# ConnNet: A Long-Range Relation-Aware Pixel-Connectivity Network for Salient Segmentation

Michael Kampffmeyer, Nanqing Dong, Xiaodan Liang, Yujia Zhang, and Eric P. Xing

***Abstract*—Salient segmentation aims to segment out attention-grabbing regions, a critical yet challenging task and the foundation of many high-level computer vision applications. It requires semantic-aware grouping of pixels into salient regions and benefits from the utilization of global multi-scale contexts to achieve good local reasoning. Previous works often address it as two-class segmentation problems utilizing complicated multi-step procedures including refinement networks and complex graphical models. We argue that semantic salient segmentation can instead be effectively resolved by reformulating it as a simple yet intuitive pixel-pair based connectivity prediction task. Following the intuition that salient objects can be naturally grouped via semantic-aware connectivity between neighboring pixels, we propose a pure Connectivity Net (ConnNet). ConnNet predicts connectivity probabilities of each pixel with its neighboring pixels by leveraging multi-level cascade contexts embedded in the image and long-range pixel relations. We investigate our approach on two tasks, namely salient object segmentation and salient instance-level segmentation, and illustrate that improvements can be obtained by modeling these tasks as connectivity instead of binary segmentation tasks. We achieve state-of-the-art performance, outperforming or being comparable to existing approaches while reducing training time due to our less complex approach.**

***Index Terms*—Salient segmentation, Convolutional neural networks, Salient instance-level segmentation, Connectivity.**

## I. Introduction

Salient segmentation, the task of locating attention-grabbing regions in the image, is a fundamental challenge in computer vision and is often used as a pre-processing step for object detection [1], video summarization [2], face detection [3] and motion detection [4]. Traditionally, salient segmentation methods have to a large extent relied on hand-crafted models and feature selection [5], [6].

Fueled by the recent advances that deep Convolutional Neural Networks (CNN) have brought to the field of computer vision, achieving state-of-the-art performance in tasks such as classification [7], [8], object detection [9], [10] and segmentation [11], [12], salient segmentation has increasingly been performed using CNNs [13], [14], [15]. These provide more robust features as more high-level information is being extracted from the image and allow for end-to-end learning, where model parameters can be learned jointly and inference can be performed in a single pass through the network.

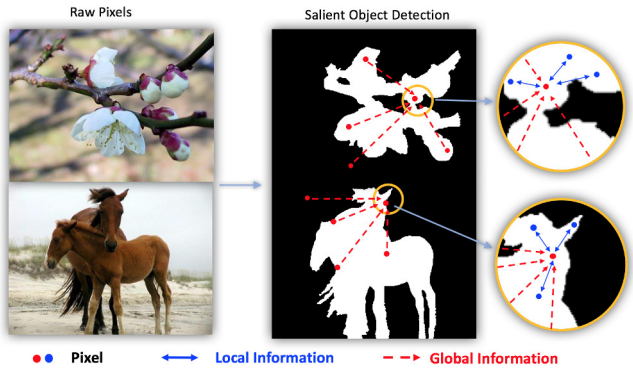



Fig. 1: Motivation for using connectivity for salient segmentation. Salient regions are modeled as connected regions. Our method predicts if a given pixel is connected to its neighbor based on local and global relations between pixels.

Modern approaches to salient segmentation make to a large extent use of fully convolutional neural networks [11], viewing the task as a binary pixel-wise classification problem. These approaches incorporate recent advances in the computer vision field such as adversarial training [16], [17] and attention models and often make use of pre-processing and post-processing steps such as superpixel segmentation [13], [18] and graphical models [13] to improve overall performance. These steps can be complex in their own right, leading to larger training and inference times. Further, these approaches often are not learnable as part of the overall architecture, leading to complex multi-stage models.

Inspired by the fact that salient segmentation models are becoming more and more complex, we propose to take a step back and look at the underlying foundation of the problem. Instead of approaching the task as a segmentation problem we believe that improvements can be achieved by splitting the segmentation task up into the sub-task of predicting foreground connectivity between neighboring pixels. We make use of

The first two authors contributed equally to this work.

M. Kampffmeyer is with Machine Learning Group, UiT The Arctic University of Norway, 9019 Tromsø, Norway. (email: michael.c.kampffmeyer@uit.no.) Work done while at Carnegie Mellon University.

N. Dong is with Cornell University, Ithaca, NY 14850, USA. (email: nd367@cornell.edu.) Work done while at Petuum Inc..

X. Liang and Eric P. Xing are with Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA. (email: xdliang328@gmail.com, epxing@cs.cmu.edu.)

Y. Zhang is with Institute of Automation, Chinese Academy of Sciences; School of Computer and Control Engineering, University of Chinese Academy of Sciences, 100190 Beijing, China. (email: zhangyujia2014@ia.ac.cn.)

a Relation-aware Convolutional Network for the prediction tasks. Due to its hierarchical nature it allows us to integrate semantic-awareness to our connectivity prediction and effectively disentangle background regions. Instead of having a combined objective that maintains semantic rationality and overall region smoothness, each sub-task only focuses on groupings in a specific direction. To preserve global multi-scale context, our approach further integrates long-range dependencies to improve overall performance. Note, this does not exclude the use of advanced approaches such as multi-scale refinement networks, conditional random fields, attention, and additional adversarial losses, to improve performance, as the overall architecture is near-identical to the architecture of a segmentation network.

Another advantage of the connectivity objective is the fact that it allows us to integrate relationship prediction between pixels explicitly into the optimization problem, which can be interpreted as mimicking graphical inference in a single unified compact model. Further, this approach can also be viewed as a way to learn better features, due to the fact that we force our model to learn robust representations that allow us to predict not only a given pixel but also its relation to the surrounding pixels. More importantly, we can also interpret the approach as an ensemble approach, utilizing the fact that connectivity is a symmetric measurement and pairs of neighboring pixels have to agree on connectivity. This will be further discussed in Section III.

Based on this intuition, our proposed architecture, ConnNet, is based on the idea that salient objects can be modeled as connected pixel regions in the image utilizing local and global relationships. This concept is illustrated in Figure 1. We utilize a convolutional neural network to predict, for a given pixel, whether the eight surrounding neighbors are connected to it. Considering the pairwise connectivity between the intermediate pixels allows us to obtain a final salient segmentation result.

The main contributions of this work are:

- We illustrate that connectivity modeling can be a good alternative to more traditional segmentation for the task of salient segmentation. Comparing our approach to an identical architecture trained for the segmentation task we observe that ConnNet outperforms the segmentation network on an extensive set of benchmark datasets.
- We develop an approach for salient object segmentation that outperforms previous state-of-the-art approaches on several datasets, but also, due to its simplicity, considerably reduces training and inference time. We also extend the idea to the task of instance-level salient segmentation.
- We investigate the influence of different pixel connectivity modeling approaches on the overall performance.

The remainder of this paper is organized as follows. Section II reviews some of the related work and places our work into context. Section III introduces the proposed connectivity-based approach ConnNet. In Section IV we perform experiments both on the salient object segmentation and the salient instance-level segmentation task. Finally, in Section V we provide concluding remarks.

## II. Related work

Our approach is related to previous work in the fields of semantic segmentation, salient object segmentation, instance segmentation and instance-level salient segmentation, which we briefly review in this section before introducing our proposed methodology.

Salient object segmentation has been a field of interest in computer vision for a long time, with traditional approaches being largely based on hand-crafted feature design. Overall, we can categorize most traditional approaches into methods that perform salient object segmentation based on either low-level appearances, such as color and texture [5], [6], [19], or high-level object knowledge, such as object detectors and global image statistics [20], [21], [22]. Further, hybrid approaches that combine both low-level and high-level features exist [23], [24]. These approaches commonly work well in the scenarios that they are designed for, but often break down in more complex cases [13], [25]. For instance, color can be largely affected by lighting conditions and local contrast features might struggle with homogeneous regions.

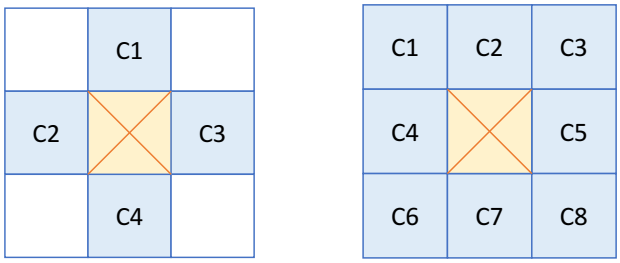
Convolutional neural networks (CNNs) have in recent years led to large advances in the field of image segmentation, or pixel-wise classification, due to their ability to learn complex high-level information without requiring the extensive design of handcrafted features [11], [8]. To achieve more robust and more precise salient object segmentation, these models have recently been extensively utilized for the task of salient object segmentation, by rephrasing the task of salient object segmentation as a binary pixel-wise classification. Initial approaches utilized patch-base approaches [18], [26], [27], where the CNN is presented with image patches and is trained to classify the center pixel or center superpixel of a given patch. Inference in these approaches is generally highly inefficient with regards to computation and memory requirements, as the models have to perform a forward pass for potentially every pixel in the image. However, they did outperform traditional, non-deep learning based methods illustrating the potential of CNNs for salient object segmentation.

More recently fully convolutional neural networks [11], networks that perform pixel-to-pixel segmentation in a single forward pass, have replaced patch-based approaches. These networks can be viewed as encoder-decoder architectures, where the original image is mapped to a lower resolution representation and then mapped back to the original architecture using fractionally strided convolutions [11]. These networks allow the design of end-to-end trainable models that extract features and perform salient segmentation for the complete image. Due to their superiority over patch-based approaches both with regards to performance and computational efficiency, they provide the base architecture for most of the recent state-of-the-art approaches for salient object segmentation [13], [14], [15].

Lately, to improve salient object segmentation performance, several additional components are added to the architecture. For instance, Liu et al. [14] and Wang et al. [15] integrate recurrent neural networks into their architecture to refine the salient object segmentation mask over several time steps. Li

et al. [13] utilize a two-stream architecture, where one is a fully convolutional CNN that produces a pixel-wise salient mask and where the second stream performs segment level salience segmentation on an image that has been segmented into superpixels. Given the two salient masks, they further propose the use of fully connected conditional random fields to merge the two streams to improve spatial coherence. Adversarial training techniques have also been proposed [16], [17], where a discriminator is trained to distinguish between the predicted and ground truth saliency maps. The discriminator loss is optimized jointly with the segmentation loss and can be interpreted as a regularization loss that penalizes higher-order inconsistencies between the prediction and ground truth. More recently, Hou et al. [28] utilizes a skip-layer architecture based on the Holistically-Nested edge detector architecture [29] by introducing short connections. To our knowledge, the current state-of-the-art in salient object segmentation is MSRNet [30], which incorporates the idea that information at different scales will be useful for the salient object segmentation task in a model that makes use of multi-scale refinement networks and utilizes attention to combine different scales.

Instance-aware salient segmentation is a more challenging task that was recently proposed in Li et al. [30], who propose to extend the salient object segmentation MSRNet to instance-level salient segmentation. This is done by finetuning a copy of their MSRNet for salient contour detection. A multiscale combinatorial grouping is then used to convert these contours into salient object proposals. The total number of salient object proposals is then reduced using a MAP-based subset optimization method to provide a small compact set of proposals. An instance-level salient segmentation result is obtained by combining the salient object segmentation and the salient object proposals and by applying a conditional random field refinement step. The task of instance-aware salient segmentation is inspired by the recent advances in semantic instance segmentation [31], [32], [33], [34]. Instance segmentation aims to combine the task of object detection and semantic segmentation in order to detect individual objects and then segment each of them. Previous approaches consider this task as end-to-end learning tasks [32], [33], by for example designing instance-aware FCNs [33] or by sequentially finding objects using a recurrent architecture [32]. Bounding boxes and the segmentation have also been optimized jointly and refined recursively in order to improve instance saliency segmentation results [35]. Another common approach is to consider this task as a multi-task learning problem [36], [34], [31]. This can, for instance, be done by predicting category-level confidences, instance numbers and the instance location using CNNs and then employing clustering methods in a merging step [36]. Alternatively, Dai et al. [34] propose a model consisting of three different network stages, one for differentiating instances using class-agnostic bounding boxes, one for pixel-wise mask estimation, and one for predicting a category label for each class. These stages are combined in a cascading manner, where the later stages not only share features with the earlier stages but also depend on the output of the previous stage. Mask-RCNN [37] is another recent approach, which adapts Faster-RCNN [10] to the instance segmentation task by adding a segmentation branch parallel to the object detection branch. It can be considered the current state-of-the-art for instance segmentation.




(a) Diamond-connectivity (b) Square-connectivity

Fig. 2: The different connectivity patterns used in this work. Unless explicitly stated, our main focus in this work will be on square-connectivity, where we for a given center pixel predict the neighboring pixels C1-C8.

## III. ConnNet

In this section, we describe our contribution ConnNet, which is a connectivity-based approach to salient segmentation. Section III-A introduces the general idea of predicting connectivity for the task of salient segmentation. Section III-B discusses how global relations are fused into the modeling of connectivity. Our general architecture for connectivity prediction is introduced in Section III-C. Finally, Section III-E illustrates how to use the predicted connectivity to achieve the final salient segmentation.

### A. Connectivity

Instead of phrasing the salient segmentation problem as semantic segmentation, we instead view it as a problem of finding connected regions. Finding connected components is a fundamental principle in image processing and many traditional methods such as connected component labeling and edge thinning rely on the modeling of connectivity [38]. In our work, we mainly limit ourselves to two different types of connectivity illustrated in Figure 2. These are 4-connectivity and 8-connectivity, which means that we predict, for each pixel, the neighboring four or eight pixels, respectively. For the 4-connectivity the city block distance is used as a metric, which is defined as $d_4(P,Q) = |(x-u)| + |(y-v)|$ for two pixels $P = (x, y)$ and $Q = (u, v)$ and will result in a diamond shape. Henceforth we will refer to it as diamond-connectivity. For the 8-connectivity instead a chessboard distance is used, $d_8(P,Q) = \max(|(x-u)|, |(y-v)|)$, resulting in a square shape, which we will refer to as square-connectivity.

Unless stated otherwise we are using square-connectivity in most experiments. This means, that given an input image, the prediction objective is to produce a $H \times W \times C$ connectivity cube, where $H$ and $W$ denote the height and the width of the input image, and $C$ denotes the number of neighboring pixels that are considered for a given pixel, i.e. $C = 8$ for square-connectivity. Two neighboring pixels are connected if both of them are salient pixels. By this criterion, all background

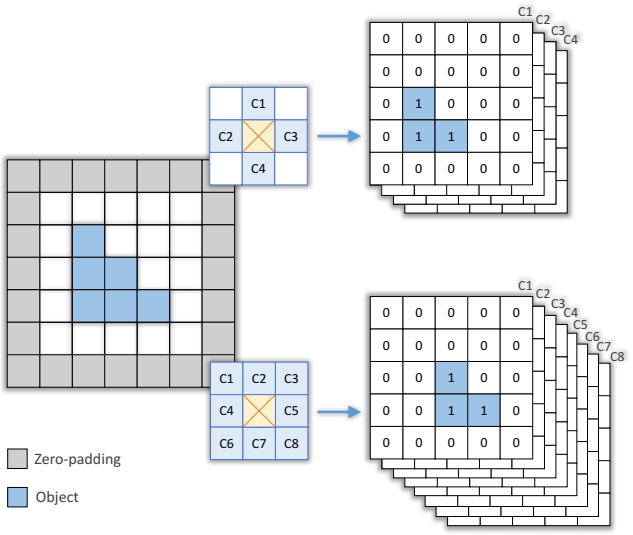



Fig. 3: Illustration of how diamond-connectivity (top) and square-connectivity (bottom) are modeled in the proposed method and how ground truth is generated. The original image ground truth is padded with background pixels and converted to connectivity cubes. For square-connectivity, the cube consists of eight $H \times W$ matrices, one for each C1-C8, assuming an input image of size $H \times W$. The cube for diamond-connectivity consists of four $H \times W$ matrices. For instance, the first slice in the square-connectivity cube indicates if pixels in the prediction mask are connected (both salient) to its top left neighbor. All pixels in the connectivity cube have binary values **0** indicating that pixels are not-connected and **1** for connected pixels.

pixels are not connected, which reduces the noise in the learning process. For connectivity cube $P$, $P_{i,j,c}$ represents the connectivity of a pixel with its neighbor at a specific position, where $i, j$ represents the spatial position of the pixel in the original image and $c$ represents the relative position of its neighbor. Provided the binary ground truth mask we generate a binary mask for each of the $C$ relative positions by checking if each pixel and its neighbor at the corresponding location are both salient. The $C$ binary masks are then stacked to produce the ground truth connectivity cube. Figure 3 illustrates this process using a small example.

We believe that decomposing the binary segmentation task into a connectivity prediction task provides several key advantages. Connectivity prediction can be viewed as a set of sub-problems of the segmentation task. In segmentation, the overall objective of grouping pixels into regions has to be achieved, while at the same time ensuring overall semantic rationality and region smoothing. For each of the connectivity sub-tasks, each task only focuses on grouping pixels in a specific direction. Introducing objectives based on pixel relations also allows the model to incorporate aspects that are commonly found in graphical models as it seamlessly integrates pair-wise relational inference into the feature learning process. This can lead to contours that are better preserved and also to less coarse salient segmentation results.

Due to the fact that connectivity is a symmetric measure we can further interpret our approach as a simple type of ensemble approach, where two neighboring pixels will predict the connectivity with respect to the other pixel. Additionally, we hypothesize that it can be viewed as a way to improve overall feature robustness, as it forces the network to learn features that are able to predict connectivity in various directions.

### B. Local and global relations

Our proposed connectivity framework for modeling local connectivity benefits from the inclusion of long-range relations in order to exploit global semantic context. Following [40] we make use of non-local blocks in our architecture to model long-range relations, effectively fusing global image context into the intermediate network feature representation. Non-local operations in the network for a given position $i$ can be defined as

$$y_i = \frac{1}{\sum_{\forall j} f(x_i, y_j)} \sum_{\forall j} f(x_i, x_j) g(x_j) \ , \tag{1}$$

where $g(x_j) = W_g x_j$ is a linear embedding, $\forall j$ accounts for all positions in the previous activation map and $f()$ represents a pairwise function that reflects the relationship between two locations. In our work we make use of an embedded Gaussian pairwise function, which has previously obtained good results for modelling non-local relations [40]. For two points $x_i$ and $x_j$ it is computed as

$$f(x_i, x_j) = e^{\Theta(x_i)^T \phi(x_j)} \ . \tag{2}$$

Here $\Theta(x_i) = W_\Theta x_i$ and $\phi(x_i) = W_\phi x_i$ represents linear embedding functions. The non-local operation fuses the local relations $x_i$ with the global relations $y_i$ as

$$z_i = W_z y_i + x_i \ . \tag{3}$$

The weight matrices $W_z$, $W_g$, $W_\Theta$ and $W_\phi$ are learned as part of the end-to-end training.

### C. Network architecture

This section introduces our network architecture for connectivity prediction. Utilizing a CNN for this task allows us to make use of high-level extracted features from the image to learn semantic-aware connectivity, allowing us to disentangle background regions effectively and incorporate object-level information. Our proposed approach can be used to adapt any semantic segmentation network to saliency segmentation. We, therefore, implement the modeling of the global and local relation based on two backbone CNN models, BlitzNet [39] and Feature Pyramid Network (FPN) [41]. The proposed network architecture is depicted in Figure 4. BlitzNet has shown its potential in real-time scene understanding and object detection, while FPN as a backbone for Mask RCNN [37] also shows its applicability in tasks like object detection and instance segmentation. The BlitzNet and FPN-based models shown in the paper all utilize the ImageNet [42] pretrained ResNet-50 [8]. According to practice in [40], one non-local block is

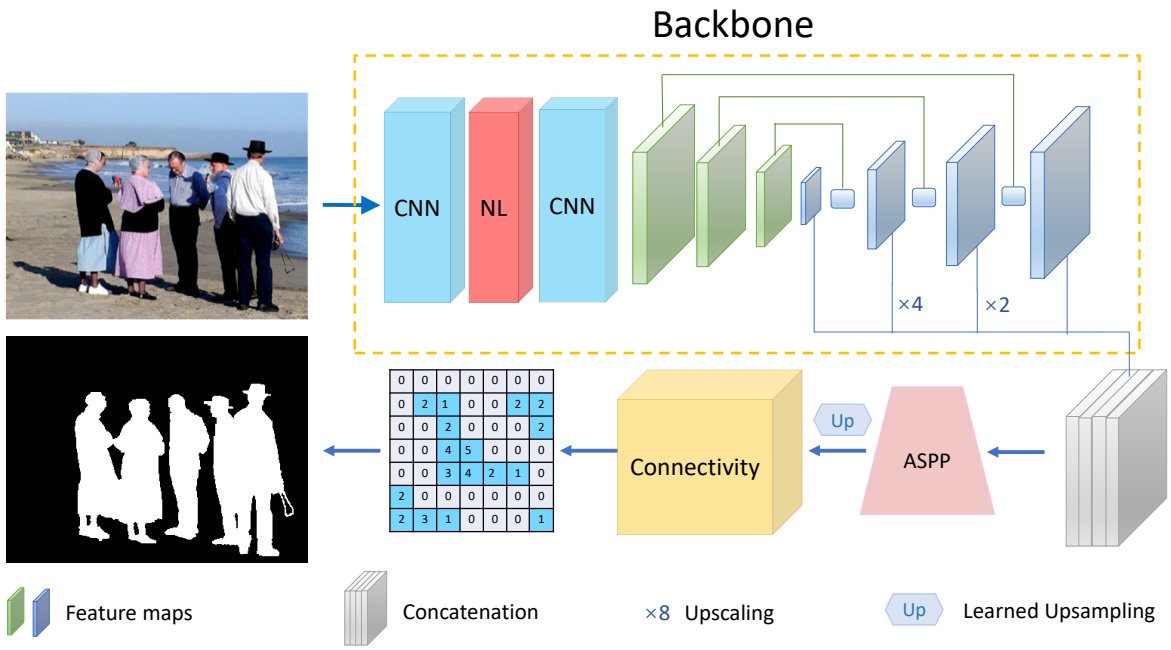



Fig. 4: Illustration of the network architecture of the proposed method. The backbone illustrated here is BlitzNet [39], however, it can be replaced with alternative architectures. Features are extracted using a CNN, in our case a ResNet-50 network, then processed by a down-scale stream (illustrated in green) and an upsampling stream (illustrated in blue). ResSkip blocks are utilized to incorporate higher resolution features from the down-scale stream during the upsampling. We incorporate multi-scale features with an ASPP module (optional) and upsample the connectivity cube to the original image dimensions using a deconvolution (fractionally strided convolution) and finally convert the cube to the salient object segmentation. A non-local block is inserted (NL).

inserted into the ResNet, right before the last Bottleneck unit of Block4, to model the global relation among pixels.

To effectively allow each pixel-pair to sense more local information in the image, we make use of DeepLabs' Atrous Spatial Pyramid Pooling (ASPP) [43]. ASPP allows the integration of multi-scale features by introducing multiple parallel atrous convolution filters at different dilation rates, which are individually processed before they are finally fused together again, allowing us to represent objects of a large variety in scale using local connectivity. This module is not required for modeling of connectivity and can therefore be considered optional. For the BlitzNet backbone, we utilize four $3 \times 3$ atrous convolutions with rates 6, 12, 18, and 24, each followed by two $1 \times 1$ convolutions to reduce the number of filters to the number of neighboring pixels $C$. We then fuse the representations by summing the results of the four different branches, yielding a connectivity cube with the same shape of the ground truth cube.

In the training phase, we utilize successive fractionally strided convolution layers with stride 2 to upsample the connectivity cube to the original input image size. During the inference phase, for images with arbitrary size, there is one more bilinear interpolation operation after the last deconvolution layer for each of the $C$ channels to restore the original resolution. A sigmoid function is applied to the connectivity cube in an element-wise manner to get the probability scores.

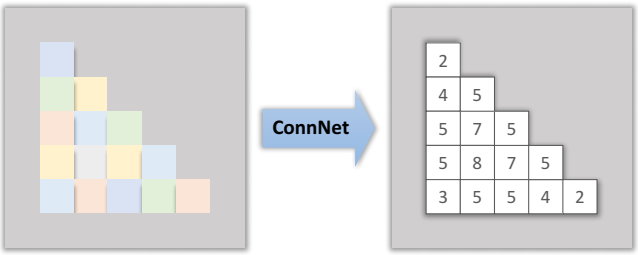



Fig. 5: The connectivity cube is converted to salient regions by counting the number of connected pixels. In the above case an ideal example is shown.

### D. Model Optimization

Training of the model is performed by optimizing the binary cross-entropy

$$L = \frac{1}{N \times C} \sum_{c=1}^{C} \sum_{i=1}^{N} [y_i^c \log \hat{y}_i^c + (1 - y_i^c) \log(1 - \hat{y}_i^c)] \quad (4)$$

where $N$ denotes the number of elements in a $H \times W$ slice of the connectivity cube and $C$ denotes the number of connected pixels that are considered. $y_i^c$ is the label indicating connectivity or non-connectivity for a given pixel in position $i$ with its neighbor pixel in location $c$, and $\hat{y}_i^c$ is the corresponding

TABLE I: Quantitative results for our proposed method (ConnNet) compared to other recent approaches. To illustrate the effect of including global relations, we use $CONN$ and $CONN+$ to denote ConnNet without and with global relations, respectively. We exclude results for the test results on the MSRA-B dataset for the RFCN and the DHSNet, as they were included in the respective training datasets following [30]. We report maximum F-measure (larger is better) and highlight the best three results for each dataset in the colors **orange**, **blue** and **green**, respectively.

| | | | | | | | | BlitzNet-backbone | | | FPN-backbone | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Data Set** | **MC** | **MDF** | **RFCN** | **DHSNet** | **DCL+** | **DSS** | **MSRNet** | **SEG** | **CONN** | **CONN+** | **SEG** | **CONN** | **CONN+** |
| **MSRA-B** | 89.4 | 88.5 | – | – | 91.6 | 92.7 | 93.0 | 91.9 | **93.2** | **93.3** | 90.5 | 91.8 | **93.1** |
| **HKU-IS** | 79.8 | 86.1 | 89.6 | 89.2 | 90.4 | 91.3 | 91.6 | 88.9 | **92.5** | **92.8** | 89.3 | 91.1 | **92.1** |
| **ECSSD** | 83.7 | 84.7 | 89.9 | 90.7 | 90.1 | 91.5 | 91.3 | 91.5 | **92.5** | **93.3** | 89.4 | 91.3 | **91.9** |
| **PASCAL-S** | 74.0 | 76.4 | 83.2 | 82.4 | 82.2 | 83.0 | **85.2** | 81.6 | 84.0 | **84.9** | 81.8 | 84.3 | **86.4** |

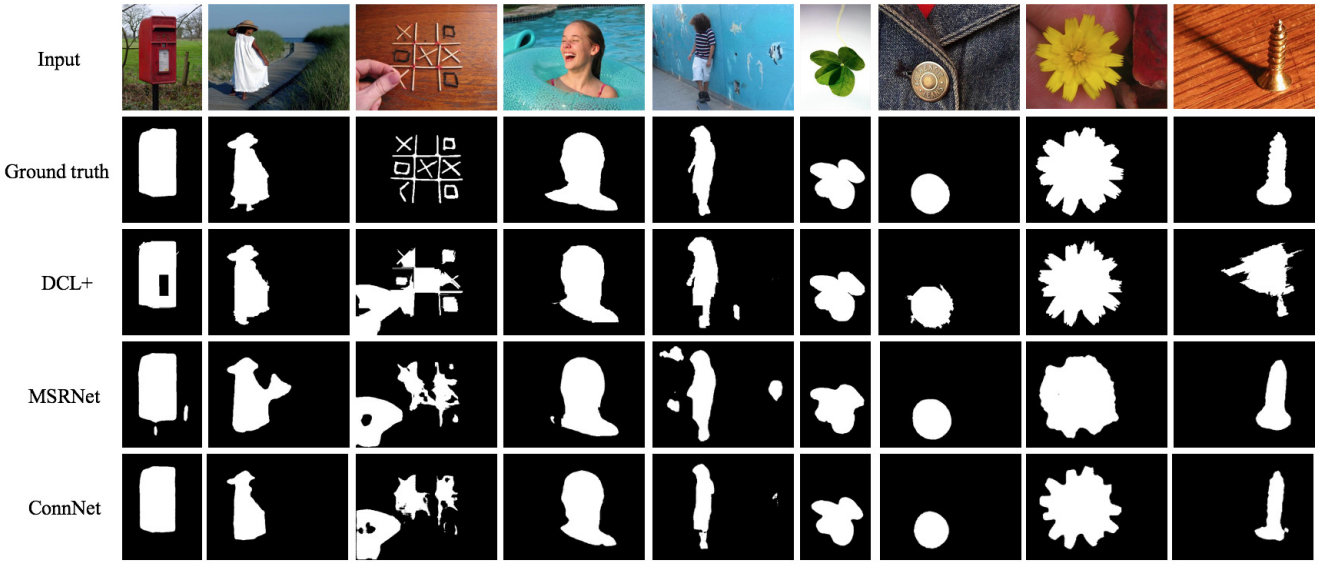



Fig. 6: Visual comparison of the saliency maps obtained by the proposed method, ConnNet+, and the highest performing methods in Table I. Reported ConnNet+ results are for BlizNet-backbone.

predicted output of the network. Note, $c$ is the relative position from a given pixel as illustrated in the masks in Figure 3.

### *E. Inference phase*

In the final step, given a $H \times W \times C$ connectivity cube, we need to reverse the ground truth generation process illustrated in Figure 3 to produce the salient mask. Connectivity is predicted for an element $\hat{y_i^c}$ if $\sigma(\hat{y_i^c}) > t$. Here $\sigma()$ corresponds to the sigmoid nonlinearity and $t$ is a threshold value, which is discussed in Section IV-A2. We require that predictions for two neighboring pixels should be in agreement with each other, i.e. assuming square-connectivity as illustrated in Figure 3, two neighboring pixels are connected if and only if the two corresponding entries in the slices of the connectivity cube agree. For example, given the predicted connectivity cube $P$ and assuming a threshold of 0.5, we predict $C5$ of a given pixel and $C4$ of its right-hand neighbor as connected if and only if $\sigma(P_{i,j,5}) > 0.5$ and $\sigma(P_{i,j+1,4}) > 0.5$. Salient pixels are then found by counting the number of connected pixels to a given pixel, allowing us to determine salient regions. This is illustrated in Figure 5. All operations can be performed efficiently using matrix operations, allowing for fast inference.

## IV. Experiments

We investigate the quantitative and qualitative improvements that are achieved by reformulating the problem as a connectivity task instead of a segmentation task. For this, we investigate two different tasks that require saliency segmentation, namely saliency object segmentation and the more recent task of instance-level saliency segmentation. In this work, we focus on evaluating the effectiveness of the connectivity cube to exploit pixel-connectivity to produce saliency masks. However, in future work, this could be extended to tasks such as semantic segmentation by adding additional output channels in order to capture class-wise connectivity.

### *A. Salient Object Segmentation*

*1) Implementation:* The proposed ConnNet is implemented in Tensorflow [44] and trained and tested on a GTX Titan X GPU. During training, we perform data augmentation by

TABLE II: Comparison to additional backbones. To illustrate that connectivity does not necessarily rely on pre-trained networks, the FCN network was randomly initialized. We observe that CONN still outperforms SEG after the specified number of epochs in Section IV-A1.

| | **FCN-backbone** | | **DeepLab-backbone** | | **BlitzNet-backbone** | | **FPN-backbone** | |
|---|---|---|---|---|---|---|---|---|
| **Data Set** | **SEG** | **CONN** | **SEG** | **CONN** | **SEG** | **CONN** | **SEG** | **CONN** |
| **MSRA-B** | 62.29 | 76.21 | 88.27 | 89.55 | 91.9 | 93.2 | 90.5 | 91.8 |
| **HKU-IS** | 59.52 | 75.65 | 84.19 | 88.69 | 88.9 | 92.5 | 89.3 | 91.1 |
| **ECSSD** | 55.89 | 73.56 | 82.19 | 87.52 | 91.5 | 92.5 | 89.4 | 91.3 |
| **PASCAL-S** | 49.59 | 72.38 | 76.68 | 78.59 | 81.6 | 84.0 | 81.8 | 84.3 |

randomly flipping the image horizontally, rescaling and random cropping. The weights of the ResNet50 network were initialized from a pre-trained model that has been trained on ImageNet [42]. We use Adam [45] and stage-wise training, where we initially only train the newly introduced layers, finetune the ResNet50 feature extractor and finally finetune the whole network end-to-end. The initial learning rate is $0.001$, and it is decreased to $0.00001$ through training. We run the model for 100K iterations, leading to an overall training time of fewer than 20 hours.

Due to the use of Fully Convolutional Neural Networks, inference is performed directly on the original image size. To increase inference robustness, in addition to the original image, we perform prediction for a horizontal flip of the image. Inspired by [30], we make the network more robust to multiple input scales, by rescaling both the flipped and the original images with five different factors ($0.5$, $0.75$, $1$, $1.25$, $1.5$), leading to a total of $10$ predictions for each test image. We then combine these predictions by averaging their connectivity predictions before we convert them into our salient object detection mask. Note, predictions for images with a scale factor different than $1$ are resized to the original image using bilinear interpolation. Inference for each image takes on average 0.03s for an image size of 320×320, resulting in a real-time prediction model.

*2) Evaluation:* We evaluate our performance on four benchmark datasets that are commonly used for the task of salient object segmentation. The datasets are MSRA-B [6], HKU-IS [26], PASCAL-S [25], and ECSSD [46]. Following [26], [13], [30], we perform training on the combined training sets of the MSRA-B and the HKU-IS datasets, which consists of $2500$ images each. Similarly, validation is performed on the combined validation sets of the two aforementioned datasets. Testing is performed on the test dataset for MSRA-B and HKU-IS, and on the combined training and test datasets for the others. This allows us to compare our method to previous state-of-the-art approaches, as well as allows us to illustrate the adaptability of the trained model to new datasets. We compare ConnNet to seven recent state-of-the-art approaches, namely, MC [18], MDF [26], RFCN [15], DHSNet [14], DCL+ [13], DSS [28], and MSRNet [30].

Performance is evaluated using the F-measure, which is defined according to [13] as

$$F_\beta = \frac{(1+\beta^2)\cdot \text{Precision}\cdot \text{Recall}}{\beta^2\cdot \text{Precision} + \text{Recall}}\,, \tag{5}$$

where $\beta^2$ is set to $0.3$, effectively up-weighing the impact of the precision more than the recall. We report F-measure as percentages. Similar to previous approaches such as DCL+ [13] and MSRNet [30], we introduce a threshold and report the maximum F-measure. However, unlike previous approaches we can not threshold the continuous saliency map directly, as our connectivity to salient map conversion produces a binary mask. Instead we threshold the continuous connectivity prediction $t$.

*3) Comparison to state-of-the-art:* Table I provides quantitative results for our method compared to previous state-of-the-art. ConnNet+ generally outperforms the existing methods or performs comparably on the benchmark datasets. Due to its simplicity, it also achieves good training and inference performance with respect to other state-of-the-art approaches. For instance, MSRNet requires $50$ hours of training time and inference takes $0.6$ seconds, compared to the less than $20$ hours of training time and $0.03$ seconds for inference in ConnNet+ with the BlitzNet-backbone. Further, MSRNet consists of a total of $82.9$ million trainable parameters compared to $71.0$ million for ConnNet+ with BlitzNet-backbone. Increasing the complexity of ConnNet+ further and introducing multi-scale modeling during training combined with attention will likely allow us to improve the results further at the cost of complexity.

A qualitative comparison of our method to the previous state-of-the-art method, MSRNet, can be seen in Figure 6. We observe that the behavior of the two methods is quite different, for instance in the first column we see an example where ConnNet+ outperforms MSRNet. The results for MSRNet contains a few isolated regions that do not correspond to the annotation. Since these regions are rather small and isolated, a graphical model as a post-processing step might have removed these, however, ConnNet+ instead is able to model these directly as the model integrates relationship prediction between pixels directly. Similarly, we observe in the second image that ConnNet+ is able to model fine details, while still not mistaking sharp edges, such as the road as part of the region. In the third image, we present an example where both methods perform poorly, as the image differs considerably from the general training data. Both models are able to segment out the hand holding the match which is not salient according to the ground truth, however, all struggle with the fine structure of the tic-tac-toe game.

*4) Segmentation vs. Connectivity:* To compare the overall improvement that we achieve by phrasing the salient object segmentation task as a connectivity task instead of a segmenta-


Fig. 7: Examples of salient mask results for the diamond-connectivity and the square-connectivity. We generally observe that square connectivity provides superior results and more complete salient masks. This agrees with our intuition that a connectivity-based approach on only four neighboring pixels might not be sufficient for the task of salient object segmentation.

TABLE III: Results for our ablation experiments. Here we analyze the effect of different connectivity patterns. Namely, we compare the diamond-connectivity and the square-connectivity, here abbreviated by the number of neighbors as $N_8$ and $N_4$, respectively. Further, we also extend the connectivity patterns to include $N_{12}$.

| **Data Set** | $N_4$ | $N_8$ | $N_{12}$ |
|---|---|---|---|
| **MSRA-B** | 90.9 | 93.1 | 93.3 |
| **HKU-IS** | 89.6 | 92.1 | 92.2 |
| **ECSSD** | 90.2 | 91.7 | 92.0 |
| **PASCAL-S** | 82.5 | 86.4 | 86.5 |

tion task, we also train our proposed ConnNet with a segmentation instead of the connectivity loss as a model variant. To enable us to do this we modify the final convolutional layer of the ConnNet to predict the background and the salient pixels. We present the results in Table I as SEG. It can be observed that our proposed method, based on connectivity, consistently outperforms the segmentation approach on all datasets and for both backbone architectures. This agrees with our intuition and is partly due to the fact that we divide the segmentation into sub-tasks, however, we additionally can view our approach as a small ensemble inside the network, as connectivity predictions have to agree for the final prediction to be correct. Further, the fact that a general feature representation needs to be learned to be able to predict connectivity to all neighboring pixels can effectively be viewed as model regularization. Note, that training time and test time for these two models are virtually identical, as the connectivity cube ground truth generation only adds a negligible overhead of $0.0105 \pm 0.0018$ seconds per batch to each training pass (standard deviation and mean reported over 1000 runs). A combined forward and backward pass during training for an identical batch of size 8 takes $0.2430 \pm 0.0047$. During inference the overhead of converting the connectivity cube back to the binary prediction mask makes up roughly one-third of the total inference time.

*5) Different backbone architectures:* To further illustrate the improvements that can be obtained by using connectivity compared to segmentation, we also evaluate two additional commonly used segmentation backbones, namely FCN [11] and DeepLab [43]. We again report both the results achieved by training the models using a common segmentation loss and compare it to a version where we employ the proposed connectivity approach. The results are illustrated in Table II and show that connectivity outperforms the segmentation approach for all datasets and all backbones. For completeness, we also include the results for the Blitznet-backbone and the FPN-backbone from Table I. Note, to illustrate that connectivity does not require a pre-trained network, no pre-trained weights were used for the FCN architecture.

*6) Ablation studies:*

**The effect of incorporating long-range relations.** The effect of introducing global relations can also be seen in Table I. Here, we use CONN+ to denote our full model that integrates the non-local block into intermediate layers of ConnNet to enable effective local and global relation fusion. By comparing

the full model CONN+ with ConnNet, we can observe that the incorporation of global relations improves the performance of our proposed connectivity network on all datasets, indicating that our connectivity model benefits from long-range pixel relations.

**The effect of connectivity structure.** To evaluate the proposed connectivity and to shed light on the effect of different connectivity types, we also investigate the use of diamond-connectivity ($N_4$) for the FPN backbone. This slightly reduces the number of parameters in the network by approximately 5000, a negligible amount when considering that the network has 71.0 million. Table III shows that square-connectivity ($N_8$) generally outperforms diamond connectivity by $1-3\%$ on all benchmark datasets. This is intuitive, as especially edges will be highly affected by the way connectivity is modeled resulting in potentially less defined edges for diamond-connectivity. When increasing the connectivity to the 12 nearest neighbors according to the city block distance ($N_{12}$), we observe diminishing returns. Small improvements at the cost of additional computational complexity.

Figure 7 displays examples of the salient segmentation masks obtained using the diamond- and the square-connectivity. The overall results obtained by the two approaches are similar, however, we note that for the square-connectivity the edges in the first image appear more well rounded similar to the ground truth. Also for the second image the tail of the kite is modeled more complete by the square-connectivity. Finally, we see in the third part that the diamond connectivity is not able to model the fine structure and lines in the tic-tac-toe game.

### *B. Instance Saliency Segmentation*

We also investigate the use of connectivity for the task of instance-level salient segmentation, a task recently proposed in [30]. Instead of just identifying salient regions, this task aims to identify object instances in these regions. As the idea of connectivity is applicable to a wide variety of models, we chose to focus on Mask R-CNN [37], a state-of-the-art model for instance-level semantic segmentation. Mask R-CNN extends Faster-RCNN [10] by introducing a segmentation branch for each region of interest. In this work, we replace this segmentation branch with a connectivity branch. Experiments are performed on the salient instance segmentation dataset provided by [30], which consists of 500 training, 200 validation, and 300 testing images, respectively. Results in Table IV illustrate that the modified Mask R-CNN outperforms the original formulation of the Mask R-CNN on the instance-level salient segmentation task and the instance-level salient segmentation MSRNet [30]. We use the mean Average Precision, $mAP^r$ [31] as our evaluation metric. Figure 8 shows some qualitative results. In general we observe similar results, however, for some examples we observe that the segmentation approach struggles to split or detect instances that our connectivity approach detects.

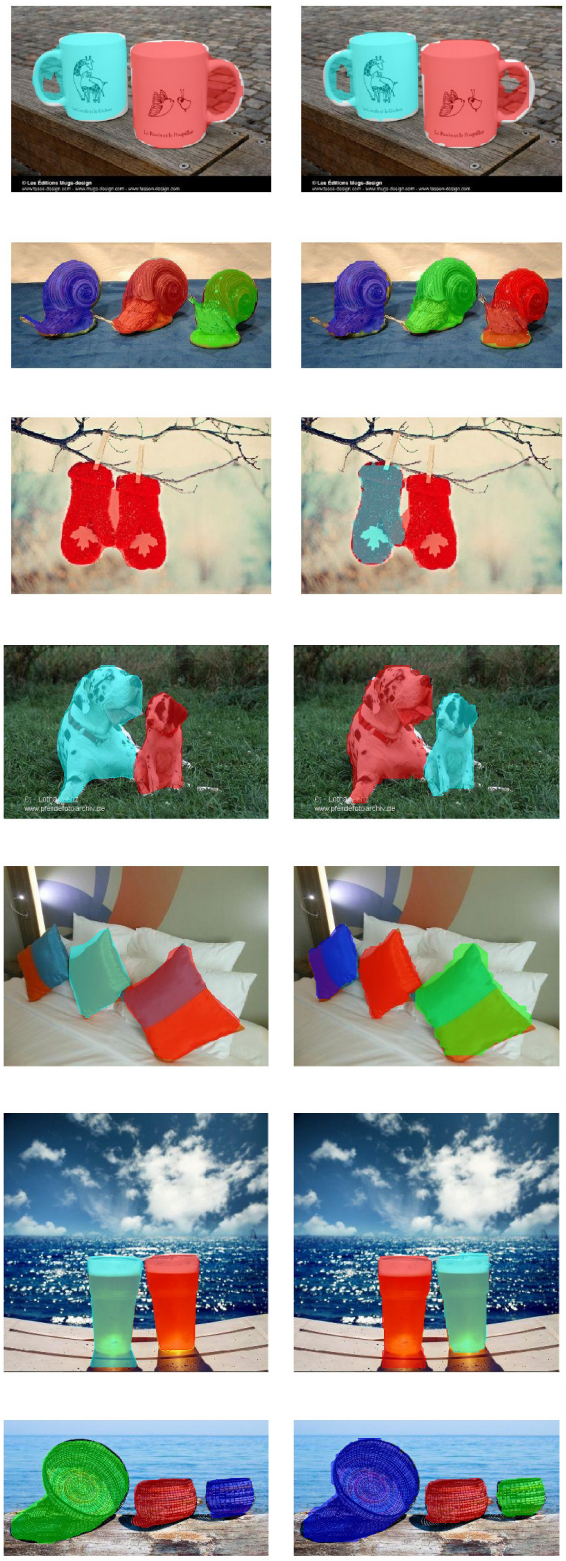


Fig. 8: Examples for instance saliency segmentation. The left side illustrates the results for Mask-RCNN and the right side the results for Mask-RCNN+CONN.

## V. Conclusion

In this paper, we present an approach to salient object and instance-level salient segmentation based on the idea of

TABLE IV: Results for the instance saliency segmentation experiment. We observe that Mask-RCNN+CONN outperforms the original formulation of Mask-RCNN.

| **Method** | $mAP^r$@0.5(%) |
|---|---|
| **MSRNet** | 65.32 |
| **Mask-RCNN** | 77.20 |
| **Mask-RCNN+CONN** | 81.06 |

connectivity modeling. The experimental results demonstrate that connectivity consistently outperforms segmentation on this task, confirming our intuition that it is beneficial to integrate relationship prediction between pixels into the salient segmentation model. We then show that a simple model based on connectivity can outperform more complex models trained on segmentation in both accuracy and speed. Finally, we perform ablation experiments to provide insights into the choice of connectivity. This work is a first step in the direction of connectivity-based salient segmentation, and we believe that more complex additions, such as conditional random fields, regularization via adversarial training, and attention based multi-resolution models will improve the overall performance further at the cost of overall complexity.

The current drawback of our proposed connectivity approach is that it does not generalize well to other tasks such as semantic segmentation as the naive approach of modeling class-wise connectivity does not scale well to large number of classes. In future work, we aim to explore ways of allowing the use of connectivity for these tasks in a more scalable manner.

## ACKNOWLEDGMENT

This work was partially funded by the Norwegian Research Council FRIPRO grant no. 239844 on developing the *Next Generation Learning Machines*.

## REFERENCES

[1] V. Navalpakkam and L. Itti, "An integrated model of top-down and bottom-up attention for optimizing detection speed," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2006, pp. 2049–2056.

[2] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li, "A user attention model for video summarization," in *Proceedings of the tenth ACM international conference on Multimedia*. ACM, 2002, pp. 533–542.

[3] Y. Liu, S. Zhang, M. Xu, and X. He, "Predicting salient face in multiple-face videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017, pp. 4420–4428.

[4] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] R. Valenti, N. Sebe, and T. Gevers, "Image saliency by isocentric curvedness and color," in *Proceedings of the IEEE international conference on computer vision*. IEEE, 2009, pp. 2185–2192.

[6] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 353–367, 2011.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Neural Information Processing Systems*, 2012.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016, pp. 770–778.

[9] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 3431–3440.

[12] X. Liang, H. Zhou, and E. Xing, "Dynamic-structured semantic propagation network," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[13] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016, pp. 478–487.

[14] N. Liu and J. Han, "Dhsnet: Deep hierarchical saliency network for salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016, pp. 678–686.

[15] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proceedings of the European Conference on Computer Vision*, 2016, pp. 825–841.

[16] J. Pan, C. Canton, K. McGuinness, N. E. O'Connor, J. Torres, E. Sayrol, and X. Giro-i Nieto, "Salgan: Visual saliency prediction with generative adversarial networks," *Conference on Computer Vision and Pattern Recognition Scene Understanding Workshop*, 2017.

[17] H. Pan and H. Jiang, "Supervised adversarial networks for image saliency detection," *arXiv preprint arXiv:1704.07242*, 2017.

[18] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 1265–1274.

[19] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 733–740.

[20] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 569–582, 2015.

[21] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 853–860.

[22] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 10, pp. 1915–1926, 2012.

[23] Y. Jia and M. Han, "Category-independent object-level saliency detection," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1761–1768.

[24] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *Proceedings of the IEEE international conference on computer vision*. IEEE, 2011, pp. 914–921.

[25] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 280–287.

[26] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 5455–5463.

[27] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 3183–3192.

[28] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," in *Proceedings of the IEEE international conference on computer vision*. IEEE, 2017, pp. 5300–5309.

[29] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1395–1403.

[30] G. Li, Y. Xie, L. Lin, and Y. Yu, "Instance-level salient object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

[31] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proceedings of the European Conference on Computer Vision*, 2014, pp. 297–312.

[32] B. Romera-Paredes and P. H. S. Torr, "Recurrent instance segmentation," in *Proceedings of the European Conference on Computer Vision*. Springer, 2016, pp. 312–329.

[33] J. Dai, K. He, Y. Li, S. Ren, and J. Sun, "Instance-sensitive fully convolutional networks," in *Proceedings of the European Conference on Computer Vision*. Springer, 2016, pp. 534–549.

[34] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.

[35] X. Liang, Y. Wei, X. Shen, Z. Jie, J. Feng, L. Lin, and S. Yan, "Reversible recursive instance-level object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 633–641.

[36] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan, "Proposal-free network for instance-level object segmentation," *arXiv preprint arXiv:1509.02636*, 2015.

[37] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*. IEEE, 2017, pp. 2980–2988.

[38] R. C. Gonzalez and R. E. Wood, "Digital image processing, 3rd edtn," 2007.

[39] N. Dvornik, K. Shmelkov, J. Mairal, and C. Schmid, "Blitznet: A real-time deep network for scene understanding," in *Proceedings of the IEEE international conference on computer vision*, 2017, p. 11.

[40] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," *Proceedings of the IEEE international conference on computer vision*, 2018.

[41] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, 2017.

[42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.

[43] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2017.

[44] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org/

[45] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

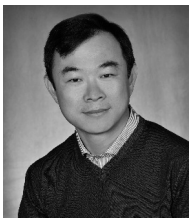
[46] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 1155–1162.


**Michael Kampffmeyer** received Masters degrees at the Physics and Technology (2014) and Computer Science Department (2015) at UiT The Arctic University of Norway, Tromsø, Norway, where he currently pursues a PhD in Machine Learning. His research interests include the development of unsupervised deep learning methods for representation learning and clustering by utilizing ideas from kernel machines and information theoretic learning. Further, he is interested in computer vision, especially related to remote sensing and health applications. His paper 'Deep Kernelized Autoencoders' with S. Løkse, F. M. Bianchi, R. Jenssen and L. Livi won the Best Student Paper Award at the Scandinavian Conference on Image Analysis, 2017. Since September 2017, he has been a Guest Researcher in the lab of Eric P. Xing at Carnegie Mellon University, Pittsburgh, PA, USA, where he will stay until July 2018. For more details visit https://sites.google.com/view/michaelkampffmeyer/.

**Nanqing Dong** is currently a research engineer at Petuum Inc., working under Prof. Eric P. Xing. He received his master degree from Cornell University in 2017. His research interests mainly include semantic segmentation, object detection and medical image analysis.

**Xiaodan Liang** is currently a project scientist in the Machine Learning Department at Carnegie Mellon University, working with Prof. Eric Xing. She received her PhD degree from Sun Yat-sen University in 2016, advised by Liang Lin. She has published several cutting-edge projects on the human-related analysis including the human parsing, pedestrian detection, instance segmentation, 2D/3D human pose estimation and activity recognition. Her research interests mainly include semantic segmentation, object/action recognition and medical image analysis.

**Yujia Zhang** is currently a PhD student in the State Key Laboratory of Management and Control for Complex Systems at the Institute of Automation, Chinese Academy of Sciences. She received her Bachelor degree at the Computer Science Department in Xi'an Jiaotong University. Her research interests are computer vision with a specific focus towards video summarization and deep learning. She is currently a visiting scholar in Eric P. Xing's group at the Machine Learning Department at Carnegie Mellon University.



**Eric P. Xing** is a Professor of Machine Learning in the School of Computer Science at Carnegie Mellon University, and the director of the CMU Center for Machine Learning and Health. His principal research interests lie in the development of machine learning and statistical methodology; especially for solving problems involving automated learning, reasoning, and decision-making in high-dimensional, multimodal, and dynamic possible worlds in social and biological systems. Professor Xing received a Ph.D. in Molecular Biology from Rutgers University, and another Ph.D. in Computer Science from UC Berkeley. His current work involves, 1) foundations of statistical learning, including theory and algorithms for estimating time/space varying-coefficient models, sparse structured input/output models, and nonparametric Bayesian models; 2) framework for parallel machine learning on big data with big model in distributed systems or in the cloud; 3) computational and statistical analysis of gene regulation, genetic variation, and disease associations; and 4) application of machine learning in social networks, natural language processing, and computer vision. He is an associate editor of the Annals of Applied Statistics (AOAS), the Journal of American Statistical Association (JASA), the IEEE Transaction of Pattern Analysis and Machine Intelligence (PAMI), the PLoS Journal of Computational Biology, and an Action Editor of the Machine Learning Journal (MLJ), the Journal of Machine Learning Research (JMLR). He is a member of the DARPA Information Science and Technology (ISAT) Advisory Group, and a Program Chair of ICML 2014.