Learning Salient Objects in a Scene using Superpixel-augmented Convolutional Neural Network

Shashank Tripathi, Yash Patel

Robotics Institute, Carnegie Mellon University
{shashant, yashp}@andrew.cmu.edu

Abstract. In this project, we attempt to determine salient objects in a given natural scene image. Salient objects are loosely defined as objects that "pop-out" in a scene in the opinion of a human observer – an object that attracts attention of the human brain and visual system. Saliency detection in natural scene images is a popularly studied problem in computer vision [1,2,3,4,5]. It has a wide range of applications in computer vision and image processing tasks, such as image/video compression and summarization [6], content-aware image resizing [7] and photo collage [8]. Saliency information has also been exploited in high-level vision tasks, such as object detection [9] and image segmentation.
In this project, we make use of a method known as *SuperCNN* proposed by He et al. [10]. This method specifically attempts to use Convolutional Neural Network (CNN) trained on super-pixels to solve saliency detection problem. The problem is formulated as a binary-classification, where salient objects are labeled 1 and background is labeled 0.
An obvious problem with using CNN directly on raw image pixels is that CNNs use a highly low-resolution image as input (227×227 pixels for the popular Alexnet [11]). Using high resolution images increase the computational cost of feed-forward and back-propagation significantly. Contrast based saliency should be detected from a larger context and such low resolutions put a cost on the context in the image. The solution as proposed in [11] is to cluster the image into Superpixels, which retain long-range context, while maintaining low-resolution input. The method builds the solution keeping following ideas in mind:
- It has been shown in psychological studies that contrast is a major factor to determine visual attention among humans.
- Objects with contrast can occur scattered across the background, but foreground objects usually tend to be locally connected.

Keywords: Saliency Detection, Convolutional Neural Networks, SLIC superpixels

1 Introduction

"Thus, from the war of nature, from famine and death, the most exalted object which we are capable of conceiving, namely, the production of the higher animals, directly follows." – On the Origin of Species, Charles Darwin.

Just as Charles Darwin attributed the success of all life on earth to a ubiquitous power called evolution, there is a strong belief that humans are only a bi-product of years of trial-and-error. The human visual system has developed over the years to optimally trade-off energy consumption and utility. It has been shown that the human visual system processes images in a hierarchical fashion [Marr and Poggio, 1983 [12]], starting from low-level context independent features like edges, contrast variation, color separation, etc to high-level features such as shape and structure. This observation has been instrumental in the development of the Convolutional Neural Networks which learn hierarchical weights in a similar fashion. The paper [10] focuses on the use of low-level features to model human attention in a natural scene image.

Nassier et al. (1964) proposed a model of the human visual system consisting of pre-attentive and attentive stages. The pre-attentive stage focuses human attention on "pop-out" features in the image. These "pop-out" features Julesz et al. [13], 1995 are regions in the image that present some form of spatial discontinuity, be it contrast separation or color variance. These features define visual saliency. Saliency detection is therefore an attempt to mimic the human visual system's pre-attentive stage. In the context of this project, we try to use the power of Convolutional Neural Networks (CNN) to identify salient regions in an image.

Many past approaches at saliency detection try to capture contrast cues to determine salient regions [Cheng et al. [14] 2011, He and Lau et al. [15] 2014]. However, just as computer vision approaches using hand-crafted features have been shown to be unsuitable for a general setting, these approaches too fail to generalize to all images. Many researchers have therefore tried to adapt state-of-the-art learning techniques to detect salient regions but have primarily focused on integrating saliency maps obtained from hand-crafted features. [Jiang et al. [16] 2013]. This project takes inspiration from the work presented by He et al. [10], where they have attempted to extract saliency labels on superpixels using a shallow Convolutional Neural Network. Using superpixels allow reduction in input dimension while retaining large image context.

2 Dataset

Throughout this project, we have used the Extended Complex Scene Saliency dataset (ECSSD) [17] which provides 1000 natural scene images, along with ground truth segmentation labels for salient regions. Few examples from the dataset are shown in 1.

3 Method

As stated earlier, a straight forward idea could be to directly feed the image pixels as input to a CNN architecture and generate saliency labels for each of the pixels. The issue with this approach is that most CNN architecture heavily down-sample the input image. In standard and conventional setups the image

Fig. 1: Sample image and corresponding ground truth segmentation labels from the ECSSD dataset [17].

is resized to 227×277 pixels for Alexnet [11] and 224×244 pixels for VGG-16 [18]. There are various problems with using such standard CNNs for saliency detections:

- Making per-pixel predictions greatly increase the number of parameters in fully-connected layers. Increased number of parameters require larger training data for effective learning.
- Resizing the images to fixed sizes destroys the original image aspect ratio. It has been recently observed that preserving image aspect ratio while training can improve the performance of deep architectures significantly.
- Low resolution of input image puts a cost on the larger context in the image.
- Conventional CNN architectures such as AlexNet [11] and VGG-16 [18] have a limited receptive field. While these architectures might be very useful for applications like image classification, saliency detection requires global information.

To mitigate the above issues, the method proposed in [10] make use of superpixel for saliency detection in natural scene images. Humans don't look at an image as a discretized matrix of pixels. To bridge the gap between how the human visual system perceives an image, many studies have proposed grouping pixels into perceptually meaningful regions that capture image redundancy.

Overall method is shown in Fig. 2. The adopted method can be categorized into following major parts:

- Superpixels are generated given a natural scene image.
- For each superpixel appropriate representations are generated as presented in [10].

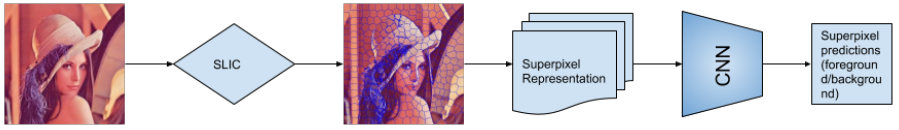

Fig. 2: Overall method: First the superpixels are generated, then superpixels representations are generated. Finally, each superpixel is classified as foreground or background using shallow CNN.

– Given the superpixel representation, each super pixel is classified as "background" or "foreground" using a shallow CNN.

3.1 Superpixel Generation

Firstly step is the generation of superpixels. In general, superpixel regions have the following properties:

1. Superpixels fit well to the object boundaries in the image.
2. Reduce pixelwise redundancies by grouping local pixels together based on color and intensity similarities.
3. Superpixels should be efficient to compute and should offer an efficiency advantage over and above their computation cost when used in a subsequent task

Achanta et al. [19] proposed a powerful and widely used algorithm for superpixel segmentation called Simple Linear Iterative Clustering (SLIC). SLIC superpixels are based on the k-means clustering algorithm at its core. A simple modification that SLIC uses over the naïve k-means clustering is that it restricts the search space to a neighborhood of the current mean. The neighborhood is proportional to the superpixel size. This reduces the computational complexity to linear time (O(N)) where N are the number pixels, and is independent of the number of superpixels. Instead of taking Euclidean distance to propose cluster assignment, a color and spatial proximity based distance measure is used instead. The algorithm for SLIC superpixels can be summarised as below [Achanta et al. [19]]:

1. Convert images from RGB colorspace to LAB color-space. CIE LAB colorspace bears a closer resemblance to human perception. It treats black and white as their own channel, i.e. it separates contrast from color, thereby having the advantage of a wider gamut. Contrast and color separation is important especially with regards to saliency detection as salient objects are localized based on both color and contrast separation in the context of the full image.
2. Divide image into a square grid with the number of grid regions equal to the number of superpixels required. Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^t$

Fig. 3: Samples images of generated superpixels using SLIC [19] on baseball game images.

randomly in each of the grid regions. Move cluster centers to the lowest gradient position in a $3x3$ neighbourhood so that the cluster center does not lie on object boundaries.

3. For each cluster center, we look at pixels within 2 gridspaces from that cluster center and iteratively assign it to the cluster center based on the following distance metric:

$$d_c = \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2}$$
$$d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$$
$$D' = \sqrt{(\frac{d_c}{N_c})^2 + (\frac{d_s}{N_s}^2)}$$

The distance metrics are normalized by the respective maximum distance, N_s and N_c within a cluster.

Few examples of generated superpixels are shown in Fig. 3. It is apparent that the superpixels accurately group pixels based on color similarities. Moreover, superpixels adhere well to image boundaries.

An inherent problem with superpixel segmentation is that each time a different set of superpixels are generated stochastically depending upon the initial initialization of the cluster means. In the process, Superpixels lose spatial information. To recover spatial information, a "color uniqueness matrix" as proposed in [10] can be defined on the superpixels. Let a given image I and the segmented regions $R = [r_1, \ldots, r_x, \ldots, r_N]^T$, a $N \times N \times 3$ color uniqueness matrix Q can be defined.

$$Q = \begin{bmatrix} q^c_{11} & \cdots & q^c_{1j} & \cdots & q^c_{1M} \\ \vdots & \ddots & & \ddots & \\ q^c_{x1} & \cdots & q^c_{xj} & \cdots & q^c_{xM} \\ \vdots & \ddots & & \ddots & \\ q^c_{N1} & \cdots & q^c_{Nj} & \cdots & q^c_{NM} \end{bmatrix}$$

where each element q^c_{xj} represents the weighted difference in the mean color vector (across channels) of superpixel region x with every other superpixel region j. So $M = N$ in our case.

$$q^c_{xj} = t(r_j).|C(r_x) - C(r_j)|.w(P(r_x), P(rj))$$

where $t(r_j)$ counts the total number of pixels in region r_j. This term weights superpixels with more number of pixels to have higher contribution to the contrast than those with fewer pixels. $C(r_x)$ is the mean color vector of region r_x and $|C(r_x) - C(r_j)|$ is the 3D vector storing absolute differences of each color channel. $P(r_x)$ is the mean position of (r_x). The term $w(P(r_x), P(rj))$ is a Gaussian weight to attach higher contribution to pixels spatially closer to each other and $w(P(r_x), P(rj)) = exp(-\frac{1}{2\sigma_s^2}||P(r_x) - P(r_j)||^2$. Each row in Q is then sorted by the spatial distance to region r_x to maintain local correlation such that the convolution operation in the CNN makes sense. Sorting groups neighbouring superpixels together. In summary, each row of the Q matrix describes the color differences between each region r_X with all other M-1 superpixels in the image. The Gaussian distance weights include spatial information into the Q matrix which would have been lost otherwise.

The color uniqueness matrix does a good job in capturing salient objects separated by color variations in the image. Q matrix essentially represents regions that show significant color rarity in their neighbourhood. However, considering just color rarity to determine saliency might be an oversimplification. Consider images in 4 and their ground-truth salient object segmentations. It is apparent that even though certain objects display color rarity in the image eg. flowers, colored balls in the background, fishes etc, they are not salient. We need to ignore superpixels that display higher local contrast separation but form part of the background. Primarily, we need to a metric to differentiate foreground objects from background objects. [Lie et al. [20] 2011] show that foreground objects are compact i.e. locally connected, whereas background objects are distributed in the image. To capture this difference, we define a new matrix Q' which is complementary to the original Q matrix.

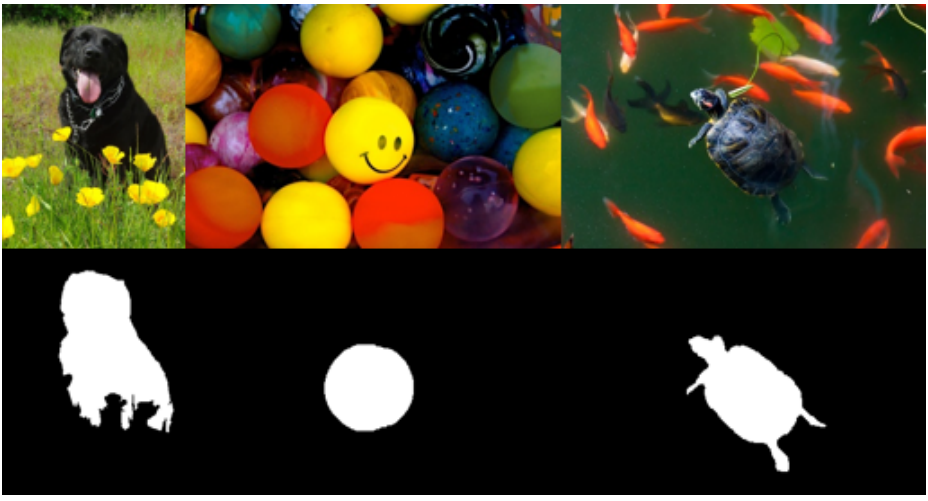

Fig. 4: Examples of scattered non-salient background.

$$Q' = \begin{bmatrix} q^d_{11} & \cdots & q^d_{1j} & \cdots & q^d_{1M} \\ \vdots & \ddots & & \ddots & \\ q^d_{x1} & \cdots & q^d_{xj} & \cdots & q^d_{xM} \\ \vdots & \ddots & & \ddots & \\ q^d_{N1} & \cdots & q^d_{Nj} & \cdots & q^d_{NM} \end{bmatrix}$$

where each element $q;^d_{xj}$ represents the weighted difference in the mean position vector (across channels) of superpixel region x with every other superpixel region j. Each q^d_{xj} can be defined as:

$$q^d_{xj} = t(r_j).|P(r_x) - P(r_j)|.w(C(r_x), C(rj))$$

where $t(r_j)$ counts the total number of pixels in region r_j. This term weights superpixels with more number of pixels to have higher contribution to the contrast than those with fewer pixels. $P(r_x)$ is the mean position vector of region r_x and $|P(r_x) - P(r_j)|$ is the 3D vector storing absolute differences of each color channel. $C(r_x)$ is the mean color of (r_x) across the 3 channels. The term $w(C(r_x), C(rj))$ is a Gaussian weight to attach higher contribution to pixels spatially closer to each other in the color space and $w(C(r_x), C(rj)) = exp(-\frac{1}{2\sigma_s^2}||C(r_x) - C(r_j)||^2$. Each row in Q' is then sorted by the spatial distance to region r_x to maintain local correlation such that the convolution operation in the CNN makes sense. This is similar to what we did while forming the Q matrix.

3.2 Superpixel Classification using CNN

In the adopted method [10] saliency detection is formulated as a two-class classification problem on superpixels. Each superpixel is classified as foreground or background. The relationship between the binary labellings and the Q matrix (3.1) can be efficiently learned by a shallow Convolutional Neural Network. Salient objects display a perceptible difference in color compared to the surrounding and are locally connected. The hypothesis is that the Q matrix (3.1) accurately encompasses these properties of salient object from which a Convolutional Neural Network can accurately learn internal representation to classify superpixels as background or foreground.

In this project, we extensively experimented with different shallow network architectures by varying number of convolutional layers, filter sizes, number of channels, loss functions and learning rates. We observed that following architecture worked best on given dataset: a three $1D$ convolutional layers with "tanh" activation functions, followed by Global-Average-Pooling [21] and two fully-connected layers with "tanh" activation functions. Detailed CNN architecture used is provided in Tab. 1.

Type	Size/Stride	Number of channels
input	M	3
Conv1D, tanh	10×1	5
MaxPool	$2 \times 1/2 \times 1$	-
Conv1D, tanh	20×1	10
Maxpool	$2 \times 1/2 \times 1$	-
Conv1D, tanh	20×1	20
Maxpool	$2 \times 1/2 \times 1$	-
Global-avg-pool	-	-
Fully-con, tanh	20	-
Dropout (0.3)	-	-
Fully-con, Softmax	2	-

Table 1: CNN architecture used in this project. Note: size of Q matrix is M×M.

It is important to note that given CNN uses $1D$ convolutional layers. Each row of Q-matrix contains the representation for given superpixel. Thus, the CNN takes each row from Q-matrix as input and predict background or foreground class. Compared to few tens of millions of parameters in $2D$ deep convolutional architectures such as AlexNet [11] and VGG-16 [18], this CNN has just $5,233$ parameters. Upon convergence the CNN achieves a classification accuracy of 89%.

Following are specific training details:

- CNN is trained on $1,000$ images provided by ECSSD dataset [17] alone.
- Categorical-cross-entropy loss function over two classes (background and foreground) is used.

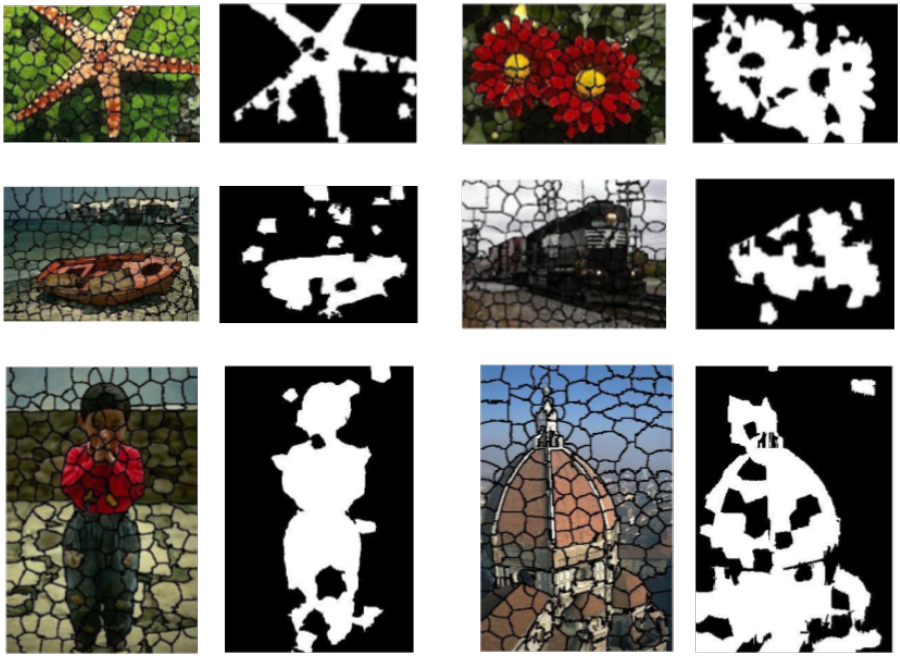

Fig. 5: Qualitative results obtained on some sample images.

- Stochastic Gradient Descent (SGD) optimizer is used to minimize the loss function over training data.
- Initial learning rate of 0.001 is used. Learning rate is decreased after every 5 epochs by a factor of 0.5.
- Batch size of 256 is used throughout the training.
- The number of background superpixels (75% approximately) is considerably higher than the number of foreground superpixel (25% approximately) in ECSSD [10] dataset. In order to handle class imbalance during training, 50% superpixels representations of foreground and 50% superpixels representations of background are used for every mini-batch.
- Convergence is observed after 15 epochs.

4 Qualitative Results

Fig. 5 demonstrates qualitative results for the adopted approach and implementation. It it evident from the results that SuperCNN based approach gives reasonable saliency detection results.

5 Conclusions

In this project, we observed that hand-crafted features and classification using Convolutional Neural Network (CNN) can give decent performance for saliency detection in natural scene images. Number of parameters in CNN are considerably low (order of 3) when compared to standard deep architectures in computer vision. Thus, training here does not require initialization of weights using the network trained on ImageNet classification dataset (widely done for fine-grained computer vision tasks using standard CNN architectures). The implementation of this project has been released publicly on github, please refer to this link for code: https://github.com/yash0307/SuperCNN.

References

1. Han, X., Gao, C., Yu, Y.: Deepsketch2face: A deep learning based sketching system for 3d face and caricature modeling. ACM Transactions on Graphics **36**(4) (July 2017)
2. Han, X., Li, Z., Huang, H., Kalogerakis, E., Yu, Y.: High-resolution shape completion using deep neural networks for global structure and local geometry inference. In: IEEE International Conference on Computer Vision (ICCV). (2017)
3. Ge, W., Yu, Y.: Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
4. Li, G., Xie, Y., Lin, L., Yu, Y.: Instance-level salient object segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
5. Li, G., Yu, Y.: Deep contrast learning for salient object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2016)
6. Ma, Y.F., Lu, L., Zhang, H.J., Li, M.: A user attention model for video summarization. In: Proceedings of the tenth ACM international conference on Multimedia. (2002)
7. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. In: ACM Transactions on graphics (TOG). (2007)
8. Wang, J., Quan, L., Sun, J., Tang, X., Shum, H.Y.: Picture collage. In: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. (2006)
9. Luo, P., Tian, Y., Wang, X., Tang, X.: Switchable deep network for pedestrian detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014)
10. He, S., Lau, R.W., Liu, W., Huang, Z., Yang, Q.: Supercnn: A superpixelwise convolutional neural network for salient object detection. International journal of computer vision (2015)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012)
12. Poggio, T.: Visual algorithms. In: Physical and biological processing of images. (1983)
13. Sarnyai, Z., Bíró, É., Gardi, J., Vecsernyés, M., Julesz, J., Telegdy, G.: Brain corticotropin-releasing factor mediates 'anxiety-like'behavior induced by cocaine withdrawal in rats. Brain research (1995)

14. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.M.: Salient object detection and segmentation. IEEE Transactions on Pattern Analysis & Machine Intelligence (2011)
15. He, S., Lau, R.W.: Saliency detection with flash and no-flash image pairs. In: European Conference on Computer Vision. (2014)
16. Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., Li, S.: Salient object detection: A discriminative regional feature integration approach. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2013)
17. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2013)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
19. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. IEEE transactions on pattern analysis and machine intelligence (2012)
20. Lie, W.N., Chen, C.Y., Chen, W.C.: 2d to 3d video conversion with key-frame depth propagation and trilateral filtering. Electronics Letters (2011)
21. Lin, M., Chen, Q., Yan, S.: Network in network. arXiv preprint arXiv:1312.4400 (2013)