Distributed Iterative Gating Networks for Semantic Segmentation

Rezaul Karim[1] , Md Amirul Islam[2,3] , Neil D. B. Bruce[2,3]
[1]York University, [2]Ryerson University, [3]Vector Institute for Artificial Intelligence
karimr31@yorku.ca, amirul@scs.ryerson.ca, bruce@ryerson.ca

Abstract

In this paper, we present a canonical structure for controlling information flow in neural networks with an efficient feedback routing mechanism based on a strategy of Distributed Iterative Gating (DIGNet). The structure of this mechanism derives from a strong conceptual foundation, and presents a light-weight mechanism for adaptive control of computation similar to recurrent convolutional neural networks by integrating feedback signals with a feed forward architecture. In contrast to other RNN formulations, DIGNet generates feedback signals in a cascaded manner that implicitly carries information from all the layers above. This cascaded feedback propagation by means of the propagator gates is found to be more effective compared to other feedback mechanisms that use feedback from output of either the corresponding stage or from the previous stage. Experiments reveal the high degree of capability that this recurrent approach with cascaded feedback presents over feed-forward baselines and other recurrent models for pixel-wise labeling problems on three challenging datasets, PASCAL VOC 2012, COCO-Stuff, and ADE20K.

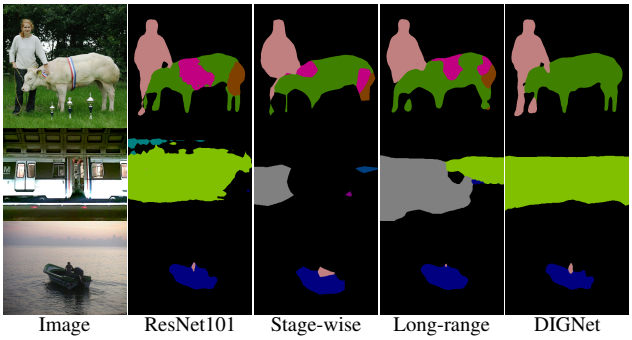

Figure 1. Examples of **DIGNet** predictions compared to other feedback routing mechanisms (ResNet101-8s as backbone) on PASCAL VOC 2012 dataset. Stage-wise feedback uses recurrent gating similar to [22], long-range uses the initial prediction as feedback signal similar to [29, 21], and our DIGNet uses cascaded feedback generation using propagator gates. Both stage-wise and long-range feedback fails to resolve categorical ambiguity, recover spatial details ($\mathbf{1^{st}}$ and $\mathbf{2^{nd}}$ row), and precisely segment smaller object ($\mathbf{3^{rd}}$ row) whereas DIGNet iteratively improves predictions by refining spatial detail and diminishing representational ambiguity within the network.

1. Introduction

Deep learning models have achieved a high degree of success for problems involving dense pixel labeling [35, 6, 37, 1, 50, 12, 20, 32, 7] with a wide range of associated applications[27, 31]. Improvements in this domain have come by virtue of increasingly deep networks [25, 44, 45, 17], pre-training that leverages data from multiple large scale datasets [9, 33] to boost overall performance, and innovations on architectural properties of networks. In this paper, we focus heavily on the last of these categories in proposing a scheme for efficient selection and routing of feed-forward information in neural networks.

There are a few specific considerations that motivate this paper, which presents a simple lightweight gating mechanism [42, 20, 28] that is *top down* wherein larger convolutional windows and more discriminative features play a role in guiding feedforward activation among earlier features that are more local and ambiguous with respect to category. (1) The trade-off between spatial resolution for additional feature layers deeper in the network can imply a loss of spatial granularity in categorical labeling. While a simple labeling problem might be by and large globally consistent, there may remain local inconsistencies that come from this limitation. (2) The nature of convolution implies that a network is limited at any layer in the spatial extent of pixels or features that can be considered in concert or related to one another. It is also the case that the effective spatial extent of convolution among deeper layers covers a broader spatial extent by virtue of downsampling. This implies that a recurrent signal for gating can allow spatially distal discriminative features to have mutual influence over each other allowing for links to be formed between object parts that make up the whole object. (3) For intermediate features, some of these may be discriminative along certain categorical boundaries but not others with respect to the label space. An implication of this is that subject to an initial

feedforward pass, features in intermediate layers may carry categorical ambiguity that is absent among more discriminative features present in deeper layers. Allowing such information to be relayed in a reverse direction can help to resolve such interference.

The specific structure we adopt is based on a symbiotic combination of propagator and modulator nodes which are very flexible and highly efficient with respect to allowing information represented in one part of the network to reach other layers. The propagator gates are responsible for generating feedback signals in a cascaded manner for distributed iterative gating and modulator gates are responsible for contextual feature reweighting on selected intermediate stages based on feedback signals. This strategy is marked by a carefully designed structure for connectivity that allows for a high degree of interaction among gating and inference blocks. Moreover, the iterative (recurrent) nature of this mechanism allows for the output and internal representations to be gradually refined and also to propagate outward spatially producing an unambiguous and globally consistent prediction (see Fig. 1).

This approach is shown to significantly boost the performance of feed-forward baselines and generate better segmentation compared to both baselines and other recurrent feedback based approaches. Furthermore, iterative inference by means of DIG is shown to converge very fast relative to other feedback mechanisms.

2. Related Work

Recent state-of-the-art semantic segmentation networks [35, 6, 37, 1, 12, 20, 32, 7] typically follow the structure of a Fully Convolutional Network (FCN). Although the feature maps produced in the higher-layers of conventional CNNs [25, 44, 45, 17] carry a strong representation of semantics, the ability to retain precise spatial details in dense labeling problems (e.g. semantic segmentation) is limited due to the poor spatial resolution.

Recent works on semantic segmentation have mainly focused on improving network performance by modifying the network architecture. However, there are limits to the degree of improvement that is possible if the networks are confined to carry out computation based only on a single feedforward pass. A few efforts [51, 36, 39, 22, 23, 29, 27, 46] have been proposed to iteratively improve the output of a feed-forward network and overall performance. In this work, we argue that the recurrent processing of inputs with an efficient feedback mechanism has more desirable properties, the value of which are evident in the similar mechanisms of processing observed in the human brain [13, 26].

Several works consider employing recurrent processing [39, 31, 48, 24, 21] or feedback based attention mechanisms [29] in combination with conventional CNNs. Another line of work [36, 51] applies a recurrent module (e.g. ConvLSTM) on top of the network to iteratively refine the initial prediction. Although feed-forward gating mechanisms [20] have shown some success for recognition tasks, recurrent feedback mechanisms play an important role in pushing performance further for several tasks of interest [51, 4, 2].

Related to our proposed approach is the idea of learning a feed-forward network in a iterative manner that involves propagating feedback in a top-down fashion. Recent feedback based approaches [51, 29, 42] follow the pipeline of correcting an initial prediction by propagating feedback in a few different ways. TDM [42] proposed a pipeline where a top-down modulation network is integrated with the bottom-up feed-forward network for object detection similar to refinement based encoder-decoder architectures [32, 37, 20, 40].

Our proposed approach differs from the above feedback based networks in that we propagate the feedback in a top-down fashion starting with the output (initial prediction) of the last layer. Our feedback mechanism iteratively adjusts the feature maps in earlier layers through feedback from higher layers and corrects initial errors towards assignment of the true category.

In summary, our feedback mechanism guides earlier features based on the feedback signal which has information from the layer immediately above, and by virtue of connectivity, from all layers above. Additionally, the iterative nature allows the feedback mechanism to carry information in a path similar to a compact hypercolumn representation and improve the quality of predictions in subsequent iterations.

3. Distributed Iterative Gating Network

In this paper, we primarily focus on efficient feedback mechanisms coupled with feed-forward semantic segmentation frameworks [17, 7]. In general, networks that include a feedback component [27, 19, 51, 22, 29] have a standard feedforward structure that consists of shallow high-resolution early spatial layers, and increasingly lower resolution richer features within deeper layers. A feedback mechanism is typically applied with iterative inference where feedback works as a correcting signal to guide the features in earlier layers based on high-level semantic representations.

Core functional parts of the feedback mechanism include (a) a selection of early or intermediate layers where a correcting signal is fed back, (b) generation of a feedback signal for each selected layer as a function of some specific deeper layers and (c) a mechanism to modulate earlier layer features applying the feedback signal. Selection of several intermediate stages [22, 21, 29] have been found to be more effective than a recurrence mechanism that only feeds back information to the first or input layer [39]. Generating feedback using information from the output layer [21, 29] or

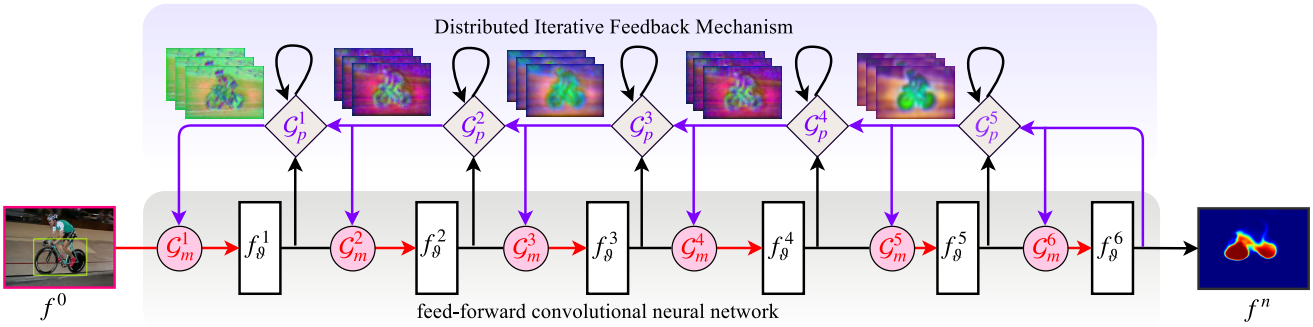

Figure 2. An illustration of our proposed **Distributed Iterative Gating Network** (DIGNet). DIGNet involves augmentation of a canonical neural network backbone through addition of gating modules, while operating in a recurrent iterative manner. $(f_\vartheta^1 \cdots f_\vartheta^6)$ are bottom-up feature blocks, $(\mathcal{G}_p^1 \cdots \mathcal{G}_p^5)$ are the propagator modules that propagate high-level information as feedback via a top-down pathway in order to guide the representation carried by intermediate and low-level feature layers. $(\mathcal{G}_m^1 \cdots \mathcal{G}_m^6)$ are modulator gates that modulate the bottom-up flow of activation with guidance from the propagator gates. A detailed description of each component is presented in Sec. 3.3.

from the output of deeper intermediate stages [22] can improve performance over baselines albeit with several limitations (See Fig. 1). Modulating intermediate features by applying a feedback signal is generally done with additive combination or multiplicative re-weighting. In this work, we mainly focus on the second part and propose a cascaded feedback generation method that works in a distributed manner.

We have made the case that the effectiveness of a feedback mechanism may depend on considerations that include the large difference in semantically relevant or category specific representation between early and deep feature layers, or equally, the large difference in spatial resolution and spatial extent of filters typical of such networks. On one extreme, low-level features are likely to capture only concepts such as edges, contours or lines. Intuitively, allowing high-level (deep) features to directly guide low-level representations may be misguided in the absence of a satisfactory bridge provided by intermediate features in providing semantic guidance to exert influence over low-level features. It is evident that central to the *right* mechanism, is efficient integration of low and high-level features to exact all of the advantages that derive from access to both strong representation of spatial resolution, and semantically rich categorical information in a compact representation that does not introduce redundancy among features.

In the following subsections, we propose a new architecture called *Distributed Iterative Gating Network* (DIGNet) that allows for feedback to propagate from deeper layers to earlier layers. This happens explicitly by virtue of connectivity among gating units, and implicitly based on updates to feedforward activation. We explain how such an architecture, namely one with a meaningful distributed feedback mechanism can produce more discriminative features by bridging the gaps in semantic specificity and resolution that exist between very deep and early layers in order to resolve categorical ambiguity.

3.1. DIGNet Architecture

In this section, we introduce our proposed DIGNet that includes an efficient distributed feedback mechanism to bridge the gap between high-level and low-level features. The main objective of DIGNet is to propagate more semantic information into earlier features that will help to provide clues about semantic content within intermediate and earlier layers. We choose conventional feed-forward network architectures (e.g. ResNet101-FCN with stride 8) as our feedforward backbone semantic segmentation network. Our proposed feedback mechanism augments the feedforward backbone with two different gating modules, (a *propagator* and a *modulator*) as illustrated in Fig. 2 to facilitate a broad exchange of information about internal representation within the network. The propagator gates together work as a lightweight parallel network that feeds information in the backward direction and generates feedback signals at each intermediate stage carrying information from all selected deeper stages. The modulator gates which are augmented between selected intermediate stages to modulate features in the feedforward backbone. The modulation is done by multiplicative re-weighting applying some weights generated from feedback signals. Details of the modulator and propagator gates are discussed in Sec. 3.3.

Our approach is motivated by the capacity to propagate more discriminative and semantically relevant information towards lower-layers which can be updated based on a subset of information from each downstream intermediate stage. In subsequent iterations, all stages are effectively informed in a relevance guided fashion about the outcome of all deeper stages of inference from the previous iteration. While this mechanism seems to provide greater flexibility from an intuitive perspective, and a more efficient control structure, we also find this strategy to be more helpful empirically in providing feedback in the form of an error correcting signal that modulates earlier layers to generate

more discriminative features and resolve categorical ambiguity due to spatial separation of discriminative features.

Previous efforts focus on iterative improvement leveraging earlier layer activation based on only the current prediction [21, 29], or feedback from the feedforward stage that immediately follows the stage where refinement is occurring [22]. Generating feedback from the output of the last layer may have several limitations. First, as the feedback signal has much lower resolution than the input or lower layers, while resolving categorical ambiguity, it can also be misguided in regions of sharp object boundaries. Second, if there is a missing object in the initial prediction, there is a possibility that the representation of that object may be lost at some intermediate stage. In incorporating intermediate features in feedback signal generation, these limitations may be overcome. Also, generating feedback from the feedforward stage that immediately follows has a major limitation of lacking strong semantic information in earlier stages which can somewhat be improved with combining additional feedback from the last layer [22]. Considering the limitations of current approaches, the proper way to generate feedback signal might be combining the best of both properties, having output and all intermediate stages to contribute in feedback generation. Therefore, we generate feedback in a cascaded manner implicitly carrying some information from all the stages in an aggregated compact representation.

Intuitively, the key idea of designing the feedback mechanism in a cascade manner (deeper $\rightarrow$ shallower) can be seen as a similar to generating a hypercolumn representation [15] where the propagated feedback signal at an earlier stage has any necessary guidance from all subsequent processing stages to correct the initial error. Naturally, the dimensionality of the feedback signal need increase with top to bottom propagation while bringing improvement subject to a hypercolumn style representation. However, in our case, the feedback signal is subject to blockwise compression through dimensionality reduction which apparently scales down the stack of feature maps by adjusting the feedback signal based on current activations before propagating towards earlier layers. The integration of this compressive strategy allows DIGNet to produce a *compact hypercolumn* representation as a feedback signal. Interestingly, we find this hypothesis efficient both in terms of computational cost and performance, as our ablation results will show.

3.2. DIGNet Iterative Inference

In this section, we discuss the iterative inference in the DIGNet. We are using the same notation as in Fig. 2 throughout the paper. For iterative inference the recurrence is unrolled for a certain number of iterations or time steps (T). During the first iteration, the modulator gates $(\mathcal{G}_m^1, \mathcal{G}_m^2, \cdots, \mathcal{G}_m^n)$ simply allow a bypass of feedforward information in a bottom-up manner and hence the network works similar to feedforward networks. The feed-forward stages $(f_\vartheta^1, f_\vartheta^2, \cdots, f_\vartheta^n)$ process the input image to produce a reasonable feature representation. So, DIGNet reduces to a basic feedforward network when $T = 1$.

In all the subsequent iterations, DIGNet executes two steps - (a) First, feedback signals are generated in a cascaded manner starting from the initial prediction towards the earlier layers. Note that all the propagator gates $(\mathcal{G}_p^1, \mathcal{G}_p^2, \cdots, \mathcal{G}_p^n)$ are activated in this step to facilitate feedback propagation. The initial output and intermediate features flow back through the propagator network generating feedback signals for all intermediate stages. (b) Another feedforward processing flows through the backbone network with modulator gates being activated. The modulators take signals from the propagator gates and modulate the feature representation received as input from the preceding feed-forward stage before forwarding it to next stage. This step can be seen as a traditional feed-forward network except that gating interacts with feedforward processing in effect producing adaptive features. Algorithm 1 describes the set of steps for iterative data flow and inference in DIGNet.

Following other feedback based approaches [2, 29, 22, 51], we optimize DIGNet with back-propagation through time (BPTT) by unrolling the recurrence for a certain number of time steps. To elicit a trade-off between performance and computational cost we set a value of *T=2* in our experiments. Note that, DIGNet does not employ any semantic supervision of intermediate predictions and only applies a cross-entropy loss to the final prediction at stage T. This speaks to the efficiency of communicating information broadly across the network as a loss at the final output is sufficient to realize substantive gains and effective modulator and propagator gates across the entire network.

3.3. DIGNet Gate Modules

Here, we discuss the careful design choices for gating modules involved in the feedback mechanism.

3.3.1 Propagator Gate

The propagator gates allow earlier layers to obtain richer semantic information in a top-down fashion, resulting in more significant interaction between low-level concepts and high-level visual features. As shown in Fig. 2, each propagator gate takes a feedback signal from the previous propagator gate and bottom-up features from the corresponding intermediate stage as input to generate a new feedback signal. Intuitively, the propagator gate learns what contextual semantic information to preserve in top-down feedback propagation. The inputs are passed through a shared series of successive operations, resulting in an updated feedback signal. The propagator module $\mathcal{G}_p$ first applies a 3×3 convolu-

Algorithm 1 DIGNet Data Flow and Iterative Inference

function DIGNET-DF($\mathcal{I}$)
  Initialize $f^0 = \mathcal{I}$
  for t $\leftarrow$ 1 to T_{steps} **do** ▷ unroll iteration, T
    if t > 1 **then** ▷ propagate feedback
      $\mathcal{F}^n = f^n$
      for k $\leftarrow (n-1)$ to 1 **do**
        $\mathcal{F}^k = \mathcal{G}_p^k(\mathcal{F}^{k+1}, f^k)$ ▷ *propagator*
      end for
    end if
    for i $\leftarrow$ 1 to n **do** ▷ number of stages, n
      $f^{(i-1)'} = \mathcal{G}_m^i(\mathcal{F}^i, f^{i-1})$ ▷ *modulator*
      $f^i = f_\vartheta^i(f^{(i-1)'})$ ▷ bottom-up feature
    end for
  end for
  return f^n
end function

tion and a ReLU non-linearity, which transforms the feedback signal input $\mathcal{F}^{(i+1)}$, bottom-up features f^i to $\mathcal{F}^{(i+1)'}$ and $f^{i'}$ respectively which have a common spatial dimensionality. The resultant feature maps are then combined through concatenation followed by a 1×1 convolution to generate the feedback signal $\mathcal{F}^i$ which is propagated backwards to the next top-down stage. The purpose of applying convolution on the concatenated feature map is to fuse the combined feature maps and reduce channel dimensionality to ensure a compact representation. If the spatial resolution of next top-down feature map f^{i-1} is higher than the feedback signal $\mathcal{F}^i$ then the feedback sample is upsampled by simple bilinear interpolation to have the same resolution. These operations are summarized as follows:

$$\mathcal{F}^i = \hat{y}(\mathbf{W_c} * (\underbrace{(\mathbf{W_a} * f^i)}_{\text{bottom-up feature}} \oplus \underbrace{(\mathbf{W_b} * \mathcal{F}^{i+1})}_{\text{feedback signal}})) \quad (1)$$

where $*$ and $\oplus$ denote a convolution operation and concatenation, $\hat{y}$ indicates upsampling through bilinear interpolation, and $\{\mathbf{W_a}, \mathbf{W_b}, \mathbf{W_c}\}$ are trainable weights. Note that the formulation for obtaining a feedback signal is the same at each top-down stage.

3.3.2 Modulator Gate

The main task of the modulator gate is to provide assistance in generating the input for the next bottom-up (feed-forward) stage by modulating information passed forward based on the feedback signal. Intuitively, the modulator gate learns to obtain a meaningful feedback signal to modulate intermediate and low-level features. The feedback signal $\mathcal{F}^i$ is processed first to have the same channel dimension as $f^{(i-1)}$. Inspired by [16, 34, 52, 8], we find that applying a global contextual prior is beneficial in generating the modulating signal as shown in Table. 1. We first create a spatial pyramid [52] of $\mathcal{F}^i$ with pooling rate $\{1, 3, 5, 7\}$. We then concatenate the pyramid features ($\mathcal{F}_1^{i'}$, $\mathcal{F}_2^{i'}$, $\mathcal{F}_3^{i'}$, $\mathcal{F}_4^{i'}$) and $\mathcal{F}^i$ to obtain the updated feedback signal $\mathcal{F}^{i'}$. A 1×1 convolution followed by a sigmoid is applied sequentially to transform and squash the channel dimension of $\mathcal{F}^{i'}$ similar to $f^{(i-1)}$, resulting in the modulating signal $\mathcal{F}^s$. Finally, $\mathcal{F}^s$ is combined with $f^{(i-1)}$ through element-wise multiplication. This new modulated bottom-up feature map $f^{(i-1)'}$ passed onto the next feed-forward stage f_ϑ^i as input.

DIGNet-ResNet101	$*$	1×1	Spatial Pyramid
	mIoU(%)	75.9	77.5

Table 1. Performance comparison of DIGNet(T=2) subject to the modulator design choices on the PASCAL VOC 2012 val set.

4. Experiments

To show the effectiveness of DIGNet, we present results from a series of experiments. Initially, we conduct ablation analysis to examine the impact of various design choices for DIGNet in considering the PASCAL VOC 2012 dataset [11]. Then, we evaluate DIGNet on three different semantic segmentation datasets, including PASCAL VOC 2012 [11], ADE20K [53], and COCO-Stuff [3]. Experimental results demonstrate the superiority of our proposed DIGNet architecture over baselines in a variety of respects.

4.1. Implementation Details

Inspired by previous work [7, 20, 7] we employ the "poly" learning rate policy to train the baseline networks and our DIGNet variant of the models. We employ a crop size of 321 $\times$ 321 and 513 $\times$ 513 during training and testing respectively to report experimental results on all datasets. We report experimental results for our baselines (ResNet101(32s), ResNet101(8s), and DeepLabv2-Res101) and corresponding DIGNet networks. For fairness, we use similar hyper-parameters for the baselines and our approach. We initialized baselines and our models with the COCO pre-trained weights where required, otherwise we initialize the network with ImageNet trained weights. Note that whenever we report experimental results for DIGNet this denotes ResNet101-DIGNet with unroll iteration, *T=2*.

4.2. Gating Semantic Information with DIGNet

To investigate the role of distributed iterative gating in DIGNet we conduct experiments under a few different settings. We focus on two major facts to validate the hypothesis of design choices, including the significance of an iterative solution and propagating more semantic information to earlier layers by applying gating modules.

Iterative solution with Cascaded Feedback Generation: In Table 2, we present quantitative results comparing different feedback routing mechanisms and significance of our iterative solution. The segmentation performance increases

by a significant margin with generating feedback in a cascaded manner compared to stage-wise recurrence or with feedback from the last stage only. This implies that DIGNet is successful in its objectives of bridging the information gaps through efficient and effective feedback propagation.

Feedback Method	T=1	T=2	T=3	T=4
Stage wise feedback [22]	71.3	73.4	73.9	74.9
Last layer feedback† [21, 29]	71.3	75.1	74.9	74.5
DIGNet	71.3	77.5	**77.7**	76.7

Table 2. Performance comparison of different feedback mechanisms subject to a varying number of unrolling iterations on the PASCAL VOC 2012 val set. In all cases, T=1 reduces the network to feedforward baseline ResNet101-8s [17].

Furthermore, the iterative nature of DIGNet provides adjustments to the earlier layers allowing for stage-wise feedback refinement and removing the ambiguity that may arise anywhere in the feed-forward network. We notice in Table. 2 that overall performance progressively improves beyond a fixed number of iterations and then starts saturating. We find this observation to be valid across different datasets and network architectures revealing fast convergence and stability. Therefore, DIGNet may be evaluated with an increasing number of iterations to improve predictive segmentation performance.

Semantic Information in Gating Low-level Features: Our solution of incorporating distributed gating modules in the feedback mechanism is inspired by the following: Feed-forward network activations closer to semantic supervision tend to capture more semantics, which can guide lower-level features to correct initial errors made in inference. Instead of immediately making a category-specific prediction based on the predicted probability in the first pass, we deploy a distributed gated feedback mechanism to propagate the predicted probability to the earlier layers to update the network. In DIGNet, semantic features extracted from the last layer are passed backward as feedback which is gated with the encoded features from each stage.

Method	$\mathcal{F}^1$	$\mathcal{F}^2$	$\mathcal{F}^3$	$\mathcal{F}^4$	$\mathcal{F}^5$	$\mathcal{F}^6$	mIoU(%)
ResNet101-FCN							65.3
						✓	70.5
					✓	✓	71.1
				✓	✓	✓	71.8
			✓	✓	✓	✓	71.9
		✓	✓	✓	✓	✓	72.6
	✓	✓	✓	✓	✓	✓	**72.5**

Table 3. Performance of DIGNet(T=2) with a varying extent of the reach of feedback gating for the PASCAL VOC 2012 val set.

We perform a series of experiments to examine the impact of distributed gating in each feed-forward stage by selecting a subset of inferential feature blocks that are subject to gating and use them to retrain DIGNet. Experimental results are shown in Table 3. It is clear that the segmentation quality gradually improves with the integration of more feedback propagation including to the early layers. Empirical results show that inclusion of all layers except for the initial layer sometimes achieves better results (Table 2), but inclusion of all layers in the gating process is often preferred as is the case in Table 3 and other results.

4.3. Results on PASCAL VOC 2012 dataset

PASCAL VOC 2012 is a popular semantic segmentation dataset consisting of 1,464 images for training, 1,449 images for validation and 1,456 images for testing, which includes 20 object categories and one background class. Following prior work [7, 35, 20, 32, 7], we use the augmented training set that includes 10,582 images [14]. First, we report experimental results on the PASCAL VOC 2012 validation set. We integrate DIG with ResNet-101 and Deeplabv2-ResNet101 architectures and explore the influence of the distributed feedback representation relative to the base network. Table 4 shows the comparison of different baselines and our proposed approach on the PASCAL VOC 2012 validation set. Interestingly, *ResNet101-DIGNet*

Method	mIoU	Method	mIoU
ResNet50-32s† [17]	59.4	**ResNet50-DIGNet**	**68**
ResNet101-32s† [17]	65.3	**ResNet101-DIGNet**	**72.5**
ResNet101-8s† [17]	71.3	**ResNet101-DIGNet**	**77.5**
DeepLabV2-Res101† [7]	74.9	**DeepLabV2-DIGNet**	**76.1**

Table 4. PASCAL VOC 2012 validation set results for baselines and DIGNet(T=2).

with *OS=32* marginally outperforms ResNet101-FCN with *OS=8* in terms of mIoU achieving 72.5% and 71.3% respectively. Also, *ResNet101-DIGNet* with *OS=8* yields better performance than Deeplabv2-ResNet101 providing a strong case for the value of our proposed distributed iterative feedback mechanism. Additionally, *DeeplabV2-DIGNet* significantly outperforms the baseline and achieves 76.1% mIoU without any *bells and whistles*. It is observed that the performance consistently increases for the baselines with the addition of DIG.

We further conduct experiments for the proposed DIGNet on the PASCAL VOC 2012 test set. Following existing works [7, 52, 37, 32], DIGNet is first trained on the augmented training set and then fine-tuned on the original PASCAL VOC 2012 trainval set. We evaluate DIGNet with muti-scale inputs including left-right flips, where the scales are $\{0.5, 0.75, 1.0, 1.25, 1.5\}$, and average the multi-scale outputs for final predictions. As shown in Table 5, DIGNet achieves 80.7% mIoU which is competitive compared to other baselines especially for a simple mechanism attached to a standard ResNet architecture. Note that, unlike many recent works, we did not employ hardware intensive optimization like training batch norm parameters , extremely time consuming procedures like pre-training on large scale databases with semantic supervision, or com-

Method	mIoU (%)
Adelaide_Very_Deep_FCN_VOC [49]	79.1
LRR_4x_ResNet-CRF [12]	79.3
DeepLabv2-CRF [7]	79.7
CentraleSupelec Deep G-CRF [5]	80.2
SegModel [41]	81.8
Deep Layer Cascade (LC) [30]	82.7
TuSimple [47]	83.1
Large_Kernel_Matters [38]	83.6
Multipath-RefineNet (Res152) [32]	83.4
PSPNet [52]	85.4
DeepLabv3 [7]	85.7
DIGNet	80.7

Table 5. Quantitative results in terms of mean IoU on PASCAL VOC 2012 test set.

bining multiple loss functions to boost performance yet we observe dramatic performance gains.

We provide a qualitative visual comparison of our approach with respect to the baselines in Fig. 3. With the proposed mechanism, we produce improved prediction results compared to the baselines and many of these regions are re-examined and refined with the help of DIG.

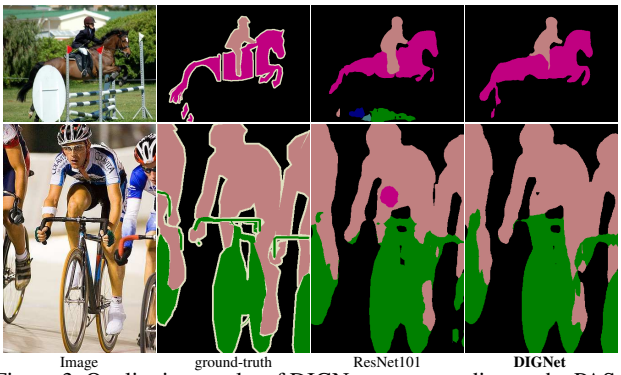

Figure 3. Qualitative results of DIGNet corresponding to the PASCAL VOC 2012 validation set.

4.4. Results on ADE20K

ADE20K is a newer and more complex dataset for scene parsing that provides semantic labels for 150 classes including 115 thing categories and 35 stuff categories, with more than 20k indoor and outdoor images. Table 6 presents the scene parsing results obtained with the ADE20K validation set for different baselines and our proposed approach. With ResNet101(8s) *DIGNet* alone yields 36.9% mIoU, significantly outperforming ResNet101-FCN and DeepLab-Res101 by about 3.3% and 1.6%, respectively. Additionally, DeepLabv2-DIGNet achieves 36.9% mIoU which outperforms the baseline.

4.5. Results on COCO-Stuff10k

COCO-Stuff10k is also a relatively recently released scene parsing dataset based on MS-COCO annotations.

Method	mIoU(%)	Pixel Acc.(%)	Overall(%)
CascadeNet [53]	34.90	74.52	54.71
DilatedNet [50]	34.3	76.4	55.3
PSPNet [52]	41.7	80.0	60.9
ResNet101†	33.6	75.4	44.2
ResNet101-DIGNet	36.9	77.3	46.6
DeepLabv2-ResNet101†	35.3	75.5	45.1
DeepLabv2-DIGNet	36.9	76.7	47.8

Table 6. Quantitative analysis of our approach based on different architectures *vs.* state-of-the-art methods based on the ADE20K validation set. † indicates our implementation.

Following the split in [3], we use 9k images for training and another 1k for testing to evaluate DIGNet. We further evaluate our model on the scene centric large-scale COCO-Stuff dataset to examine the value of the proposed distributed iterative gating mechanism. Comparison of scene parsing results on the COCO-Stuff dataset are reported in Table 7. Similar to previous experiments, we mainly focus on the effect of augmenting ResNet based architectures using DIGNet. Augmenting ResNet101(32s) for DIGNet provides improvement of 2.7% over the baseline. Similarly, augmenting ResNet101(8s) improves the performance significantly (33.4% *v.s.* 36.9%). We further apply DIG on the DeepLabv2 network which improves the baseline to some degree (34.1% *v.s.* 35.9%). For this challenging dataset, these improvements are quite significant.

Method	pAcc(%)	mAcc(%)	mIoU(%)
DeepLab [6]	57.8	38.1	26.9
OHE + DC + FCN [18]	66.6	45.8	34.3
DAG-RNN + CRF [43]	63.0	42.8	31.2
RefineNet-Res101 [32]	65.2	45.3	33.6
CCL [10]	66.3	**48.8**	35.7
ResNet101-32s [17]†	58.7	38.0	26.4
ResNet101-DIGNet	61.8	40.7	29.1
ResNet101-8s [17]†	64.6	44.9	33.4
ResNet101-DIGNet	**67.3**	47.4	**36.3**
DeepLabv2 (ResNet-101)† [7]	65.1	45.5	34.1
DeepLabv2-DIGNet	67.0	46.4	35.9

Table 7. Comparison of scene parsing results on the Coco-Stuff test set. † refers to our own implementation.

4.6. Study of Error Correction with DIGNet

We characterize the computational properties of generic unrolling operations in DIGNet given that it performs identical operations in each iteration. We address this consideration from three different vantage points by focusing on the initial prediction of the feed-forward network.

Categorical Ambiguity: When the initial class assignment is predicted incorrectly - for instance segmenting a horse as a cow or vice versa- we empirically found that DIGNet is capable of correcting the initial prediction in the very first iteration (see Fig. 4), highlighting the powerful influence of DIGNet to correct the poor initial prediction.

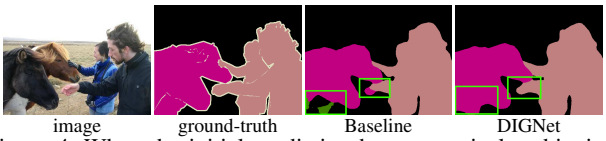

Figure 4. When the initial prediction has categorical ambiguity, DIGNet iteratively adjusts information passed forward through the feedback signal resulting in recognition of the correct class.

Partial Segmentation: When the initial prediction has coarse-grained or spatially limited mask (see Fig. 5), DIGNet improves partial segmentation to generate a detailed mask by incorporating distributed gating in the feedback mechanism, in some instances completing the object.

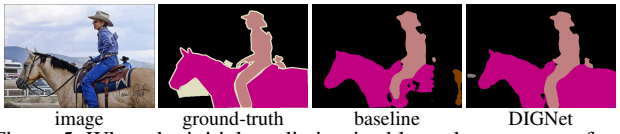

Figure 5. When the initial prediction is able to detect a part of an object, DIGNet gradually aligns the output more accurately with semantic labels, while labeling the initially missing regions.

Recover Missing Small Objects: When the initial prediction misses small objects in front of large objects, DIGNet can recover missing small objects (see Fig. 6). DIGNet succeeds because when feedback is generated in a cascaded manner including intermediate feature maps, some earlier representation where the object is more strongly represented is fed back to guide the next iteration.

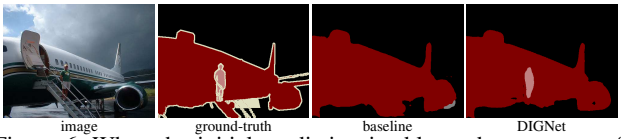

Figure 6. When the initial prediction is able to detect a part of an object, DIGNet gradually aligns output more accurately with semantic labels, while labeling the initially missing regions.

Coarse-to-Fine Representation: DIGNet processes at a relatively coarse spatial resolution due to the output stride applied on the image features with the absence of a refinement/decoder network. While the performance improvement is remarkable in just one additional iteration with DIGNet, we show that the hierarchical addition of propagator and modulator gates in the feedback mechanism can be represented as a coarse-to-fine refinement scheme.

Figure 7. Visualization of label quality after top-down addition of distributed iterative gating modules. For each row, we show the input image, ground-truth, ResNet101(32s) prediction, and the predicted label map of DIGNet when distributed iterative gating modules are included in a top→down manner.

Fig. 7 illustrates the degree of refinement obtained after integrating stage-wise gating modules. DIGNet$\ll \mathcal{F}^n \gg$ refers to feedback propagated until block n. Interestingly, with the addition of top-down recurrent feedback, DIGNet predictions continue to improve by recovering spatial details while aligning to resolve categorical ambiguity.

DIGNet's ability to iteratively resolve categorical ambiguity with precise localization of sharp object boundaries (Fig.4), improve partial segmentation (Fig.5), and correct initial errors by way of coarse-to-fine refinement (Fig.7) provides a convincing case for the effectiveness of distributed iterative gating mechanism.

4.7. Analyzing the Failure Cases of DIGNet

Despite the consistent performance improvement for the majority of cases, there are rare cases that are more challenging to predict. When DIGNet is allowed to iteratively propagate high-level semantics to earlier layers, it progressively improves the label map by way of top-down modulation (Fig. 7 and Table 3). In extreme cases, when the initial prediction of any foreground object shares similar visual features with the surrounding background it may gradually move to partial incorrect labeling (see Fig. 8). Here, DIGNet is able to predict the correct class including both the people and the airplane in the background. However, when feedback modulates the feedforward signal, the airplane is suppressed. Such a case may occur when confidence in two classes is very similar and the airplane in this case shares notable features with the background. Interestingly, the label is globally consistent which underscores the ability to successfully propagate confidence spatially despite an incorrect adjustment to the class label.

Figure 8. An example where the final labeling in each case tends to be globally consistent over objects.

5. Conclusion

In this paper we have presented a scheme for *Distributed Iterative Gating* called DIGNet. This strategy involves iterative inference by unfolding a recurrent architecture for a certain number of time steps that includes a cascaded feedback signal guiding shallower layers to learn more discriminative adaptive features. This is achieved through a carefully designed top-down structure that allows all deeper layers the potential to influence feedforward inference. Ablation studies and associated analysis reveal a strong capacity for spatial and categorical ambiguity to be resolved across feature layers and over space with rapid convergence on an optimal decision. Furthermore, DIGNet presents promising potential for improving inference capability of semantic segmentation in challenging cases with a feedback guided iterative inference mechanism.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for scene segmentation. *TPAMI*, 2017. 1, 2

[2] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. In *FG*, 2017. 2, 4

[3] H. Caesar, J. Uijlings, and V. Ferrari. Coco-stuff: Thing and stuff classes in context. In *CVPR*, 2018. 5, 7

[4] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *CVPR*, 2016. 2

[5] S. Chandra and I. Kokkinos. Fast, exact and multi-scale inference for semantic image segmentation with deep gaussian crfs. In *ECCV*, 2016. 7

[6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In *ICLR*, 2015. 1, 2, 7

[7] 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. *TPAMI*, 2018. 1, 2, 5, 6, 7

[8] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 5

[9] 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. 1

[10] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *CVPR*, 2018. 7

[11] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 2015. 5

[12] G. Ghiasi and C. C. Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *ECCV*, 2016. 1, 2, 7

[13] C. D. Gilbert and W. Li. Top-down influences on visual processing. *Nature Reviews Neuroscience*, 2013. 2

[14] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 6

[15] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015. 4

[16] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *TPAMI*, 2015. 5

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 2, 6, 7

[18] H. Hu, Z. Deng, G.-T. Zhou, F. Sha, and G. Mori. Labelbank: Revisiting global perspectives for semantic segmentation. *arXiv:1703.09891*, 2017. 7

[19] Q. Huang, W. Wang, K. Zhou, S. You, and U. Neumann. Scene labeling using gated recurrent units with explicit long range conditioning. *arXiv:1611.07485*, 2016. 2

[20] M. A. Islam, M. Rochan, N. D. B. Bruce, and Y. Wang. Gated feedback refinement network for dense image labeling. In *CVPR*, 2017. 1, 2, 5, 6

[21] X. Jin, Y. Chen, Z. Jie, J. Feng, and S. Yan. Multi-path feedback recurrent neural networks for scene parsing. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. 1, 2, 4, 6

[22] R. Karim, M. A. Islam, and N. D. B. Bruce. Recurrent iterative gating networks for semantic segmentation. In *WACV*, 2019. 1, 2, 3, 4, 6

[23] J. U. Kim, H. G. Kim, and Y. M. Ro. Iterative deep convolutional encoder-decoder network for medical image segmentation. In *EMBC*, 2017. 2

[24] S. Kong and C. C. Fowlkes. Recurrent scene parsing with perspective understanding in the loop. In *CVPR*, 2018. 2

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 2

[26] V. A. Lamme and P. R. Roelfsema. The distinct modes of vision offered by feedforward and recurrent processing. *Trends in neurosciences*, 2000. 2

[27] K. Li, B. Hariharan, and J. Malik. Iterative instance segmentation. In *CVPR*, 2016. 1, 2

[28] Q. Li, Z. Li, L. Lu, G. Jeon, K. Liu, and X. Yang. Gated multiple feedback network for image super-resolution. In *BMVC*, 2019. 1

[29] X. Li, Z. Jie, J. Feng, C. Liu, and S. Yan. Learning with rethinking: Recurrently improving convolutional neural networks through feedback. *Pattern Recognition*, 2018. 1, 2, 4, 6

[30] X. Li, Z. Liu, P. Luo, C. Change Loy, and X. Tang. Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade. In *CVPR*, 2017. 7

[31] M. Liang, X. Hu, and B. Zhang. Convolutional neural networks with intra-layer recurrent connections for scene labeling. In *NIPS*, 2015. 1, 2

[32] G. Lin, A. Milan, C. Shen, and I. Reid. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017. 1, 2, 6, 7

[33] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hayes, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 1

[34] W. Liu, A. Rabinovich, and A. C. Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015. 5

[35] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1, 2, 6

[36] L. McIntosh, N. Maheswaranathan, D. Sussillo, and J. Shlens. Recurrent segmentation for variable computational budgets. In *CVPRW*, 2018. 2

[37] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 1, 2, 6

[38] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *CVPR*, 2017. 7

[39] P. H. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *ICML*, 2014. 2

[40] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *ECCV*, 2016. 2

[41] F. Shen, R. Gan, S. Yan, and G. Zeng. Semantic segmentation via structured patch prediction, context crf and guidance crf. In *CVPR*, 2017. 7

[42] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta. Beyond skip connections: Top-down modulation for object detection. *arXiv:1612.06851*, 2016. 1, 2

[43] B. Shuai, Z. Zuo, B. Wang, and G. Wang. Scene segmentation with dag-recurrent neural networks. *TPAMI*, 2017. 7

[44] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 1, 2

[45] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*. 1, 2

[46] A. Veit and S. Belongie. Convolutional networks with adaptive computation graphs. *arXiv:1711.11503*, 2017. 2

[47] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell. Understanding convolution for semantic segmentation. In *WACV*, 2018. 7

[48] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1568–1576, 2017. 2

[49] Z. Wu, C. Shen, and A. v. d. Hengel. Bridging category-level and instance-level semantic image segmentation. *arXiv:1605.06885*, 2016. 7

[50] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 1, 7

[51] A. R. Zamir, T.-L. Wu, L. Sun, W. B. Shen, B. E. Shi, J. Malik, and S. Savarese. Feedback networks. In *CVPR*, 2017. 2, 4

[52] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017. 5, 6, 7

[53] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 5, 7