CO-SALIENCY DETECTION USING COLLABORATIVE FEATURE EXTRACTION AND HIGH-TO-LOW FEATURE INTEGRATION

Jingru Ren[†‡], Zhi Liu[†‡], Gongyang Li[†‡], Xiaofei Zhou[§], Cong Bai [*], Guangling Sun[‡]*

[†]Shanghai Institute for Advanced Communication and Data Science, Shanghai University, China
[‡]School of Communication and Information Engineering, Shanghai University, China
[§]Institute of Information and Control, Hangzhou Dianzi University, China
[*] College of Computer Science, Zhejiang University of Technology, China
renjingru716@shu.edu.cn, liuzhi@staff.shu.edu.cn, ligongyang@shu.edu.cn, zxforchid@hdu.edu.cn, congbai@zjut.edu.cn, sunguangling@shu.edu.cn

ABSTRACT

Co-saliency detection, as a developing research branch of saliency detection, devotes to identify the common salient objects in a group of related images. The major challenge of co-saliency detection is how to effectively represent features considering both intra-image and inter-image information. In this paper, we propose a co-saliency detection model using collaborative feature extraction and high-to-low feature integration. We first feed the target image and its co-images into the Individual Feature Extraction Module (IFEM) to produce multi-level individual features. Then, to capture the collaborative inter-image information, the Collaborative Feature Extraction Module (CFEM) is applied to all highest-level individual features, generating the collaborative feature. Finally, we build a High-to-low Feature Integration Module (HFIM), which integrates the collaborative feature and multi-level individual features of the target image, to enrich the collaborative feature with individual intra-image information. Extensive experiments on two public datasets demonstrate that the proposed model achieves the state-of-the-art performance.

Index Terms— Co-saliency detection, collaborative feature extraction, high-to-low feature integration

1. INTRODUCTION

Saliency detection is defined as the task of discovering the most attractive objects in a scene automatically and serves as a pre-processing step for many computer vision tasks, such as salient object segmentation [1, 2], image manipulation [3], object-based image retrieval [4], weakly supervised semantic segmentation [5, 6] and so on. In recent years, with the popularity of the Internet and smartphones, we can easily collect a large number of images sharing similar objects. Co-saliency detection, which aims at highlighting the common and salient objects in a group of related images, has received extensive attention. It benefits lots of visual tasks including object co-segmentation [7, 8], co-localization [9], and weakly supervised localization [10].

Inspired by human visual attention mechanism, many early co-saliency detection models leverage hand-crafted features and explore heuristic prior knowledge to predict co-saliency maps. For example, in [11], three visual attention cues, *i.e.* contrast cues, spatial cue, and corresponding cue, are devised to measure the cluster-level co-saliency. Cao et al. [12] formalized the consistency property among salient regions as the rank constraint, which is employed to calculate the fused weights of existing saliency maps self-adaptively. In [13], a region-level fusion method exploits the similarities between regions, and a pixel-level refinement method integrates color-spatial similarity with border connectivity prior. These models have achieved satisfying results in some scenarios. However, they hardly make use of high-level semantic features which are crucial to robustly mine the collaborative information in complex scenes.

Recently, learning-based models become the mainstream research direction for co-saliency detection. These models utilize various machine learning techniques to learn co-saliency patterns from the image groups. For instance, Zhang et al. [14] designed a multiple-instance learning model, which introduces the self-paced learning theory for selecting training samples and gradually learns the co-saliency cues from confident image regions to ambiguous ones. In [15], the proposed model jointly learns discriminative feature representation and co-salient object detector by optimizing an objective function that embeds a metric learning regularization term into support vector machine training. Besides, convolutional neural network (CNN) can directly process two-dimensional images and keep the spatial position of images all the time. The model proposed in [16] is a typical CNN based co-saliency model, which builds a united learning scheme for exploring the

[*]Zhi Liu is the corresponding author. This work was supported by the National Natural Science Foundation of China under Grants 61771301, 61901145 and U1908210.

978-1-7281-1331-9/20/$31.00 © 2020 IEEE

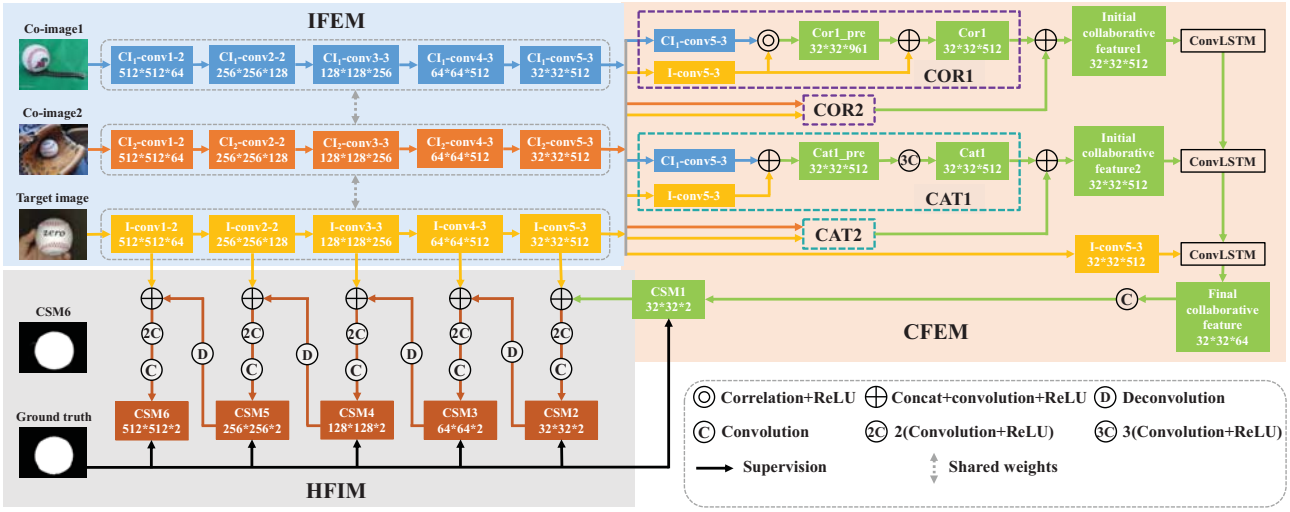


Fig. 1. Flowchart of the proposed co-saliency detection model. Our model consists of three components: the Individual Feature Extraction Module (IFEM), the Collaborative Feature Extraction Module (CFEM), and the High-to-low Feature Integration Module (HFIM).

intrinsic correlations between saliency detection in individual image and the image group. In [17], multi-layer convolutional features are fused through four stages to output co-saliency maps. Despite the great progress made by these learning-based models, most of them have not fully mined both intra-image and inter-image information.

To solve the problem of capturing collaborative inter-image information, some relevant studies have tried some beneficial exploration. In [18], the deep object co-segmentation model exploits a mutual correlation layer to detect common objects in a pair of images. The correlation layer is first introduced in Flownet [19] for estimating optical flow. It computes the correlation between each pair of patches from two features and outputs a feature map which highlights the common objects. The convolutional LSTM module (ConvLSTM) [20] substitutes dot products with convolutional operations in traditional fully connected LSTM [21] to fit spatial features. It can progressively discard the irrelevant information and fuse recurrent important information of image sequences through updating the memory cell. The ConvLSTM is often utilized in video processing tasks [22, 23] for learning relevant information between video frames.

Motivated by the correlation layer and ConvLSTM, in this paper, we propose a CNN based co-saliency detection model with two key modules. In the Collaborative Feature Extraction Module (CFEM), the correlation layer based feature extraction strategy (COR) and the concatenation layer based feature extraction strategy (CAT) are applied to all highest-level individual features for capturing the relevant information between images. The resultant two initial collaborative features are further combined with the highest-level individual feature of target image by three consecutive ConvLSTMs. In the High-to-low Feature Integration Module (HFIM), to balance the individual intra-image information, the collaborative feature is gradually integrated with the multi-level individual features of target image from high level to low level. Due to the two key modules, the proposed model effectively extracts features containing both inter-image and intra-image information.

Our main contributions are summarized as follows:

1) We propose a co-saliency detection model using collaborative feature extraction and high-to-low feature integration, which can effectively generate collaborative feature via co-images and integrate with the multi-level individual features of target image, for co-saliency detection.

2) We design a Collaborative Feature Extraction Module (CFEM), which exploits the feature extraction strategies of correlation and concatenation as well as consecutive ConvLSTMs, to obtain the collaborative feature.

3) We build a High-to-low Feature Integration Module (HFIM) to enrich the collaborative feature with the intra-image information. The HFIM integrates the collaborative feature with multi-level individual features of the target image in a high-to-low manner.

2. PROPOSED MODEL

The flowchart of the proposed co-saliency detection model is illustrated in Fig. 1. Given the target image $\mathbf{I}$, multiple co-images $\{\mathbf{CI}_1, \mathbf{CI}_2, ...\}$ can be randomly selected from the same category. Considering the issue of computational efficiency and memory usage, the number of co-images is set to 2 in our model. So, the target image and its two co-images are combined into a Co-saliency Detection Group (CDG), as the input to our model. The output of our model

is the co-saliency map $\mathbf{CSM}_6$. In the following subsections, we will elaborate our co-saliency detection model. Sec. 2.1 briefly introduces how to extract individual features, Sec. 2.2 presents the generation process of collaborative feature, and Sec. 2.3 gives a detailed description of integrating multi-level individual features and collaborative feature abided by a high-to-low manner.

2.1. Individual feature extraction module

To obtain abundant individual intra-image features, we build the Individual Feature Extraction Module (IFEM) on the basis of VGG16 network [24], as shown in the top left of Fig. 1. The VGG16 network is primitively designed for image classification and has shown a great success in extracting features for many computer vision tasks [25, 26]. Specifically, we discard the last max-pooling layer and three fully connected layers of the original VGG16 network, to fit the pixel-wise co-saliency detection task. The modified VGG16 network can provide individual features at five convolutional layers with different scales, *i.e.* conv1_2, conv2_2, conv3_3, conv4_3, and conv5_3. Corresponding to the three input images in CDG, the IFEM is formed by three parallel modified VGG16 networks, which share the weights with each other for ensuring the undifferentiated feature extraction. The extracted five-level individual features of all images in CDG are denoted as $\{\mathbf{IF}_j^i, i=0,1,2; j=1,2,3,4,5\}$, with the superscript $i=0$ denoting the target image, $i=1,2$ denoting the two co-images, and the subscript j denoting the individual feature generated by the j^{th} convolutional layer.

2.2. Collaborative feature extraction module

In general, the high-level convolutional features have more sematic information than low-level convolutional features, and the semantic features are more suitable for extracting the relevant information between images. In this subsection, we focus on the highest-level individual features of the target image and its two co-images, and design a Collaborative Feature Extraction Module (CFEM), as shown in the top right of Fig. 1. The CFEM contains two different feature extraction strategies: one strategy based on correlation layer (COR), and another strategy based on concatenation layer (CAT). Both of COR and CAT appear twice in CFEM: COR1 and CAT1 take $\mathbf{IF}_5^0$ and $\mathbf{IF}_5^1$ as the input; COR2 and CAR2 take $\mathbf{IF}_5^0$ and $\mathbf{IF}_5^2$ as the input.

Concretely, in the first feature extraction strategy, the process of COR1 and COR2 is performed by:

$$\mathbf{CF}_{cor1} = \phi(Conv(Cat(\phi(Cor(\mathbf{IF}_5^0, \mathbf{IF}_5^1)), \mathbf{IF}_5^0))), \quad (1)$$

$$\mathbf{CF}_{cor2} = \phi(Conv(Cat(\phi(Cor(\mathbf{IF}_5^0, \mathbf{IF}_5^2)), \mathbf{IF}_5^0))), \quad (2)$$

where $\mathbf{CF}_{cor1}$ and $\mathbf{CF}_{cor2}$ denote the resultant features of COR1 and COR2. $Cor()$ is a correlation layer which performs multiplicative patch comparisons between two features. $\phi()$ denotes the ReLU activation function, and $Cat()$ denotes the concatenation operation along channel direction. Here the intermediate result $\phi(Cor(\mathbf{IF}_5^0, \mathbf{IF}_5^1))$ (or $\phi(Cor(\mathbf{IF}_5^0, \mathbf{IF}_5^2))$) is cascaded with $\mathbf{IF}_5^0$ for replenishing the individual feature of the target image. $Conv()$ is a normal convolutional layer, and it comes after the concatenation layer for further combination. We incorporate the resultant features of COR1 and COR2 as follows:

$$\mathbf{CF}_{initial1} = \phi(Conv(Cat(\mathbf{CF}_{cor1}, \mathbf{CF}_{cor2}))), \quad (3)$$

where $\mathbf{CF}_{initial1}$ represents the initial collaborative feature generated by the first feature extraction strategy.

In the second feature extraction strategy, the process of CAT1 and CAT2 is performed by:

$$\mathbf{CF}_{cat1} = \mathbb{C}^3(\phi(Conv(Cat(\mathbf{IF}_5^0, \mathbf{IF}_5^1)))), \quad (4)$$

$$\mathbf{CF}_{cat2} = \mathbb{C}^3(\phi(Conv(Cat(\mathbf{IF}_5^0, \mathbf{IF}_5^2)))), \quad (5)$$

where $\mathbf{CF}_{cat1}$ and $\mathbf{CF}_{cat2}$ denote the resultant features of CAT1 and CAT2. $\mathbb{C}^3()$ denotes three recurrent operation compositions, and each composition covers a convolutional layer and a ReLU activation function. Similar to the first feature extraction strategy, $\mathbf{CF}_{cat1}$ and $\mathbf{CF}_{cat2}$ are incorporated as follows:

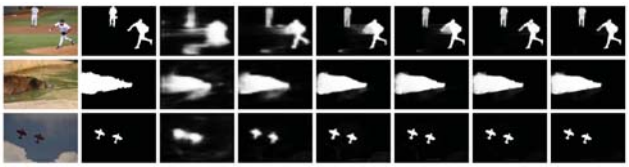
$$\mathbf{CF}_{initial2} = \phi(Conv(Cat(\mathbf{CF}_{cat1}, \mathbf{CF}_{cat2}))), \quad (6)$$

where $\mathbf{CF}_{initial2}$ represents the initial collaborative feature generated by the second feature extraction strategy.

The first feature extraction strategy COR places emphasis on the mutual relation between features at different spatial positions, while the second feature extraction strategy CAT focuses attention on the mutual relation between features at the cross-channel manner. They promote and complement each other. Therefore, the two initial collaborative features, $\mathbf{CF}_{initial1}$ and $\mathbf{CF}_{initial2}$, with the additional $\mathbf{IF}_5^0$, are further combined as follows:

$$\mathbf{CF} = \mathbb{R}(\mathbb{R}(\mathbb{R}(\mathbf{CF}_{initial1}), \mathbf{CF}_{initial2}), \mathbf{IF}_5^0), \quad (7)$$

where $\mathbf{CF}$ denotes the final collaborative feature. $\mathbb{R}()$ represents the ConvLSTM, which works by continually updating the cell memory and hidden state according to three controlling gates, namely input gate, forget gate and output gate. Here the inputs to the three consecutive ConvLSTMs are $\mathbf{CF}_{initial1}$, $\mathbf{CF}_{initial2}$ and $\mathbf{IF}_5^0$ in sequence. The pervious ConvLSTM passes the updated cell memory and hidden state to the next ConvLSTM. The hidden state generated by the third ConvLSTM is the final collaborative feature $\mathbf{CF}$.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 2. Visualization of intermediate and outputted co-saliency maps. (a) Images; (b) ground truths; five intermediate co-saliency maps including (c) $\mathbf{CSM}_1$; (d) $\mathbf{CSM}_2$; (e) $\mathbf{CSM}_3$; (f) $\mathbf{CSM}_4$; (g) $\mathbf{CSM}_5$; outputted co-saliency map (h) $\mathbf{CSM}_6$.

2.3. High-to-low feature integration module

The final collaborative feature $\mathbf{CF}$ includes collaborative information of images, but may lack individual information and boundary details of the target image. Therefore, we build a High-to-low Feature Integration Module (HFIM), as shown in the bottom left of Fig. 1, to supplement the collaborative feature with intra-image information. The HFIM exploits the individual features of target image at five levels, $\{\mathbf{IF}_j^0, j=1,2,3,4,5\}$, to gradually optimize the final collaborative feature from high level to low level. At each level, the current-level individual feature is integrated with the previous integrated feature at the higher level, and the current-level integrated feature will be sequentially integrated with the individual feature at the lower level.

Specifically, before entering into HFIM, the channel of the final collaborative feature $\mathbf{CF}$ is reduced to 2 by a convolutional layer with 1×1 kernel size, producing the intermediate co-saliency map $\mathbf{CSM}_1$. The integration process at five levels is performed by:

$$\begin{cases} \mathbf{CSM}_2 = Conv(\mathbb{C}^2(Cat(\mathbf{IF}_5^0, \mathbf{CSM}_1))) \\ \mathbf{CSM}_k = Conv(\mathbb{C}^2(Cat(\mathbf{IF}_{7-k}^0, Dec(\mathbf{CSM}_{k-1})))) \end{cases}, \quad (8)$$

where $k=3,4,5,6$. $\mathbf{CSM}_2$, $\mathbf{CSM}_3$, $\mathbf{CSM}_4$, $\mathbf{CSM}_5$ and $\mathbf{CSM}_6$ are the integrated features at five levels. $Dec()$ is a deconvolutional layer which is used for 2×upsampling to adapt to the size of lower-level feature. $\mathbb{C}^2()$ denotes two recurrent operation compositions, and each composition covers a convolutional layer and a ReLU activation function. The last operation in Eq. (8) is a convolutional layer with 1×1 kernel size to reduce the number of channels to 2. The first four integrated features can also be regarded as intermediate co-saliency maps, and the finally integrated feature $\mathbf{CSM}_6$ is the outputted co-saliency map.

In Fig. 2, we visualize the five intermediate co-saliency maps and the outputted co-saliency map to verify the effectiveness of our high-to-low feature integration module. It is obvious that the worst prediction result is $\mathbf{CSM}_1$ before feature integration. As the processing of feature integration from high level to low level, *i.e.* from $\mathbf{CSM}_2$ to $\mathbf{CSM}_6$, the prediction accuracy and boundary details are gradually improved. This indicates that the individual features of target image are fully exploited in the HFIM.

In addition, as shown in Fig. 1, we adopt six supervision branches for the five intermediate co-saliency maps and the outputted co-saliency map to enhance the depth of supervision. The proposed network is trained end-to-end using six softmax losses between each co-saliency map and the ground truth (GT) with the corresponding scale. At each branch, the loss function is defined as:

$$L = -\sum_{x,y} l_{x,y} \log(P_{x,y}) + (1-l_{x,y}) \log(1-P_{x,y}), \quad (9)$$

where $l_{x,y} \in \{0,1\}$ is the label of pixel (x,y) in the GT, and $P_{x,y}$ is the normalized probability of pixel (x,y) belong to the co-salient objects according to the co-saliency map. Following [17], the inter-image propagation based refinement is employed to improve the spatial coherence of the outputted co-saliency map.

3. EXPERIMENTAL RESULTS

3.1. Experimental setting

Datasets: We train our model on the Cosal2015 dataset [27], the PASCAL-VOC dataset [28] and the Coseg-Rep dataset [29], in which all images are manually annotated with pixel-wise binary ground truths. We test our model on two public co-saliency detection datasets including the iCoseg dataset [30] and the MSRC dataset [31]. The iCoseg dataset contains 38 categories and has a total of 643 images. The MSRC dataset has a total of 233 images in 7 categories.

Evaluation metrics: We evaluate the performance of the proposed model using precision-recall (PR) curve, F-measure and mean absolute error (MAE). The PR curve is drawn by using the precision value versus the recall value at each integer threshold from 0 to 255. F-measure is defined as the harmonic mean of precision and recall values obtained by using an adaptive thresholding method [32], and the balance factor β is set to 0.3 as suggested in [10]. MAE measures the average difference at pixel level between the co-saliency map and the ground truth.

Implementation details: Our model is implemented on MATLAB R2014a platform with the Caffe toolbox [33]. The weights of IFEM are initialized from the VGG16 network [24]. We use Adam [34] to train our model with learning rate $1e-4$, which is decreased to $1e-5$ after 50k iterations. And the mini-batch size and momentum are set to 16 and 0.9, respectively. The proposed model needs about 80k training iterations for convergence, and takes 0.06s to generate the outputted co-saliency map of an image with a resolution of 512×512.

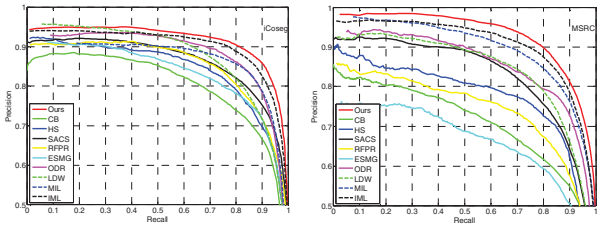


Fig. 3. Comparisons of precision-recall (PR) curves with nine co-saliency models on two public datasets.

Table 1. Comparisons of F-measure and MAE with nine co-saliency models. The best results are shown in bold.

Models	iCoseg [30]		MSRC [31]	
	$F_\beta\uparrow$	MAE↓	$F_\beta\uparrow$	MAE↓
Ours	**0.856**	**0.100**	**0.864**	**0.152**
CB [11]	0.695	0.167	0.573	0.317
HS [35]	0.702	0.181	0.726	0.281
SACS [12]	0.784	0.224	0.769	0.263
RFPR [13]	0.777	0.165	0.702	0.292
ESMG [36]	0.706	0.126	0.630	0.270
ODR [37]	0.800	0.107	0.780	0.191
LDW [27]	0.605	0.178	0.721	0.257
MIL [14]	0.616	0.159	0.753	0.212
IML [17]	0.846	0.101	0.851	0.164

Table 2. Ablation study for the proposed model on iCoseg dataset. The best results are shown in bold.

Model setting	$F_\beta\uparrow$	MAE↓
Ours	**0.856**	0.100
w/o refinement	0.838	**0.091**
w/o HFIM	0.808	0.107
CFEM w/o ConvLSTM	0.843	0.104
CFEM w/o COR	0.825	0.109
CEFM w/o CAT	0.839	0.103

3.2. Comparison with the state-of-the-art

We compare our model with nine state-of-the-art co-saliency detection models including CB [11], HS [35], SACS [12], RFPR [13], ESMG [36], ODR [37], LDW [27], MIL [14], and IML [17]. The co-saliency maps are either provided by the authors or generated by the implementation codes with the recommended parameter settings.

Quantitative Evaluation: Fig. 3 shows the comparisons of PR curves with nine co-saliency detection models on the iCoseg dataset and the MSRC dataset. Tab. 1 lists all the results of F-measure and MAE for two datasets. We observe that our model achieves the best performance on both datasets in terms of PR curve, F-measure and MAE. This clearly demonstrates the effectiveness and superiority of our model.

Qualitative Evaluation: The visual comparisons of the proposed model and nine co-saliency detection models on the iCoseg dataset and the MSRC dataset are shown in Fig. 4. Comparing with other models, our model highlights co-salient objects more consistently and suppresses background more effectively with well-defined boundaries. This proves that our model is robust for images with complex background and objects with various scales and shapes.

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l)

Fig. 4. Visual comparisons of co-saliency maps on iCoseg dataset (top 10 rows) and MSRC dataset (bottom 5 rows). (a) Images; (b) ground truths; (c) Ours; (d) CB [11]; (e) HS [35]; (f) SACS [12]; (g) RFPR [13]; (h) ESMG [36]; (i) ODR [37]; (j) LDW [27]; (k) MIL [14]; (l) IML [17].

3.3. Ablation study

To verify the contribution of each component in the proposed model, we provide five variants of our model with different settings. The five variants are set as follows: (1) "w/o refinement" means to remove the inter-image propagation based refinement; (2) "w/o HFIM" means to remove the HFIM and directly take $\mathbf{CSM}_1$ as the outputted co-saliency map; (3) "CFEM w/o ConvLSTM" represents that we use multiple concatenation layers and convolutional layers instead of the three ConvLSTMs; (4) "CFEM w/o COR" denotes that the CFEM has only one feature extraction strategy CAT; (5) "CFEM w/o CAT" denotes that the CFEM has only one feature extraction strategy COR. As shown in Tab. 2, we evaluate the five variants using F-measure and MAE on the iCoseg dataset. Obviously, all components in the proposed model contribute to promote the co-saliency detection performance.

4. CONCLUSION

In this paper, we propose a novel co-saliency detection model, which extracts features considering both intra-image and inter-image information. The proposed model focuses on the effective feature extraction and integration. We first employ an individual feature extraction module to get multi-level individual features of the image group. Then, we design a collaborative feature extraction module, in which all highest-level individual features of the image group are used to capture collaborative inter-image information. Finally, the multi-level individual feature of target image is exploit by the high-to-low feature integration module, to balance individual intra-image information. The qualitative and quantitative experiments on two public datasets demonstrate the effectiveness of the proposed model.

5. REFERENCES

[1] L. Ye, Z. Liu, L. Li, L. Shen, C. Bai, and Y. Wang, "Salient object segmentation via effective integration of saliency and objectness," *IEEE Trans. Multimedia*, vol. 19, no. 8, pp. 1742-1756, Aug. 2017.

[2] Q. Hou, J. Liu, M.-M. Cheng, A. Borji, and P.-H.-S. Torr, "Three birds one stone: A unified framework for salient object segmentation, edge detection and skeleton extraction," *arXiv preprint arXiv:1803.09860*, 2018.

[3] X. Liu, Z. Liu, Q. Jiao, O. Le Meur, and W. Zhao, "Saliency-aware inter-image color transfer for image manipulation," *Multimedia Tools and Applications*, vol. 78, no. 15, pp. 21629-21644, Aug. 2019.

[4] C. Bai, J. Chen, L. Huang, K. Kpalma, and S. Chen, "Saliency-based multi-feature modeling for semantic image retrieval," *J. Vis. Commun. Image Represent.*, vol. 50, pp. 199-204, Jan. 2018.

[5] R. Fan, Q. Hou, M.-M. Cheng, G. Yu, R.-R. Martin, and S. Hu, "Associating inter-image salient instances for weakly supervised semantic gegmentation," in *ECCV*, 2018.

[6] Y. Zeng, Y. Zhuge, H. Lu, and L. Zhang, "Joint learning of saliency detection and weakly supervised semantic segmentation," in *ICCV*, 2019.

[7] L. Li, Z. Liu, and J. Zhang, "Unsupervised image co-segmentation via guidance of simple images," *Neurocomputing*, vol. 275, pp. 1650-1661, Jan. 2018.

[8] K. Jerripothula, J. Cai, and J. Yuan, "Image co-segmentation via saliency co-fusion," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1896-1909, Sept. 2016.

[9] M. Cho, S. Kwak, C. Schmid, and J. Ponce, "Unsupervised object discovery and localization in the wild: part-based matching with bottom-up region proposals," In *CVPR*, 2015.

[10] D. Zhang, H. Fu, J. Han, A. Borji, and X. Li, "A review of cosaliency detection algorithms: Fundamentals, applications, and challenges," *ACM Trans. Intell. Syst. And Technol*., vol. 9, no. 4, pp. 1-31, Jan. 2018.

[11] H. Fu, X. Cao, and Z. Tu, "Cluster-based co-saliency detection," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3766-3778, Oct. 2013.

[12] X. Cao, Z. Tao, B. Zhang, H. Fu, and W. Feng, "Self-adaptively weighted co-saliency detection via rank constraint," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 4175-4186, Sep. 2014.

[13] L. Li, Z. Liu, W. Zou, X. Zhang, and O. Le Meur, "Co-saliency detection based on region-level fusion and pixel-level refinement," in *ICME*, 2014.

[14] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865-878, May. 2017.

[15] J. Han, G. Cheng, Z. Li, and D. Zhang, "A unified metric learning-based framework for co-saliency detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2473–2483, Oct. 2018.

[16] L. Wei, S. Zhao, O. Bourahla, X. Li, and F. Wu, "Group-wise deep co-saliency detection," in *IJCAI*, 2017.

[17] J. Ren, Z. Liu, X. Zhou, C. Bai, and G. Sun, "Co-saliency detection via integration of multi-layer convolutional features and inter-image propagation," *Neurocomputing*, vol. 371, pp. 137-146, Jan. 2020.

[18] W. Li, O.-H. Jafari, and C. Rother, "Deep object co-segmentation," in *ACCV*, 2018.

[19] P. Fischer, A. Dosovitskiy, E. Ilg, P. Hausser, C. Hazırbas, and V. Golkov, "FlowNet: Learning optical flow with convolutional networks," in *ICCV*, 2016.

[20] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *NeurIPS*, 2015.

[21] S. Hochreiter, and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Apr. 1997.

[22] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper ConvLSTM for video salient object detection," in *ECCV*, 2018.

[23] T. Liu, M. Xu, and Z. Wang, "Removing rain in videos: a large-scale database and a two-stream ConvLSTM approach," in *ICME*, 2019.

[24] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[25] X. Zhou, Z. Liu, C. Gong, G. Li, and M. Huang, "Video saliency detection using deep convolutional neural networks," in *PRCV*, 2018.

[26] G. Li, Z. Liu, R. Shi, and W. Wei, "Constrained fixation point based segmentation via deep neural network," *Neurocomputing*, vol. 368, pp. 180-187, Nov. 2019.

[27] D. Zhang, J. Han, C. Li, J. Wang, X. Li, "Detection of co-salient objects by looking deep and wide," *Int. J. Comput. Vision*, vol. 120, no. 2, pp. 215–232, Nov. 2016.

[28] M. Everingham, L. Van Gool, C.-K.-I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vision*, vol. 88, no. 2, pp. 303-338, Jun. 2010.

[29] J. Dai, Y. Wu, J. Zhou, and S. Zhu, "Cosegmentation and cosketch by unsupervised learning," in *ICCV*, 2013.

[30] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "iCoseg: Interactive co-segmentation with intelligent scribble guidance," in *CVPR*, 2010.

[31] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *ICCV*, 2005.

[32] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.*, vol. 9, no. 1, pp. 62-66, Jan. 1979.

[33] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *ACM MM*, 2014.

[34] D.-P. Kingma, and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[35] Z. Liu, W. Zou, L. Li, L. Shen, and O. Le Meur, "Co-saliency detection based on hierarchical segmentation," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 88-92, Jan. 2014.

[36] Y. Li, K. Fu, Z. Liu, and J. Yang, "Efficient saliency-model-guided visual co-saliency detection," *IEEE Signal Process. Lett.*, vol. 22, no. 5, pp. 588-592, May 2015.

[37] L.Ye, Z.Liu, J.Li, W.Zhao and L.Shen, "Co-saliency detection via co-salient object discovery and recovery," *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 2073-2077, Nov. 2015.