Gated Recurrent Networks for Scene Parsing

by

Rezaul Karim

A thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

Department of Computer Science
University of Manitoba
Winnipeg, MB, Canada

Thesis advisor

Dr. Neil Bruce

Author

Rezaul Karim

Gated Recurrent Networks for Scene Parsing

Abstract

In this thesis, we consider the problem of feedback routing and gating mechanisms in deep neural networks for dense pixel labeling tasks including scene parsing and semantic segmentation. The goal of semantic segmentation is to label every pixel in an image or video frame according to a specific set of object classes while scene parsing involves labeling both objects (e.g. person, car) and stuff (e.g. sky, road, field). Semantic segmentation and scene parsing have a wide variety of practical application including robot navigation and for autonomous vehicles. Recently there has been great progress with deep convolutional neural network-based solutions for scene parsing and semantic segmentation through increasing the depth and architectural complexity of the networks. Current successful feedforward architectures lack recurrent feedback connections that allow for information routing and dynamics, a phenomenon that is ubiquitous in the human brain. Such networks are reaching towards a limit on performance of inference capabilities possibly due to their implementation involving a single feedforward pass. Motivated by the dynamics of feedforward and recurrent processing in the brain, we propose a recurrent feedback gating mechanism that allows strong inference to be possible in an iterative manner. Our initially proposed Recurrent Iterative Gating Networks (RIGNet) reveal the powerful capability

of feedback to improve the inference capability of almost any network. Based on this observation, we later propose Distributed Iterative Gating Networks (DIGNet), which can be considered as a canonical feedback routing mechanism with appropriate gating modules, capable of boosting inference capabilities to an even greater extent than RIGNet. Experimental results on several benchmark datasets demonstrate the effectiveness of feedback gating in deep neural networks for scene parsing and the superiority of the proposed feedback gating mechanism.

Contents

List of Figures

List of Tables

Acknowledgments

At first of all, I offer my heartfelt gratitude to the Almighty Allah who gave me the ability to complete my whole thesis work without any breaking.

I would like to have special thanks to my advisor Dr. Neil Bruce. His thoughtful insight to solving problems has continuously helped me to find solutions presented in this thesis. He is an extremely smart and kind man who has always been very supportive of me. It has been an absolute pleasure to work with him.

I would also choose this moment to thank my committee members, Dr. Yang Wang, Dr. James Young, and Dr. Ahmed Ashraf who have provided valuable suggestions and constructive feedback in perfecting my thesis. Thank you very much for your time.

I also take this opportunity to express my gratefulness to - The University of Manitoba, Faculty of Graduate Studies, The Government of Manitoba (MGS funding), and NSERC Canada Discovery Grants program for their continuous financial support.

I consider myself very lucky and honored to have such fine, brilliant people in the lab leading me through the accomplishment of this thesis. Special thanks to Md Amirul Islam.

I would like to express my hearty complement to my parents and family members for their encouragement to come at this stage and pursue my dream. I have a special thanks for Tabassum Nijhum and Ayaan Muntasir for being my inspiration.

Last but not the least I would like to express my gratitude to all the people who have supported me along the way.

This thesis is dedicated to my parents
for their love, endless support
and encouragement.

Publications

Some of the ideas, materials and figures in this thesis have appeared previously in the following publications, pre-prints, and submitted manuscripts:

1. **Rezaul Karim**, M. A. Islam, and N. Bruce. Recurrent Iterative Gating Networks for Semantic Segmentation. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Hawaii, USA, January 2019.

Chapter 1

Introduction

In recent years, deep learning models have achieved significant success for problems involving dense pixel labeling [41; 9; 45; 1; 63; 16; 26; 38; 10] with a wide range of associated applications[33; 37]. Improvements in this domain have come by virtue of increasingly deep networks [31; 55; 57; 21], pre-training that leverages data from multiple datasets [12; 39] to boost overall performance, and innovations on architectural properties of networks. In this thesis, we focus heavily on the last of these categories in proposing a scheme for efficient selection and routing of feedforward information in neural networks.

The recent architectural improvement of deep neural networks for scene parsing has mostly been accompanied by increasing the depth of the network for strong inference, using atrous convolution to retain spatial resolution[10], and feature pyramids[10; 66] to capture properties of objects of different scale. While these architectural improvements have been able to gradually improve inference capability, we may go one step further with feedback gating that is beyond the limit of possibility with solely

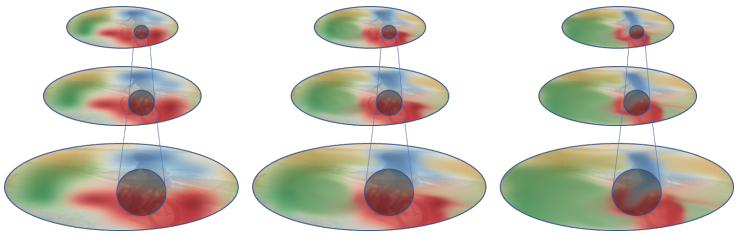

Figure 1.1: **A recurrent iterative gating based model**. A conceptual illustration of how higher layers of the network influence lower layers by gating information that flows forward. When applied iteratively (left to right), this results in belief propagation for features in ascending layers, that propagates over iterations both spatially and in feature space.

feedforward computation.

There is a good reason to believe that solutions with even stronger inference capability might be presented with appropriate feedback routing and gating mechanisms. Evidence of this comes from both examples of existing neural networks that consider such principles [15; 49; 42; 26; 38; 25; 27; 7; 6; 65], and also the very significant role that recurrence and gating play in biological vision systems as a mechanism for task, context or input dependent adaptation[32]. The principal motivating factors behind considering recurrent processing with feedback gating are as follows:

1. Simultaneous decision making for all spatial locations in single feedforward pass presents a risk for the inference capabilities of a network tied to relative spatial and semantic context. This can be overcome in iterative inference with the careful guidance of semantic context through the means of feedback signals from later layers to earlier layers.

2. Due to decreasing spatial resolution in deeper layers, inference made in one pass at the end of the deep network may result in globally consistent results but may have local inconsistencies. Iterative inference with feedback signals with proper

gating mechanisms may help the earlier layers in the next pass to make locally consistent feature activation and associated inference.

3. In the case that local features in a scene are diagnostic of semantic category, there should be a mechanism for this local diagnosticity to propagate outwards both in features carried forward and to envelop a larger spatial extent (see Fig. 1.1). That is, belief in a semantic concept may propagate spatially by virtue of the recurrent gating mechanism.

4. To ensure that activation among earlier layers is consistent with the higher-level interpretation reached by later layers. For example, if a deep layer signals with high confidence that a face is present in some location, features that carry more importance for faces should be emphasized and features that are not related to faces should be suppressed in that location.

We initially propose a *Recurrent Iterative Gating Network*, RIGNet with a simple but quite powerful feedback mechanism. Through this simple RIGNet formulation, we show that a wide range of different network architectures reveal better performance through the use of feedback gating. Specially, RIGNet is found to be able to recover missing parts and resolve categorical ambiguity with respect to surrounding context. Moreover, we also show that simpler versions of networks (e.g. ResNet-50) can be as capable as much deeper networks (e.g. ResNet-101) when modified to form a RIG-Net.

We further propose *Distributed Iterative Gating Networks*, DIGNet based on a symbiotic combination of propagator and modulator nodes that is both very flexible

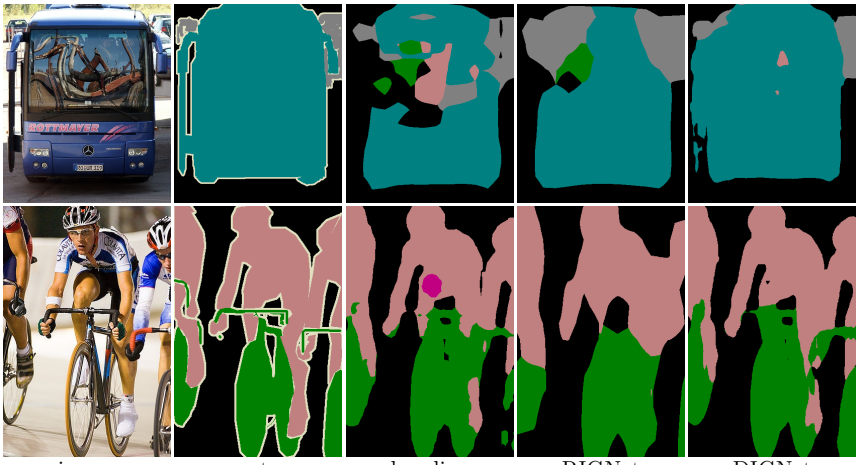

Figure 1.2: Examples of inference improvement from vanilla ResNet to **RIGNet** and **DIGNet**. **RIGNet** is able to recover missing detail and semantic correction but suffers from poor localization near sharp object boundaries. **DIGNet** can further improve object boundaries and resolve categorical ambiguity with feature refinement by propagating modulating signal in the form of a compact hypercolumn representation.

and highly efficient with respect to allowing information represented in one part of the network to reach other layers. One noticeable failure case of RIGNet involving poor localization and sharp boundary detection is further improved with the proposed DIGNet formulation. 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 a globally consistent prediction (see Fig. 1.2 and 1.3).

1.1 Contributions

We summarize our main contributions as follows:

- First, we present a network mechanism that involves *Recurrent Iterative Gating*

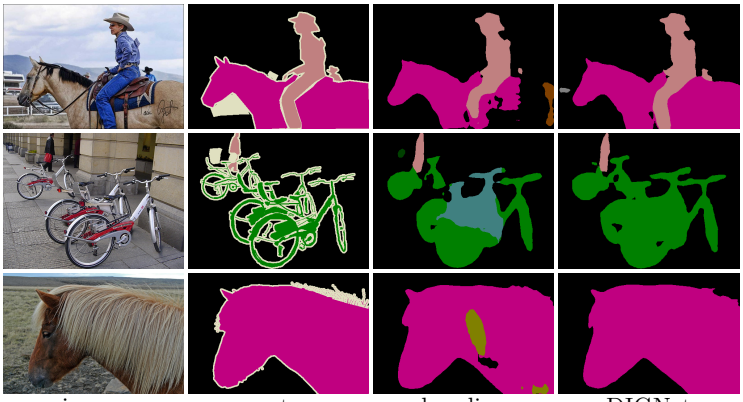

Figure 1.3: Examples of **DIGNet** predictions. DIGNet iteratively improves predictions by refining spatial detail and diminishing representational ambiguity within the network over a single iteration. This refines boundaries of objects, and fills in missing object parts from the initial feedforward pass (**1st** row). The mechanism also succeeds in resolving categorical ambiguity through refinement (and **2nd** and **3rd** rows).

for semantic segmentation, showing it is a valuable contribution in its compatibility with virtually any feedforward neural network providing a general mechanism for boosting performance. Our experimental results on two challenging datasets demonstrate that the proposed model performs significantly better than the baselines. With the experiments on this network, we also found that feedback based approaches are capable of developing a meaningful coarse-to-fine representation without bringing much complexity into the network architecture. As one specific example, ResNet-50 as a RIGNet is shown to outperform ResNet-101.

- We present a canonical feedback routing mechanism where the feedback signal carries information from all subsequent stages working like an incremental compact hypercolumn representation. The proposed scheme, *Distributed Iterative*

Gating Network also called DIGNet, is shown to greatly increase the inference capabilities of a variety of archetypal networks across different baselines. This becomes possible through a carefully designed top-down structure that allows all deeper layers the potential to influence feedforward inference. Analysis reveals a strong capacity for spatial and categorical ambiguity to be resolved across feature layers and over space with rapid convergence on an optimal decision.

- The proposed feedback gating architectures have been found to significantly boost the performance of a number of archetypal models across different benchmarks, while also allowing simple networks to outperform deeper networks with similar structure or those that incorporate significantly more complex architectures. This has important implications for semantic segmentation and also in how the nature of recurrent processing in general is viewed. Moreover, this has the possibility to extend to any network that solves a dense image labeling task.

1.2 Thesis Organization

The remainder of the thesis is organized as follows. In Chapter 2, we review related works and summarize important contributions in the literature related to this thesis. This includes a brief description of previous works for dense image labeling tasks and related works with gating and iterative refinement based approaches. In Chapter 3, we describe *Recurrent Iterative Gating Networks* as a simple but quite powerful feedback routing and gating mechanism for iterative inference. The presented recurrent feedback mechanism provides insights to the promising potential of feedback gating in deep neural networks for dense labeling tasks. We also describe

extensions of the basic Recurrent Iterative Gating Network mechanism achieved by generating a feedback signal from multiple stages that can further improve the inference capability of iterative solutions. In Chapter 4, we present a canonical feedback routing mechanism, *Distributed Iterative Gating Networks*, with a combination of two types of gating modules that is developed on the foundational basis of the Recurrent Iterative Gating Network. We further present extensive experimental results on several datasets involving semantic segmentation and scene parsing to demonstrate the effectiveness of the proposed feedback routing mechanism. Finally, we conclude this thesis in summarizing important findings and contributions, and also discuss possible future directions in Chapter 6.

Chapter 2

Related Works

Recent state-of-the-art semantic segmentation networks [41; 51; 9; 45; 1; 16; 26; 38; 10; 28] typically follow the structure of a Fully Convolutional Network (FCN). Although the feature maps produced in the higher-layers of conventional CNNs [31; 55; 57; 21] 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 in the higher layers.

2.1 Efforts at refinement

There have been several proposals to recover spatial resolution and fine object boundaries in the output segmentation map. Some proposals use Conditional Random Fields (CRFs) as a post-processing module[9] which adds huge computational overhead at the benefit of slightly improved inference. DeconvNet and SegNet are methods proposed that add a decoder module[46; 59] while having learnable param-

eters to gradually recover spatial resolution instead of one step upsampling. Another notable proposal was to reduce the stride and use dilated convolution in deeper layers to retain spatial resolution[10].

Another interesting challenge in semantic segmentation is that, there might be objects at different scales and sizes in the input image. DeepLab[10] and PSPNet[66] have attempted to overcome this with pyramid modules on top of a feature extractor. The architecture of pyramid modules differs in the sense that DeepLab has used several parallel branches with different types of dilation of convolution kernels and PSPNet uses several parallel branch with different adaptive average pooling strategies.

The earlier layers of a deep neural network have high spatial resolution and capture fine details while the deeper layers have more abstract features with lower spatial resolution. To combine the best of both these properties, there have been several attempts to combine features from multiple stages with skip connections[51]. Most recently, gating mechanisms to combine features from multiple stages[26] have been found to improve overall quality of prediction and provide strong inference in the case of ambiguous context like horse vs cow.

2.2 Approaches involving feedback

Several efforts [65; 43; 49; 29; 30; 34; 33; 58] have been proposed to iteratively improving the quality of inference beyond what is possible in a single feed forward pass. Several works consider employing recurrent processing [49; 37] or feedback based attention mechanisms [34] in combination with conventional CNNs, the value of which are evident in the similar mechanisms of processing observed in human brains [17; 32].

Another line of work [43; 65] applies a recurrent module (e.g. ConvLSTM) on top of the network to iteratively refine the initial prediction. Although feed-forward gating mechanisms [26] have shown good success for recognition tasks, recurrent feedback mechanisms play an important role in pushing performance further for several tasks of interest [65; 6; 2].

Related to our proposed approach is the idea of learning a feed-forward network in a iterative manner by propagating feedback in a top-down fashion. Recent feedback based approaches [65; 34; 53] follow the pipeline of correcting an initial prediction by propagating feedback in a few different ways. TDM [53] 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 [38; 45; 26; 50].

Our proposed approach differs from the above feedback based networks in how the feedback routing interacts with different network components and how gating plays a role to propagate feedback and modulate intermediate features. In summary, our feedback mechanism guides earlier features based on the feedback signal which has information from the layer immediately above, and implicitly 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.

Chapter 3

Recurrent Iterative Gating Networks: RIGNet

In this chapter, we present a recurrent iterative gating mechanism with different recurrent unrolling schemes. We also highlight some theory towards the logical explanation for the recurrent gating module and several core advantages of different unrolling schemes. Finally, we propose our top-down feedback based *Recurrent Iterative Gating Networks*, the RIGNet for semantic segmentation.

We first begin with the basic RIGNet with a block-wise/stage-wise recurrent feedback mechanism. We integrate iterative gating modules inside the feed-forward network which can be seen as rerouting the captured information based on features that flow in a backward direction. For this basic RIGNet, We explore two different unrolling mechanisms to control the flow of information in a top-down manner, these are sequential unroll and parallel unroll.

Next, we extend our idea of a recurrent iterative mechanism with multi range

gating using an additional long range feedback taken from the output of the last stage. The additional long range feedback is combined with the short range block/stage-wise feedback to generate a single feedback signal and subsequently all computation for feature re-weighting is similar to that of the basic RIGNet.

The core elements of the proposed RIGNet involve recurrent connections that control the flow of information in neural networks in a top-down manner. In this iterative mechanism feedback signals modulate or re-weight features that propagate over a number of iterations. This mechanism has broad compatibility with common existing networks and reveals the powerful capability of feedback to generate predictions that are more semantically correct than their feed-forward counterparts.

We also show that more shallow networks (e.g. ResNet50) with feedback gating may be made to perform better than much deeper networks (e.g. ResNet101) that do not include feedback modules. This proposed approach can be thought of as a novel formulation of feedback based recurrent design on deep convolutional neural networks that can emulate attentive vision to facilitate top down attention and improve spatial and semantic context for inference.

3.1 Overview of RIGNet Formulation

The process involved in iterative feedback based approaches has few key elements: (1) Output from some layers/blocks/stages in the network is fed-back to some earlier layers through a gating module where the simplest gate may be an identity transform or skip connection (2) The feedback is combined with a representation at an earlier layer through concatenation/multiplication/addition to generate the input for

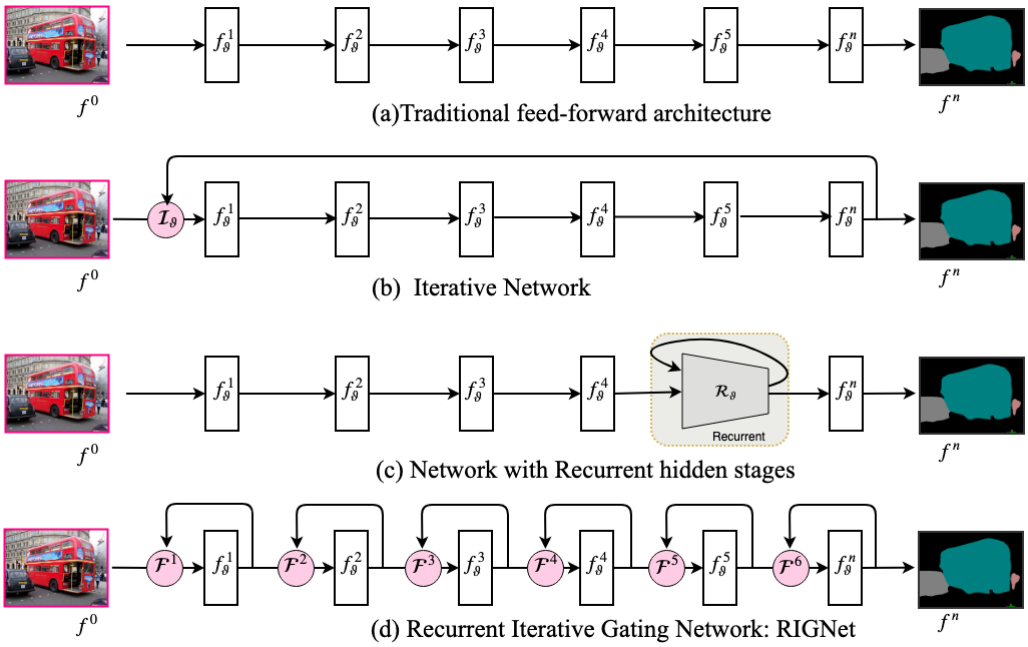

Figure 3.1: Illustration of (a) traditional iterative feed-forward network (b)traditional iterative network (c) network with one or more recurrent unit in hidden stages (d) our recurrent iterative gating network for semantic segmentation. In contrast to previous works, our framework involves recurrent iterative gating that control the flow of information passed forward in a top-down manner.

next iteration and (3) The final output is generated at the end of the last iteration. The input image undergoes shared convolutional stages repeatedly to predict labels at each step. The premise behind this is that iterating the feed-forward modules (most cases RNN) a few times produces a different final output at the last iteration. However, there is no backward interaction involving the feed-forward modules. In the proposed RIGNet architecture, we take the output of each block of layers as feedback, modulating the signal that forms the input of that block. The outcome of our recurrent feedback based approach is seen to be beneficial in few respects: (a) modulate the initial input with the feedback signal to emulate attentive vision

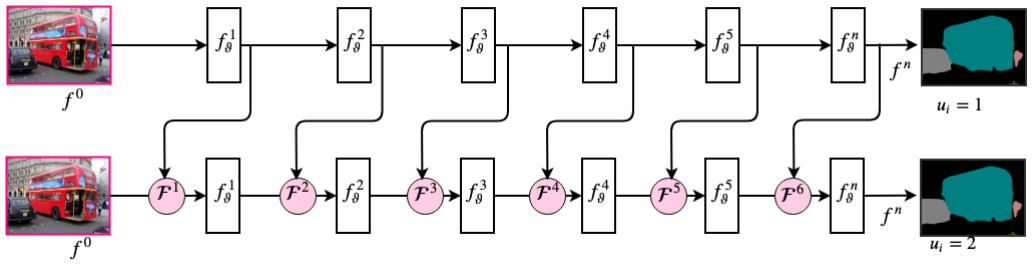

Figure 3.2: Our proposed network when unrolled in different time-steps ($u_i = 1, 2..$). The difference with the traditional approach is that the loop connections feed into layers that share parameters with previous layers in addition to gating being controlled top-down. Consider iteration $u_i = 1$ where the feed-forward network receives the input image and predicts the initial output which is gated with the corresponding iterative gating module in iteration $u_i = 2$.

(b) a compact representation that can compete performance-wise with deeper architectures through unrolling for more iteration (c) this allows a common architecture suitable for a wide range of devices with different computational throughput by varying the number of iterations (d) this introduces a hierarchical structure that leads to an implicit coarse-to-fine representations to improve spatial and semantic context of the inference. Overall, this allows for refinement of feature specific activations, and confidence for categories separated by space to propagate over the image.

An illustration of the RIGNet architecture is shown in Fig. 3.1. Unlike existing work, we integrate iterative feedback modules inside the feed-forward network which can be seen as rerouting the captured information based on information that flows in a backward direction. At a high-level, RIGNet mimics the cyclical structure of the human brain which is created by integrating *iteration* in the feedback modules of a feed-forward network.

3.2 Iterative Gating Mechanism

In this section, we share the details of our proposed recurrent iterative gating/feedback mechanism which is based on stacking feedback modules in each stage of the feed-forward network. Recent works on semantic segmentation that have shown success typically share common strategies involving dilation [10; 64; 23], encoder-decoder structure [45; 25; 1], and coarse-to-fine refinement [25; 50; 7; 38] to balance semantic context and fine details by recovering per-pixel categorization. Most of these approaches are increasingly precise in their performance but introduce additional model complexity. Our main objective is to demonstrate a compact representation of complex deep networks which can achieve similar performance and is agnostic to the network it is paired with. To accomplish this, we propose a network with several iterative feedback modules (shown as a feedback gate in Fig. 3.1) similar to recurrent neural networks. More specifically, we apply recurrent top-down feedback in a block-wise/stage-wise manner rather than layer-wise [36; 37] or over the full network [49]. We argue that formulating top-down feedback layer-wise similar to [36; 37] suffers from few major drawbacks: (1) adds a huge overhead in number of parameters (2) limiting for transfer learning (3) the output of a single deeper layer may not contain sufficiently rich semantic information for feedback. Moreover, the recurrence over the whole network [49] is also unable to leverage semantic and spatial contextual information through feedback to refine the previous layers. In this context, the behaviour of filters remains fixed and only varies based on the current label hypotheses rather than any internal feature representations. In contrast, in RIGNet feedback is taken from the output of a block of convolution layers resulting in a larger effective field

of view and more abstraction. Also, this mechanism allows the lower-level layers to be influenced by the weight/activation of higher-level features resulting in refined weight/activation in the earlier layers that refines information passed forward and importantly also allows for spatial propagation through internal feature representations throughout the network. The mechanism is further defined by a hyper-parameter (u_i) denoted as the *unroll iteration* of the network that determines the number of times the feedback loop will be instantiated to predict the final output. The unrolling parameter (u_i) can be viewed the same as the unrolling steps in RNNs. Fig. 3.2 shows the unrolling effect on the network presented in Fig. 3.1 (c).

The final output is generated with multiple iterations over the network where the number of iterations is determined by the unroll iterations (u_i). In the very first iteration, all feedback gates remain disconnected from the network (i.e. there is no gating without prior knowledge of the image among subsequent layers). In subsequent iterations, the output of the previous iteration is subject to modulation by the feedback gate at every layer in the most extreme case all the way from the first layer to the deepest layers. This operation repeats for all the stages of the network to obtain a prediction for the current iteration. The reverse information flow towards the input all the way from output layers allows the earlier layers to be implicitly subject to adjusted parameters at the time of inference to provide guidance to remove ambiguity that may arise anywhere within the network. Similarly, the loopy structure inside the feedback modules allows a shallower network to be influenced by a rich feature representation. The iterative nature allows for stage-wise refinement and spatial propagation of refinement. This implies that much simpler networks can

perform more powerful inference given that their behaviour is not fixed and can be modulated by recurrent feedback.

3.3 Unroll Mechanism

For a recurrent approach to be practical, there is a need for a specific recurrent structure (and implied unrolling mechanism) and also a constraint that the unrolling iterations should be finite. It is important to note that often only a finite number of iterations is of value given that the final output will tend to converge on a certain set of labels and show little change beyond a fixed number of iterations. Note also that when we set the unroll iteration $u_i = 1$, RIGNet becomes a purely end-to-end feed-forward neural network. We explore two different unrolling mechanisms to control the flow of information in a top-down manner.

3.3.1 Sequential Unroll

Sequential unrolling involves the recurrence within block tied to a particular iteration, with the order of recurrence following subsequent blocks all involved in a single feedforward pass. Activations are forwarded to the next recurrent block when iterations in all previous network blocks within a single recurrent iteration are finished. Fig. 3.3 depicts the sequential unrolling mechanism when the network is unrolled for three iterations ($u_i = 3$). Conceptually, the sequential unroll mechanism increases the effective depth of the unrolled network by a multiplicative factor. For example, given a network with l_t layers in the RIGNet formulation, l_f of l_t layers remain similar whereas we introduce l_r layers within the blockwise recurrent feedback stage. So, the

feed-forward network has effective depth similar to the original depth of the network ($l_e = l_f + l_r$) however the RIGNet with unroll iteration u_i has effective depth of $l_e = l_f + l_r * u_i$.

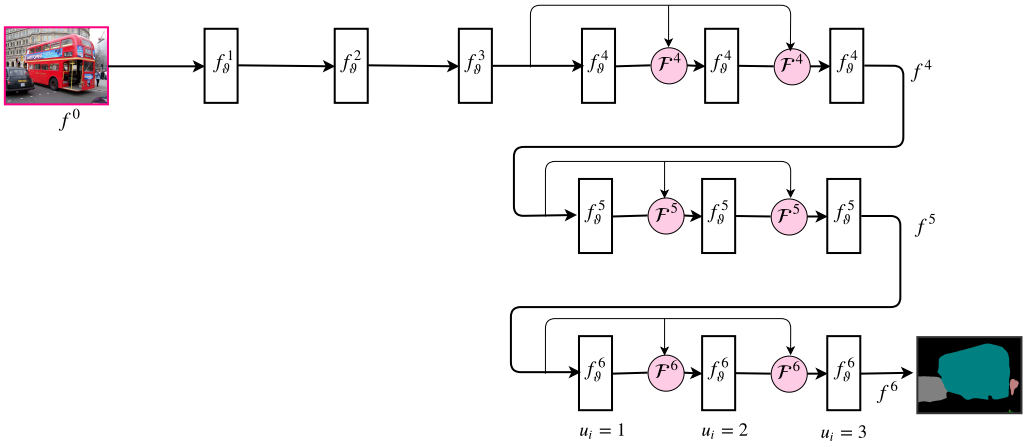

Figure 3.3: Illustration of RIGNet with sequential unroll for three iteration in last three feed-forward blocks. Note that f_ϑ^i refers to a feed-forward block and $\mathcal{F}^i$ denotes the recurrent feedback gate.

3.3.2 Parallel Unroll

In the parallel unrolling mechanism, the network initially gathers the final representation (activations) of the first iteration and then recurrence proceeds by way of feedback from the first block to the last (deep $\rightarrow$ shallow). The formulation of obtaining the final representation in subsequent iterations is similar to the first iteration. Our proposed RIGNet in Fig. 3.2 is a feed-forward network with a parallel unrolling mechanism. Fig. 3.4 illustrates the parallel unrolling mechanism where a RIGNet with feedback in last three blocks is unrolled for three iterations. The parallel unrolling mechanism has less effective depth compared to the sequential one since

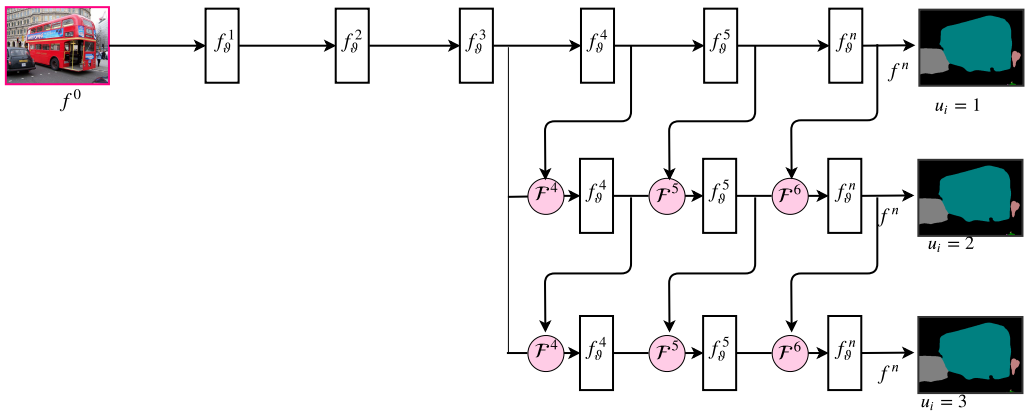

Figure 3.4: Illustration of RIGNet with parallel unroll in the last three feed-forward blocks for three iterations.

it increases the effective depth multiplicatively only for the last feedback block. To continue with the example in Sec. 3.3.1, assume the last feedback block has l_{rk} layers among l_r whereas the remaining blocks with feedback have $l_{rj} = l_r - l_{rk}$ layers. In this case the effective depth of the RIGNet with parallel unrolling is $l_e = l_f + l_{rj} + l_{rk} * u_i$. This depth analysis between sequential and parallel unrolling mechanisms reveals the impact of increasing the effective depth on performance improvement. This analysis also sheds light on the hypothesis that an effective *depth increase* helps to improve the overall performance as opposed to the role of long range semantic context (across layers and spatially) through feedback bringing improvement. Additionally, parallel recurrence brings another advantage of generating a early prediction (coarse representation) at the end of each iteration which can be used to facilitate taxonomy learning where some coarse grained predictions can be made at initial iterations (like vehicles) and then fine predictions in final iterations (like cars/bus). This can be done by backpropagation of loss defined by coarse predictions in initial iteration similar to the

concept explored in [65] for classification task. Since we are principally interested in the final prediction, we only backpropagate loss once at the end of final iteration, but it should be noted that the generality of the RIGNet structure presents a wide range of directions that may be explored further.

3.4 Iterative Gating Module Design

In this section we present the formulation for both our basic feedback gate. Each feedback gate module takes the output of the forward block (next stage feature map) f_{ϑ}^{i+1} as input and learns to pass information relevant to gating backwards through the following sequence of operations: First we apply an average pooling (resultant feature map $f_{\vartheta}^{i'+1}$) followed by a 3×3 convolution to obtain a feature map $f_b^{i'+1}$ which is capable of carrying context with a larger field of view. We then apply a sigmoid operation followed by bilinear upsampling to produce a feature map whose spatial resolution is the same as the input to the feedback gate. The resultant feature map provides the input to the feedback gate. The i^{th} stage feedback gate $\mathcal{F}^{i+1}$ combines the inputs through an element-wise product resulting in a modulated feature map f_r^i. We can summarize these operations as follows:

$$f_b^{i'+1} = \mathcal{S}(\mathcal{C}_{3\times3}(\mathcal{A}_p(f_{\vartheta}^{i+1}); \Theta), \;\; f_b^{i'+1} = \xi(f_b^{i'+1})$$
$$f_r^i = f_b^{i'+1} \otimes f_{\vartheta}^{i-1} \tag{3.1}$$

where $\mathcal{A}_p$ represents an average pooling operation and (Θ) denotes the parameters of the convolution $\mathcal{C}$. ξ refers to upsampling operation.

3.5 Recurrent Iterative Gating Network Extension: RIGNext

Having the initial success with our basic stage wise feedback mechanism, we further explored in the direction of extending the idea of recurrent iterative gating with a more sophisticated feedback routing mechanism: RIGNet multi range, alternatively termed as RIGNext.

3.5.1 Multi Range Feedback

Our multi range recurrent feedback network as shown in Fig. 3.5 is most suitable as a parallel recurrence mechanism as it requires additional feedback from the logits of final stage. The concept of multi range feedback is more analogous to biological vision systems and has been considered to a limited degree in the context on image classification [44]. The major difference compared to the basic RIGNet is that the feedback signal or weights are not directly generated from the previous output of the current stage, rather it is generated by combining additional information from the logits of the last stage with the previous output of the current stage. The modulation of intermediate signals in the extended feedback routing mechanism, RIGNext, is the same as in the basic RIGNet.

3.5.2 Multi Range Feedback Gate

In multi range feedback routing, we face the requirement of an additional mechanism to combine two features/signals to generate the feedback signal. Instead of mod-

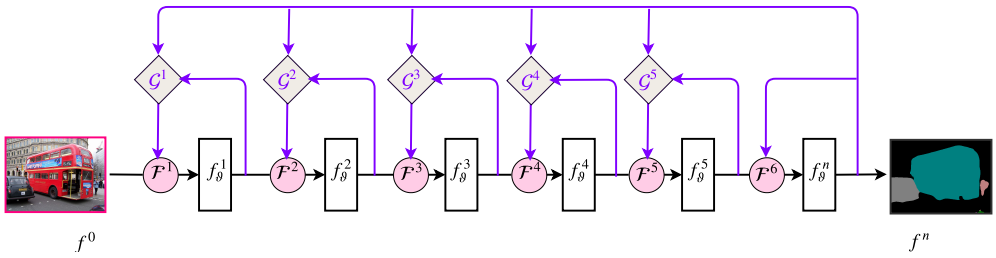

Figure 3.5: RIGNExt: **R**ecurrent **I**terative **G**ating **N**etwork **Ext**ension with multi range feedback routing mechanism.

ifying the existing feedback gate $\mathcal{F}$, we consider having an additional module/gate to generate the feedback signal. This design choice leaves us with simplicity of design and incremental improvement opportunity. We term this new module a feedback generator $\mathcal{G}$. Additional long range feedback is taken from the output of the last stage. For example, we denote the output from the last block as f_c^i at iteration i. The long range feedback f_c^i is first passed through convolution and an interpolation layer resulting in a feature map of $f_c^{i'}$ with a number of channels and spatial dimensions matching the short range feedback f_ϑ^{i+1} . We then concatenate $f_c^{i'}$ and f_ϑ^{i+1} and use a convolution operation to fuse the concatenated features to obtain a feature map F^i which is our feedback signal. These are the mechanisms involved in the feedback generator $\mathcal{G}$ (see Eq. 3.2). The generated feedback is then passed to the previously designed feedback gate $\mathcal{F}$ and the rest of the operations performed by the feedback gate are similar to the basic feedback gate with F^i instead of f_ϑ^{i+1}.

We can summarize the operations in the feedback generator $\mathcal{G}$ as follows:

$$f_c^{i'} = \xi(\mathcal{C}_{3\times3}(f_c^i; \Theta))$$
$$F^i = \mathcal{C}_{3\times3}(f_c^{i'} || f_\vartheta^{i+1}; \Theta) \tag{3.2}$$

where (Θ) denotes the parameters of the convolution $\mathcal{C}$. ξ refers to an upsampling operation.

3.6 Experiments

We perform a series of experiments to evaluate the performance of RIGNet. We begin by providing implementation details of RIGNet and baseline approaches. Initially, we perform a controlled study to analyze and investigate the contribution of each component in RIGNet. Then we experiment on the object-centric PASCAL VOC 2012 dataset [14]. In addition to semantic segmentation, we also explore how the recurrent iterative gating mechanism can improve scene parsing performance on the COCO-Stuff dataset [4].

3.6.1 Implementation Details and Baseline Networks

Our experimental pipeline and the pre-trained models are based on the open source toolbox PyTorch [47]. We begin with experiments using simpler networks and gradually move to more complex networks to show performance improvements related to different network architectures with the recurrent iterative gating mechanism. First, we report evaluation performance for the vanilla ResNet baselines denoted as

ResNet50-FCN, ResNet101-FCN and our corresponding proposed ResNet50-RIGNet, ResNet101-RIGNet networks. So given an input image $I \in \mathbb{R}^{h \times w \times d}$, the networks produce a feature map of size $\lfloor \frac{h}{32}, \frac{w}{32} \rfloor$. Then we extend our experiments with a more sophisticated network architecture to examine the effectiveness of RIGNet. We choose DeepLabV2 [10] as our new baseline network due to it's superior performance on pixel-wise labeling tasks. DeepLabV2 uses the dilated structure to balance the semantic context and fine details, resulting in a feature map of size $\lfloor \frac{h}{8}, \frac{w}{8} \rfloor$ given an input image $I \in \mathbb{R}^{h \times w \times d}$.

For ease of representation, we use the following notations to report numbers throughout the experiment section. Sequential unroll ($\mathbb{S}_u$), Parallel unroll ($\mathbb{P}_u$), Parallel unroll with multi-range feedback ($\mathbb{P}_u^f$), unroll iteration (u_i). $\mathbb{F}_b \ll j..i \gg$ refers to feedback used in blocks from i to j.

3.6.2 Dataset and Evaluation Metrics

We evaluate our proposed RIGNet architecture on the following benchmark datasets.

PASCAL VOC 2012: This semantic segmentation dataset consisting of 1,464, 1,449 and 1,456 images for training, validation and testing respectively, which includes 20 object categories and one background class. We use the augmented training set that includes extra labeled PASCAL VOC images [18].

COCO-Stuff: This is a recently released scene parsing dataset based on MS-COCO annotations. Following the split in [4], we use 9k images for training and another 1k for testing. We use the segmentation labels which contain a total of 182 categories including 91 things and 91 stuff classes.

We use standard dense labelling evaluation measures named Mean Intersection over Union (Mean IOU), Mean Class Accuracy (MCA) and per-pixel accuracy (PPA). To give definitions of the measures, let n_{ij} be the number of pixels having class label i that have been predicted as class j, and $n_i = \sum_j n_{ij}$ be the total number of pixels having class label i and K be the total number of classes. The metrics are defined as follows.

- **Class IoU):** $n_{ii}/(n_i + \sum_j n_{ji} - n_{ii})$
- **Mean IOU:** $(1/K) \sum_i n_{ii}/(n_i + \sum_j n_{ji} - n_{ii})$
- **Per-class accuracy:** n_{ii}/n_i
- **Mean Class Accuracy (MCA):** $(1/K) \sum_i n_{ii}/n_i$.
- **Per-Pixel Accuracy(PPA):** $\sum_i n_{ii}/\sum_i n_i$

3.6.3 RIGNet Ablation Studies

In this section, we perform ablation studies to investigate the role of the *recurrent iterative gating* mechanism. We highlight a few major facts to validate the design choices: 1) the role of applying iterative gating modules in different stages 2) the length of iteration in a gating module. 3) the design choice associated with the feedback gate 4) the influence of network-wide vs block-wise feedback mechanism.

Unroll Mechanism and Feedback Blocks

To evaluate the value of applying iterative gating modules in different convolutional stages, we perform a control study where we train models by adding iterative

gating modules step by step to different layers to evaluate their effect on performance. More specifically, we train a feed-forward network by adding feedback modules at the last stage only and compute mIoU for the final predictions. We repeat this operation several times until we reach the first convolutional stage to examine the importance of integrating recurrent iterative gating mechanisms at the final layer, many deep layers, or all layers. We first report mIoU for the vanilla ResNet baselines in Table 3.1. Table 3.2 shows the depth-wise performance of the ResNet50-RIGNet architecture on the PASCAL VOC 2012 validation set. From this analysis, it is clear that inclusion of iterative gating modules improves the overall performance gradually.

Methods	Parameters	Mean IoU
ResNet50-FCN	32-s	59.4
ResNet101-FCN	32-s	65.3

Table 3.1: Quantitative results in terms of mIoU on PASCAL VOC 2012 validation set for ResNet-FCN based baselines.

Methods	Feedback Blocks					
	$\ll 6 \gg$	$\ll 6..5 \gg$	$\ll 6..4 \gg$	$\ll 6..3 \gg$	$\ll 6..2 \gg$	$\ll 6..1 \gg$
$\mathbb{S}_u$	61.6	65.1	**65.4**	65.2	65.3	65.3
$\mathbb{P}_u$	62.3	64.8	**65.2**	64.9	64.8	65.1
$\mathbb{P}_u^f$	61.6	66.3	66.7	66.8	67.1	**67.2**

Table 3.2: Comparison of mIoU for different variants of our proposed ResNet50-RIGNet architecture.

Unroll Iteration

To justify the significance of an iterative solution, we examine the extent of the recurrent gating module in terms of iterations. Table 3.3 shows the experimental

results of the iterative gating module per the discussion. We keep the best performing combination for the three different scenarios.

Iter.	RIGNet ($\mathbb{S}_u$) $\ll$6..4$\gg$	RIGNet ($\mathbb{P}_u$) $\ll$6..4$\gg$	RIGNet ($\mathbb{P}_u^f$) $\ll$6..1$\gg$
2	65.4	65.2	67.2
4	68.3	68.3	68.5
6	**68.8**	**68.9**	**69.5**

Table 3.3: Comparison of mIoU for varying number of unroll iterations with ResNet50-RIGNet variants on PASCAL VOC 2012.

For this analysis, we compare evaluation performance for *unroll iter 2, unroll iter 4, and unroll iter 6*. We observe that overall performance progressively improves with each successive stage of iteration. We empirically found this observation to be valid across datasets and different network architectures. Interestingly, ResNet50-RIGNet with unroll iteration 3 outperforms the ResNet101-FCN which further validates the impact of increasing the number of iterations in recurrent gating.

Feedback Gate Design Choices

Additionally, concerning the design choice of feedback gate, we try a variety of alternative design choices and report the number in Table 3.4. When we use (additive + ReLU) interaction in recurrent gating modules, ResNet50-RIGNet achieves 64.2% mIoU on the PASCAL VOC 2012 validation set. In comparison, our proposed ResNet50-RIGNet with an (multiplicative + sigmoid) interaction in the gating modules achieves 65.2% mIoU. Multiplicative feedback routing is demonstrably valid from a performance point of view, but also intuitive in that it provides a stronger capacity to resolve categorical ambiguity present among earlier layers in the extreme case completely inhibiting activation in an earlier layer.

Methods	Add + ReLU	Mul + Sigmoid	Mul + Tanh
ResNet50-RIGNet	64.2	**65.2**	65.2
ResNet101-RIGNet	67.0	**68.9**	68.0

Table 3.4: Impact of the choice of activation function in iterative gating on PASCAL VOC 2012 validation set.

Moreover, we also investigate the impact of different pooling operations (w/o pooling, max pool, and avg pool) in the design choice of recurrent iterative gating. Table 3.5 presents the results of alternative design choice in terms of different pooling operations. For ResNet50-RIGNet all the three different design choices achieve similar mIoU on PASCAL VOC 2012 val set. Interestingly, the ResNet101-RIGNet with an average pooling in the recurrent gating achieves better performance compared to alternatives.

Methods	w/o Pooling	Max Pool	Average Pool
ResNet50-RIGNet	65.2	65.2	**65.2**
ResNet101-RIGNet	68.3	68.3	**68.9**

Table 3.5: Impact of choice of pooling operation in the iterative gating with $\mathbb{P}_u \ll 6..4 \gg$, $u_i = 2$.

Feedback: Network-wide vs Block-wide

In Table 3.6, we present the results comparing different feedback routing mechanisms shown in Fig. 3.1. Note that existing works incorporate network-wide recurrence [49] with a shallower base network and the results comply with our general intuition of the superiority of the gated block-wide feedback mechanism.

Methods	Recurrence Mechanism	mIoU
ResNet50-FCN	feed-forward	59.4
RCNN	Network-wide, similar to [49]	59.6
RCNN	Network-wide gated feedback	59.9
ResNet50-RIGNet	routing similar to [34]	65.0
ResNet50-RIGNet	$\mathbb{P}_u \ll 6..4 \gg$, $u_i = 2$	65.2
ResNet50-RIGNet	$\mathbb{P}_u^f \ll 6..1 \gg$, $u_i = 2$	**67.2**

Table 3.6: Comparison of network-wide vs block-wide recurrence based methods on PASAL VOC val. Note that we implement the ideas in [49] with ResNet50. Also, we adapt [34] by implementing their routing mechanism attached with our basic feedback gate.

3.6.4 Experiments on PASCAL VOC 2012

We evaluate the performance of our proposed recurrent iterative gating network on the PASCAL VOC 2012 dataset, one of the most commonly used semantic segmentation benchmarks. Following prior works [42; 10; 26], we use the augmented training set comprised of 10,581, 1449, and 1456 images in training, validation, and testing respectively. The models are trained on the augmented training set and tested on the validation set. Table 3.7 shows results for the comparison between our proposed approach and the ResNet baselines on the validation set.

Note that we only report the best result for ResNet50-RIGNet in Table 3.7 since the depth-wise results are already reported in Table 3.2. It is evident that, *ResNet50-RIGNet* and *ResNet101-RIGNet* outperform the baselines significantly in terms of mIOU achieving 68.9% and 71.6% respectively. It is worth mentioning that our proposed ResNet50-RIGNet yields mIoU better than ResNet101-FCN without any post-processing techniques providing a convincing case for the value of our iterative gating mechanism. We also experiment with ResNet101-FCN (stride 8) as a base

Method	Parameters	mIoU (%)
ResNet50	-	59.4
ResNet50-RIGNet	$\mathbb{P}_u \ll 6..4 \gg$, $u_i = 6$	**68.9**
ResNet101	-	65.3
ResNet101-RIGNet	$\mathbb{S}_u \ll 6..4 \gg$, $u_i = 6$	71.2
ResNet101-RIGNet	$\mathbb{P}_u \ll 6..4 \gg$, $u_i = 6$	71.4
ResNet101-RIGNet	$\mathbb{P}_u^f \ll 6..1 \gg$, $u_i = 6$	**71.6**
ResNet101 (8s)	-	71.3
ResNet101-RIGNet(8s)	$\mathbb{P}_u \ll 6..4 \gg$, $u_i = 4$	**74.9**
DeepLabV2	-	74.9
DeepLabV2-RIGNet	$\mathbb{P}_u \ll 6..4 \gg$, $u_i = 2$	**75.9**

Table 3.7: PASCAL VOC 2012 validation set results for several baselines and RIGNet with different unroll mechanisms.

network and achieve superior performance (71.3% vs. 74.9% mIoU) compared to the baseline. As shown in Table 3.7, *DeepLabv2-RIGNet* outperforms the baseline significantly which further validates the importance of iteratively refining initial outcomes through recurrent gating modules.

Figure 3.6 depicts a visual comparison of our approach with respect to the baselines. We can see that *ResNet50-RIGNet* is capable of producing predictions superior to ResNet101-FCN, and the RIG mechanism has a powerful impact on network performance for all cases.

3.6.5 Experiments on COCO-Stuff

To further confirm the value and generality of proposed *recurrent iterative gating* mechanism on scene parsing, we evaluate on the large-scale COCO-Stuff dataset. This dataset contains images of high-complexity including things and stuff. The

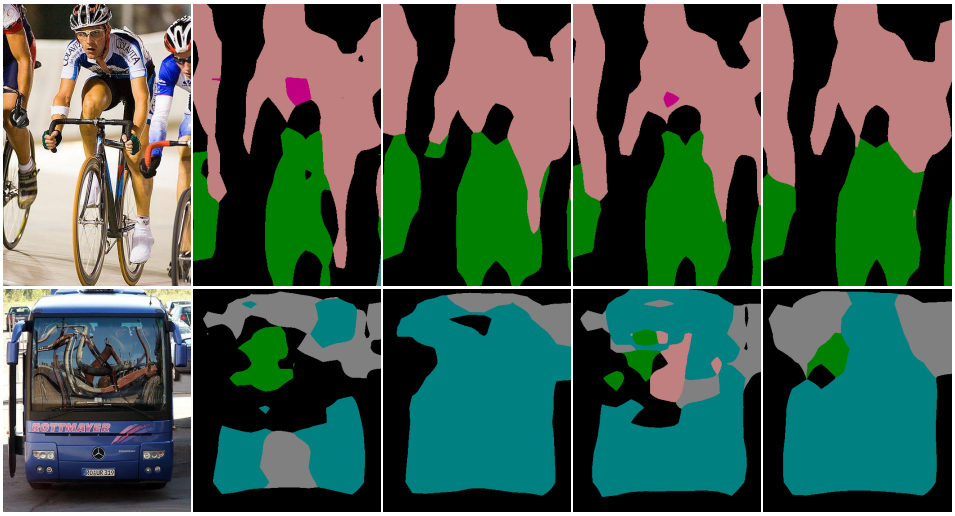

Figure 3.6: Qualitative results corresponding to the PASCAL VOC 2012 validation set for 2 iteration.

COCO-Stuff dataset extends the COCO annotation by adding dense pixel-wise stuff annotations and provides dense semantic labels for the whole scene, which has 9,000 training images and 1,000 test images. It includes total annotation of 182 classes consisting of 91 thing classes and 91 stuff classes.

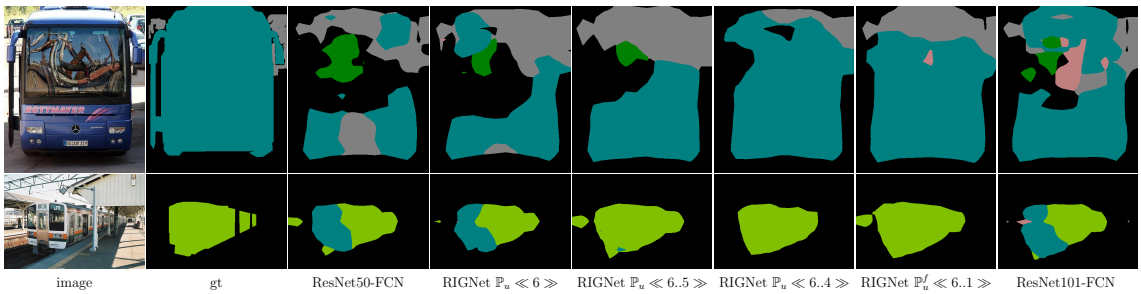

Figure 3.7: Some samples of output quality after stage-wise addition of recurrent iterative gating modules. For each row, we show the input image, ground-truth, the vanilla ResNet-50 prediction, the predicted segmentation map of ResNet in a top→down manner when iterative gating is included, and the output of vanilla ResNet101-FCN.

We use the same architectures and training procedures mentioned for evaluating

performance on the COCO-Stuff dataset. Table 3.8 shows the quantitative comparison of our approach with respect to vanilla ResNet-FCN and DeepLabv2 based baselines. Our proposed *ResNet50-RIGNet* achieves better mIoU (*28.8%* vs. *24.3%*) than the baseline. We further perform experiments on the COCO-Stuff 10k dataset with more sophisticated models (DeepLabV2-Res101). As shown in Table 3.8, RIGNet consistently outperforms the baselines by a significant margin.

Methods	Parameters	pAcc	mAcc	mIoU
ResNet50	-	57.2	35.2	24.3
ResNet50-RIGNet	$\mathbb{P}_u \ll 6..4 \gg, u_i = 2$	59.8	38.2	26.6
ResNet50-RIGNet	$\mathbb{P}_u \ll 6..4 \gg, u_i = 6$	61.4	40.0	**28.8**
ResNet101	-	58.7	38.2	26.4
ResNet101-RIGNet	$\mathbb{P}_u \ll 6..4 \gg, u_i = 2$	60.8	39.4	28.0
ResNet101-RIGNet	$\mathbb{P}_u \ll 6..4 \gg, u_i = 6$	62.6	41.1	29.5
ResNet101-RIGNet	$\mathbb{P}_u^f \ll 6..1 \gg, u_i = 6$	62.3	41.6	**29.9**
DeepLabV2	-	65.4	44.6	34.1
DeepLabV2-RIGNet	$\mathbb{P}_u \ll 6..1 \gg, u_i = 2$	66.1	46.6	**35.0**

Table 3.8: Comparison of several ResNet based networks w/o and w/ RIG on COCO-Stuff 10K validation set.

The superior performance achieved by RIGNet reveals that integrating recurrent iterative gating modules in the feed-forward network are very effective in capturing more contextual information for labeling complex scenes.

3.7 Discussion

Towards examining the practical grounds for recurrent iterative gating networks with top-down feedback, we aimed to verify a few specific hypotheses with our experiments. Firstly, the recurrent iterative gating mechanism can allow more parsimonious

networks to outperform deeper architectures with careful selection of the gating structure. This is revealed to be the case for the RIGNet architecture, with clear evidence of ResNet50-RIGNet outperforming ResNet101-FCN with three iterations.

Secondly, our proposed RIGNet is more precise and semantically meaningful compared to the baselines with respect to qualitative results presented in Fig. 3.6. Fig. 3.7 illustrates the impact of integrating a recurrent gating module. We can see that the recurrent iterative gating scheme progressively improves the details of a predicted segmentation map by recovering the missing spatial details which can be seen as coarse-to-fine refinement.

Moreover, the improvement in performance as a function of recurrent gating depth (RIG blocks) reveals that the RIGNet formulation of the feed-forward convolutional network improves the representational power of the model by incorporating semantic and relational context in a top-down manner. Results presented reveal the capability for correcting errors made in a single feedforward pass through Recurrent Iterative Gating as an exciting and important direction for future work, which even for deep networks, allows for stronger representational capacity.

While there may exist alternatives for feedback gate design, we present an intuitive way of designing recurrent feedback gates. The demonstrable capability of the proposed mechanism, this feedback mechanism inspired us toward developing a canonical architecture of feedback mechanism for dense image labeling which we are going to present in next chapter. We expect that this work will also inspire significant interest in exploring feedback based approaches within future work including an emphasis on top-down processing, gating, and iteration.

Chapter 4

Distributed Iterative Gating Network: DIGNet

In this chapter, we present a canonical structure for controlling information flow in neural networks based on a strategy of Distributed Iterative Gating (DIGNet). The proposed architecture of DIGNet is based on the theoretical foundation and feedback hypothesis developed with the previously presented RIGNet and motivated by the multi range feedback routing in the extended RIGNExt. The structure of this mechanism derives from a strong conceptual foundation, and presents a light-weight mechanism for adaptive control of feedforward computation compatible with virtually any existing neural network. This DIGNet formulation improves over RIGNet in terms of quality of inference, number of parameters, speed of computation and convergence over the number of unroll iterations.

To demonstrate DIGNet, we primarily focus on efficient feedback mechanisms coupled with feed-forward semantic segmentation frameworks [21; 10]. In general,

networks that include a feedback component [65; 33; 24; 34] have a standard feed-forward 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 iteratively as a correcting signal to guide the features in earlier layers based on high-level semantic representations. Some mechanisms for implementing such guidance through feedback involve either allowing direct influence of earlier layers [34] based on high-level semantic representations, or to guide each intermediate layer based on feedback from the next deepest layer in a top-down manner [29]. The typical feedback mechanism in each stage can be formulated as:

$$f^{i'} = f^i \odot \underbrace{\mathcal{F}(\mathbf{W_a} * f^{i+1})}_{\text{feedback signal}}, \; f^{i'} = f^i \odot \underbrace{\mathcal{F}(\mathbf{W_a} * f^c)}_{\text{feedback signal}} \tag{4.1}$$

where $f^{i'}$ is the updated(re-weighted) feature map at i^{th} stage; f^i stands for bottom-up feature map generated at i^{th} stage, $*$ denotes convolution, $\mathbf{W_a}$ are trainable weights, and $\mathcal{F}(.)$ refers to feedback from either the $(i+1)^{th}$ or last stage (f^c). Intuitively, features with higher i (deeper stage) tend to capture more semantically relevant information albeit with lower spatial resolution.

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 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 sections, we propose a new architecture called *Distributed Iterative Gated Networks* (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.

4.1 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. In particular, this is done in a manner where the decision of what information to pass from deeper layers is part of the training. We choose conventional feed-forward network architectures (e.g. ResNet101-FCN) as our backbone semantic segmentation network. The architecture of DIGNet is illustrated in Fig. 4.1.

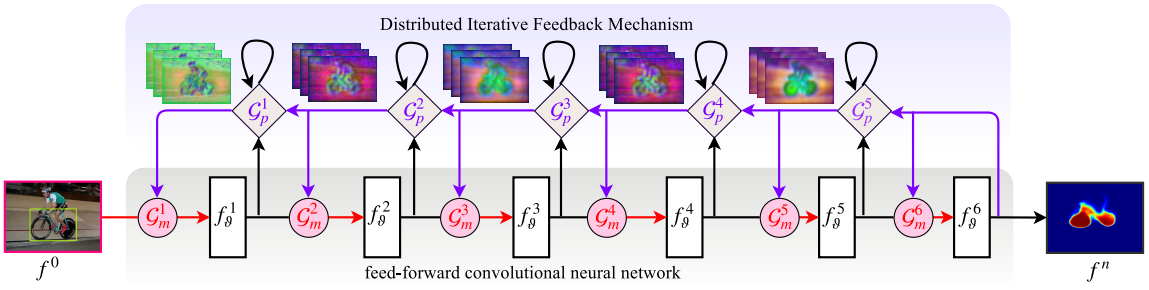

Figure 4.1: An illustration of our proposed **Distributed Iterative Gating Network** (DIGNet). DIGNet involves augmentation of a canonical neural network backbone through addition of gating modules, while operating in a recurrent iterative manner. $(f_\vartheta^1 \cdots f_\vartheta^6)$ are bottom-up 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. 4.3.

Our proposed feedback mechanism does not simply propagate high-level features into earlier layers in a top-down manner similar to [49; 3; 34; 29]. Instead it uses two different gating modules, (a *propagator* and a *modulator*) in each feed-forward stage to facilitate a broad exchange of information about internal representation within the network. The propagator gate guides the earlier layers by propagating feedback signals in a top-down manner while the modulator gate modulates inputs passed forward from each feed-forward stage accounting for the feedback signal from propagator gates. Details of the modulator and propagator gates are discussed in Sec. 4.3.

Our approach is motivated by the capacity to propagate more discriminating and semantically relevant information towards lower-layers which can be updated based on a subset of information from each downstream intermediate stage in a manner that is integrated with the training of the network. 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, which guides refinement of features at intermediate and early stages of representation. Previous efforts focus on iterative improvement leveraging earlier layer activation based on only the current prediction [34], or feedback from the feedforward stage that immediately follows the stage where refinement is occurring [29]. In contrast, our proposed distributed feedback mechanism guides earlier layers by way of feedback which essentially preserves information about all the subsequent forward stages that is most relevant; the connectivity among gating modules in the reverse (top-down) direction allows for selective use of information from any subsequent layer in an adaptive manner that is determined during training. While this 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. As shown in Table 4.1, the segmentation performance increases by a significant margin, which implies that the DIGNet is successful in its objectives of bridging the aforementioned information gaps through efficient and effective feedback propagation.

Intuitively, the key idea of designing the feedback mechanism in a cascade manner (deeper $\rightarrow$ shallower) can be seen as a similar to a hypercolumn representation [19] 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

∗	Feedback Method	mIoU (%)
ResNet50 [21]	Baseline†	59.4
	Recurrent CNN† [49]	59.9
	Learning-with-Rethinking† [34]	65.0
	Recurrent Gating [29]	67.2
	DIGNet	**68**
ResNet101 [21]	Baseline†	65.3
	Learning-with-Rethinking† [34]	68.3
	Recurrent Gating [29]	68.9
	DIGNet	**72.5**

Table 4.1: Performance comparison of feedback based approaches with respect to DIGNet on the PASCAL VOC 2012 validation set. † refers to our implementation.

while bringing improvement subject to a hypercolumn style representation. However, in our case, the feedback signal is subject to block-wise 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.

4.2 DIGNet Data Flow and Iterative Inference

In this section, we discuss the data flow and iterative inference in the DIGNet. During the first iteration, the modulator gates $(\mathcal{G}_m^1, \mathcal{G}_m^2, \cdots, \mathcal{G}_m^n)$ act like a short circuit and simply allow a bypass of feedforward information similar to a reciprocal gate in a bottom-up manner. The feed-forward stages $(f_\vartheta^1, f_\vartheta^2, \cdots, f_\vartheta^n)$ process the input image to produce a reasonable feature representation. The feedback mechanism starts from

the last (deepest) layer of bottom-up feedforward network. For instance, in case of ResNet101-FCN the input to the feedback mechanism is the `res5c` output. In the next iteration, DIGNet executes two steps - (a) First, feedback signals are generated in a cascaded manner starting from the initial prediction towards the earlier stages. 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. (b) The modulator gates become activated and take signals from the propagator gates to modulate signals received as input from the preceding feed-forward blocks. This step can be seen as a traditional feed-forward network except that gating interacts with feedforward processing in effect producing adaptive features. All subsequent iterations in DIGnet proceed with executing these same steps, focusing on generating new feedback signals from the output and intermediate representations based on previous iterations, and finally modulating the intermediate bottom-up features for gating the next forward pass. Algorithm 1 describes the set of steps for iterative data flow and inference in DIGNet.

In the implementation of DIGNet, propagator gates have no influence in the first feed-forward iteration since the bottom-up features are not available until the first iteration is finished. At the T^{th} training stage, DIGNet iteratively passes the prediction of the $(T-1)^{th}$ stage in a reverse direction to guide the feed-forward operation at the T^{th} stage. With the increase of iterative steps, the propagator gates allows 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. To elicit a trade-off between performance and computational cost we set a value of $T = 2$ in our experiments.

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
    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
    $\mathcal{F}^n = f^n$
    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 for
  return f^n
end function

4.3 DIGNet Gate Modules

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

4.3.1 Propagator Gate

As shown in Fig. 4.1, each propagator gate takes the feedback signal and bottom-up features 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 convolution 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 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:

$$\mathcal{F}^i = \hat{y}(\mathbf{W_c} * (\underbrace{(\mathbf{W_a} * f^i)}_{\text{bottom-up feature}} \oplus \underbrace{(\mathbf{W_b} * \mathcal{F}^{i+1})}_{\text{feedback signal}})) \tag{4.2}$$

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 of obtaining a feedback signal is similar at each top-down stage.

4.3.2 Modulator Gate

The main task of the modulator gate is to assist in generating the input for next bottom-up (feedforward) 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. As mentioned prior (Sec. 4.2), in the very first iteration, no feedback signal is fed into the modulator gate. The modulator gate $\mathcal{G}_m$ simply acts as a bypass for feedforward feature activation. In subsequent iterations, $\mathcal{G}_m^i$ takes the feedback signal $\mathcal{F}^{(i)}$ and feed-forward feature

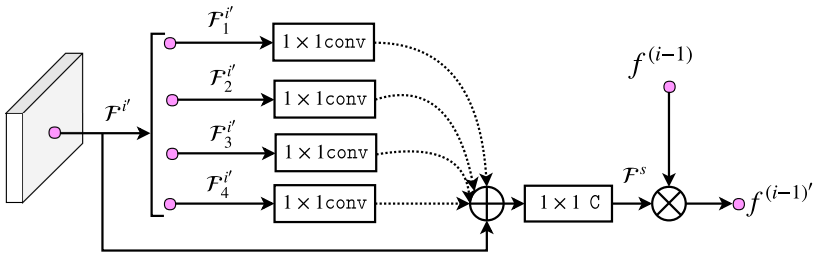

Figure 4.2: Illustration of the proposed modulator gate.

map f^{i-1} as inputs and processes this to generate the modulated signal $f^{(i-1)'}$. The feedback signal $\mathcal{F}^i$ is processed first to have the same channel dimension as $f^{(i-1)}$. Inspired by [20; 40; 66; 11], we find that applying a global contextual prior is beneficial in generating the modulating signal. We first create a spatial pyramid [66] 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. A detailed overview of the modulator gate is shown in Fig. 4.2.

Following other feedback based approaches [3; 34; 29; 65], we optimize DIGNet by unfolding it as a deep feed-forward convolutional neural network through back-propagation. 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.

4.4 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 [14]. Then, we evaluate DIGNet on three different semantic segmentation datasets, including PASCAL VOC 2012 [14], ADE20K [67], and COCO-Stuff [5]. Experimental results demonstrate the superiority of our proposed DIGNet architecture over baselines in a variety of respects.

4.4.1 Implementation Details

Our implementation is based on the open source platform PyTorch [47]. Inspired by previous work [10; 26; 10] 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. Recent works [10; 66] showed that overall performance can be improved by training batch normalization layers with a larger batch and crop size. Note that

whenever we report experimental results for DIGNet this denotes ResNet101-DIGNet.

4.4.2 Dataset and Evaluation Metrics

PASCAL VOC 2012: This 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 [10; 42; 26; 38; 10], we use the augmented training set that includes extra labeled PASCAL VOC images [18].

ADE20K MIT: This is a newer and more complex dataset for scene parsing that provides semantic labels for 150 classes including 115 things and 35 stuffs, with more than 20k indoor and outdoor images. Following [66; 38], we use the provided validation set of 2000 images for quantitative evaluation.

COCO-Stuff: COCO-Stuff is also a relatively recently released scene parsing dataset based on MS-COCO annotations. Following the split in [5], we use 9k images for training and another 1k for testing to evaluate DIGNet. This dataset provides segmentation labeling for the entire scene for MS-COCO images making it more complex. We use the segmentation labels which contain a total of 182 categories including 91 things and 91 stuff classes.

We use standard dense labelling evaluation measures named Mean Intersection over Union (Mean IOU), Mean Class Accuracy (MCA) and per-pixel accuracy (PPA) which are already defined in Section 3.6.2.

4.4.3 Gating Semantic Information with DIGNet

To investigate the role of distributed iterative gating in DIGNet, we conduct experiments under a few different settings. We mainly focus on a two major considerations to validate the design choices, including propagating more semantic information to earlier layers by applying gating modules, and adjusting the feature dimensionality that is used to propagate a feedback signal to earlier layers.

Semantic Information in Gating Low-level Feature: 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. 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 the DIGNet. Experimental results are shown in Table 4.2. 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 4.2 and other results.

Method	$\mathcal{F}^1$	$\mathcal{F}^2$	$\mathcal{F}^3$	$\mathcal{F}^4$	$\mathcal{F}^5$	$\mathcal{F}^6$	mIoU(%)
ResNet101-FCN(32s)							65.3
						✓	70.5
					✓	✓	71.1
				✓	✓	✓	71.8
			✓	✓	✓	✓	71.9
		✓	✓	✓	✓	✓	72.6
	✓	✓	✓	✓	✓	✓	**72.5**

Table 4.2: Performance of DIGNet with a varying extent of the reach of feedback gating for the PASCAL VOC 2012 val set.

DIGNet Compression with Explicit Channel Reduction: To demonstrate that the further improvement seen above is brought by a distributed gating strategy rather than propagating the high-level semantic features themselves, backward propagation of features is subject to a channel reduction strategy. The natural way to propagate features backward is to keep the original dimensionality but interestingly we find that scaling down the feature map by a large ratio in the feedback process does not degrade performance, and moreover, brings improvement in retaining only relevant feedback signals. Even by keeping a significantly smaller portion of features passed back, performance is still better than carrying a high dimensional feedback signal. Table 4.3 shows the result of DIGNet in terms of feature depth in propagating the feedback signal used for gating. The corresponding mIoU is marginally higher for the compression strategy - it also reduces network complexity in terms of parameters. Additionally, smaller feature map propagation in the feedback process can yield faster inference.

	Feature propagation choice	mIoU
DIGNet-Res101(32s)	{**class** → 256 → 384 → 448 → 480 → 512}	70.3
	{**class** → 256 → 256 → 128 → 64 → 32}	**72.5**

Table 4.3: Performance of DIGNet-ResNet101(32s) for two different feature propagation choices on PASCAL VOC 2012 validation set.

4.4.4 Results on PASCAL VOC 2012 dataset

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 4.4 shows the comparison results of different baselines and our proposed approach on the PASCAL VOC 2012 validation set. Interestingly, *ResNet101-*

Method	mIoU	Method	mIoU
ResNet50-32s† [21]	59.4	**ResNet50-32s-DIGNet**	**68**
ResNet101-32s† [21]	65.3	**ResNet101-32s-DIGNet**	**72.5**
ResNet101-8s† [21]	71.3	**ResNet101-8s-DIGNet**	**77.5**
DeepLabV2-Res101† [10]	74.9	**DeepLabV2-DIGNet**	**76.1**

Table 4.4: PASCAL VOC 2012 validation set results for baselines and DIGNet.

DIGNet with *OS=32* marginally outperforms the 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 any baseline with the addition of DIG. We further conduct experiments for the proposed DIGNet on the PASCAL VOC 2012 test set. Following

Method	mIoU (%)
Adelaide_Very_Deep_FCN_VOC [61]	79.1
LRR_4x_ResNet-CRF [16]	79.3
DeepLabv2-CRF [10]	79.7
CentraleSupelec Deep G-CRF [8]	80.2
HikSeg_COCO [56]	81.4
SegModel [52]	81.8
Deep Layer Cascade (LC) [35]	82.7
TuSimple [60]	83.1
Large_Kernel_Matters [48]	83.6
Multipath-RefineNet (Res152) [38]	83.4
ResNet-38_MS_COCO [62]	84.9
PSPNet [66]	85.4
DeepLabv3 [10]	85.7
DIGNet	80.7

Table 4.5: Quantitative results in terms of mean IoU on PASCAL VOC 2012 test set.

existing works [10; 66; 45; 38], 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 4.5, DIGNet achieves 80.7% mIoU which is competitive compared to other baselines especially for a simple mechanism attached to a standard ResNet architecture.

We provide a qualitative visual comparison of our approach with respect to the baselines in Fig. 4.3. 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.

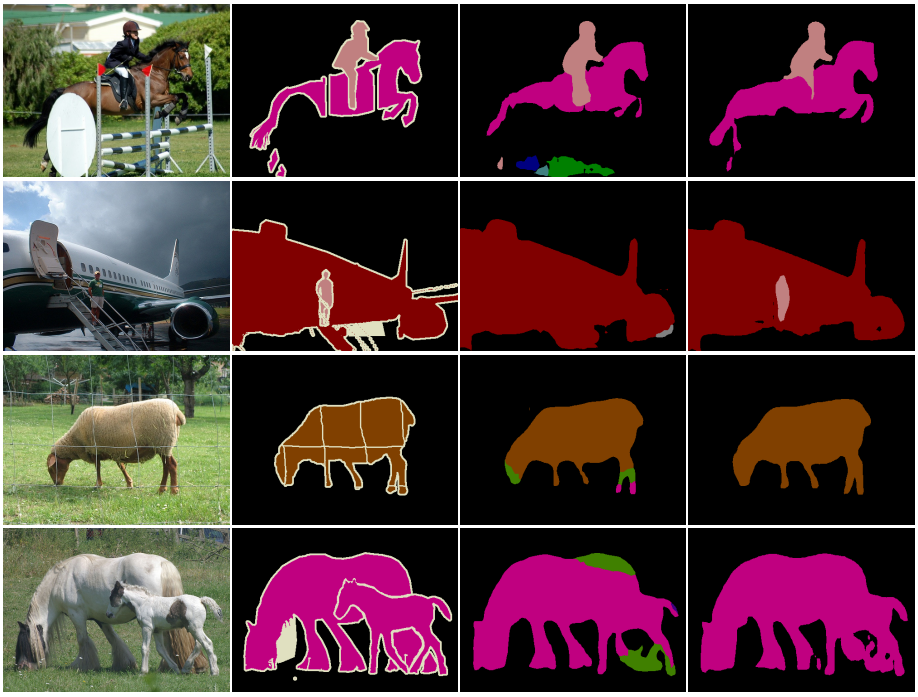

Figure 4.3: Qualitative results of DIGNet corresponding to the PASCAL VOC 2012 validation set.

4.4.5 Results on ADE20K

Table 4.6 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.

Fig. 4.4 depicts a visual comparison of DIGNet with respect to the baselines. ResNet101-DIGNet is capable of capturing object/stuff with high accuracy compared to the baseline.

Method	mIoU(%)	Pixel Acc.(%)	Overall(%)
FCN [42]	29.4	71.3	50.4
CascadeNet [67]	34.90	74.52	54.71
DilatedNet [63]	34.3	76.4	55.3
PSPNet [66]	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 4.6: Quantitative analysis of our approach based on different architectures *vs.* state-of-the-art methods based on the ADE20K validation set. † indicates our implementation.

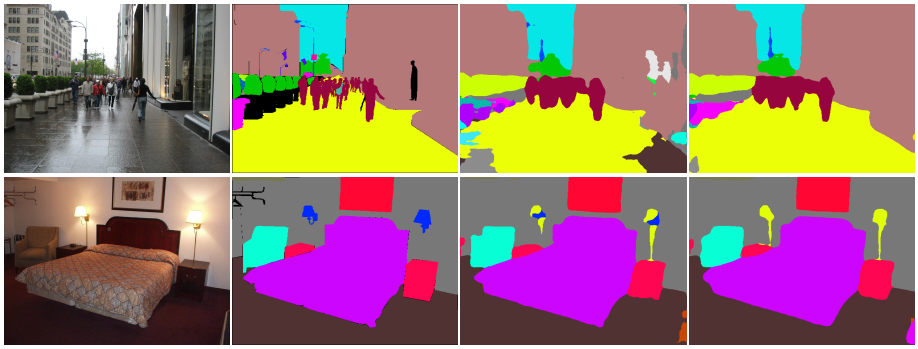

Image gt ResNet101 **DIGNet**

Figure 4.4: ADE20K results. Left to Right: Input image, ground-truth, ResNet101, and ResNet101-DIGNet.

4.4.6 Results on COCO-Stuff

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 COCO-Stuff dataset are reported in Table 4.7. 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 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(%)
FCN-8s [42]	60.4	38.5	27.2
OHE + DC + FCN [22]	66.6	45.8	34.3
DAG-RNN + CRF [54]	63.0	42.8	31.2
RefineNet-Res101 [38]	65.2	45.3	33.6
CCL [13]	66.3	**48.8**	35.7
ResNet101-32s [21]†	58.7	38.0	26.4
ResNet101-DIGNet	61.8	40.7	29.1
ResNet101-8s [21]†	64.6	44.9	33.4
ResNet101-DIGNet	**67.3**	47.4	**36.3**
DeepLabv2 (ResNet-101)† [10]	65.1	45.5	34.1
DeepLabv2-DIGNet	67.0	46.4	35.9

Table 4.7: Comparison of scene parsing results on the Coco-Stuff test set. † refers to our own implementation.

4.4.7 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. 4.5), highlighting the powerful influence of DIGNet to correct the poor initial prediction. Table 4.8 further reveals the stronger capacity of DIGNet on resolving

Figure 4.5: When the initial prediction has categorical ambiguity, DIGNet iteratively adjusts information passed forward in a bottom-up fashion through the feedback signal resulting in recognition of the correct class.

Method	Categorial ambiguity					
	cow	horse	bike	mbike	dog	cat
ResNet101(8s) [21]	73.6	70.8	39.6	77.5	77.5	85.0
ResNet101-DIGNet	**86.0**	**85.6**	**45.7**	**84.9**	**94.0**	**88.1**

Table 4.8: Influence of DIGNet on resolving categorical ambiguity for a few initially confused categories in PASCAL VOC 2012 validation set.

categorical ambiguity.

Partial Segmentation: When the initial prediction has coarse-grained or spatially limited mask (see Fig. 4.6), DIGNet improves partial segmentation to generate a detailed mask by incorporating distributed gating in the feedback mechanism, in some instances completing the object.

Figure 4.6: 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.

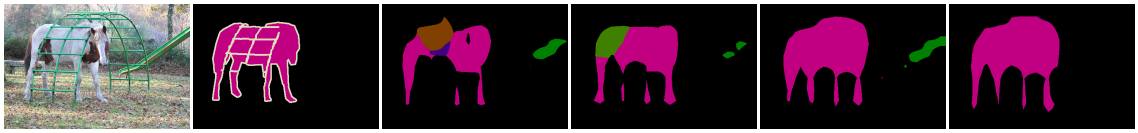

Figure 4.7: Visualization of label quality after top-down addition of distributed iterative gating modules. For each row, we show the input image, ground-truth, ResNet101(32s) prediction, and the predicted label map of DIGNet when distributed iterative gating modules are included in a top→down manner.

Fig. 4.7 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 (Fig.4.5), improve partial segmentation (Fig.4.6), and correct initial errors by way of coarse-to-fine refinement (Fig.4.7) provides a convincing case for the effectiveness of distributed iterative gating mechanism.

4.4.8 Analyze the Failure Cases of DIGNet

Despite the consistent performance improvement for the majority of cases, there are cases that are more challenging to predict. When DIGNet is allowed to itera-

tively propagate high-level semantics to earlier layers, it progressively improves the label map by way of top-down modulation (Fig. 4.7 and Table 4.2). In extreme cases, when the initial prediction of any foreground object share similar visual feature with surrounding background it may gradually move to partial incorrect labeling (see Fig. 4.8). 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.

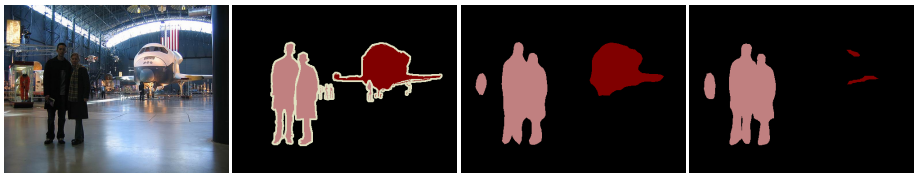

image ground-truth baseline DIGNet

Figure 4.8: An example of a rare failure case. It is interesting to note that the final labeling in each case tends to be globally consistent over objects.

Chapter 5

Conclusion and Future Work

In this thesis, we have presented recurrent routing mechanisms for deep convolutional neural networks for the purpose of dense image labeling tasks, and demonstrated the powerful capability of feedback in generating rich features. We have proposed two novel feedback routing approaches in the field of semantic segmentation and scene parsing. Both of the feedback routing mechanisms are quite general in that they can be fused with almost any feedforward network to boost the performance of that network. First, we proposed a quite simple yet powerful feedback routing mechanism termed RIGNet which is able to produce much richer feature representation compared to baseline networks. In particular, feedback based recurrent processing has been found to correct errors in the inferences along fine object boundaries and also recover missing parts of large objects. The most important contribution of RIGNet is to justify the effectiveness of feedback based processing and forming a theoretical basis of feedback based mechanisms. Second, based on the foundation of RIGNet, we have proposed a canonical feedback routing mechanism with a sym-

biotic combination of propagator and modulator gates where intermediate features are re-weighted with feedback signal virtually carrying information from all deeper stages. This novel feedback routing mechanism for *Distributed Iterative Gating* called DIGNet is shown to greatly increase the inference capabilities of a variety of archetypal networks across different baselines. Moreover, DIGNet allows simpler networks to outperform much deeper or more complex counterparts while presenting only a marginal degree of overhead in additional computation. 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. In addition, we show that the role of neuroscience inspired recurrent feedback gating as one powerful mechanism to address the major challenges in scene parsing using feedforward deep neural networks.

Many interesting research questions arise from the approach and results presented in this thesis. One especially fruitful avenue for further investigation is to extend the feedback based feature re-weighting and iterative inference for video scene parsing including allowing for a role of temporal coherence. The canonical feedback routing mechanism is demonstrated through the simplistic design of propagator and modulator gates. Adding sophistication to these gate modules with architectural improvement inside the gating units can be another interesting future research direction. In addition, the findings of feedback routing and gating mechanisms can be further extended to deep neural networks for other scene understanding tasks. In short, the

capability of feedback for correcting errors made in a single feedforward pass through an appropriate recurrent gating mechanism justifies feedback gating as an exciting and important research direction for various scene understanding tasks, and many possible forms for it's structure remain open for further investigation.

Bibliography

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for scene segmentation. *TPAMI*, 2017. 1, 8, 15

[2] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. In *FG*, 2017. 10

[3] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. In *FG*, 2017. 37, 43

[4] H. Caesar, J. Uijlings, and V. Ferrari. Coco-stuff: Thing and stuff classes in context. *arXiv:1612.03716*, 2016. 23, 24

[5] H. Caesar, J. Uijlings, and V. Ferrari. Coco-stuff: Thing and stuff classes in context. In *CVPR*, 2018. 44, 45

[6] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *CVPR*, 2016. 2, 10

[7] A. Casanova, G. Cucurull, M. Drozdzal, A. Romero, and Y. Bengio. On the iterative refinement of densely connected representation levels for semantic segmentation. *arXiv:1804.11332*, 2018. 2, 15

[8] S. Chandra and I. Kokkinos. Fast, exact and multi-scale inference for semantic image segmentation with deep gaussian crfs. In *ECCV*, 2016. 49

[9] 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, 8

[10] 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, 8, 9, 15, 24, 29, 34, 44, 45, 48, 49, 52

[11] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 43

[12] 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

[13] 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. 52

[14] 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. 23, 44

[15] A. Gazzaley and A. C. Nobre. Top-down modulation: bridging selective attention and working memory. *Trends in cognitive sciences*, 2012. 2

[16] G. Ghiasi and C. C. Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *ECCV*, 2016. 1, 8, 49

[17] C. D. Gilbert and W. Li. Top-down influences on visual processing. *Nature Reviews Neuroscience*, 2013. 9

[18] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *ICCV*, 2011. 24, 45

[19] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015. 38

[20] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *TPAMI*, 2015. 43

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 8, 34, 39, 48, 52, 53

[22] H. Hu, Z. Deng, G.-T. Zhou, F. Sha, and G. Mori. Labelbank: Revisiting global perspectives for semantic segmentation. *arXiv:1703.09891*, 2017. 52

[23] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 15

[24] 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. 35

[25] M. A. Islam, S. Naha, M. Rochan, N. Bruce, and Y. Wang. Label refinement network for coarse-to-fine semantic segmentation. *arXiv:1703.00551*, 2017. 2, 15

[26] 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, 8, 9, 10, 29, 44, 45

[27] M. A. Islam, M. Rochan, S. Naha, N. Bruce, and Y. Wang. Gated feedback refinement network for coarse-to-fine dense semantic image labeling. *arXiv:1806.11266*, 2018. 2

[28] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 1175–1183. IEEE, 2017. 8

[29] R. Karim, M. A. Islam, and N. D. B. Bruce. Recurrent iterative gating networks for semantic segmentation. In *WACV*, 2019. 9, 35, 37, 38, 39, 43

[30] J. U. Kim, H. G. Kim, and Y. M. Ro. Iterative deep convolutional encoder-decoder network for medical image segmentation. *arXiv:1708.03431*, 2017. 9

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 8

[32] V. A. Lamme and P. R. Roelfsema. The distinct modes of vision offered by feedforward and recurrent processing. *Trends in neurosciences*, 2000. 2, 9

[33] K. Li, B. Hariharan, and J. Malik. Iterative instance segmentation. In *CVPR*, 2016. 1, 9, 35

[34] X. Li, Z. Jie, J. Feng, C. Liu, and S. Yan. Learning with rethinking: Recurrently improving convolutional neural networks through feedback. *Pattern Recognition*, 2018. ix, 9, 10, 29, 35, 37, 38, 39, 43

[35] 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. 49

[36] M. Liang and X. Hu. Recurrent convolutional neural network for object recognition. In *CVPR*, 2015. 15

[37] M. Liang, X. Hu, and B. Zhang. Convolutional neural networks with intra-layer recurrent connections for scene labeling. In *NIPS*, 2015. 1, 9, 15

[38] G. Lin, A. Milan, C. Shen, and I. Reid. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017. 1, 2, 8, 10, 15, 45, 49, 52

[39] 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

[40] W. Liu, A. Rabinovich, and A. C. Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015. 43

[41] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1, 8

[42] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2, 29, 45, 51, 52

[43] L. McIntosh, N. Maheswaranathan, D. Sussillo, and J. Shlens. Recurrent segmentation for variable computational budgets. *arXiv:1711.10151*, 2017. 9, 10

[44] A. Nayebi, D. Bear, J. Kubilius, K. Kar, S. Ganguli, D. Sussillo, J. J. DiCarlo, and D. L. Yamins. Task-driven convolutional recurrent models of the visual system. *arXiv:1807.00053*, 2018. 21

[45] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 1, 8, 10, 15, 49

[46] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015. 8

[47] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 23, 44

[48] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *CVPR*, 2017. 49

[49] P. H. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *ICML*, 2014. ix, 2, 9, 15, 28, 29, 37, 39

[50] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *ECCV*, 2016. 10, 15

[51] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(4):640–651, 2017. 8, 9

[52] F. Shen, R. Gan, S. Yan, and G. Zeng. Semantic segmentation via structured patch prediction, context crf and guidance crf. In *CVPR*, 2017. 49

[53] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta. Beyond skip connections: Top-down modulation for object detection. *arXiv:1612.06851*, 2016. 10

[54] B. Shuai, Z. Zuo, B. Wang, and G. Wang. Scene segmentation with dag-recurrent neural networks. *TPAMI*, 2017. 52

[55] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 1, 8

[56] H. Sun, D. Xie, and S. Pu. Mixed context networks for semantic segmentation. *arXiv:1610.05854*, 2016. 49

[57] 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*, 2015. 1, 8

[58] A. Veit and S. Belongie. Convolutional networks with adaptive computation graphs. *arXiv:1711.11503*, 2017. 9

[59] R. C. Vijay Badrinarayanan, Alex Kendall. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:2481–2495, 201. 8

[60] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell. Understanding convolution for semantic segmentation. In *WACV*, 2018. 49

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

[62] Z. Wu, C. Shen, and A. Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019. 49

[63] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 1, 51

[64] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 15

[65] 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, 9, 10, 20, 35, 43

[66] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017. 1, 9, 43, 44, 45, 49, 51

[67] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 44, 51