above results, it is observed that the accuracy obtained is 76%.

The old YOLO neural network has the following limitations:

1. Only 20 objects detect
2. It draws 2 bounding boxes per object
3. Resolution /Dimension: Input Size of image 224 x 224
4. Grid Size: 7 × 7
5. Output Image Size: 416 x 416
6. Detect Only One Object in one grid cell
7. Detect Only 49 Object in an image
8. Use VOC Dataset
9. Struggles to detect Small & Close Object
10. More Localization Error
11. Feature Extraction Network Darknet Framework

Therefore, the work needs to modify the algorithm in specify context here! In the old YOLO neural network, there is no Batch Normalization layer. But the new YOLO neural network adds the Batch Normalization layer. Batch normalization normalizes the input layer by altering it slightly and scaling the activations. Batch normalization decreases the shift in unit value in the hidden layer and by doing so it improves the stability of the neural network. By adding batch normalization to convolutional layers in the architecture MAP (mean average precision) has been improved by 2%. It also helped the model regularise and overfitting has been reduced overall.

The new YOLO neural network also adds Multi-scale Training. An old YO LO has a weakness in detecting objects with different input sizes which says that if YOLO is trained with small images of a particular object, it has issues detecting the same object on the image of a bigger size. New YOLO trains the model with images of different sizes. During the training, the image dimensions are randomly selected every ten batches.

Old YOLO has a bad performance in the localization of bounding boxes. New YOLO adds the anchor boxes to help the localization.

The new algorithm provides the following features:

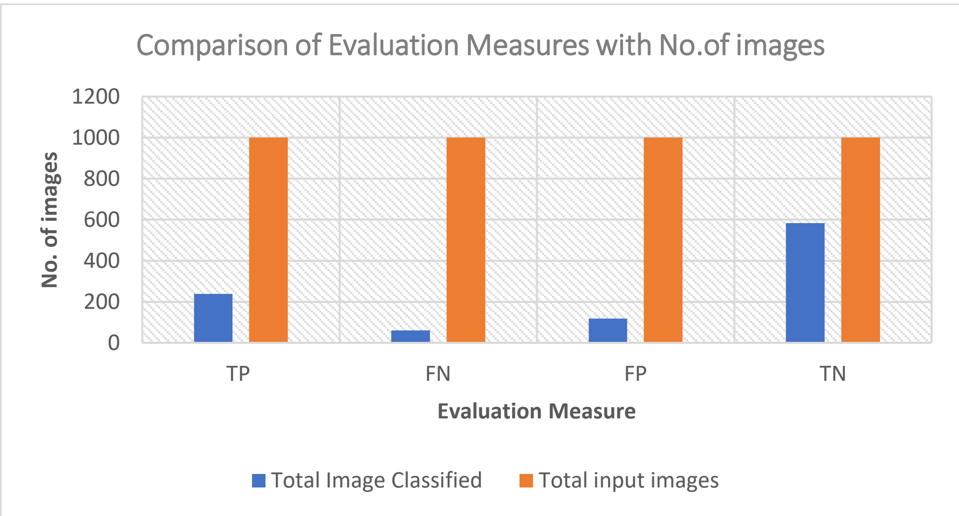
1. 80+ objects detect
2. It draws a single bounding box per object
3. Resolution /Dimension: Input Size of image 608 x 608
4. Grid Size: 13 x 13
5. Output Image Size: 608 x 608
6. Detect the number of objects in one grid cell
7. Detect 80+ Object in an image
8. Use COCO Dataset
9. Detect Small & Close Object
10. Less Localization Error
11. Feature Extraction Network Darknet 53

By giving the same images as input to the system, the following results are obtained. It is observed from the results that accuracy is increases after modification.

Table 5: Results on Evaluation Measure (Training Dataset)

	TP	**FN**	**FP**	**TN**
Total Image Classified	239	61	118	583
Total Input Images	1001	1001	1001	1001
Percentage	24%	6%	12%	58%

Graphically the results on Evaluation Measure are shown as below.



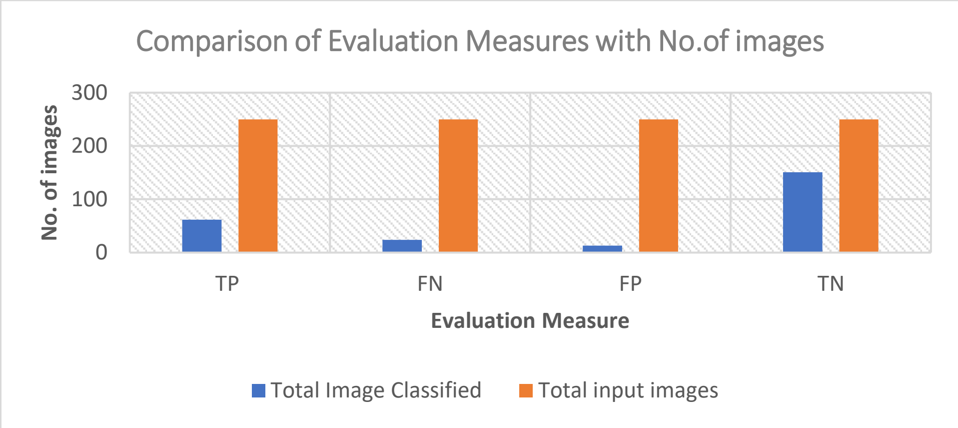
Graph 5: The Results on Evaluation Measure (Training Dataset)

Now, the results on evaluation measure of Testing dataset images are calculated.

Table 6: Results on Evaluation Measure (Testing Dataset)

	TP	**FN**	**FP**	**TN**
Total Image Classified	62	24	13	151
Total Input Images	250	250	250	250
Percentage	25%	10%	5%	60%

Graphically the results on Evaluation Measure are shown as below.



Graph 6: The Results on Evaluation Measure (Testing Dataset)

From the values of above evaluation measure, the results of performance parameter are obtained as follows:

iii. Performance parameter values for the training dataset, i.e., from Table 5

specificity $= \frac{TN}{TN+FP}$

$= 0.83$

accuracy$= \frac{TP+TN}{TP+TN+FP+FN}$

$= 0.82$

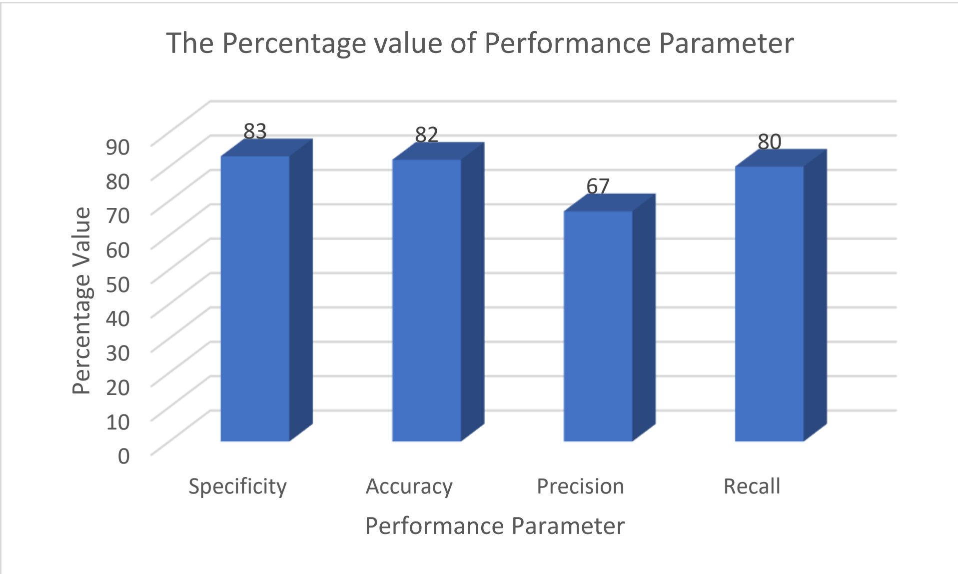
precision $= \frac{TP}{TP+FP}$

$= 0.67$

recall $= \frac{TP}{TP+FN}$

$= 0.80$

Graphically the results of performance parameter are shown as below.



Graph 7: The Results on Evaluation Measure (Training Dataset)

iv. Performance parameter values for the testing dataset, i.e., from Table 6

specificity $= \frac{TN}{TN+FP}$

$= 0.92$

accuracy$= \frac{TP+TN}{TP+TN+FP+FN}$

$= 0.85$

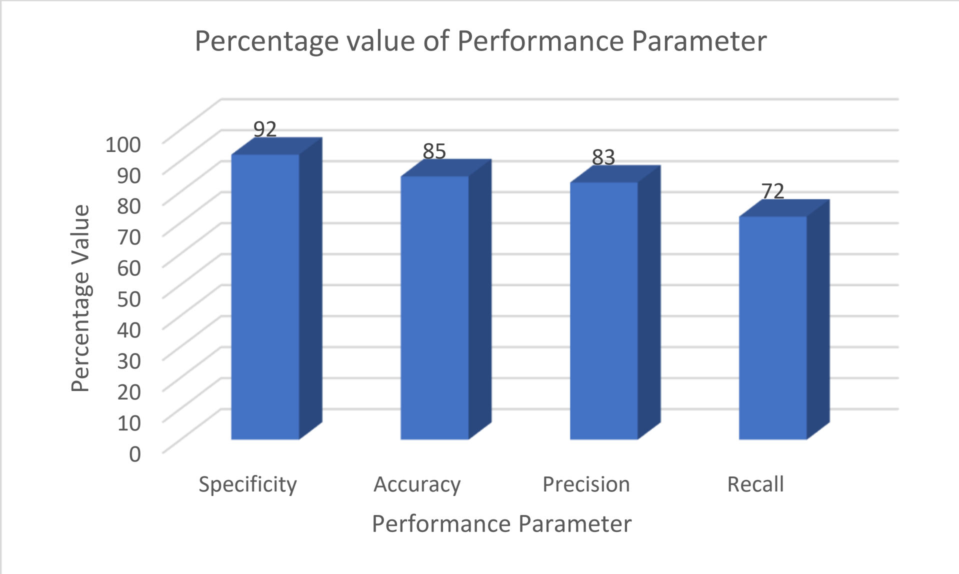
precision $= \frac{TP}{TP+FP}$

$= 0.83$

recall $= \frac{TP}{TP+FN}$

$= 0.72$

Graphically the results of performance parameter are shown as below



Graph 8: The Results on Evaluation Measure (Testing Dataset)

Then, the K-fold cross-validation method is applied, and the following results are obtained. Taken the value of K as 5.

Table 7: 5-fold results on the training dataset

Fold	Accuracy	Specificity	Precision	Recall
1	0.82	0.83	0.67	0.80
2	0.83	0.85	0.70	0.76
3	0.83	0.86	0.71	0.78
4	0.83	0.86	0.73	0.78
5	0.82	0.84	0.67	0.78
Average	0.83	0.85	0.70	0.78

Table 8: 5-fold results on the testing dataset

Fold	Accuracy	Specificity	Precision	Recall
1	0.85	0.92	0.83	0.72
2	0.84	0.83	0.68	0.85
3	0.80	0.82	0.64	0.76
4	0.80	0.81	0.54	0.78
5	0.84	0.88	0.80	0.79
Average	0.83	0.85	0.70	0.78

Observations:

1. From the above results, it is observed that accuracy obtained is 83% in the modified YOLO neural network which greater than the previous YOLO neural network.

2. The TP detection rate increases nearly about by 50% in the modified YOLO neural network than the previous YOLO neural network.

3. The modified YOLO neural network detects images faster than the previous YOLO neural network.

The Comparison of images obtained from the result:

➢ Multiple Bounding boxes in the previous work.

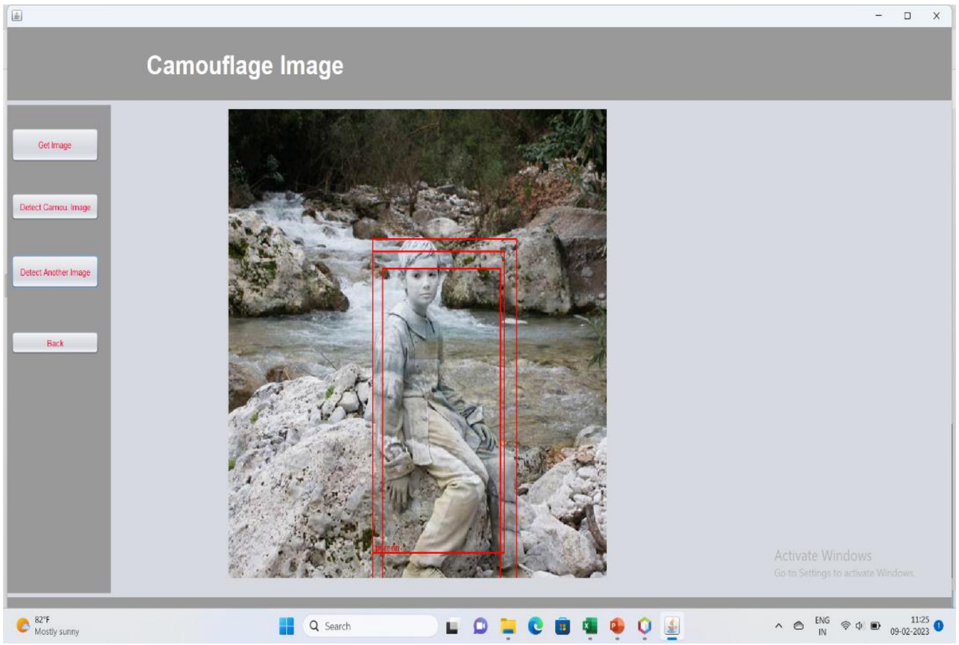


Fig 1: Image from test dataset- showing multiple bounding box

In the modified work, a single bounding box for the image is detected as a camouflage image with the proper class name and confidence score above the bounding box.

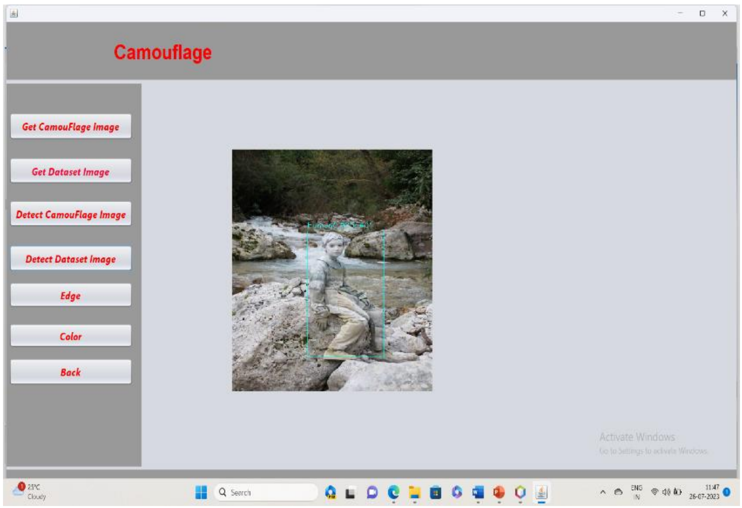


Fig 2: Image from test dataset – showing single bounding box

➢ Small objects have not been detected in the previous work as shown below.

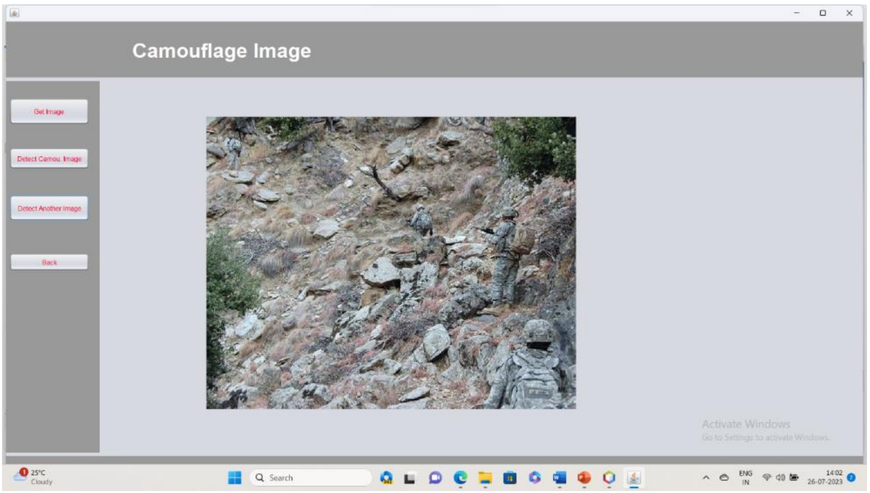


Fig 3: Image from train dataset – small camouflaged objects have not been detected

Small objects have been detected in the modified work as shown below.

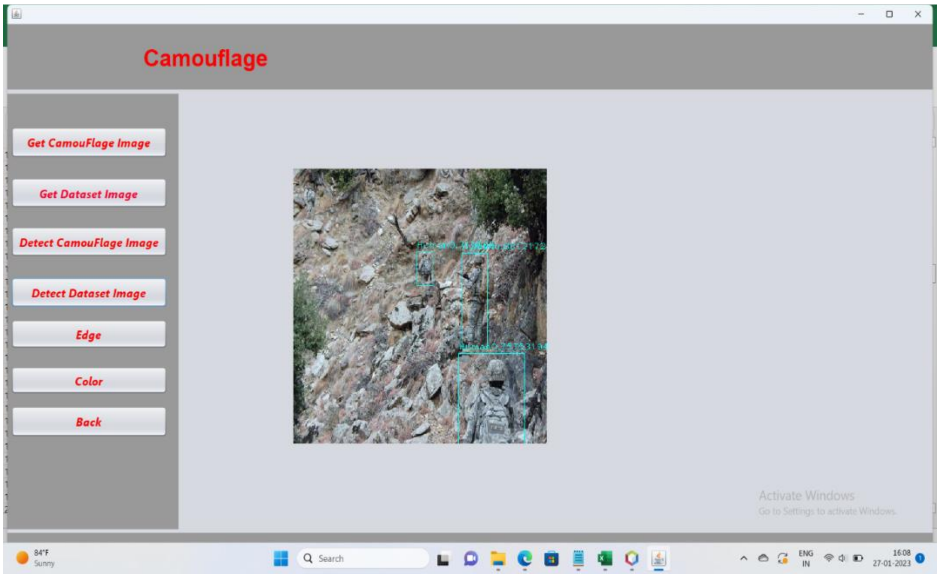


Fig 4: Image from train dataset – small camouflaged objects have been detected

Time required to detect the image the below image is 16817 milliseconds and camouflage detection has not been done here.

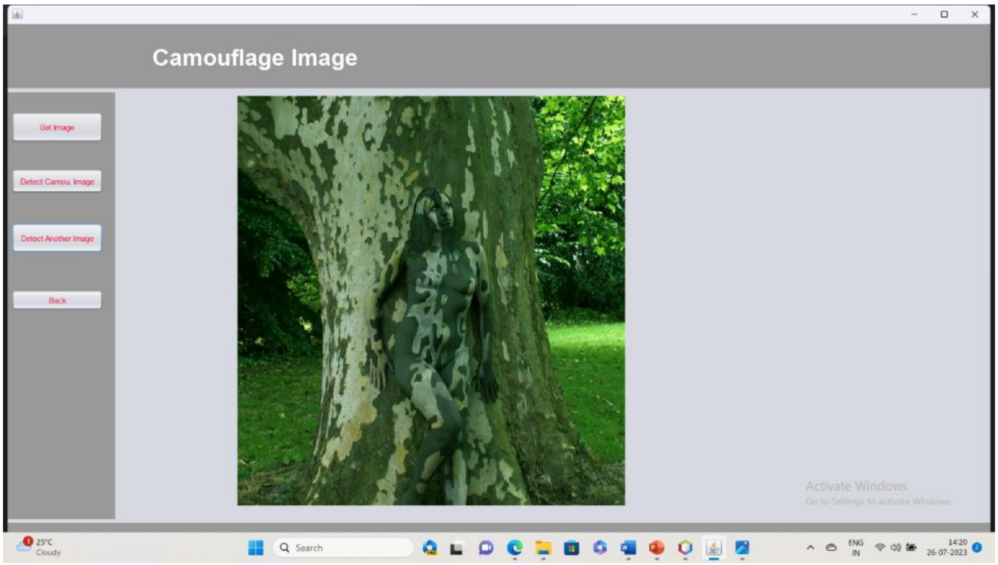


Fig 5: Image from train dataset – detection time required 16817 milliseconds

Time required to detect the same image is 2623 milliseconds in the modified work and correct camouflage detection has been done here.

Fig 6: Image from train dataset – detection time required 2623 milliseconds

The Researcher also provides the count of camouflage objects detected, the class of objects, their confidence score, X and Y coordinates, the width and height of the image, and the number of colours present in the images. For the above image (Fig 6), experimentation shows the following results.

Human = 0.7554915

X:157.0 Width: 116.0 Y: 131.0 Height:272.0

Total Colour are Present:

Red: 33897558

Green: 51789566

Blue: 27069933

5. Conclusion

This work detects camouflage images using YOLO neural network. The algorithm identifies and detects the camouflage image of classes. Experimentation is carried out on training and testing set of datasets. The results shows that the author's previous work provides 76% accuracy and modified work provides 83% accuracy. This technique can be applied to seeing attackers in the field of battle, finding manufacturing flaws in products, and spotting duplicate products in logistics.

References

[1] Meirav Galun, Eitan Sharon, Ronen Basri, Achi Brandt, "Texture Segmentation by Multiscale Aggregation of Filter Responses and Shape Elements", Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV'03), 2003 IEEE

[2] Sable, N.P., Rathod, V.U. (2023). Rethinking Blockchain and Machine Learning for Resource-Constrained WSN. In: Neustein, A., Mahalle, P.N., Joshi, P., Shinde, G.R. (eds) AI, IoT, Big Data and Cloud Computing for Industry 4.0. Signals and Communication Technology. Springer, Cham. https://doi.org/10.1007/978-3-031-29713-7_17.

[3] Nagappa U. Bhajantri and P Nagabhusan, "Camouflage Defect Identification: A Novel Approach", 9th International Conference on Information Technology (ICIT'06)0-7695-2635-7/06 2006 IEEE

[4] V. U. Rathod and S. V. Gumaste, "Role of Deep Learning in Mobile Ad-hoc Networks", IJRITCC, vol. 10, no. 2s, pp. 237–246, Dec. 2022.

[5] I. Huerta, D. Rowe, M. Mozerov, and J. Gonzalez, "Improving Background Subtraction Based on a Casuistry of Color-Motion Segmentation Problems", IbPRIA '07 Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part II Pages 475 - 482 Springer-Verlag Berlin, Heidelberg 2007

[6] N. P. Sable, V. U. Rathod, R. Sable and G. R. Shinde, "The Secure E-Wallet Powered by Blockchain and Distributed Ledger Technology," 2022 IEEE Pune Section International Conference (PuneCon), Pune, India, 2022, pp. 1-5, doi: 10.1109/PuneCon55413.2022.10014893.

[7] P. Sengottuvelan, Amitabh Wahi, A. Shanmugam, “Performance of Decamouflaging Through Exploratory Image Analysis”, ICETET 2008 IEEE.

[8] Hongxing Guo, Yaling Dou, Ting Tian, Jingli Zhou, Shegsheng Yu, “ A Robust Foreground Segmentation Method by Temporal Averaging Multiple Video Frames”, ICALIP 2008 IEEE

[9] P. Siricharon, S. Aramvith, T.H. Chalidabhongse and S. Siddhichai, “Robust Outdoor Human Segmentation based on Color- based Statistics Approach and Edge Combination”, 978-1-4244-6878-2/10 2010 IEEE

[10] R. E. Ch.Kavitha, B.Prabhakara Rao, A.Govardhan, “An Efficient Content Based Image Retrieval Using Color and Texture Of Image Sub blocks”, International Journal of Engineering Science and Technology (IJEST) ISSN: 0975-5462 Vol. 3 No. 2 Feb 2011

[11] Zhou Liu, Kaiqi Huang and Tieniu Tan, “Foreground Object Detection Using Top-down Information Based on EM Framework”, IEEE Transactions on Image Processing 2011

[12] Song Liming and Geng Weidong, “A new camouflage Texture Evaluation Method Based on WSSIM and Nature Image Features Multimedia Technology (ICMT)”, International Conference Page(s): 1 – 4 Cited by 1 IEEE 2010

[13] Yuxin Pan, Yiwang Chen, Qiang Fu, Ping Zhang, and Xin Xu, “Study on the Camouflaged Target Detection Method Based on 3D Convexity”, Modern Applied science, Vol 5, No 4, 2011 Canadian Center of science and education

[14] Jianqin Yin Yanbin Han Wendi Hou Jinping Li, “Detection of the Mobile Object with Camouflage Color under Dynamic Background Based on Optical Flow”, Advanced in control Engineering and Information Science, Elsevier 2011

[15] V. U. Rathod and S. V. Gumaste, "Role of Routing Protocol in Mobile Ad-Hoc Network for Performance of Mobility Models," 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), Lonavla, India, 2023, pp. 1-6, doi: 10.1109/I2CT57861.2023.10126390.

[16] Sujit K Singh ,Chitra A Dhawale and Sanjay Misra, “Survey of Object Detection Methods in Camouflaged Image”, 2013 International Conference on Electronic Engineering and Computer Science, IERI Procedia 4 (2013) 351 – 357, doi: 10.1016/j.ieri.2013.11.050

[17] Trung-Nghia Le, Tam V. Nguyen, Zhongliang Nie, Minh-Triet Tran, Akihiro Sugimoto, “Anabranch network for camouflaged object segmentation”, Computer Vision and Image Understanding 184 (2019), 45–56

[18] Ajeet Ram Pathak, Manjusha Pandey, Siddharth Rautaray, “Application of Deep Learning for Object Detection”, International Conference on Computational Intelligence and Data Science (ICCIDS 2018), Procedia Computer Science 132 (2018), 1706–1717

[19] N. P. Sable, V. U. Rathod, P. N. Mahalle and D. R. Birari, "A Multiple Stage Deep Learning Model for NID in MANETs," 2022 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2022, pp. 1-6, doi: 10.1109/ESCI53509.2022.9758191.

[20] Mohana, HV Ravish Aradhya, “Object Detection and Tracking using Deep Learning and Artificial Intelligence for Video Surveillance Applications”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 12, 2019

[21] Rathod, V.U. and Gumaste, S.V., 2022. Role of Neural Network in Mobile Ad Hoc Networks for Mobility Prediction. International Journal of Communication Networks and Information Security, 14(1s), pp.153-166.

[22] Mr. Abhijeet Tanaji Khot, Mr. Chetan J. Awati, Ms. Sonam S. Kharade, “Image Transmutation in Reversible Manner for Camouflage Image”, Third International Conference on Computing, Communication, Control And Automation (ICCUBEA, 2017 IEEE

[23] Shuai Li, Dinei Florencio, Yaqin Zhao, Chris Cook, Wanqing Li, “Foreground Detection In Camouflaged Scenes”, ICIP, 2017 IEEE

[24] N. P. Sable, M. D. Salunke, V. U. Rathod and P. Dhotre, "Network for Cross-Disease Attention to the Severity of Diabetic Macular Edema and Joint Retinopathy," 2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), Bangalore, India, 2022, pp. 1-7, doi: 10.1109/SMARTGENCON56628.2022.10083936.

[25] N. P. Sable, V. U. Rathod, P. N. Mahalle, and P. N. Railkar, “An Efficient and Reliable Data Transmission Service using Network Coding Algorithms in Peer-to-Peer Network”, IJRITCC, vol. 10, no. 1s, pp. 144–154, Dec. 2022.

[26] Vijay U. Rathod* & Shyamrao V. Gumaste, “Effect Of Deep Channel To Improve Performance On

Mobile Ad-Hoc Networks", J. Optoelectron. Laser, vol. 41, no. 7, pp. 754–756, Jul. 2022.

[27] Heng Yao, Xiaokai Liu, Zhenjun Tang, Yu-Chen Hu, Chuan Qin, "An Improved Image Camouflage Technique Using Color Difference Channel Transformation and Optimal Prediction-Error Expansion", Volume 6, 2018

[28] Przemysław Skurowski, Paweł Kasprowski, "Evaluation of Saliency Maps in a Hard Case – Images of Camouflaged Animals", International Conference on Image Processing, Applications and Systems (IPAS), 2018 IEEE

[29] Piyush K. Sharma, Adrienne Raglin, "Image-Audio Encoding for Information Camouflage and Improving Malware Pattern Analysis", 17th IEEE International Conference on Machine Learning and Applications, 2018

[30] N. P. Sable, R. Sonkamble, V. U. Rathod, S. Shirke, J. Y. Deshmukh, and G. T. Chavan, "Web3 Chain Authentication and Authorization Security Standard (CAA)", IJRITCC, vol. 11, no. 5, pp. 70–76, May 2023.

[31] Mr. Kaustubh Patil, Promod Kakade. (2014). Self-Sustained Debacle Repression Using Zig-Bee Communication. International Journal of New Practices in Management and Engineering, 3(04), 05 - 10. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/32

[32] G, A. ., K, S. ., S, B. ., M, B. ., & M, P. . (2023). Power Consumption and Carbon Emission Equivalent for Virtualized Resources – An Analysis: Virtual Machine and Container Analysis for Greener Data Center. International Journal on Recent and Innovation Trends in Computing and Communication, 11(1), 110–116. https://doi.org/10.17762/ijritcc.v11i1.6057

[33] Dhingra, M., Dhabliya, D., Dubey, M. K., Gupta, A., & Reddy, D. H. (2022). A review on comparison of machine learning algorithms for text classification. Paper presented at the Proceedings of 5th International Conference on Contemporary Computing and Informatics, IC3I 2022, 1818-1823. doi:10.1109/IC3I56241.2022.10072502 Retrieved from www.scopus.com