Quasi-equilibrium Feature Pyramid Network for Salient Object Detection

Yue Song, Hao Tang, Mengyi Zhao, Nicu Sebe, and Wei Wang

***Abstract*—Modern saliency detection models are based on the encoder-decoder framework and they use different strategies to fuse the multi-level features between the encoder and decoder to boost representation power. Motivated by recent work in implicit modelling, we propose to introduce an implicit function to simulate the equilibrium state of the feature pyramid at infinite depths. We question the existence of the ideal equilibrium and thus propose a quasi-equilibrium model by taking the first-order derivative into the black-box root solver using Taylor expansion. It models more realistic convergence states and significantly improves the network performance. We also propose a differentiable edge extractor that directly extracts edges from the saliency masks. By optimizing the extracted edges, the generated saliency masks are naturally optimized on contour constraints and the non-deterministic predictions are removed. We evaluate the proposed methodology on five public datasets and extensive experiments show that our method achieves new state-of-the-art performances on six metrics across datasets.**

***Index Terms*—Salient Object Detection, Low-level Vision.**

I. Introduction

HUMAN Visual System (HVS) has the innate ability to detect salient objects out of visual scenes rapidly without training [1]. Salient Object Detection (SOD) aims at simulating HVS to detect distinct regions or objects, where people would orient their eye direction and fix on them [2], [3]. It has attracted much interest from research communities, mainly because it helps finding objects or regions that can represent a scene efficiently, a useful step in tasks like image segmentation [4], visual tracking [5], and image retrieval [6]. In the past decades, saliency detection models have evolved from traditional hand-crafted approaches via different saliency cues (*e.g.*, global contrast [7], background prior [8], and spectral analysis [9]) to Fully Convolutional Neural Networks (FCN) [10] based methods.

Even though FCN-based solutions [13], [14], [15], [16], [17], [18], [19], [12], [11], [20] have made remarkable progress so far, there still exist two main challenges: (i) most saliency detection models are based on the encoder-decoder framework, where the encoder extracts the multi-level representations from the image and the decoder takes as input the cross-scale features to generate the final saliency maps. The bottleneck of this framework is how to effectively fuse the multi-level representations of the feature pyramid between the encoder and the decoder. Different fusion strategies and connection pathways have been developed to aggregate multi-scale features for better representation [21], [15], [22], [23], [24], [25], [26], [3], [17], [27], [13], [28]. However, the connection topology of their architecture is often ad-hoc hand-crafted and complex, which needs careful design by human-beings and does not fully exploit the representation power of the feature pyramid. (ii) The contour information of the objects is hard to predict and is prone to be misclassified. Recent methods introduced boundary information into the model to improve the prediction performances [16], [17], [18], [29], [24], [30], [20]. These methods explored different approaches, such as introducing edge map as extra supervision [20], extracting object boundaries from the low-level feature maps [16], and applying dedicated loss function that focuses on the contours [17], to make the model learn the edge information and therefore become boundary-aware. By doing so, the contours of the object are expected to be preserved. Nonetheless, they all implicitly enforce contour constraints on the saliency masks, but none of them directly optimize the edges from the mask. Due to these two issues, existing methods might fail to generate accurate saliency maps with sharp boundaries and coherent details (see Figure 1).

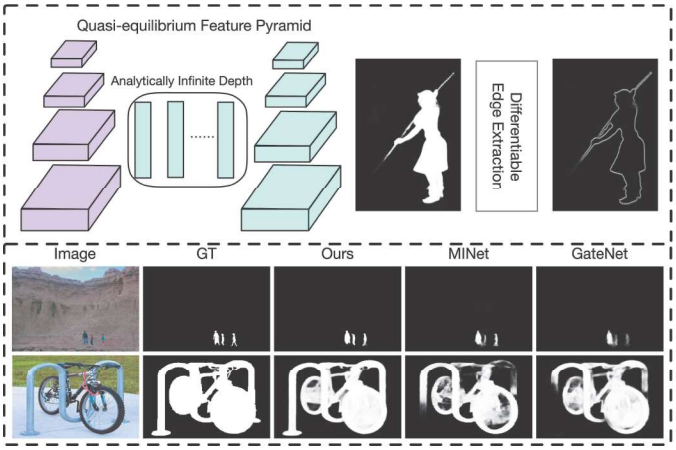

Fig. 1. (Top) Our method models a quasi-equilibrium feature pyramid at its infinite depth and use a differentiable edge extraction module that directly optimizes the edges of the mask. (Bottom) Qualitative comparison between our method and two recent state-of-the-art methods MINet [11] and GateNet [12]. Our approach can precisely segment objects of various sizes with subtle details.

This paper targets the aforementioned issues and proposes a novel solution. Recent work in implicit modeling [31], [32], [33] shows that by deriving the forward and backward formulations from the convergence states, any deep model can have analytically infinite depth with a single layer. This suggests a promising direction in increasing the receptive field and strengthening the representation power of the multi-level feature pyramid. Motivated by this, we introduce the

Manuscript received April 19, 2021; revised August 16, 2021.

implicit modelling method into the SOD framework to attain the implicit feature pyramid. From the empirical observation and analysis, we show that the convergence condition, *i.e.,* the so-called equilibrium state which requires the first-order derivative of hidden states to be zero, is almost impossible to be achieved in practice. The assumed equilibrium state happens practically only when the second-order derivative is equal to zero. We thus propose to relax the convergence condition by using Taylor expansion around the origin to approximate the equilibrium state of the hidden states. Our modification takes the non-zero first-order derivative into the root solver and leads to a quasi-equilibrium model. The infinite depth of the quasi-equilibrium simulates a more realistic convergence state of unrolling feature pyramid to infinite layers, which leads to a more informative feature pyramid compared to the ordinary equilibrium. Specific to SOD, the quasi-equilibrium reaches a status where the feature pyramid well combines the representations across different layers and contains all the desired information for the saliency map extraction. It has brought significant improvements in the SOD performances.

Moreover, we integrate the traditional edge detection algorithm into SOD methods and propose a differentiable edge extractor to directly optimize the edges that are extracted from the mask. Different from other edge-preservation mechanisms, our approach directly enforces contour constraints on the mask, and the structural information is naturally kept. The proposed edge extractor can also help the model to eliminate the non-deterministic values and generate more binary predictions. Our method has been applied in five benchmark datasets and achieves state-of-the-art results against the other 10 recent baselines on six widely used metrics.

Our contributions are summarized as follows:

- We empirically identify the non-existence of ideal equilibrium states for implicit models and propose a novel quasi-equilibrium feature pyramid network that models more realistic convergence states at infinite depths.
- We propose a novel differentiable edge extractor that directly optimizes the mask on edge constraints. Compared with existing methods, our approach can help the model to preserve contour information better and generate more deterministic predictions.
- Extensive experiments demonstrate that our method achieves state-of-the-art performances against other leading methods.

II. Related Work

A. Deep Saliency Detection

Early saliency detection methods in hand-engineered era mainly rely on various saliency cues, including global or local contrast [7], [34], background prior [8], and spectral analysis [9], [35]. Due to the page limits, the readers are kindly referred to [2] for a detailed review. Here we recap modern approaches in the deep learning era. These FCN-based methods can be broadly divided into two families:

Aggregation-based Methods. Most modern saliency detection models are based on the encoder-decoder framework to integrate multi-level features and leverage contextual information across different layers [21], [15], [22], [23], [24], [25], [26], [3], [17], [27], [13], [28], [19], [11]. During the past years, researchers have developed lots of feature fusion mechanisms and feature connection pathways for better representation. [13] added short connections to FCN and combined multi-layer features to generate accurate saliency maps. [14] proposed a hierarchical pixel-wise contextual attention network to learn the local and global context for each pixel. However, the aforementioned approaches depend on hand-crafted and complex connection pathways to leverage context information, and thus the representation power of the feature pyramid has not been fully explored. More recently, some vision transformer [36] based approaches have been proposed to tackle the challenge of SOD [37], [38], [39], [40]. Compared with convolutional networks, transformers allows for better modelling of global context [7], which is inherently suitable for SOD task.

Edge-guided Methods. In recent years, many approaches incorporate the edge/contour information to assist SOD task [27], [41], [16], [17], [18], [29], [24], [30], [20]. For instance, [21] proposed a boundary-aware IoU loss to preserve the sharp boundaries. [27] incorporated edge-aware feature maps in low-level layers to enhance the accuracy of the predicted boundary. [16] used edge label to supervise low-level feature maps to make the network pay attention to the edges. [17] designed a hybrid loss for boundary-aware saliency predictions. These methods enforce the network to learn the edge information and implicitly pose contour constraints on the saliency mask. However, none of them directly optimize the contours of the generated saliency maps. Motivated by the traditional edge detection algorithm, we have proposed a differentiable edge extractor to explicitly extract edges from the generated mask. By optimizing the generated edges, the boundary information of the mask is naturally kept. Moreover, optimizing the edge maps can remove the non-binary values of the saliency mask and make the model generate more deterministic predictions.

B. Implicit Deep Learning

One of the key concepts in modern deep learning is the computational graph, which is explicitly created by the trajectory of the forward pass and allows the error to back-propagate in the reverse order. Implicit methods, on the other hand, do not necessarily have such prescribed computational graphs. Instead, they pose specific criterion for the models (*e.g.,* the endpoint for neural ODE [42] and the root for non-linear equations [31]). It brings a superior benefit where the forward pass can rely on any black-box solvers, while the analytically backward gradient is independent of the forward trajectory. In the past decade, implicit methods are extensively explored to model the hidden states of deep models [42], [43], [44], [31], [32], [33]. For example, [45], [46] proposed recurrent back-propagation to train the network by implicit differentiation. Neural ODE [42], [43] adopted black-box ODE solvers to implicitly model the residual block. DEQ [31] and MDEQ [32] used fixed-point iteration and Broyden's method [47] to simulate infinite-depth networks for modelling sequential data and vision recognition domain, respectively.

[33] applied MDEQ [32] method in feature pyramid modelling for object recognition. Similar to [33], [32], we also simulate an infinite-depth network for deep multi-scale feature fusion. Differently, we question the existence of ideal equilibrium and relax the convergence condition by approximating the linear tangent neighborhood. This modification considers the non-zero first-order derivative in the black-box root solver and this leads to a more practical fixed-point iteration through the quasi-equilibrium states.

III. Methodology

A. Deep Equilibrium Model Revisited

For any input-injected deep sequence model, the i-th layer can be formulated as:

$$\mathbf{z}_{i+1} = f_\theta^i(\mathbf{z}_i; \mathbf{x}),\ i = 0, 1, \ldots, L-1, \tag{1}$$

where L denotes the network depth, $\mathbf{z}_i$ is the hidden sequence of layer i, and f_θ^i is the non-linear transformation, respectively. Recent work shows that employing the same transformation for each layer ($f_\theta^i = f_\theta, \forall i$) can still achieve competitive performances [48], [44], [49]. Stacking such layer to infinite depths ($L \to \infty$) makes the output tend to converge to an ideal equilibrium:

$$\mathbf{z}^\star = f_\theta(\mathbf{z}^\star; \mathbf{x}). \tag{2}$$

[31], [32] proposed to directly solve for the fixed point instead of iterating through layers, and the formulation can be cast as a root-finding problem:

$$g_\theta(\mathbf{z}^\star; \mathbf{x}) := f_\theta(\mathbf{z}^\star; \mathbf{x}) - \mathbf{z}^\star,\ \mathbf{z}_0 = 0. \tag{3}$$

The forward pass can rely on any black-box root solver (*e.g.*, Newton or quasi-Newton methods) and the back-propagation can be conducted through the equilibrium state using the Jacobian matrix of g_θ at $\mathbf{z}^\star$. [31], [32] further proposed to use Broyden's method [47] to approximate the inverse Jacobian for efficient forward and backward propagation. Formally, the forward pass of DEQ is defined as:

$$\mathbf{z}_{i+1} = \mathbf{z}_i - \alpha B g_\theta(\mathbf{z}_i, \mathbf{x}), \tag{4}$$

where α is the step size, and B is the low-rank approximation of the Jacobian inverse $J_{g_\theta}^{-1}|\mathbf{z}_i$. Given the loss function l propagated to $\mathbf{z}^\star$, the backward gradients w.r.t θ and $\mathbf{x}$ can be calculated using the chain rule:

$$\begin{aligned} \frac{\partial l}{\partial \theta} &= \frac{\partial l}{\partial \mathbf{z}^\star}(-J_{g_\theta}^{-1}|\mathbf{z}^\star)\frac{\partial f_\theta(\mathbf{z}^\star; \mathbf{x})}{\partial \theta}; \\ \frac{\partial l}{\partial \mathbf{x}} &= \frac{\partial l}{\partial \mathbf{z}^\star}(-J_{g_\theta}^{-1}|\mathbf{z}^\star)\frac{\partial f_\theta(\mathbf{z}^\star; \mathbf{x})}{\partial \mathbf{x}}. \end{aligned} \tag{5}$$

As the Jacobian of g_θ is expensive to compute, [31] suggested using Broyden's method [47] again and solve for a linear equation involving a vector-Jacobian product:

$$\mathbf{x}(J_{g_\theta}|\mathbf{z}^\star) + \frac{\partial l}{\partial \mathbf{z}^\star} = 0. \tag{6}$$

In principle, the network could have analytically infinite depths attained by unrolling a fixed number of layers. For the detailed derivation and proof, please refer to [31], [32].

B. Deep Quasi-equilibrium Model

Although DEQ [31] and MDEQ [32] have presented a promising analytically infinite-depth network that are derived from the equilibrium states, there is no convincing argument that guarantees such ideal convergence. The analytical ideal equilibrium requires that at a certain layer, the hidden states no longer change versus depth ($\mathbf{z}^\star = \mathbf{z}_i = \mathbf{z}_{i+1} = \cdots$) and the first-order derivative of the input sequences are zero ($||\mathbf{z}_{i+1} - \mathbf{z}_i|| = 0$). Consider the highly non-linear transformation f_θ of each layer and the large size of hidden states $\mathbf{z}_i$ (0.6M elements for each image), it is almost impossible to reach the fixed point $\mathbf{z}^\star$.

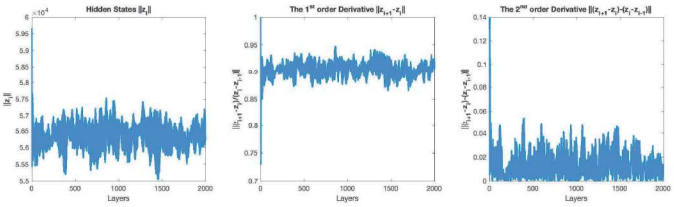

Fig. 2. The hidden states $\mathbf{z}_i$, their 1st order derivative, and the 2nd order derivative. For simplicity concern, we only compute the numerator of the derivative here. The 1st order derivative that is supposed to be zero at the equilibrium states turns out to fluctuate around a non-zero value. When the hidden states have small variation, the 2nd order derivative is more likely to be zero.

To investigate the empirical existence of the ideal equilibrium, we simulate the hidden states $\mathbf{z}_i$ by unrolling $5,000$ weight-tied layers and compute their first-order derivative $||\mathbf{z}_{i+1} - \mathbf{z}_i||$ and the second-order derivative $||(\mathbf{z}_{i+1} - \mathbf{z}_i) - (\mathbf{z}_i - \mathbf{z}_{i-1})||$. Note that here we take the numerator of the derivative for simplicity concern, as we only care if they can reach zero at some layers. As can be seen from Figure 2, the hidden states and their first-order derivative still fluctuate greatly even after thousands of layers. That being said, the analytical fixed point where the first-order derivative equals to zero cannot be reached, and the root for Eq. (3) does not exist in practice. Nonetheless, the second-order derivative is very likely to be or close to zero at deep layers when the hidden states have small variation. This phenomenon implies that when the assumed convergence practically happens, the hidden states in adjacent layers change with linear variation and therefore can be linearly approximated in the close neighborhood.

To simulate a more realistic equilibrium, we propose to relax the convergence condition by Taylor expansion (*i.e.*, applying tangent line approximation) to seek for the quasi-equilibrium states where the second-order derivative of hidden states are zero. Suppose at certain layer $f_\theta(\mathbf{z}^\star; \mathbf{x})$ varies little and reaches such quasi-equilibrium, for any point $\hat{\mathbf{z}}$ close to $\mathbf{z}^\star$, we should have the following approximation:

$$f_\theta(\mathbf{z}^\star; \mathbf{x}) - f_\theta(\hat{\mathbf{z}}; \mathbf{x}) = (\mathbf{z}^\star - \hat{\mathbf{z}})\frac{\partial f_\theta(\hat{\mathbf{z}}; \mathbf{x})}{\partial \hat{\mathbf{z}}}. \tag{7}$$

Here it is a Taylor expansion of function f_θ at $\hat{\mathbf{z}}$. As the second-order derivative is zero, the transformation f_θ can be linearly approximated around $\mathbf{z}^\star$. Thus, the equality generally holds when the $||\hat{\mathbf{z}} - \mathbf{z}^\star||$ is sufficiently small. The first-order

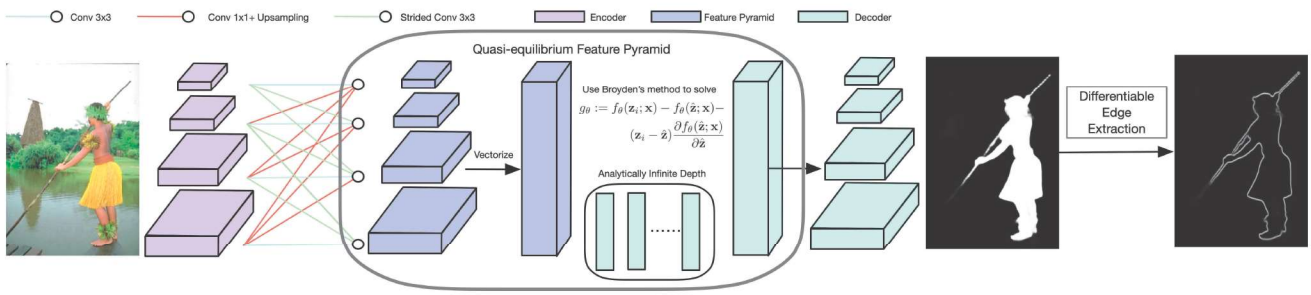

Fig. 3. Overview of our proposed model. The left encoder extracts the multi-scale feature maps from the image and enriches the representation power of each level by cross-scale and fully-connected pathways. Subsequently, our proposed quasi-equilibrium feature pyramid absorbs the features and reshapes them into a vector. Broyden's method is used to simulate the final quasi-equilibrium states of the feature vector at infinite depths. Then we feed the produced multi-level feature at the quasi-equilibrium into the decoder to generate the final saliency map. A differentiable edge extractor is further proposed to extract the object contours. By optimizing the extracted contours, the edge constraints are enforced on the mask and the non-deterministic predictions are removed.

derivative is taken into consideration to simulate more realistic equilibrium states. To derive $\hat{\mathbf{z}}$ in the close neighborhood of $\mathbf{z}^\star$, we can perturb $\mathbf{z}^\star$ by setting a small fraction of elements to 0:

$$\hat{\mathbf{z}} = \mathbf{z}^\star \cdot \mathcal{B}(p), \tag{8}$$

where $\mathcal{B}(\cdot)$ is Bernoulli distribution and p indicates the probability of being 0. Then $f_\theta(\mathbf{z}^\star;\mathbf{x})$ can be expressed using Eq. (7) after some rearrangements:

$$f_\theta(\mathbf{z}^\star;\mathbf{x}) = f_\theta(\hat{\mathbf{z}};\mathbf{x}) + (\mathbf{z}^\star - \hat{\mathbf{z}})\frac{\partial f_\theta(\hat{\mathbf{z}};\mathbf{x})}{\partial \hat{\mathbf{z}}}. \tag{9}$$

Injecting Eq. (9) into Eq. (3) leads to the re-formulation of root-finding equation:

$$g_\theta := f_\theta(\mathbf{z}^\star;\mathbf{x}) - f_\theta(\hat{\mathbf{z}};\mathbf{x}) - (\mathbf{z}^\star - \hat{\mathbf{z}})\frac{\partial f_\theta(\hat{\mathbf{z}};\mathbf{x})}{\partial \hat{\mathbf{z}}}. \tag{10}$$

We use Broyden's method to find the solution of this equation as the forward pass. Since f_θ is unfortunately not twice differentiable, Eq. (9) cannot be applied in the backward solver defined in Eq. (6). During back-propagation, we still use Eq. (3) to compute the analytical gradients.

MDEQ [32] proposed to use Variational Dropout [50] on $\mathbf{z}^\star$ to simulate the subtle differences between $\mathbf{z}^\star$ and $f_\theta(\mathbf{z}^\star;\mathbf{x})$. However, the simulation is not involved into the root-finding solver but is inserted after obtaining the root $\mathbf{z}^\star$, which implicitly requires that the transformation f_θ is a Bernoulli distribution when close to the equilibrium. This assumption has no theoretical basis or empirical justification. By contrast, our proposed quasi-equilibrium model is based on the experimental observation and directly involves $f_\theta(\mathbf{z}^\star;\mathbf{x})$ in the black-root solver.

C. Model Architecture

Figure 3 displays the overview of our proposed model. The left encoder passes the multi-level feature maps to the implicit feature pyramid in a fully-connected manner. The implicit feature pyramid converts the cross-scale feature maps into a large vector and uses Broyden's method to seek for the quasi-equilibrium state at infinite depths. Finally, the decoder takes in the multi-level features and predicts the final saliency maps.

Encoder. The encoder is fed with the input image and outputs the multi-level representations at five levels $\{F_i | i{=}1,\ldots,5\}$, each of which has different receptive fields. As discussed in [26], the lowest feature map F_1 has too much redundant information, which increases large computational burdens but brings little performance improvement. Similar to most methods, we discard it and only pass the rest feature maps to subsequent modules.

Quasi-equilibrium Feature Pyramid. To leverage the multi-level context information, our implicit feature pyramid first builds the connection pathway in a fully-connected manner. The representation at each scale is augmented by features from the other levels. Specifically for fusing representation in different resolutions, features in coarser scales are processed by $3{\times}3$ strided convolution to reduce the spatial dimension, whereas we perform successively $1{\times}1$ convolution and upsampling on features in finer scales to increase the resolution. After the fully-connected pathway, the feature pyramid owns stronger representation power. We then convert the feature pyramid into a vector for root finding:

$$\mathbf{x} = \mathrm{concat}(\mathrm{vec}(F_2), \mathrm{vec}(F_3), \mathrm{vec}(F_4), \mathrm{vec}(F_5)), \tag{11}$$

where $\mathrm{vec}(\cdot)$ denotes the vectorization, and $\mathrm{concat}(\cdot)$ is the concatenation of the vectors. The quasi-equilibrium states $\mathbf{z}^\star$ are initialized with zero. We use Broyden's method to solve the root for Eq. (10) as the forward pass and the solution for Eq. (6) as the backward propagation.

Decoder. We pass the resultant hidden states $\mathbf{z}^\star$ at analytically infinite depths of the implicit feature pyramid to the decoder. The decoder takes in the multi-level feature map and outputs the final saliency mask.

D. Differentiable Edge Extractor

Motivated by classical edge detection procedure, we propose to directly optimize the saliency mask by extracting their edges using the pre-defined Sobel edge operator. As shown in Fig. 4, the vertical and horizontal Sobel operators are defined as:

$$S_x = \begin{Bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{Bmatrix}, \quad S_y = \begin{Bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{Bmatrix}. \tag{12}$$

The horizontal and vertical image gradients are computed as the convolution of the edge filter and the image:

$$G_x = \mathrm{Conv}(I, S_x), \;\; G_y = \mathrm{Conv}(I, S_y). \tag{13}$$

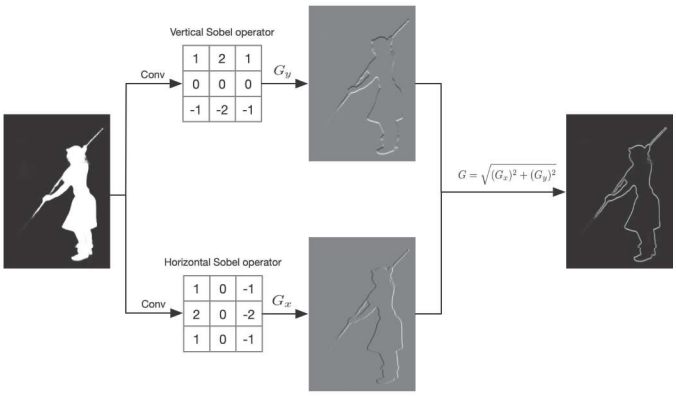

Fig. 4. Visualization of our proposed differentiable edge extraction. The Sobel filters are applied on the image to compute the vertical and horizontal image gradients. The edge map is obtained by combining the gradient magnitudes in two directions.

where I denotes the image. The parameters of both edge filters are fixed to maintain their functions of edge detection. Then we combine their magnitudes in two directions by:

$$G = \sqrt{(G_x)^2 + (G_y)^2}. \tag{14}$$

Traditional edge detection algorithm still performs non-max suppression and thresholding after combing the magnitudes. As both the ideal mask and ground truth are binary in our case, it is sufficient to get the combination of magnitudes. The whole process is differentiable and we can implement it using any language with AutoGrad package (*e.g.,* PyTorch). Optimization on the extracted edges would propagate the gradients back to the mask, and this is equivalent to optimizing the contour information of the mask.

In addition to the explicit edge optimization, the proposed edge extractor can bring another advantage, which is to help the model to eliminate the non-deterministic predictions (*i.e.,* non-binary values between 0 and 1). As the function of edge filter is to identify any discontinuities in the mask, the non-deterministic predictions would trigger discontinuities in the neighborhood. These points would be directly indicated on the extracted edge maps. Since an ideal mask such as the ground truth would only have discontinuities exactly on the object boundary, optimizing the edges using the edge map extracted from the ground truth would remove the non-binary values and make the network generate more deterministic predictions. Note that the edge operators defined in Eq. (12) can be substituted with any other edge detection filters.

E. Loss Function

Our loss function l is defined as a composition of loss functions on the mask and on the edge:

$$l = l_{CE}(G(S), G(E)) + l_{CE}(S, E) + l_{IoU}(S, E), \tag{15}$$

where S indicates the saliency mask, E represents the ground truth, and $G(\cdot)$ is the edge extractor. The first term calculates the cross-entropy loss of the edges extracted from the saliency mask and the ground truth, which directly enforces edge constraints on the saliency mask. The second term is the cross-entropy loss of the saliency mask and the ground truth. The third term computes the Intersection over Union (IoU) loss to ensure the detected objects overlap with the ground truth.

IV. Experiments

Datasets. We follow [11], [12] and conduct extensive experiments on five widely used benchmark datasets to evaluate the proposed method, *i.e.*, ECSSD [51], PASCAL-S [52], DUT-OMRON [8], HKU-IS [53], and DUTS [54].

Evaluation Metrics. We use six widely used metrics to evaluate the proposed method, *i.e.*, Mean Absolute Error (MAE) [55], mean F-measure (m F_β) [56], weighted F-measure (F_β^ω) [57], Max F-measure (Max F_β), S measure (S_m) [58], and precision-recall curve. More details about the datasets and the evaluation metrics can be found in the supplementary material.

Implementation Details. In line with most existing methods [17], [19], [12], [11], we use the DUTS-TR dataset for training and the rest of the datasets as the test set for evaluation. ResNet-50 [59] classifier pre-trained on ImageNet [60] is used as backbone to initialize the model, and the other parameters are randomly initialized. Our network is trained end-to-end for 50 epochs with a mini-batch size of 32 by stochastic gradient descent (SGD). The momentum and weight decay are set to 0.9 and 0.0005, respectively. We set the maximum learning rate to 0.005 for the ResNet-50 backbone and 0.05 for the other parts. Warm-up and linear decay strategies are also used. During training, we use random horizontal flip, random crop, and multi-scale input images for data augmentation. The probability of being 0 of Bernoulli distribution to perturb the hidden states $\mathbf{z}^\star$ is set as $5e{-}7$. We warm up the feature pyramid by unrolling 5 weight-tied layers for the first $3,000$ iteration and then switch to our proposed quasi-equilibrium root solver. The images are resized to the resolution of $352{\times}352$ during testing and fed into the network to generate the saliency prediction without any post-processing step. Resizing with bilinear interpolation is consistently used throughout all the experiments.

A. State-of-the-Art Comparisons

We demonstrate the effectiveness of our model by comparing with 10 other state-of-the-art models, including BMPM [15], CPD [26], EGNet [16], BANet [18], BASNet [17], SCRN [30], PoolNet [29], F3Net [19], GateNet [12], MINet [11], VTS [37] and GVT [38]. To assure comparison fairness, the saliency maps are all provided by the authors and evaluated using the same set of codes implemented by [19].

Quantitative Evaluation. Table I displays the performances of aforementioned methods on five datasets. Our method consistently outperforms other models and achieves the best performances across datasets, refreshing the leaderboard and setting the new baseline. In particular, we have greatly improved the mean F-score (m F_β) on all datasets, with a 0.9% increase on ECSSD, 3.0% gain on DUTS-TE, 1.6% increase on DUT-OMRON, 1.3% improvement on PASCAL-S, 1.2% gain on HKU-IS. Significant improvements are also observed in weighted F-score (F_β^ω).

This article has been accepted for publication in IEEE Transactions on Image Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIP.2022.3220058

TABLE I
QUANTITATIVE RESULTS COMPARED WITH STATE-OF-THE-ART METHODS ON FIVE DATASETS WITH FIVE METRICS. FOR ALL METRICS EXCEPT FOR MAE, HIGHER IS BETTER. ALL THE METHODS USE RESNET-50 AS THE SAME BACKBONE. THE BEST THREE RESULTS ARE HIGHLIGHTED IN RED, BLUE, AND GREEN RESPECTIVELY. THE VALUE IN THE BRACKET DENOTES THE TOTAL NUMBER OF IMAGES IN THE DATASET.

Method$_{year}$	ECSSD (# 1,000)					DUTS-TE (# 5,019)					DUT-OMRON (# 5,168)					PASCAL-S (# 850)					HKU-IS (# 4,447)				
	MAE	m F_β	F_β^ω	Max F_β	S_m	MAE	m F_β	F_β^ω	Max F_β	S_m	MAE	m F_β	F_β^ω	Max F_β	S_m	MAE	m F_β	F_β^ω	Max F_β	S_m	MAE	m F_β	F_β^ω	Max F_β	S_m
BMPM$_{2018}$ [15]	0.045	0.868	0.871	0.928	0.911	0.049	0.745	0.761	0.852	0.862	0.064	0.692	0.681	0.774	0.809	0.076	0.769	0.782	0.862	0.842	0.039	0.871	0.859	0.921	0.907
CPD-R$_{2019}$ [26]	0.037	0.917	0.898	0.939	0.918	0.043	0.805	0.795	0.865	0.869	0.056	0.747	0.719	0.797	0.825	0.074	0.829	0.800	0.870	0.844	0.034	0.891	0.875	0.925	0.905
EGNet-R$_{2019}$ [16]	0.037	0.920	0.903	0.947	0.925	0.039	0.815	0.816	0.889	0.887	0.053	0.756	0.738	0.815	0.841	0.075	0.831	0.807	0.878	0.853	0.031	0.901	0.887	0.935	0.918
BANet$_{2019}$ [18]	0.035	0.923	0.908	0.945	0.924	0.040	0.815	0.811	0.872	0.879	0.059	0.746	0.736	0.803	0.832	0.070	0.838	0.817	0.879	0.853	0.032	0.899	0.887	0.930	0.913
BASNet$_{2019}$ [17]	0.037	0.880	0.904	0.942	0.834	0.048	0.791	0.803	0.860	0.866	0.056	0.756	0.751	0.805	0.836	0.079	0.777	0.797	0.860	0.834	0.032	0.895	0.889	0.928	0.909
SCRN$_{2019}$ [30]	0.037	0.918	0.899	0.950	0.927	0.040	0.809	0.803	0.888	0.885	0.056	0.746	0.720	0.811	0.837	0.065	0.839	0.816	0.890	0.867	0.033	0.897	0.878	0.935	0.917
PoolNet$_{2019}$ [29]	0.035	0.919	0.904	0.949	0.925	0.037	0.819	0.817	0.889	0.887	0.054	0.752	0.725	0.805	0.831	0.067	0.838	0.819	0.885	0.864	0.030	0.903	0.888	0.936	0.919
F3Net$_{2020}$ [19]	0.033	0.925	0.912	0.945	0.924	0.035	0.791	0.835	0.891	0.888	0.053	0.766	0.747	0.813	0.838	0.064	0.844	0.823	0.882	0.855	0.028	0.910	0.900	0.937	0.917
GateNet$_{2020}$ [12]	0.040	0.916	0.894	0.945	0.924	0.040	0.807	0.809	0.888	0.885	0.055	0.746	0.729	0.818	0.838	0.071	0.830	0.804	0.881	0.854	0.033	0.899	0.880	0.933	0.915
MINet$_{2020}$ [11]	0.033	0.924	0.911	0.947	0.925	0.037	0.828	0.825	0.884	0.884	0.055	0.756	0.738	0.810	0.833	0.064	0.842	0.821	0.882	0.857	0.028	0.908	0.899	0.935	0.920
Ours	0.031	0.932	0.920	0.951	0.930	0.035	0.853	0.849	0.896	0.895	0.057	0.778	0.762	0.825	0.845	0.062	0.855	0.838	0.890	0.867	0.026	0.921	0.913	0.944	0.927

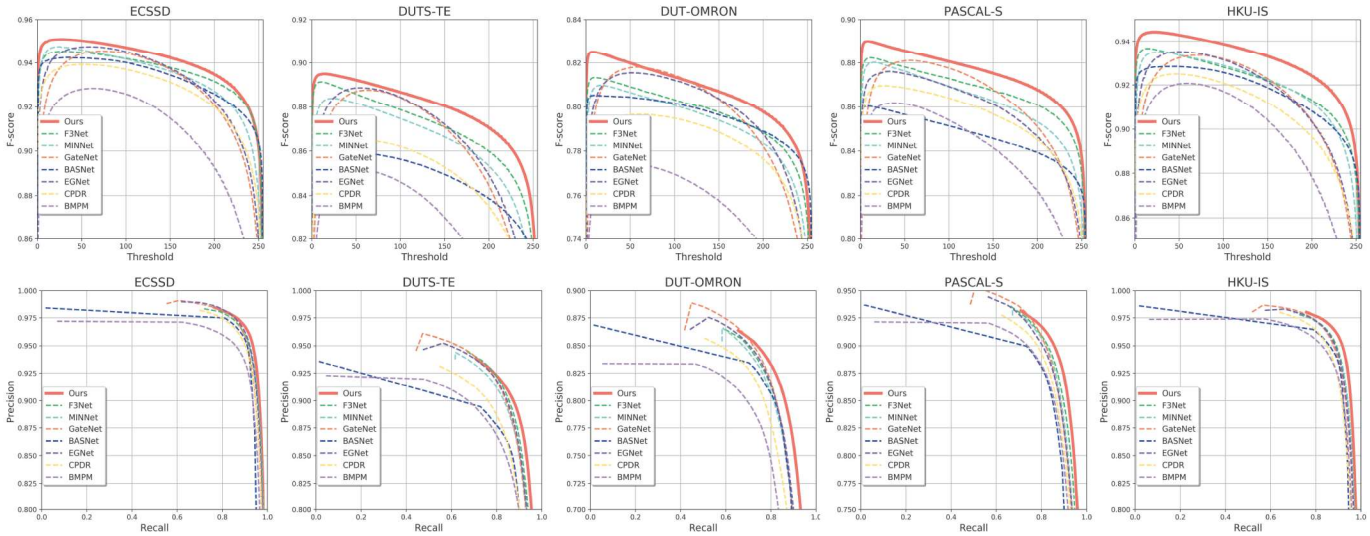

Fig. 5. The F-measure curve versus different thresholds (*first row*) and precision-recall curve (*second row*).

Figure 5 shows the F-measure curve (1_{st} row) and the precision-recall curve (2_{nd} row) of all the methods. Our F-measure curve consistently lies above other methods and achieves the best performances. Across all datasets, our F-measure curve has the flattest slope and largest area under the curve, demonstrating that our generated saliency maps present good quality against varying thresholds. Moreover, our precision-recall curve is significantly shorter and lies above other methods, which indicates that our method has fewer *false negative* predictions in the saliency maps.

TABLE II
COMPARISON AGAINST TRANSFORMER METHODS.

Method	ECSSD		DUTS-TE		HKU-IS		PASCAL-S	
	MAE	mF_β	MAE	mF_β	MAE	mF_β	MAE	mF_β
VTS [37]	0.034	0.911	0.037	0.842	0.030	0.903	0.067	0.832
GVT [38]	0.036	0.914	**0.035**	**0.850**	0.029	0.906	0.063	0.830
Ours	**0.031**	**0.920**	**0.035**	0.849	**0.026**	**0.913**	**0.062**	**0.838**

Comparison against Transformer-based Approaches. Table II compares the performance against transformer-based approaches. Our method achieves very competitive results.

Qualitative Evaluation. Some representative visual examples are shown in Figure 6. We select images from some challenging scenarios, including complex object in cluttered backgrounds (1_{st} row), partly occluded object (2_{nd} row), small object with low color contrast (3_{rd} row), object with low background contrast (4_{th} row), object with irregular and complex shapes (5_{th} row), multiple objects with shadows (6_{th} row), multiple large objects (7_{th} row), and complex objects with high inter-object contrast (last row). It can be seen that our method consistently well suppresses background noise and accurately segments the salient objects of various sizes with coherent details.

TABLE III
MODEL SIZE OF OUR APPROACH AND OTHER METHODS.

Method	Ours	MINet	GateNet	F3Net	EGNet	BASNet
#Params (M)	**27.82**	162.38	128.63	**25.54**	87.06	111.69
Speed (FPS)	99.81	45.02	42.15	74.67	**24.00**	37.54

Model Size and Speed Comparison. Table III compares the model size and speed of some state-of-the-art methods. Due to the proposed quasi-equilibrium feature pyramid that can model realistic infinite-depth network with a single layer, our model only has marginally more parameters than F3Net [19] and is far more light-weighted than the others. Since our method

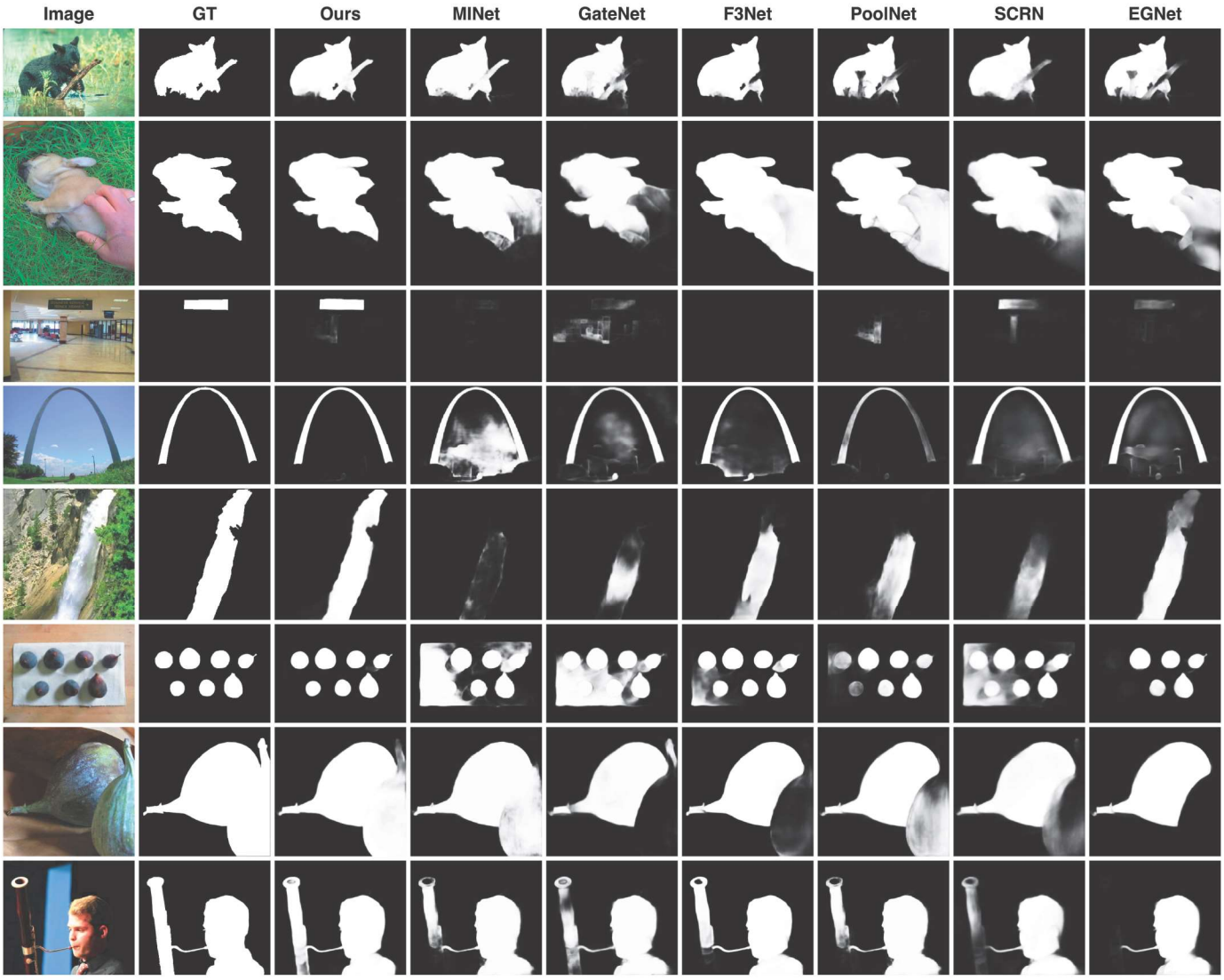

Fig. 6. Visual comparison of our method with other state-of-the-art methods in different challenging scenarios. Our method can well distinguish salient objects and suppress background noise, giving better visual appeal than others.

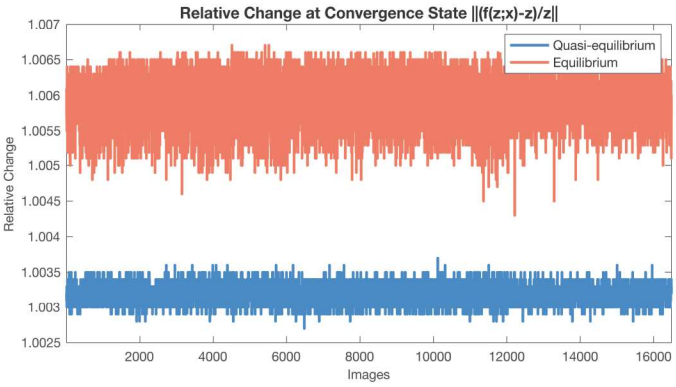

Fig. 7. Relative changes at the convergence state of our quasi-equilibrium model and the equilibrium model [32], [31].

requires solving fixed-point equation with a large size vector, it is more computational expensive compared with other explicit deep learning methods based SOD approaches. Nonetheless, we note that the slow inference speed is an inherent weakness of implicit deep learning. The weakness comes with the benefit of low memory usage.

B. Empirical Existence of Quasi-equilibrium

In order to investigate the empirical existence of the proposed quasi-equilibrium, we compute the relative change at the convergence state $||(f_\theta(\mathbf{z}^\star;\mathbf{x})-\mathbf{z}^\star)/\mathbf{z}^\star||$ of both our quasi-equilibrium model and equilibrium model [32], [31] on five evaluation datasets that have $16,484$ images in total. If the resultant hidden states $\mathbf{z}^\star$ are closer to the ideal equilibrium, the relative changes should be smaller. Figure 7 visualizes the relative changes of all the test images. Our method has much smaller relative changes with less variation than the equilibrium model [31], [32]. This behavior demonstrated that the proposed method has modelled more realistic and stable equilibrium states of the feature pyramid.

C. Failure Cases

Despite the empirical results of our method, there exists some failure cases. Here we show few examples in Fig. 8. As can be observed, our method might only generate parts of the object or predict unsalient objects., which could be due to the lack of semantic information. Also, the strong global context modelling abilities our method might aggravate this issue: focusing too much on the global context is likely to make the model neglect the object parts.

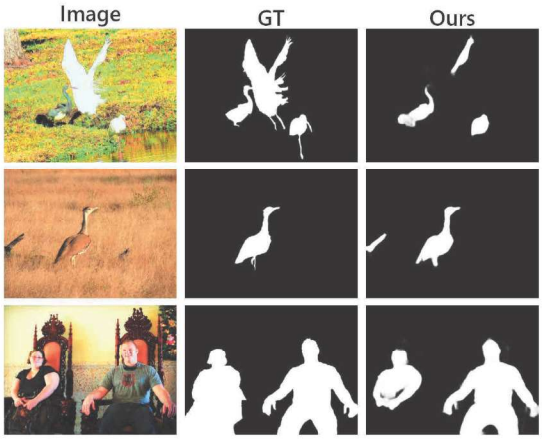

Fig. 8. Failure cases due to the strong global context modelling of our method and the lack of semantic supervision.

D. Connection with Video Object Segmentation

Our proposed approach is a general methodology for representation learning of feature pyramids. So it also fits other similar tasks such as Video Object Segmentation (VOS) [61], [62], [63], [64]. However, there exist two challenges that might stop directly applying our method on VOS. Firstly, the video inputs need a much larger feature pyramid compared with image inputs, which would incur heavier computational burdens. Secondly, the rich temporal cues between video frames should be carefully incorporated and exploited by some dedicatedly-designed modules. How to address the above two issues is worth further research in our future work.

E. Ablation Study

In this section, we conduct some ablation studies to validate the effectiveness of our proposed quasi-equilibrium model and differentiable edge extractor.

TABLE IV
COMPARISON OF PERFORMANCES ON DUT-OMRON DATASET USING DIFFERENT SOLVERS.

Solver	Depth	$MAE \downarrow$	m $F_\beta \uparrow$	$F_\beta^\omega \uparrow$	Max $F_\beta \uparrow$	$S_m \uparrow$
Unrolling	1	0.063	0.753	0.744	0.814	0.834
	2	0.063	0.767	0.754	0.823	0.840
	3	0.062	0.769	0.755	0.822	0.841
Equilibrium [32]	∞	0.060	0.771	0.757	0.822	0.843
Our quasi-equilibrium	∞	**0.057**	**0.778**	**0.762**	**0.825**	**0.845**

Effect of Quasi-equilibrium. We compare our proposed quasi-equilibrium model with other alternate solvers, *i.e.,* the equilibrium in MDEQ [32] and unrolling a weight-tied transformation for a few layers. As shown in Table IV, unrolling the transformation for more layers can improve the performance to some extent but is not comparable against equilibrium model [32]. Our proposed quasi-equilibrium solver significantly outperforms the equilibrium model thanks to the more realistic modelling of the convergence states.

TABLE V
PERFORMANCES OF DIFFERENT EDGE OPTIMIZATION MECHANISMS ON DUT-OMRON DATASET.

Method	MAE	m F_β	F_β^ω	Max F_β	S_m	Binary Ratio (%)
SSIM loss	**0.057**	0.774	0.760	0.819	0.841	97.45
weighted IoU loss	0.058	0.773	0.759	0.822	**0.846**	97.32
low-level edges	0.059	0.768	0.755	0.820	0.843	97.03
w/o edge extractor	0.059	0.758	0.750	0.818	0.830	96.32
Our edge extractor	**0.057**	**0.778**	**0.762**	**0.825**	0.845	**97.87**

Effect of Differentiable Edge Extractor. To evaluate the impact of our proposed differentiable edge extractor, we substitute it with other popular edge-preservation mechanisms. These mechanisms include SSIM loss [17], boundary-aware IoU loss [21], extracting edges from the feature at the lowest level [16]. As discussed in Sec. III-D, our method can also help the network to eliminate the non-binary predictions. To validate this, we compare the binary value ratio of the generated saliency maps (*i.e.*, defined as the ratio of pixels whose values are either larger than 0.9 or smaller than 0.1). Table V compares the performances and ratio of binary values. Obviously, none of these methods have comparable performances against ours on the evaluation metrics. Our proposed edge extractor also makes the generated the saliency maps have larger binary ratios and closer to the ideal binary map. To better illustrate the effectiveness of our edge filter, we give some visual examples in Fig. 9. When the proposed differentiable edge filter is applied, the object boundaries are indeed better kept and the nondeterministic predictions (*i.e., artifacts*) are largely removed.

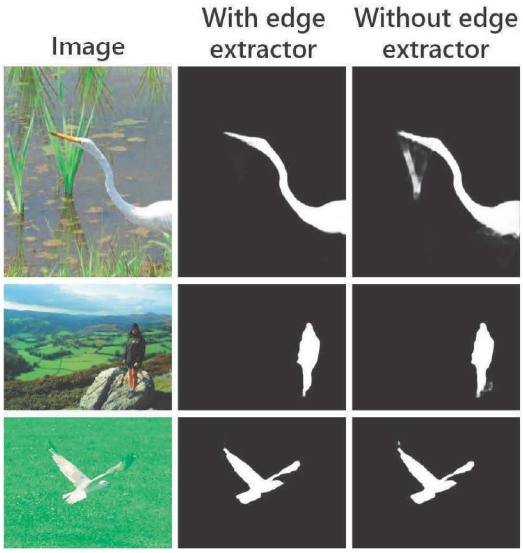

Fig. 9. Visual impact of our edge filter. Our proposed method can indeed keep the sharp boundary of object and eliminate nondeterministic predictions.

TABLE VI
COMPARISON OF PERFORMANCES USING DIFFERENT EDGE DETECTION FILTERS ON DUT-OMRON DATASET.

Edge Operator	$MAE \downarrow$	m $F_\beta \uparrow$	$F_\beta^\omega \uparrow$	Max $F_\beta \uparrow$	$S_m \uparrow$
Prewitt	0.058	0.770	0.761	0.825	0.840
Laplacian	0.059	0.768	0.755	0.822	0.841
Frei-Chen	0.058	0.773	0.760	**0.827**	0.843
Our Sobel	**0.057**	**0.778**	**0.762**	0.825	**0.845**

Impact of Edge Filter. As discussed in Sec. III-D, our proposed differentiable edge extractor can also support any other edge detection filters. We try some other classical edge detectors, including Prewitt filter, Frei-Chen filter, and Laplacian filter. These filters are defined as:

$$P=\begin{bmatrix}1 & 0 & -1\\ 1 & 0 & -1\\ 1 & 0 & -1\end{bmatrix}, F=\begin{bmatrix}1 & 0 & 1\\ \sqrt{2} & 0 & -\sqrt{2}\\ 1 & 0 & -1\end{bmatrix}, L=\begin{bmatrix}0 & 0 & 0\\ 0 & 1 & -1\\ 0 & 0 & 0\end{bmatrix}. \quad (16)$$

The differences between these filters mainly lie in the strength of diagonal edge detection. For conciseness concern, the normalization constant before each filter is omitted. Table VI shows the performances using different filters. Prewitt and Laplacian edge filters can not achieve competitive performances, as they fail to consider the gradients in diagonal directions. Our used Sobel filter achieves the best evaluation results against the others. We believe that it is mainly because the Sobel filter well combines and balances the gradients in different directions.

TABLE VII
COMPARISON OF PERFORMANCES USING EXTRA SUPERVISION AND SMALLER FEATURE PYRAMIDS ON DUT-OMRON DATASET.

Edge Operator	$MAE \downarrow$	m $F_\beta \uparrow$	$F_\beta^\omega \uparrow$	Max $F_\beta \uparrow$	$S_m \uparrow$
Intermediate Supervision	**0.056**	**0.780**	**0.763**	0.824	0.837
Feature Pyramid omitting F_2	0.060	0.762	0.751	0.810	0.829
Our Quasi-equilibrium	0.057	0.778	0.762	**0.825**	**0.845**

Impact of Feature Level and Intermediate Supervision. As done in some SOD methods [16], [13], adding extra supervision at intermediate features might improve the performance. It is thus natural to consider enforcing intermediate feature supervision in our method. Also, checking the performance of a smaller feature pyramid (omitting F_2 in the concatenation) would help understand the underlying working mechanism; the feature pyramid without F_2 greatly reduces the computational burden. To answer the above questions, we conduct another ablation study and present the results in Table VII. As can be observed, enforcing supervision to intermediate features bring very marginal performance improvements on MAE and F_β. The incurred additional parameters and time complexity outweigh the performance improvements. As for the feature pyramid, omitting F_2 leads to obvious performance drops across all the metrics. This meets our expectation as F_2 mainly contains crucial information about objects' boundaries. Abandoning this feature might lose the edge information and thus hurt the performance.

V. CONCLUSION

Based on the empirical observation, we identify the non-existence of ideal equilibrium model and propose a novel quasi-equilibrium feature pyramid network that seeks for more realistic convergence states. We further propose a differentiable edge extractor that explicitly optimizes the contour information and eliminates the non-deterministic predictions. Extensive experimental results demonstrate that the proposed method achieves new state-of-the-art performances on five benchmark datasets under six measures.

Acknowledgments. This research was supported by the EU H2020 projects AI4Media (No. 951911).

REFERENCES

[1] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, 1980. 1
[2] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE TIP*, vol. 24, no. 12, pp. 5706–5722, 2015. 1, 2
[3] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang, "Salient object detection in the deep learning era: An in-depth survey," *arXiv preprint arXiv:1904.09146*, 2019. 1, 2
[4] T. Kadir and M. Brady, "Saliency, scale and image description," *Springer IJCV*, vol. 45, no. 2, pp. 83–105, 2001. 1
[5] V. Mahadevan and N. Vasconcelos, "Saliency-based discriminant tracking," in *CVPR*, 2009. 1
[6] Y. Gao, M. Wang, D. Tao, R. Ji, and Q. Dai, "3-d object retrieval and recognition with hypergraph analysis," *IEEE TIP*, vol. 21, no. 9, pp. 4290–4303, 2012. 1
[7] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE TPAMI*, vol. 37, no. 3, pp. 569–582, 2014. 1, 2
[8] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *CVPR*, 2013. 1, 2, 5
[9] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *CVPR*, 2007. 1, 2
[10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015. 1
[11] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *CVPR*, 2020. 1, 2, 5, 6
[12] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *ECCV*, 2020. 1, 5, 6
[13] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. Torr, "Deeply supervised salient object detection with short connections," in *CVPR*, 2017. 1, 2, 9
[14] N. Liu, J. Han, and M.-H. Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *CVPR*, 2018. 1, 2
[15] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *CVPR*, 2018. 1, 2, 5, 6
[16] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," in *ICCV*, 2019. 1, 2, 5, 6, 8, 9
[17] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *CVPR*, 2019. 1, 2, 5, 6, 8
[18] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *ICCV*, 2019. 1, 2, 5, 6
[19] J. Wei, S. Wang, and Q. Huang, "F^3net: Fusion, feedback and focus for salient object detection," in *AAAI*, 2020. 1, 2, 5, 6
[20] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, and Q. Tian, "Label decoupling framework for salient object detection," in *CVPR*, 2020. 1, 2
[21] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection," in *CVPR*, 2017. 1, 2, 8
[22] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *CVPR*, 2018. 1, 2
[23] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *ECCV*, 2018. 1, 2

[24] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *CVPR*, 2019. 1, 2

[25] R. Wu, M. Feng, W. Guan, D. Wang, H. Lu, and E. Ding, "A mutual learning method for salient object detection with intertwined multi-supervision," in *CVPR*, 2019. 1, 2

[26] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *CVPR*, 2019. 1, 2, 4, 5, 6

[27] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *ICCV*, 2017. 1, 2

[28] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *CVPR*, 2018. 1, 2

[29] 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 *CVPR*, 2019. 1, 2, 5, 6

[30] Z. Wu, L. Su, and Q. Huang, "Stacked cross refinement network for edge-aware salient object detection," in *ICCV*, 2019. 1, 2, 5, 6

[31] S. Bai, J. Z. Kolter, and V. Koltun, "Deep equilibrium models," *NeurIPS*, 2019. 1, 2, 3, 7

[32] S. Bai, V. Koltun, and J. Z. Kolter, "Multiscale deep equilibrium models," *NeurIPS*, 2020. 1, 2, 3, 4, 7, 8

[33] T. Wang, X. Zhang, and J. Sun, "Implicit feature pyramid network for object detection," *arXiv preprint arXiv:2012.13563*, 2020. 1, 2, 3

[34] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Springer IJCV*, vol. 59, no. 2, pp. 167–181, 2004. 2

[35] C. Guo, Q. Ma, and L. Zhang, "Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform," in *CVPR*, 2008. 2

[36] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *ICLR*, 2021. 2

[37] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *ICCV*, 2021. 2, 5, 6

[38] J. Zhang, J. Xie, N. Barnes, and P. Li, "Learning generative vision transformer with energy-based latent space for saliency prediction," *NeurIPS*, 2021. 2, 5, 6

[39] K. Huang, C. Tian, J. Su, and J. C.-W. Lin, "Transformer-based cross reference network for video salient object detection," *Pattern Recognition Letters*, 2022. 2

[40] C. Xie, C. Xia, M. Ma, Z. Zhao, X. Chen, and J. Li, "Pyramid grafting network for one-stage high resolution saliency detection," in *CVPR*, 2022. 2

[41] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *CVPR*, 2019. 2

[42] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, "Neural ordinary differential equations," *arXiv preprint arXiv:1806.07366*, 2018. 2

[43] E. Haber and L. Ruthotto, "Stable architectures for deep neural networks," *Inverse Problems*, vol. 34, no. 1, p. 014004, 2017. 2

[44] S. Bai, J. Z. Kolter, and V. Koltun, "Trellis networks for sequence modeling," in *ICLR*, 2018. 2, 3

[45] F. Pineda, "Generalization of back propagation to recurrent and higher order neural networks," in *Neural information processing systems*. Citeseer, 1987, pp. 602–611. 2

[46] L. B. Almeida, "A learning rule for asynchronous perceptrons with feedback in a combinatorial environment," in *Artificial neural networks: concept learning*, 1990, pp. 102–111. 2

[47] C. G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Mathematics of computation*, vol. 19, no. 92, pp. 577–593, 1965. 2, 3

[48] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and L. Kaiser, "Universal transformers," in *ICLR*, 2018. 3

[49] R. Dabre and A. Fujita, "Recurrent stacking of layers for compact neural machine translation models," in *AAAI*, 2019. 3

[50] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," *arXiv preprint arXiv:1512.05287*, 2015. 4

[51] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *CVPR*, 2013. 5

[52] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *CVPR*, 2014. 5

[53] G. Li and Y. Yu, "Visual saliency detection based on multiscale deep cnn features," *IEEE TIP*, vol. 25, no. 11, pp. 5012–5024, 2016. 5

[54] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *CVPR*, 2017. 5

[55] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *CVPR*, 2012. 5

[56] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *CVPR*, 2009. 5

[57] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *CVPR*, 2014. 5

[58] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *ICCV*, 2017. 5

[59] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 5

[60] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009. 5

[61] D. Zhang, J. Han, L. Yang, and D. Xu, "Spftn: A joint learning framework for localizing and segmenting objects in weakly labeled videos," *IEEE T-PAMI*. 8

[62] R. Cong, J. Lei, H. Fu, F. Porikli, Q. Huang, and C. Hou, "Video saliency detection via sparsity-based reconstruction and propagation," *IEEE TIP*, 2019. 8

[63] P. Wen, R. Yang, Q. Xu, C. Qian, Q. Huang, R. Cong, and J. Si, "Dmvos: Discriminative matching for real-time video object segmentation," in *ACM MM*, 2020. 8

[64] P. Huang, J. Han, N. Liu, J. Ren, and D. Zhang, "Scribble-supervised video object segmentation," *IEEE/CAA Journal of Automatica Sinica*, 2021. 8

Yue Song received B.S. *cum laude* from KU Leuven, Belgium and joint M.Sc. *summa cum laude* from University of Trento, Italy and KTH Royal Institute of Technology, Sweden. Currently, he is a Ph.D. student with the Multimedia and Human Behavior Understanding Group at University of Trento, Italy. His main research interests are computer vision and numerical methods.

Hao Tang is currently a Postdoctoral with Computer Vision Lab, ETH Zurich, Switzerland. He received the master's degree from the School of Electronics and Computer Engineering, Peking University, China and the Ph.D. degree from the Multimedia and Human Understanding Group, University of Trento, Italy. He was a visiting scholar in the Department of Engineering Science at the University of Oxford. His research interests are deep learning, machine learning, and their applications to computer vision.

Mengyi Zhao is a Ph.D. student with Beihang University. Her research interests are deep learning, motion prediction, and computer vision.

Nicu Sebe is a Professor with the University of Trento, Italy, leading the research in the areas of multimedia information retrieval and human behavior understanding. He was the General CoChair of the IEEE FG Conference 2008 and ACM Multimedia 2013, and the Program Chair of the International Conference on Image and Video Retrieval in 2007 and 2010, ACM Multimedia 2007 and 2011. He was the Program Chair of ICCV 2017 and ECCV 2016, and a General Chair of ACM ICMR 2017. He is a fellow of the IAPR.

Wei Wang is an Assistant Professor of Computer Science at University of Trento, Italy. Previously, after obtaining his PhD from University of Trento in 2018, he became a Postdoc at EPFL, Switzerland. His research interests include machine learning and its application to computer vision.