

A Robust Co-saliency Object Detection Model by Applying CLAHE and Otsu Segmentation Method

Anuj Mangal¹, Hitendra Garg², Charul Bhatnagar³

Submitted: 28/06/2023

Revised: 07/08/2023

Accepted: 27/08/2023

Abstract: The co-saliency detection technique is utilized in several applications namely image retrieval, picture annotation, as well as surveillance, among others. Its purpose is to identify the most significant and comparable patterns from a relevant group of images. In this study, there has been proposed a novel approach to co-saliency detection that utilizes the modified ResNet 50 model as a classification model. The contrast-limited adaptive histogram equalization (CLAHE) along with Otsu segmentation techniques were used as pre-processing steps to better identify and isolate prominent objects in the image. These techniques helped the model recognize patterns more efficiently. The Thin ResNet model was fine-tuned and optimized for improved accuracy, and the SGDM optimizer was used for network compilation. For training and testing of the model, we used the Cosal2015 and COCO datasets, with a 70:30 ratio. The proposed model demonstrated superior performance metrics such as MAE as well as F-measure score in comparison to state-of-the-art techniques. This proposed model obtains a high F1-score of 98.6% along with MAE of 0.034 on widely known Cosal2015 dataset. Also, this model obtained F1-score and MAE scores 94.2%, and 0.164, respectively on the COCO dataset.

Keywords: CLAHE, Cosal2015 Dataset, COCO dataset, Otsu segmentation, Object Detection, Thin ResNet.

1. Introduction

Co-saliency is a method that aims to recognize prominent and repeated patterns in a group of images. It involves identifying similar and prominent foreground portions in an image group by partitioning the group into four portions: related foreground, related background, unrelated foreground, and unrelated background. Finding the most recurrent as well as comparable objects or patterns in the distinct cluster of pictures is primary objective of co-saliency algorithms. To determine which foreground region in a series of images is the most similar, different co-saliency algorithms rely on three criteria: (a) obtain the descriptive characteristics that identify the image's foreground location, (b) detect elements that describe the co-saliency characteristics, and (c) build an efficient structure [1], [2].

Co-saliency refers to a relatively new field of computer vision that involves detecting and grouping salient objects in multiple images or videos that are semantically related or have a common context. Unlike traditional saliency detection, which focuses

¹ Department of CEA, GLA University, Mathura, India

Email Id- anuj.mangal@gla.ac.in

² GLA University, Mathura, India

Email Id- hitendra.garg@gla.ac.in

³ GLA University, Mathura, India

Email Id- charul@gla.ac.in

on identifying the most visually conspicuous object in a particular picture, the co-saliency analysis seeks to determine the objects that stand out within a cluster of associated pictures or videos, providing a more comprehensive understanding of the scene [3], [4]. Co-saliency is useful in multifarious applications such as image and video segmentation, object recognition, as well as scene understanding, among others [5]–[7]. The field is still evolving, with researchers exploring new algorithms and approaches for enhancing accuracy as well as efficiency of co-saliency detection [8], [9]. Co-saliency analysis typically involves two main steps: co-saliency detection as well as co-saliency refinement. Furthermore, the co-saliency detection involves identifying all communal regions or salient features across a group of related pictures, while co-saliency refinement involves further processing and refinement of the co-saliency information to improve its accuracy and usefulness for specific applications [10].

There have been several models proposed for co-saliency detection and refinement over the years. The Multi-Task Deep Co-Saliency model [11] uses a deep CNN to jointly detect and refine co-saliency maps for the group of associated pictures. For improving overall accuracy of co-saliency detection, the model learns to exploit both inter as well as intra-image information. Co-Saliency Detection via Constrained Clustering and

Convex Relaxation [8] model proposes a clustering-rooted approach to co-saliency detection. It first clusters the image features into different groups and then applies a convex relaxation algorithm for refining co-saliency maps. Moreover, the co-saliency detection based on the multi-view deep feature fusion model uses a multi-view DFF (Deep Feature Fusion) strategy for capturing all inter-images co-saliency information [12].

2. Literature Review

The model leverages both inter as well as intra-image information to improve the accuracy of co-saliency detection technique. B. Jiang et al. in [13], explored co-saliency detection via an unsupervised multi-modal manifold ranking model which proposes an unsupervised multi-modal manifold ranking approach to co-saliency detection. The model combines the manifold ranking algorithm with multi-modal image representation to identify co-salient areas with a group of related pictures. These are just a few examples of the many models that were explored for refinement and co-saliency detection. Each model has advantages and disadvantages, and the best one to utilize will depend on the kind of application and datasets having been used.

Co-saliency models typically use a combination of higher-level as well as lower-level features to recognize common salient areas across a set of related pictures. Here are some common features used by co-saliency models. Color is one of the most commonly used low-level features for co-saliency detection [14]. Models use color histograms or color moments to identify regions with similar color properties across multiple images. Texture features [15] such as Gabor filters, Local Binary Patterns (LBP) [16], along with Scale-Invariant Featured-Transform (SIFT) [17] are often used to capture the texture information of images. Co-saliency models use these features to identify regions with similar texture properties across multiple images. Shape features such as contour information [18], edge detection [19], and shape context are used to identify the common shape information across multiple images. Some co-saliency models use semantic information such as object detection, scene understanding, and image categorization to determine the communal salient regions across multifarious pictures. Deep features along with rise of deep learning, many co-saliency frameworks utilizes deep features [20] extracted from pre-trained convolutional neural networks (CNNs) [21] namely VGG [22], ResNet [23], or Inception

[24]. These features are learned in a data-driven manner and are capable of capturing both lower-level and higher-level information from images. The choice of features used by co-saliency frameworks depends upon specific application and the type of database containing the images data being used. In general, models that use a combination of features are more robust and perform better in identifying common salient regions across multiple images.

A modified ResNet 50 model is suggested in [25], by K. Karthik et al. for detecting co-salient objects. The CLAHE method has been utilized to improve the key areas of the images, which helps the machine comprehend the prominent features more effectively. However, this model has some limits in terms of implementation embranglement, much time taking, etc. The Otsu segmentation method explored by S. D. M. Mehdizadeh in [26], better partitions the foreground and background portion among existing data samples and thus allows the network to focus on the prominent object. Nevertheless, this method consumes much time in the segmentation procedure along with lower accuracy on small datasets. In [27], S. M. Pizer et al. utilized a modified network along with an SGDM optimizer for training which helps to achieve minimal loss and significantly boost the model's performance accuracy. Nevertheless, the proposed modified network has some limits such as intricate to implement as well as instance segmentation of diversified images remains challenging. In [28], T. Li et al. explored a framework for instance co-segmentation (ICOS) and co-saliency image detection (CSID) based on the attention graph clustering method. The CSID is to discover salient objects as well as concurrent patterns through the cluster of relevant pictures. While, the ICOS is aimed to segment and determine all ICOS, as well as originate relevant masks for all the instances. For the simultaneous, handling of multiple tasks, in this research, a new adaptive graph-based convolutional network has been built. This proposed framework has been trained on multiple datasets. Nevertheless, this framework offers less accuracy and a large time in image segmentation.

This research study has been structured in five sections. The first section explores introduction and background details. Related work has been explored in section two. Further, the material and methods are explained in section three, including the pre-processing techniques applied to the dataset and the modifications made to the ResNet model. Results and discussion is presented in section four and a

comparison analysis of this proposed model with previous techniques has been given from a standard dataset, both qualitatively and quantitatively. Finally, the last fifth section concludes the proposed work.

3. Materials and Method

The suggested technique is intended to find co-saliency in RGB photos. It utilizes a single saliency detection algorithm to analyse the complexity characteristics of all images in a group and identify matching associations between different image groups. The images are pre-processed using the CLAHE technique to improve contrast without amplifying noise, followed by segmentation using the Otsu algorithm to separate pixels into foreground and background classes. The model's hyper-parameters are altered to optimize precision with minimal loss, and the following levels' parameters have been chosen arbitrarily. During testing, the trained network receives the segmented image and produces a co-saliency map, which is equated with ground truth value to calculate performance parameters. The output parameters of the entire layer of connection are preserved while the co-saliency technique attempts to retrieve prominent and recurrent objects while lowering the network's parameter count. Figure 1 provides a conceptual overview of this proposed model. In this work, the K-Fold cross-validation method is chosen for proposed model performance estimation. In this approach, the real samples taken from the datasets have been split arbitrarily into K equal-size subsamples. Furthermore, a lonesome subsample was retained as validation data out of K-subsamples for model testing and the rest of the K-1 subsample has been utilized as training datasets. This K-Fold cross-validation method has been applied for overfitting elimination and to attain highly accurate outcomes on distinct datasets.

3.1. Datasets details:

In this work, there have been selected two separate databases namely common objects in Context (COCO), and CoSal2015 for training as well as model testing. The COCO dataset is one of the larger-scale datasets comprised of 80 object kinds of images along

with an overall 2.50 million labeled examples with 330k pictures. This dataset is aimed to carry out advanced investigations for object identification, image captioning, and segmentation in real-time. Another dataset used in this work is the CoSal2015 dataset. It is also a larger-scale database that includes 50 clusters of the 2015 diversified images. This is commonly employed to access co-salient object detection (CoSOD) algorithms. In earlier research, performance fluctuation has been measured at approximately 1.0 points, therefore considered one of the pragmatic datasets.

3.2. System Configuration

This research is performed on a computer system that is installed with the Intel Xeon E222 4G GHz processor and a graphics processing unit (GPU) having dual NVIDIA™ Quadro RTX4000. Further, this system has installed with a random access memory (RAM) of 16 GB, Windows 11, along with 64-bit operating systems. All the programming task has been carried out using the widely recognized software MATLAB.

3.3. CLAHE method

CLAHE is an image processing method used to enhance contrast of an inputted image. It is a variant of the histogram equalization method, which redistributes the pixel values in an image to make the image's histogram more uniformly distributed. The goal of CLAHE is to increase the local contrast in an image while limiting the overall contrast enhancement to avoid over-amplification of noise and other artifacts. To avoid over-enhancement of contrast, a contrast-limiting operation is applied to each tile's histogram. This ensures that the contrast of each tile is not amplified beyond a certain threshold, which is determined by a parameter called the clip limit. CLAHE is widely used in medical image processing, where it can help to enhance the visibility of structures and details in images with low contrast. It is also used in computer vision applications namely object detection as well as recognition, where improving contrast of an image can make it easier to distinguish objects from their background.

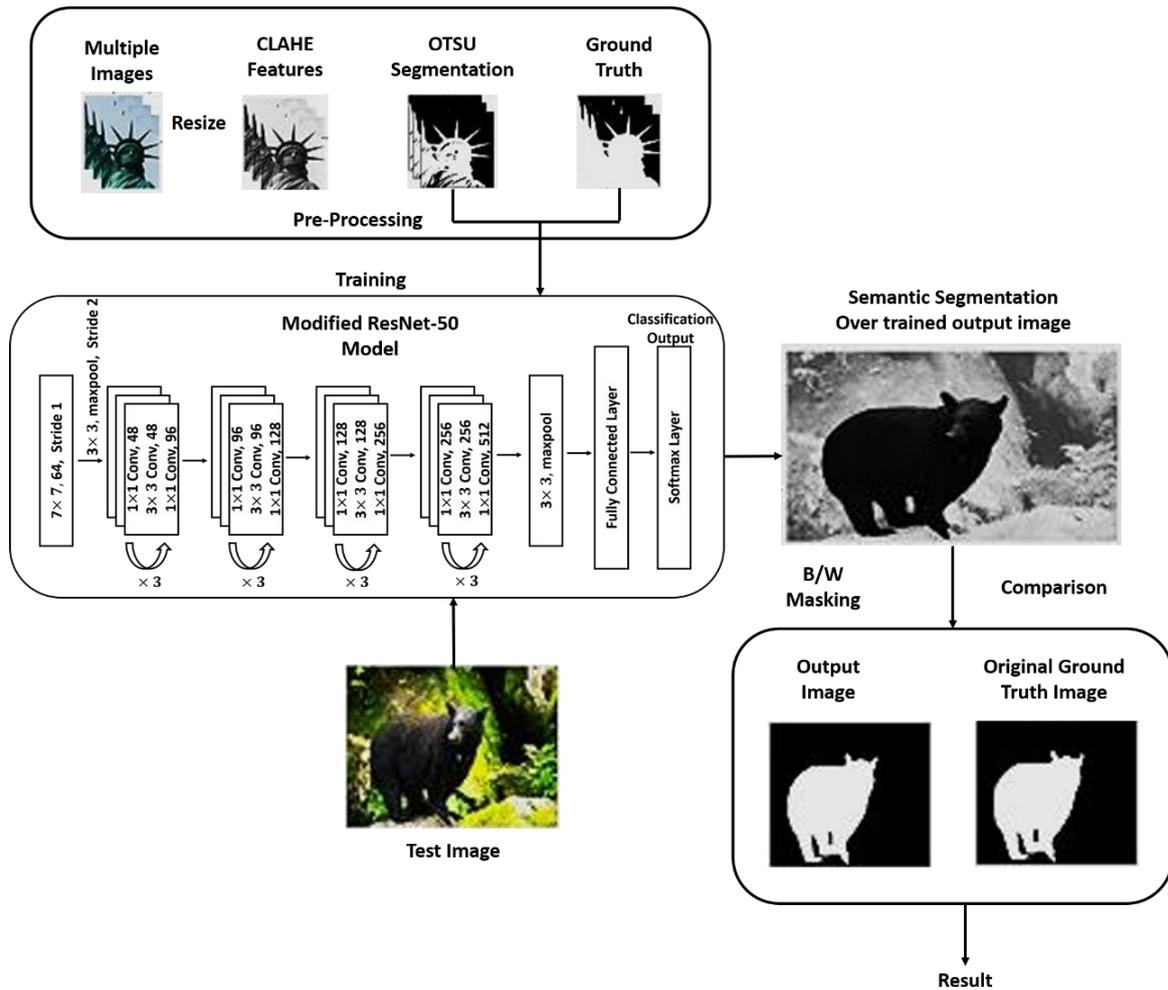


Fig 1: A conceptual representation of the suggested model.

The Pseudo-code for CLAHE can be described in the following steps:

1. To split the image in non-overlapping tiles having size $M \times N$ pixels.
2. For each tile, calculate the histogram of pixel intensities. Let $H(i)$ be the histogram for tile i , where $i=1,2,\dots,T$ (T is the overall tiles within the image).
3. To compute cumulative distribution function (CDF) for each histogram $H(i)$. Let $C(i, k)$ be the CDF for tile i at intensity k , where $k=0,1,2,\dots,L-1$ (wherein, the L is maximal intensity value).
4. To calculate mapping function for each tile as:

$$f(i, k) = \min \{ L-1, \max \{ 0, (C(i, k) \times (L-1)) / S(i) - b \} \}$$

where $S(i)$ = sum of pixel counts in tile i , and b is the contrast limit factor.

5. Apply the mapping function to each pixel in the corresponding tile:

$$g(r, c) = f(i, f(r, c))$$

where (r, c) = pixel's position within the image

$f(r, c)$ is the corresponding intensity value in tile i .

6. Repeat steps 2-5 for all tiles in the image.
7. Combine the processed tiles to form the final CLAHE-enhanced image.

The CLAHE algorithm can be further optimized by using overlapping tiles or adaptive tile sizes that vary depending on the image's local properties.

3.4. Otsu Method:

The Otsu method is a popular image thresholding technique that automatically determines the optimal threshold value for separating a picture in two separate pairs of categories, named foreground as well as background classes. In this Otsu method, it is assumed that the picture contains two categories of pixels, and the pixel intensities in each class follow a Gaussian distribution.

The Pseudo-Code for the Otsu method is described using the following steps:

1. Calculate the histogram of an image to obtain the number of pixels with each intensity level.
2. Normalize the histogram so that it represents a probability density function (PDF):
 - a. $P(i) = n(i) / N$

Wherein $P(i)$ is the probability of intensity level i , $n(i)$ denotes to no. of the pixel having intensity i , and N denotes the complete no. of pixels.

3. Compute CDF of the histogram:
 - a. $W(k) = \sum_{i=0}^k P(i)$

Where $W(k)$ is the cumulative sum of the probabilities up to intensity level k .

4. Compute the mean intensity values of the two classes (foreground and background) for all possible threshold values t :
 - i. $\mu_0(t) = \sum_{i=0}^{t-1} i \times P(i) / W(t)$
 - ii. $\mu_1(t) = \sum_{i=t}^{L-1} i \times P(i) / (1 - W(t))$

Where $\mu_0(t)$ and $\mu_1(t)$ are the background and

foreground class mean intensity values, respectively, for threshold t , and L is the maximum intensity value.

5. Compute the variance between the two classes for all possible threshold values t :
 - i. $\sigma^2_b(t) = W(t) \times (1 - W(t)) \times (\mu_1(t) - \mu_0(t))^2$

Where $\sigma^2_b(t)$ is the between-class variance for threshold t .

6. Select the threshold value which maximizes variance in between distinct class:
 - i. $t_{opt} = \text{argmax}_{t=0,1,\dots,L-1} \{\sigma^2_b(t)\}$

Where t_{opt} is the optimal threshold value.

7. To create a binary picture, apply a threshold to the image.
 - i. $g(x,y) = 0$ if $f(x,y) < t_{opt}$
 - ii. $= 1$ if $f(x,y) \geq t_{opt}$

Where $f(x,y)$ denotes pixel's intensity value at position (x,y) in real picture, as well as $g(x,y)$ is a corresponding binary value after thresholding.

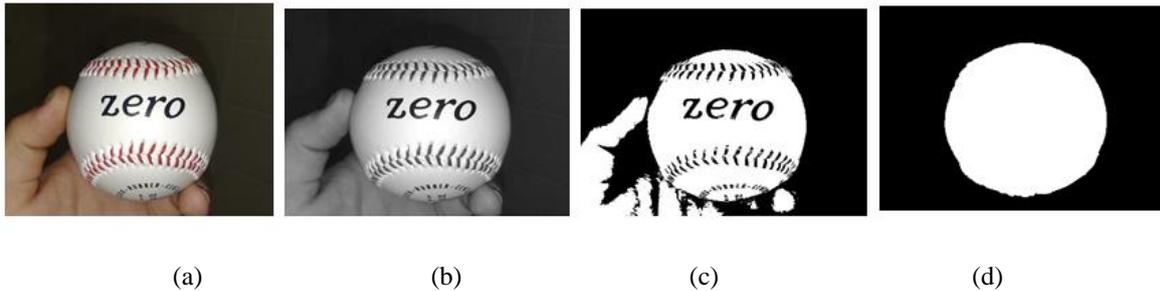


Fig 2: (a) Shows the original image (b) depicts image processed after CLAHE method (c) shows the image processed after Otsu segmentation and (d) illustrates the ground truth of original image respectively.

Figure 2 (a) Shows the original image (b) depicts image processed after CLAHE method (c) shows the image processed after Otsu segmentation and (d) illustrates the ground truth of original image respectively. The proposed network, called Thin-ResNet, has less no. of parameters than the original ResNet-50 model, as illustrated in Figure 3. ResNet-50 achieved impressive results in the ILSVRC 2015 classification competition, with an error rate of only 3.57%, prompting machine learning experts to recommend adding extra layers to deep CNN network to enhance the performance. Deep residual nets incorporate residual blocks to enhance accuracy, with skip connections being particularly valuable in reducing the problem of vanishing gradient problem

helping to reduce the error ratio. Further, this research work was assessed against many deep neural network exhibiting superior performance. The models use batch normalization with ReLUs before activation, with modified convolutional filters as well as their modified dimensions have been specified in every row. The layers are adjusted to suit the input image and classification objectives, with hyper-parameters tweaked to achieve maximum accuracy and minimum loss during training using the SGDM optimizer. The trained network generates an output feature map of the test image during testing, with CLAHE features extracted from the binary output image for semantic segmentation.

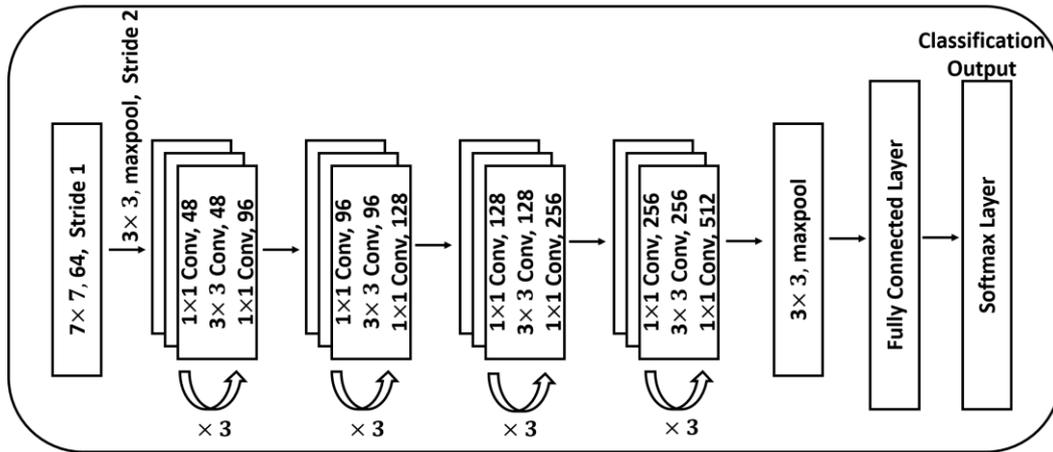


Fig 3: Architecture of Thin ResNet model.

4. Results and Discussion

The assessment outcomes of this proposed work are done on two popular co-saliency benchmark datasets. The common objects in context (COCO) image dataset [27] is one of the larger-scale object detection, captioning, and segmentation dataset. This dataset contains more than 330,000 distinct images with over 2.50 million different object instances labelled and annotated with over 80 object categories, while CoSal2015 [29] is a benchmark dataset for co-saliency detection, which is task of identifying communal salient objects within a set of pictures. The dataset contains 2,015 images from 50 different object categories, such as animals, plants, and vehicles. This proposed model has been trained as well as tested on a 70:30 ratio dataset with each image resized to 224 by 224 size. The F-measure (F1 Score) was utilized for evaluating the model performance. We used the Mean Absolute Error (MAE) to govern how closely all predicted co-saliency maps match along with actual ground truth.

Precision, recall, along with F-measure are few common performance evaluation metrics utilized in the information retrieval, classification, and machine learning to assess the performance measures of a model or system. There have been utilized three matrices for performance evaluation.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

In the aforementioned equations, TP defines the true positive and FN defines the false negative, as well as

FP, denotes the false positive.

4.1. Implementation Details

The goal of this study is to use a co-saliency detection model for determining similar objects with multiple groups of images. To improve contrast, the CLAHE method is applied in pre-processing. Otsu segmentation is used to distinguish the background and foreground of the image, resulting in separating one threshold level and classifying pixels into foreground and background classes. The modified ResNet50 model is trained using highly contrasted features and ground truth features, and fine-tuning is done to avoid overfitting problems. We have successfully fine-tuned our model and it now has better validation accuracy. The implementation was done using MATLAB with Intel Xeon E222 4G GHz processor and a GPU having dual NVIDIA™ Quadro RTX4000. To optimize the results, we compiled the network using the SGDM optimizer to decrease loss function effectively. Further, this model was trained for 30 epochs using an initial learning rate 0.001 as well as batch size was 20. During training, the accuracy and loss were evaluated after every epoch on the test dataset in real time. Based on loss of testing dataset, hyper-parameters were modified. The network needs between 382 and 512 milliseconds to finish 30 epochs.

4.2. Experimental Result analysis

The performance of this suggested model has been compared and assessed along with distinct state-of-the-art methods mentioned in Table 1. Further, the comparison has been made in terms of visual results and is presented in Figure 4 and Figure 5. These Figures demonstrate that the proposed model is effective in handling various challenging scenarios

such as complex and cluttered backdrops, brightness changes, small-size entities, extreme occlusions, etc. The proposed model has produced better visual results than the previous SOTA models, indicating that the changes made to the existing ResNet model were successful. The co-saliency maps originated through the GoNet model and ICNet are noisy and fail to detect

prominent objects such as berry, pineapples, fish, and others as shown in Figure 4 and Figure 5. In comparison to the previous approaches, the proposed network proposes extra accurate limit facts, which may result in co-saliency maps along with more detailed border features.

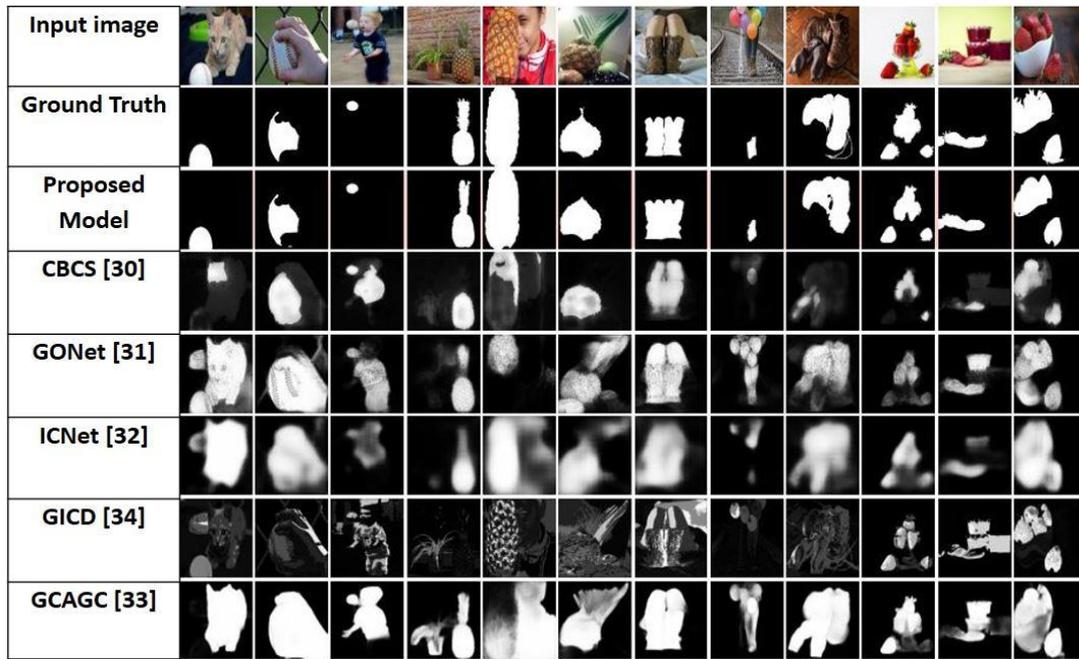


Fig 4: Illustrates the qualitative analysis of SOTA model on COCO data set [30]–[34].

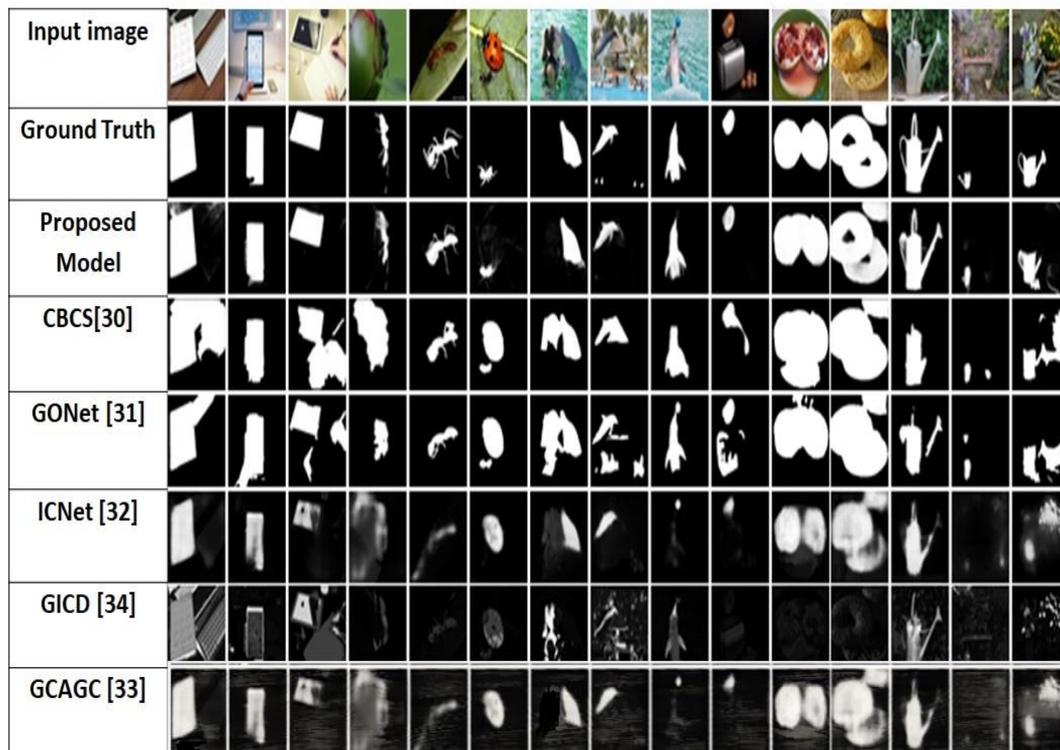


Fig 5: Illustrates the qualitative analysis of SOTA model on the CoSal2015 data set [30]–[34].

Table 1 provides a breakdown of the F1 measure and MAE statistical outcomes on the COCO and Cosal2015 benchmark datasets in comparison to different previous models. Our suggested model is demonstrated to perform substantially better than other methods, including GCAGC, and CBCS, which are the most up-to-date SOTA methods. The proposed model surpasses all other techniques in every evaluation metric for both the Cosal2015 and COCO datasets and obtains high scores from all reviews.

Table 1 shows that the measured MAE score and F Measure score for distinct models i.e., CBCS, GCAG, ICNet, GICD, GoNet, and ResNet50 on CoSal2015 dataset are 0.231, 0.168, 0.160, 0.069, 0.341, 0.052 as well as 0.726, 0.774, 0.641, 0.749, 0.632, 0.973, respectively. However, our proposed model obtains the MAE score 0.034 and F Measure 0.986 respectively, which is improved when compared with the existing research work. Furthermore, the measured MAE score and F Measure score for distinct models i.e., CBCS, GCAG, ICNet, GICD, GoNet, and ResNet50 on COCO dataset are 0.314, 0.260, 0.396, 0.189, 0.315, 0.214 as well as 0.863, 0.875, 0.662, 0.806, 0.730, 0.882, respectively. However, our proposed model obtains the MAE score 0.164 and F Measure 0.942 respectively, which is enhanced while comparing with previous models. This suggested model performs exceptionally well with an F1-score of 98.6% and an MAE of 3.4% on distinct benchmark databases, outperforming other existing models, such as GoNet [30], with an F1-score of 87.4%. The proposed model also outperforms with existing ResNet-50 model with a measured F1-score of 97.3%.

Table 1: Quantitative comparison with CoSal2015 and COCO dataset.

Model Name	CoSal2015		COCO	
	F Measure Score	MAE Score	F Measure Score	MAE Score
CBCS	0.726	0.231	0.863	0.314
GCAG	0.774	0.168	0.875	0.260
ICNet	0.641	0.160	0.662	0.396
GICD	0.749	0.069	0.806	0.189
GoNet	0.632	0.341	0.730	0.315
ResNet50	0.973	0.052	0.882	0.214
Ours	0.986	0.034	0.942	0.164

The performance graph in Figure 6 clearly shows the superiority of the proposed network over other standard models with respect to F1-measure and MAE

values. The proposed model gives excellent outcomes on a voluminous dataset that contains complex images. To retrieve structural characteristics from image groups, the network utilizes CLAHE features, as co-saliency significantly relies on categorization.

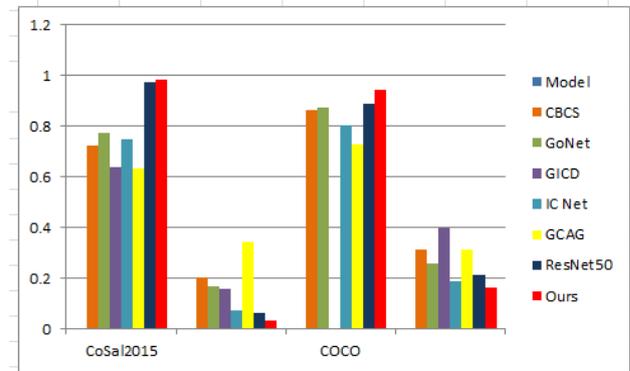


Fig 6: Performance chart with respect to MAE as well as F1-score on COCO and Cosal2015 dataset.

5. Conclusion

In this research, there has been introduced a novel deep learning-rooted method for the co-saliency detection problem, achieved by modifying the existing ResNet network. Our model captures concept-level attributes of co-salient entities with suppressing all other irrelevant backgrounds by using an optimized ResNet-50 framework for co-segmentation mask generation. CLAHE is used for preprocessing, followed by the Otsu segmentation method. The model is first pre-processed with CLAHE for enhancing contrast of entities inside the picture as well as the network is trained along with their corresponding ground truth images using seventy percent of both the benchmarked dataset and the rest thirty percent is used for testing. The proposed method achieves a high F1-score of 98.6% along with an MAE of 0.034 on the Cosal2015 dataset, and an F1-score and MAE score of 94.2%, 0.164 on the COCO dataset, outperforming other existing methods. By pre-processing our model, the network better understands complex features and gives great results on voluminous data. In the future, more investigation may be carried out for the exploration of the usage of Bayesian formulations along with the group semantic information to escalate all performance measures.

References

- [1] Q. Hou, M. M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. S. Torr, "Deeply Supervised Salient Object Detection with Short Connections," *IEEE Trans.*

- Pattern Anal. Mach. Intell.*, 2019, doi: 10.1109/TPAMI.2018.2815688.
- [2] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, 2010, doi: 10.1109/TIP.2009.2030969.
- [3] J. Guo, T. Ren, L. Huang, X. Liu, M. M. Cheng, and G. Wu, "Video salient object detection via cross-frame cellular automata," 2017. doi: 10.1109/ICME.2017.8019389.
- [4] Y. Su, J. Deng, R. Sun, G. Lin, and Q. Wu, "A Unified Transformer Framework for Group-based Segmentation: Co-Segmentation, Co-Saliency Detection and Video Salient Object Detection," vol. 14, no. 8, pp. 1–18, 2022, doi: 10.1109/TMM.2023.3264883.
- [5] X. Yang, S. Li, J. Ma, J. Yang, and J. Yan, "Co-saliency-regularized correlation filter for object tracking," *Signal Process. Image Commun.*, vol. 103, p. 116655, 2022, doi: <https://doi.org/10.1016/j.image.2022.116655>.
- [6] Y. Zhao, H. Wang, Y., & Li, "Saliency Constrained Arbitrary Image Style Transfer using SIFT and DCNN," 2022.
- [7] X. Qian, X. Cheng, G. Cheng, X. Yao, and L. Jiang, "Two-Stream Encoder GAN with Progressive Training for Co-Saliency Detection," *IEEE Signal Process. Lett.*, 2021, doi: 10.1109/LSP.2021.3049997.
- [8] K. Zhang, M. Dong, B. Liu, X. T. Yuan, and Q. Liu, "DeepACG: Co-Saliency Detection via Semantic-aware Contrast Gromov-Wasserstein Distance," 2021. doi: 10.1109/CVPR46437.2021.01349.
- [9] L. Kong, P. Ganesh, T. Wang, J. Liu, Y. Chen, and L. Zhang, "Free Lunch for Co-Saliency Detection: Context Adjustment." 2021.
- [10] X. Qian, Y. Zeng, W. Wang, and Q. Zhang, "Co-saliency Detection Guided by Group Weakly Supervised Learning," *IEEE Trans. Multimed.*, 2022, doi: 10.1109/TMM.2022.3167805.
- [11] X. Li, L. Zhao, L. Wei, M. H. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang, "DeepSaliency: Multi-Task Deep Neural Network Model for Salient Object Detection," *IEEE Trans. Image Process.*, 2016, doi: 10.1109/TIP.2016.2579306.
- [12] R. Hu, Z. Deng, and X. Zhu, "Multi-scale Graph Fusion for Co-saliency Detection," 2021. doi: 10.1609/aaai.v35i9.16951.
- [13] B. Jiang, X. Jiang, J. Tang, B. Luo, and S. Huang, "Multiple graph convolutional networks for co-saliency detection," 2019. doi: 10.1109/ICME.2019.00065.
- [14] Y. Liu, Y. Gao, and W. Yin, "An improved analysis of stochastic gradient descent with momentum," 2020.
- [15] Z. Zhang, W. Jin, J. Xu, and M. M. Cheng, "Gradient-Induced Co-Saliency Detection," 2020. doi: 10.1007/978-3-030-58610-2_27.
- [16] K. Zhang, T. Li, S. Shen, B. Liu, J. Chen, and Q. Liu, "Adaptive graph convolutional network with attention graph clustering for co-saliency detection," 2020. doi: 10.1109/CVPR42600.2020.00907.
- [17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.
- [18] Z. Liu, W. Zou, L. Li, L. Shen, and O. Le Meur, "Co-saliency detection based on hierarchical segmentation," *IEEE Signal Process. Lett.*, 2014, doi: 10.1109/LSP.2013.2292873.
- [19] X. Cao, Z. Tao, B. Zhang, H. Fu, and W. Feng, "Self-adaptively weighted co-saliency detection via rank constraint," *IEEE Trans. Image Process.*, 2014, doi: 10.1109/TIP.2014.2332399.
- [20] D. J. Jeong, I. Hwang, and N. I. Cho, "Co-Salient Object Detection Based on Deep Saliency Networks and Seed Propagation over an Integrated Graph," *IEEE Trans. Image Process.*, 2018, doi: 10.1109/TIP.2018.2859752.
- [21] L. Wei, S. Zhao, O. El Farouk Bourahla, X. Li, and F. Wu, "Group-wise deep co-saliency detection," 2017. doi: 10.24963/ijcai.2017/424.
- [22] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Salient Object Detection with Recurrent Fully Convolutional Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019, doi: 10.1109/TPAMI.2018.2846598.
- [23] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect Globally, Refine Locally: A Novel Approach to Saliency Detection," 2018. doi: 10.1109/CVPR.2018.00330.
- [24] K. Zhang, J. Chen, B. Liu, and Q. Liu, "Deep object co-segmentation via spatial-semantic network modulation," 2020. doi: 10.1609/aaai.v34i07.6977.
- [25] K. Karthik and S. S. Kamath, "A deep neural network model for content-based medical image retrieval with multi-view classification," *Vis. Comput.*, 2021, doi: 10.1007/s00371-020-01941-2.
- [26] S. D. M. Mehdizadeh, "Study of Effect of

Adaptive Histogram Equalization on Image Quality in Digital Preapical Image in Pre Apex Area,” *Journal of Biological Sciences*, vol. 4, no. 8. pp. 922–924, 2009.

- [27] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld, “ADAPTIVE HISTOGRAM EQUALIZATION AND ITS VARIATIONS.,” *Comput. vision, Graph. image Process.*, 1987, doi: 10.1016/S0734-189X(87)80186-X.
- [28] T. Li, K. Zhang, S. Shen, B. Liu, Q. Liu, and Z. Li, “Image Co-Saliency Detection and Instance Co-Segmentation Using Attention Graph Clustering Based Graph Convolutional Network,” *IEEE Trans. Multimed.*, 2022, doi: 10.1109/TMM.2021.3054526.
- [29] D. Zhang, J. Han, C. Li, J. Wang, and X. Li, “Detection of Co-salient Objects by Looking Deep and Wide,” *Int. J. Comput. Vis.*, 2016, doi: 10.1007/s11263-016-0907-4.
- [30] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” 2005. doi: 10.1109/CVPR.2005.177.
- [31] J. Winn, A. Criminisi, and T. Minka, “Object categorization by learned universal visual dictionary,” 2005. doi: 10.1109/ICCV.2005.171.
- [32] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2016. doi: 10.1109/CVPR.2016.90.
- [33] K. J. Hsu, C. C. Tsai, Y. Y. Lin, X. Qian, and Y. Y. Chuang, “Unsupervised CNN-Based Co-saliency Detection with Graphical Optimization,” 2018. doi: 10.1007/978-3-030-01228-1_30.
- [34] W. Da Jin, J. Xu, M. M. Cheng, Y. Zhang, and W. Guo, “ICNet: Intra-saliency correlation network for co-saliency detection,” 2020.
- [35] Chinthamu, N. ., Gooda, S. K. ., Shenbagavalli, P. ., Krishnamoorthy, N. ., & Selvan, S. T. . (2023). Detecting the Anti-Social Activity on Twitter using EGBDT with BCM. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(4s), 109–115. <https://doi.org/10.17762/ijritcc.v11i4s.6313>
- [36] Tanaka, A., Min-ji, K., Silva, C., Cohen, D., & Mwangi, J. Predictive Analytics for Healthcare Resource Allocation. *Kuwait Journal of Machine Learning*, 1(4). Retrieved from <http://kuwaitjournals.com/index.php/kjml/article/view/150>
- [37] Chaudhury, S., Dhabliya, D., Madan, S., Chakrabarti, S. Blockchain technology: A global provider of digital technology and services (2023) *Building Secure Business Models Through Blockchain Technology: Tactics, Methods, Limitations, and Performance*, pp. 168-193.