Generating Multi-sentence Natural Language Descriptions of Indoor Scenes

Dahua Lin[1]
dhlin@ie.cuhk.edu.hk
Sanja Fidler[2]
fidler@cs.toronto.edu
Chen Kong[3]
chenk@cs.cmu.edu
Raquel Urtasun[2]
urtasun@cs.toronto.edu

[1] Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, China.
[2] Department of Computer Science, University of Toronto, Toronto, Canada.
[3] Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA.

Abstract

This paper proposes a novel framework for generating lingual descriptions of indoor scenes. Whereas substantial efforts have been made to tackle this problem, previous approaches focusing primarily on generating a single sentence for each image, which is not sufficient for describing complex scenes. We attempt to go beyond this, by generating coherent descriptions with multiple sentences. Our approach is distinguished from conventional ones in several aspects: (1) a 3D visual parsing system that jointly infers objects, attributes, and relations; (2) a generative grammar learned automatically from training text; and (3) a text generation algorithm that takes into account coherence among sentences. Experiments on the NYU-v2 dataset show that our framework is able to generate natural multi-sentence descriptions, outperforming those produced by a baseline.

1 Introduction

Image understanding has been the central goal of computer vision. Whereas a majority of work on image understanding focuses on class-based annotation, we believe, however, that describing an image using natural language is still the best way to show one's understanding. The task of automatically generating textual descriptions for images has received increasing attention from both the computer vision and natural language processing communities. This is an important problem, as an effective solution to this problem can enable many exciting real-world applications, such as human robot interaction, image/video synopsis, and automatic caption generation.

While this task has been explored in previous work, existing methods mostly rely on predefined templates [1, 15], which often result in tedious descriptions. Another line of work solves the description generation problem via retrieval, where a description for an image is borrowed from semantically most similar image from the training set [8, 27]. This setting is, however, less applicable to complex scenes composed of a large set of objects in diverse configurations, such as for example indoor environments.

Pages 93.1-93.13
DOI: https://dx.doi.org/10.5244/C.29.93

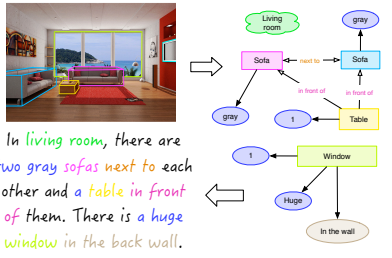


Figure 1: Our method visually parses an RGB-D image to get a *scene graph* that represents objects, their attributes and relations between objects. Based on the scene graph we then generate a multi-sentence textual description via a learned grammar. The description generation takes into account co-reference and saliency of how people describe scenes.

Recently, the field has witnessed a boom in generating image descriptions via deep neural networks [4, 12, 13] which are able to both, learn a weak language model as well as generalize description to unseen images. These approaches typically represent the image and words/sentences with vectors and reason in a joint embedding space. The results have been impressive, perhaps partly due to powerful representation on the image side [16]. This line of work mainly generates a single sentence for each image, which typically focus on one or two objects and typically contain very few prepositional relations between objects.

In this paper, we are interested in generating multi-sentence descriptions of cluttered indoor scenes, which is particularly relevant for indoor robotics. Complex, multi-sentence output requires us to deal with challenging problems such as consistent co-referrals to visual entities across sentences. Furthermore, the sequence of sentences needs to be as natural as possible, mimicking how humans describe the scene. This is particularly important for example in the context of social robotics to enable realistic communications.

Towards this goal, we develop a framework with three major components: (1) a *holistic visual parser* based on [21] that couples the inference of objects, attributes, and relations to produce a semantic representation of a 3D scene (Fig. 1); (2) a *generative grammar* automatically learned from training text; and (3) a *text generation algorithm* that takes into account subtle *dependencies across sentences*, such as logical order, diversity, saliency of objects, and co-reference resolution.

To test the effectiveness of our approach, we construct an augmented dataset based on NYU-RGBD [35], where each scene is associated with up to 5 natural language descriptions from human annotators. This allows us to learn a language model to describe images the way that humans do. Experiments show that our method produces natural descriptions, significantly improving the F-measures of ROUGE scores over the baseline.

2 Related Work

A large body of existing work deals with images and text in one form or the other. The dominant subfield exploits text in the form of tags or short sentences as weak labels to learn visual models [10, 20, 29, 36], as well as attributes [25, 34]. This type of approaches have also been explored in videos to learn visual action models from textual summaries of videos [30], or learning visual concepts from videos described with short sentences [41]. Another direction is to exploit short sentences associated with images in order to improve visual recognition tasks [9, 14]. Just recently, an interested problem domain was introduced in [23] with the aim to learn how to answer questions about images from Q&A examples. In [22], the authors address visual search with complex natural lingual queries.

There has been substantial work in automatically generating a caption for an image. The most popular approach has been to retrieve a sentence from a large corpus based on visual similarity [8, 18, 27, 32, 40]. This line of work bypasses having to deal with language tem-

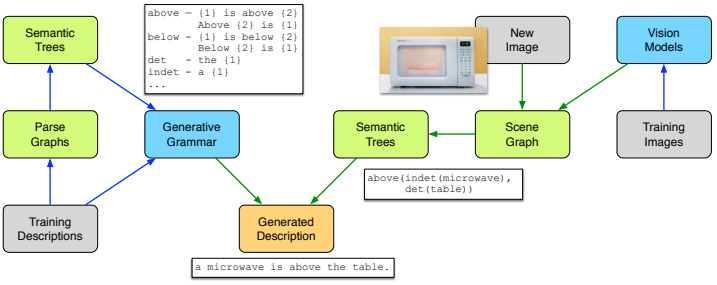


Figure 2: The overall framework for description generation. The task consists of the training and the testing phase. In training, the vision models and the generative grammar are respectively learned from a set of RGB-D images and their descriptions. In testing, given a new image, it constructs a scene graph taking into account objects, their attributes and relationships between objects, and transforms it to a series of semantic trees. The learned grammar then generates textual descriptions for these trees.

plate specification or template learning. However, typically such approaches adopt a limited representation such as triplets action-object-scene [8]. This makes a restrictive setting, as neither the image representation nor the retrieved sentence can faithfully model a truly complex scene. In [19] the authors go further by only learning phrases from related images.

Parallel to our work, a popular approach has been to generate captions with deep networks [4, 6, 7, 12, 13, 24, 38]. These methods encode the image as well as a sentence with a vector representation and learn a joint embedding between the two modalities. The output is typically a short sentence. In contrast, our goal here is to generate *multiple* dependent sentences that describe the salient objects in the scene, their properties and spatial relations.

Generating descriptions has also been explored in the video domain. [1, 15] output a video description in the form of subject-action-object. In [5], "concept detectors" are formed, which are detectors for combined object and action or scene in a particular chunk of a video. Via lingual templates the concept detectors of particular types then produce cohesive video descriptions. Due to a limited set of concepts and templates the final descriptions do not seem very natural. [32] predicts semantic representations from low-level video features and uses machine translation techniques to generate a sentence.

The closest to our work is [17, 19, 26] which, like us, is able to describe objects, their modifiers, and prepositions between objects. However, our paper differs from [17, 26] in several important ways. We reason in 3D as opposed to 2D giving us more natural *physical* interpretations. We aim to describe rich indoor scenes that contain many objects of various classes and appear in various arrangements. In such a setting, describing every detectable object and all relations between them as in [17] would generate prohibitively long and unnatural descriptions. Our model tries to mimic *what* and *how* people describe such complex 3D scenes, thus taking into account visual saliency at the level of objects, attributes and relations, as well as the ordering and coherence of sentences. Another important aspect is that instead of using a few hand-crafted templates, we *learn* the grammar from training text.

3 Framework Overview

Our framework for generating descriptions for indoor scenes is based on a key rationale: images and their corresponding descriptions are two different ways to express the underlying *common semantics* shared by both. As shown in Fig. 2, given an image, it first recovers the *semantics* through holistic visual analysis [21], which results in a *scene graph* that captures

detected objects and the spatial relations between them (*e.g. on-top-of* and *near*, etc).

The *semantics* embodied by a visual scene usually has multiple aspects. When describing such a *complex* scene, humans often use a paragraph comprised of multiple sentences, each focusing on a specific aspect. To imitate this behavior, this framework transforms the *scene graph* into a sequence of *semantic trees*, and yields multiple sentences, each from a *semantic tree*. To make the results as natural as possible, we adopt two strategies: (1) Instead of prescribing templates in advance, we learn the *grammar* from a training set – a set of RGB-D scenes with descriptions provided by humans. (2) We take into account dependencies among sentences, including *logical order*, *saliency*, *coreference* and *diversity*.

4 From RGB-D Images to Semantics

Given an RGB-D image, we extract semantics via holistic visual parsing. We first parse the image to obtain the objects of interest, their attributes, and their physical relations, and then construct a *scene graph*, which provides a coherent summary of these aspects.

4.1 Holistic Visual Parsing

To parse the visual scene we use a recently proposed approach for 3D object detection in RGB-D data [21]. We briefly summarize this approach here. First, a set of *"objectness" regions* are generated following [2], which are encouraged to respect intensity as well as occlusion boundaries in 3D. These regions are projected to 3D via depth and then cuboids are fit tightly around them, under the constraint that they are parallel to the ground floor.

A *holistic CRF model* is then constructed to jointly reason about the classes of the *cuboids* as well as the class of the scene (*e.g.*, kitchen, bathroom). The CRF thus has a random variable for each cuboid representing its class, and a variable for the scene. To have the possibility to remove a bad, non-object cuboid, we have an additional background state for each cuboid. The model exploits various geometric and semantic relations by incorporating them into the CRF formulation as *potentials*, which are summarized below:

Scene Appearance. To incorporate global information, a unary potential over the scene label is computed by means of a logistic on top of the scene classification score [39].

Cuboid class potential. Appearance-based classifiers, including CPMC-o2 [3], superpixel scores [31] are used to classify cuboids into a pre-defined set of object classes. In this paper, we additionally use CNN [16] features for classification. The classification scores for each cuboid are used as different unary potentials in the CRF.

Object geometry. Cuboids are also classified based on geometric features (*e.g. height*, *aspect ratio*, etc) with SVM, and the classification scores used as another unary potential.

Semantic context. Two co-occurrence relationships are used: *scene-object* and *object-object*. The potential values are estimated from the training set by counting the co-occurences.

Geometric context. Two potentials are used to exploit the spatial relations between cuboids in 3D, encoding *close-to* and *on-top-of* relations. The potentials are defined to be the empirical co-occurrence frequencies for each type of relation.

The CRF weights to combine the potentials are learned with a primal dual learning framework [11], and inference of class labels is done with an approximated algorithm [33].

4.2 Scene Graphs

Based on the extracted visual information, we construct a *scene graph* that captures *objects*, their *attributes*, such as color and size, and the relations between them. In particular, a *scene*

graph uses *nodes* to represent objects and their attributes, and *edges* to represent relations between nodes. Here, we consider three kinds of edges: *attribute edges* that link objects to their attributes, *position edges* that represent the positions of objects relative to the scene, (*e.g. corner-of-room*), and *pairwise edges* that characterize the relative positions between objects (*e.g. on-top-of* and *next-to*).

Given an image, a set of objects (with class labels) and the scene class are obtained through visual parsing as explained in the previous Section. However, to form a *scene graph*, we still need further analysis to extract *attributes* and *relations*. For each object we also compute *saliency*, i.e. how likely an object will be described. We next describe how we obtain such information.

Object attributes: For each object, we use RGB histograms and C-SIFT, and cluster them to obtain a visual word representation. We train classifiers for nine colors that are most mentioned in the training set, as well as two material properties (*wooden* and *bright*). We also train classifiers for four different sizes (*wide*, *tall*, *large*, and *small*) using geometric features. To encode the correlations between size and the object class, we augment the feature with a class indicator vector.

Object saliency: The dataset of [14] contains alignment between the nouns in a sentence and the visual objects in the scene. We make use of this information to train a classifier predicting whether an object in the scene is likely to be mentioned in text. We train an SVM classifier using class-based features (classification scores for each cuboid), geometric relations (volume, distance to camera), and color features.

Object relations: We consider six types of *object locations* (*corner-of-room*, *front-of-camera*, *far-away-from-camera*, *center-of-room*, *left-of-room*, *right-of-room*), and eight types of *pairwise relations* (*next-to*, *near*, *top-of*, *above*, *in-front-of*, *behind*, *to-left-of*, and *to-right-of*). We manually specify a few rules for deciding whether a relation is present or not.

5 Generating Lingual Descriptions

Given a *scene graph*, we generate a descriptive paragraph in two steps. First, we transform the scene graph into a sequence of *semantic trees*, each focusing on a certain *semantic aspect*. Then, we produce sentences, one from each semantic tree, following a *generative grammar*.

5.1 Semantic Trees

A *semantic tree* captures information such as *what entities are being described* and *what are the relationships between them*. Specifically, a semantic tree contains a set of *terminal nodes* corresponding to individual entities or their attributes and *relational nodes* that express relations among them. Consider a sentence *"A red box is on top of a table"*. The corresponding semantic tree can be expressed as

```
on-top-of(indet(color(box, red)), indet(table))
```

This tree has three terminals: *"box"*, *"table"*, and *"red"*. The relation node *"color(box, red)"* describes the relation between *"box"* and *"red"*, namely, *"red"* specifying the color of the *"box"*. The relation *"indet"* qualifies the cardinality of its child; while *"on-top-of"* characterizes the spatial relation between its children.

5.2 Dependencies among Sentences

In human descriptions, sentences are put together in a way that makes the resultant paragraphs coherent. In particular, the *dependencies* among sentences, as outlined below, play a crucial role in preserving the coherence a descriptive paragraph:

Logical order. When describing a scene, people present things in certain orders. The leading sentence often mentions the type of the entire scene and one of the most salient object, *e.g.* *"There is a table in the dining room."*

Diversity. People generally avoid using the same prepositional relation in multiple sentences. Also, when an object is mentioned in multiple sentences, it usually plays a different role, *e.g.* *"There is a table near the wall. On top of the table is a microwave oven."* Here, *"table"* respectively serves as a *source* and a *target* in these two sentences[1].

Saliency. *Saliency* influences the order of sentences. The statistics in [14] shows that bigger objects are often mentioned earlier on in a description and co-referred across sentences, *e.g.* one would say *"This room has a dining table with a mug on top. Next to the table is a chair."* and not *"There is a mug on a table. Next to the mug is a chair."* *Saliency* also depends on context, *e.g.* for bathrooms, toilets are often mentioned.

Co-reference. When an object is mentioned for the second time following its debut, a pronoun is often used to make the sentence concise.

Richness vs. Conciseness. When talking about an object for the first time, describing its color/size makes the sentence interesting and informative. However, this is generally unnecessary the next time the object is mentioned.

5.3 From Scene Graphs to Semantic Trees

Motivated by these considerations, we devise a method below that transforms a *scene graph* into a sequence of *semantic trees*, each for a sentence.

First of all, we initialize $w_i^s = w_i^t = \mathfrak{s}_i \cdot \mathfrak{c}_i$. Here, w_i^s and w_i^t are the weights that respectively control how likely the i-th object will be chosen as a *source* or a *target* in the next sentence; $\mathfrak{s}_i$ is a positive value measuring the *saliency* of the i-th object, while $\mathfrak{c}_i$ is given by the classifier to indicate its confidence as to whether it makes a correct prediction of the object's class. These weights are updated as the generation proceeds.

To generate the leading sentence, we first draw a *source* i with a probability proportional to w_i^s, and create a semantic tree by choosing a relation, say *"in"*, which would lead to a sentence like *"There is a table in the dining room."* Once the i-th object is chosen to be a source, w_i^s will be set to 0, precluding it from being chosen as a source again. However, w_i^t remains unchanged, as it remains fine for it to serve as a target later.

For each subsequent sentence, we draw a source i, a target j, and a relation r between i and j, with probability proportional to $w_i^s w_j^t \rho_r$, where ρ_r is the prior weight of the relation r. At each iteration, one may also choose to terminate without generating a new sentence, with a probability proportional to a positive value τ. These choices together result in a semantic tree in the form of *"r(make_tree(i), make_tree(j))"*. Here, *"make_tree(i)"* creates a sub-tree describing the object i, which may be *"indet(color(table, black))"* when the color is known.

After the generation of this semantic tree, the weights w_i^s, w_j^t, and ρ_r will be set to zero to prevent the objects i and j from being used again for the same role, and the relation r

[1]Each relation is considered as an edge. For example, in phrases *"A on-top-of B"* and *"A near B"*, *"A"* is considered as the source, while *"B"* considered as the target.

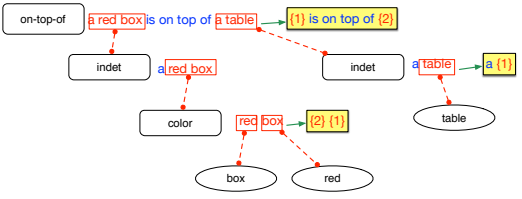


Figure 3: Deriving templates by matching semantic nodes to parts of sentence. Starting from root node, our learning algorithm identifies ranges of words corresponding to the child nodes, and replaces them with a placeholder to obtain a template. This proceeds downward recursively until all relation nodes are processed.

from being chosen next time. Our algorithm also takes care of *co-references* – if an object is selected again in the next sentence, it will be replaced by a pronoun.

5.4 Grammar and Derivation

Given a *semantic tree*, our approach produces a sentence via a *generative grammar*, *i.e.* a map from each semantic relation to a set of templates (derivation rules), as illustrated below:

```
indet      --> a {1}
color      --> {2} {1}
on-top-of --> {1} is on top of {2}
              On top of {2} is {1}
              There is {1} on top of {2}
```

Each template has a weight set to its frequency in the training set. Generating a sentence from a semantic tree proceeds from the root downward, recursively to the terminals. For each relation node, a template is chosen, with a probability proportional to the associated weight. Below is an example showing how a sentence is derived following the grammar above.

```
   {on-top-of(indet(color(box, red)), indet(table))}
=> {indet(color(box, red))} is on top of {indet(table)}
=> a {color(box, red)} is on top of a table
=> a red box is on top of a table
```

As the choices of templates for relational nodes are randomized, different sentences can be derived for the same tree, with different probabilities.

5.5 Learning the Grammar

The *grammar* for generating sentences are often specified manually in previous work [1, 5]. This way, however, is time consuming, unreliable, and tends to oversimplify the language. In this work, we explore a new approach, that is, to learn the grammar from data. The basic idea is to construct a semantic tree from each sentence through linguistic parsing, and then derive the templates by matching nodes of the semantic tree to parts of the sentence.

First, we use the Stanford parser [37] to obtain a *parse tree* for each sentence, which is then simplified through a series of filtering operations. For example, we merge noun phrases (*e.g.* *"fire distinguisher"*) into a single node and compress common prepositional phrases (*e.g.* *"in the left of"*) into a single link.

A semantic tree can then be derived by recursively translating the simplified trees. This is straightforward. For example, a noun *"box"* with an adjective *"red"* will be translated into *"color(box, red)"*; a noun with a definite or indefinite article will be translated into an *det* and *indet* relation node; two nouns or noun phrases *"A"* and *"B"* linked by a prepositional link *"above"* will be translated into *"above(A, B)"*.

With a sentence and a semantic tree constructed, we can derive the template through *recursive matching*, where matched children are replaced by a placeholder, while other words

objects	config	ROUGE1			ROUGE2			ROUGES		
		R	P	F	R	P	F	R	P	F
baseline		0.3000	0.2947	0.2968	0.0667	0.0657	0.0661	0.1026	0.1006	0.1014
GT	L0	0.3332	0.3249	0.3281	0.0786	0.0765	0.0773	0.1372	0.1334	0.1348
GT	L1	0.3378	0.3294	0.3327	0.0838	0.0816	0.0824	0.1397	0.1359	0.1373
GT	L2	0.3392	0.3308	0.3340	0.0849	0.0827	0.0835	0.1409	0.1370	0.1385
GT	L3	0.3770	0.3676	0.3712	**0.1092**	**0.1067**	**0.1076**	**0.1629**	**0.1584**	**0.1601**
GT	L4	**0.3775**	**0.3680**	**0.3716**	0.1064	0.1040	0.1049	0.1598	0.1554	0.1570
GT	L5	0.3755	0.3658	0.3695	0.1008	0.0984	0.0993	0.1563	0.1519	0.1536
Real	L0	0.3243	0.3161	0.3192	0.0752	0.0735	0.0742	0.1306	0.1270	0.1283
Real	L1	0.3347	0.3266	0.3296	0.0814	0.0795	0.0802	0.1362	0.1325	0.1338
Real	L2	0.3338	0.3256	0.3286	0.0816	0.0796	0.0803	0.1356	0.1319	0.1332
Real	L3	0.3641	0.3541	0.3580	0.1045	0.1019	0.1029	0.1546	0.1499	0.1517
Real	L4	0.3663	0.3560	0.3600	0.1039	0.1011	0.1022	0.1534	0.1486	0.1504
Real	L5	0.3675	0.3570	0.3611	0.1021	0.0994	0.1004	0.1526	0.1478	0.1496

Table 1: ROGUE scores for the baseline and our approach under different configurations. "GT" and "Real" refer to results obtained based on GT objects and detections [21], respectively. For each metric, we report recall (R), precision (P), and F-scores (F) averaged over all scenes and 10 randomized runs.

are preserved literally in the template. Fig. 3 illustrates this procedure. We collect templates for each relation, and set the weight of each template to its frequency. Empirically, we observed a long tailed distribution – a small number of templates occur many times, while a dominant portion of templates are used sporadically. To improve the reliability, we discard all templates that occur less than 5 times and all relations whose total weight is less than 20.

6 Experimental Evaluation

We test the proposed framework on the NYU-v2 dataset [35] augmented with an additional set of textual descriptions, one for each image. Particularly, we focus on assessing both the relevance and quality of the generated descriptions.

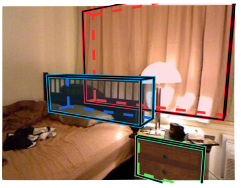
NYU-v2 has 1449 RGB-D images of indoor scenes (*e.g.* dining rooms, kitchens, etc). We follow the train/test partition used in [21] with 795 training scenes, while the test set contains the remaining 654. We use descriptions from [14] which were collected by asking MTurkers to provide detailed descriptions of scenes. The number of sentences per description ranges from 1 to 10 with an average of 3. There are on average 40 words per description.

We learn the generative grammar using the algorithm described in Section 5.5 from the training set of descriptions. We also train the CRF for visual analysis and apply it to detect objects and predict their attributes and relations, following the procedure described in Section 4.1. These models are then used to produce textual descriptions for each test scene.

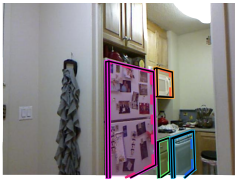
6.1 Performance Metrics

To evaluate our method, we look at metrics typically used in machine translation, which include BLEU [28] and ROUGE metrics. BLEU measures precision on n-grams, and is thus less suitable for our goal of image description, as already noted in [5, 26]. On the other hand, ROUGE is an n-gram recall oriented measures which evaluates the information coverage between summaries produced by the human annotators and those automatically produced by systems. ROUGE-1 (unigram) recall is the best option to use for comparing descriptions based only on predicted keywords [5]. ROUGE-2 (bigram) and ROUGE-SU4 (skip-4 bigram) are best to evaluate summaries with respect to coherence and fluency. We use ROUGE metrics following [5] who uses it to evaluate video summarization.

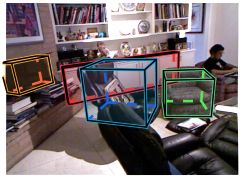
There is a brown bed in the bedroom. The bed is in front of a headboard. Near the bed is a blinds. We can see a brown curtain near the blinds. There is a chest near the headboard.



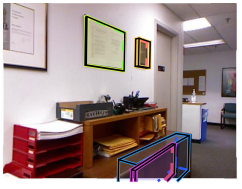
A wooden curtain is in the bedroom. The curtain is on top of a wooden headboard. We can see a chest in front of the curtain. The headboard is near the chest.



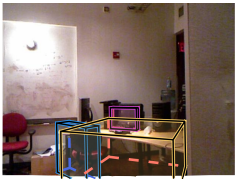
In the kitchen, there is a refrigerator. A green cabinet is near a gray oven. Near the refrigerator is the cabinet. We can see a microwave near the cabinet. The oven is behind the refrigerator.



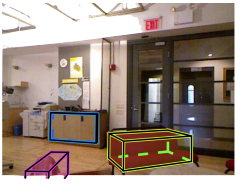
There is a sofa in the living room. Behind the sofa is a white cabinet. We can see a black chair in front of the cabinet. There is a mantel near the chair.



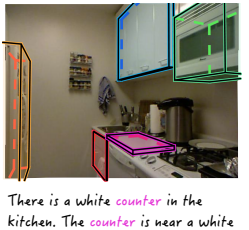
In the office, there is a board. We can see a cabinet in front of the board. We can see a monitor near the board. In the office, there is a table.



In the living room, there is a monitor. The monitor is behind a chair. We can see the monitor on top of a table. There is the table near the monitor. The chair is near the table.



In the kitchen, there is a chair. A cabinet is behind a sofa. The sofa is near the chair.



There is a white counter in the kitchen. The counter is near a white cabinet. Near a refrigerator is the cabinet. We can see a green microwav on the right of the cabinet. The refrigerator is near a shelf.

Figure 4: Examples of descriptions generated using our framework. In the top row, the method builds on the GT cuboids, while the bottom row shows results using the visual parser. Note that in the case of GT, the input is the full set of GT objects for the image, thus the method still needs to take into account the saliency of what to talk about. We color-code object cuboids and nouns referring to them in text.

6.2 Comparison of Results

The proposed text generation method has five optional switches, controlling whether the following features are used during generation: (1) `diversity`: encourage diversity of the sentences by suppressing the entities and relations that have been mentioned; (2) `saliency`: draw salient objects with higher probability; (3) `scene`: leading sentence mentions the class of the scene; (4) `attributes`: use colors and sizes to describe objects when they are available; (5) `coreference`: use a pronoun to refer to an object when it is mentioned in the previous sentence. We test the approach with six feature-levels, level-0 to level-5, where the level-k configuration uses the first k features when generating the sentences. In particular, level-0 uses none of the above features, and thus each sentence is generated independently using the grammar; level-5 uses all of these features.

We compare our method to an intelligent baseline which follows a conventional approach in description generation. The baseline describes an image by retrieving visually the most similar image from the training set, and simply uses its description. To compute our baseline, we use a battery of visual features such as spatial pyramids of SIFT, HOG, LBP, geometric context, etc, and kernels with different distances. We use [39] to compute the kernels. Based on a combined kernel, we simply retrieve the training image with the highest matching score.

Table 1 shows results. We evaluate two settings: using ground-truth objects (denoted with GT) and using the results obtained via the visual parser (denoted with Real). We can see that the proposed method outperforms the baseline in all three ROGUE measures. Also, configurations above level 3 are better than level 1 and 2, which indicates that a special leading sentence that gives an overview of the scene is important for description generation.

Figure 4 shows descriptions generated using our approach on a diverse set of scenes. It can be seen that linguistic issues such as sentence diversity, using attributes to describe objects, and using pronouns for coreferences have been properly addressed. However, there remain some problems that need future efforts to address. For example, since the choices of templates for different sentences are independent, sometimes an unfortunate selection of a template sequence may make the paragraph slightly unnatural.

7 Conclusion

We presented a new framework for generating natural descriptions of indoor scenes. Our framework integrates a CRF model for visual parsing, a generative grammar automatically learned from training text, as well as a transformation algorithm to derive semantic trees from scene graphs, which takes into account the dependencies across sentences. Our experiments show better descriptions than those produced by a baseline. This indicates that high quality description generation requires not only reliable image understanding, but also delicate attention to linguistic issues, such as diversity, coherence, and logical order of sentences.

References

[1] A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, S. Narayanaswamy, D. Salvi, L. Schmidt, J. Shangguan, J. Siskind, J. Waggoner, S. Wang, J. Wei, Y. Yin, and Z. Zhang. Video-in-sentences out. In *UAI*, 2012.

[2] J. Carreira and C. Sminchisescu. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *TPAMI*, 2012.

[3] J. Carreira, R. Caseiroa, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV12*, 2012.

[4] X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. In *arXiv:1411.5654*, 2014.

[5] P. Das, C. Xu, R. F. Doell, and J. J Corso. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *CVPR*, 2013.

[6] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *arXiv:1411.4389*, 2014.

[7] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollar, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. In *arXiv:1411.4952*, 2014.

[8] A. Farhadi, M. Hejrati, M. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences for images. In *ECCV*, 2010.

[9] S. Fidler, A. Sharma, and R. Urtasun. A sentence is worth a thousand pixels. In *CVPR*, 2013.

[10] A. Gupta and L. Davis. Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In *ECCV*, 2008.

[11] T. Hazan and R. Urtasun. A primal-dual message-passing algorithm for approximated large scale structured prediction. In *NIPS*, 2010.

[12] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *arXiv:1412.2306*, 2014.

[13] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *arXiv:1411.2539*, 2014.

[14] C. Kong, D. Lin, M. Bansal, R. Urtasun, and S. Fidler. What are you talking about? text-to-image coreference. In *CVPR*, 2014.

[15] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama. Generating natural-language video descriptions using text-mined knowledge. In *AAAI*, July 2013. URL http://www.eecs.berkeley.edu/~sguada/pdfs/2013-AAAI-generating-final.pdf.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[17] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. Berg, and T. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.

[18] P. Kuznetsova, V. Ordonez, A. Berg, T. Berg, and Y. Choi. Collective generation of natural image descriptions. In *Association for Computational Linguistics (ACL)*, 2012.

[19] P. Kuznetsova, V. Ordonez, T. L. Berg, and Y. Choi. Treetalk: Composition and compression of trees for image descriptions. In *TACL*, 2014.

[20] L. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding:classification, annotation and segmentation in an automatic framework. In *CVPR*, 2009.

[21] D. Lin, S. Fidler, and R. Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*, 2013.

[22] D. Lin, S. Fidler, C. Kong, and R. Urtasun. Visual semantic search: Retrieving videos via complex textual queries. In *CVPR*, 2014.

[23] M. Malinowski and M. Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *NIPS*, 2014.

[24] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Explain images with multimodal recurrent neural networks. In *arXiv:1410.1090*, 2014.

[25] C. Matuszek, N. FitzGerald, L. Zettlemoyer, L. Bo, and D. Fox. A joint model of language and perception for grounded attribute learning. In *International Conference on Machine Learning*, 2012.

[26] M. Mitchell, J. Dodge, A. Goyal, Kota Yamaguchi, K. Sratos, X. Han, A. Mensch, A. C. Berg, T. L. Berg, and H. Daume III. Midge: Generating image descriptions from computer vision detections. In *European Chapter of the Association for computational Linguistics*, 2012.

[27] V. Ordonez, G. Kulkarni, and T. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.

[28] K. Papineni, S.; Roukos, T. Ward, and W. J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.

[29] A. Quattoni, M. Collins, and T. Darrell. Learning visual representations using images with captions. In *CVPR07*, 2007.

[30] V. Ramanathan, P. Liang, and L. Fei-Fei. Video event understanding using natural language descriptions. In *ICCV*, 2013.

[31] X. Ren, L. Bo, and D. Fox. Rgb-(d) scene labeling: Features and algorithms. In *CVPR*, 2012.

[32] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *ICCV*, 2013.

[33] A. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Distributed message passing for large scale graphical models. In *CVPR*, 2011.

[34] C. Silberer, V. Ferrari, and M. Lapata. Models of semantic representation with visual attributes. In *Association for Computational Linguistics (ACL)*, 2013.

[35] N. Silberman, P. Kohli, D. Hoiem, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[36] R. Socher and L. Fei-Fei. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *CVPR*, 2010.

[37] K. Toutanova, D. Klein, and C. Manning. Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL*, 2003.

[38] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *arXiv:1411.4555*, 2014.

[39] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

[40] Y. Yang, C. L. Teo, H. Daumé, III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, pages 444–454, 2011.

[41] H. Yu and J. M. Siskind. Grounded language learning from video described with sentences. In *Association for Computational Linguistics (ACL)*, 2013.