Single-Pixel Salient Object Detection via Discrete Cosine Spectrum Acquisition and Deep Learning

advisor

Dr. Maryam abedi

student

Mohammad shahpouri arani

fall

2021

Table of Contents

1- Theories and methods

They first calculated discrite cosine transform from image with below formula:

$$P\pm(x,\ y;\ u,\ v)=\pm\ a\ .c(u)c(v)C(x\ ,u\ ,\ M)C(\ y\ ,\ v\ ,\ N)+b$$

Equation 1- Discrete cosine transform

where (x, y) is the spatial coordinates; (u, v) is the spatial frequency; a is the contrast; c(τ) = 1/ 2 when τ = 0; c(τ) = 1 when $\tau \neq 0$; C(δ, ω, κ) = cos [(δ + 0.5)ωλ/κ]; M and N indicate that size of image R(x, y) to be reconstructed is M × N; b is a constant to ensure no negative number in P ± (x, y; u, v).

To binarize the gray-scale patterns P ± (x, y; u, v) they used this formula:

$$P_B\pm(x\ ,\ y;\ u\ ,\ v)=F\{U_\eta\{P\pm(x,\ y;\ u,\ v)\}\}$$

Equation 2- Make digital micromirror device

where $P_B\pm(x,\ y;\ u\ ,\ v)$ are binary discrete cosine basis patterns; a and b are set to 0.5; U_η\{•\} denotes η-folds upsampling where $P_B\pm(x,\ y;\ u\ ,\ v)$ are binary discrete cosine basis patterns; a and b are set to 0.5; U_η\{•\} denotes η-folds upsampling by 'bicubic' image interpolation algorithm; F\{•\} denotes Floyd-Steinberg error diffusion dithering.

corresponding responses D ± (u, v) from the scene was:

$$D\pm(u,\ v)=k\ .\sum_{0}^{\eta M-1}\sum_{0}^{\eta N-1}P_B\pm(x,\ y\ ;u\ ,\ v)\ .\ T(x,\ y)+e$$

Equation 3- Illuminating the scene

where k is the scale factor depending on the size and location of detector and e is the response of detector to background illumination.

Inverse discrete cosine transform was calculated as below:

$$RD(u, v) = \left\{ D_+(u, v) - \frac{D_+(u, v) + D_-(u, v)}{2} \right\} / h$$

Equation 4- Inver discrete cosine transform

where (U, V) is a selected and fixed frequency; h is a constant.

To train CNN-based network, they applied inverse discrete cosine transformed on RD(u, v) to achieve R(x, y) to be the input of the nework. The network was used to extract regions of the salient objects.

Origin of their network inspired by PoolNet [1] with ResNet-50 [2]. They used DUTS [3] dataset. First, they converted images to gray-scale and resized to 128 × 128. Then, DCT is applied to the preprocessed images to obtain their corresponding DCS. Next, image pyramid is made by 4 squares with different sizes (respectively 64 × 64, 32 × 32, 16 × 16, 8 × 8). Then, inverse DCT applied to achieve corresponding images. Shown in Fig. 1.

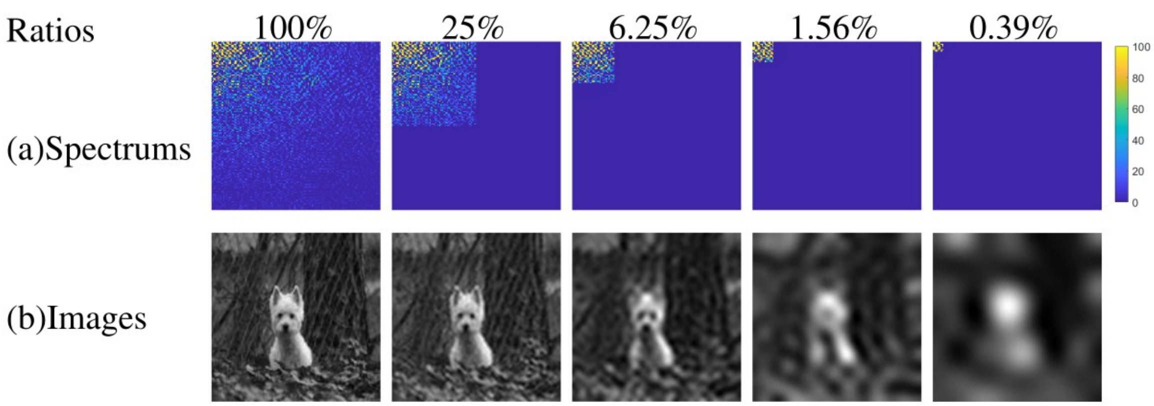

Figure 1- Training images with different utilization ratios of DCS.

So, 5 sets of images (in total 52765) with different utilization ratios (respectively 100%, 25%, 6.25%, 1.56%, 0.39%) of spectrum are generated as training set. In the training phase, the 52765 training images with 128 × 128 pixels are mixed together to train the network. And as the settings in [1], the loss function is standard binary cross entropy loss. The network is trained for 24 epochs by Adam [4] optimizer with weight decay of 5e-4. The learning rate is initially 5e-5 and changed to 5e-6 after 15 epochs. Parameters of ResNet-50 pretrained on the ImageNet dataset [5] is used to initial the backbone and rest parameters of the network are randomly initialized.

2- Experimental results

They set M = N = 128, η = 2 and U = V = 8. For test set the 1000 images in ECSSD [6] were adopted. 4-level image pyramid of test set were reconstructed as input of CNN network.

2-1 Evaluation

For evaluation they used F-measure score and mean absolute error (MAE).

F-measure is denoted as F β :

$$F_{\beta} = \frac{(1 + \beta^{2}) \times Precision \times Recall}{\beta^{2} \times Precision \times Recall}$$

Equation 5- F-measure

where Precision = |B ∩ G| / |B|; Recall = |B ∩ G| / |G|; B is a mask generated by binarizing the prediction saliency map S with a threshold; G is ground truth; |•| denotes the accumulation of non-zero entries; β^2 is empirically set to 0.3.

MAE is calculated by:

$$MAE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |S(i, j) - G(i, j)|$$

Equation 6- Mean absolute error

References

[1]...........J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, “A Simple Pooling-Based Design for Real-Time Salient Object Detection,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 3912–3921. doi: 10.1109/CVPR.2019.00404.

[2]......................K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[3]L. Wang *et al.*, “Learning to Detect Salient Objects with Image-Level Supervision,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 3796–3805. doi: 10.1109/CVPR.2017.404.

[4]D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *ArXiv14126980 Cs*, Jan. 2017, Accessed: Oct. 11, 2021. [Online]. Available: http://arxiv.org/abs/1412.6980

[5]......A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems*, 2012, vol. 25. Accessed: Oct. 11, 2021. [Online]. Available: https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

[6]……..Q. Yan, L. Xu, J. Shi, and J. Jia, “Hierarchical Saliency Detection,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2013, pp. 1155–1162. doi: 10.1109/CVPR.2013.153.