LANGUAGE CONTROLS MORE THAN TOP-DOWN ATTENTION: MODULATING BOTTOM-UP VISUAL PROCESSING WITH REFERRING EXPRESSIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

How to best integrate linguistic and perceptual processing in multimodal tasks is an important open problem. In this work we argue that the common technique of using language to direct visual attention over high-level visual features may not be optimal. Using language throughout the bottom-up visual pathway, going from pixels to high-level features, may be necessary. Our experiments on several English referring expression datasets show significant improvements when language is used to control the filters for bottom-up visual processing in addition to top-down attention.

1 INTRODUCTION

As human beings, we can easily understand the surrounding environment with our visual system and interact with each other using language. Since the work of Winograd (1972), developing a system that understands human language in a situated environment is one of the long-standing goals of artificial intelligence. Recent successes of deep learning studies in both language and vision domains have increased the interest in tasks that combine language and vision (Antol et al., 2015; Xu et al., 2015; Krishna et al., 2016; Suhr et al., 2017; Anderson et al., 2018b; Hudson & Manning, 2019). However, how to best integrate linguistic and perceptual processing is still an important open problem. In this work we investigate whether language should be used to control the filters for bottom-up visual processing as well as top-down attention.

In the human visual system, attention is driven by both *"top-down"* cognitive processes (*e.g.* focusing on target's color or location) and *"bottom-up"* salient, behaviourally relevant stimuli (*e.g.* fast moving objects) (Corbetta & Shulman, 2002; Connor et al., 2004; Theeuwes, 2010). Studies on embodied language explore the link between linguistic and perceptual representations (Pulvermüller, 1999; Vigliocco et al., 2004; Gallese & Lakoff, 2005) and it is often assumed that language has a *high-level* effect on perception and drives the *"top-down"* visual attention (Bloom, 2002; Jackendoff & Jackendoff, 2002; Dessalegn & Landau, 2008). However, recent studies from cognitive science point out that language comprehension also affects low-level visual processing (Meteyard et al., 2007; Boutonnet & Lupyan, 2015). Motivated by this, we propose a model[1] that can modulate either or both of *"bottom-up"* and *"top-down"* visual processing with language conditional filters.

Current deep learning systems for language-vision tasks typically start with low-level image processing that is not conditioned on language, then connect the language representation with high level visual features to control the visual focus. To integrate both modalities, concatenation (Malinowski et al., 2015), element-wise multiplication (Malinowski et al., 2015; Lu et al., 2016; Kim et al., 2016) or attention from language to vision (Xu et al., 2015; Xu & Saenko, 2016; Yang et al., 2016; Lu et al., 2017; Anderson et al., 2018a; Zellers et al., 2019) may be used. Specifically they do not condition low-level visual features on language. One exception is De Vries et al. (2017) which proposes conditioning the ResNet (He et al., 2016) image processing network with language conditioned batch normalization parameters at every stage. Our model differs from these architectures by having explicit *"bottom-up"* and *"top-down"* branches and allowing us to experiment with modulating one or both branches with language generated kernels.

[1]We will release our code and pre-trained models along with a reproducible environment after the blind review process.

We evaluate our proposed model on the task of *image segmentation from referring expressions* where given an image and a natural language description, the model returns a segmentation mask that marks the object(s) described. We can contrast this with purely image based object detection (Girshick, 2015; Ren et al., 2017) and semantic segmentation (Long et al., 2015; Ronneberger et al., 2015; Chen et al., 2017) tasks which are limited to predefined semantic classes. Our task gives users more flexibility to interact with the system by allowing them to describe objects of interest in free form language. The language input may contain various visual attributes (e.g., color, shape), spatial information (e.g., "on the right", "in front of"), actions (e.g., "running", "sitting") and interactions/relations between different objects (e.g., "arm of the chair that the cat is sitting in"). This makes the task both more challenging and suitable for comparing different strategies of language control.

The perceptual module of our model is based on the U-Net image segmentation architecture (Ronneberger et al., 2015). This architecture has clearly separated bottom-up and top-down branches which allows us to easily vary what parts are conditioned on language. The bottom-up branch starts from low level visual features and applies a sequence of contracting filters that result in successively higher level feature maps with lower spatial resolution. Following this is a top-down branch which takes the final low resolution feature map and applies a sequence of expanding filters that eventually result in a segmentation mask at the original image resolution. Information flows between branches through skip connections between contracting and expanding filters at the same level. We experiment with conditioning one or both of these branches with language.

To make visual processing conditional on language, we add language-conditional filters at each level of the architecture, similar to Misra et al. (2018). Our baseline only applies language-conditional filters on the top-down branch. Modulating only the top-down/expanding branch with language means the high level features extracted by the bottom-up/contracting branch cannot be language-conditional. Our model expands on this baseline by modulating both branches with language-conditional filters. Empirically, we find that adding language modulation to the bottom-up/contracting branch has a significant positive improvement on the baseline model. Our proposed model achieves state-of-the art performance on three different English referring expression datasets.

2 Related Work

In this section, we review related work in several related areas: Semantic segmentation classifies the object category of each pixel in an image without language input. Referring expression comprehension locates a bounding box for the object(s) described in the language input. Image segmentation from referring expressions generates a segmentation mask for the object(s) described in the language input. We also cover work on language-conditional (dynamic) filters and studies that use them to modulate deep-learning models with language.

2.1 Semantic Segmentation

Primitive semantic segmentation models are based on Fully Convolutional Networks (FCN) (Long et al., 2015). DeepLab (Chen et al., 2017) and U-Net (Ronneberger et al., 2015) are the most notable state-of-the-art semantic segmentation models related to our work. DeepLab replaces regular convolutions with atrous (dilated) convolutions in the last residual block of ResNets (He et al., 2016) and implements Atrous Spatial Pyramid Pooling (ASPP) which fuses multi-scale visual information. The U-Net architecture (Ronneberger et al., 2015) improves over the standard FCN by connecting contracting (bottom-up) and expanding (top-down) paths at the same resolution: the output of the encoder layer at each level is passed to the decoder at the same level.

2.2 Referring Expression Comprehension

Early models for this task were typically built using a hybrid LSTM-CNN architecture (Hu et al., 2016b; Mao et al., 2016). Newer models (Hu et al., 2017; Yu et al., 2016; 2018; Wang et al., 2019) use an Region-based CNN (R-CNN) variant (Girshick et al., 2014; Ren et al., 2017; He et al., 2017) as a sub-component to generate object proposals. Nagaraja et al. (2016) proposes a solution based on multiple instance learning. Cirik et al. (2018) implements a model based on Neural Module Networks (NMN) by using syntax information. Among the literature, Compositional Modular Network

(CMN) (Hu et al., 2017), Modular Attention Network (MAttNet) (Yu et al., 2018) and Neural Module Tree Networks (NMTree) (Liu et al., 2019) are the most notable state-of-the-art methods, and all of them are based on NMN (Andreas et al., 2016).

2.3 IMAGE SEGMENTATION FROM REFERRING EXPRESSIONS

Notable models for this task include Recurrent Multimodal Interaction (RMI) model (Liu et al., 2017), Recurrent Refinement Networks (RRN) (Li et al., 2018), Dynamic Multimodal Network (DMN) (Margffoy-Tuay et al., 2018), Convolutional RNN with See-through-Text Embedding Pixel-wise heatmaps (Step-ConvRNN or ConvRNN-STEM) (Chen et al., 2019a), Caption-aware Consistent Segmentation Model (CAC) (Chen et al., 2019b), Bi-directional Relationship Inferring Network (BRINet) Hu et al. (2020) and Linguistic Structure guided Context Modelling (LSCM) module Hui et al. (2020). RRN which has a structure similar to U-Net, is built on top of a Convolutional LSTM (ConvLSTM) (SHI et al., 2015) network. Unlike our model, ConvLSTM filters are not generated from language representation and the multi-modal representation is used only in the initial time step. DMN generates 1 x 1 language-conditional filters for language representation of each word. It performs convolution operation on visual representation with language-conditional filters to generate multi-modal representation for each word. Like RMI, word-level multi-modal representations are fed as input to a multi-modal RRN to obtain multi-modal representation for image/language pairs. Step-ConvRNN starts with a visual-textual co-embedding and uses a ConvRNN to iteratively refine a heatmap for image segmentation. Step-ConvRNN uses a bottom-up and top-down approach similar to this work, however, our model uses spatial language generated kernels within a simpler architecture. CAC also generates 1 x 1 language-conditional dynamic filters. Unlike our model, CAC applies these dynamic filters to single resolution / single feature map and additionally generates location-specific dynamic filters (e.g. left, bottom) to capture relations between the objects exist at the different parts of the image. BRINet implements two different attention mechanisms: language-guided visual attention and vision-guided linguistic attention. LSCM implements a dependency parsing guided bottom-up attention mechanism to predict masks.

2.4 LANGUAGE-CONDITIONAL FILTERS

To control a deep learning model with language, early work such as Modulated ResNet (MODERN) (De Vries et al., 2017) and Feature-wise Linear Modulation (FiLM) (Perez et al., 2018) used conditional batch normalization layers with only language-conditioned coefficients rather than customized filters. Finn et al. (2016) generates action-conditioned dynamic filters. Li et al. (2017) is the first work which generates dynamic language-conditional filters. Gao et al. (2018) proposes a VQA solution method which has a group convolutional layer whose filters are generated from the question input. Gavrilyuk et al. (2018) introduces a new task called as actor and action segmentation and to solve this task, proposes an architecture which uses dynamic filters for multiple resolutions. Similar to our work, Misra et al. (2018) adds language conditional filters to a U-Net based architecture for the task of mapping instructions to actions in virtual environments. **?** also uses an architecture based on U-Net and Misra et al. (2018) to solve a navigation and spatial reasoning problem. Those models only modulate top-down visual processing with language.

Referring expression models that incorporate language-conditional filters into the architecture include (Chen et al., 2019b; Margffoy-Tuay et al., 2018). Margffoy-Tuay et al. (2018) generates language-conditional filters for words individually rather than whole sentence. Chen et al. (2019b) generates 1 x 1 language-conditional filters from expressions. To make 1 x 1 language-conditional filters spatially aware, different filters are generated for different image regions (e.g. top, left, right, bottom).

Our main contribution in this work is an explicit evaluation of language conditional filters for bottom-up visual processing in comparison to only using language for top-down attention control.

3 MODEL

Figure 1 shows an overview of our proposed architecture. For a given referring expression S and an input image I, the task is predicting a segmentation mask M that covers the object(s) referred to. First, the model extracts a $64 \times 64 \times 1024$ tensor of low-level features using a backbone convolutional

neural network and encodes the referring expression S to a vector representation r using a long short-term memory (LSTM) network (Hochreiter & Schmidhuber, 1997). Starting with the visual feature tensor, the model generates feature maps in a contracting and an expanding path where the final map represents the segmentation mask, similar to U-Net (Ronneberger et al., 2015). 3x3 convolutional filters generated from the language representation r (language kernels) are used to modulate both the contracting and the expanding paths. Our experiments show that modulating both paths improves the performance dramatically.

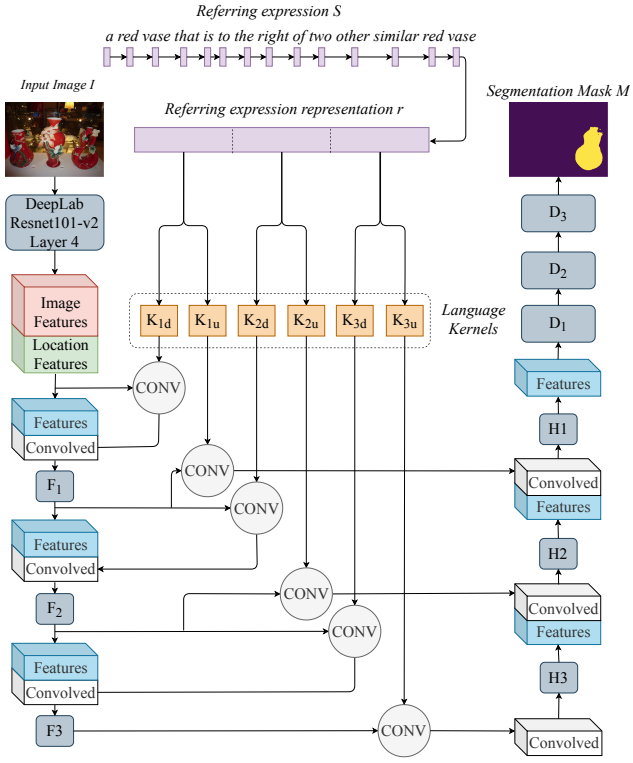

Figure 1: Overview of our model.

3.1 Low-level Image Features

Given an input image I, we extract visual features from the fourth layer of the DeepLab ResNet101-v2 network (Chen et al., 2017) pre-trained on the Pascal VOC dataset (Everingham et al., 2010). We set $W = H = 512$ as the image size for our experiments. Thus, the output of the fourth convolutional layer of DeepLab ResNet101-v2 produces a feature map with the size of $(64, 64)$ and 1024 channels for this setup. We concatenate 8-D location features to this feature map following previous work (Hu et al., 2016b; Liu et al., 2017; Ye et al., 2019; Chen et al., 2019a). The final representation, I_0, has 1032 channels, and the spatial dimensions are $(64, 64)$.

3.2 Language Representation

Consider a referring expression $S = [w_1, w_2, ..., w_n]$ where w_i represents the i'th word. In this work, each word w_i is represented with a 300-dimensional GloVe embedding (Pennington et al., 2014), i.e. $w_i \in \mathbb{R}^{300}$. We map the referring expression S to hidden states using a long short-term memory network (Hochreiter & Schmidhuber, 1997) as $h_i = LSTM(h_{i-1}, w_i)$. We use the final hidden state of the LSTM as the textual representation, $r = h_n$. We set the size of hidden states to 256, i.e. $h_i \in \mathbb{R}^{256}$.

3.3 SEGMENTATION MODEL

After generating image (I_0) and language (r) representations, our model generates a segmentation Mask M. We take the U-Net (Ronneberger et al., 2015) image segmentation model as the visual processing backbone. Our model extends the U-Net by conditioning both contracting and expanding branches on language using spatial language kernels.

Our model applies m convolutional modules to the image representation I_0. Each module, F_i, takes the concatenation of the previously generated feature map ($Down_{i-1}$) and its convolved version with a 3×3 language kernel K_{id} and produces an output feature map ($Down_i$). Each F_i has a 2D convolution layer followed by batch normalization (Ioffe & Szegedy, 2015) and ReLU activation function (Maas et al., 2013). The convolution layers have 5×5 filters with $stride = 2$ and $padding = 2$ halving the spatial resolution, and they all have the same number of output channels.

Following Misra et al. (2018), we split the textual representation r to m equal parts (t_i) to generate language-conditional filters (language kernels). We use each t_i to generate a language-conditional kernel (K_{id}):

$$K_{id} = \text{AFFINE}_i(\text{DROPOUT}(t_i)) \tag{1}$$

Each AFFINE_i is an affine transformation followed by normalizing and reshaping to convert the output to a convolutional filter. DROPOUT is the dropout regularization (Srivastava et al., 2014). After obtaining the kernel, we convolve it over the feature map obtained from the previous module to relate expressions to image features:

$$G_{id} = \text{CONVOLVE}(K_{id}, Down_{i-1}) \tag{2}$$

Then, the concatenation of the resulting text-modulated features (G_{id}) and the previously generated features ($Down_{i-1}$) is fed into module F_i for the next step.

In the expanding branch, we generate m feature maps starting from the final output of the contracting branch as follows:

$$G_{ju} = \text{CONVOLVE}(K_{ju}, I_j) \tag{3}$$

$$Up_m = H_m(G_{mu}) \tag{4}$$

$$Up_j = H_j(G_{mu} \oplus Up_{j-1}) \tag{5}$$

Similar to the bottom-up phase, G_{ju} is the modulated feature map with language-conditional kernels generated as follows:

$$K_{ju} = \text{AFFINE}_j(\text{DROPOUT}(t_j)) \tag{6}$$

where AFFINE_j is again an affine transformation followed by normalizing and reshaping. Here, we convolve the kernel (K_{ju}) over the feature maps from the contracting branch ($Down_j$). Each upsampling module H_m gets the concatenation ($\oplus$) of the text related features and the feature map (Up_j) generated from the previous module. Only the first module operates on just convolved features. Each H_j consists of a 2D deconvolution layer followed by a batch normalization and ReLU activation function. The deconvolution layers have 5×5 filters with $stride = 2$ and $padding = 2$ doubling the spatial resolution, and they all have the same number of output channels.

After generating the final feature map Up_1, we apply a stack of layers (D_1, D_2, ..., D_m) to map Up_1 to the exact image size. Similar to upsampling modules, each D_k is a 2D deconvolution layer followed by batch normalization and the ReLU activation. The deconvolutional layer has 5×5 filters with $stride = 2$ and $padding = 2$ to double the spatial sizes of the input. Each D_k preserves the number of channels except for the last one which maps the features to a single channel for the mask prediction. There is no batch norm operation and the ReLU activation for the final module, instead we apply a sigmoid function to turn the final features into probabilities ($P \in \mathbb{R}^{H \times W}$).

3.4 LEARNING

Given the probabilities ($P \in \mathbb{R}^{H \times W}$) for each pixel belonging to the target object(s), and the ground-truth mask $G \in \mathbb{R}^{H \times W}$, the main training objective is the pixel-wise Binary-Cross-Entropy (BCE) loss:

Table 1: Ablation results and comparison with the previous works on the val set of UNC dataset with *prec@X* and *IoU* metrics.

Method	*prec@0.5*	*prec@0.6*	*prec@0.7*	*prec@0.8*	*prec@0.9*	*IoU*
Top-Down Modulation w/ FiLM layers	60.97	53.19	43.71	31.62	10.57	54.21
Top-Down Modulation w/ 1x1 filters	63.00	54.20	44.71	32.01	10.74	55.04
Top-Down Modulation w/ 3x3 filters	64.29	55.26	46.29	32.96	11.78	56.13
Bottom-Up Modulation (disconnected) w/ 3x3 filters	69.92	62.84	52.16	32.70	7.18	58.77
Bottom-Up Modulation w/ 3x3 filters	72.13	65.92	57.93	43.80	17.49	60.73
Dual Modulation w/ 1x1 filters	71.76	65.77	58.19	44.80	17.05	60.75
Dual Modulation w/ 3x3 filters (full model)	**73.53**	**67.53**	**60.00**	**46.96**	**18.80**	**61.95**
LSCM (Hui et al., 2020)	70.84	63.82	53.67	38.69	12.06	61.54
BRINet (Hu et al., 2020)	71.83	65.05	55.64	39.36	11.21	61.35
Step-ConvRNN (Chen et al., 2019a)	70.15	63.37	53.15	36.53	10.45	59.13
CMSA (Ye et al., 2019)	66.44	59.70	50.77	35.52	10.96	58.32
RRN (Li et al., 2018)	61.66	52.5	42.4	28.13	8.51	55.33
DMN (Margffoy-Tuay et al., 2018)	65.83	57.82	46.80	27.64	5.12	54.83
RMI (Liu et al., 2017)	42.99	33.24	22.75	12.11	2.23	45.18

$$J = \frac{1}{HW} \sum_{i}^{H} \sum_{j}^{W} G_{ij} log(P_{ij}) + (1 - G_{ij}) log(1 - P_{ij}) \tag{7}$$

4 EXPERIMENTS

In this section we first give the details of the datasets and our experimental configurations (Section 4.1). A detailed analysis of the contribution of our idea and the different parts of the architecture is given in Section 4.2. Then we present our main results and compare our model with the state-of-the-art (Section 4.3). Finally, Section 4.4 shows some qualitative results.

4.1 DATASETS AND EXPERIMENT SETUP

Datasets: We evaluate our model on and ReferIt (130.5k expressions, 19.9k images), UNC (142k expressions, 20k images), UNC+ (141.5k expressions, 20k images) (Yu et al., 2016) and Google-Ref (G-Ref) (104.5k expressions, 26.7k images) (Mao et al., 2016) (Kazemzadeh et al., 2014) datasets. Unlike UNC, location-specific expressions are excluded in UNC+ through enforcing annotators to describe objects by their appearance. ReferIt, UNC, UNC+ datasets are collected through a two-player game (Kazemzadeh et al., 2014) and have short expressions (avg. 4 words). G-Ref have longer and richer expressions, since its expressions are collected from Amazon Mechanical Turk instead of a two-player game. ReferIt images are collected from IAPR Tc-12 dataset (Escalante et al., 2010) and the others use images present in MS COCO dataset (Lin et al., 2014).

Evaluation Metrics: Following the previous work (Liu et al., 2017; Margffoy-Tuay et al., 2018; Ye et al., 2019; Chen et al., 2019a), we use overall intersection-over-union (*IoU*) and $precision@X$ as evaluation metrics. Given the predicted segmentation mask and the ground truth, the *IoU* metric is the ratio between the intersection and the union of the two. The overall *IoU* calculates the total intersection over total union score. The second metric, $precision@X$, calculates the percentage of test examples that have *IoU* score higher than the threshold X. In experiments, $X \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$.

Implementation Details: As (Liu et al., 2017; Margffoy-Tuay et al., 2018; Ye et al., 2019; Chen et al., 2019a), we limit the maximum length of expressions to 20. In all convolutional layers, we set the filter size, stride, and number of filters (ch) as $(5, 5)$, 2, and 96, respectively. The depth is 4 in the U-Net part of the network. We set the dropout probability to 0.2 throughout the network. We use Adam optimizer (Kingma & Ba, 2014) with default parameters. We freeze the DeepLab ResNet101-v2 weights. There are 60 examples in each minibatch. We train our model for 15 epochs on a Tesla V100 GPU and each epoch takes at most two hours depending on the dataset.

Table 2: Comparison with the previous works on four datasets. Evaluation metric is the overall *IoU* and higher is better. Bold scores indicate the state-of-the-art performances. "-" indicates that the model has not been evaluated on the dataset. "n/a" indicates that splits are not same.

	UNC			UNC+			G-Ref	ReferIt
Method	val	testA	testB	val	testA	testB	val	test
CNN+LSTM (Hu et al., 2016a)	-	-	-	-	-	-	28.14	48.03
RMI (Liu et al., 2017)	45.18	45.69	45.57	29.86	30.48	29.5	34.52	58.73
DMN (Margffoy-Tuay et al., 2018)	49.78	54.83	45.13	38.88	44.22	32.29	36.76	52.81
DynamicFilters (Li et al., 2017)	-	-	-	-	-	-	-	54.30
KWA (Shi et al., 2018)	-	-	-	-	-	-	36.92	59.09
RRN (Li et al., 2018)	55.33	57.26	53.93	39.75	42.15	36.11	36.45	63.63
CMSA (Ye et al., 2019)	58.32	60.61	55.09	43.76	47.6	37.89	39.98	63.80
CAC (Chen et al., 2019b)	58.90	61.77	53.81	-	-	-	44.32	-
Step-ConvRNN (Chen et al., 2019a)	60.04	63.46	57.97	48.19	52.33	40.41	46.4	64.13
BRINet (Hu et al., 2020)	61.35	63.37	**59.57**	48.57	52.87	42.13	48.04	63.46
LSCM (Hui et al., 2020)	61.47	**64.99**	59.55	49.34	53.12	**43.50**	48.05	**66.57**
MAttNet (Yu et al., 2018)	56.51	62.37	51.70	46.67	52.39	40.08	n/a	-
NMTree (Liu et al., 2019)	56.59	63.02	52.06	47.40	53.01	41.56	n/a	-
Our Model	**61.95**	63.85	58.14	**50.42**	**54.16**	42.15	**49.76**	64.63

4.2 Ablation Results

We performed ablation studies to better understand the contributions of the different parts of our model. Table 1 shows the performances of the different architectures on the UNC validation split with *prec@X* and overall *IoU* metrics. Unless otherwise specified, 3×3 language-conditional filters are used in our models.

Modulating both top-down and bottom-up visual processing: We implemented three models, Top-down Modulation, Bottom-Up Modulation and Dual Modulation, to show the effect of modulating language in expanding and contracting visual branches. Since language information leaks through cross-connections between visual branches, we also experimented with a bottom-up modulation model which has no connection between visual branches. Bottom-up Modulation outperforms Top-down Modulation with $\approx$4.6 IoU improvement. Modulating language in both visual branches yields the best results by improving Bottom-up Modulation model with $\approx$1.2 IoU score.

Language-conditional Spatial Filters: When we compare the performances of Top-Down Modulation w/ 1×1 filters and Top-Down Modulation models, we see that the usage of language-conditional spatial filters brings additional improvement over the base model. Similarly, if we use 1×1 filters in our full model, the performance of the model decreases significantly. We performed the same experiment on G-Ref dataset and observed $\approx$1.3 IoU difference again.

FiLM layers vs. Language-conditional Filters: Another method for modulating language is using conditional batch normalization De Vries et al. (2017) or its successor, FiLM layers. Thus, we also replaced language-conditional filters with FiLM layers in Top-Down Modulation w/ 1x1 model and observed $\approx$0.8 IoU improvement. Morever, since we can take advantage of language-conditional spatial filters, Top-Down Modulation w/ 3x3 model baseline outperforms its FiLM variation with $\approx$1.9 IoU improvement.

4.3 Quantitative Results

Table 2 shows the comparison of our model with the previous work. Our model outperforms all previous models on all datasets. When we compare our model with the previous state-of-the-art model, Step-ConvRNN, the most significant improvement is on the G-Ref dataset.

We also compare our model with MAttNet and NMTree which are referring expression comprehension models. Since they present segmentation results after they predict bounding boxes for objects, they are comparable with our work. Our model is significantly better than MAttNet, NMTree which depends on an explicit object proposal network that is trained on more COCO images. This result shows the ability of our model to detect object regions and relate them with expressions.

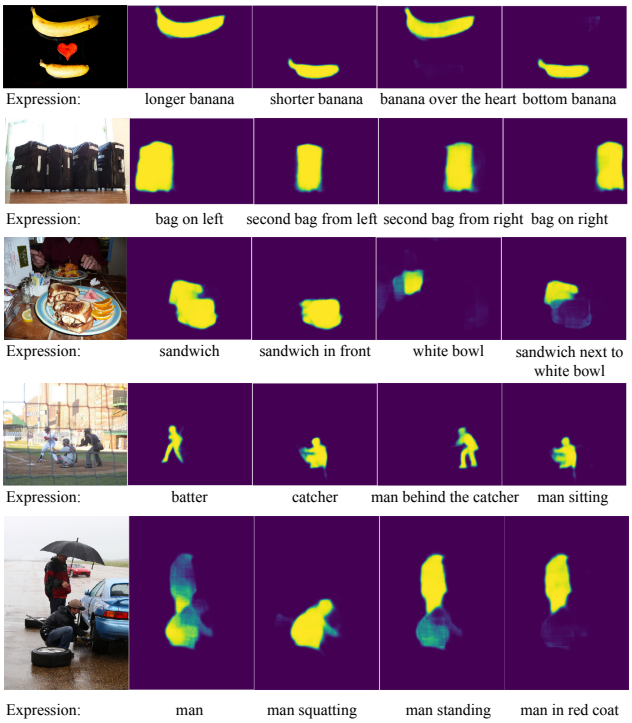

Figure 2: Some correct predictions of our model on UNC validation set. First column shows the input images and others show the predictions for the given referring expressions.

Table 1 presents the comparison of our model with the state-of-the-art in terms of $prec@X$ scores. The difference between our model and the state-of-the-art increases when the threshold increases. This indicates that our model is better at both finding and segmenting the referred objects.

4.4 Qualitative Results

In this section, we visualize some of the segmentation predictions of our model to gain better insights about the trained model.

Figure 2 shows some of the cases that our model segments correctly. These examples demonstrate that our model can learn a variety of language and visual reasoning patterns. For example, the first two examples of the first row show that our model learns to relate superlative adjectives (e.g., *longer*, *shorter*) with visual comparison. Examples include spatial prepositions (e.g., *on right, on left, next to, behind, over, bottom*) demonstrate the spatial reasoning ability of the model. We also see that the model can learn a domain-specific nomenclature (*catcher, batter*) that is present in the dataset. Lastly, we can see that the model can distinguish the different actions (e.g., *standing, squatting, sitting*).

Figure 3 shows some of the incorrect segmentation predictions from our model on the UNC validation dataset. In the figure, each group shows one of the observed patterns within the examples. One of them (a) is that our model tends to combine similar objects or their parts when they are hard to distinguish. Another reason for the errors is that some of the expressions are ambiguous (b), where there are multiple objects that could be the correspondence of the expression. And the model segments both possible objects. Some of the examples (d) are hard to segment completely due to the lack of light or objects that mask the referred objects. Finally, some of the annotations contain incorrect or incomplete ground-truth mask (c).

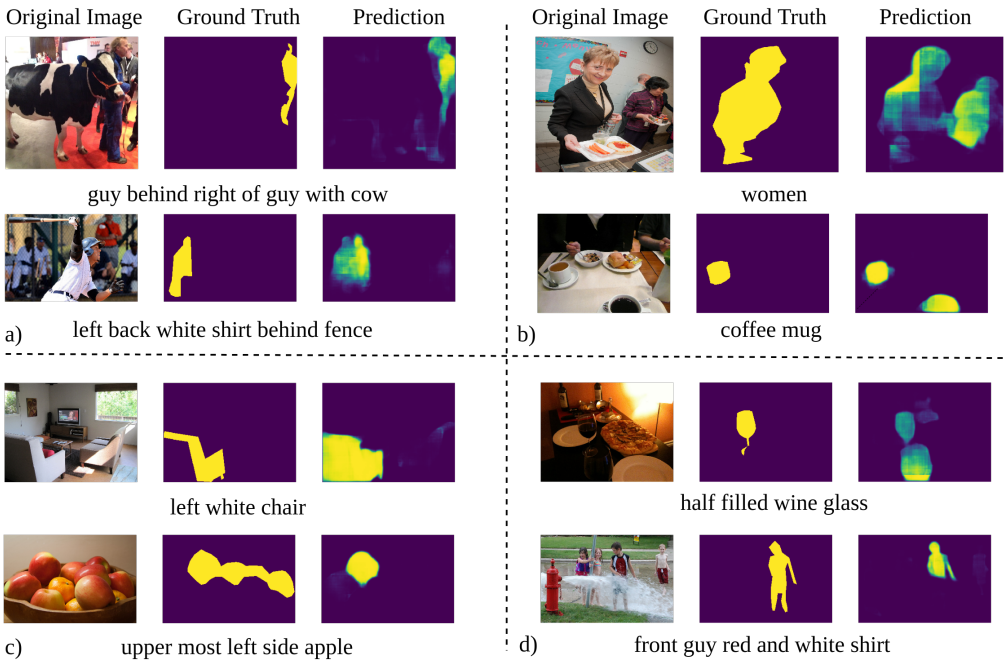

Figure 3: Some incorrect predictions of our model on UNC validation set. Each group (a-d) shows one pattern we observed within the predictions. In each group, the first column shows the original image, the second one is the ground truth mask and the third one is the prediction of our model.

5 Conclusion

We showed that modulating not only top-down but also bottom-up visual processing with language input improves the performance significantly. Our experiments showed that the proposed model achieves state-of-the-art results on 4 different benchmarks and performs significantly ($\approx$ 6 IoU) better than a baseline which uses language only to direct top-down attention. Our future work will focus on using it as a sub-component to solve a far more language-vision task like mapping natural language instructions to sequences of actions.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6077–6086, 2018a.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018b.

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 39–48, 2016.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

Paul Bloom. *How children learn the meanings of words*. MIT press, 2002.

Bastien Boutonnet and Gary Lupyan. Words jump-start vision: A label advantage in object recognition. *Journal of Neuroscience*, 35(25):9329–9335, 2015.

Ding-Jie Chen, Songhao Jia, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. See-through-text grouping for referring image segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 7454–7463, 2019a.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848, 2017.

Y.-W. Chen, Y.-H. Tsai, T. Wang, Y.-Y. Lin, and M.-H. Yang. Referring expression object segmentation with caption-aware consistency. In *British Machine Vision Conference (BMVC)*, 2019b.

Volkan Cirik, Taylor Berg-Kirkpatrick, and Louis-Philippe Morency. Using syntax to ground referring expressions in natural images. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Charles E Connor, Howard E Egeth, and Steven Yantis. Visual attention: bottom-up versus top-down. *Current biology*, 14(19):R850–R852, 2004.

Maurizio Corbetta and Gordon L Shulman. Control of goal-directed and stimulus-driven attention in the brain. *Nature reviews neuroscience*, 3(3):201–215, 2002.

Harm De Vries, Florian Strub, Jérémie Mary, Hugo Larochelle, Olivier Pietquin, and Aaron C Courville. Modulating early visual processing by language. In *Advances in Neural Information Processing Systems*, pp. 6594–6604, 2017.

Banchiamlack Dessalegn and Barbara Landau. More than meets the eye: The role of language in binding and maintaining feature conjunctions. *Psychological science*, 19(2):189–195, 2008.

Hugo Jair Escalante, Carlos A Hernández, Jesus A Gonzalez, Aurelio López-López, Manuel Montes, Eduardo F Morales, L Enrique Sucar, Luis Villaseñor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer vision and image understanding*, 114 (4):419–428, 2010.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338, 2010.

Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in neural information processing systems*, pp. 64–72, 2016.

Vittorio Gallese and George Lakoff. The brain's concepts: The role of the sensory-motor system in conceptual knowledge. *Cognitive neuropsychology*, 22(3-4):455–479, 2005.

Peng Gao, Hongsheng Li, Shuang Li, Pan Lu, Yikang Li, Steven CH Hoi, and Xiaogang Wang. Question-guided hybrid convolution for visual question answering. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 469–485, 2018.

Kirill Gavrilyuk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. Actor and action video segmentation from a sentence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5958–5966, 2018.

Ross Girshick. Fast r-cnn. *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015. doi: 10.1109/iccv.2015.169. URL http://dx.doi.org/10.1109/ICCV.2015.169.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun 2014. doi: 10.1109/cvpr.2014.81. URL http://dx.doi.org/10.1109/CVPR.2014.81.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. doi: 10.1109/iccv.2017.322. URL http://dx.doi.org/10.1109/ICCV.2017.322.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. *Lecture Notes in Computer Science*, pp. 108–124, 2016a. ISSN 1611-3349. doi: 10.1007/978-3-319-46448-0_7. URL http://dx.doi.org/10.1007/978-3-319-46448-0_7.

Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016b.

Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. doi: 10.1109/cvpr.2017.470. URL http://dx.doi.org/10.1109/CVPR.2017.470.

Zhiwei Hu, Guang Feng, Jiayu Sun, Lihe Zhang, and Huchuan Lu. Bi-directional relationship inferring network for referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4424–4433, 2020.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Tianrui Hui, Si Liu, Shaofei Huang, Guanbin Li, Sansi Yu, Faxi Zhang, and Jizhong Han. Linguistic structure guided context modeling for referring image segmentation. 2020.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Ray Jackendoff and Ray S Jackendoff. *Foundations of language: Brain, meaning, grammar, evolution*. Oxford University Press, USA, 2002.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 787–798, 2014.

Jin-Hwa Kim, Sang-Woo Lee, Donghyun Kwak, Min-Oh Heo, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Multimodal residual learning for visual qa. In *Advances in neural information processing systems*, pp. 361–369, 2016.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. 2016. URL https://arxiv.org/abs/1602.07332.

Ruiyu Li, Kaican Li, Yi-Chun Kuo, Michelle Shu, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Referring image segmentation via recurrent refinement networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5745–5753, 2018.

Zhenyang Li, Ran Tao, Efstratios Gavves, Cees GM Snoek, and Arnold WM Smeulders. Tracking by natural language specification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6495–6503, 2017.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Chenxi Liu, Zhe L. Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan L. Yuille. Recurrent multimodal interaction for referring image segmentation. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1280–1289, 2017.

Daqing Liu, Hanwang Zhang, Feng Wu, and Zheng-Jun Zha. Learning to assemble neural module tree networks for visual grounding. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances in neural information processing systems*, pp. 289–297, 2016.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 375–383, 2017.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, pp. 3, 2013.

Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *Proceedings of the IEEE international conference on computer vision*, pp. 1–9, 2015.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016. doi: 10.1109/cvpr.2016.9. URL http://dx.doi.org/10.1109/CVPR.2016.9.

Edgar Margffoy-Tuay, Juan C Pérez, Emilio Botero, and Pablo Arbeláez. Dynamic multimodal instance segmentation guided by natural language queries. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 630–645, 2018.

Lotte Meteyard, Bahador Bahrami, and Gabriella Vigliocco. Motion detection and motion verbs: Language affects low-level visual perception. *Psychological Science*, 18(11):1007–1013, 2007.

Dipendra Misra, Andrew Bennett, Valts Blukis, Eyvind Niklasson, Max Shatkhin, and Yoav Artzi. Mapping instructions to actions in 3d environments with visual goal prediction. *arXiv preprint arXiv:1809.00786*, 2018.

Varun K. Nagaraja, Vlad I. Morariu, and Larry S. Davis. Modeling context between objects for referring expression understanding. *Lecture Notes in Computer Science*, pp. 792–807, 2016. ISSN 1611-3349. doi: 10.1007/978-3-319-46493-0_48. URL http://dx.doi.org/10.1007/978-3-319-46493-0_48.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

Friedemann Pulvermüller. Words in the brain's language. *Behavioral and brain sciences*, 22(2): 253–279, 1999.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, Jun 2017. ISSN 2160-9292. doi: 10.1109/tpami.2016.2577031. URL http://dx.doi.org/10.1109/TPAMI.2016.2577031.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Hengcan Shi, Hongliang Li, Fanman Meng, and Qingbo Wu. Key-word-aware network for referring expression image segmentation. In *The European Conference on Computer Vision (ECCV)*, September 2018.

Xingjian SHI, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun WOO. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 28*, pp. 802–810. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/5955-convolutional-lstm-network-a-machine-learning-approach-for-precipitation-nowc pdf.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. A corpus of natural language for visual reasoning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 217–223, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P17-2034.

Jan Theeuwes. Top–down and bottom–up control of visual selection. *Acta psychologica*, 135(2): 77–99, 2010.

Gabriella Vigliocco, David P Vinson, William Lewis, and Merrill F Garrett. Representing the meanings of object and action words: The featural and unitary semantic space hypothesis. *Cognitive psychology*, 48(4):422–488, 2004.

Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. doi: 10.1109/cvpr.2019.00206. URL http://dx.doi.org/10.1109/CVPR.2019.00206.

Terry Winograd. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.

Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pp. 451–466. Springer, 2016.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057, 2015.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 21–29, 2016.

Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-modal self-attention network for referring image segmentation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. doi: 10.1109/cvpr.2019.01075. URL http://dx.doi.org/10.1109/CVPR.2019.01075.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions. *Lecture Notes in Computer Science*, pp. 69–85, 2016.

Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L. Berg. Mattnet: Modular attention network for referring expression comprehension. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. doi: 10.1109/cvpr.2018.00142. URL http://dx.doi.org/10.1109/CVPR.2018.00142.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6720–6731, 2019.

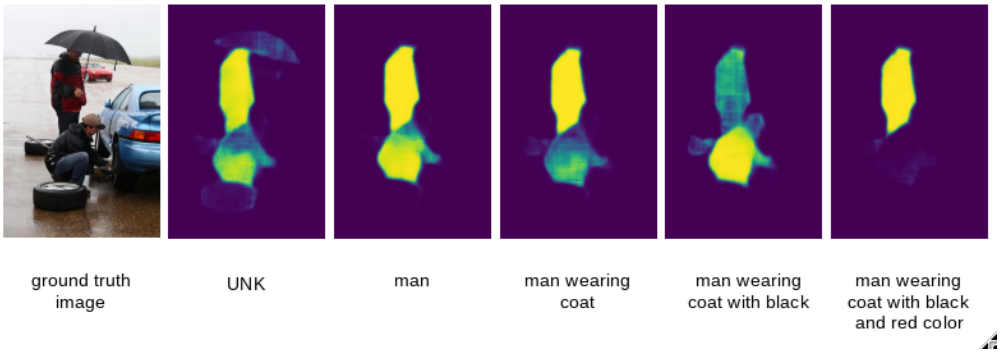

Figure 4: Incremental segmentation result of our model on UNC test split instance.

A Appendix

A.1 Incremental Segmentation

We also analyzed the behaviour of our model with respect to incrementally given language input in Figure 4. In the initial step, our model only sees an unknown word token. In the second step, our model sees only the first word of the expression. In every step, our model starts to see a new word in addition to the previous ones. Figure 4 shows that our model can capture ambiguities in input image and expressions pairs. For unknown token input, our model captures all salient objects since there is no restriction. When the *man* word is fed, the model discards unrelated objects like umbrella and wheel. Additionally, when our model starts to see color words for coat, it initially focuses on both men, since both coats has black color. When it sees the final expression, it shifts its focus to the correct object.