BabyTalk: Understanding and Generating Simple Image Descriptions

Girish Kulkarni, Visruth Premraj, Vicente Ordonez, *Student Member*, *IEEE*, Sagnik Dhar, Siming Li, *Student Member*, *IEEE*, Yejin Choi, Alexander C. Berg, *Member*, *IEEE*, and Tamara L. Berg, *Member*, *IEEE*

Abstract—We present a system to automatically generate natural language descriptions from images. This system consists of two parts. The first part, content planning, smooths the output of computer vision-based detection and recognition algorithms with statistics mined from large pools of visually descriptive text to determine the best content words to use to describe an image. The second step, surface realization, chooses words to construct natural language sentences based on the predicted content and general statistics from natural language. We present multiple approaches for the surface realization step and evaluate each using automatic measures of similarity to human generated reference descriptions. We also collect forced choice human evaluations between descriptions from the proposed generation system and descriptions from competing approaches. The proposed system is very effective at producing relevant sentences for images. It also generates descriptions that are notably more true to the specific image content than previous work.

Index Terms—Computer vision, image description generation

1 Introduction

Natural language, whether spoken, written, or typed, makes up much of human communication. A significant amount of this language describes the visual world either directly around us or in images and video. Connecting visual imagery with visually descriptive language is a challenge for computer vision that is becoming more relevant as recognition and detection methods are beginning to work.

There is an enormous amount of visually descriptive text available—both closely associated with images in captions and in pure text documents. Studying such language has the potential to provide 1) training data for understanding how people describe the world, as well as 2) more general knowledge about the visual world implicitly encoded in human language.

This paper explores techniques to benefit from both of these possible sources of information. We exploit the first type of textual information as a prior to modulate global inference over computer vision-based recognition of objects, appearance characteristics, and background regions. The second type of language information is exploited to convert the resulting keyword-based predictions into complete and human-like natural language descriptions.

In addition to the direct outputs of our system—automatically generated natural language descriptions for images—there are also a number of possible related applications. These include improving accessibility of images for the visually impaired and creating text-based indexes of visual data for improving image retrieval algorithms. In addition, our work is in line with a more general research direction toward studying visually descriptive text and delving deeper into the connection between images and language that has the potential to suggest new directions for research in computer vision. For instance, a better understanding of what information it is important to extract from an image in order to choose appropriate descriptive language may lead to new or more observer-focused goals for recognition.

It is subtle, but several factors distinguish the challenge of taking images as input and generating natural language descriptions from many other tasks in computer vision. As examples, when forming descriptive language, people go beyond simply listing which objects are present in an image—this is true even for very low-resolution images [40] and for very brief exposure to images [16]. In both of these settings, and in language in general, people include specific information describing not only scenes, but specific objects, their relative locations, and modifiers adding additional information about objects. Mining the absolutely enormous amounts of visually descriptive text available in special library collections and on the web in general makes it possible to discover what modifiers people use to describe objects and what prepositional phrases are used to describe relationships between objects. These can be used to select and train computer vision algorithms to recognize these constructs in images. The output of the computer vision processing can also be "smoothed" using language statistics

• *The authors are with the Computer Science Department, Stony Brook University, Stony Brook, NY 11794. E-mail: {girish86, visroo, vicente.ordonez, sagnik.dhar, seemingly}@gmail.com, {ychoi, aberg, tlberg}@cs.stonybrook.edu.*

Manuscript received 31 Mar. 2012; accepted 29 June 2012; published online 23 July 2012.
Recommended for acceptance by P. Felzenszwalb, D. Forsyth, P. Fua, and T.E. Boult.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMISI-2012-03-0237.
Digital Object Identifier no. 10.1109/TPAMI.2012.162.

Fig. 1. Our system automatically generates the following descriptive text for this example image: *"This picture shows one person, one grass, one chair, and one potted plant. The person is near the green grass, and in the chair. The green grass is by the chair, and near the potted plant."*

and then combined with language models in a natural language generation process.

The proposed approach is comprised of two stages. In the first, *content planning*, the sometimes noisy output of computer vision recognition algorithms is smoothed with statistics collected from visually descriptive natural language. Once the content to be used in generation is chosen, the next stage is *surface realization*, finding words to describe the chosen content. Once again text statistics are used to choose surface realization that is more similar to constructions in commonly used language.

Natural language generation constitutes one of the fundamental research problems in natural language processing (NLP) and is core to a wide range of NLP applications such as machine translation, summarization, dialogue systems, and machine-assisted revision. Despite substantial advancement within the last decade, natural language generation still remains an open research problem. Most previous work in NLP on automatically generating captions or descriptions for images is based on retrieval and summarization. For instance, Aker and Gaizauskas [1] rely on GPS metadata to access relevant text documents and Feng and Lapata [18] assume relevant documents are provided. The process of generation then becomes one of combining or summarizing relevant documents, in some cases driven by keywords estimated from the image content [18]. From the computer vision perspective these techniques might be analogous to first recognizing the scene shown in an image and then *retrieving* a sentence based on the scene type. It is very unlikely that a retrieved sentence would be as descriptive of a particular image as the *generated* sentence in Fig. 1.

From the computer vision community, work has considered matching a whole input image to a database of images with captions [15], [33]. The caption of the best matching image can then be used for the input image.

One major potential practical advantage of the approach presented in this paper is that it can generate descriptions without requiring related text or similar images with descriptions. Instead, it builds a caption for an image in a bottom up fashion, starting from what computer vision systems recognize in an image and then constructing a novel caption around those predictions, using text statistics to smooth these (sometimes) noisy vision predictions. However, the downside of such an approach is that descriptions are constructed entirely from scratch. The alternative approaches mentioned above [15], [33] that sample directly from human written text may produce more natural sounding, albeit possibly less directly relevant or descriptive output.

These and other competing desirable traits (e.g., accuracy to content, and naturalness of expression) in natural language description pose challenges for evaluation. In addition to reviewing the generation approach of Kulkarni et al. [26] and presenting a new surface realization strategy using more flexible optimization, this paper presents extensive novel evaluations of the generated sentences of this system and evaluations comparing the generated sentences with those from competing approaches. Evaluations are performed either automatically by measuring similarity of generated sentences to reference examples written by humans, or by directly asking humans which of two sentences is a better description for an image.

Evaluation shows that the proposed system is effective at producing relevant sentences for images. The proposed system also generates descriptions that are measurably more true to the specific image content than previous work, outperforming past approaches in terms of human evaluation.

2 Related Work

There are many areas of related work, including: using word and picture information jointly for labeling images, learning models of categories, attributes, or spatial relationships from data, and methods to compose descriptions for images. We briefly review some of the most relevant work here.

2.1 Integrating Words and Pictures

Early work on connecting words and pictures focused on associating individual words with image regions [2], [3], [11] for tasks such as clustering, auto-annotation or auto-illustration.

Other work has made use of text as a source of noisy labels for predicting the content of an image. This works especially well in constrained recognition scenarios—for recognizing particular classes of objects—such as for labeling faces in news photographs with associated captions [4], [5] or characters in television or movie videos with associated scripts [12], [39]. Other object classes that have been considered include animal images from the web [7], [30], [37] where text from the containing webpage can be utilized for improved image classification.

2.2 Learning Models of Categories or Relationships

Some recent work has attempted to learn models of the world from language or images. Saenko and Darrell [36] learn visual sense models for polysemous words (e.g., "jaguar") from textual descriptions on Wikipedia. Yanai and Barnard [44], [45] directly predict the visualness of concepts (e.g., "pink" is a more visual concept than "thoughtful") from web image search results.

A related body of work on image parsing and object detection, learns the spatial relationships between labeled

parts—either detections or regions. These relationships were used as contextual models to improve labeling accuracy, but the spatial relationships themselves were not considered outputs in their own right [10], [20], [38], [41]. Estimates of spatial relationships between objects form an important part of the output of the computer vision aspect of our approach and are used to drive sentence generation.

2.3 Attributes

There is a great deal of ongoing research on estimating attributes for use in computer vision [6], [13], [19], [27], [29], [43] that maps well to our process of estimating modifiers for objects in images. We use the low level features from Farhadi et al. [13] for modifier estimation. Most past approaches have either constructed the set of attribute terms in an ad hoc manner or taken them from an application appropriate ontology [13], [19], [27], [29], but some approaches have tried to learn the attributes directly from image-text pairs [6], [43]. We take a related approach, by training models for attribute terms that commonly co-occur with our object categories in Flickr descriptions and for which we can produce reliable attribute classifiers. Ultimately, our work combines priors for visually descriptive language with estimates of objects given by detectors and modifiers predicted by attribute classifiers around these object detections.

2.4 Describing Images

There has been some recent work very close in spirit to our own with the overall goal of producing relevant descriptions for images. These methods can be divided into two main types of approaches—methods that make use of *existing text* to describe images, and methods that predict image content and then build descriptions *from scratch*.

The first type of image description approach utilizes *existing text* to describe query images. In some cases the text is already associated with the image. For example, Feng and Lapata [18] construct captions for news images from the enclosing article text using summarization techniques [49]. In other cases, retrieval-based methods are used to gather relevant text for composition. Aker and Gaizauskas [1] use GPS metadata to retrieve relevant text documents for a query image. Farhadi et al. parse images into a meaning representation "triple" describing 1 object, 1 action, and 1 scene [15]. This predicted triple is used to retrieve whole descriptive sentences from a collection written to describe similar images. Other recent methods have also been based on retrieval, including nonparametric methods for composing captions by transferring whole existing captions from a large database of captioned images [33], or by transferring individual relevant phrases and then constructing a novel caption [28].

The second class of image description approach builds descriptive text *from scratch* rather than retrieving existing text. Our work falls into this category—detecting multiple objects, modifiers, and their spatial relationships, and generating novel sentences to fit these constituent parts, as opposed to retrieving sentences whole. Other methods that have taken the generate from scratch method for description include Yao et al. [47] who look at the problem of generating text with a comprehensive system built on various hierarchical knowledge ontologies and using a human in the loop for hierarchical image parsing (except in specialized circumstances). In contrast, our work automatically mines knowledge about textual representation, and parses images fully automatically—without a human operator—and with a much simpler approach overall. Li et al. [31] take a similar approach to ours, but focus on introducing creativity in sentence construction. Finally, Yang et al. [46] also compose descriptions in a bottom up fashion, detecting objects and scenes, and then using text statistics to "hallucinate" verbs for objects. Descriptions are then composed in an HMM framework.

2.5 Describing Videos

In addition to these generation efforts in images, there has also been work related to linking humans and their actions in video [21], [23], [24]. Applications include methods to construct plots for sports activities [21], or to recognize and retrieve video depicting activities using associated text [22], [23]. People use high-level structure—goal directed partonomic hierarchies—to describe human actions [48]. We do not pursue this angle further here, but note that describing humans and their activities is an important aspect of visual description.

3 Method Overview

An overview of our system is presented in Fig. 2. For an input image:

1. Detectors are used to detect things (e.g., bird, bus, car, person, etc.) and stuff (e.g., grass, trees, water, road, etc.). We will refer to these as *things* and *stuff*, or collectively as objects.
2. Each candidate object (either thing or stuff) region is processed by a set of attribute classifiers.
3. Each pair of candidate regions is processed by prepositional relationship functions.
4. A CRF is constructed that incorporates the unary image potentials computed by 1-3, with higher order text-based potentials computed from large text corpora.
5. A labeling of the graph is predicted.
6. Sentences are generated.

The rest of the paper first describes the content planning stage of our description generation process. A conditional random field (CRF) is used to predict a labeling for an input image (Section 4), then the image-based potentials (Section 5.1) and higher order text-based potentials (Section 5.2). Next, we describe the surface realization stage of our generation process. Various methods for surface realization (forming natural language descriptions) are covered in Section 6. Finally, we discuss extensive evaluation results in Section 7, and conclude in Section 8.

4 Content Planning

We use a conditional random field to predict the best labeling for an image (e.g., Fig. 3)—i.e., for the content planning step of our description generation process. Nodes

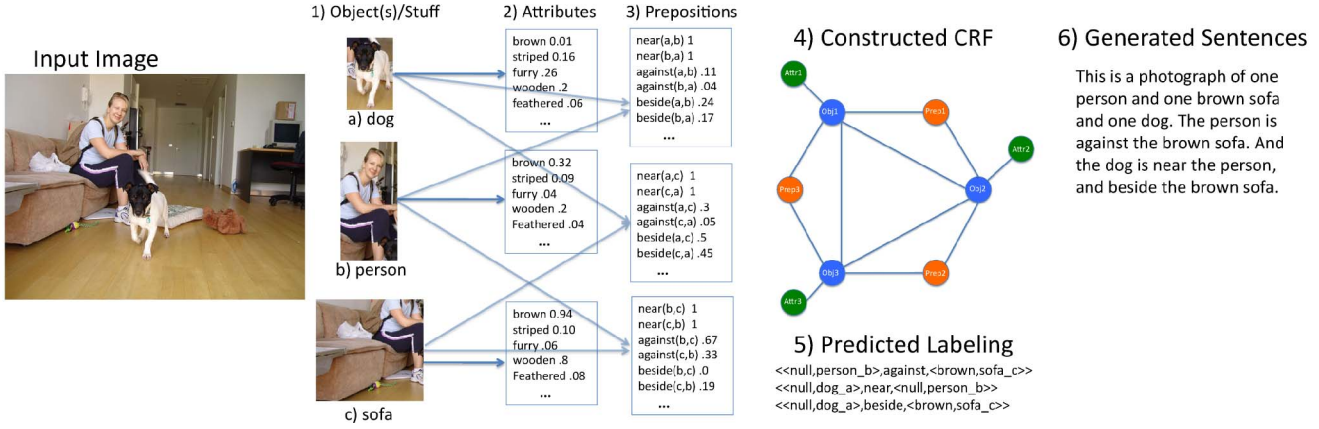


Fig. 2. System flow for an example image: (1) Object and stuff detectors find candidate objects, (2) each candidate region is processed by a set of attribute classifiers, (3) each pair of candidate regions is processed by prepositional relationship functions, (4) a CRF is constructed that incorporates the unary image potentials computed by 1-3 and higher order text-based potentials computed from large document corpora, (5) a labeling of the graph is predicted, and (6) sentences are generated.

of the CRF correspond to several kinds of image content: 1) objects—things or stuff, 2) attributes which modify the appearance of a object, and 3) prepositions which refer to spatial relationships between object-object pairs (including things and stuff).

For a query image, we run a large set of (thing) object detectors across the image and collect the set of high scoring detections. We merge detections that are highly overlapping (greater than 0.3 intersection/union) into groups and create an object node for each group. In this way we avoid predicting two different object labels for the same region of an image which can occur when two different object detectors fire on the same object. We also run our stuff detectors across the image and create nodes for stuff categories with high scoring detections. Note that this means that the number of nodes in a graph constructed for an image depends on the number of object and stuff detections that fired in that image (something we have to correct for during parameter learning). For each object and stuff node we classify the appearance using a set of trained attribute classifiers and create a modifier node. Finally, we create a preposition node for each pair of object and stuff detections. This node predicts the probability of a set of prepositional relationships based on the spatial relationship between two object regions.

The domain (of possible labels) for each node is node dependent. For an object (or stuff) node the domain corresponds to the set of object (or stuff) detectors that fired at that region in the image. For the attribute nodes the domain corresponds to a set of appearance attributes that can modify the visual characteristics of an object (e.g., green or furry). For the preposition nodes the domain corresponds to a set of prepositional relations (e.g., on, under, near) that can occur between two objects.

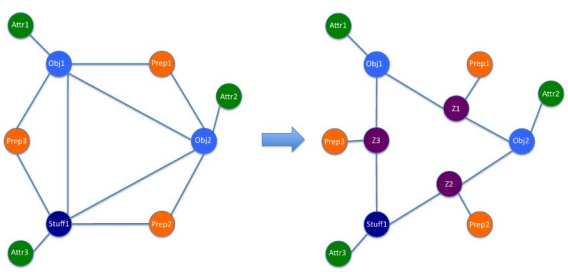


Fig. 3. CRF for an example image with two object detections and one stuff detection. Left shows original CRF with trinary potentials. Right shows CRF reduced to pairwise potentials by introducing z variables whose domains are all possible triples of the original 3-clique.

The energy function for an image labeling (assignment of each node to one of the values of its domain) is described by

$$E(L;I,T) = \sum_{i \in objs} F_i + \frac{2}{N-1} \sum_{ij \in objPairs} G_{ij}, \quad (1)$$

where N is the number of objects and 2/(N-1) normalizes—for a variable number of node graphs—the contribution from object pair terms so that they contribute equally with the single object terms to the energy function. Here,

$$F_i = \alpha_0 \beta_0 \psi(obj_i; objDet) + \alpha_0 \beta_1 \psi(attr_i; attrCl) \quad (2)$$

$$+ \alpha_1 \gamma_0 \psi(attr_i, obj_i; textPr), \quad (3)$$

$$G_{ij} = \alpha_0 \beta_2 \psi(prep_{ij}; prepFuns) \quad (4)$$

$$+ \alpha_1 \gamma_1 \psi(obj_i, prep_{ij}, obj_j; textPr). \quad (5)$$

The three unary potential functions are computed from image-based models and refer to: the detector scores for object(s) proposed by our trained object and stuff detectors ($\psi(obj_i; objDets)$), the attribute classification scores for an object (or stuff) region as predicted by our trained attribute classifiers ($\psi(attr_i; attrCl)$), and the prepositional relationship score computed between pairs of detection regions ($\psi(prep_{ij}; prepFuns)$). Descriptions of the particular detectors, classifiers, and functions used are provided in Section 5.1.

The pairwise ($\psi(mod_i, obj_i; textPr)$) and trinary ($\psi(obj_i, prep_{ij}, obj_j; textPr)$) potential functions model the pairwise scores between object and attribute node labels, and the trinary scores for an object-preposition-object triple labeling, respectively. These higher order potentials could be learned from a large pool of labeled image data. However, for a reasonable number of objects and prepositions the amount of

labeled image data that would be required is daunting. Instead we learn these relationships from large text collections. By observing in text how people describe objects, attributes, and prepositions between objects we can well model the relationships between node labels. Descriptions of our text-based potentials are provided in Section 5.2.

4.1 Converting to Pairwise Potentials

Since preposition nodes describe the relationship between a preposition label and two object labels, they are most naturally modeled through trinary potential functions:

$$\psi(obj_i, prep_{ij}, obj_j; textPr). \tag{6}$$

However, most CRF inference code accepts only unary and pairwise potentials. Therefore, we convert this trinary potential into a set of unary and pairwise potentials through the introduction of an additional z node for each 3-clique of obj-prep-obj nodes (see Fig. 3). Each z node connecting two object nodes has domain O1xPxO2, where O1 is the domain of object node1, P is our set of prepositional relations, and O2 is the domain of object node2. In this way the trinary potential is converted to a unary potential on z:

$$\psi(z_{ij}; textPr). \tag{7}$$

Plus 3 pairwise potentials, one for each of object node1, preposition node, and object node2 that enforce that the labels selected for each node are the same as the label selected for Z:

$$\psi(z_{ij}, obj_i) = \begin{cases} 0, & \text{if } Z_{ij}(1) = O_i \\ inf, & \text{o.w.}, \end{cases} \tag{8}$$

$$\psi(z_{ij}, prep_{ij}) = \begin{cases} 0, & \text{if } Z_{ij}(2) = P_{ij}, \\ inf, & \text{o.w.}, \end{cases} \tag{9}$$

$$\psi(z_{ij}, obj_j) = \begin{cases} 0, & \text{if } Z_{ij}(3) = O_j, \\ inf, & \text{o.w.} \end{cases} \tag{10}$$

4.2 CRF Learning

We take a factored learning approach to estimate the parameters of our CRF from 100 hand-labeled images. In our energy function ((1)-(5)), the α parameters represent the tradeoff between image and text-based potentials. The β parameters represent the weighting between image-based potentials. And, the γ parameters represent the weighting between text-based potentials. In the first stage of learning we estimate the image parameters β while ignoring the text-based terms (by setting α_1 to 0). To learn image potential weights we fix β_0 to 1 and use grid search to find optimal values for β_1 and β_2. Next, we fix the β parameters to their estimated value and learn the remaining parameters—the tradeoff between image and text-based potentials (α parameters) and the weights for the text-based potentials (γ parameters). Here we set α_0 and γ_0 to 1 and use grid search over values of α_1 and γ_1 to find appropriate values.

It is important to carefully score output labelings fairly for graphs with variable numbers of nodes (dependent on the number of object detections for an image). We use a scoring function that is graph size independent:

$$\frac{obj_{t-f}}{N} + \frac{(mod, obj)_{t-f}}{N} + \frac{2}{N-1}\frac{(obj, prep, obj)_{t-f}}{N},$$

measuring the score of a predicted labeling as: 1) the number of true obj labels minus the number of false obj labels normalized by the number of objects, plus 2) the number of true mod-obj label pairs minus the number of false mod-obj pairs, plus 3) the number of true obj-prep-obj triples minus the number of false obj-prep-obj triples normalized by the number of nodes and the number of pairs of objects (N choose 2).

4.3 CRF Inference

To predict the best labeling for an input image graph (both at test time or during parameter training) we utilize the sequential tree reweighted message passing (TRW-S) algorithm introduced by Kolmogorov [25], which improves upon the original TRW algorithm from Wainwright et al. [42]. These algorithms are inspired by the problem of maximizing a lower bound on the energy. TRW-S modifies the TRW algorithm so that the value of the bound is guaranteed not to decrease. For our image graphs, the CRF constructed is relatively small (on the order of 10s of nodes). Thus, the inference process is quite fast, taking on average less than a second to run per image.

5 Potential Functions

In this section, we present our image-based and descriptive language-based potential functions. At a high level the image potentials come from hand designed detection strategies optimized on external training sets (we use some off-the-shelf detectors and train others in order to cover more object categories). In contrast the text potentials are based on text statistics collected automatically from various corpora.

5.1 Image-Based Potentials

$\psi(obj_i; objDet)$—Object and Stuff Potential

Object detectors. We use an object detection system based on Felzenszwalb et al.'s mixtures of multiscale deformable part models [17] to detect "thing objects." We use the provided detectors for the 20 PASCAL 2010 object categories and train four additional non-PASCAL object categories for *flower, laptop, tiger*, and *window*. For the non-PASCAL categories, we train new object detectors using images and bounding box data from Imagenet [9]. The output score of the detectors are used as potentials.

Stuff detectors. Classifiers are trained to detect regions corresponding to non-part-based object categories. We train linear SVMs on the low-level region features of [13] to recognize: sky, road, building, tree, water, and grass stuff categories. SVM outputs are mapped to probabilities. Training images and bounding boxes are taken from ImageNet and evaluated at test time on a coarsely sampled grid of overlapping square regions over whole images. Pixels in any region with a classification probability above a fixed threshold are treated as detections, and the max probability for a region is used as the potential value.

$\psi(attr_i; attrCl)$—Attribute Potential

Attribute classifiers. We train visual attribute classifiers that are relevant for our object (and stuff) categories.

Therefore, we mine our large text corpus of Flickr descriptions (described in Section 5.2) to find attribute terms commonly used with each object (and stuff) category, removing obviously nonvisual terms. The resulting list consists of 21 visual attribute terms describing color (e.g., blue, gray), texture (e.g., striped, furry), material (e.g., wooden, feathered), general appearance (e.g., rusty, dirty, shiny), and shape (e.g., rectangular) characteristics. Training images for the attribute classifiers come from Flickr, Google, the attribute dataset provided by Farhadi et al. [14], and ImageNet [9]. An RBF kernel SVM is used to learn a classifier for each visual attribute term (up to 150 positive peer class with all other training examples as negatives). The outputs of the classifiers are used as potential values.

$\psi(prep_{ij}; prepFuns)$—*Preposition Potential*

Preposition functions. We design simple prepositional functions that evaluate the spatial relationships between pairs of regions in an image and provide a score for each of 16 preposition terms (e.g., above, under, against, beneath, in, on, etc). For example, the score for $above(a, b)$ is computed as the percentage of $region_a$ that lies in the image rectangle above the bounding box around $region_b$. The potential for $near(a, b)$ is computed as the minimum distance between $region_a$ and $region_b$ divided by the diagonal size of a bounding box around $region_a$. Similar functions are used for the other preposition terms. We include synonymous prepositions to encourage variation in sentence generation but sets of synonymous prepositions share the same potential. Note for each preposition we compute both prep(a, b) and prep(b, a) as either labeling order can be predicted in the output result.

5.2 Text-Based Potentials

We use two potential functions calculated from large text corpora. The first is a pairwise potential on attribute-object label pairs $\psi(attr_i, obj_i; textPr)$ and the second is a trinary potential on object-preposition-object triples $\psi(obj_i, prep_{ij}, obj_j; textPr)$. These potentials represent the probability of various attributes for each object and the probabilities of particular prepositional relationships between object pairs.

Parsing potentials. To generate the attribute-object potential $\psi_p(attr_i, obj_i; textPr)$ we collect a large set of Flickr image descriptions (similar in nature to captions, but less regulated). For each object (or stuff) category we collect up to 50,000 image descriptions (fewer if less than 50,000 exist) by querying the Flickr API[1] with each object category term. Each sentence from this descriptions set is parsed by the Stanford dependency parser [8] to generate the parse tree and dependency list for the sentence. We then collect statistics about the occurence of each attribute and object pair using the adjectival modifier dependency $amod(attribute, object)$. Counts for synonyms of object and attribute terms are merged together.

For generating the object-preposition-object potential $\psi_p(obj_i, prep_{ij}, obj_i; textPr)$ we again collect a large set of Flickr image descriptions (about 1.4 million total), in this case using queries based on pairs of object terms. All descriptive sentences containing an occurrence of at least two of our object (or stuff) categories plus a prepositional term (about 140k) are parsed using the Stanford dependency parser. We then collect statistics for the occurrence of each prepositional dependency between object categories. For a prepositional dependency occurrence, object1 is automatically picked as either the subject or object part of the prepositional dependency based on the voice (active or passive) of the sentence, while object2 is selected as the other. Again, counts for an object and its synonyms are merged together.

Google potentials. Though we parse thousands of descriptions, the counts for some objects can still be sparse. Therefore, we collect additional Google Search-based potentials: $\psi_g(attr_i, obj_i; textPr)$ and $\psi_g(obj_i, prep_{ij}, obj_j; textPr)$. These potentials are computed as the number of search results approximated by Google for string match queries on each of our attribute-object pairs (e.g., "brown dog") and preposition-object-preposition triples (e.g., "dog on grass").

Smoothed potentials. Our final potentials are computed as a smoothed combination of our parsing-based potentials with the Google potentials: $\alpha\psi_p + (1 - \alpha)\psi_g$.

6 Surface Realization

The output of our CRF is a predicted labeling of the image. This forms the content we want to encode in our surface realization step, generation of the final natural language descriptions. This labeling encodes three kinds of information: objects present in the image (nouns), visual attributes of those objects (modifiers), and spatial relationships between objects (prepositions). Therefore, it is natural to extract this meaning into a triple (or set of triples), e.g.,

$$\ll white, cloud>, in, <blue, sky\gg.$$

Based on this triple, we want to generate a complete sentence such as *"There is a white cloud in the blue sky."* For simplicity, we make the following restrictions on generation: First, the set of words in the meaning representation is fixed and generation must make use of all given content words. Second, generation may insert only gluing words (i.e., *function words* such as "there," "is," "the," etc.) to complete the sentences—generation should not introduce new content. These restrictions could be lifted in future work.

We present three generation techniques for producing a surface realization. The first is based on decoding using n-gram language models, Section 6.1. Next, we introduce a more flexible ILP-based optimization that can handle a wider range of constraints on generation, Section 6.2. Finally, we present a template-based approach, Section 6.3.

6.1 Decoding Using Language Models

An N-gram language model is a conditional probability distribution $P(x_i|x_{i-N+1}, \ldots, x_{i-1})$ of N-word sequences $(x_{i-N+1}, \ldots, x_i)$ such that the prediction of the next word depends only on the previous $N - 1$ words. That is, with $N - 1$th order Markov assumption, $P(x_i|x_1, \ldots, x_{i-1}) = P(x_i|x_{i-N+1}, \ldots, x_{i-1})$. Language models are shown to be simple but effective for improving machine translation and automatic grammar corrections.

In this work, we make use of language models to predict gluing words (i.e., *function words*) that put together words in the meaning representation. As a simple example, suppose we want to determine whether to insert a function word x between a pair of words α and β in the meaning representation. Then, we need to compare the length-normalized probability $\hat{p}(\alpha x \beta)$ with $\hat{p}(\alpha\beta)$, where $\hat{p}$

1. http://www.flickr.com/services/api/.

takes the n'th root of the probability p for n-word sequences, and $p(\alpha x \beta) = p(\alpha)p(x|\alpha)p(\beta|x)$ using bigram (2-gram) language models. If considering more than two function words between α and β, dynamic programming can be used to find the most optimal sequence of function words efficiently. Because the ordering of words in each triple of the meaning representation coincides with the typical ordering of words in English, we retain the original ordering for simplicity. Note that this approach composes a separate sentence for each triple, independently from all other triples. To train the language model n-grams we crawl Wikipedia pages, describing objects our system can recognize.

6.2 ILP Decoding

In our last surface realization method we allow for some creativity in the generation process, which produces more human-like descriptions than our two previous methods (Sections 6.1 and 6.3). This approach makes use of language models and appropriate linguistic constraints to generate sentences for images in an ILP framework. ILP provides a general computational framework to efficiently solve challenging optimization problems, while also providing a nice way to incorporate long range constraints on sentence content and syntax that would otherwise be difficult to enforce using other approaches.

We formulate sentence generation as an optimization problem over the choice of word at each position in a sentence. The objective function will encode the negative log likelihood of the sentence under the language model (smaller is better). The tuples output by the vision system for an image will form constraints for the optimization—generated sentences should discuss the detected content of the image. In addition, looking at generation as an optimization problem makes it possible to easily add additional constraints on the structure of the generated sentences. Finally, in this paper, we formulate the optimization as an integer linear program, dictating the form of the objective function and constraints.

What we hope to gain by using a global optimization framework like ILP is the ability to add long-range constraints to the generation process that can make the resulting sentences more likely to be grammatical and pleasing. This can be difficult when using a local search strategy for generating sentences according to a language model, as enforcing longer range constraints may not be possible or may be very cumbersome in such settings (e.g., some dynamic programming and randomized search approaches). Results will show that a small number of longer range constraints can be added to the n-gram language model to produce pleasing and grammatical descriptive sentences.

We will write the objective function and constraints in terms of indicator variables for sequences of n words ending at position i in the generated sentence:

$$x_{iw^0 \dots w^{n-1}} = \begin{cases} 1, & \text{if } X_{i-j} = w^j \text{ for } j = 0 \dots n-1, \\ 0, & \text{Otherwise,} \end{cases}$$

where X_i is the word in the ith position in the generated sentence and each $w^j \in V$ is a word in the vocabulary V.

We will then optimize an objective function written in terms of the indicator variables that expresses how well the generated sentence agrees with a language model:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \sum_{n=1\dots4} \sum_{iw^0 \dots w^{n-1}} C_{w^0 \dots w^{n-1}} x_{iw^0 \dots w^{n-1}} \\ \text{subject to} \quad & f_i(x_*) \le b_i, \ i = 1 \dots n_{\text{le}} \\ & g_j(x_*) = c_i, \ j = 1 \dots n_{\text{eq}} \\ & x_* \in \{0, 1\}, \end{aligned}$$

where x_* is shorthand for all of the indicator variables. The costs,

$$C_{w^0 \dots w^{n-1}} = -\log P(w^0 | w^{n-1} \dots w^1),$$

are the negative log probability of word w^0 following the sequence of words $w^{n-1}, w^{n-2}, \dots, w^1$. Note that the objective function expresses the negative log of the product of unigram, bigram, trigram, and 4-gram language models.

Next, we catalog the various constraints. These are written using $f_i(x_*) \le b_i$ and $g_j(x_*) = c_j$ in the optimization problem above, but below are left in simpler forms for brevity.

1. **Constraints for variable consistency.** Each place holder should get exactly one word assigned $\forall_i \sum_{w \in V} x_{iw} = 1$.

   Indicator variables should be consistent with each other:

   $$\forall_i \sum_{w^{n-1} \in V} x_{iw^0 \dots w^{n-1}} = X_{iw^0 \dots w^{n-2}},$$
   $$\forall_i \sum_{w^0 \in V} x_{iw^0 \dots w^{n-1}} = X_{(i-1)w^1 \dots w^{n-1}}.$$

2. **Constraints for image semantics.** Each content word (word specified in the vision output tuple) should occur in the final sentence:

   $$\forall_{w \in \text{Content Words}} \sum_{i=1}^{L} x_{iw} = 1.$$

   Adjectives. Sentences should not contain adjective-adjective bigrams, so let A be the set of adjective words:

   $$\forall_{m_{1,2} \in A} \forall_{i:2 \le i \le L} (x_{im_2 m_1} = 0).$$

   Adjectives should occur within a short distance (D set by cross validation) of the object being modified, so for any adjective object pair m, o:

   $$\forall_i \forall_{j:|i-j|>D} (x_{im} + x_{jo} \le 1).$$

   We avoid Object2-Verb-Adjective2 patterns because it confuses the association of subject and verb, so for any Adjective2 Object2 pair, m_2, o_2:

   $$\forall_{i,3 \le i \le L} \forall_{v \in \text{Verbs}} (x_{im_2 v o_2} = 0).$$

   Prepositions. Prepositions should not occur to the right of, following, Object2, so for a pair of objects o_1, o_2:

   $$\forall_{p \in \text{Prep}} \forall_i \forall_{j>i} (x_{io_2} + x_{jp} <= 1).$$

   Object1 must not be in between the preposition and Object2. However, we do allow Object2 between the preposition and Object1. So, let $o_1$be Object1 and o_2 be Object2 and

Fig. 4. Results of sentence generation using our method with template-based sentence generation. These are "*good*" results as judged by human annotators.

$$\forall_{p\in \text{Prep}}\forall_i\forall_{j>i}\forall_{k>j}(x_{ip} + x_{jo_1} + x_{ko_2} <= 2).$$

We note that if the same object type appears twice in a triple, then it is duplicated in the vocabulary.

3. **Constraints for linguistic rules.** No function word (vocabulary not in the vision output) should be repeated more than twice, $\forall_{w\in \text{ Funtion Words}} \sum_i x_{iw} \leq 2$. *Verbs.* Sentences should contain at least one verb, $\sum_{i=1}^{L} \sum_{w\in \text{ Verbs}} x_{iw} \geq 1$, but the first and last (Lth) words cannot be verbs, $\forall_{w\in \text{Verbs}}(x_{1w} = x_{Lw} = 0)$.

6.3 Templates with Linguistic Constraints

Decoding based on language models is a statistically principled approach; however, two main limitations are: 1) It is hard to enforce grammatically correct sentences using language models alone; 2) it is ignorant of discourse structure (coherency among sentences) as each sentence is generated independently. We address these limitations by constructing templates with linguistically motivated constraints. This approach is based on the assumption that there are a handful of salient syntactic patterns in descriptive language.

For example, a simple template might read: "`This is a photograph of <count> <object(s)>.`" or "`Here we see <count> <object(s)>.`" To encode spatial (prepositional) relationships we use templates such as "`The <nth> <adjective> <object1> is <prep> <nth> <adjective> <object2>.`" In this manner we can produce sentences like "This is a photograph of one sky.", "Here we see one person and one train." or "The first black person is by the blue sky." Additional example results can be seen in Figs. 4 and 6.

7 Experimental Evaluation

For evaluation, we use the UIUC PASCAL sentence dataset[2] [35], which contains up to five human-generated sentences that describe 1,000 images. From this set we evaluate results on 847 images (153 have been used to set CRF parameters and detection thresholds).

2. http://vision.cs.uiuc.edu/pascal-sentences/.

Two forms of quantitative evaluation are performed, automatic evaluation using standard methods for evaluating generated sentences (Section 7.1) and human forced evaluations to directly compare the results between our method and several previous methods (Section 7.2). In each case we also quantitatively evaluate and compare to two previous approaches for image description generation used on the same dataset [35]. The first comparison method is the bottom up HMM approach from Yang et al. [46], which detects objects and scenes, and then hallucinates plausible verbs for generation (using text statistics). The second comparison method is a retrieval-based approach from Farhadi et al. [15]. This method detects objects, scenes, and actions and then retrieves descriptive sentences from similar images through the use of a meaning space. Finally, we also discuss our results qualitatively (Section 7.1).

7.1 Automatic Evaluation

We evaluate our results using two standard methods for automatic evaluation of machine generated sentences, BLEU [34] and ROUGE [32] scores. These two measures are commonly used in the machine translation community to evaluate the goodness of machine translated results against ground truth human translations.

The BLEU score measures the modified n-gram precision of machine generated sentences with respect to human generated sentences. Because our task can be viewed as machine translation from images to text, BLEU may seem like a reasonable choice at first glance. Upon a close look however, one can see that there is inherently larger variability in generating sentences from images than translating a sentence from one language to another. For instance, from the image shown in Fig. 1, our system correctly recognizes objects such as "chair," "green grass," and "potted plant," none of which is mentioned in the human-generated description available in the UIUC PASCAL sentence dataset. As a result, BLEU will inevitably penalize many correctly generated sentences, which in turn can cause a low correlation with human judgment of quality. In addition, because BLEU measures precision, it

Fig. 5. Comparison of our three surface realization generation methods.

TABLE 1
Automatic Evaluation: BLEU Score Measured at 1 for Generated Descriptions versus the Set of Descriptions Produced by Human Annotators

Method	w/o	w/ synonym
Human	0.60	0.63
Language model-based generation	0.22	0.26
Template-based generation	0.18	0.21
ILP-based generation	0.24	0.29
[15]	0.23	0.25
[46]	0.33	0.37

TABLE 2
Automatic Evaluation: ROUGE Score for Generated Descriptions versus the Set of Descriptions Produced by Human Annotators

Method	w/o	w/ synonym
Human	0.55	0.58
Language model-based generation	0.21	0.27
Template-based generation	0.25	0.32
ILP-based generation	0.23	0.29
[15]	0.22	0.23
[46]	0.24	0.28

inherently penalizes the long descriptions produced by our template-based generation method (even though as will be seen in Section 7.2 humans judge this method to produce the best results). Nevertheless, we report BLEU score as a standard evaluation method and to provide a quantification of its shortcomings.

BLEU score results are shown in Table 1. In our case, we measure the BLEU score of generated descriptions against the *set* of five human written descriptions provided with each image [35]. The first column shows BLEU score when measured with exact match for each word, and the second column shows BLEU when we give full credit for synonymous words. To give a sense of upper bound, we also compute the BLEU score of human-generated sentences; we compute the average over all images of the BLEU score for one human-generated sentence with respect to the others for that image. Finally, we also compute BLEU score for two other related image description methods, Farhadi et al. [15] and Yang et al. [46]. Notice that the BLEU score for these methods is much better than for our method, though human judgment seems to indicate the opposite to be true (Section 7.2). This is perhaps because our method tends to produce much longer descriptions than the previous methods and BLEU inherently penalizes long descriptions; ours produces on average 24 words for the template approach, 18 for language model-based generation, and 16 for ILP, while Farhadi et al. [15] produce, on average, nine words and Yang et al. [46] produces seven words on average.

In addition, we also evaluate using ROUGE score. ROUGE score is quite similar to BLEU, except that it measures the n-gram recall (rather than precision) of machine generated sentences with respect to human generated sentences. The results are shown in Table 2. Here we see that ROUGE scores for our methods are similar to or slightly better than those for the previous methods [15], [46]. This

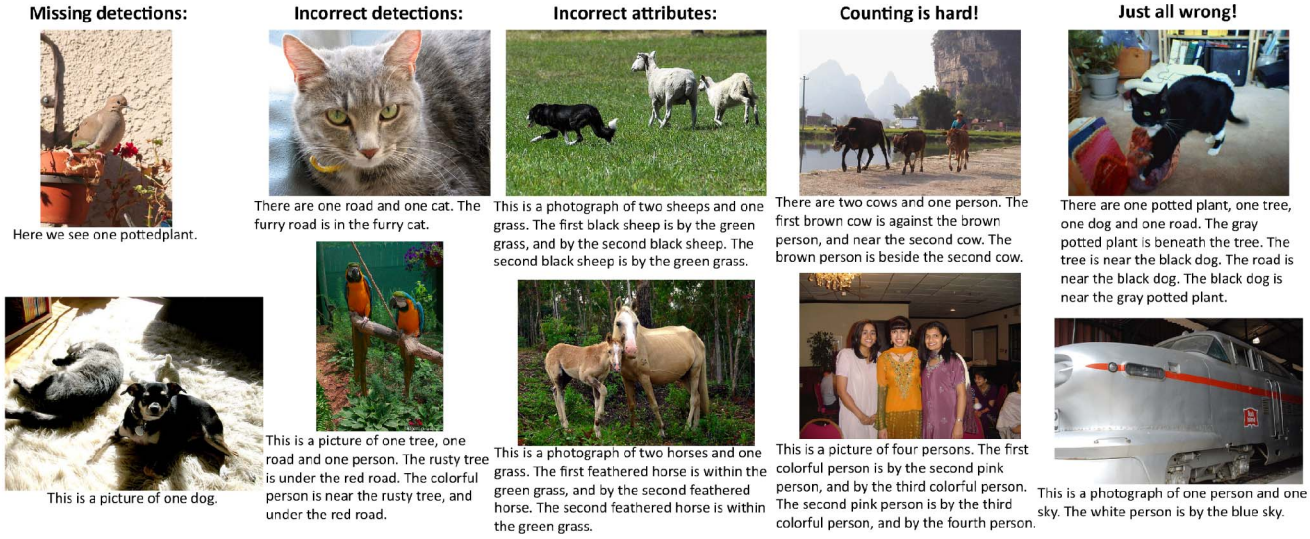


Fig. 6. Results of description generation using our template-based surface realization method. These are "*bad*" results as judged by human annotators.

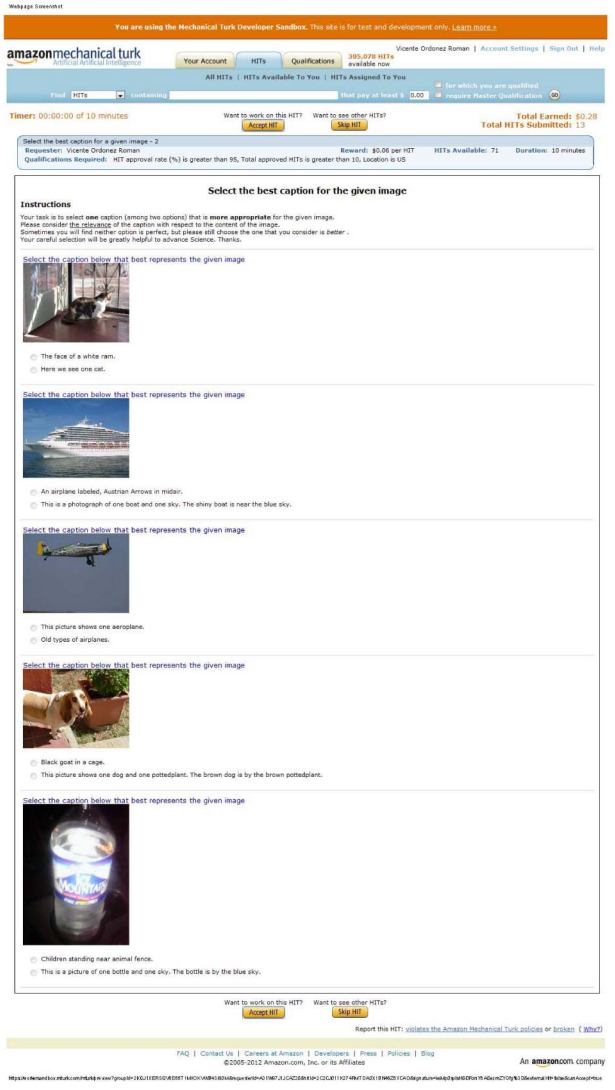


Fig. 7. An example forced choice experiment on Mechanical Turk to evaluate two image description methods (in this case our CRF and template-based generation versus the method from Farhadi et al. [15]). A user is presented with a set of images and two descriptions for each image. They must select the description that is most appropriate for the image, considering the relevance of the caption with respect to the image content. Care is taken to randomize order of method presentation to avoid user selection bias.

correlates better with the observations made in our forced choice human judgment experiments (Section 7.2).

7.2 Forced Choice Evaluation

In addition to the above automatic methods for evaluation, we also perform human subject-based evaluations. These evaluations take the form of forced choice experiments to directly compare results between methods on the Pascal Sentence Dataset [35]. An example forced choice experiment can be seen in Fig. 7. In these experiments users are presented with a set of images (five in our experiments). For each image, they are shown two descriptions, one from each of two methods you want to compare—in Fig. 7 we show an experiment using results from the method of Farhadi et al. [15] with our method using template-based generation. Users are instructed to select the description that is most appropriate for each image, considering the relevance of the caption with respect to the image content. Care is also taken to randomize order of method presentation (top versus bottom) to avoid user bias in selection.

TABLE 3
Pairwise Method Forced Choice Evaluation

Comparison	Pref1	Pref2
Template vs LM	61.9	38.1
Template vs ILP	86.4	13.6
[46] vs [15]	72.2	27.8
Template vs [15]	80.0	20.0
Template vs [46]	61.9	38.1

For each pair of methods we evaluate human preferences on produced results. Column Pref1 denotes the percentage of times the left method was selected, while Column Pref2 denotes the percentage of times the right method was selected.

Results of our forced choice experiments are shown in Table 3. We first evaluate our two language model-based methods for generation (LM and ILP) against our template generation method and find that the template model performs best in human subjective evaluations. Note that this evaluation compares the results from full generation systems where many factors can influence an observer's perception, including content, grammar, sentence length, etc. It seems that for both language model and ILP-based decoding, some lack of fluency in the produced results really influences human evaluation. So, while these methods perform reasonably well under automatic measures of performance, they perform poorly under human judgments.

Next, we evaluate our best human judged generation method—template generation—against the results from two other approaches to image description [15], [46]. Because our method sometimes does not produce a description (when no object is detected), for those descriptions we assume that the competing method always wins the forced choice test. Results show that both our description method and Yang et al. [46] are preferred in human judgment experiments over Farhadi et al. [15], preferred 80 and 72.2 percent, respectively. In comparisons between our method and that of Yang et al. [46], we see a slightly smaller difference, but are still preferred 61.9 percent of the time over this approach.

7.3 Qualitative Evaluation

The majority of our generated sentences look quite good. Some example results are shown in Fig. 4 (for the template-based generation scheme) that represent some "good" generation results on PASCAL images. However, in fact most of our results look quite good. Even "bad" results almost always look reasonable and are relevant to the image content (Fig. 6). Only in a small majority of the images are the generated descriptions completely unrelated to the image content (Fig. 6, two right-most images). In cases where the generated sentence is not quite perfect this is usually due to one of three problems: a failed object detection that misses an object, a detection that proposes the wrong object category, or an incorrect attribute prediction. However, because of our use of powerful vision systems (state of the art detectors and attribute methodologies) the results produced are often astonishingly good.

In general, we find that the object detectors are more reliable than the predicted spatial (prepositional)

relationships, a task that is quite difficult to accomplish with entirely 2D reasoning about inherently 3D relationships.

We can also compare the outputs of our three generation methods, language model decoding, ILP decoding, and our template method. Fig. 5 shows results from each of these methods. Though the template method is simplest and produces somewhat robotic sounding sentences, it does generate consistent sentences by construction. Both the simple decoding method and ILP-based decoding sometimes produce grammatically incorrect sentences (e.g., "the cow and by the tree"). Human judgments between the template and decoding methods reflect this problem. However, template methods will likely be insufficient to produce more complex image descriptions beyond the simple cases we currently consider.

8 Conclusions and Future Work

We have demonstrated a surprisingly effective, fully automatic, system that generates natural language descriptions for images. The system works well and can produce results much more specific to the image content than previous automated methods. Human-forced choice experiments demonstrate the quality of the generated sentences over previous approaches. One key to the success of our system was automatically mining and parsing large text collections to obtain statistical models for visually descriptive language. The other is taking advantage of state-of-the-art vision systems and combining all of these in a CRF to produce input for language generation methods.

Future work includes methods to produce more natural sounding image descriptions, incorporating ideas of content importance in the description process, and extending the method to handle more general image content (beyond the 20 Pascal object categories). We would also like to expand our approach to include actions and scenes, as well as applying related techniques to describe the content of videos.

Acknowledgments

This work was supported in part by US National Science Foundation Faculty Early Career Development (CAREER) Award #1054133 for T.L. Berg and by the Stony Brook University Vice President of Research for Y. Choi and A.C. Berg.

References

[1] A. Aker and R. Gaizauskas, "Generating Image Descriptions Using Dependency Relational Patterns," *Proc. 28th Ann. Meeting Assoc. for Computational Linguistics,* pp. 1250-1258, 2010.

[2] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan, "Matching Words and Pictures," *J. Machine Learning Research,* vol. 3, pp. 1107-1135, 2003.

[3] K. Barnard, P. Duyguly, and D. Forsyth, "Clustering Art," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* June 2001.

[4] T.L. Berg, A.C. Berg, J. Edwards, and D.A. Forsyth, "Who's in the Picture?" *Proc. Neural Information Processing Systems Conf.,* 2004.

[5] T.L. Berg, A.C. Berg, J. Edwards, M. Maire, R. White, E. Learned-Miller, Y.-W. Teh, and D.A. Forsyth, "Names and Faces," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

[6] T.L. Berg, A.C. Berg, and J. Shih, "Automatic Attribute Discovery and Characterization from Noisy Web Data," *Proc. European Conf. Computer Vision,* 2010.

[7] T.L. Berg and D.A. Forsyth, "Animals on the Web," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[8] M.-C. de Marnee and C.D. Manning, *Stanford Typed Dependencies Manual,* 2009.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[10] C. Desai, D. Ramanan, and C. Fowlkes, "Discriminative Models for Multi-Class Object Layout," *Proc. 12th IEEE Int'l Conf. Computer Vision,* 2009.

[11] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth, "Object Recognition as Machine Translation," *Proc. European Conf. Computer Vision,* 2002.

[12] M. Everingham, J. Sivic, and A. Zisserman, "Hello! My Name Is.. Buffy—Automatic Naming of Characters in TV Video," *Proc. British Machine Vision Conf.,* 2006.

[13] A. Farhadi, I. Endres, D. Hoiem, and D.A. Forsyth, "Describing Objects by Their Attributes," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[14] A. Farhadi, I. Endres, D. Hoiem, and D.A. Forsyth, "Describing Objects by Their Attributes," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[15] A. Farhadi, M. Hejrati, A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D.A. Forsyth, "Every Picture Tells a Story: Generating Sentences for Images," *Proc. European Conf. Computer Vision,* 2010.

[16] L. Fei-Fei, C. Koch, A. Iyer, and P. Perona, "What Do We See When We Glance at a Scene," *J. Vision,* vol. 4, no. 8, 2004.

[17] P.F. Felzenszwalb, R.B. Girshick, and D. McAllester, "Discriminatively Trained Deformable Part Models, Release 4," http://people.cs.uchicago.edu/pff/latent-release4/, 2012.

[18] Y. Feng and M. Lapata, "How Many Words Is a Picture Worth? Automatic Caption Generation for News Images," *Proc. Assoc. for Computational Linguistics,* pp. 1239-1249, 2010.

[19] V. Ferrari and A. Zisserman, "Learning Visual Attributes," *Proc. Neural Information Processing Systems Conf.,* 2007.

[20] A. Gupta and L.S. Davis, "Beyond Nouns: Exploiting Prepositions and Comparative Adjectives for Learning Visual Classifiers," *Proc. European Conf. Computer Vision,* 2008.

[21] A. Gupta, P. Srinivasan, J. Shi, and L.S. Davis, "Understanding Videos Constructing Plots: Learning a Visually Grounded Storyline Model from Annotated Videos," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[22] S. Gupta and R. Mooney, "Using Closed Captions to Train Activity Recognizers that Improve Video Retrieval," *Proc. IEEE Computer Vision and Pattern Recognition Workshop Visual and Contextual Learning from Annotated Images and Videos,* June 2009.

[23] S. Gupta and R.J. Mooney, "Using Closed Captions as Supervision for Video Activity Recognition," *Proc. 24th AAAI Conf. Artificial Intelligenc,* pp. 1083-1088, July 2010.

[24] A. Kojima, T. Tamura, and K. Fukunaga, "Natural Language Description of Human Activities from Video Images Based on Concept Hierarchy of Actions," *Int'l J. of Computer Vision,* vol. 50, pp. 171-184, 2002.

[25] V. Kolmogorov, "Convergent Tree-Reweighted Message Passing for Energy Minimization," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 10, pp. 1568-1583, Oct. 2006.

[26] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A.C. Berg, and T.L. Berg, "Babytalk: Understanding and Generating Simple Image Descriptions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2011.

[27] N. Kumar, A.C. Berg, P.N. Belhumeur, and S.K. Nayar, "Attribute and Simile Classifiers for Face Verification," *Proc. 12th IEEE Int'l Conf. Computer Vision,* 2009.

[28] P. Kuznetsova, V. Ordonez, A.C. Berg, T.L. Berg, and Y. Choi, "Collective Generation of Natural Image Descriptions," *Proc. Conf. Assoc. for Computational Linguistics,* 2012.

[29] C. Lampert, H. Nickisch, and S. Harmeling, "Learning to Detect Unseen Object Classes by Between-Class Attribute Transfer," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[30] L.-J. Li and L. Fei-Fei, "OPTIMOL: Automatic Online Picture Collection via Incremental Model Learning," *Int'l J. Computer Vision,* vol. 88, pp. 147-168, 2009.

[31] S. Li, G. Kulkarni, T.L. Berg, A.C. Berg, and Y. Choi, "Composing Simple Image Descriptions Using Web-Scale n-Grams," *Proc. 15th Conf. Computational Natural Language Learning,* pp. 220-228, June 2011.

[32] C.-Y. Lin and E. Hovy, "Automatic Evaluation of Summaries Using N-Gram Co-Occurrence Statistics," *Proc. Conf. North Am. Chapter of the Assoc. for Computational Linguistics on Human Language Technology),* 2003.

[33] V. Ordonez, G. Kulkarni, and T.L. Berg, "Im2text: Describing Images Using 1 Million Captioned Photographs," *Proc. Neural Information Processing Systems),* 2011.

[34] K. Papineni, S. Roukos, T. Ward, and W. Jing Zhu, "Bleu: A Method for Automatic Evaluation of Machine Translation," *Proc. 40th Ann. Meeting of Assoc. for Computational Linguistics,* pp. 311-318, 2002.

[35] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier, "Collecting Image Annotations Using Amazon's Mechanical Turk," *Proc. NAACL HLT Workshop Creating Speech and Language Data with Amazon's Mechanical Turk,* 2010.

[36] K. Saenko and T. Darrell, "Unsupervised Learning of Visual Sense Models for Polysemous Words," *Proc. Neural Information Processing Systems,* 2008.

[37] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting Image Databases from the Web," *Proc. 11th IEEE Int'l Conf. Computer Vision,* 2007.

[38] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context," *Int'l J. Computer Vision,* vol. 81, pp. 2-23, Jan. 2009.

[39] J. Sivic, M. Everingham, and A. Zisserman, ""Who Are You?"—Learning Person Specific Classifiers from Video," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[40] A. Torralba, R. Fergus, and W. Freeman, "80 Million Tiny Images: A Large Data Set for Non-Parametric Object and Scene Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 11, pp. 1958-1970, Nov. 2008.

[41] A. Torralba, K.P. Murphy, and W.T. Freeman, "Using the Forest to See the Trees: Exploiting Context for Visual Object Detection and Localization," *Comm. ACM,* vol. 53, pp. 107-114, Mar. 2010.

[42] M.J. Wainwright, T.S. Jaakkola, and A.S. Willsky, "Map Estimation via Agreement on (Hyper)Trees: Message-Passing and Linear-Programming Approaches," *IEEE Trans. Information Theory,* vol. 51, no. 11, pp. 3697-3717, Nov. 2005.

[43] J. Wang, K. Markert, and M. Everingham, "Learning Models for Object Recognition from Natural Language Descriptions," *Proc. British Machine Vision Conf.,* 2009.

[44] K. Yanai and K. Barnard, "Image Region Entropy: A Measure of 'Visualness' of Web Images Associated with one Concept," *Proc. 13th Ann. ACM Int'l Conf. Multimedia,* 2005.

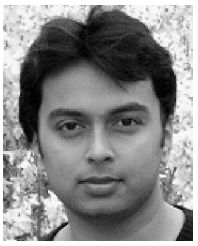
[45] K. Yanai and K. Barnard, "Finding Visual Concepts by Web Image Mining," *Proc. 15th Int'l Conf. World Wide Web,* pp. 923-924, 2006.

[46] Y. Yang, C.L. Teo, H. Daumé III, and Y. Aloimonos, "Corpus-Guided Sentence Generation of Natural Images," *Proc. Conf. Empirical Methods in Natural Language Processing,* 2011.

[47] B. Yao, X. Yang, L. Lin, M.W. Lee, and S.-C. Zhu, "I2t: Image Parsing to Text Description," *Proc. IEEE,* vol. 98, no. 8, Aug. 2010.

[48] J. Zacks, B. Tversky, and G. Iyer, "Perceiving, Remembering, and Communicating Structure in Events," *J. Experimental Psychology General,* vol. 130, pp. 29-58, Mar. 2001.

[49] L. Zhou and E. Hovy, "Template-Filtered Headline Summarization," *Proc. ACL Workshop Text Summarization Branches Out,* July 2004.



Girish Kulkarni received the master's degree in computer science from Stony Brook University, advised by Dr. Tamara Berg. His main research areas include computer vision and natural language processing. He worked at IBM-GBS for 2 years and presently works at Epic Systems.

Visruth Premraj received the master's degree in computer science from the State University of New York, Stony Brook, in 2010, advised by Dr. Tamara Berg. His research interests include object detection and interaction with multimedia through real-time action and gesture recognition. Currently, he is working as a software developer at Epic Systems (Research and Development), Madison.

Vicente Ordonez received the engineering degree from the Escuela Superior Politécnica del Litoral in Guayaquil, Ecuador. He is working toward the PhD degree in the Computer Science Department at Stony Brook University, advised by Dr. Tamara Berg. His research interests include digital media, computer vision, and large scale image understanding. He is also a recipient of a Renaissance Technologies Fellowship 2009-2011. He is a student member of the IEEE.

Sagnik Dhar received the BS degree in computer science from Visvesvaraya Technological University and the master's degree in computer science from Stony Brook University in 2010, advised by Dr. Tamara Berg. Currently, he works for Sportvision, a company which develops techniques to augment visual and textual information to enhance live TV broadcasts of sporting events. Prior to that, he spent a year at the Honda Research Institute.

Siming Li is working toward the PhD degree at Stony Brook University. Her research interests include algorithm design, wireless sensor network, and machine learning. She is a student member of the IEEE.

Yejin Choi received the BS degree in computer science and engineering from Seoul National University and the PhD degree in computer science from Cornell University in 2010. Currently, she is an assistant professor in the Computer Science Department at Stony Brook University. Her area of research is natural language processing, where she has been focusing on the connection between computer vision and language processing, sociocognitive stylometric analysis, and opinion analysis.

Alexander C. Berg received the BA and MA degrees in mathematics from Johns Hopkins University and the PhD degree in computer science from the University of California, Berkeley, in 2005. Currently, he is an assistant professor at Stony Brook University. Prior to that, he was a research scientist at Yahoo! Research and later Columbia University. His research addresses challenges in visual recognition at all levels, from features for alignment and object recognition, to action recognition in video, to object detection, to high-level semantics of hierarchical classification, attributes, and natural language descriptions of images. Much of this work has a focus on efficient large-scale solutions. He is a member of the IEEE.

Tamara L. Berg received the BS degree in mathematics and computer science from the University of Wisconsin–Madison and the PhD degree in computer science from the University of California, Berkeley, in 2007. Currently, she is an assistant professor at Stony Brook University. Prior to that, she was a research scientist at Yahoo! Research. Her main research focus is on integrating disparate information from text and image analysis for collection organization, recognition, search, and describing the visual world using natural language. Other areas of interest include visual recognition and integrating computational recognition with social analysis, especially for style and trend analysis. She is a member of the IEEE.