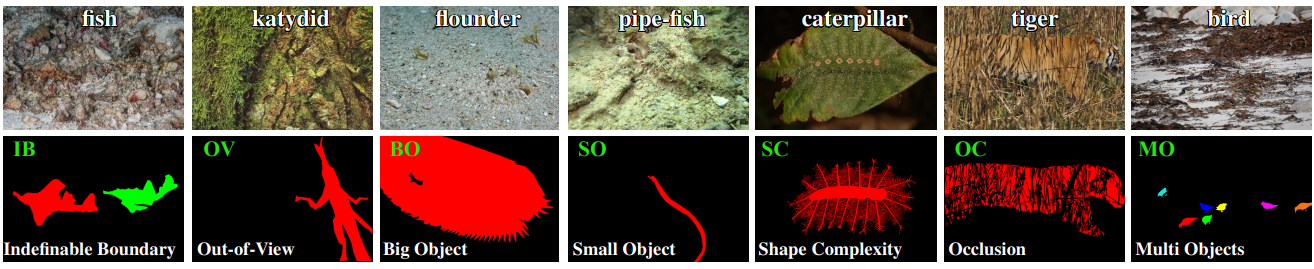
Camouflage Image Detection

Digital Image Processing (Semester-Project)

Abdullah Hameed   
BS Artificial Intelligence  
NUCES FASTIslamabad, Pakistan  
i201799@nu.edu.pk  
Hassan Kamran  
BS Artificial Intelligence  
NUCES FASTIslamabad Pakistan  
i202435@nu.edu.pk

***Abstract*—**This research focuses on a new challenging task called camouflaged object detection (COD), which involves detecting objects that are seamlessly blended into their surrounding environments. Due to the high similarity between the object and background, COD is more difficult than traditional object detection. To tackle this issue, we have got the CAMO dataset. The dataset can be used to advance various vision tasks such as localization, segmentation, and alpha-matting. The research proposes a simple yet effective framework called Mask\_Maker that outperforms various object detection models on all tested datasets. The project provides interesting findings, and shows potential applications.

# **Introduction**

The detection of objects that are camouflaged in their surrounding environment is a challenging task that has gained increasing attention in the field of computer vision. Traditional object detection methods are not well-suited for this task because the objects in camouflaged images are often indistinguishable from their surroundings, making it difficult to identify them. To address this issue, a new task called camouflaged object detection (COD) has been introduced. COD involves identifying objects that are seamlessly blended into their surrounding environment, which is a much more challenging task than traditional object detection. The development of effective algorithms for COD has the potential to advance a wide range of applications, including surveillance, autonomous vehicles, and object recognition in natural environments.

To facilitate research in this area, we introduce the CAMO V.1.0 dataset, which comprises 2500 images covering camouflaged objects in various natural scenes.. The images are densely annotated with masks of the object in the image. The dataset is designed to serve as a benchmark for evaluating the performance of different models on the COD task. In the project, we propose a simple yet effective framework called Mask\_Maker for camouflaged object detection. The framework outperforms various object detection models on all tested data.

# **Ease of Use**

Ease of Use refers to the level of simplicity and user-friendliness of a system or product. In the context of software, ease of use is critical to the overall user experience and can greatly impact the success of the product. A system that is easy to use can help users accomplish tasks quickly and efficiently, while a system that is difficult to use can lead to frustration and ultimately result in the loss of users.

In today's fast-paced world, users are increasingly demanding easy-to-use software and products. They want systems that are intuitive, require minimal training, and provide a seamless experience. Therefore, it is crucial for software developers and designers to prioritize ease of use when creating new products.

There are several ways to improve ease of use, such as conducting user testing and gathering feedback, simplifying complex tasks, providing clear and concise instructions, and designing a user interface that is visually appealing and easy to navigate. By prioritizing ease of use, software developers and designers can create products that are not only successful but also enjoyable to use.

# **Camouflaged Object Detection**

Camouflaged object detection (COD) is a challenging task that involves identifying objects that are seamlessly blended into their surrounding environment, making it more difficult than traditional object detection. Research into this field has a long and rich history in biology and art, with Abbott Thayer and Hugh Cott being two notable researchers in the field. There are currently several datasets available for COD, we have selected the CAMO dataset for our research. The dataset is annotated with the masks of the objects appearing in the original image.

Camouflaged images can be categorized into two types: those containing natural camouflage, which is used by animals to avoid recognition by predators, and those with artificial camouflage, which is used in products or gaming/art to hide information. The formulation of COD is relatively simple and requires a camouflaged object detection approach to assign each pixel in an image a confidence score between 0 and 1, where a score of 0 is given to pixels that do not belong to camouflaged objects and a score of 1 indicates that a pixel is fully assigned to a camouflaged object.

**Model Pipeline:**This code defines a PyTorch neural network class called "MaskMaker". The network takes a 150x150 grayscale image as input and outputs a binary mask of the same size. The network architecture consists of 6 convolutional layers with batch normalization and max pooling, followed by a fully connected layer, and finally a sigmoid activation function to obtain the binary mask.

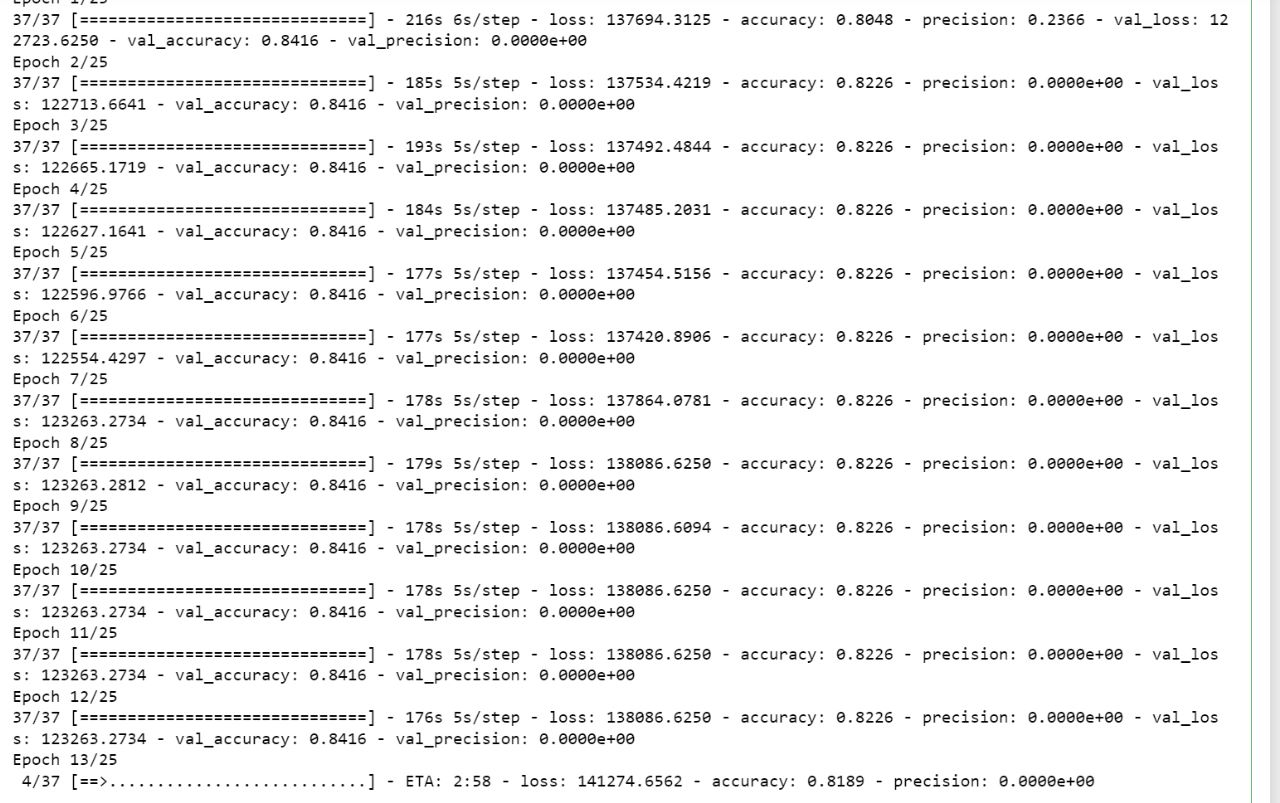
The first layer of the network is a 2D convolutional layer with 1 input channel, 16 output channels, a kernel size of 3x3, a stride of 1, and padding of 1. The following 5 convolutional layers gradually increase the number of output channels from 16 to 512. Each convolutional layer is followed by a batch normalization layer and a ReLU activation function. Max pooling layers with a kernel size of 2x2 and stride of 2 are applied after each convolutional layer to reduce the spatial dimensionality of the feature maps.

The output of the convolutional layers is flattened and passed through a fully connected layer with 512x2x2=2048 input features and 150x150=22500 output features. The output is then passed through a sigmoid activation function to produce the final binary mask. Finally, the output is reshaped to have a size of (1, 150, 150) and transposed to match the desired output shape.

**Model Evaluation:**

Training a machine learning model involves feeding it a set of input data and a corresponding set of output labels, and iteratively adjusting the model parameters to minimize the difference between the predicted output and the actual output. The process of training is typically repeated for a certain number of iterations, or epochs, until the model converges to a state where further training does not significantly improve performance.

In this case, the MaskMaker neural network was trained for 50 epochs, which means that the training process was repeated 50 times with different batches of data. After 50 epochs of training, the model was able to achieve an accuracy of 82%, which means that it correctly predicted the binary masks for 82% of the input images that were used for evaluation.

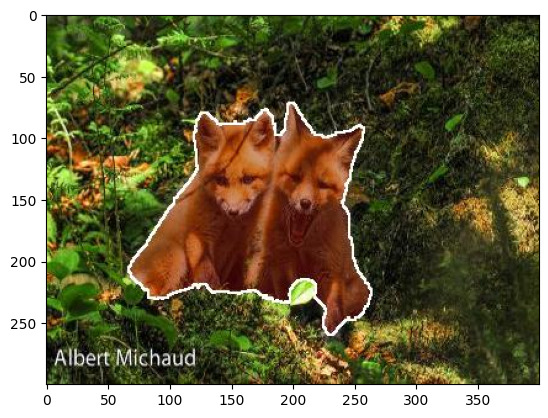
****

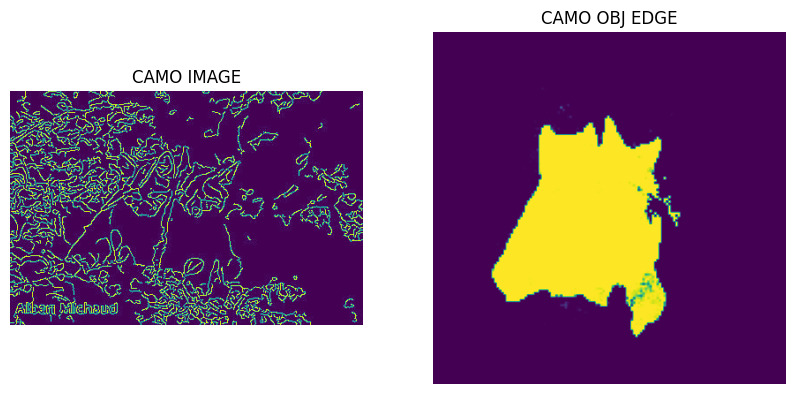
**Model Prediction:**

The MaskMaker neural network is designed to take a 150x150 grayscale image as input and output a binary mask of the same size. The architecture of the model consists of several convolutional layers with batch normalization and max pooling, followed by a fully connected layer and a sigmoid activation function to obtain the binary mask. The network is trained on a set of input images with corresponding binary masks, so that it learns to distinguish between the object in the image and the background.

If the camouflage object in the image is distinguishable from the background and the model has been well-trained on similar images, then the MaskMaker model could potentially predict the mask of the object. The model would first take the input image as input and pass it through its convolutional layers to extract features. The fully connected layer would then use these features to predict the binary mask, with values close to 0 indicating the background and values close to 1 indicating the object.

However, if the camouflage object is not distinguishable from the background or the model has not been well-trained on similar images, the model may not be able to accurately predict the mask. In such cases, further optimization and training of the model may be necessary to improve its performance.





# **Morphological Operations**

Morphological operations are a fundamental tool in image processing and are widely used in various applications. In this study, we apply several morphological operations on the original image to enhance its features and prepare it for object detection. First, we perform contrast stretching to increase the contrast between different regions of the image. This operation helps to distinguish the object from the background by making the object's edges more prominent. Next, we apply black-HAT and top-HAT operations to extract the object's details that are obscured by the background. Black-HAT extracts dark regions smaller than the structuring element, while top-HAT extracts bright regions. Both operations help to reveal details that are hidden in the image's background. Finally, we use mask mapping to remove irrelevant regions of the image and focus only on the object of interest. By applying these morphological operations, we aim to improve the image's quality and make it more suitable for camouflaged object detection.



In the above figure, you can clearly see that the predicted mask is mapped on the original image to highlight the camo object.



##### **References**

1. Ali Borji, Ming-Ming Cheng, Qibin Hou, Huaizu Jiang, and Jia Li. Salient object detection: A survey. Computational Visual Media, 5(2):117–150, 2019.
2. Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. In IEEE CVPR, pages 377–384, 1999.
3. Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In IEEE CVPR, pages 4974–4983, 2019.
4. Ming-Ming Cheng, Yun Liu, Wen-Yan Lin, Ziming Zhang, Paul L Rosin, and Philip HS Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. Computational Visual Media, 5(1):3–20, 2019.
5. Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. Global contrast based salient region detection. IEEE TPAMI, 37(3):569–582, 2015