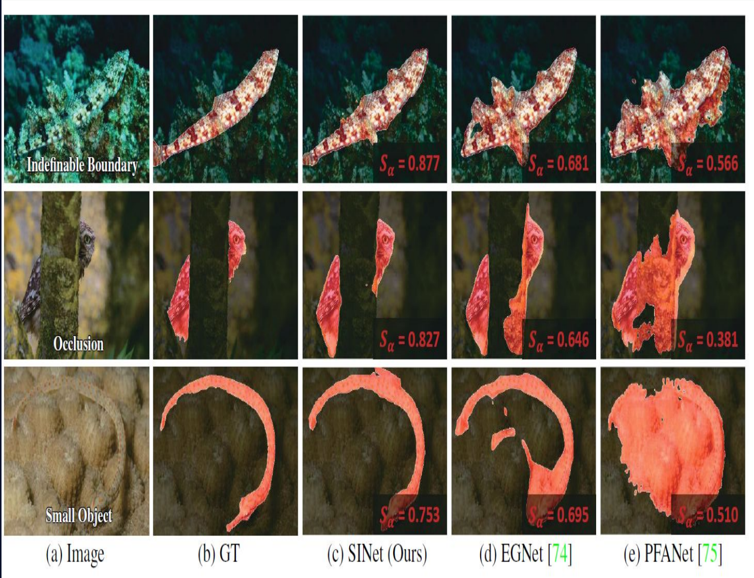
Camouflaged Object Detection

GNR 638 Course Project

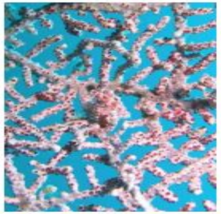
Shubham Lohiya (18D100020)

Aditya Iyengar (180100007)

Sharvaree Sinkar (19D070058)



Contents



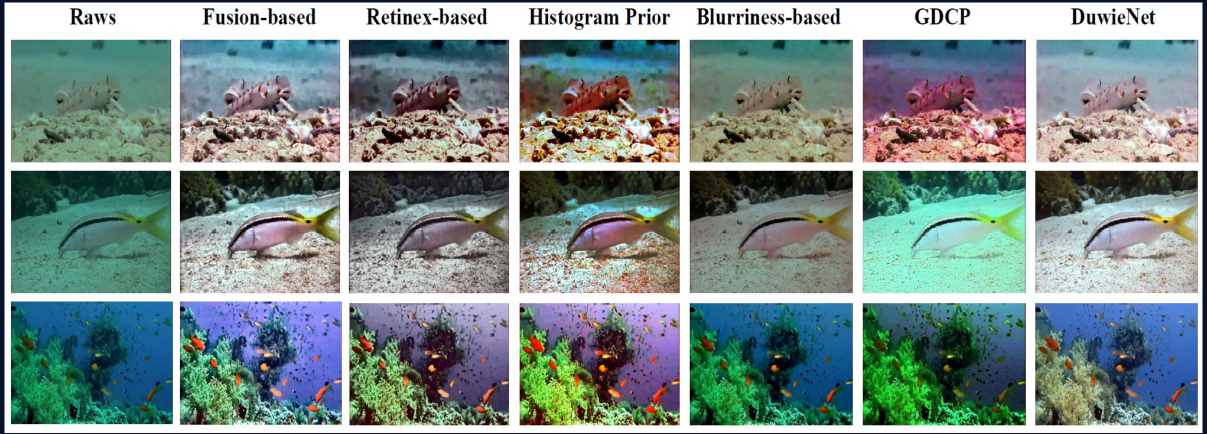
Problem Statement:
Camouflaged Object Detection

To precisely search and identify camouflaged objects.
To overcome the challenge of object detection despite of being high intrinsic similarities between target object and background.

Background Study

- The oldest recorded studies in camouflaged objects come from papers by Thayer and Cott classifying camouflage into natural (e.g. animals) and artificial (e.g. adulterants, defective products) kinds.
- The most well-known datasets prior to *COD10K* were the unpublished *CHAMELEON* and *CAMO*, however both provided few images of acceptable quality and were sparsely annotated.
- Detection of camouflaged objects was performed at a pixel-level, by assigning each pixel with a confidence probability. The higher the probability, the greater are the chances of the pixel belonging to a camouflaged object.
- The MAE metric, while popular with salient object detection, cannot judge structural similarities, hence the need for a new evaluation metric.

COD10K Dataset



- Contains 10K images covering 78 camouflaged object categories such as aquatic, flying, amphibians, terrestrial etc.
    - 6K training images
    - 4K testing images

COD10K Dataset

- All the camouflaged images are hierarchically annotated (taxonomic system) as:
  - Category
  - Bounding box
  - Attribute
  - Object/Instance
- Facilitates many vision tasks, such as localization, object proposal, semantic edge detection, task transfer learning etc.
- Each camouflaged image is assigned challenging attributes and matting level which provides deeper insights into the algorithm

Major Contributions by the Paper

- Carefully assembled the COD10K dataset:
  - 10K images (78 categories)
- Used two existing datasets + collected COD images, rigorous evaluation of 12 state-of-the-art baselines making largest COD study ever
- Proposed SINet framework which outperformed all existing methods
  - Provided potential solution to the highly challenging problem of camouflaged object detection

The Project Pipeline

Data Loading
+
Pre-processing

The Architecture:
SINet
Framework

Training +
Testing
Results

Data Loading + Data Pre-processing

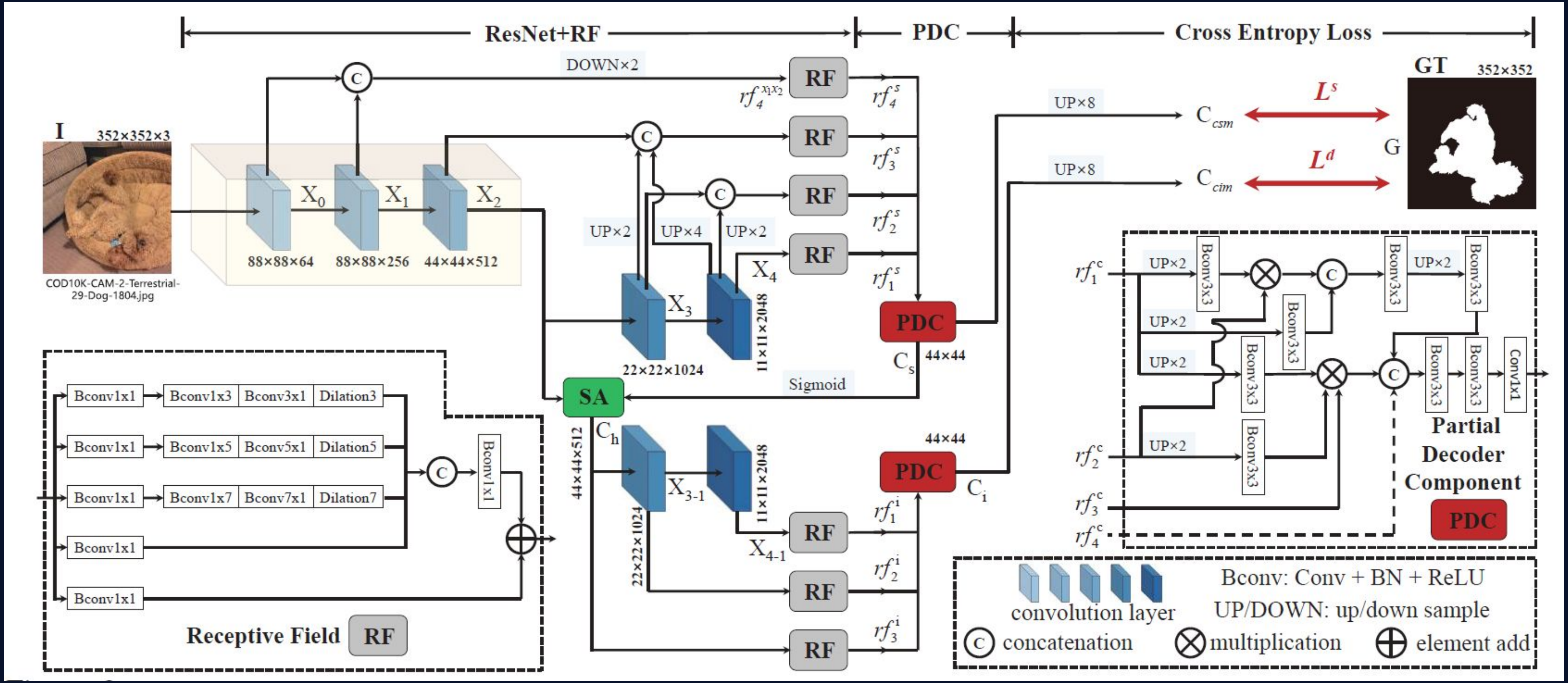
- A PyTorch DataLoader is defined to collate the training as well as test datasets into separate iterable *dataloader* objects.
- The images are then resized, converted to a PyTorch tensor and normalized.
- Images are further processed into one of two forms:
  - Colour image to 3-channel RGB image
  - Colour image to single channel grayscale image (“L” mode)

The Architecture

- SINet framework
  - Search Module(search of camouflaged objects)
  - Identification Module(precise detection)

    (Inspired by hunting!)
- Two Components
  - Receptive Field Module
  - Partial Decoder Component

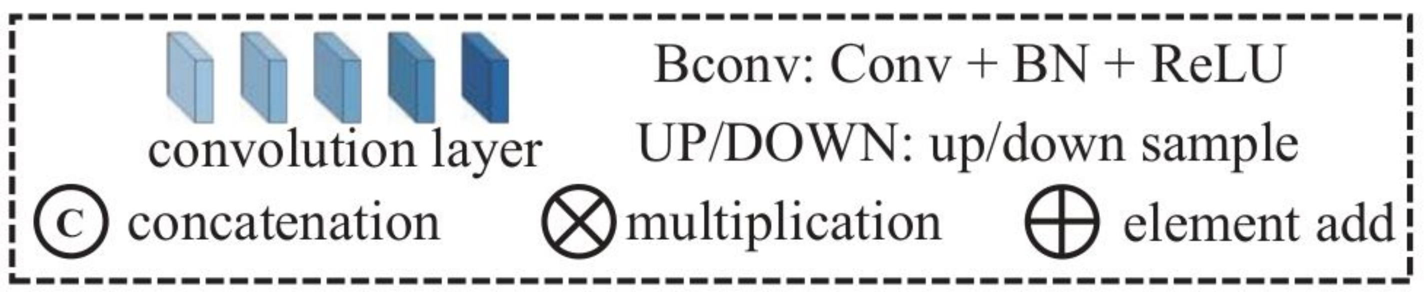
Switcher: Search Attention Module



SINet Architecture

Define: The Bconv Operation

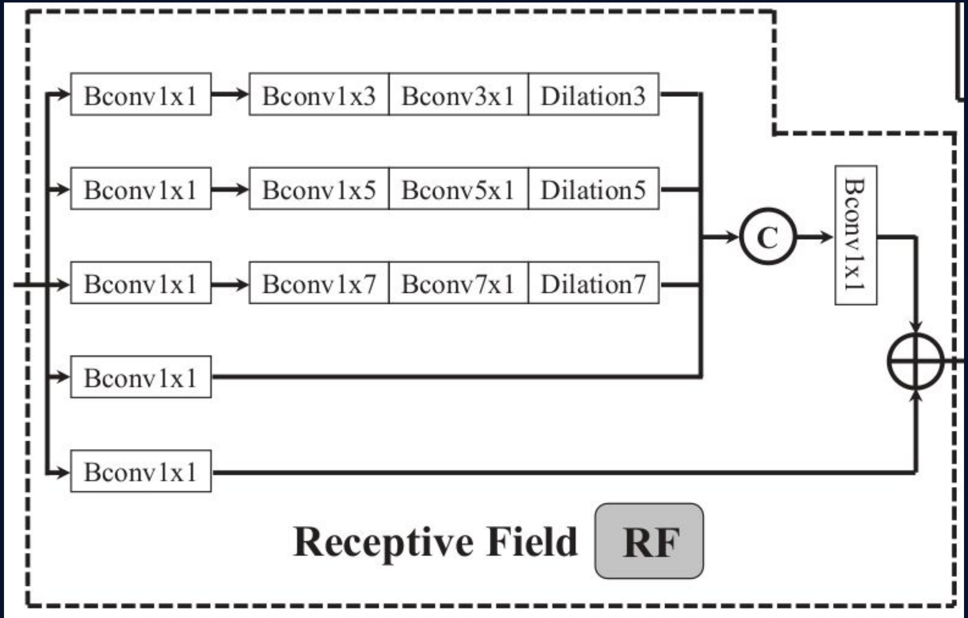
- **BConv:**
  - **Convolutional layer**
  - **Batchnorm layer**
  - **ReLU activation**



The Search Module

- Uses **Receptive Field component** to incorporate more discriminative feature representations during the searching stage
- For input image, set of features extracted from **ResNet50**:
  - X0, X1 : Low level features
  - X2 : Middle level features
  - X3, X4 : High level features
- Extracted features concatenated, upsampled, downsampled to form dense net
- Set of enhanced features are obtained after feeding into RF component for learning robust cues

Receptive Field Component

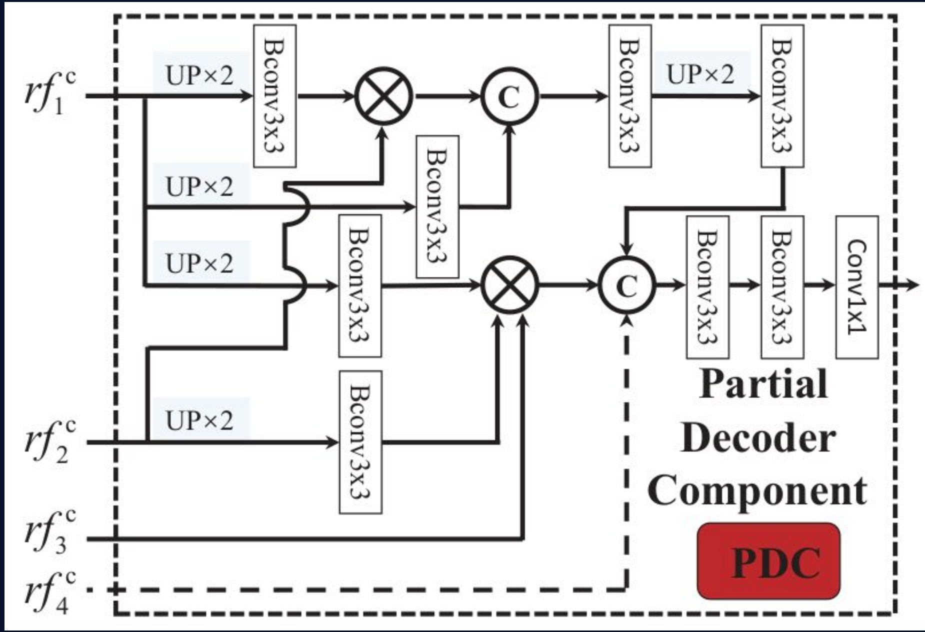


- Includes 5 branches: b_k,(k=1,...,5)
- In each branch:
  - First conv. layer has dimensions as 1x1 to reduce the channel size to 32
  - Followed by (2k-1)x(2k-1) Bconv layer & dilation = 2k-1
- b_1, b_2, b_3, b_4: concatenated
- b_5 added and model fed to ReLU

The Identification Module

- Uses **Partial Decoder component(PDC)** to precisely detect candidate features obtained from previous search module
- PDC:
  - Integrates 4 levels of features from Search Module
  - Obtains coarse camouflaged map C_s
- Switcher **Search Attention Module** Introduced which:
  - Enhances middle level features X2
  - Effectively eliminates interference from irrelevant features

Partial Decoder Component(PDC)



- New features generated from existing features coming from search and identification stages
- Element-wise multiplication adopted to decrease the gap between adjacent features

Partial Decoder Component(PDC)

- **The features coming from search and identification stages are given as:** $\{rf_k^c, k \in [m, \ldots, M], c \in [s, i]\}$
- **New features generated from PDC:** $\{rf_k^{c1}\}$
- **Shallow features:** $rf_M^{c1} = rf_M^{c2}$ when $k = M$.
- **Deeper features:** $rf_k^{c1}, k < M$, we update it as rf_k^{c2}:

$$rf_k^{c2} = rf_k^{c1} \otimes \Pi_{j=k+1}^{M} Bconv(UP(f_j^{c1})),$$

where $k \in [m, \ldots, M-1]$.

Search Attention Module

- Coarse camouflaged map C_s(from search module): $C_s = PD_s(rf_1^s, rf_2^s, rf_3^s, rf_4^s)$, where $\{rf_k^s = rf_k, k = 1, 2, 3, 4\}$
- SA module enhances features as: $C_h = f_{max}(g(\mathcal{X}_2, \sigma, \lambda), C_s)$,
    - g(.) : SA function(which is a gaussian filter with kernel_size = 4 and dev.=32)
    - f_max : maximum function that highlights initial camouflaged regions
- To obtain high level features, PDC is aggregated to another three levels of features, enhanced with RF to obtain final camouflaged map C_i:

$C_i = PD_i(rf_1^i, rf_2^i, rf_3^i)$. where $\{rf_k^i = rf_k, k = 1, 2, 3\}$

Loss Function

C_{csm}

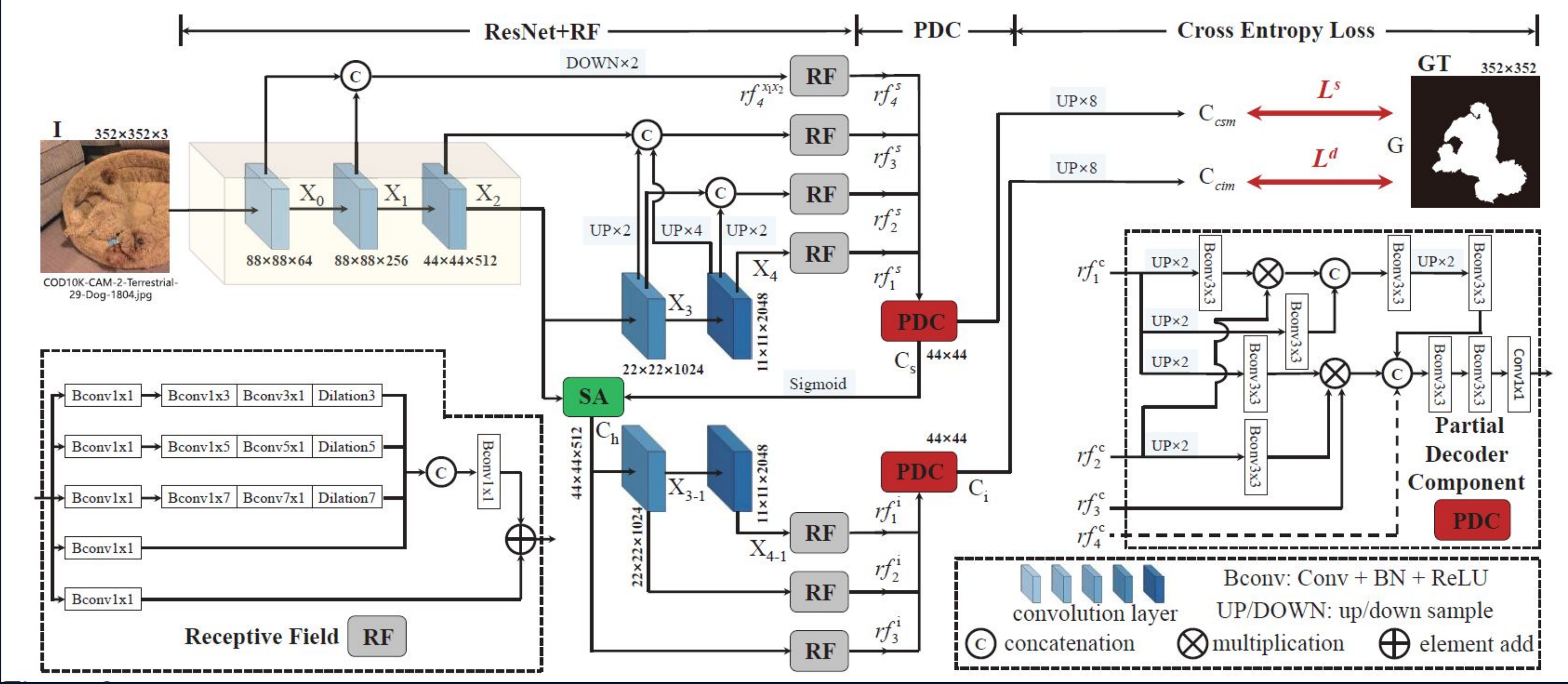
Camouflaged map obtained after upsampling C_s to a resolution of 352x352

C_{cim}

Camouflaged map obtained after upsampling C_i to a resolution of 352x352

$$L = L^{s}_{CE}(C_{csm}, G) + L^{i}_{CE}(C_{cim}, G).$$

Summarizing SINet Framework



Training and Testing Results:

Training

- Trained models (base and modified) for 20-40 epochs
- Achieved MAE on base model of about 0.02 on the train set in only 20 epochs (computational constraints)
- Achieved MAE on modified model (modified for speed) of about 0.027 on the train set in 40 epochs

Test

- Achieved a MAE of 0.091 averaged on test set using base model in only 20 epochs
- Achieved a MAE of 0.094 averaged on test set using modified model in 40 epochs

Modifications

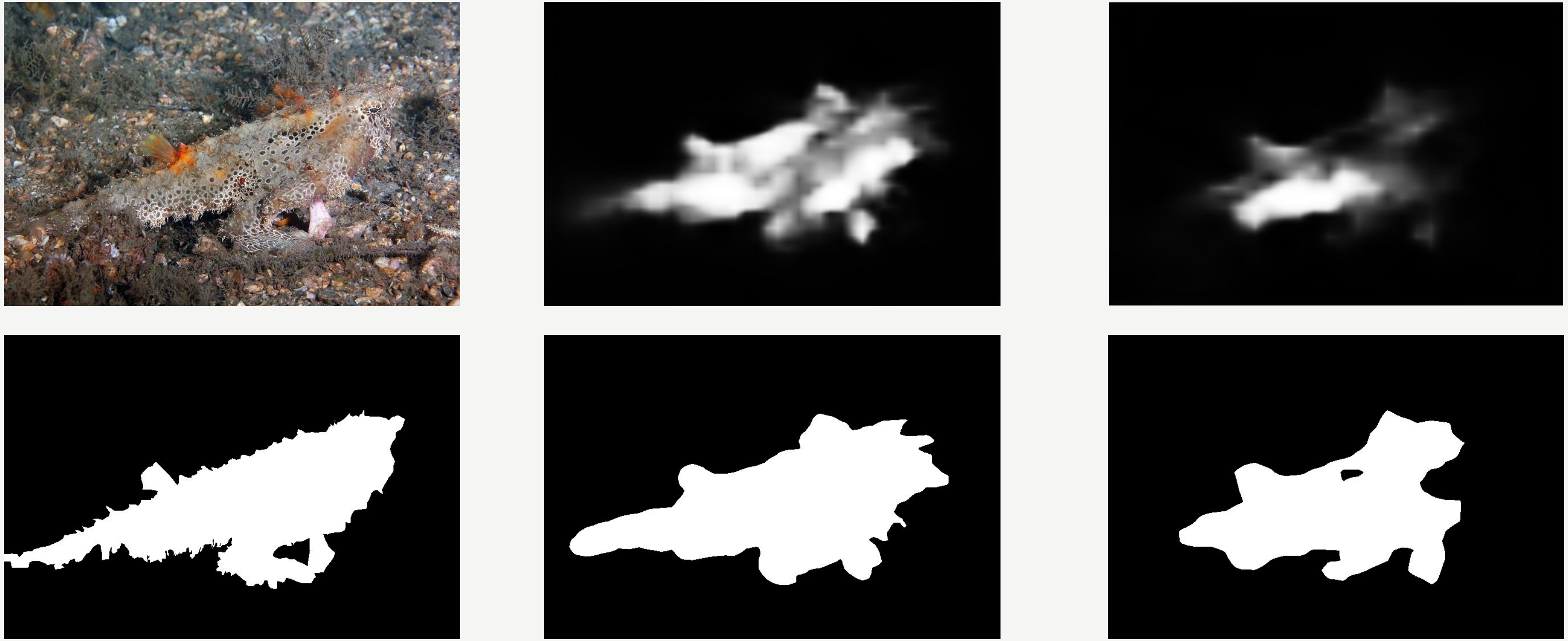
Proposed modifications

- Decrease resolution of input images to speeden up the training process while still achieving our goal of detecting and localizing the camouflaged object
- Doubling the number of channels to 64 in the RF module (lead to exploding gradients)
- Increasing the depth of the network (hurdle of computational resources)

Modification 1 showed promising results

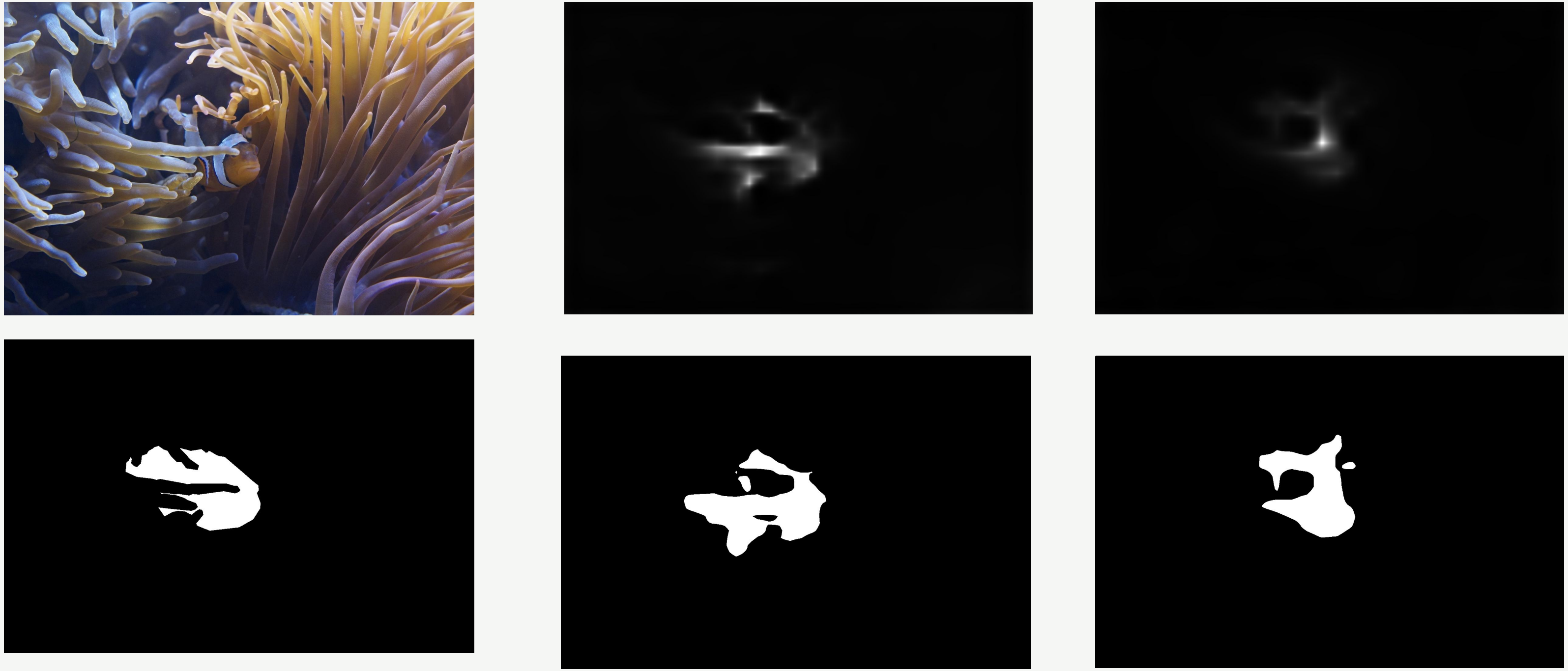
Results

Camouflaged testing img　　base model　　modified



Results

Camouflaged testing img	base model	modified



Results

Camouflaged testing img

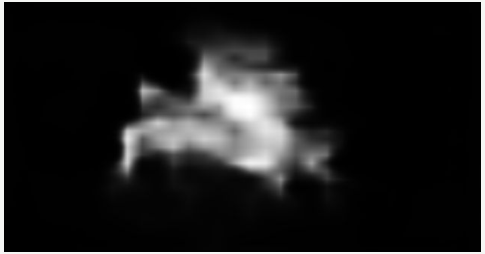
base model

modified

Results

Camouflaged testing img

base model



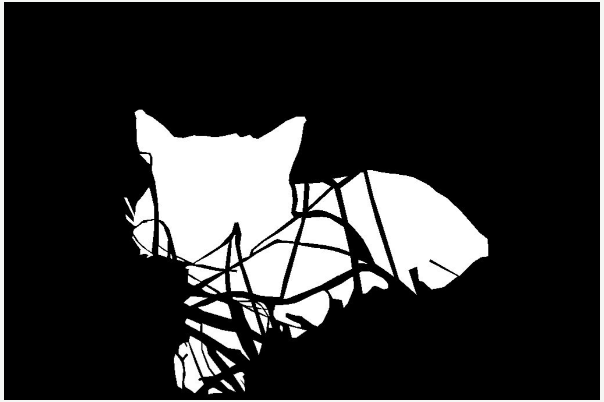
modified

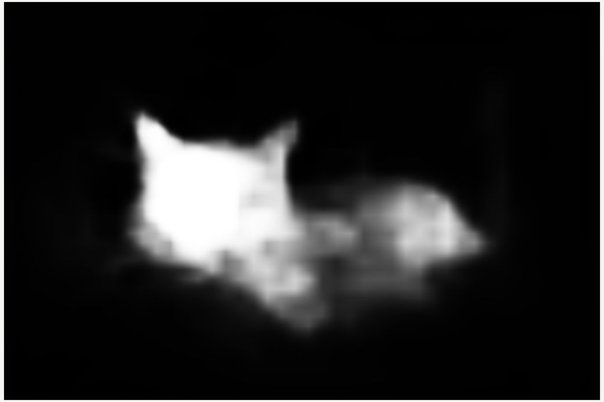
Results

Camouflaged testing img

ground truth

base model



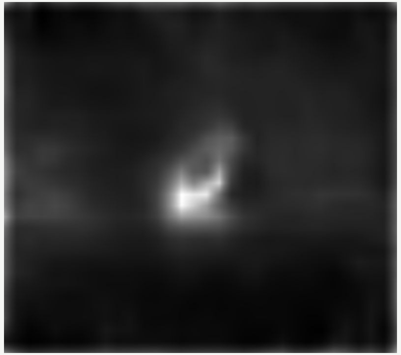


Results

Non -Camouflaged test img

base model

modified



References:

- Link to the original paper: Camouflaged Object Detection(CVPR 2020)
- Dataset: COD10K
- Original Github Repository: SINet
- ResNet50: ResNet50

Contributions

- Shubham Lohiya:
  - Dataset acquirement + extraction
  - Training base model
  - Training modified models
  - Evaluation of models on test sets
  - Analysis of evaluation data
  - Post-processing of results

Contributions

- Aditya Iyengar:
    - Shortlisting the paper and finalizing the problem statement
    - Background study by reviewing similar papers
    - Creating the data loader for training and testing data
    - Comparison and implementation of various techniques for preprocessing
    - Training with several loss functions to identify the best fit
    - Performed systematic documentation for the entire code

Contributions

- Sharvaree Sinkar:
  - Created modified ResNet50 backbone code
  - Wrote code for SINet framework from scratch
  - Analysis of Search and Identification Modules
  - Study of Partial Decoder Component and Receptive Field
  - Examined relations between PDC and RF
  - Entire Analysis of SINet architecture
  - Made Presentation slides