Annexure 4 – Distinctive Issues

A-4.1 Search Engines - Google Image Search and vSearch

The topic presents query responses & related issues of state of the art image search technologies - Google Image Search and vSearch. Current version of the image search engine of Google, supporting image query or image url was launched recently in June 2011. Sources: (i) http://www.google.com/insidesearch/press/launch.html and (ii) http://computervisioncentral.com/content/google-rolling-out-content-based-image-search01668. Prior versions were supporting only textual queries. As being proprietary / commercial products, no authentic technical details of Google image search and vSearch are available. The inferred block diagram and related issues of Google search engine are illustrated below.

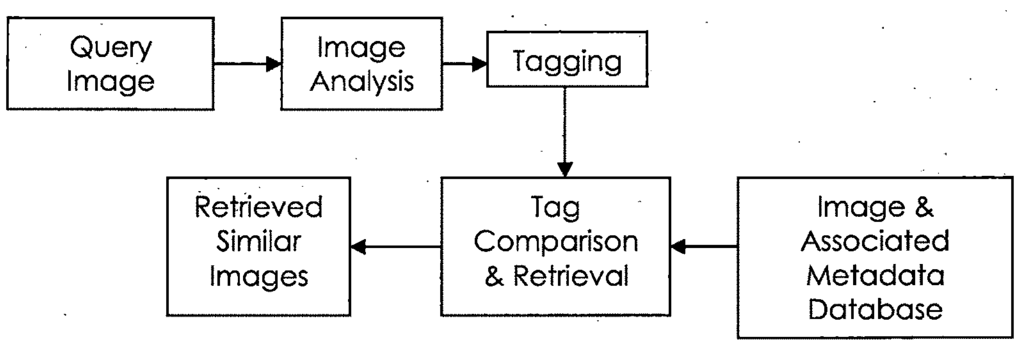


Figure 112. Inferred block diagram for Google like search engines

The Google achieves similarity retrieval by comparing Tags of the query image with pre-generated, stored & indexed Tags of database images. The Google addressed the issue of diversities in the human perception with the help of Google labeler. Google labeler was on line during 2006 to 2011. The Google labeler was a game to be played by multiple players who were given same images for manual labeling with all possible Tags. The same images would be given to a large number of users (players) to have exhaustive labeling. The exhaustive collections of image-tags have been utilized for the purpose of image retrieval. A tag for an image was subsequently weighted proportionate to the number of times it was perceived by the human beings. Hence, the label due to rarely or wrongly perceived contents would be given less weight, putting such image at the lower rank at the time of retrieval for a match due to least important tag.

Though the current version of the Google gives better results in terms of the Precision for many queries, including the illustrative one – *black rose*, it is not free from the limitations. Two typical query examples to illustrate current state of the art and limitations of the search engine are shown below in Figure 113 (as on 10-04-2012), where automatic tagging of query images was not performed by Google and the user was asked to describe the image. The selected query images are from the standard database - Berkeley Segmentation Dataset and Benchmark (BSDB) [Fowlkes, on line]. Without describing prompted image contents, Google produced results of visually similar images which **were not containing** any images of Baby girl and Tigers respectively for given two queries on the first page of retrieved results. As observed, retrieval was mainly based on the color distributions.

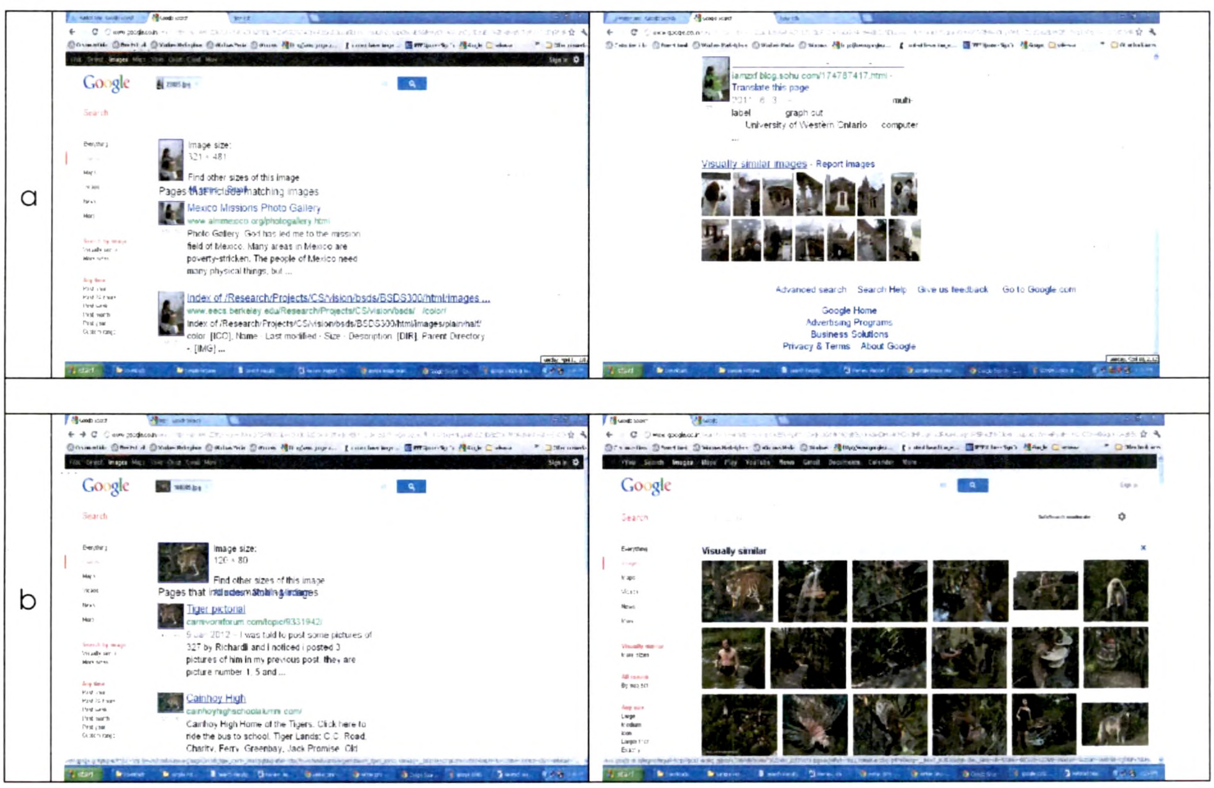


Figure 113. Query response for Google – could not tag & retrieving irrelevant images

Third typical Google query example shown below in Figure 114 is for the winter image of MS Windows operating system. The resulted automatic tags for the query image were winter & pins. The first tag is pertaining to a concept where as the second is a wrongly annotated tag, producing many dissimilar images on a first page.

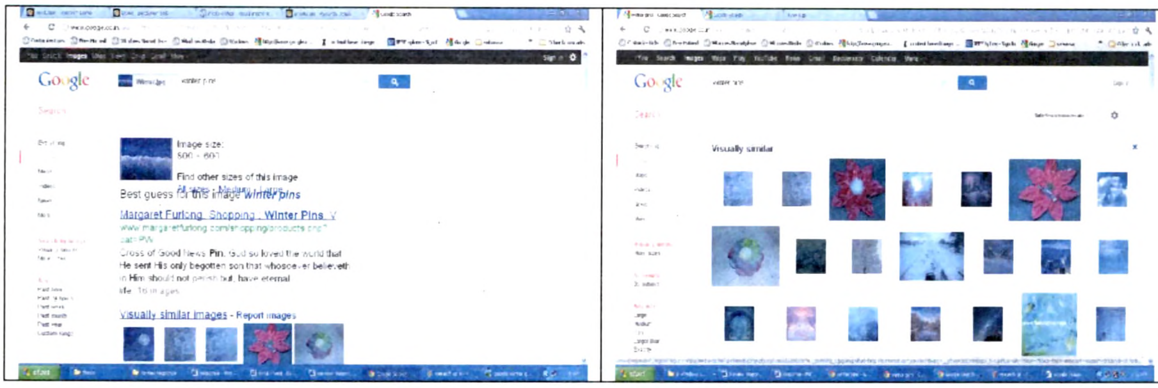


Figure 114. Query response for Google - tagging wrongly & retrieving many irrelevant images

Typical vSearch query results for 4 different queries are shown below in Figure 115. The top-left image in the table cells are respective query images. Resulted images of all 4 queries contain many dissimilar images. The inferred dominant method of similarity comparison is based on color distributions.

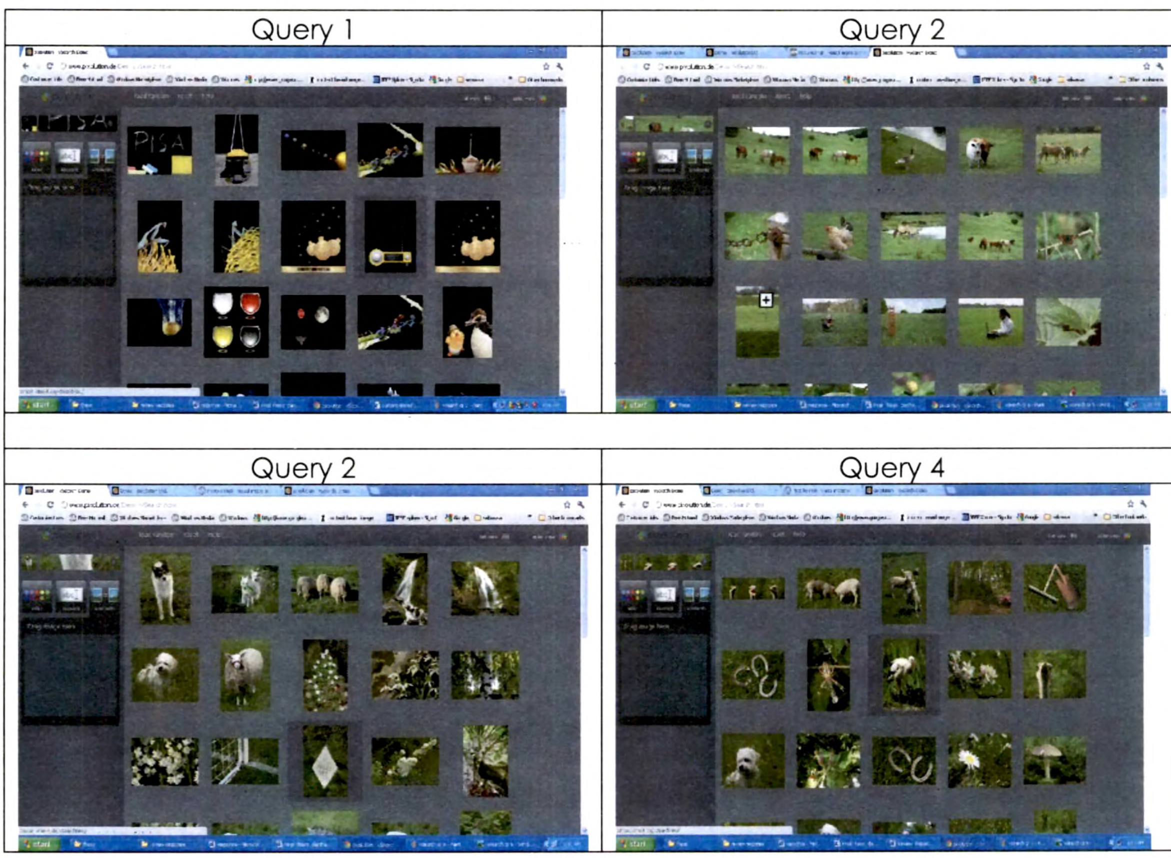


Figure 115. Query response for vSearch - retrieving many irrelevant (?) images

Our proposed novel techniques are based on the theme – "Relaxed feature description for better Recall and simultaneous emphasizing of reliable processing of cues leading to precise feature extraction for better Precision." The user has been given choice to select method of image retrieval to map his needs & perception. The broader color code description of whole image performs search on color similarity with higher Recall. The other options are foreground shape based technique, performing similarity comparison based on the detected foreground shape and a combinational technique that enables user to select proportionate weight of foreground shape and foreground color descriptors for similarity comparisons.

A-4.2 Quantitative Analysis & Comparisons of Edge Responses

Qualitative comparisons of edge responses of the proposed method with ACD Photo Editor, Adobe Photoshop and MS Photo Editor have been presented in Section 4.3.3. Figure 116 & Figure 117 show quantitative analysis of edge responses with performance measures $Precision_e$ (P_e), $Recall_e$ (R_e) and F – measure (F_e) along with qualitative comparisons for two sample images of BSDB [Fowlkes, on line] [Martin, 2001]. $Precision_e$ is a measure of how many detected edges are correct and $Recall_e$ is a measure of how many correct edges are detected with reference to ground truth. F-$measure_e$ (F_e) combines $Precision_e$ and $Recall_e$ to yield performance reflective single number given by $2 / (1/P_e + 1/R_e)$. These measures are computed for detected perceptually significant edges with reference to human segmented image of BSDB [Fowlkes, on line] [Martin, 2001]. The computation of $Precision_e$ (P_e) and $Recall_e$ (R_e) are carried out by locating detected edges in a vicinity of +/- one pixel in all directions with reference to ground truth edges.

Figure 116 (a) left and Figure 117 (a) left show original images of BSDB [Fowlkes, on line] [Martin, 2001]. Corresponding human segmented images are shown in Figure 116 (a) middle & Figure 117 (a) middle respectively. Figure 116 (a) right & Figure 117 (a) right present edge responses of proposed method with threshold 25. All three leading tools - ACD Photo Editor, Adobe Photoshop and MS Photo Editor produce & present edge response as a color image as shown in Figure 116 (b) to (d) & Figure 117 (b) to (d) at first column. These responses are converted to Gray images and thresholded with thresholds 25, 64 & 128 for precision & recall computations with white representing edge pixel. The RGB to Gray conversion & thresholding is performed with a Matlab program. It should also be noted that Adobe & MS Photo produce color edge responses characterized by white background with colored edges as shown in Figure 116 (c) & (d) and Figure 177 (c) & (d). And hence, corresponding Gray images are required to be negated before thresholding for carrying out quantitative comparisons.

Precision, Recall and F – measure plotted at different thresholds for quantitative comparisons of edge responses of the proposed method with ACD Photo editor, Adobe

Photoshop and MS Photo editor have been presented in Figure 118, Figure 119 & Figure 120 respectively.

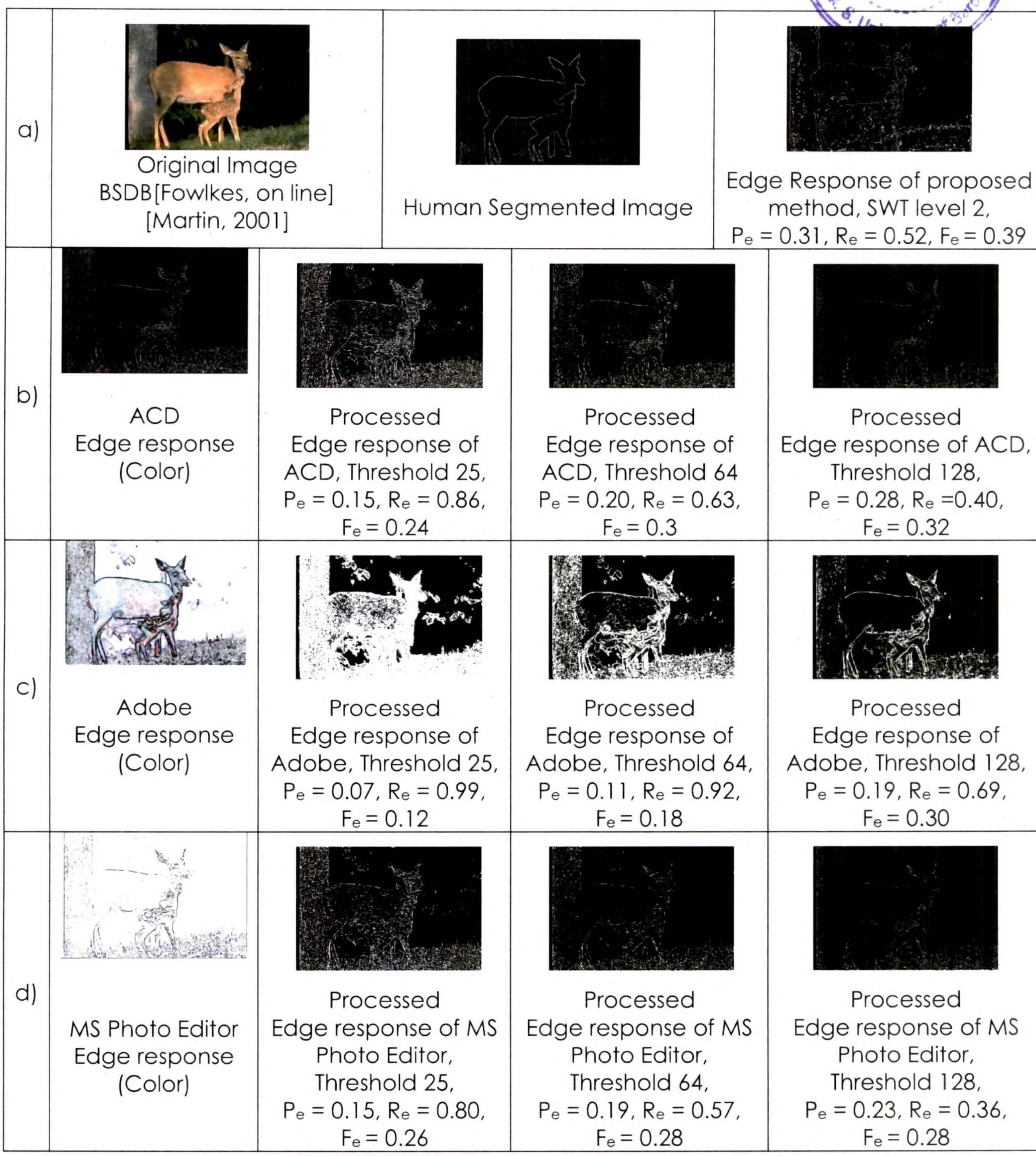


Figure 116. Edge Response Comparison & quantitative analysis – example 1

a)	 Original Image BSDB[Fowlkes, on line] [Martin, 2001]	 Human Segmented Image BSDB[Fowlkes, on line] [Martin, 2001]	 Edge Response of proposed method, SWT level 2, $P_e = 0.17$, $R_e = 0.53$, $F_e = 0.26$	
c)	 ACD Edge response (Color)	 Processed Edge response of ACD, Threshold 25, $P_e = 0.10$, $R_e = 0.93$, $F_e = 0.18$	 Processed Edge response of ACD, Threshold 64 $P_e = 0.12$, $R_e = 0.87$, $F_e = 0.22$	 Processed Edge response of ACD, Threshold 128, $P_e = 0.14$, $R_e = 0.75$, $F_e = 0.24$
d)	 Adobe Edge response (Color)	 Processed Edge response of Adobe, Threshold 25, $P_e = 0.05$, $R_e = 0.98$, $F_e = 0.09$	 Processed Edge response of Adobe, Threshold 64, $P_e = 0.06$, $R_e = 0.96$, $F_e = 0.11$	 Processed Edge response of Adobe, Threshold 128, $P_e = 0.08$, $R_e = 0.92$, $F_e = 0.14$
e)	 MS Photo Editor Edge response (Color)	 Processed Edge response of MS Photo Editor, Threshold 25, $P_e = 0.10$, $R_e = 0.91$, $F_e = 0.18$	 Processed Edge response of MS Photo Editor, Threshold 64, $P_e = 0.12$, $R_e = 0.83$, $F_e = 0.2$	 Processed Edge response of MS Photo Editor, Threshold 128, $P_e = 0.14$, $R_e = 0.70$, $F_e = 0.24$

Figure 117. Edge response comparison & quantitative analysis – example 2

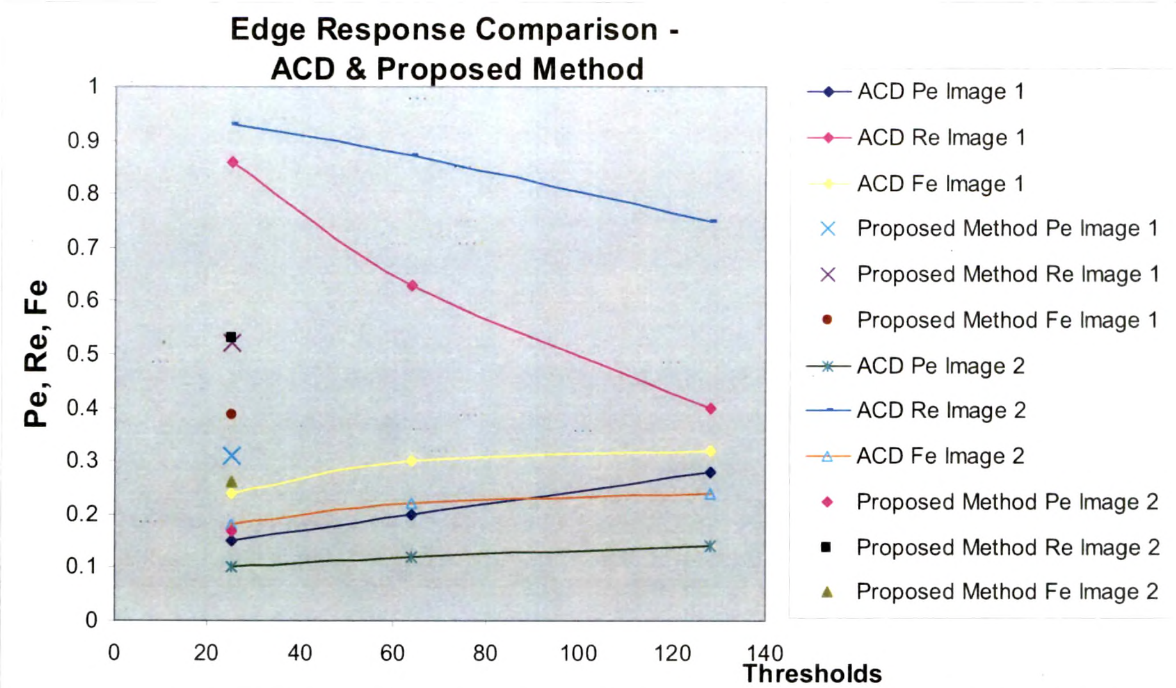


Figure 118. Quantitative comparison of edge responses: ACD Photo editor & proposed method

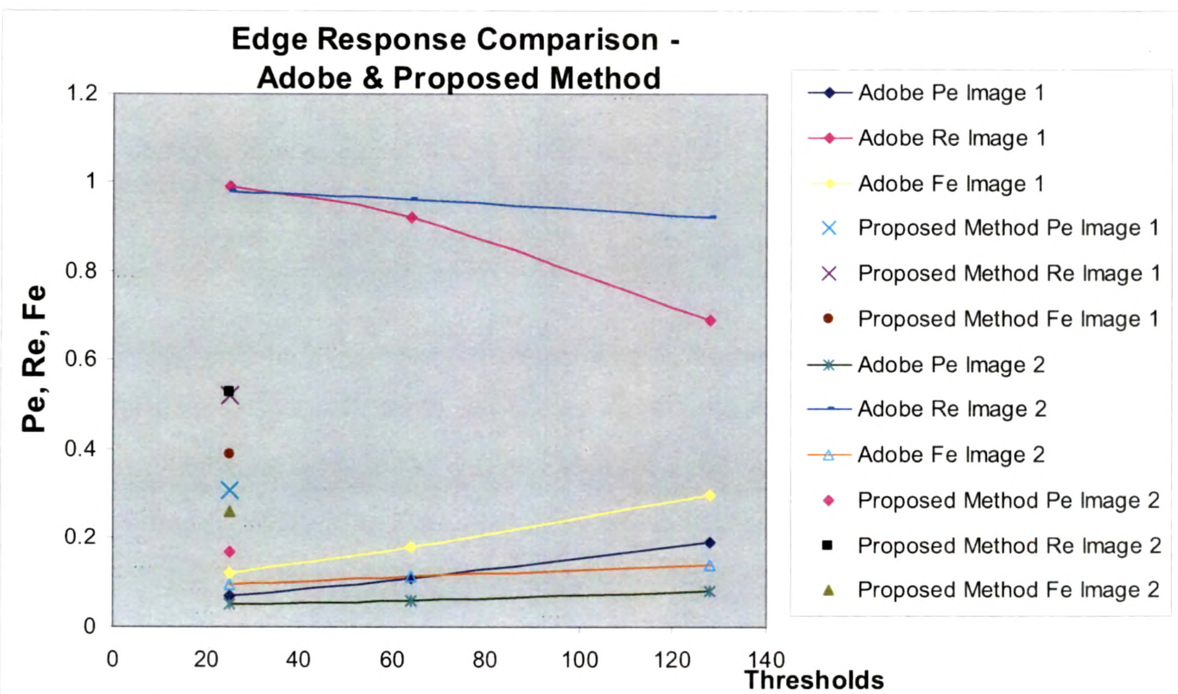


Figure 119. Quantitative comparison of edge responses: Adobe Photoshop & proposed method

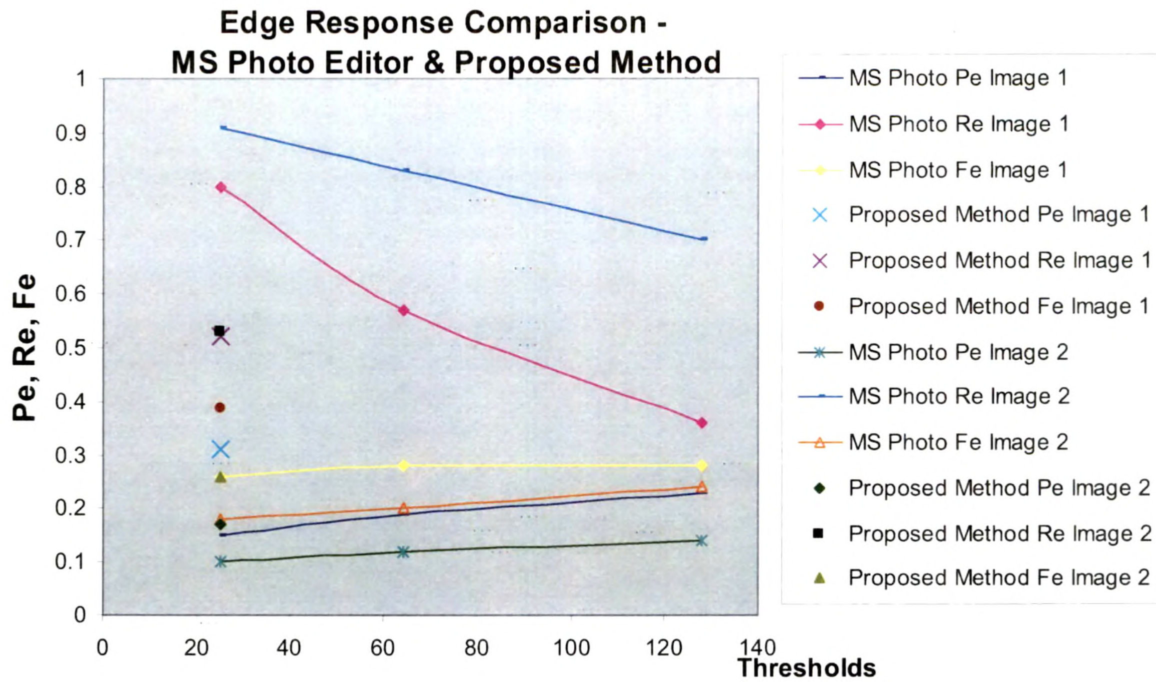


Figure 120. Quantitative comparison of edge responses: MS Photo editor & proposed method

A-4.2.1 Discussion

- Edge responses of the three software packages are characterized by low Precision resulted due to detection of perceptually significant as well as insignificant edges. Precision of Adobe is the lowest. Precision of the proposed method is better than others. The edge responses of the three packages are to be thresholded explicitly at different required levels for better Precisions & F-measures.
- Large numbers of detected edges yield better Recall for the three software packages compared to the proposed method. The higher Recall obtained in the three software packages is at a cost of Precision. F-measures of the proposed method is better than the three software packages.
- The results of the proposed method outperform others for i) detection of significant perceptual edges ii) elimination of insignificant edges corresponding background and foreground textures.

A-4.3 Quantitative Analysis of Results of Proposed Method for Foreground Extraction w. r. t. Ground Truth

The quantitative comparisons of the results of proposed method for foreground extraction have been carried out with performance measures $Precision_{fg}$ and $Recall_{fg}$, computed with respect to Ground Truth foreground. High values of the performance measures are noteworthy (Figure 121 to Figure 126).

The computation of $Precision_{fg}$ and $Recall_{fg}$ differs for region based segmentation compared to the Precision and Recall used for measuring image retrieval performances. They are defined as follows:

$Precision_{fg}$ is a ratio of area of intersection of detected foreground regions with Ground Truth foreground region to area of Detected foreground regions.

$Recall_{fg}$ is a ratio of area of intersection of detected foreground regions with Ground Truth foreground regions to area of Ground Truth foreground regions.

E.g., $Precision_{fg}$ of 0.5 indicates that correctly detected foreground (with reference to Ground-Truth) is 50 % of the total detected foreground. $Recall_{fg}$ of 0.6 indicates that the correctly detected foreground (with reference to Ground-Truth) is 60% of the total correct (Ground-Truth) foreground.

In addition to a large number of results presented in the thesis for foreground extraction for qualitative comparisons with the Human segmented images of BSDB [Fowlkes, on line] [Martin, 2001], qualitative & quantitative analysis for performance measures have been carried on sample images of BSDB [Fowlkes, on line] [Martin, 2001], SIMPLIcity [SIMPLIcity, on line] and ALOI [ALOI, on line] [Geusebroek, 2001].

Ground Truth Foreground Images: The BSDB [Fowlkes, on line] [Martin, 2001] provides Human segmented Ground Truth images. These images contain segmented foreground and background regions. And hence, the Ground Truth foreground images have been produced manually using Adobe Photoshop from these Human segmented images. The images from other databases have been also processed with Adobe Photoshop to generate Ground Truth foreground images.

Following Figures give the qualitative and quantitative comparisons of the results with the Ground Truth. The original images and corresponding human segmented images of BSDB [Fowlkes, on line] [Martin, 2001] are shown in Figure 121(a) and Figure 122 (b) respectively. Figure 121 (c) shows Ground-Truth foreground images produced with Adobe Photoshop from respective images of Figure 121 (b). The foreground regions are marked with White. Figure 121 (d) indicates level of Haar SWT used for proposed algorithm. Figure 121 (e) shows the extracted foreground regions from original images of (a) with proposed foreground extraction algorithm. These extracted foreground regions are marked with White and can be qualitatively compared with the corresponding Ground Truth foreground shown in Figure 121 (c). These foreground regions (White) of Figure 121 (e) are mapped to images to yield foreground images shown in Figure 121 (f) containing background marked as Black. Figure 121 (g) and Figure 121 (h) are the quantitative measures of Precision$_{fg}$ and Recall$_{fg}$ respectively, for extracted foreground with proposed algorithm with reference to the Ground Truth foreground.

Figure 122 illustrates results and quantitative & qualitative analysis for foreground extraction carried out at two different levels of Haar SWT decomposition for images of BSDB [Fowlkes, on line] [Martin, 2001]. The chart for Precision$_{fg}$ & Recall$_{fg}$ for results of Figure 121 and Figure 122 have been presented in Figure 123 indicative of high average Precision$_{fg}$ & high average Recall$_{fg}$.

Figure 124 and Figure 125 illustrate high performance measures with respect to ground truth for images of other databases - SIMPLIcity [Wang, 2001] [SIMPLIcity, on line] and ALOI [ALOI, on line] [Geusebroek, 2001]. The corresponding chart has been presented in Figure 126.

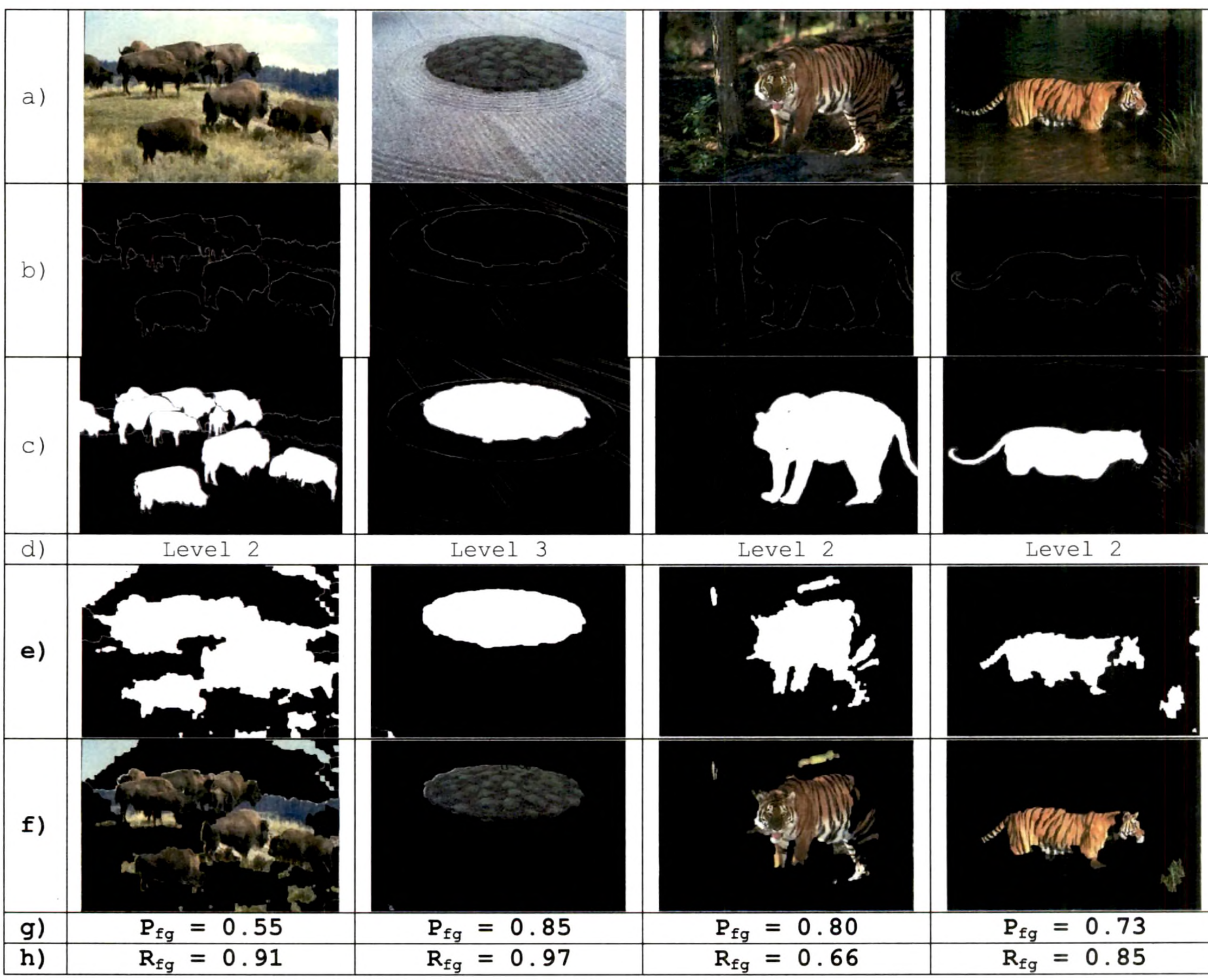


Figure 121. Qualitative & Quantitative Performance Comparisons for foreground extraction. (a) Original BSDB Images BSDB [Fowlkes, on line] [Martin, 2001] (b)Human segmented Ground Truth images at BSDB[Fowlkes, on line] [Martin, 2001]. (c)Ground Truth foreground from (b), produced with Adobe Photoshop. (d) Level of Haar SWT used. (e) Extracted foreground regions from Original images of (a) produced with proposed algorithm. (f) Corresponding foreground image, mapped from (e). (g) and (h) Precisionfg and Recallfg respectively for extracted foreground regions of (e).

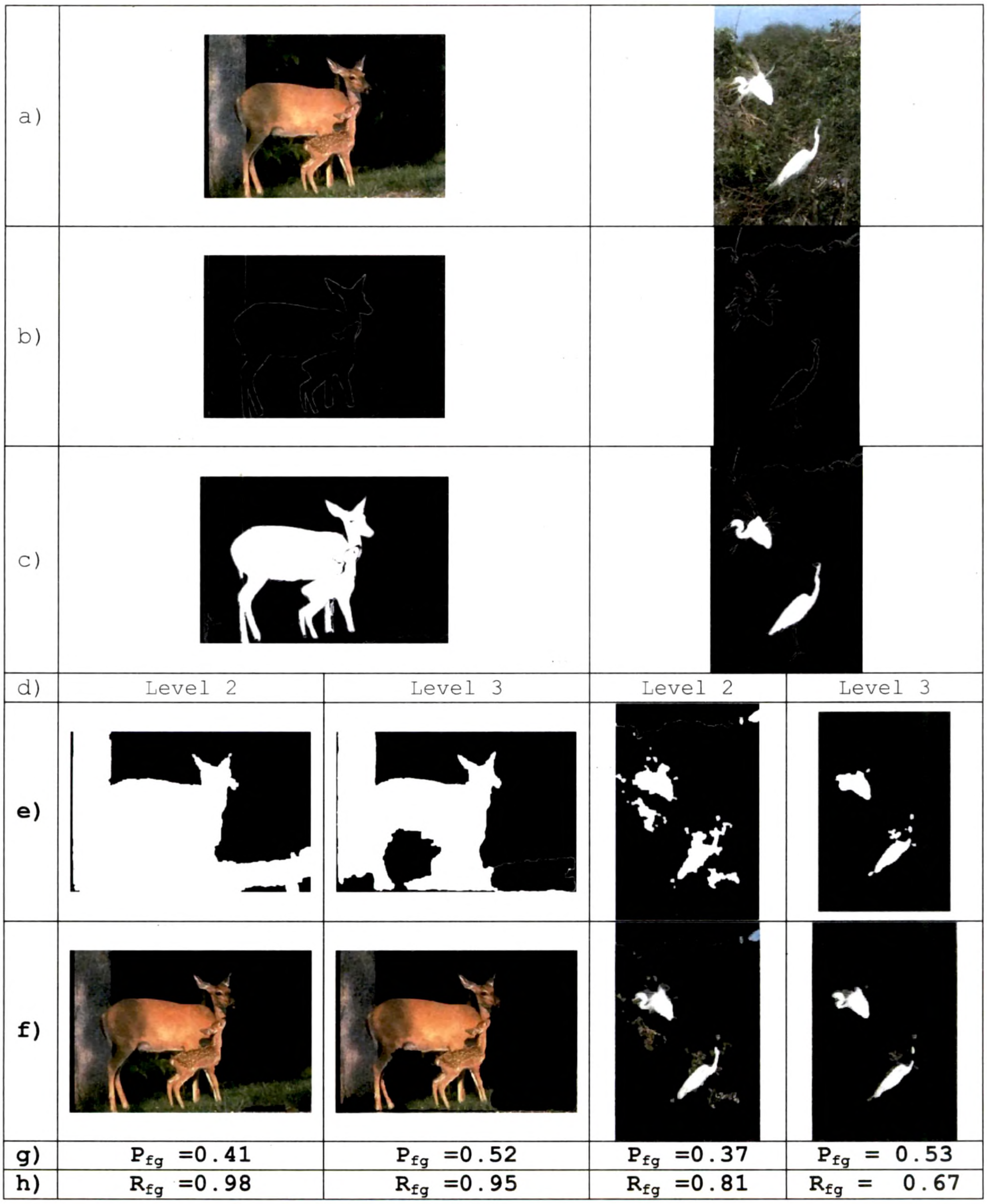


Figure 122. Qualitative & Quantitative Performance Comparisons for foreground extraction with respect to different levels of Haar SWT. (a) Original BSDB Images BSDB [Fowlkes, on line] [Martin, 2001]. (b)Human segmented Ground Truth images at BSDB[Fowlkes, on line] [Martin, 2001]. (c) Ground Truth foreground from (b), produced with Adobe Photoshop. (d) Level of Haar SWT used. (e) Extracted foreground regions from Original images of (a) produced with proposed algorithm. (f) Corresponding foreground image, mapped from (e). (g) and (h) Precisionfg and Recallfg respectively for extracted foreground regions of (e).

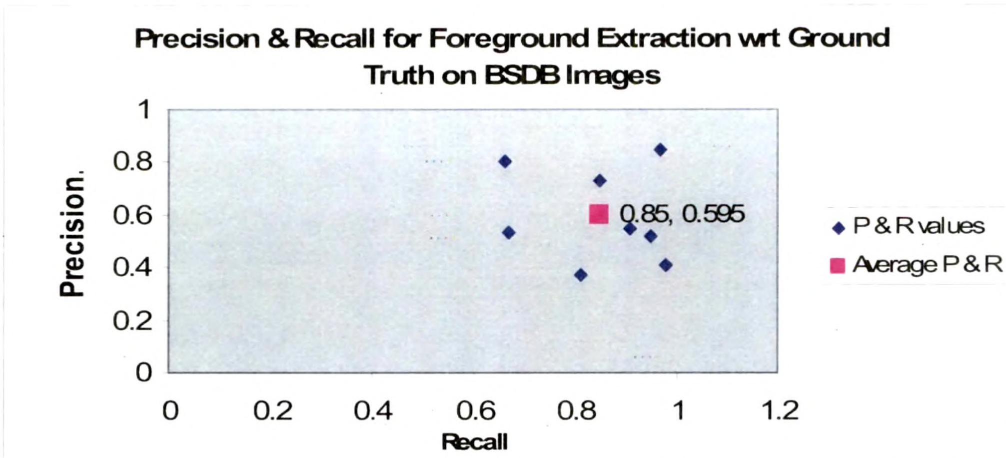


Figure 123. Quantitative analysis w.r.t. ground truth for foreground extraction on BSDB Images

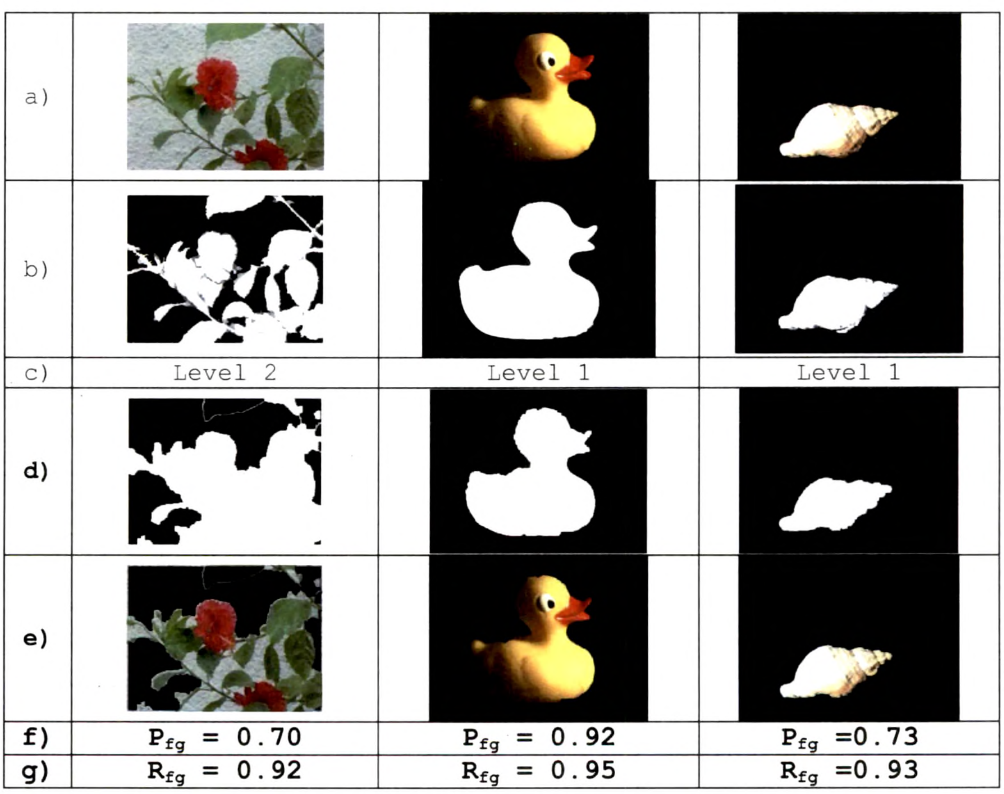


Figure 124. Qualitative & Quantitative Performance Comparisons for foreground extraction on images with illumination variations. (a) Original images, Left - size reduced image photographed by an amateur, Middle & Right from ALOI [ALOI, on line] [Geusebroek, 2001]. (b) Ground Truth foreground from (a) produced with Adobe Photoshop. (c) Level of Haar SWT used. (d) Extracted foreground regions from Original images of (a) produced with proposed algorithm. (e) Corresponding foreground image, mapped from (d). (f) and (g) Precisionfg and Recallfg respectively for extracted foreground regions of (d).

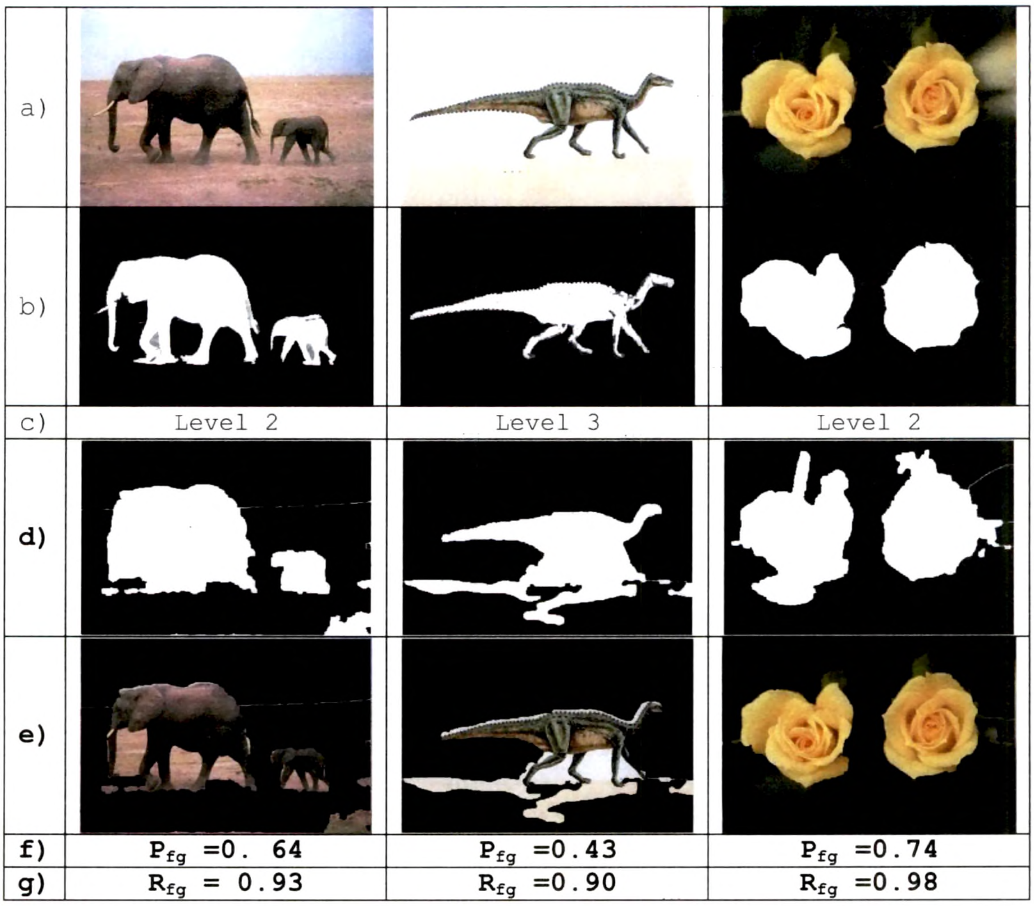


Figure 125. Qualitative & Quantitative Performance Comparisons for foreground extraction. (a) Original images [Wang, 2001] [SIMPLIcity, on line]. (b) Ground Truth foreground from (a) produced with Adobe Photoshop. (c) Level of Haar SWT used. (d) Extracted foreground regions from Original images of (a) produced with proposed algorithm. (e) Corresponding foreground image, mapped from (d). (f) and (g) $Precision_{fg}$ and $Recall_{fg}$ respectively for extracted foreground regions of (d).

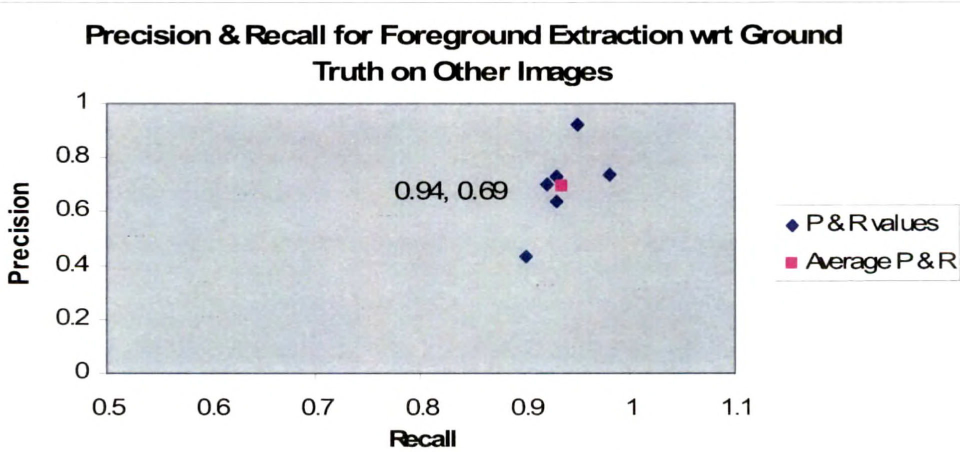


Figure 126. Precision – Recall analysis w. r. t. ground truth for foreground extraction on BSDB images [Fowlkes, on line]

A-4.3.1 Discussion

- The proposed method of foreground extraction is effective on diversified images as can be concluded by qualitative & quantitative comparisons of Ground Truth foregrounds with extracted foregrounds with proposed method. $Precision_{fg}$ & $Recall_{fg}$ - quantitative performance measures with the respect to Ground truth are quite high.
- The $Precision_{fg}$ of extracted foreground is high even for the complex natural images of BSDB [Fowlkes, on line] [Martin, 2001]. The average Precision of 0.595 with average Recall of 0.85 for sample image set is quite significant.
- The work reported so far in the literature for foreground extraction of challenging images of BSDB [Fowlkes, on line] [Martin, 2001] is mainly based on Graph cuts. The proposed method and the qualitative results with high Precision & Recall are unique, novel & not reported so far.

A-4.4 Color Codes

Innovative and unique 27 (25 + 2) color codes to effectively represent entire spectrum of RGB color space (2^{24} colors) are formulated and used for the purpose of image segmentation and feature extraction leading to color similarity based image retrieval. The codes are formulated by exploiting intra-tuple RGB relationship as shown in Table 27. A Color Code represents a set of colors satisfying corresponding intra-tuple RGB relationship. These color codes are the broadest color descriptors used to represent color attributes of images. Color code_0 represents pure black (with R = G = B = 0) differentiating it from Code_1. Code_26 is a special code to represent colors around boundaries of color codes.

A-4.4.1 Results - Color Code Based Segmentation

Figure 127 illustrates effectiveness of proposed novel color codes to represent image color attributes leading to image segmentation. The depicted sample images are some of the most challenging images of standard databases of BSDB [Fowlkes, on line] [Martin, 2001] and SIMPLIcity [Wang, 2001]. A standard image of Baboon possessing typical textures and color combinations is also segmented effectively with 4.1 seconds as segmentation time on the dual core processor with 1.49 GB of RAM. Segmentation time analysis is further reported in Section A-4.5.

Despite being broadest color descriptors, their effective representations for colors of images have been exploited for image retrieval to increase the Recall. The combination of foreground color codes and foreground shape with selectable percentage weight for image retrieval not only maps the need & perception of a user but also increases the Precision without sacrificing Recall much.

Table 27. Formulation of Color Codes

Sr. No.	Code	R G B Relationship Deciding Set of Colors Mapping to Respective Color Code
1.	Code_1	R =B=G & R != 0
2.	Code_2	R=G & R >B
3.	Code_3	R=B & R > G
4.	Code_4	G=B & G >R
5.	Code_5	R = G & B > R
6.	Code_6	R = B & G > R
7.	Code_7	G = B & R >G
8.	Code_8	R >B>G & (R-B) = (B-G)
9.	Code_9	R >B>G & (R-B) > (B-G)
10.	Code_10	R >B>G & (R-B) < (B-G)
11.	Code_11	R > G > B & (R-G) = (G - B)
12.	Code_12	R > G>B & (R-G) > (G-B)
13.	Code_13	R > G>B (R-G) < (G-B)
14.	Code_14	G > R>B & (G-R) = (R - B)
15.	Code_15	G > R>B & (G-R) > (R - B)
16.	Code_16	G > R>B & (G-R) < (R - B)
17.	Code_17	G > B>R & (G-B) = (B - R)
18.	Code_18	G > B>R & (G-B) > (B - R)
19.	Code_19	G > B>R & (G-B) < (B - R)
20.	Code_20	B > G > R & (B - G) = (G – R)
21.	Code_21	B > G > R & (B –G) > (G – R)
22.	Code_22	B > G > R & (B –G) < (G – R)
23.	Code_23	B > R > G & (B – R) = (R – G)
24.	Code_24	B > R > G & (B – R) > (R – G)
25.	Code_25	B > R > G & (B – R) < (R – G)
26.	Code_0	R = B = G = 0
27.	Code_26	Special

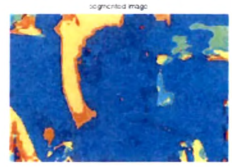
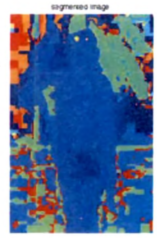
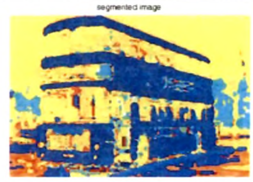
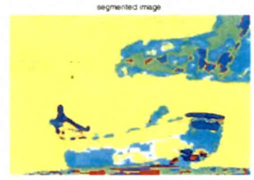
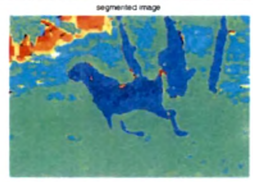
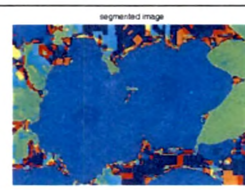
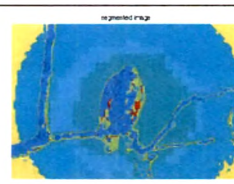
Sr. No.	Original Images SIMPLIcity [Wang, 2001], BSDB [Fowlkes, on line] [Martin, 2001]	Color code based segmented images
1)	1.jpg 	
2)	44.jpg 	
3)	310.jpg 	
4)	122.jpg 	
5)	740.jpg 	
6)	670.jpg 	
7)	42049.jpg 	

Figure 127. Results: Color codes based segmentation.

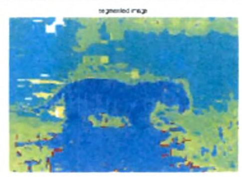
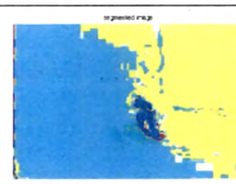
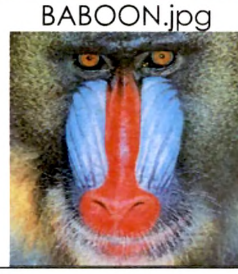
Sr. No.	Original Images SIMPLIcity [Wang, 2001], BSDB [Fowlkes, on line] [Martin, 2001]	Color code based segmented images
9)	108073.jpg 	
10)	30091.jpg 	
11)	BABOON.jpg 	

Figure 127 (Contd.). Results: Color codes based segmentation.

A-4.5 Processing Time Analysis

The development, testing & implementation of algorithms has been carried out on a machine having dual core Intel processor (T2050 @ 1.6 GHz) with 1.49 GB of RAM. The high processing time required particularly for prominent boundaries based algorithms demands high end servers for their deployment at real time.

Table 28 presents processing time for segmentation with Color codes and total processing time for edge & prominent boundaries detection and foreground extraction on sample images of SIMPLIcity images [Wang, 2001] of size 384 x 256. The processing time for boundary detection based approach is tabulated for SWT Haar level 1 and level 2. Whereas Table 29 summarizes processing time for various images of BSDB images [Fowlkes, on line] [Martin, 2001] of size 421 X 381 processed for color codes based segmentation and boundary based algorithms with SWT Haar decomposition at level 2 and level 3.

Table 28. Processing time analysis for SIMPLIcity images

Sr. No	SIMPLIcity Images [Wang, 2001] Size - 384 x 256	Processing Time Seconds	
		Color Code Based Segmentation Algorithm	Boundary Detection Based Algorithms
		Attempt 1	SWT Haar Level 1
		Attempt 2	SWT Haar Level 2
1	1.jpg 	1.9	234.9
		1.65	205.2
2	44.jpg 	1.5	154.1
		1.85	103.3
3	310.jpg 	1.68	2177.7
		2.32	605.2
4	122.jpg 	1.54	677.6
		1.77	809
5	740.jpg 	1.90	91.1
		1.79	177.76
6	670.jpg 	2.1	229.75
		2.7	141.3

Table 29. Processing time analysis for BSDB images

Sr. No	BSDB Images [Fowlkes, on line] [Martin, 2001] Size - 421 X 381	Processing Time Seconds	
		Color Code Based Segmentation Algorithm	Boundary Detection Based Algorithms
		Attempt 1	SWT Haar Level 2
		Attempt 2	SWT Haar Level 3
1	42049.jpg 	1.64	544.7
		1.89	374.3
2	300091.jpg 	1.93	232.2
		1.7	175.7
3	296059.jpg 	1.78	246.3
		2.03	226.7
4	38092.jpg 	1.65	438.5
		1.70	290.5
5	108005.jpg 	1.70	547.5
		1.68	245.3
6	108073.jpg 	1.71	393.7
		1.78	228.2

A-4.5.1 Discussion

- Color code based segmentation requires less time compared to boundary detection based algorithms.
- Processing time of images containing textures is significantly high for boundary detection based algorithms. Processing time reduces at higher levels of SWT.
- Processing time of color codes based segmentation does not depend on textures or categories of images. It is merely proportionate to the size of the image.
- Considering effectiveness and suitability of proposed boundary detection based algorithms, high end servers are needed & recommended to meet their computational requirements.