Chapter 3

Design and Implementation of Salient Object Detection Model using Active Contours and Gradient Vector Flow

This chapter describes three statistical models developed for salient object detection. The models are introduced in Section 3.1. The theoretical foundations of the proposed model and brief introduction of state-of-the-art methods are described in Section 3.2. The description of each proposed model is given in Section 3.3. Section 3.4 gives the result generated by each of the models and their analysis. Section 3.5 concludes the chapter.

3.1 Background

The first model presented in this chapter is a framework based on the automatic segmentation of images to identify salient objects. This work utilizes an automated multilevel multiplane threshold-based algorithm and a region-based algorithm to realize a better productive salient object detection. The suggested scheme contains two components. Initially, a multilevel multiplane threshold-based procedure is utilized to find image portions with an excellent prominent object probability; then, a region-based method was employed to obtain the complete salient object. The proposed framework was experimented on three publicly available datasets, and was resilient to intensity differences, delicate boundaries, besides being extra computationally productive when measured to the region-based procedure in the absence of the threshold-based data.

A second model is proposed for salient object detection using cues provided by edge detection. The segmentation procedure of the Active Contour Model (ACM) is thus improvised to give a result which not only segments the image but also includes the salient object in the segmented outputs. Edge cues are used based on the assumption that salient objects have a higher contrast from the background present in the images. They also help to solve the initialization problem of active contours. The edge cues help to determine image regions with a higher probability of containing a salient object. These regions are then fed to the Active Contour Model to find out the specific salient object present in the image. The method is based on basic human intuition to find salient objects in an image. The evaluation has been done

on three publicly available datasets. Several recent works have been used to compare the proposed model on precision, recall and false-positive rate parameters.

The third model attempts to analyze scene information present in the image by augmenting salient object information with background information. The salient object is initially identified using a method called as Minimum Directional Contrast (MDC). The underlying assumption behind using this method for defining salient objects is that salient pixels have higher minimum directional contrast than non-salient pixels. Finding MDC provides us with a raw salient metric. The Gradient Vector Flow (GVF) model of image segmentation inculcates the raw saliency information. The gradient of MDC is calculated and added to the data term of the energy functional of GVF so that the contour formation utilizes not only edge formation but also saliency information. The result obtained gives us not only the salient object but also added background information. Three public datasets have been used to evaluate the results obtained. The comparative study of the proposed method for salient object detection with other state-of-the-art methods available in the literature is presented in terms of precision, recall, and F1-Score.

3.2 Literature of segmentation and saliency computation methods for proposed models

In this section, the theoretical foundation of the proposed model is described. Section 3.2.1 describes Otsu's thresholding method, which is used for introducing multi-level

multi-plane thresholding. Section 3.2.2 describes the working of the Active Contour Model, which is used together with edge-based cues. Section 3.2.3 and Section 3.2.4 discuss Minimum Directional Contrast and Gradient Vector Flow, respectively, which are used to form the third model for salient object detection.

3.2.1 Otsu's Thresholding Method

This section describes the multi-level Otsu's thresholding [175] method used in the first proposed model.

Let the pixels of a given picture be represented in L gray levels [1, 2, ..., L]. The number of pixels at level i is denoted by n_i and the total number of pixels by $N = n_1 + n_2 + \cdots + n_L$. The gray-level histogram is normalized and regarded as a probability distribution:

$$p_i = \frac{n_i}{N} \qquad p_i \geq 0 \qquad \sum_{i=1}^{L} p_i = 1 \tag{3.1}$$

Now suppose that the pixels are separated into m classes $C_0, C_1, C_2, \ldots, C_m$ by thresholds at level $k_0, k_1, k_2, \ldots, k_{m-1}$; C_0 designates pixels with levels [1, 2, 3, ..., k_0], C_1 denotes pixels with levels $[k_0+1, k_0+2, k_0+3, \ldots, k_1]$, C_2 denotes pixels with levels $[k_1+1, k_1+2, k_1+3, \ldots, k_2]$, C_m denotes pixels with levels $[k_{m-1}+1, k_{m-1}+2, k_{m-1}+3, \ldots, L]$. Then the probabilities of class occurrence and the class mean levels, respectively, are given by

$$\omega_0 = Pr(C_0) = \sum_{i=1}^{k_0} p_i = \omega(k_0) \tag{3.2}$$

$$\omega_1 = Pr(C_1) = \sum_{i=k_0+1}^{k_1} p_i = \omega(k_1) \tag{3.3}$$

$$\omega_2 = Pr(C_2) = \sum_{i=k_1+1}^{k_2} p_i = \omega(k_2) \tag{3.4}$$

$$\begin{aligned}\omega_m = Pr(C_m) &= \sum_{i=k_{m-1}+1}^{L} p_i = \omega(L) \\ &= 1 - [\omega(k_0) + \omega(k_1) + \omega(k_2) + \cdots + \omega(k_{m-1})]\end{aligned} \tag{3.5}$$

and

$$\mu_0 = \sum_{i=1}^{k_0} iPr(i|C_0) = \sum_{i=1}^{k_0} i\frac{p_i}{\omega_0} = \frac{\mu(k_0)}{\omega(k_0)} \tag{3.6}$$

$$\mu_1 = \sum_{i=k_0+1}^{k_1} iPr(i|C_1) = \sum_{i=k_0+1}^{k_1} i\frac{p_i}{\omega_1} = \frac{\mu(k_1)}{\omega(k_1)} \tag{3.7}$$

$$\mu_2 = \sum_{i=k_1+1}^{k_2} iPr(i|C_2) = \sum_{i=k_1+1}^{k_2} i\frac{p_i}{\omega_2} = \frac{\mu(k_2)}{\omega(k_2)} \tag{3.8}$$

$$\begin{aligned}\mu_m = \sum_{i=k_{m-1}+1}^{L} iPr(i|C_m) &= \sum_{i=k_{m-1}+1}^{L} i\frac{p_i}{\omega_m} = \frac{\mu(L)}{\omega(L)} \\ &= \frac{\mu_T - [\mu(k_0) + \mu(k_1) + \mu(k_2) + \cdots + \mu(k_{m-1})]}{1 - [\omega(k_0) + \omega(k_1) + \omega(k_2) + \cdots + \omega(k_{m-1})]}\end{aligned} \tag{3.9}$$

where

$$\omega(k_0) = \sum_{i=1}^{k_0} p_i \tag{3.10}$$

$$\mu(k_0) = \sum_{i=1}^{k_0} i p_i \tag{3.11}$$

are the zeroth and first-order cumulative moments of the histogram up to the level k_0, respectively, and

$$\mu_T = \mu(L) = \sum_{i=1}^{L} i p_i \tag{3.12}$$

is the total mean of the actual image. For any selection of $k_0, k_1, k_2, \ldots, k_{m-1}$ it can be effortlessly confirmed that

$$\omega_0\mu_0 + \omega_1\mu_1 + \omega_2\mu_2 + \cdots + \omega_m\mu_m = \mu_T \tag{3.13}$$

$$\omega_0 + \omega_1 + \omega_2 + \cdots + \omega_m = 1 \tag{3.14}$$

The class variances are given by

$$\sigma_0^2 = \sum_{i=1}^{k_0} (i-\mu_0)^2 Pr(i|C_0) = \sum_{i=1}^{k_0} (i-\mu_0)^2 \frac{p_i}{\omega_0} \tag{3.15}$$

$$\sigma_1^2 = \sum_{i=k_0+1}^{k_1} (i-\mu_1)^2 Pr(i|C_1) = \sum_{i=k_0+1}^{k_1} (i-\mu_1)^2 \frac{p_i}{\omega_1} \tag{3.16}$$

$$\sigma_2^2 = \sum_{i=k_1+1}^{k_2} (i-\mu_2)^2 Pr(i|C_2) = \sum_{i=k_1+1}^{k_2} (i-\mu_2)^2 \frac{p_i}{\omega_2} \tag{3.17}$$

$$\sigma_m^2 = \sum_{i=k_{m-1}+1}^{L} (i-\mu_m)^2 Pr(i|C_m) = \sum_{i=k_{m-1}+1}^{L} (i-\mu_m)^2 \frac{p_i}{\omega_m} \tag{3.18}$$

These need second-order cumulative moments (statistics). For assessing the "goodness"of the threshold (at level $k_0, k_1, k_2, \ldots, k_m$), the following discriminant criterion measures (or measures of class separability) are placed which are used in the discriminant analysis:

$$\lambda = \frac{\sigma_B^2}{\sigma_W^2} \qquad \kappa = \frac{\sigma_T^2}{\sigma_W^2} \qquad \alpha = \frac{\sigma_B^2}{\sigma_T^2} \tag{3.19}$$

where

$$\sigma_T^2 = \sum_{i=1}^{L} (i - \mu_T)^2 p_i \tag{3.20}$$

$$\sigma_W^2 = \omega_0 \sigma_0^2 + \omega_1 \sigma_1^2 + \omega_2 \sigma_2^2 + \cdots + \omega_m \sigma_m^2 \tag{3.21}$$

$$\sigma_B^2 = \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2 + \omega_2 (\mu_2 - \mu_T)^2 + \cdots + \omega_m (\mu_m - \mu_T)^2 \tag{3.22}$$

are the within-class variance, the between-class variance and the total variance of levels, respectively. Then the problem is lessened to an optimization problem to find threshold $k_0, k_1, k_2, \ldots, k_{m-1}$ which will maximize one of the object functions.
This point of view is driven by a intuition that well-thresholded classes are different in gray levels, and contrarily, a threshold presenting the best distinction of classes in gray levels is the best threshold. The discriminant basis maximizing λ, κ and α, respectively, for $k_0, k_1, k_2, \ldots, k_{m-1}$ are, but, equal to each other; e.g. $\kappa = \lambda + 1$ and $\alpha = \frac{\lambda}{\lambda+1}$ in terms of λ, since the following basic relation always holds:

$$\sigma_W^2 + \sigma_B^2 = \sigma_T^2 \tag{3.23}$$

It is noticed that σ_W^2 and σ_B^2 are functions of threshold levels $k_0, k_1, k_2, \ldots, k_{m-1}$ but σ_T^2 is independent of $k_0, k_1, k_2, \ldots, k_{m-1}$. It is also observed that σ_W^2 is build on the second-order statistics (class variances), whereas σ_B^2 is build on the first-order statistics (class means). Therefore, α is the easiest measure with respect to $k_0, k_1, k_2, \ldots, k_{m-1}$. Therefore, α is taken as the basis measure for assessing the "goodness"(or distinctness) of the threshold at level $k_0, k_1, k_2, \ldots, k_{m-1}$ [175].

3.2.2 Active Contour Model

ACM [176] procedures show a contour utilizing a zero level of a function named as the level set function (ψ), and the image was partitioned by developing the level-set expression. The evolution equation was acquired by minimizing an energy function of the Signed Pressure Function (SPF):

$$\frac{\partial \psi}{\partial t} = spf(I(x,y)).\beta|\nabla\psi| \qquad x, y \in \Delta \tag{3.24}$$

where β is the balloon force, and Δ is a bounded subset of $I(x, y \in \Re^2)$ in the given image. The initial contour size decided the bounded subset size. The signed pressure function (spf) has values in the range [-1, 1] and is described as

$$spf(I(x,y)) = \frac{I(x,y) - \frac{c_1+c_2}{2}}{max(|I(x,y) - \frac{c_1+c_2}{2}|)} \qquad x, y \in \Delta \tag{3.25}$$

where c_1 and c_2 are two constants, which represent average intensities interior and exterior regions of the contour.

The SPF regulates the signs of the pressure forces in the interior and the exterior region of interest such that the contour contracts if it is larger than the object or increases if it is smaller than the object. A Gaussian filter regulates the level set to stabilize it and manage its susceptibility to noise. The balloon force (β) and standard deviation (σ) of the Gaussian filter are the main variables that depend very much on the image. The balloon force parameter restraints the contraction/ expansion of the development of the contour. A high β can be the reason for the contour to go past faint edges, while a low β can be the reason for the contour not to go by small fragments of the object. Putting σ excessively small can increase contour susceptibility to noise, whereas a high σ can blur the images severely for edge leakage to happen. The algorithm has both local as well as global segmentation characteristics. But, just the local characteristic is taken into consideration because the global characteristic can be computationally costly and susceptible to noisy regions.

3.2.3 Saliency Computation

Global contrast cues have been used to differentiate the foreground object from the background. Huang and Zhang [177] brought minimum directional contrast to generate the raw saliency map of an image. Concerning a center pixel j, the whole image can be divided into four parts based on their locations. The directional

contrast from each part Γ is given as

$$DC_{j,\Gamma} = \sqrt{\sum_{k \in \Gamma} \sum_{ch=1}^{M} \left(I_{j,ch} - I_{k,ch}\right)^2} \tag{3.26}$$

where, I is the input image with M color channels in RGB color space. Γ represents each of the regions concerning j, i.e., Top-Left (TL), Top-Right (TR), Bottom-Left (BL) and Bottom-Right (BR). Minimum directional contrast can be calculated as:

$$\begin{aligned} T(j) &= min\left(DC_{j,\Gamma}\right) \\ &= \sqrt{min\left(\sum_{k \in \Gamma} \sum_{ch=1}^{M} \left(I_{j,ch} - I_{k,ch}\right)^2\right)} \quad \Gamma \in TL, TR, BL, BR \end{aligned} \tag{3.27}$$

From the equation, it seems that a lot of mathematical computation time is required. In their work, Huang and Zhang [177] explain how the computation can be done in O(1) time using the concept of the integral image. They mention that for an image of Quarter Video Graphics Array (QVGA) resolution, it takes only 1.5 ms to compute the minimum directional contrast.

3.2.4 Gradient Vector Flow

In [176], Kass *et al.* propose active contour model for image segmentation. There are two categories of active contour models: parametric and geometric. This model focuses on the parametric active contour model. Edges attract the contours. There are two sets of forces. The external force pulls the curve towards contour, and the

internal force keeps the contour together. Traditional snakes model, however, face limitations due to two reasons:

1. Wrong initialization of the contours results in an improper segmentation.

2. The contour has difficulty moving into concave boundaries.

The multiple resolutions approach solves both problems but cannot simplify the procedure of the contour moving in different resolutions. In [178], Xu and Prince propose Gradient Vector Flow (GVF) fields, which are derived from the image by "minimizing an energy functional in a variational framework." Minimization is done using a decoupled linear partial differential equation. Snakes using these fields for segmentation are called as GVF snakes.

The snake is defined as a curve $z(u) = |x(u), y(u)|, u \in [0, 1]$ that passes through the image for minimizing the energy functional

$$E = \int_0^1 \frac{1}{2} \left(\rho \left| z'(u) \right|^2 + \zeta \left| z''(u) \right|^2 \right) + E_{ext}(z(u))\, du \tag{3.28}$$

ρ and ζ control the tension and rigidity of the contour. The term E_{ext} is responsible for pulling the contour to the edges.

Xu and Prince [178] begin by defining an edge map $g(x, y)$ as:

$$g(x, y) = -E_{ext}(x, y) \tag{3.29}$$

There are two common alternatives to choose E_{ext}. It can be the gradient of an image or gradient of a Gaussian-filtered image. A Gaussian filter is applied to counter-effect the noise present in the image.

Then they define the vector field $\mathbf{W}(x, y) = (v(x, y), w(x, y))$ which minimizes the energy functional

$$E = \int\int \xi \left(v_x^2 + v_y^2 + w_x^2 + w_y^2\right) + |\nabla g|^2 |\mathbf{W} - \nabla g|^2 \, dxdy \tag{3.30}$$

The first term is the smoothness term, and the second term is the data term. By using the calculus of variations, GVF can be found by solving Euler equations.

Following are some state-of-the-art methods against which the proposed models have been compared.

- BMS [179]: The algorithm uses a topological structure of Boolean maps to find the salient object.
- DRFI [180]: In this work, the authors calculate saliency scores at different levels and then finally integrate the different saliency maps. Regional and backgroundness descriptors are used to form saliency maps at different levels.
- DSR [181]: The authors first generate background templates from image boundaries extracted via superpixels. Dense and sparse reconstruction errors are then calculated, which are then propagated through different image regions based on context. Saliency measures obtained from each of these reconstruction errors are integrated using the Bayesian technique.

- GC [182]: The algorithm segments an image into perceptually similar elements and finds the salient object using similarity and spatial representation of pixels.

- GS [183]: This algorithm uses background and connectivity priors to subtract the background from the image and thus locating the salient object.

- MR [164]: The salient regions are obtained by using graph-based manifold ranking. The superpixels act as nodes. Every pixel is ranked based on cues obtained from the background and foreground.

- PCAS [184]: Margolin *et al.* studied the inner statistics of patches to find unique patterns. They obtain a principal component analysis of each patch and study pattern and color of each patch.

- RC [166]: This utilizes local contrast and coherence scores to find the salient object.

- SMD [185]: The authors use the structured matrix decomposition model to capture image structure and use Laplacian regularization to enhance distance between non-salient and salient objects.

- UFO [186]: In this model salient object is achieved by using contrast for uniqueness, finding focused areas in an image, and ensuring completeness of the object.

3.3 Methods and Models

This section describes the methodology of each of the proposed models in detail. The main contributions are: the development of edge maps, fusing saliency and segmentation and developing an idea of scene analysis. Section 3.3.1 explains the first model proposed using thresholding and region-based segmentation. The second model is given in Section 3.3.2. Section 3.3.3 describes the model proposed using Gradient Vector Flow and Minimum Directional Contrast.

3.3.1 Salient Object Detection using Threshold and Region-Based Segmentation

This section proposes an image segmentation system that focuses on identifying salient objects from images. Besides, the proposed framework aims to resolve the flaws of both the threshold-based and region-based algorithms by applying a two-step method. The initial step is to recognize salient object spots with the help of a threshold-based algorithm. The next stage brings out the location and complete range of the possible salient object by the use of a region-based algorithm. Description of the suggested framework (threshold-based and region-based parts) are illustrated in FIGURE 3.1. The threshold-based element uses multilevel plane-by-plane thresholding to detect the location of possible salient objects. The threshold component remarkably lessens the computation time, whereas the region-based component refines the last segmentation output.

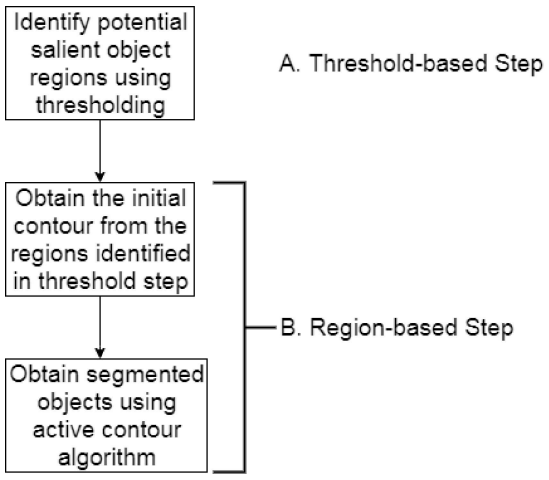

FIGURE 3.1: Description of the proposed framework for threshold-based model.

3.3.1.1 Threshold-Based Component

Explanation of the suggested threshold-based stage, mentioned in part A of FIGURE 3.1, is as follows:

Segmentation: By inputting the images, a salient object detection step is applied for the segmentation of salient pixels from background objects in the pictures. The suggested threshold-based stage uses a histogram-based algorithm, recognized as Otsu's method, that instinctively calculates the thresholds and segments salient objects from the background objects. The discriminant basis chooses the optimal group of thresholds, that is by maximizing the discriminant measure.

Images, however, can have several salient objects. Because of the presence of multiple objects and their high-intensity variations within the image, it is hard to select a single best threshold point for the Otsu's method. To make the proposed framework robust to these challenges, the multilevel plane-by-plane threshold-based algorithm

was proposed.

The proposed algorithm segments the image while extracting potential salient objects. Multiple thresholds were obtained for each plane separately. Seven levels were used after observation. The multiplane multilevel algorithm showed better detection of salient locations in contrast to the standard multilevel Otsu threshold algorithm.

3.3.1.2 Region-Based Component

Threshold-based algorithms cannot detect the complete range of salient objects because they can be over segmented and split because of the flaws of the threshold-based algorithms. Region-based algorithms control the limitations of threshold-based techniques by extracting the entire area of salient objects. The suggested system uses a region-based technique given in [187]. An ACM and a level-set algorithm based on a novel region-based signed pressure force (SPF) function is applied. It uses the statistical details of the regions interior and exterior to the contour to restrain the evolution of the contour. This characteristic lets the procedure efficiently halt the contours at faint or indistinct edges and therefore, can automatically find the interior and exterior borders of likely salient objects. The region-based part, mentioned in part B of FIGURE 3.1, follows three stages.

1. Initial Contour: The region-based stage utilizes an ACM and a level-set algorithm that uses an initial contour to separate the area of interest. Initialization of the level set is determined as a constant ρ, which is zero at the boundary,

FIGURE 3.2: Step-wise Result (a) Input image (b) Result of Threshold-based segmentation (c) Result of Region-based segmentation (d) Ground Truth.

negative inside, and positive outside the contour. The proposed framework used the salient object detected from the threshold-based stage as the first contour ρ.

2. Segmentation: ACM is used for the segmentation of images using contours.

3. Final Image: Here, modified salient object discrimination is used. For the final image, only the top 10 regions from each plane having a maximum pixel-wise area are labeled as potential salient objects. This parameter was chosen as a trade-off between missing out salient objects and gathering non-salient objects. The region-based component is utilized for the detection of the full extent of the salient objects.

FIGURE 3.2 shows a step-wise result obtained by the proposed method.

3.3.2 An Improved Active Contour Model for Salient Object Detection using Edge Cues

In this section, the second proposed model is described. It uses active contours and edge cues to locate the salient object.

For solving the problem of salient object detection, the proposed model uses active contour with the level-set method. This algorithm is used because of 3 main reasons:

1. It helps to obtain sharp boundaries.
2. Intensity heterogeneity is overcome.
3. By simple modifications to the algorithm, multiple salient objects in the image can be detected.

The result of an active contour model relies on the choice of initial contour. If the initial contour and probable salient region intersect, the result is expected to be better than when they (initial contour and likely salient area) do not intersect. To make the initial contour contain the much possible area of a probable salient region, a primary edge operator - Sobel - is used. Using Sobel, preliminary image segmentation is performed. The result is then used as the initial contour. This initial contour is then trimmed/expanded to carve out the complete salient object. The procedure is shown in FIGURE 3.3

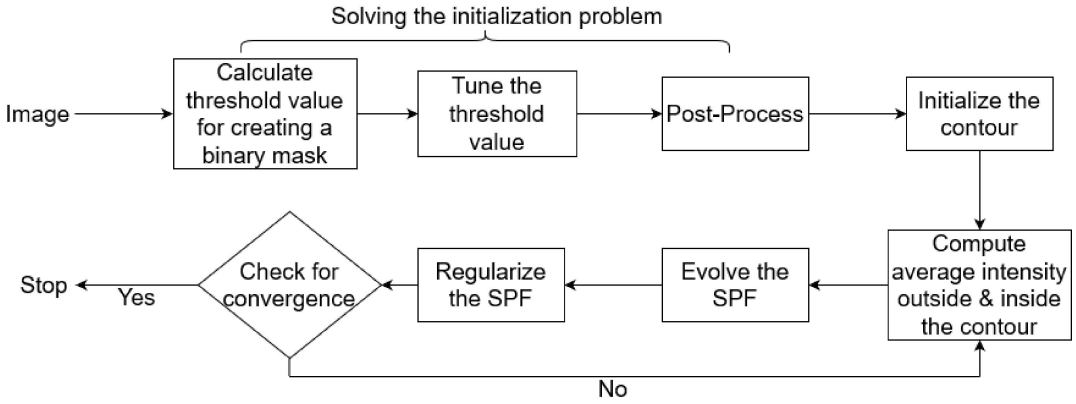

FIGURE 3.3: Flowchart for the proposed algorithm using active contours and edge cues.

The preliminary step is to find a threshold to build the binary mask. For this, the Sobel operator is used. For generating the binary mask containing the probable salient object segmentation is performed after finetuning the threshold obtained. After the segmentation, the binary mask is processed by dilating, filling gaps, and smoothening. The post-processed binary mask is used as an initial contour for the active contour algorithm. The average intensity interior and exterior to the contour are calculated, which determines the evolution of the signed pressure function (spf). A Gaussian filter is used to regularize the spf function. For checking the convergence, the number of iteration is set experimentally. The procedure is described visually in FIGURE 3.4 taking sample images from three publicly available datasets.

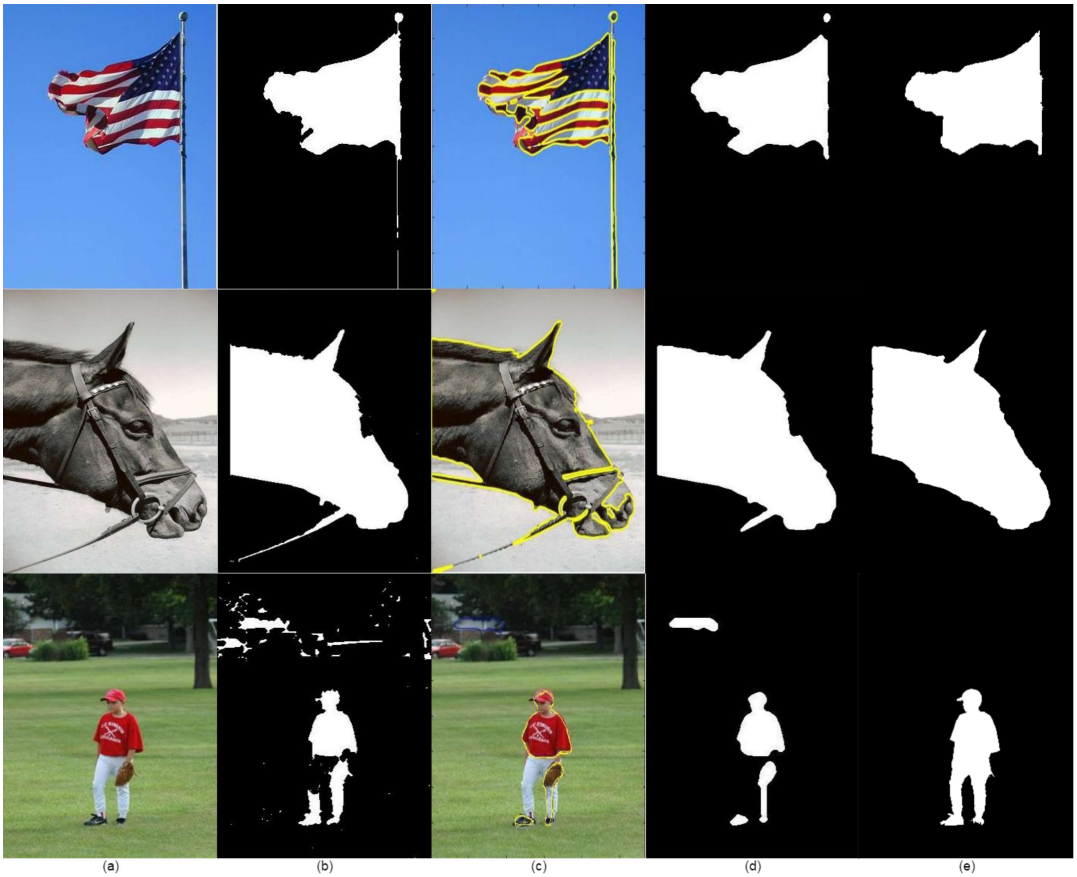

FIGURE 3.4: Visual steps for the proposed algorithm. (a) Original image (b) Segmented image (c) Final contour marked in yellow (d) Salient object (e) Ground truth.

3.3.2.1 Generating Initial Contour using Sobel Operator

The primary goal of this step is to generate a binary mask that holds the salient object and will serve as the initial contour. There is a clear contrast difference between the salient object present in the foreground and the background of the image. To exploit this contrast cues gradient of the image is calculated. After the gradient is calculated, a threshold value is used to generate the binary mask, as shown in the second and third images of the first row in FIGURE 3.5. The binary

mask is enhanced using dilation, hole-filling, and smoothening. The results of the post-processing of binary mask are shown in first and second images of the second row in FIGURE 3.5. The final gradient mask, which serves as the initial contour, is shown in the second image of the second row in FIGURE 3.5. An active contour is initiated with this mask. The result of the active contour step showing the final salient object is depicted in the last image of the second row in FIGURE 3.5.

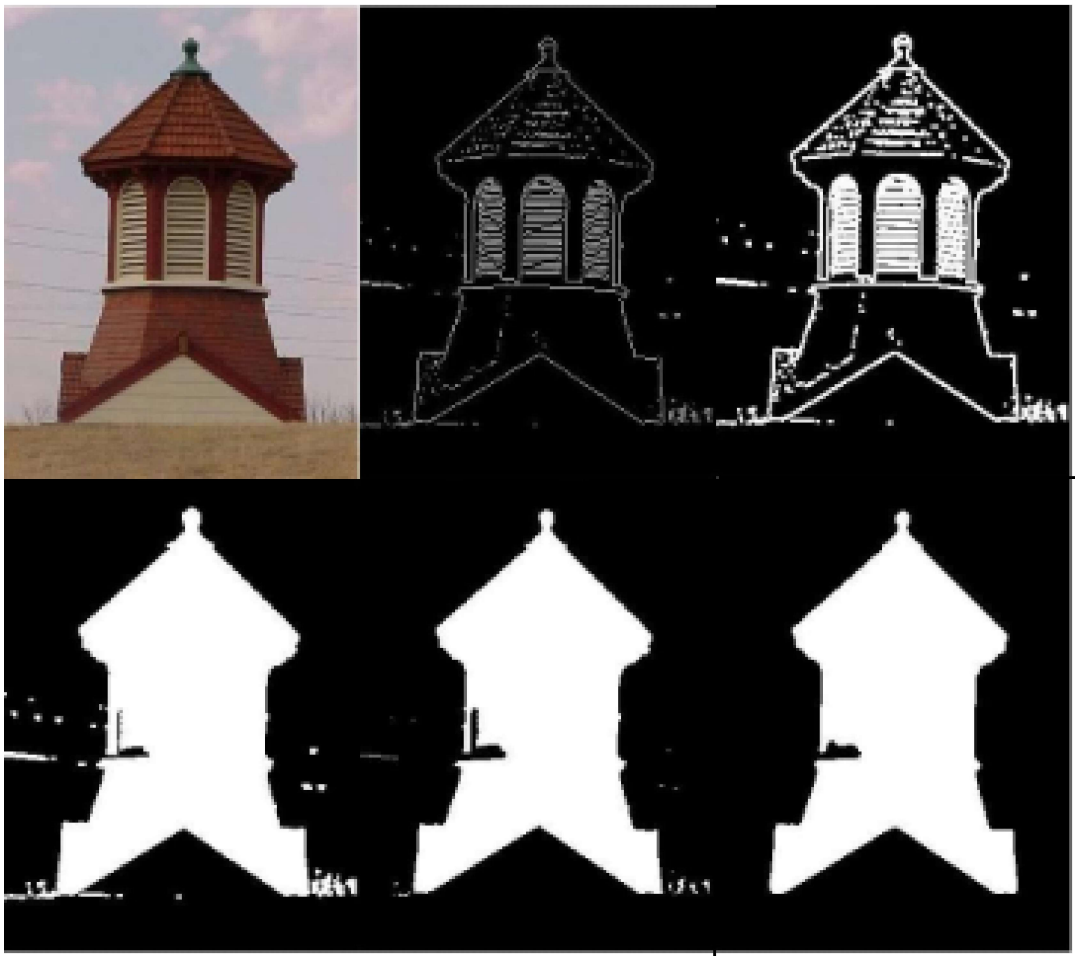

FIGURE 3.5: First row left to right: Original Image, Gradient Mask, Dilated Gradient Mask. Second Row left to right: Dilated Gradient Mask after Filling Holes, Smoothened Gradient mask, Salient Object.

3.3.2.2 Pseudocode

The pseudocode for the proposed method is given in Algorithm 1.

Algorithm 1: Proposed algorithm

Result: ϕ : contour of the salient object

I is initialized with the input image;
The threshold is obtained by applying the Sobel operator to the image;
The threshold is tuned;
The tuned threshold is used with the Sobel operator to obtain the binary gradient mask;
Post-processing of mask is done by dilation, filling gaps and erosion;
Contour is initialized with this post-processed image;
while *convergence* **do**
    c_1=Average intensity outside the contour;
    c_2=Average intensityinside the contour;
    Calculate signed pressure function as given in Eq. 3.25;
    Evolve the contour as given in Eq. 3.24;
    Regularize the contour by convolving it with Gaussian Function;
end

3.3.3 An efficient modification of Gradient Vector Flow using Directional Contrast for Salient Object Detection and Intelligent Scene Analysis

In this section, the goal is to utilize saliency in image segmentation with the help of Gradient Vector Flow. The following sub-sections explain each of the steps in detail.

3.3.3.1 Edge Map Formation

The edge map is constructed to preserve the edge features of an image. But not all edges are relevant to the purpose of salient object detection, and hence, some edge information must be removed. For calculating the edge map, the first step is to calculate the gradient of the image.

$$F = avg\left|\nabla I_{ch}\right| \quad ch = R, G, B \tag{3.31}$$

where, ch represents the number of image channels for an RGB image. I is the input image. After the calculation of the gradient, the edge map is generated as follows:

$$g\left(i,j\right) = \begin{cases} F\left(i,j\right) & \text{if } F\left(i,j\right) > mean\left(F\right) \\ 0 & \text{otherwise} \end{cases} \tag{3.32}$$

3.3.3.2 Computation of the gradient of MDC for Saliency

Section 3.2.3 describes the calculation of the minimum directional contrast. The gradient of MDC gives us information about how the contrast changes in the image for salient objects.

$$\left(m_x, m_y\right) = \nabla U \tag{3.33}$$

where, m_x and m_y are the gradient of MDC obtained in the x and y-directions.

3.3.3.3 Proposed Energy Functional for Salient Object Detection

Based on the proposed edge map and saliency information, the new energy functional can be introduced as

$$E = \int\int \xi \left(v_x^2 + v_y^2 + w_x^2 + w_y^2\right) + |\nabla g|^2 \left(\theta \left|\nabla T\right|^2 + |\mathbf{W} - \nabla g|^2\right) dxdy \tag{3.34}$$

where the value of θ is set as 8% of the average value of minimum directional contrast. v and w can be updated as a function of time using the following equations:

$$v(x,y) = \xi \nabla^2 v(x,y) - (v(x,y) - g_x(x,y) + \theta |\nabla T|^2)|\nabla g|^2 \tag{3.35}$$

$$w(x,y) = \xi \nabla^2 w(x,y) - (w(x,y) - g_y(x,y) + \theta |\nabla T|^2)|\nabla g|^2 \tag{3.36}$$

The above equations are obtained by modifying the solutions provided by Xu and Prince in [178] to accommodate the directional contrast factor.

$$\nabla g = (g_x, g_y) \tag{3.37}$$

$$\theta = 0.08 \times mean\,(T) \tag{3.38}$$

3.3.3.4 The algorithm of the proposed model

This section gives the algorithmic steps for the implementation of the proposed model. Algorithm 2 gives the details of the procedure.

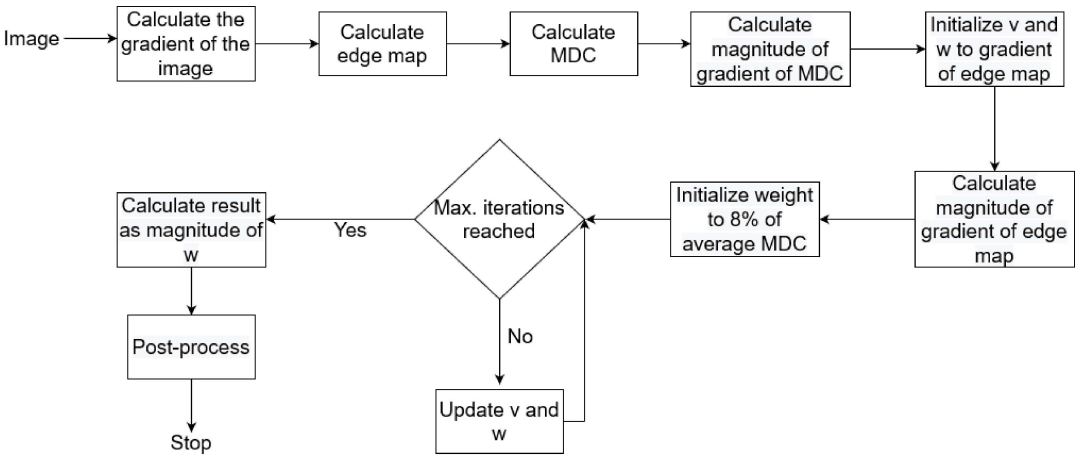

FIGURE 3.6: The diagram of the proposed GVF-based algorithm.

Algorithm 2: Algorithm for the proposed method

Result: Salient Object

Read an image I;
Calculate the gradient of the image F using Eq. 3.31;
Calculate the edge map g of F as shown in Eq. 3.32;
Calculate Minimum Directional Contrast (MDC) T using Eq. 3.27;
Calculate the magnitude of the gradient of MDC $|\nabla T|$ from Eq. 3.33;
Initialize v and w to the gradient of the edge map;
Calculate the magnitude of the gradient $|\nabla g|$ of the edge map g;
Initialize θ to 8% of the average of T;
while *i != iteration* **do**
    Update v and w as shown in Eq. 3.35 and 3.36 repectively;
    Update i = i+1;
end
Calculate result as $\mathbf{W} = \sqrt{v^2 + w^2}$;
Post process the result using dilation, hole filling and erosion;

The algorithm can be represented as a flowchart, as shown in the FIGURE 3.6. The progress of the algorithm is also depicted by FIGURE 3.7, which shows the result obtained at each step.

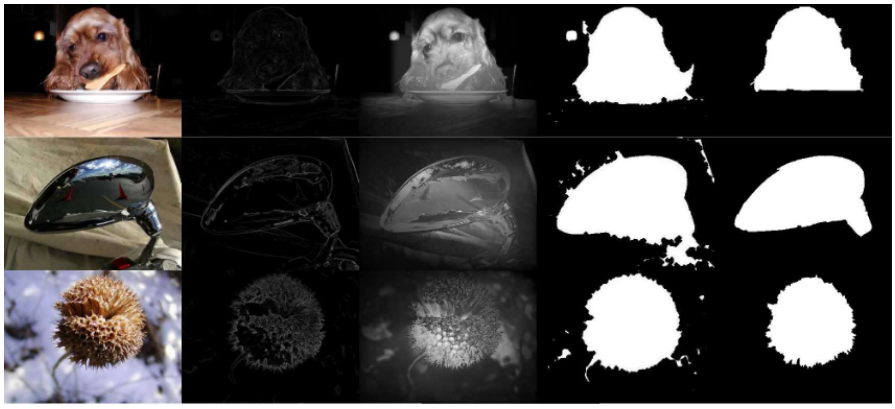

FIGURE 3.7: Result obtained at each step of the proposed method. From left to right: original images, edge map, saliency map, final output, ground truth.

3.4 Results

The result of the proposed models is discussed in this section. Section 3.4.1 presents results obtained by the threshold-based model. Section 3.4.2 examines results obtained by the edge-based model, and section 3.4.3 analyses the result obtained by using GVF based model.

3.4.1 Result analysis of threshold based model

Images from MSRA10K, PASCAL-S, and DUT-OMRON are used for the assessment of the functioning of the proposed framework and the comparison method. FIGURE 3.8 depicts the visual comparisons of the proposed approach to existing methods. The results of existing methods have been obtained from [40].

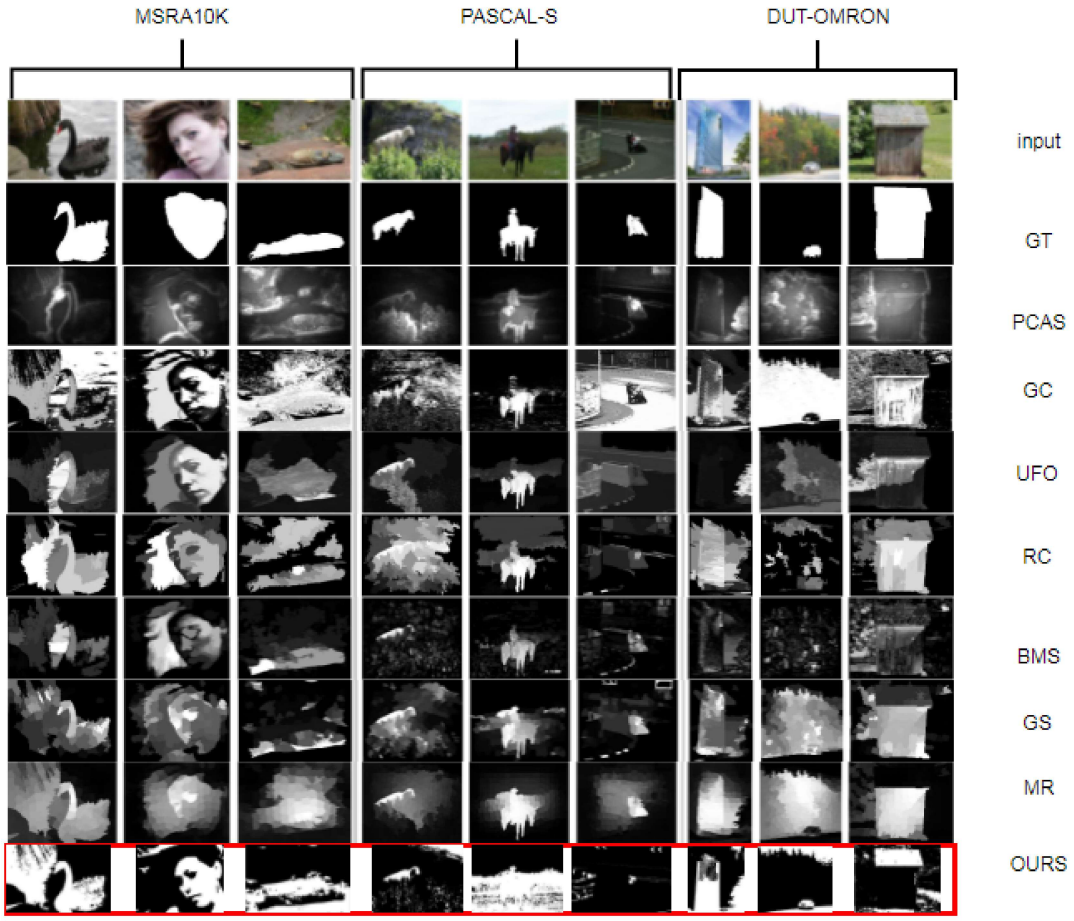

FIGURE 3.8: Visual comparisons of other methods to the proposed method and ground truth (GT).

The images are used for the assessment of the following algorithms: PCAS [184], GC [182], UFO [186], RC [166], BMS [179], GS [183] and MR [164].

The region-based component has two crucial parameters $\mu = 25$ and $\sigma = 1$ which highly rely on the image. The ACM algorithm converges after 160 iterations. All the parameter values were chosen after experimental evaluation. For comparison seven salient object detection algorithms proposed by different researchers are considered.

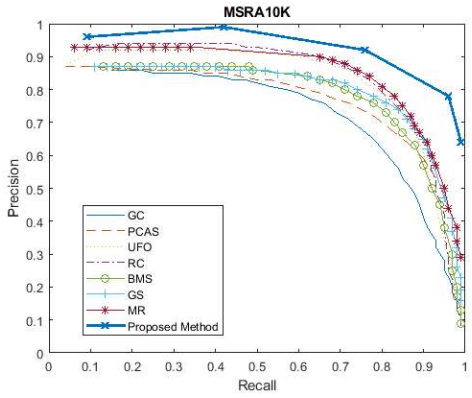

FIGURE 3.9: MSRA10K Precision Recall Curve.

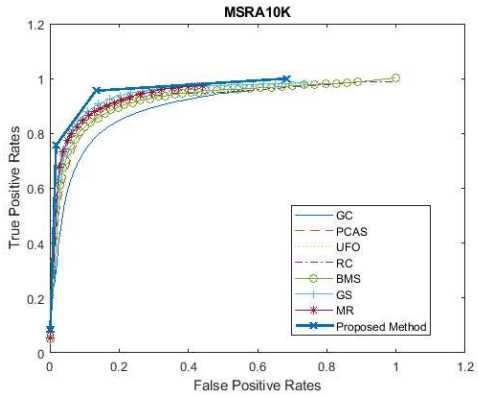

FIGURE 3.10: MSRA10K ROC Curve.

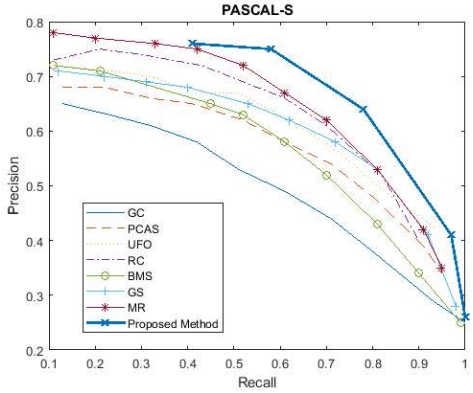

FIGURE 3.11: PASCAL-S Precision Recall Curve.

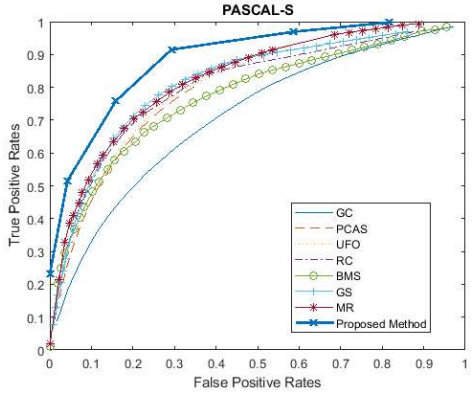

FIGURE 3.12: PASCAL-S ROC Curve.

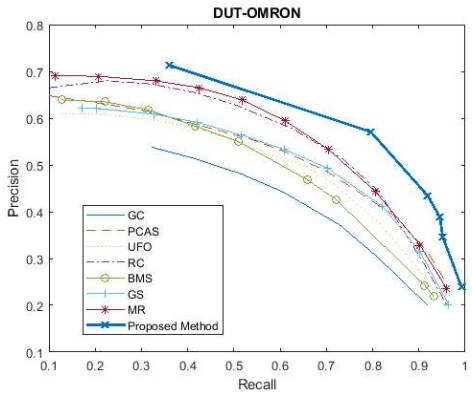

FIGURE 3.13: DUT-OMRON Precision Recall Curve.

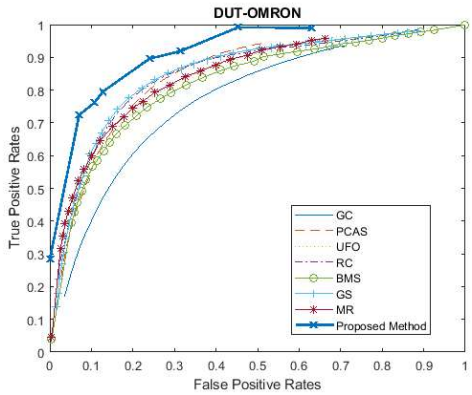

FIGURE 3.14: DUT-OMRON ROC Curve.

TABLE 3.1: Comparison of the proposed model with state-of-the-art algorithms on the basis of F-Score

Methods \ Datasets	MSRA10K	PASCAL-S	DUT-OMRON
PCAS [184]	0.58	0.49	0.45
GC [182]	0.51	0.44	0.44
UFO [186]	0.54	0.5	0.44
RC [166]	0.51	0.51	0.46
BMS [179]	0.54	0.47	0.42
GS [183]	0.56	0.5	0.45
MR [164]	0.58	0.51	0.47
Proposed Method	**0.64**	**0.58**	**0.53**

The algorithms were evaluated for PR curve, ROC curve and F-Measure. The PR curve assessment is shown in FIGURE 3.9, FIGURE 3.11 and FIGURE 3.13. The ROC curve result is shown in FIGURE 3.10, FIGURE 3.12 and FIGURE 3.14. The bold values in the TABLE 3.1 represent the best average performance. The proposed framework shows better results when compared with other algorithms. The proposed method performed better in detecting salient objects from the three datasets.

3.4.2 Result analysis of edge based model

This section deals with the results obtained from the proposed model. The model has been tested on 64-bit MATLAB R2017a (9.2.0.556344) on a desktop with Processor configuration Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40Ghz.

Various edge detection operators like Canny, Laplacian of Gaussian, Prewitt, Roberts, Sobel, and Zero-Cross were checked for suitability. The results that satisfied the purpose were obtained by Sobel edge detection. The threshold obtained from the first step is tuned to generate a binary mask. The direct threshold value does not give

a suitable binary mask; thus, it is tuned by some multiplicative factor. This factor is called the Fudge Factor and is experimentally evaluated to be 0.7. Another parameter is the standard deviation for Gaussian regularization. The parameter value was decided on the base of the accuracy obtained. The value obtained was 32. The number of iteration is set to be 100. The value of the balloon force is set as 25. Both the values are set after checking the accuracy obtained. When the parameters were set, the model was compared to seven state-of-the-art algorithms. The results are shown in the graphs. As it is clear from the graphs the proposed model has more area under the curve, hence it performs better than other algorithms. Data for other algorithms have been taken from [40]. For a visual comparison of the result, a collection of sample images from all databases for all the compared algorithms is given. From the visual result, it is seen that the proposed algorithm gives better output than the other algorithms. No post-processing is performed on the final salient object obtained from the active contour algorithm. FIGURE 3.15, FIGURE 3.16, FIGURE 3.17 and FIGURE 3.18 depict all the results obtained using the proposed model. FIGURE 3.19 shows a visual comparison of the proposed model to state-of-the-art methods.

3.4.3 Result analysis of GVF based model

This section explains the results obtained and compares it with the results obtained from other algorithms.

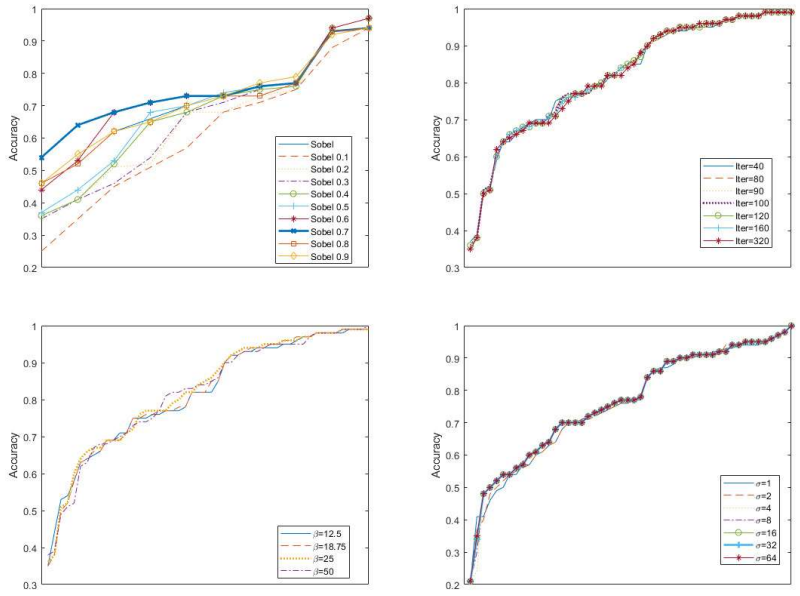

FIGURE 3.15: **Top Left**: Sobel accuracy for different fudge factor values. **Top Right**: Accuracy for different values of iteration **Bottom Left**: Accuracy for differnt values of balloon force **Bottom Right**: Accuracy for different values of standard deviation.

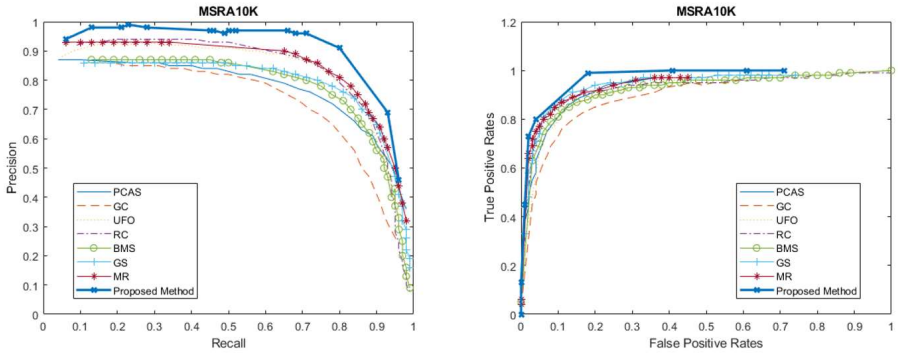

FIGURE 3.16: MSRA 10 K Precision Recall Curve and ROC Curve.

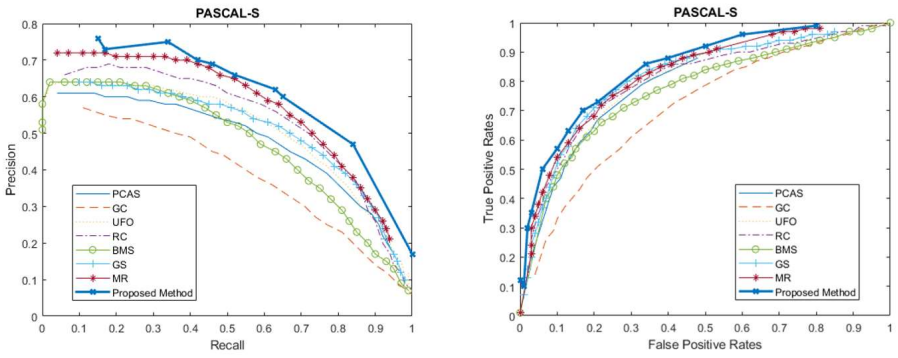

FIGURE 3.17: PASCAL-S precision Recall Curve and ROC Curve

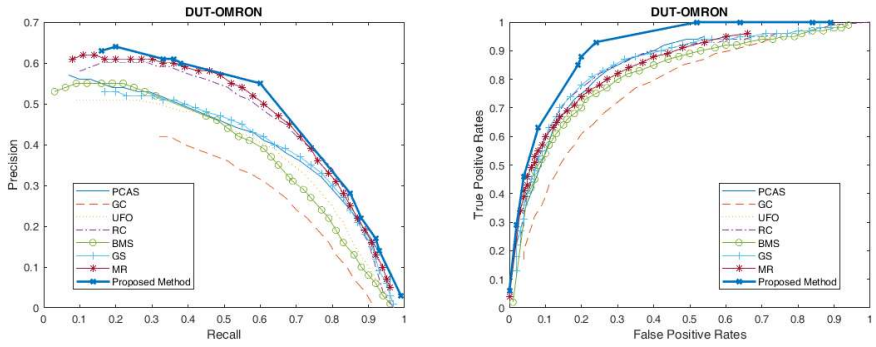

FIGURE 3.18: DUT-OMRON Precision Recall Curve and ROC Curve.

3.4.4 Setting Parameter Values

The value of ξ is set as 0.0007, and the number of iterations is set to 10. The value 0.08 is decided for θ experimentally keeping in mind the trade-off between not to lose necessary information and not to include unnecessary information. The selection of the parameters was made experimentally. The values were set after testing the algorithm on 1000 sample images from three datasets.

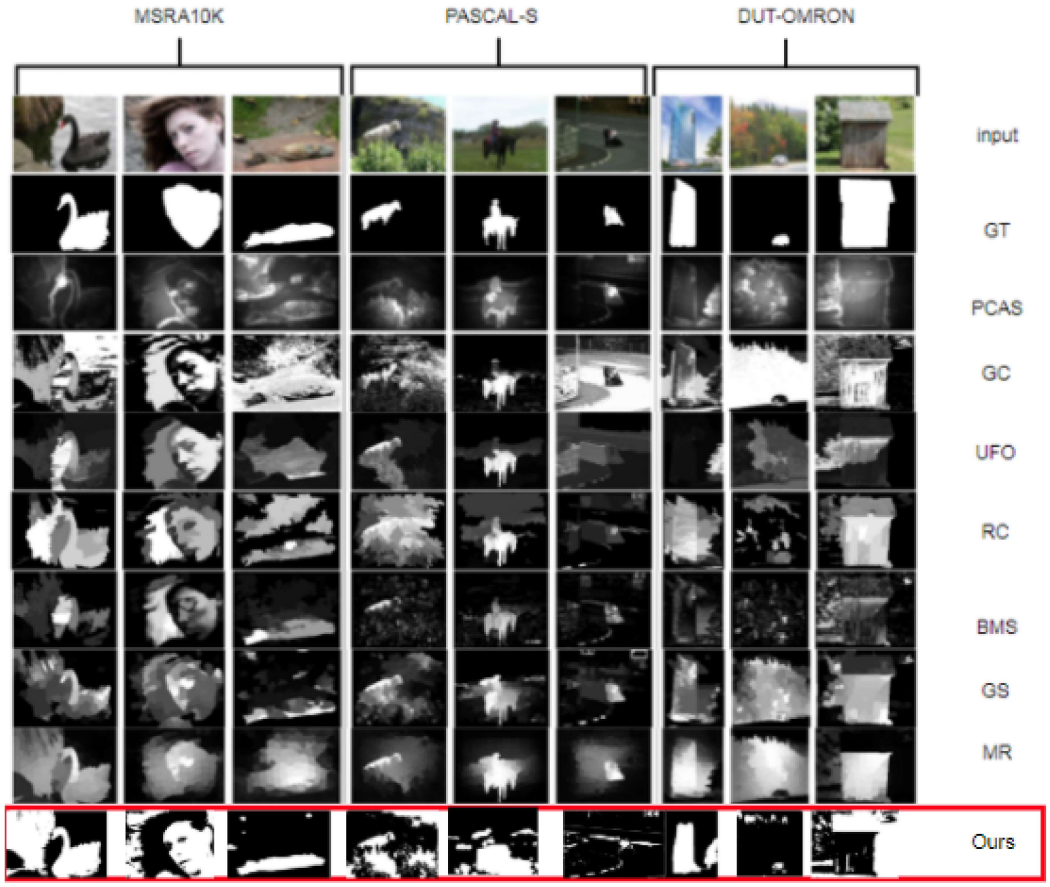

FIGURE 3.19: Output of different algorithms for sample images from different dataset.

3.4.4.1 Comparison with GVF and MDC algorithm

Experiments were conducted for salient object detection by using only GVF and only MDC saliency maps. FIGURE 3.20 clearly shows that they are not individually successful in extracting the salient object. Keeping the number of iterations the same, GVF is not able to converge cleanly to the object boundaries. In this type of case, MDC comes to help as it can provide cleaner object boundaries. For MDC, the problem occurs when the objects are sharply heterogeneous. If the object has a certain part similar to the background, the map causes a leakage situation and

FIGURE 3.20: Comparing the results of salient object detection using only GVF, only MDC, and the proposed algorithm. First column: Original image; Second column: Ground truth; Third column: Output of using only GVF algorithm; Fourth column: Output of using only MDC saliency map; Final column: Output of the proposed algorithm.

is not able to cover the object completely. This is very clear in row 2 of FIGURE 3.20. The MDC does not cover the transparent area in the ice. In such cases, GVF is of great help in covering the object as a whole. The last row signifies that with few iterations, the GVF has not yet started to converge. The saliency map leaves out the pink part of the object. In such cases also, the combination of both of them using the proposed algorithm brings up a neat result.

3.4.4.2 Comparison with other algorithms

The proposed model is compared with seven state-of-the-art algorithms for salient object detection. They are: BMS [179], GC [182], GS [183], PCAS [184], RC [166], DSR [181], SMD [185] and DRFI [180]. The results are shown in TABLE 3.2 and in FIGURE 3.21. For FIGURE 3.21, the values are obtained by running each algorithm

TABLE 3.2: Comparison of MAE of the proposed algorithm with state-of-the art algorithms

Methods \ Datasets	MSRA10K	DUT-OMRON	PASCAL-S
BMS [179]	0.2293	0.2263	0.2350
GC [182]	0.2304	0.2374	0.2461
GS [183]	0.1312	0.1419	0.1932
PCAS [184]	0.1775	0.1867	0.2157
RC [166]	0.1423	0.1520	0.2043
DSR [181]	0.1101	0.1404	0.1867
SMD [185]	0.1032	0.1465	0.2041
DRFI [180]	0.1233	0.1216	0.1948
Proposed Method	**0.1001**	**0.1200**	**0.1330**

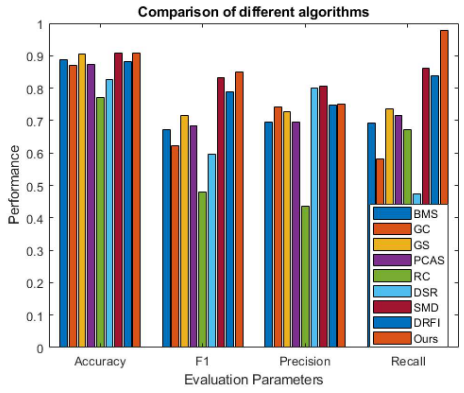

FIGURE 3.21: Performance of different algorithms.

for 1000 sample images taken from the three databases. PR-curves and F-measure curves, as shown in FIGURE 3.22, is drawn for all the algorithms.

It is visible from the visual output as well as the graph results that the proposed algorithm performs significantly better than these algorithms. The proposed algorithm's accuracy is very close to GS and BMS. However, for the other three factors, the proposed algorithm is better than them.

It is also worth mentioning that the high recall obtained is due to the minimum

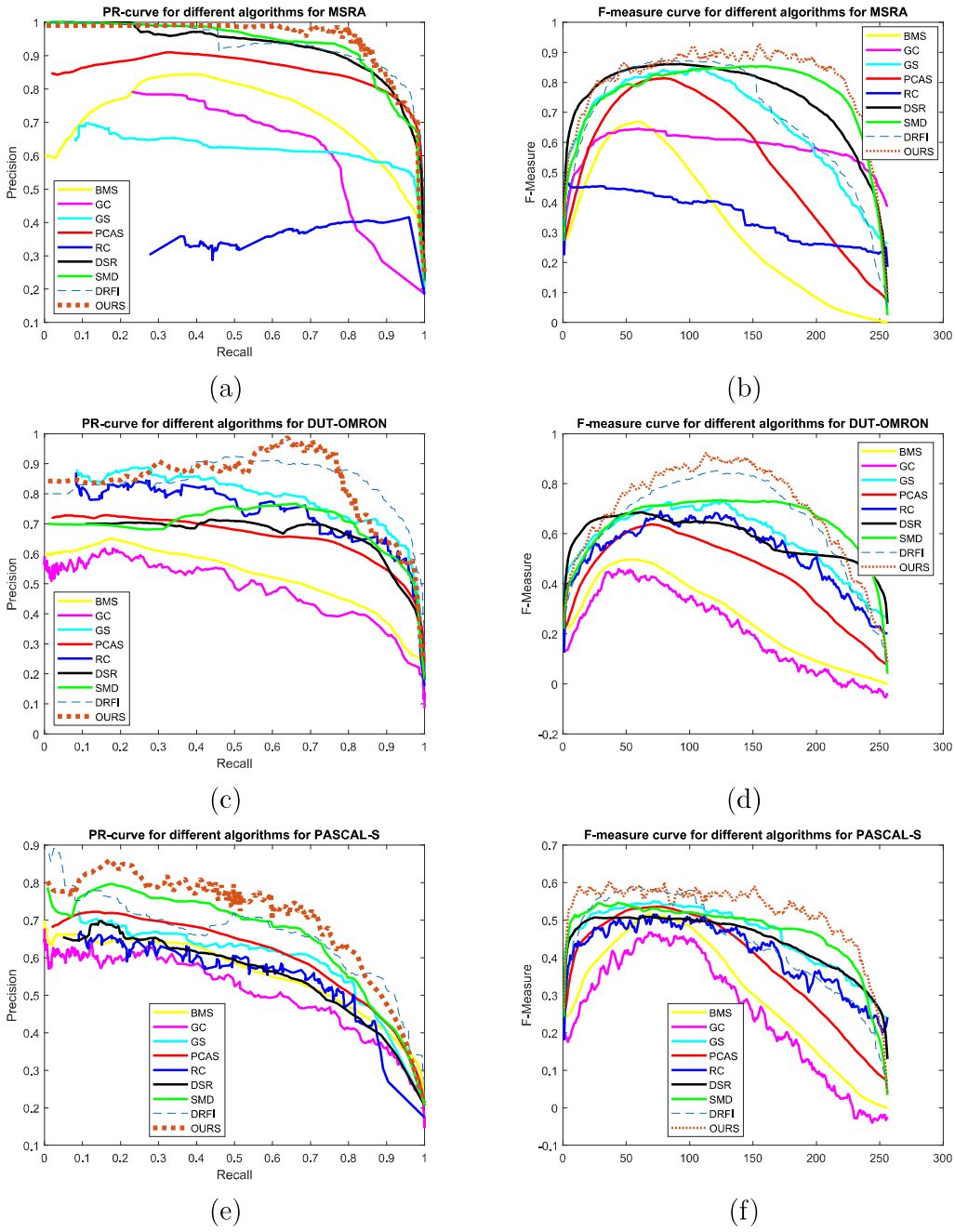

FIGURE 3.22: PR-Curves: (a) MSRA (c) DUT-OMRON (e) PASCAL-S
F-Measure Curves: (b) MSRA (d) DUT-OMRON (f) PASCAL-S.

directional contrast used for finding the salient object. The object located anywhere in the image (center or corner), will always have high MDC compared to the rest of

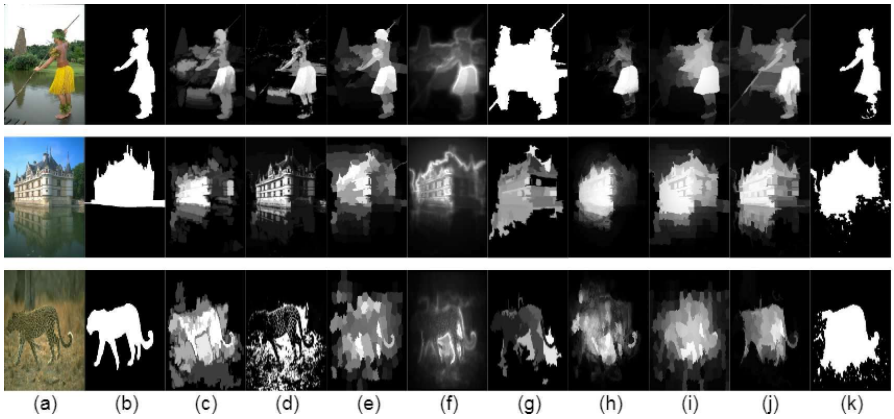

FIGURE 3.23: Visual comparison of the output of the different algorithms. From left to right: (a) Original image, (b) Ground truth, (c) BMS [179], (d) GC [182], (e) DSR [181], (f) GS [183], (g) PCAS [184], (h) RC [166], (i) SMD [185], (j) DRFI [180] (k) Output of proposed model.

the objects. So, in any case, this model will never miss the salient object. A figure depicting the output of different algorithms is shown in FIGURE 3.23.

3.4.4.3 Salient Object and Scene Analysis

For scene analysis, a relevant background (or scene) information is also required while finding the salient object. The weight factor, θ, plays a significant role here. In the result section, it is mentioned that θ is set as a meager 8% of the average MDC value. The lower the value of θ, the more it concentrates on the prime object. As the value of θ increases, surrounding objects next to the most salient object are included in the result. This can be shown by the FIGURE 3.24.

For experimental analysis, θ is kept low to match the algorithms that focus only on the salient object. From the results and graphs, it can be seen that not only

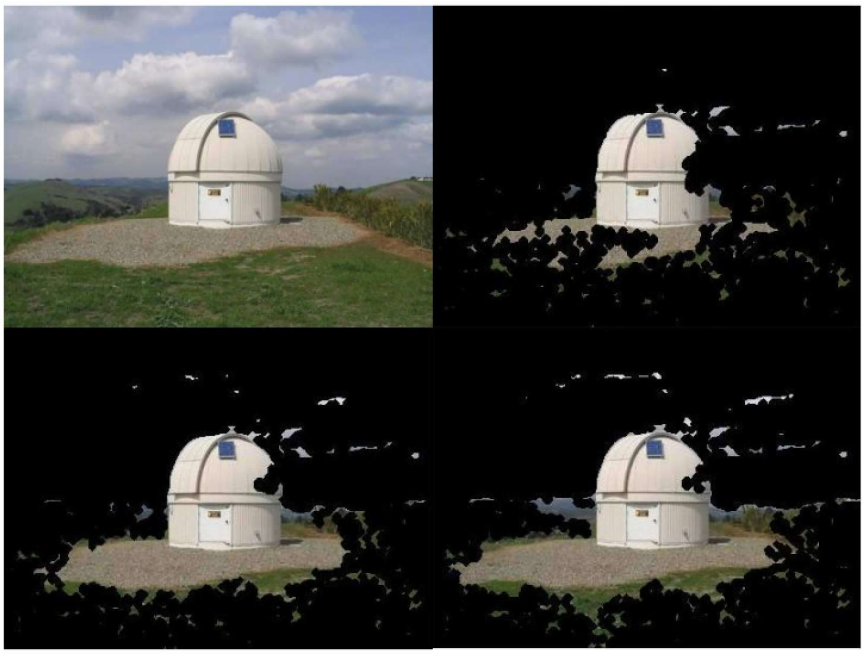

FIGURE 3.24: Visual depiction of how the algorithm moves from salient object to background objects on relevance basis by increasing the factor θ. In clockwise direction: Original image,output with θ=0.08, output with θ=0.2 and output with θ=0.5.

the proposed algorithm performs better than existing algorithms for salient object detection, but a slight modification to weight vector can also extend the proposed algorithm's scope to intelligent scene analysis. It is also worth mentioning that an increase in information does not increase the irrelevancy of the output. As in the above FIGURE 3.24, it can be seen that on increasing θ from 8% to 50%, the information about the sky is not added, which is irrelevant. The irrelevancy is mentioned here by asking nine candidates to mention three principal objects of the image shown above. Only 3 out of the nine answers mention cloud in the third place, whereas,

rest of the answers only included labels ‘house/building/stupa/shed’, ‘rocks/pebbles/gravel’, and ‘grass’. So it can be seen that irrelevancy is not getting added. Of the 9 answers mention cloud in the third place, whereas, rest of the answers only included labels ‘house/building/stupa/shed’, ‘rocks/pebbles/gravel’and ‘grass’. So it can be seen that relevancy is maintained.

3.5 Conclusion

This chapter proposed three statistical models for salient object detection. The first model was based on multi-level multi-plane thresholding and region-based segmentation. The second model was based on edge-based cues and active contour models for finding the salient object. The third model proposed to find a salient object using Gradient Vector Flow and Minimum Directional Contrast. All the models had been compared against various state-of-the-art methods. It can be observed that incremental performance is achieved using the models. Among the three models proposed, the first model has the least performance improvement, whereas the third model improves upon the state-of-the-art methods significantly.