Learning to Detect Salient Object

Mentor :
Prof. Amitabha Mukerjee

amit@cse.iitk.ac.in

Avinash Koyya	Diwakar Chauhan
(Y9156)	(Y9203)
avinashk@iitk.ac.in	diwakarc@iitk.ac.in

April 22, 2012

Abstract

The project aims to implement a solution to the salient object detection problem for images as proposed in [2]. The problem is formulated as a binary labeling task where the salient object is separated from the background. A set of features, namely, multiscale contrast, center-surround histogram and color spatial distribution, are obtained to describe a salient object locally, regionally, and globally. A conditional random field is learned to combine these features for salient object detection. The learning is done over a dataset of labelled images and the results thus obtained are put up.

Introduction

The human brain and visual system pay more attention to some parts of an image. Finding the regions of visual saliency or attention is of a great interest. Major applications for visual attention include automatic image cropping, adaptive image display on small devices, image/video compression, advertising design, and image collection browsing. Visual attention helps object recognition, tracking, and detection as well.

When the visual attention over an image is focused upon an object in the image, the object gaining the attention is called the salient object or foreground objects. An example would be the leaf in the image shown. The paper [2] studies one aspect of visual attentionsalient object detection.

Related Work

Most of the visual attention models are mostly based on the low level features of image such as intensity, contrast and motion.
There are three major steps in the evaluation of the saliency model: -
1) Feature Extraction
2) Saliency Map Computation
3) Selectiong key locations

Itti's model is based on the these steps. The low level features in his model are Color, Intensity and orientation.
Based on Itti's work, there is a toolbox in matlab named "saliencytoolbox".
These approaches don't have notation of object in the image. For correct detection of salient object by these algorithm, either parameters are needed to set and these parameters vary according to the type of object in the image or category of the object is required to know. Different type of salient object need different parameters for calculation. In our implementation we don't require prior information about the object in the image.

Formulation of Problem

The problem of finding salient object is formulated as follows:-
For a given image I of size $height$ x $width$, a binary mask M of same size is calculated $i.e.$

Figure 1: Comparison of different algorithms. From left to right: Itti's Saliency Map, proposed approach, and ground truth.

$$M(i,j) = \{m_x | m_x \in \{0,1\}\} \tag{1}$$

This label m_x indicates if pixel x belongs to the salient object.

Salient Object Features

These features are obtained to describe the local, regional and global properties of the image and thus incorporate them in the CRF learning. The method [2] proposes three features.

1. Multi Scale Contrast

2. Center Surround Histogram

3. Color Spacial Distribution

Multi Scale Contrast

Human visual receptive fields are more sensitive to the intensity changes. Thus, sudden changes in the contrast across the image are likely to gain human attention. Contrast feature is most commonly used feature in detection of salient object.

Figure 2: Intermediate Contrast Pyramid

For each image a six level gaussian pyramid of image is obtained and contrast map for image at each level of pyramid is calculated. The maps thus obtained over the levels are then added to give the resultant Multi Scale Contrast map.

$$f_c(x, I) = \sum_{l=1}^{6} \sum_{x' \in N(x)} ||I^l(x) - I^l(x')||^2 \tag{2}$$

Here $N(x)$ is the set of 8 pixels surrounding the pixel x and l is the pyramid level in the image. The in the gaussain pyramid, images each level are obtained by applying gaussian filter in both direction in the image. The gaussian kernel used is: 0.05 0.25 0.4 0.25 0.05

Figure 3: Images and Their Multiscale Contrast output

Multi Scale Contrast is a local feature of image. In this process the regions with similar colors are eliminated while the places where there is a large contrast change, are highlighted. Greater the change in contrast more highlighted is the region.

Center Surround Histogram

The salient object usually has a larger extent than local contrast and can be distinguished from its surrounding context. Therefore, a regional salient feature was proposed to identify the salient object based on how distinct it appears from its immediate surroundings.
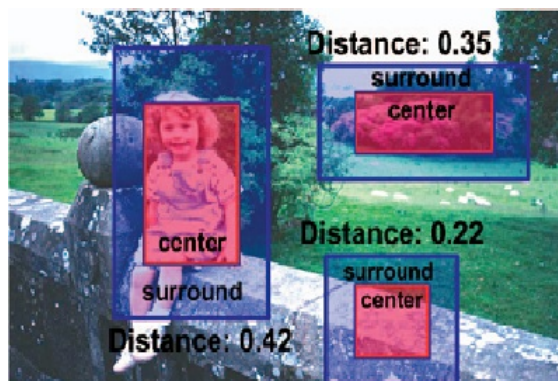

Figure 4: Center-surround histogram distances with different locations and sizes.

Suppose the salient object is enclosed by a rectangle R. Construct a surrounding contour R_S with the same area of R, as shown in the above figure. To measure how distinct the salient object

in the rectangle is with respect to its surroundings, the distance between R and R_S is taken to be the χ^2 distance between histograms of RGB color:

$$\chi^2(R, R_S) = \frac{1}{2}\sum \frac{(R^i - R_S^i)^2}{R^i + R_S^i} \tag{3}$$

where R^i is the histogram of rectangle R centered at pixel i.

A histogram is an array of numbers in which each element, bin, corresponds to the frequency of a range of values in the given data. In this case, each bin counts the number of pixels having color values in the same range. We use histograms because they are a robust global description of appearance. They are insensitive to small changes in size, shape, and viewpoint. The histogram of a rectangle with any location and size can be very quickly computed by means of an **integral histogram** [4].

An integral histogram is a map from every pixel co-ordinate in an image to a RGB colour histogram of the rectangle cornered by the top-left corner of the image and the pixel co-ordinate itself. Such a map is composed, starting from the top-left corner, traversing first to right and then to the bottom the value of the cumulative image at the current pixel is obtained by the addition of the left and the up pixel and subtraction of the upper left pixels cumulative values.This happens in O(height x width of image) operations. This makes it trivial to obtain the RGB colour histogram of a rectangle given by any co-ordinates, in the image in linear amount of computation.

$$hist_{rectangle} = hist_{bot-right} - hist_{bot-left} - hist_{top-right} + hist_{top-left} \tag{4}$$

To handle varying aspect ratios of the object, five templates with different aspect ratios { 0.5 , 0.75, 1 .0, 1.5, 2.0 } were used. The size range of the rectangle R(x) is set to [0.1, 0.7] (in steps of 0.1) x min(width ,height of image). The most distinct rectangle, $R^*(x)$, centered at each pixel x by varying the size and aspect ratio:

$$R^*(x) = argmax_{R(x)}\chi^2(R(x), R_S(x)) \tag{5}$$

Then,the center-surround histogram feature $f_h(x;I)$ is defined as a sum of spatially weighted distances:

$$f_h(x, I) \propto \sum_{\{x'|x \epsilon R^*(x')\}} w_{xx'}\chi^2(R^*(x'), R_S^*(x')) \tag{6}$$

where R*(x') is the rectangle centered at x' and containing the pixel x .

Figure 5: Images and Their Center Surround Histogram Features

Hence, this method ensures that each pixel in the vicinity of the most distinct rectangle inherits weight in the map, proportional to the distinctness of the rectangle. Hence the more number of distinct rectangles a pixel is a part of, the more weight it acquires in the map and the greater

opportunity to be a part of the salient object.

The weight

$$w_{xx'} = exp(-0.5\sigma_{x'}^{-2}||x - x'||^2) \tag{7}$$

is a Gaussian falloff weight with variance $\sigma_{x'}^2$, which is set to one-third of the size of R*(x'). This weight imposes a sense of distance from of the pixel in the vicinity, x' to that at the center of the distinct rectangle, x. The more the distance, the lesser proportion x' inherits from x. Finally, the feature map $f_h(.,I)$ is also normalized to the range [0, 1].

Color Spacial Distribution -

This feature is proposed on the notion that the more widely a color is distributed in the image, the less possible it is that a salient object contains this color. The global spatial distribution of a specific color can be used to describe the saliency of an object.

To describe the spatial distribution of a specific color, the simplest approach is to compute the spatial variance of the color.

The aimis to represent all colors in the image by Gaussian Mixture Models (GMMs) $\{w_c, \mu_c, \Sigma_c\}_{c=1}^C$, where $\{w_c, \mu_c, \Sigma_c\}$ is the weight, the mean color, and the covariance matrix of the cth component [1].

1. Initialize the means μ_k, covariances Σ_k and weight w_k, and evaluate the initial value of the log likelihood.

   The initialisation is done by K-means algorithm, which aims to partition the set of pixels in the given image into K clusters in which each observation belongs to the cluster with the nearest mean. We computed clusters for **K=6**, initialised the covariances and weights for each cluster.

   Then, the Gaussian Mixture Models are obtained as follows:

2. **Expectation** step. Evaluate the responsibilities using the current parameter values

$$\gamma(z_{nk}) = \frac{w_k \mathbf{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^K w_j \mathbf{N}(x_n|\mu_j, \Sigma_j)} \tag{8}$$

   $\gamma(z_{nk})$ is the responsibility for pixel value n with colour label x_n (refered to as I_x later) and cluster(or colour component) value k. K is the total number of colour components. $\mathbf{N}(x_n|\mu_k, \Sigma_k)$ is the normal distribution pdf.

Figure 6: Images and Their Color Spacial Distribution Output

3. **Maximisation** step. Re-estimate the parameters using the current responsibilities.

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) x_n \tag{9}$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k^{new})(x_n - \mu_k^{new})^T \tag{10}$$

$$w_k^{new} = \frac{N_k}{N} \tag{11}$$

where

$$N_k = \sum_{n=1}^{N} \gamma(z_{nk})x_n \tag{12}$$

4. Evaluate the log likelihood

$$ln\ p(\mathbf{X}|\mu, \mathbf{\Sigma}, \mathbf{w}) = \sum_{n=1}^{N} ln\{\sum_{k=1}^{K} \mathbf{N}(x_n|\mu_k, \Sigma_k)\} \tag{13}$$

and check for convergencr of log likelihood. If the convergence criterion is not satisfied return to step 2.

The resultant values of $\{w_k, \mu_k, \Sigma_k\}_{k=1}^{K}$ are taken to bethe required Gaussian Mixture Models (GMMs) $\{w_c, \mu_c, \Sigma_c\}_{c=1}^{C}$ where C=K.
Each pixel is assigned to a color component with the probability:

$$p(c|I_x) = \frac{w_c \mathbf{N}(I_x|\mu_c, \Sigma_c)}{\sum_{j=1}^{C} w_j \mathbf{N}(x_n|\mu_j, \Sigma_j)} \tag{14}$$

Then, the horizontal variance $V_h(c)$ of the spatial position for each color component c is

$$V_h(c) = \frac{1}{|X|_c} \sum_{x} p(c|I_x)|x_h - M_h(c)|^2 \tag{15}$$

$$M_h(c) = \frac{1}{|X|_c} \sum_{x} p(c|I_x)x_h \tag{16}$$

where x_h is the x co-ordinate of the pixel x and

$$|X|_c = \sum_{x} p(c|I_x) \tag{17}$$

The vertical variance $V_v(c)$ is similarly defined. The spatial variance of a component c is $V(c) = V_h(c) + V_v(c)$.
$V(c)$ is normalized to range [0,1].

Note that the spatial variance of the color at the image corners or boundaries may also be small because the image is cropped from the whole scene. Toreduce this artifact, a center-weighted, spatial-variance feature is defined as

$$f_s(x, I) \propto \sum_{c} p(c|I_x)(1 - V(c))(1 - D(c)) \tag{18}$$

where

$$D(c) = \sum_{x} p(c|I_x)d_x \tag{19}$$

is the weight which assigns less importance to colors nearby image boundaries and is also normalized to [0,1]. d_x is the distance from pixel x to the image center.

Dataset

We used the MSRA salient object database in the project. This database consists of two sets. One set consists of 20,000 images with images labelled by three users. The labelling was done by separately showing the image to an user and asking to enclose the salient object by a rectangle. The other set consists of highly consistent 5000 images. In these images the salient object is selected with no ambiguity. These images are labelled by 9 users.

Labelling of Images

We took 500 labelled images from second set for training. Then we calculated label on the image

$$a_x = \frac{1}{M} \sum_{i=1}^{M} a_x^m \qquad (20)$$

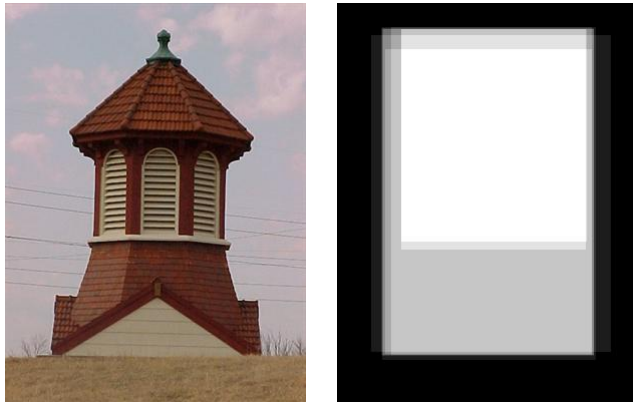M is number of users who have labelled the image.

Figure 7: Image and User Labels

CRF Learning

Once the individual salient object feature for images are obtained, the feature maps obtained are to be incorporated in optimum combination inorder to gain maximum consistency with the ground truth (or the probable mask).

In CRF framework, The probability of a labeling configuration M, given the observation image I , is modeled as a conditional distribution

$$P(M/I) = \frac{1}{Z} \exp\left(-E(M/I)\right) \qquad (21)$$

where Z is the partition function.
The energy $E(M/I)$ is defined as a linear combination of a set of static salient features, including a number of k unary features $f_1, f_2, \cdots, f_k$ and a pairwise feature $S(a_x, a_{x'}, I)$, which can be viewed as a penalty term when adjacent pixels are assigned with different labels. The more similar the colors of the two pixels are, the less likely it is that they are assigned different labels.

$$E(M/I) = \sum_x \sum_{k=1}^{K} \lambda_k f_k(a_x, I) + \sum_{x,x'} S(a_x, a_{x'}, I) \qquad (22)$$

7

where λ_k is the weight of the kth feature and x and x' are two adjacent pixels.
CRF maximizes the function below and gives the corresponding λ values.

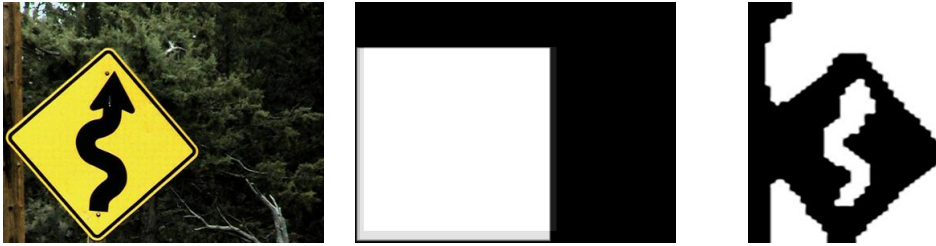$$\{\lambda\} = argmax_\lambda \sum_n log(P(M^n|I^n, \lambda)) \tag{23}$$

Figure 8: Images, Labels and Calculated Output

   This has been implemented with the help of the CRF2D module [3].
The module takes as input the feature data, and labelled data for training the images and calculated weights for features. For **400** training images the weights calculated were:-
0.0911, -0.1038, 0.0180, -0.1235, 0.9613, 0.1996, 0.1913, 0.4842

Evaluation of output

The validity of the output and its consistency with the user labels are measured by

1. Precision

2. Recall

3. F-measure

- The salient object is enclosed in a rectangle with pixel values set to 1 and set the rest of the image to 0.

- Normalise the user labelled data, the pixel values of labels to the range of [0,1] so that the values of the calculated and labelled output remain in consistent with each other for further calculation.

Let g_x is the pixel value of some pixel of the labelled image and a_x is the pixel value of the calculated salient object.

Precision

Precision represents the proportion of the computed output data that matches with the user labels.

$$precision = \frac{\sum_x g_x a_x}{\sum_x a_x} \tag{24}$$

Recall

Recall represents the proportion of the user labelled data that matches with the computed output.

$$recall = \frac{\sum_x g_x a_x}{\sum_x g_x} \tag{25}$$

F-measure

$$F = \frac{1.5 * precision * recall}{0.5 * precision + recall} \tag{26}$$

Results

For **400** training images the feature weights calculated by CRF learning were
0.0911, -0.1038, 0.0180, -0.1235, 0.9613, 0.1996, 0.1913, 0.4842

With **10** training images, evaluationvalues obtained are
Precision = 0.6198, Recall = 0.8212, F-measure = 0.6749

For **400** training images
Precision = 0.5295, Recall = 0.8567, F-measure = 0.6068

Limitations in Our implementation

When the object is quite similar to the background in color, contrast, pattern then lot of the background gets included in the reculting calculated object. And images where there are more than salient objects(or parts different from the background), all of them are included in the output along with the region between them.

Figure 9: From left: Original Image, Groung truth(in white). The object could not be differentiatied with the background and hence no salient object is reported.
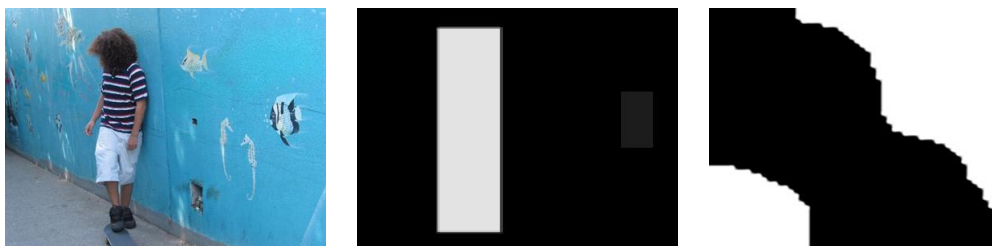

Figure 10: From left: Original Image, Groung truth(in white), Our result (in black)

Bibliography

[1] Christopher M. Bishop and Nasser M. Nasrabadi. *Pattern Recognition and Machine Learning.* *J. Electronic Imaging*, 16(4):049901, 2007.

[2] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011.

[3] Stéphane Nicolas, J. Dardenne, Thierry Paquet, and Laurent Heutte. Document image segmentation using a 2d conditional random field model. In *ICDAR*, pages 407–411, 2007.

[4] Fatih Murat Porikli. Integral histogram: A fast way to extract histograms in cartesian spaces. In *CVPR (1)*, pages 829–836, 2005.