Instance-Level Salient Object Segmentation

...

Aniket Joshi (20161166)
Prakyath Madadi (20161236)
Vashist Madiraju (20161222)

Objective

The main objective of this project is to implement a salient instance segmentation method that produces a saliency mask with distinct object instance labels for an input image.

Motivation

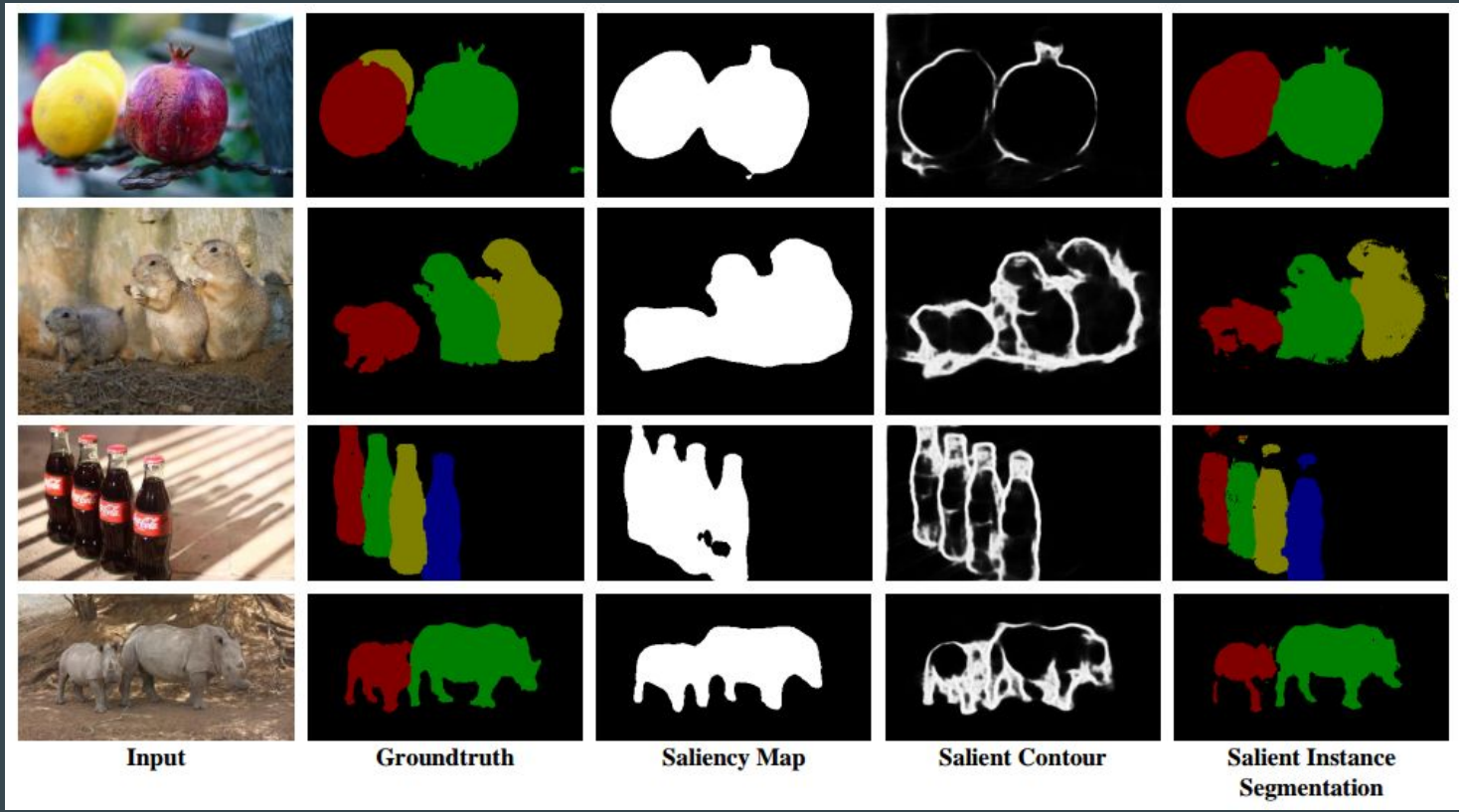
The main motivation behind this project is that most of the current instance segmentation methods are unable to find the object instances from the detected saliency regions and this method. These image object instances are important for object recognition and improving the vision based pipelines.

Overview

What is Instance-Level Salient Object Detection?

- ➔ Salient object detection attempts to locate the most noticeable and eye-attracting object regions in images.
- ➔ It is a fundamental problem in computer vision and serves as a pre-processing step to facilitate a wide range of vision applications.
- ➔ This project tackles a more challenging task, instance-level salient object segmentation, which aims to identify individual object instances in the detected salient regions.

Expected Results



Steps Involved

1) Estimating binary saliency map

In this sub-task, a pixel-level saliency mask is predicted,indicating salient regions in the input image.

2) Detecting salient object contours

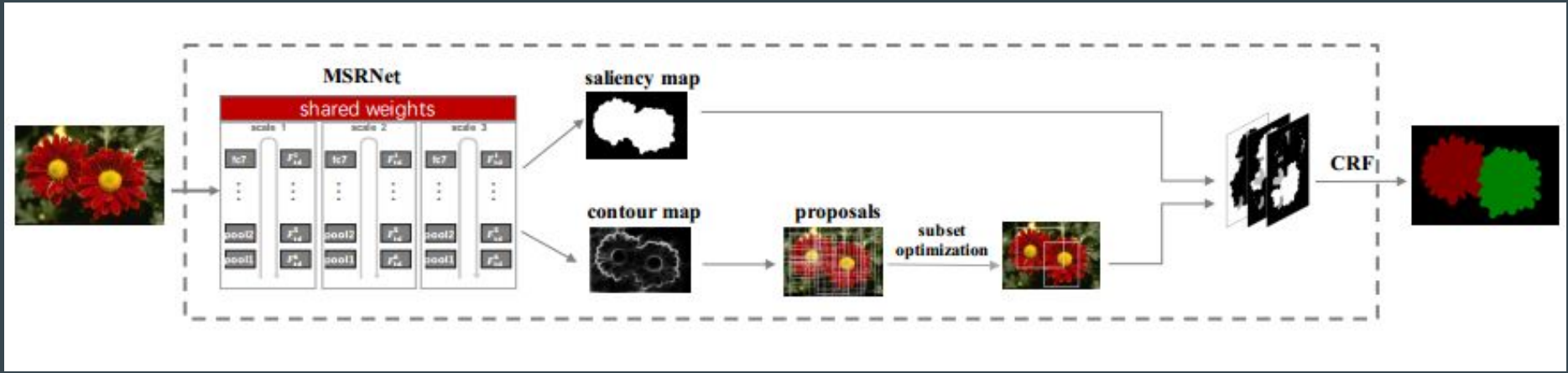
In this sub-task, we perform contour detection for individual salient object instances.

3) Identify salient object instances

In this sub-task, salient object proposals are generated, and a subset of salient objects proposals are selected to cover the salient regions.

CRF based refinement method is applied to improve the spatial coherence.

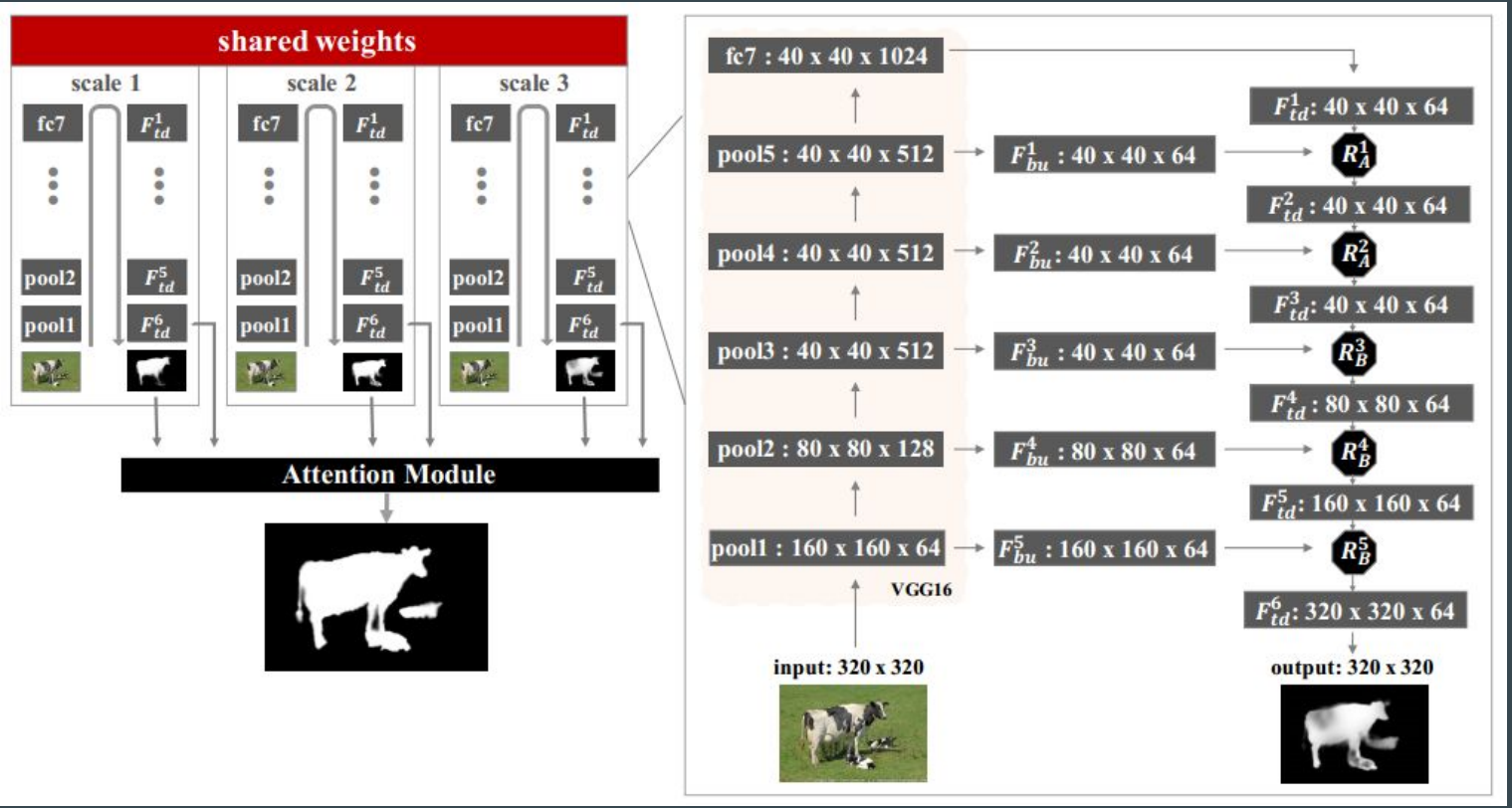
Algorithm Pipeline



Binary Saliency Map and Object Contouring

- We will use a deep multi-scale refinement network (MSRNet), which can generate very accurate results for both salient region detection and object contour detection.
- The deep network consists of three parallel streams processing scaled versions of the same input image and a learned attention model to fuse results at different scales from the three streams. The three streams share the same network architecture, a refined VGG network, and its associated parameters. This refined VGG network is designed to integrate the bottom-up and top-down information in the original network.
- MSRNet can not only integrate bottom-up and top-down information for saliency inference but also attentionally determine the pixel-level weight of each salient map by looking at different scaled versions of the same image.

MSRNet Architecture



MSRNet Features

- Bottom-Up Network :
    - We use a modified VGG network. It takes low level features (pixels) as input, such as colours and texture, and propagates it up through the layers.
    - Information from an input image needs to be passed from the bottom layers up in a deep network before being transformed into high-level semantic information.
- Top-Down Network :
    - We use a top down model to use and incorporate high level semantic information.
    - It is propagated from the top layers further down, and is integrated with low-level information obtained from the intermediate stages of the Bottom-up network.
    - This integration of high level information with low level features, results in high precision contour detection results.

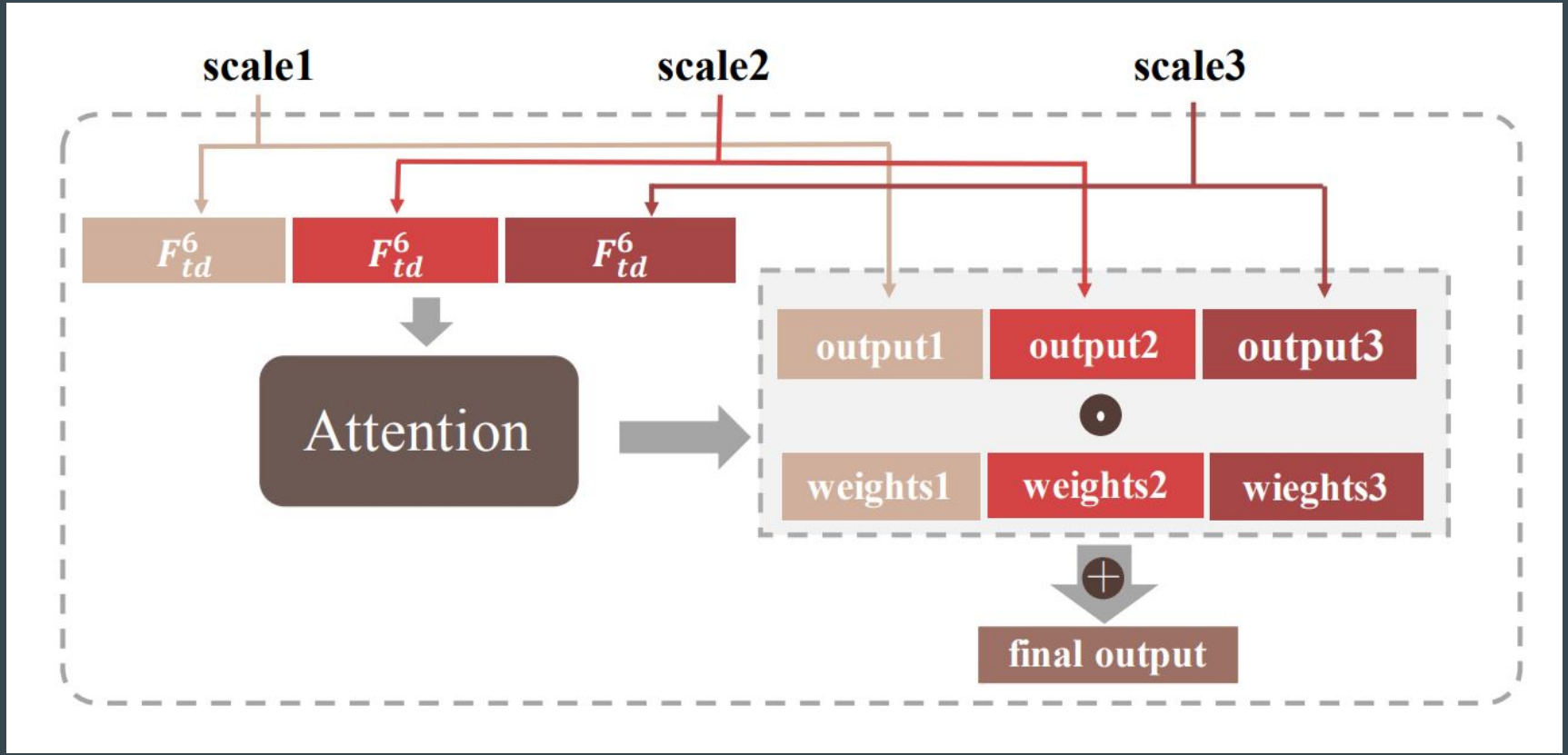
MSRNet Features

- We transform the original VGG16 into a Fully Convolutional Network, which serves as our bottom-up backbone network. This is done by modifying the final layers.
- FCNs are used to produce semantic segmentation. The output is of the same size as the original input image, but with different number of channels.
- FCNs use “deconvolutions”, or essentially backwards convolutions, to upsample the intermediate tensors so that they match the width and height of the original input image.
- The upsampled output from the final convolutional tensor seemed to be inaccurate.
    - Too much spatial information had been lost by all the downsampling in the network.
    - Upsampled output from that final intermediate tensor is combined with upsampled output from earlier tensors, to get more precise spatial information.

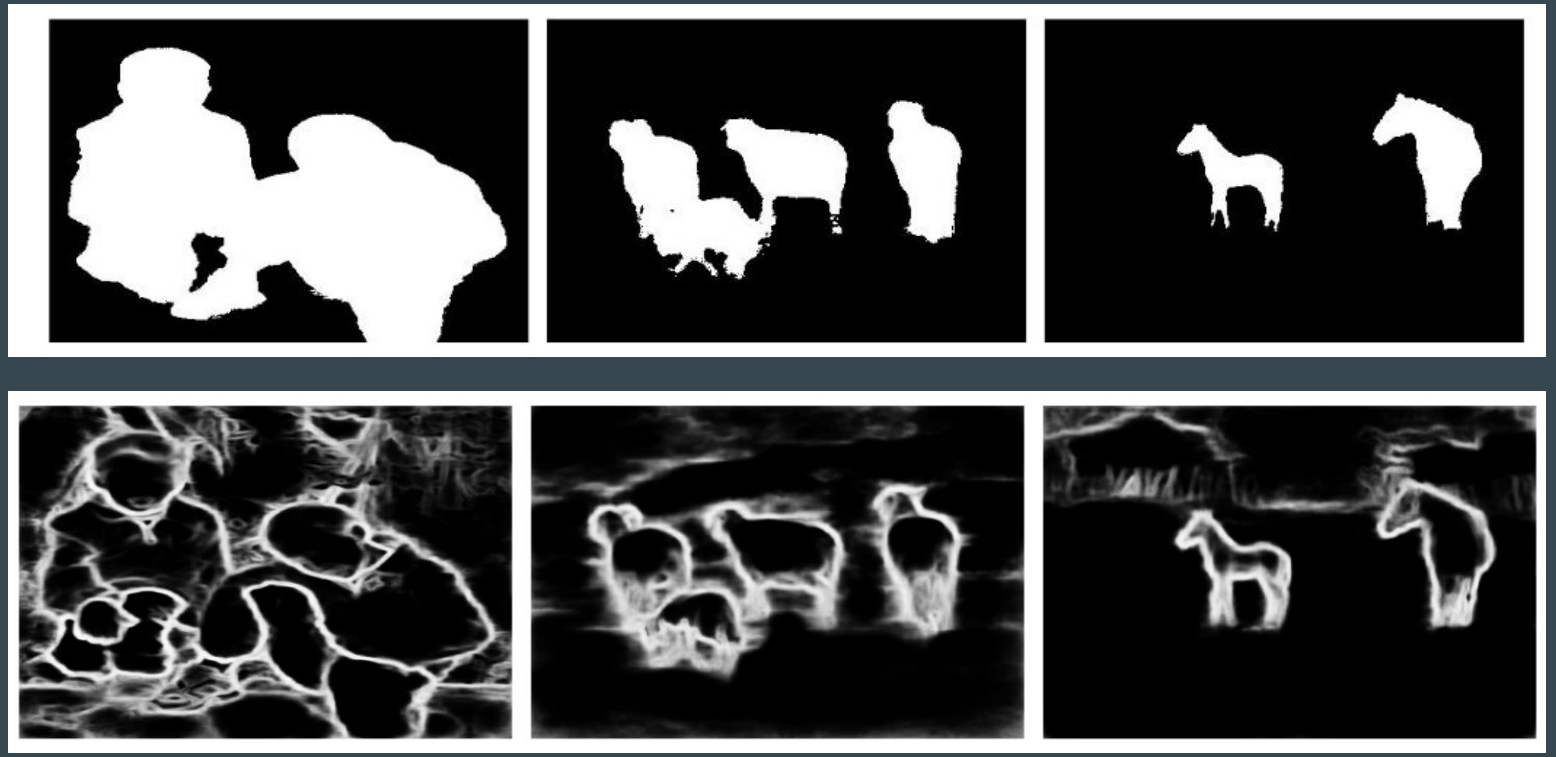
MSRNet Features

- MSRNet :
  - As we can conclude from the properties of the above networks, a network should consider both bottom-up and top-down information propagation and output a label map.
  - We want the saliency map and contour size to have the same resolution as the input image. Thus the output should be a label map of same resolution as the input image.

Attention Module Architecture



MSR-NET Results



Identifying Salient Object Instances

- Given the detected contours of salient object instances, we apply multiscale combinatorial grouping (MCG) to generate a number of salient object proposals.
- Though the generated object proposals are of high quality, they are still noisy and tend to have severe overlap.
- We apply MAP based subset optimisation to optimize the number and locations of the detected windows from the set of object proposals.
- We further filter out noisy or overlapping proposals and produce a compact set of segmented salient object instances. Finally, a fully connected CRF model is employed to improve spatial coherence and contour localization in the initial salient instance segmentation.

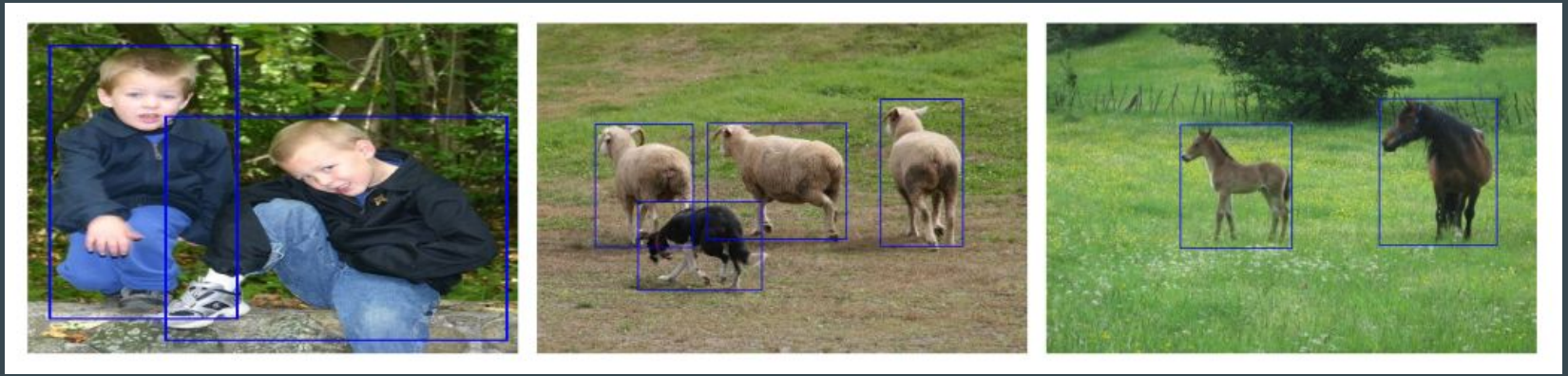
Multiscale Combinatorial Grouping (MCG)

- MCG is a unified approach for bottom-up hierarchical image segmentation and object candidate generation.
- This is done by replacing the contour detector in MCG (gPb) with the MSRNet based salient object contour detector.
- Given an input image, four salient object contour maps (three from scaled versions of the input and one from the fused map) are generated.
- Each of these four contour maps is used to generate a distinct hierarchical image segmentation represented as an ultrametric contour map (UCM).
- These four hierarchies are aligned and combined into a single hierarchical segmentation, and a ranked list of object proposals.

MAP based subset optimization

- Given the set of initially screened salient object proposals, we further apply a MAP based subset optimization method to produce a compact set of object proposals.
- These proposals tend to be highly overlapping and noisy.
- Based on the Maximum a Posteriori principle, a novel subset optimization framework is used to generate a compact set of detection windows out of noisy proposals.
- Each remaining salient object proposal is a detected salient instance.
- We can then use these proposals to generate the final instance segmentation using CRF based optimization.

Results after MCG and Subset Optimization



Conditional Random Field Model(CRF)

- After receiving the proposals from MCG, we consider the final instance segmentation using CRF as a Multi-class labelling problem.
- We consider there are K labels (Main Object Proposals), and the background is considered to be the K+1th label. At the end, every pixel is assigned with one of the K + 1 labels using a CRF model
- We first define a probability map with K + 1 channels, each of which corresponds to the probability of the spatial location being assigned with one of the K + 1 labels.
- After obtaining the initial salient instance probability map, we employ a fully connected CRF model for refinement.

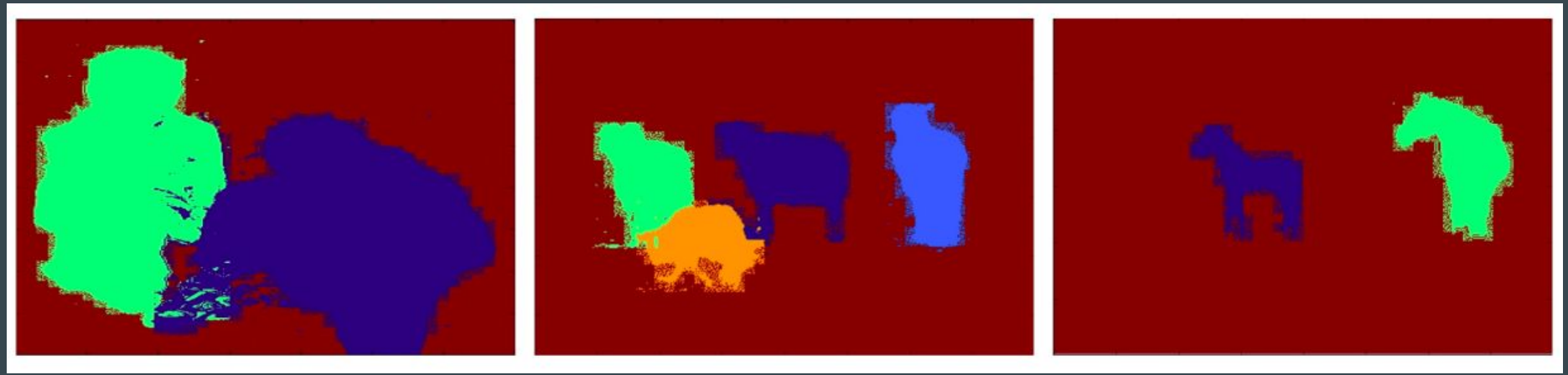
Conditional Random Field Model(CRF)

- The pixel labels are optimized with respect to the following energy function of the CRF:
-

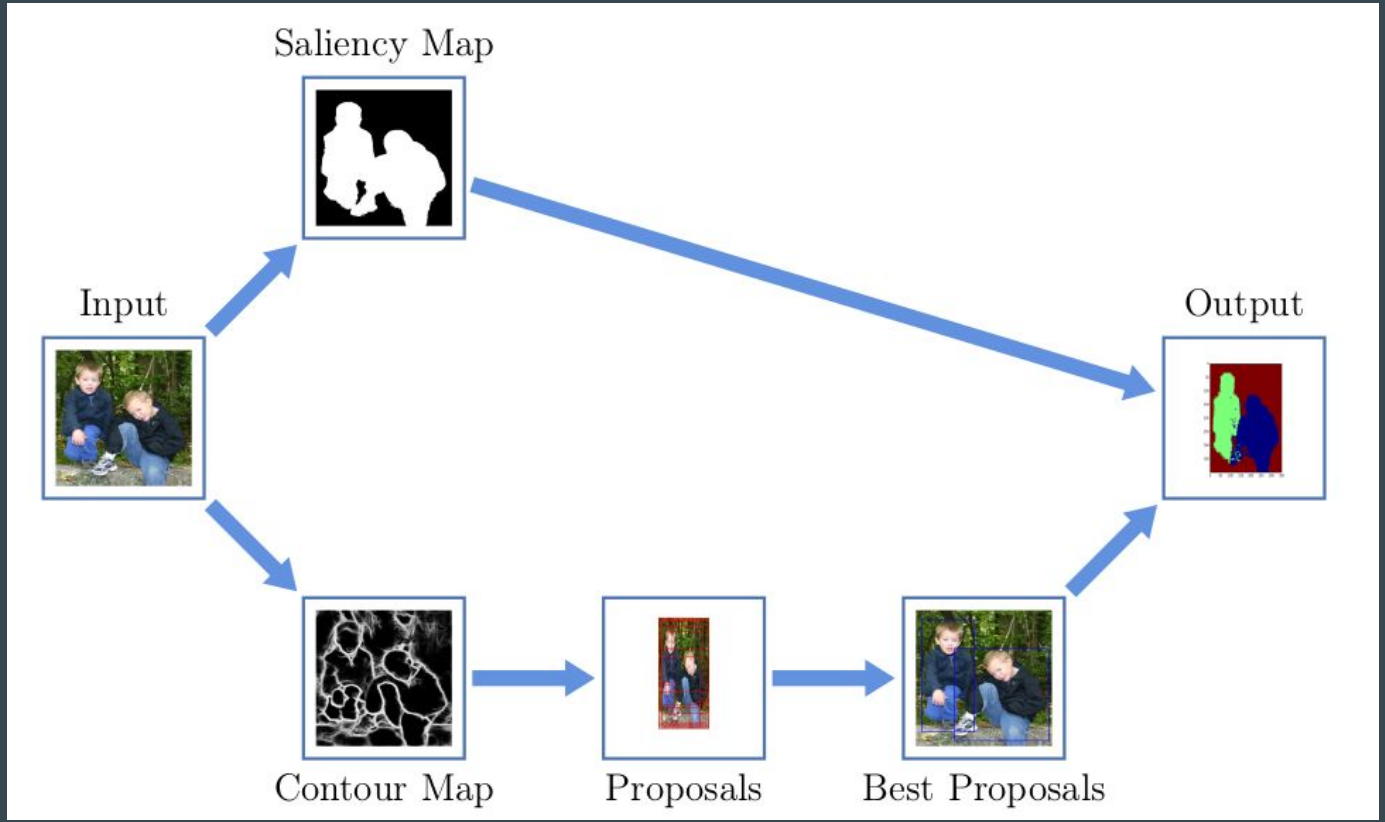
$$E(x) = -\sum_i \log P(x_i) + \sum_{i,j} \theta_{ij}(x_i, x_j)$$

$$\theta_{ij} = \mu(x_i, x_j)\left[\omega_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + \omega_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)\right]$$

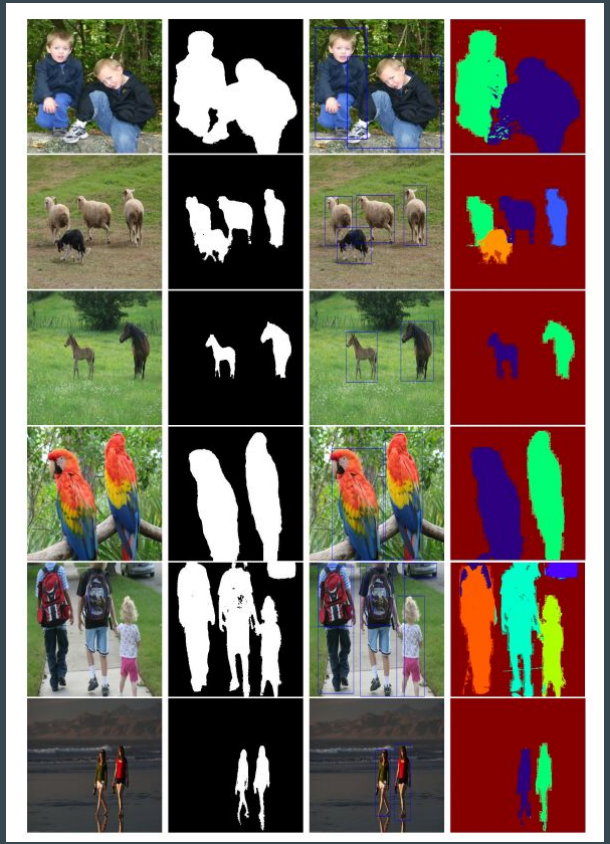
Results after CRF



Result Pipeline



Final Results



Thank You