LANGUAGE AND PERCEPTUAL CATEGORIZATION IN COMPUTATIONAL VISUAL RECOGNITION

Vicente Ordóñez Román

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2015

Approved by:

Tamara L. Berg

Alexander C. Berg

Yejin Choi

Jan-Michael Frahm

Alexei A. Efros

ABSTRACT

VICENTE ORDÓÑEZ ROMÁN: LANGUAGE AND PERCEPTUAL CATEGORIZATION IN COMPUTATIONAL VISUAL RECOGNITION.
(Under the direction of Tamara L. Berg.)

Computational visual recognition or giving computers the ability to understand images as well as humans do is a core problem in Computer Vision. Traditional recognition systems often describe visual content by producing a set of isolated labels, object locations, or by even trying to annotate every pixel in an image with a category. People instead describe the visual world using language. The rich visually descriptive language produced by people incorporates information from human intuition, world knowledge, visual saliency, and common sense that go beyond detecting individual visual concepts like objects, attributes, or scenes. Moreover, due to the rising popularity of social media, there exist billions of images with associated text on the web, yet systems that can leverage this type of annotations or try to connect language and vision are scarce.

In this dissertation, we propose new approaches that explore the connections between language and vision at several levels of detail by combining techniques from Computer Vision and Natural Language Understanding. We first present a data-driven technique for understanding and generating image descriptions using natural language, including automatically collecting a big-scale dataset of images with visually descriptive captions. Then we introduce a system for retrieving short visually descriptive phrases for describing some part or aspect of an image, and a simple technique to generate full image descriptions by stitching short phrases. Next we introduce an approach for collecting and generating referring expressions for objects in natural scenes at a much larger scale than previous studies. Finally, we describe methods for learning how to name objects by using intuitions from perceptual categorization related to basic-level and entry-level categories.

The main contribution of this thesis is in advancing our knowledge on how to leverage language and intuitions from human perception to create visual recognition systems that can better learn from and communicate with people.

Dedicada a la memoria de mi abuelo Ángel Pacífico Román Silva (1931 - 2014).

ACKNOWLEDGMENTS

My deepest gratitude to my advisor Tamara for her guidance and help throughout my graduate career. Her brilliance, and passion for knowledge and science have always been a motivation and a source of inspiration. I am thankful and proud to have worked with her for many years, some of the happiest and most eventful years of my life. I also want to thank Alex Berg and Yejin Choi who further contributed to shape the researcher in me both as my close collaborators and mentors. I have been lucky to effectively rely on their advice as well. Special thanks to David Forsyth for being a role model, and for his trust and unconditional support to my career.

Thanks also to all the professors and teachers who contributed to my graduate education. Special thanks to Dimitris Samaras for his teachings, wisdom, and helpful discussions. To Luis Ortiz who helped me on more than one occasion with his insights and deep technical expertise. To Greg Zelinsky for his unique feedback during my presentations and for opening the doors of the Eye Cognition Lab at Stony Brook to me. To Jan-Michael Frahm for welcoming me to North Carolina and for his feedback and service on my thesis committee. To Alyosha Efros for being part of my thesis committee and for being an inspiration in my approach to research. Thanks to Prof. Fred Brooks and Ron Alterovitz for their class on technical communication. Thanks to Xavier Ochoa and by extension to his advisor Erik Duval with whom I was also lucky to briefly collaborate.

I also want to thank all my outside mentors from industry, especially to Rodrigo Carceroni. Rodrigo is perhaps the single person who has had the most impact on my overall career. His guidance and encouragement during my first internship at Google cemented my interest in pursuing a career as a computer vision scientist. I was lucky enough to collaborate with him again during my time as a graduate student, and I have kept in touch ever since. I also thank Neil Hunt for his mentorship and opportune career advice. Thanks also to Ce Liu for his guidance during my internship at Microsoft Research and for his continued advice and support. Likewise, I am thankful to all my collaborators at the eBay Research Labs, especially to Vignesh Jagadeesh and Robinson Piramuthu.

Thanks to all my many collaborators, especially to Polina Kuznetsova with whom we developed a productive streak of collaborations; to Sahar Kazemzadeh whose collaboration during her senior year of college made possible a chapter of this thesis; to Girish Kulkarni whose collaboration produced the first chapter in this thesis; and to Jia Deng whose work and collaboration was the starting point for our award-winning research on entry-level categories that is also part of this thesis. Thanks to the great group of people who participated in the workshop on learning from visually descriptive text at the Center for Language and Speech Processing at Johns Hopkins University during the summer of 2011 whose work and collaboration is also reflected in a chapter of this thesis. Special thanks to my friend, and long time and closest labmate Kota Yamaguchi whose integrity, intelligence, and attention to detail earned my deepest admiration. I tried to borrow as much as I could from Kota.

I also would like to acknowledge several other people whose work was also influential in my research, some of whom I have been lucky enough to meet: Julia Hockenmaier, Mirella Lapata, Ali Farhadi, Kobus Barnard, Larry Zitnick, Margaret Mitchell, Fei-fei Li, Abhinav Gupta, Ehud Reiter, Luke Zettlemoyer, Stephen Kosslyn, and Eleanor Rosch.

Thanks also to all my fellow students, labmates, and friends who have made my life easier in New York and North Carolina, others that I have met through internships in Silicon Valley and New England, or through research conferences including: Paul Arauz, Xiang Gao, Andrii Cherniak, Debaleena Chattopadhyay, Yifan Peng, Marc Serra, Crystal Liu, Cecilia Cheung, Raquel Perales, Nafees Ahmed, Nicholas Tsoi-a-sue, Raul Santos, Chen-ping Yu, Jean Honorio, Tomas Yago, Michail Misyrlis, Kiwon Yun, Jane Gutenko, Song Feng, Xufeng Han, Sirion Vittayakorn, Hadi Kiapour, Licheng Yu, Yipin Zhou, Eunbyung Park, Sridutt Bhalachandra, Wei Liu, Katharina Schwarz, True Price, Dinghuang Ji, Enliang Zheng, Rohit Gupta, Johannes Schoenberger, Chad Armstrong, Tianfan Xue, Miki Rubinstein, Aaron Potechin, Zitao Liu, Anuradha Uduwage, Juan Carlos Niebles, Rebecca Mason, Adriana Kovashka, Enrique Dunn, Jorge Poco, Diego Gavinowich, Helder Suzuki, Wei Hua, Keir Mierle, Hong Guo, Hui Fang, James Wu, Lin Yang, Ramiro Málaga, Vasilis Tsiropoulos, Maria SanMartin, and truly many others left unmentioned. Thanks to Jefferson Torres, Boris Floril and Adrián Peñaloza for staying close as friends for very long despite the distances. Many thanks to Tilsa Oré for being with me and sharing many stories through the last lap in my graduate career and beyond. Thanks to my family, for their continuous support through my entire life, to my beloved mother Ivonne, to my brother Javier, my sister Nadia, and my cousin Daniel.

Thanks to the Departments of Computer Science at Stony Brook University in New York State and the University of North Carolina at Chapel Hill, the professors, and administrative staff, for their trust and active investment in my graduate education. I will always be a proud alumni of those two great institutions. Thanks to my former professors, colleagues, and all my friends back in Ecuador. Thanks to the thousands of users in Amazon Mechanical Turk who through their hard work helped us advance our understanding of the world. Thanks to Renaissance Technologies for providing me with a generous fellowship during the first three years of my graduate studies, and Yahoo! for selecting me among their Key Scientific Challenges awardees. Finally, thanks to the National Science Foundation. This work was supported by NSF Career Award #1444234 and NSF Award #1445409.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

The objective of computational visual recognition is to ultimately duplicate the recognition capabilities of human vision using computational methods. This is a definition that can be very broadly interpreted, and as such, computer vision systems have focused on several well defined tasks that output different types of symbolic information given input visual data like images, sets of images, or video. More concretely, traditional computer vision systems often output a label or a set of disconnected labels (categorization or tagging), a set of labeled boxes (detection), or even try to group and label every individual pixel in an image with a semantic category (semantic segmentation, parsing). While we humans are able to perform these kind of tasks, as we often use these abilities to annotate the training data used in vision systems, our everyday interpretation of the visual world is more naturally expressed using language. The main goal of this thesis is to study and bridge the gap between the output of computer vision systems and what humans describe about the visual world using natural language.

Moreover, computational visual recognition systems have seen a rapid improvement in the last couple of years and this trend continues for several standard tasks like categorization and object detection. As we obtain systems that can reason more effectively about basic visual structures in images, there is an increasing need to understand visual content at an even higher - more human - level of abstraction. We propose in this thesis four tasks and techniques that can generate explanations of the visual world that are closer to human interpretations using natural language.

1.1 Previous work

There are several works that have looked before at the connections between words and pictures for various tasks (Duygulu et al., 2002; Barnard et al., 2003; Barnard and Yanai, 2006; Berg et al., 2004). While these these systems are either able to associate isolated words to image content or learn visual models from text, they do not produce language as an output. Since generating language is not the ultimate goal for these systems, they also discard a lot of information from language to focus on some specific content like names or nouns.

Some later work studied image description generation (Aker and Gaizauskas, 2010a; Farhadi et al., 2010; Kulkarni et al., 2013; Feng and Lapata, 2013) but there are a few important distinctions with our work. The work of (Aker and Gaizauskas, 2010a) and (Feng and Lapata, 2013) assume that there is already text associated with images or it can be readily obtained, and then apply a summarization approach. Our methods do not assume the existence of text for an input image. The method of (Kulkarni et al., 2013) and other similar methods proposed later rely on a constructive approach where the language is built word by word and directly from the output of computer vision detections. Our automatic image description methods rely heavily on a data-driven approach that tries to borrow as much as possible from actual captions written by people. The work of (Farhadi et al., 2010) used an intermediate triplet representation coupled with a retrieval approach to associate descriptions with images but the set of descriptions in their pool was limited.

We use a pool of image descriptions that is two orders of magnitude larger. Additionally, we propose a hybrid caption generation approach that combines retrieving pieces of text and composing new descriptions. We also cover in this thesis the significantly unexplored problem of referring to objects in the context of complex natural scenes using referring expressions, and propose a new task of learning how to name objects with entry-level categories using computational visual recognition.

1.2 Outline of Contributions

In Chapter 2, we introduce a data-driven sentence retrieval approach to produce full sentence descriptions for new images (Ordonez et al., 2011). An overview of our baseline

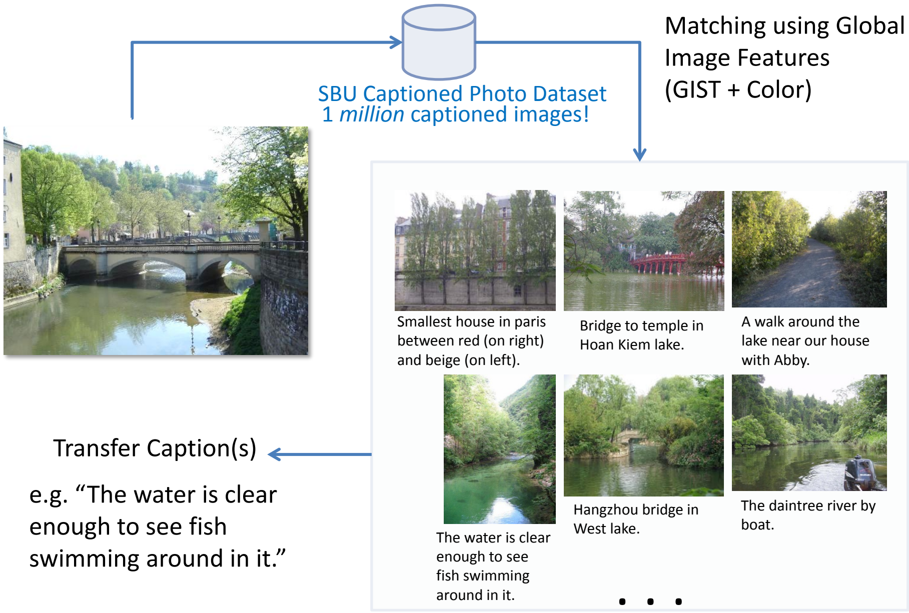
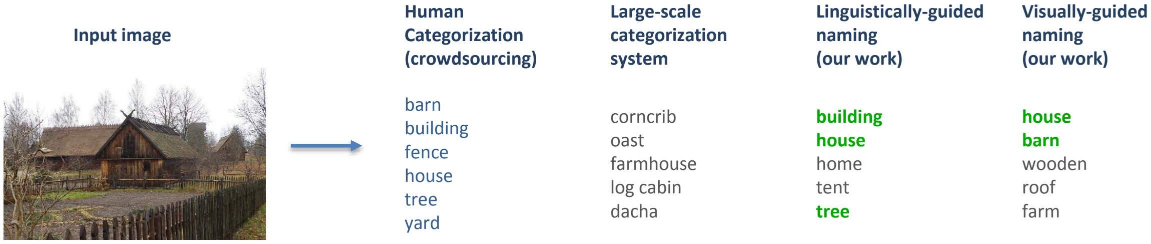

Matching using Global Image Features (GIST + Color)
SBU Captioned Photo Dataset
1 million captioned images!
Smallest house in paris between red (on right) and beige (on left).
Bridge to temple in Hoan Kiem lake.
A walk around the lake near our house with Abby.
Transfer Caption(s)
e.g. "The water is clear enough to see fish swimming around in it."
The water is clear enough to see fish swimming around in it.
Hangzhou bridge in West lake.
The daintree river by boat.
. . .

Figure 1.1: We approach this task in a data-driven manner by first building a 1 million dataset of images with visually relevant captions. We use standard global image feature descriptors such as GIST and *tiny images* (Torralba et al., 2008) to retrieve similar images from which we can directly transfer captions.

system and starting idea is presented in Figure 1.1. We leverage object detection, stuff detection, people detection, and scene recognition coupled with text statistics to learn and improve our similarity metric used during the retrieval step. The effectiveness of this system heavily relies on data, therefore a key contribution of this work was also to devise a method to collect from the web a big scale dataset of images paired with visually descriptive captions. This dataset, that we named the *SBU Captioned Dataset*, has been used in several other later publications that aim to produce natural language (Mitchell et al., 2012; Kuznetsova et al., 2012; Gupta et al., 2012; Mason and Charniak, 2014; Kuznetsova et al., 2014), including more recent methods that rely on *deep learning* (Kiros et al., 2014; Vinyals et al., 2014).

In Chapter 3, we introduce a system that given an input image produces short-descriptive text phrases that describe only a part or an aspect of an image (Ordonez et al., 2013b). This is a middle ground between outputting individual labels and full sentences. Short phrases have a descriptive power that goes beyond labels while also being potentially more generalizable to new images. This system also allows to compose different phrases using text statistics. This approach coupled with more sophisticated language models and constraints was used in a related set of publications (Kuznetsova et al., 2012, 2014). For a complete overview of these and related approaches refer to (Kuznetsova, 2014).

In Chapter 4, we study task-dependent descriptions where the objective is to identify an individual object using *Referring Expressions*. We introduce one of the first studies on referring expressions in the context of natural scenes. We also collect one of the largest



Human Categorization (crowdsourcing)	Large-scale categorization system	Linguistically-guided naming (our work)	Visually-guided naming (our work)
barn	corncrib	**building**	**house**
building	oast	**house**	**barn**
fence	farmhouse	home	wooden
house	log cabin	tent	roof
tree	dacha	**tree**	farm
yard			

Figure 1.2: Category predictions for a given input image for a large-scale categorization system and our translated outputs using linguistically and visually-informed models. The first column includes names given by people for this image that we collected using crowdsourcing to measure the performance of our models. We highlight in green the predicted names that were also mentioned by people. Note that *oast* is a type of farm building for drying hops and a *dacha* is a type of Russian farm building.

datasets for referring expressions by using a purpose-driven game. We also present a detailed analysis of this data and a technique based on constraint optimization to generate referring expressions using our dataset statistics (Kazemzadeh, Ordonez et al., 2014).

Finally, we found that even when predicting isolated words, good computational visual recognition systems still often produce sets of categories that do not correspond to the set of categories that people would use. In Chapter 5, we present a system that can translate encyclopedic categories used in large scale image categorization systems into names that people use in everyday language (Ordonez et al., 2013a). We introduce a sample output of two of our methods presented here in Figure 1.2 to showcase our motivation. This problem is related to the notion of basic-level and entry-level categories in cognitive psychology.

In summary the novel contributions presented in this thesis are as follows:

1. A big-scale dataset of images with visually descriptive captions collected automatically by leveraging existing captioned images on the web and a data-driven

approach to retrieve image descriptions using various measures of visual similarity (Chapter 2).

2. A system to retrieve and rank short-text phrases that describe parts or aspects of an image and two applications of this system to a) generate image descriptions and b) resolve complex image queries (Chapter 3).

3. A purpose-driven game to collect Referring Expressions of objects in natural scenes and a system that can use this dataset to generate referring expressions from annotated input images with target objects (Chapter 4).

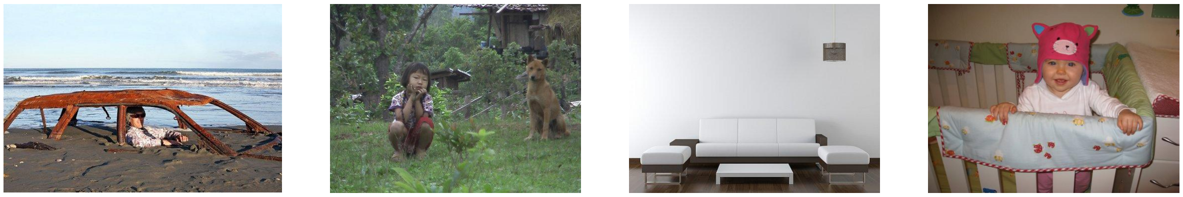
4. A category translation system that predicts the names that people use in everyday language from encyclopedic concepts and input images. And an application to retrieve image descriptions using those predicted names (Chapter 5)

CHAPTER 2: DATA-DRIVEN IMAGE CAPTIONING

Producing a full sentence or image caption that is both relevant and accurate for an arbitrary input image is extremely challenging. Even if computational visual recognition systems were able to accurately recognize every visual element in an image it would still be difficult to use this information to generate a coherent idea about a scene. However there are already billions of images with visually descriptive captions on the web. We present a $d$ata-driven approach for caption generation. We first describe a technique for automatically collecting and filtering a big scale collection of images with visually descriptive text. Then, we use this dataset to retrieve captions using a simple non-parametric approach in the spirit of previous research that makes use of big data for various applications (Hays and Efros, 2008; Torralba et al., 2008; Tighe and Lazebnik, 2010). We additionally show that using noisy predictions of image content we can learn a better similarity metric that can return more relevant visual results and captions.

The collected dataset described in this chapter contains 1 million images with visually descriptive captions (see examples in Figure 2.1). In addition to using this dataset for sentence generation, we also use it as the basis for our short-descriptive phrase prediction system in Chapter 3 and is an important component in the entry-level category prediction system in Chapter 5.

We describe the dataset collection in Section 2.1, caption generation using a global representation in Section 2.2, content estimation for various content types in Section 2.3,



Man sits in a rusted car buried in the sand on Waitarere beach

Little girl and her dog in northern Thailand. They both seemed interested in what we were doing

Interior design of modern white and brown living room furniture against white wall with a lamp hanging.

Emma in her hat looking super cute

Figure 2.1: **SBU Captioned Photo Dataset:** Photographs with user-associated captions from our web-scale captioned photo collection. We collect a large number of photos from Flickr and filter them to produce a data collection containing over 1 million well captioned pictures.

and we finally present an extension to our generation method that incorporates content estimates in Section 2.4. This work was originally published in (Ordonez et al., 2011) and is also summarized in (Ordonez et al., 2013b).

2.1 Building a Web-Scale Captioned Collection

One key contribution presented in this chapter is a novel web-scale database of photographs with associated descriptive text. To enable effective captioning of novel images, this database must be good in two ways: 1) It must be large so that image based matches to a query are reasonably similar, 2) The captions associated with the database photographs must be visually relevant so that transferring captions between pictures is useful. To achieve the first requirement we query Flickr using a huge number of pairs of query terms (objects, attributes, actions, stuff, and scenes). This produces a very large, but noisy initial set of photographs with associated text. To achieve our second requirement we filter this set of photos so that the descriptions attached to a picture are relevant and visually descriptive. To encourage visual descriptiveness in our collection, we select

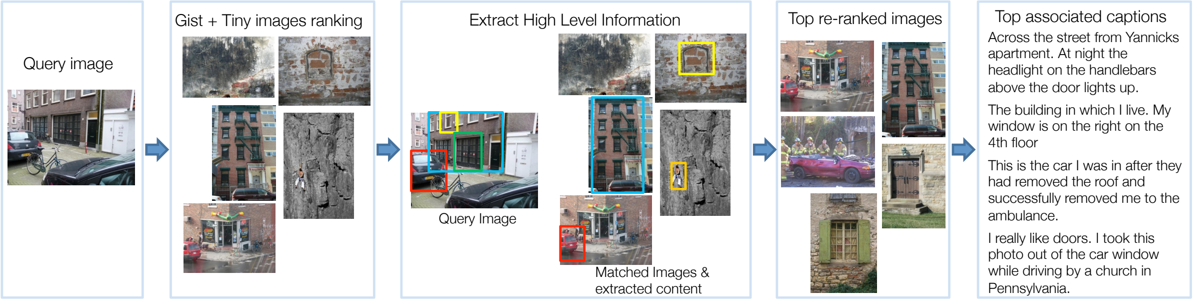

Query image
Gist + Tiny images ranking
Extract High Level Information
Query Image
Matched Images & extracted content
Top re-ranked images
Top associated captions
Across the street from Yannicks apartment. At night the headlight on the handlebars above the door lights up.
The building in which I live. My window is on the right on the 4th floor
This is the car I was in after they had removed the roof and successfully removed me to the ambulance.
I really like doors. I took this photo out of the car window while driving by a church in Pennsylvania.

Figure 2.2: **System flow:** 1) Input query image, 2) Candidate matched images retrieved from our web-scale captioned collection using global image representations, 3) High level information is extracted about image content including objects, attributes, actions, people, stuff, scenes, and tfidf weighting, 4) Images are re-ranked by combining all content estimates, 5) Top 4 resulting captions.

only those images with descriptions of satisfactory length based on observed lengths in visual descriptions. We also enforce that retained descriptions contain at least 2 words belonging to our term lists and at least one prepositional word, e.g. “on”, “under” which often indicate visible spatial relationships.

This results in a final collection of over 1 million images with associated text descriptions – the *SBU Captioned Photo Dataset*. These text descriptions generally function in a similar manner to image captions, and usually directly refer to some aspects of the visual image content (see fig 4.5 for examples). Hereafter, we will refer in this chapter to this web based collection of captioned images as C.

Query Set: We randomly sample 500 images from our collection for evaluation of our generation methods (examples are shown in Figure 4.5). As is usually the case with web photos, the photos in this set display a wide range of difficulty for visual recognition algorithms and captioning, from images that depict scenes (e.g. beaches), to images with

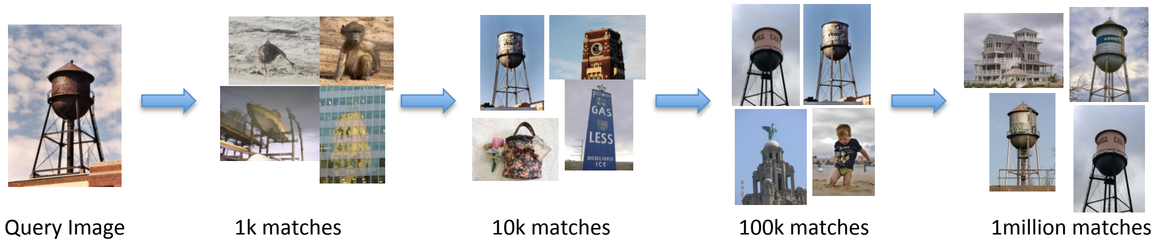


Figure 2.3: **Size Matters:** Example matches to a query image for varying data set sizes.

a relatively simple depictions (e.g. a horse in a field), to images with much more complex depictions (e.g. a boy handing out food to a group of people).

2.2 Global Description Generation

Internet vision papers have demonstrated that if your data set is large enough, some very challenging problems can be attacked with very simple matching methods (Hays and Efros, 2008; Torralba et al., 2008; Tighe and Lazebnik, 2010). In this spirit, we harness the power of web photo collections in a non-parametric approach. Given a query image, I_q, our goal is to generate a relevant description. We achieve this by computing the global similarity of a query image to our large web-collection of captioned images, C. We find the closest matching image (or images) and simply transfer over the description from the matching image to the query image. We also collect the 100 most similar images to a query – our matched set of images $I_m \in M$ – for use in our our content based description generation method (Sec 2.4).

For image comparison we utilize two image descriptors. The first descriptor is the well known gist feature, a global image descriptor related to perceptual dimensions – naturalness, roughness, ruggedness etc – of scenes. The second descriptor is also a global image descriptor, computed by resizing the image into a “tiny image”, essentially a thumbnail of size 32x32. This helps us match not only scene structure, but also the overall color of images. To find visually relevant images we compute the similarity of the query image to images in C using a sum of gist similarity and tiny image color similarity (equally weighted).

Results – Size Matters! Our global caption generation method is illustrated in the first 2 panes and the first 2 resulting captions of Figure 2.2. This simple method often performs surprisingly well. As reflected in past work (Hays and Efros, 2008; Torralba et al., 2008), image retrieval from small collections often produces spurious matches. This can be seen in Figure 2.3 where increasing data set size has a significant effect on the quality of retrieved global matches. Quantitative results also reflect this (see Table 2.1).

2.3 Image Content Estimation

Given an initial matched set of images $I_m \in M$ based on global descriptor similarity, we would like to re-rank the selected captions by incorporating estimates of image content. For a query image, I_q and images in its matched set we extract and compare 5 kinds of image content:

- Objects (e.g. cats or hats), with shape, attributes, and actions – sec 2.3.1
- Stuff (e.g. grass or water) – sec 2.3.2
- People (e.g. man), with actions – sec 2.3.3
- Scenes (e.g. pasture or kitchen) – sec 2.3.4

- TFIDF weights (text or detector based) – sec 2.3.5

Each type of content is used to compute the similarity between matched images (and captions) and the query image. We then rank the matched images (and captions) according to each content measure and combine their results into an overall relevancy ranking (Sec 2.4).

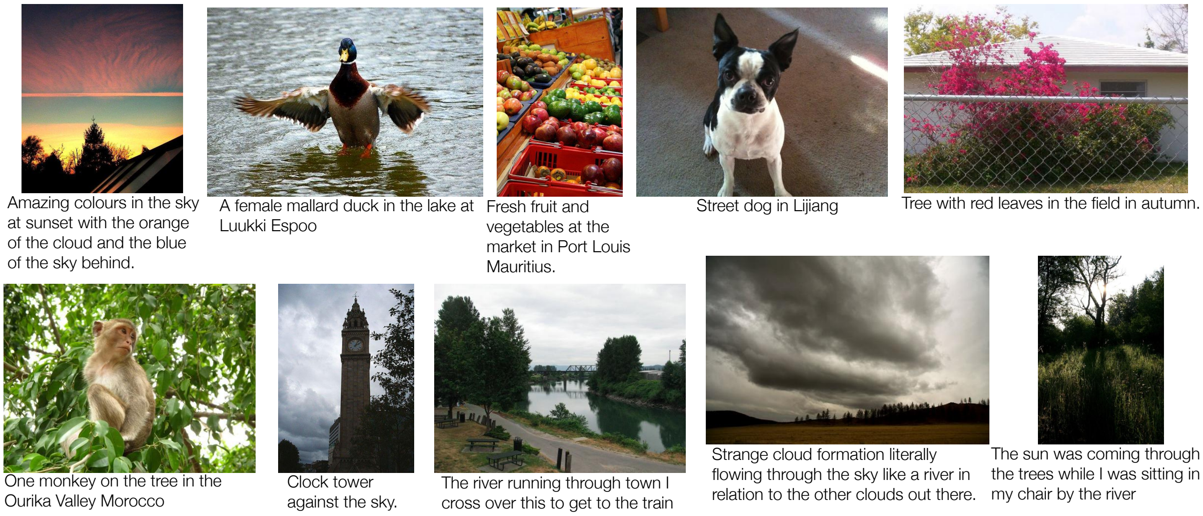
2.3.1 Objects

Detection & Actions: Object detection methods have improved significantly in the last few years, demonstrating reasonable performance for a small number of object categories (Everingham et al., 2010), or as a mid-level representation for scene recognition (Li et al., 2010). Running detectors on general web images however, still produces quite noisy results, usually in the form of a large number of false positive detections. As the number of object detectors increases this becomes even more of an obstacle to content prediction. However, we propose that if we have some prior knowledge about the content of an image, then we can utilize even these imperfect detectors. In our web collection, C, there are strong indicators of content in the form of caption words – if an object is described in the text associated with an image then it is likely to be depicted. Therefore, for the images, $I_m \in M$, in our matched set, we run only those detectors for objects (or stuff) that are mentioned in the associated caption. In addition, we also include synonyms and hyponyms for better content coverage, e.g. "dalmatian" triggers "dog" detector. This produces pleasingly accurate detection results. For a query image we can essentially perform detection verification against the relatively clean matched image detections.

Specifically, we use mixture of multi-scale deformable part detectors (Felzenszwalb et al., 2010) to detect a wide variety of objects – 89 object categories selected to cover a reasonable range of common objects. These categories include the 20 Pascal categories, 49 of the most common object categories with reasonably effective detectors from Object Bank (Li et al., 2010), and 20 additional common object categories.

For the 8 animate object categories in our list (e.g. cat, cow, duck) we find that detection performance can be improved significantly by training *action specific detectors*, for example "dog sitting" vs "dog running". This also aids similarity computation between a query and a matched image because objects can be matched at an action level. Our object action detectors are trained using the standard object detector with pose specific training data.

Representation: We represent and compare object detections using two kinds of features: shape and appearance. To represent *object shape* we use a histogram of HoG (Dalal and Triggs, 2005) visual words, computed at intervals of 8 pixels and quantized into 1000 visual words. These are accumulated into a spatial pyramid histogram (Lazebnik et al., 2006). We also use an *attribute representation* to characterize object appearance. We use the attribute list from (Kulkarni et al., 2013) which covers 21 visual aspects describing color (e.g. blue), texture (e.g. striped), material (e.g. wooden), general appearance (e.g. rusty), and shape (e.g. rectangular). Training images for the attribute classifiers come from Flickr, Google, the attribute dataset provided by (Farhadi et al., 2009), and ImageNet (Deng et al., 2009). An RBF kernel SVM is used to learn a classifier for each



Amazing colours in the sky at sunset with the orange of the cloud and the blue of the sky behind.

A female mallard duck in the lake at Luukki Espoo

Fresh fruit and vegetables at the market in Port Louis Mauritius.

Street dog in Lijiang

Tree with red leaves in the field in autumn.

One monkey on the tree in the Ourika Valley Morocco

Clock tower against the sky.

The river running through town I cross over this to get to the train

Strange cloud formation literally flowing through the sky like a river in relation to the other clouds out there.

The sun was coming through the trees while I was sitting in my chair by the river

Figure 2.4: **Results:** Some good captions selected by our system for query images.

attribute term. Then appearance characteristics are represented as a vector of attribute responses to allow for generalization.

If we have detected an object category, c, in a query image window, O_q and a matched image window, O_m, then we compute the probability of an object match as:

$$P(O_q, O_m) = e^{-D_o(O_q, O_m)}$$

where $D_o(O_q, O_m)$ is the Euclidean distance between the object (shape or attribute) vector in the query detection window and the matched detection window.

2.3.2 Stuff

In addition to objects, people often describe the stuff present in images, e.g. “grass”. Because these categories are more amorphous and do not display defined parts, we use a region based classification method for detection. We train linear SVMs on the low level region features of (Farhadi et al., 2009) and histograms of Geometric Context output probability maps (Hoiem et al., 2007) to recognize: sky, road, building, tree, water, and grass stuff categories. While the low level features are useful for discriminating stuff by their appearance, the scene layout maps introduce a soft preference for certain spatial locations dependent on stuff type. 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.

If we have detected a stuff category, s, in a query image region, S_q and a matched image region, S_m, then we compute the probability of a stuff match as:

$$P(S_q, S_m) = P(S_q = s) * P(S_m = s)$$

where $P(S_q = s)$ is the SVM probability of the stuff region detection in the query image and $P(S_m = s)$ is the SVM probability of the stuff region detection in the matched image.

2.3.3 People & Actions

People often take pictures of people, making "person" the most commonly depicted object category in captioned images. We utilize effective recent work on pedestrian detectors to detect and represent people in our images. In particular, we make use of detectors from (Bourdev et al., 2010) which learn poselets – parts that are tightly clustered in configuration and appearance space – from a large number of 2D annotated regions on person images in a max-margin framework. To represent activities, we use follow up work from (Maji et al., 2011) which classifies actions using a poselet activation vector. This has been shown to produce accurate activity classifiers for the 9 actions in the PASCAL VOC 2010 static image action classification challenge (Everingham et al., 2010). We use the outputs of these 9 classifiers as our action representation vector, to allow for generalization to other similar activities.

If we have detected a person, P_q, in a query image, and a person P_m in a matched image, we compute the probability that the people share the same action (pose) as:

$$P(P_q, P_m) = e^{-D_p(P_q, P_m)}$$

where $D_p(P_q, P_m)$ is the Euclidean distance between the person action vector in the query detection and the person action vector in the matched detection.

2.3.4 Scenes

The last commonly described kind of image content relates to the general scene where an image was captured. This often occurs when examining captioned photographs of vacation snapshots or general outdoor settings, e.g. “my dog at the beach”. To recognize scene types we train discriminative multi-kernel classifiers using the large-scale SUN scene recognition dataset and code (Xiao et al., 2010). We select 23 common scene categories for our representation, including indoor (e.g. kitchen) outdoor (e.g. beach), man-made (e.g. highway), and natural (pasture) settings. Again, here we represent the scene descriptor as a vector of scene responses for generalization.

If a scene location, L_m, is mentioned in a matched image, then we compare the scene representation between our matched image and our query image, L_q as:

$$P(L_q, L_m) = e^{-D_l(L_q, L_m)}$$

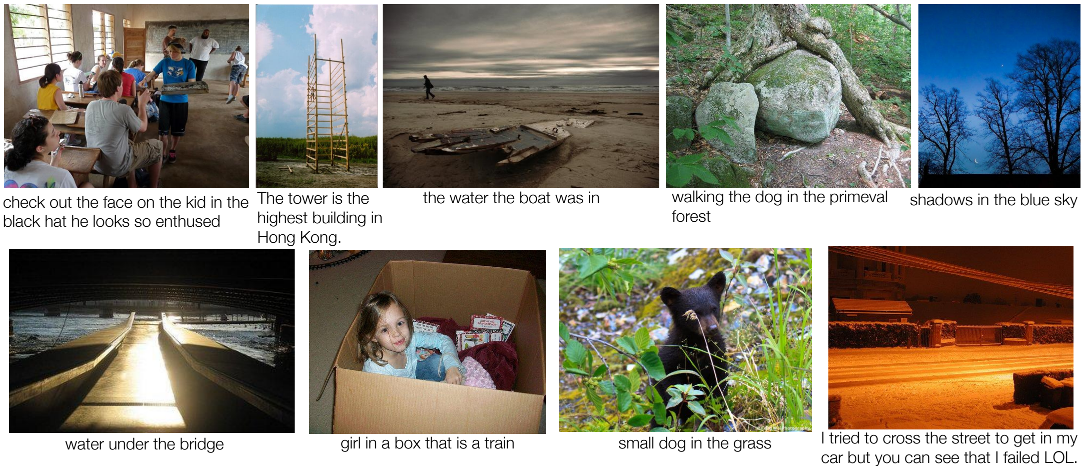
where $D_l(L_q, L_m)$ is the Euclidean distance between the scene vector computed on the query image and the scene vector computed on the matched image.

2.3.5 TFIDF Measures

For a query image, I_q, we wish to select the best caption from the matched set, $I_m \in M$. For all of the content measures described so far, we have computed the similarity of the query image content to the content of each matched image independently. We would also like to use information from the entire matched set of images and associated captions to predict importance. To reflect this, we calculate TFIDF on our matched sets. This is computed as usual as a product of term frequency (tf) and inverse document frequency (idf). We calculate this weighting both in the standard sense for matched caption document words and for detection category frequencies (to compensate for more prolific object detectors).

$$tfidf = \frac{n_{i,j}}{\sum_k n_{k,j}} * log \frac{|D|}{|j : t_i \in d_j|}$$

We define our matched set of captions (images for detector based tfidf) to be our document, j and compute the tfidf score where $n_{i,j}$ represents the frequency of term i in



check out the face on the kid in the black hat he looks so enthused

The tower is the highest building in Hong Kong.

the water the boat was in

walking the dog in the primeval forest

shadows in the blue sky

water under the bridge

girl in a box that is a train

small dog in the grass

I tried to cross the street to get in my car but you can see that I failed LOL.

Figure 2.5: **Funny Results:** Some particularly funny or poetic results.

the matched set of captions (number of detections for detector based tfidf). The inverse document frequency is computed as the log of the number of documents $|D|$ divided by the number of documents containing the term i (documents with detections of type i for detector based tfidf).

2.4 Content Based Description Generation

For a query image, I_q, with global descriptor based matched images, $I_m \in M$, we want to re-rank the matched images according to the similarity of their content to the query. We perform this re-ranking individually for each of our content measures: object shape, object attributes, people actions, stuff classification, and scene type (Sec 2.3). We then combine these individual rankings into a final combined ranking in two ways. The first method trains a linear regression model of feature ranks against BLEU scores. The second method divides our training set into two classes, positive images consisting of the top 50% of the training set by BLEU score, and negative images from the bottom 50%.

A linear SVM is trained on this data with feature ranks as input. For both methods we perform 5 fold cross validation with a split of 400 training images and 100 test images to get average performance and standard deviation. For a novel query image, we return the captions from the top ranked image(s) as our result.

For an example matched caption like “The little boy sat in the grass with a ball”, several types of content will be used to score the goodness of the caption. This will be computed based on words in the caption for which we have trained content models. For example, for the word “ball” both the object shape and attributes will be used to compute the best similarity between a ball detection in the query image and a ball detection in the matched image. For the word “boy” an action descriptor will be used to compare the activity in which the boy is occupied between the query and the matched image. For the word “grass” stuff classifications will be used to compare detections between the query and the matched image. For each word in the caption tfidf overlap (sum of tfidf scores for the caption) is also used as well as detector based tfidf for those words referring to objects. In the event that multiple objects (or stuff, people or scenes) are mentioned in a matched image caption, the object (or stuff, people, or scene) based similarity measures will be a sum over the set of described terms. For the case where a matched image caption contains a word, but there is no corresponding detection in the query image, the similarity is not incorporated.

Results & Evaluation: Our content based captioning method often produces reasonable results (examples are shown in Fig 2.4). Usually results describe the main subject of the photograph (e.g. “Street dog in Lijiang”, “One monkey on the tree in the Ourika

Valley Morocco"). Sometimes they describe the depiction extremely well (e.g. "Strange cloud formation literally flowing through the sky like a river...", "Clock tower against the sky"). Sometimes we even produce good descriptions of attributes (e.g. "Tree with red leaves in the field in autumn"). Other captions can be quite poetic (Fig 2.5) – a picture of a derelict boat captioned "The water the boat was in", a picture of monstrous tree roots captioned "Walking the dog in the primeval forest". Other times the results are quite funny. A picture of a flimsy wooden structure says, "The tower is the highest building in Hong Kong". Once in awhile they are spookily apropos. A picture of a boy in a black bandana is described as "Check out the face on the kid in the black hat. He looks so enthused." – and he doesn't.

We also perform two quantitative evaluations. Several methods have been proposed to evaluate captioning (Kulkarni et al., 2013; Farhadi et al., 2010), including direct user ratings of relevance and BLEU score (Papineni et al., 2002). User rating tends to suffer from user variance as ratings are inherently subjective. The BLEU score on the other hand provides a simple objective measure based on n-gram precision. As noted in past work (Kulkarni et al., 2013), BLEU is perhaps not an ideal measure due to large variance in human descriptions (human-human BLEU scores hover around 0.5 (Kulkarni et al., 2013)). Nevertheless, we report it for comparison.

As can be seen in Table 2.1 data set size has a significant effect on BLEU score; more data provides more similar and relevant matched images (and captions). Local content matching also improves BLEU score somewhat over purely global matching.

In addition, we propose a new evaluation task where a user is presented with two

Method	BLEU
Global Matching (1k)	0.0774 +- 0.0059
Global Matching (10k)	0.0909 +- 0.0070
Global Matching (100k)	0.0917 +- 0.0101
Global Matching (1million)	0.1177 +- 0.0099
Global + Content Matching (linear regression)	0.1215 +- 0.0071
Global + Content Matching (linear SVM)	0.1259 +- 0.0060

Table 2.1: Automatic Evaluation: BLEU score measured at 1

photographs and one caption. The user must assign the caption to the most relevant image (care is taken to remove biases due to placement). For evaluation we use a query image and caption generated by our method. The other image in the evaluation task is selected at random from the web-collection. This provides an objective and useful measure to predict caption relevance. As a sanity check of our evaluation measure we also evaluate how well a user can discriminate between the original ground truth image that a caption was written about and a random image. We perform this evaluation on 100 images from our web-collection using Amazon's mechanical turk service, and find that users are able to select the ground truth image *96%* of the time. This demonstrates that the task is reasonable and that descriptions from our collection tend to be fairly visually specific and relevant. Considering the top retrieved caption produced by our final method – global plus local content matching with a linear SVM classifier – we find that users are able to select the correct image *66.7%* of the time. Because the top caption is not always visually relevant to the query image even when the method is capturing some information, we also perform an evaluation considering the top 4 captions produced by our method. In this case, the best caption out of the top 4 is correctly selected *92.7%* of the time. This demonstrates the strength of our content based method to produce relevant captions for images.

2.5 Discussion

We have described a caption generation method for general web images. This method relies on collecting and filtering a large data set of images from the internet to produce a novel web-scale captioned photo collection. We present two variations on our approach, one that uses only global image descriptors to retrieve captions, and one that incorporates estimates of image content for caption retrieval.

One problem with this approach is that a million image descriptions is still a limited number if the goal is to be able to represent a large number of novel complex images using these descriptions. We propose in Chapter 3 a way to describe parts of the image using text at the phrase level. This allows us more flexibility in the types of things that we can describe using our dataset and we also present a way to compose new descriptions using these phrases or pieces of text.

CHAPTER 3: SELECTING PHRASES THAT DESCRIBE IMAGES

In our previous chapter we focused on producing full image sentences given a query input image. This approach has the problem that it will be very difficult to find a sentence that can describe every new picture even with an enormous amount of data. We instead break down the problem into a smaller problem, that of finding descriptive short phrases that describe only a part or an aspect of an image. We can then use those short descriptive phrases to stitch them together to compose new sentences. One key aspect of this problem is making sure that the phrases have smooth transitions between each other. We use language models that use text statistics to encourage this type of consistency. This has parallels to data-driven approaches in other domains. For instance in texture synthesis previous research found that borrowing patches of pixels while maintaining consistency at the seams, as opposed to producing individual pixel models to synthesize new texture, produced better qualitative results (Liang et al., 2001; Efros and Freeman, 2001; Kwatra et al., 2003). In addition we also present an application for complex query image retrieval where the user can specify sentences to retrieve visually relevant images.

3.1 Retrieving and Reranking Phrases Describing Local Image Content

In this section we present methods to retrieve natural language phrases describing local and global image content from our large database of captioned photographs introduced in Chapter 2. Because we want to directly retrieve relevant phrases about objects, scene elements, etc, a large amount of image and text processing is first performed on the collected database (Sec 3.1.1) to extract useful and accurate estimates of local image content as well as the phrases that refer to that content. For a novel query image, we can then use image similarity measures to retrieve sets of visually relevant phrases describing image content (Sec 3.1.2). Finally, we use collective reranking methods to select the most relevant phrases for the query image (Sec 3.1.3). This work was originally described as part of (Ordonez et al., 2013b), and is closely related to the work in (Kuznetsova et al., 2012).

3.1.1 Dataset Processing

We perform four types of dataset processing: object detection, rough image parsing to obtain background elements, scene classification, and caption processing. This allows us to obtain textual phrases describing both local (e.g. objects and local object context) and global (e.g. general scene context) image content within our large data collection.

Object detection: We extract object category detections using deformable part models (Felzenszwalb et al., 2010) for 89 common object categories (Li et al., 2010; Ordonez et al., 2011). Here care must be taken because running tens or hundreds of object detectors on an image produces extremely noisy results (e.g., Fig 3.1). Instead, we place priors on image content – by only running detectors for objects (or their synonyms and hyponyms, e.g., Chihuahua for dog) mentioned in the caption associated with a database image. This produces *much cleaner results* than blindly running all object detectors.

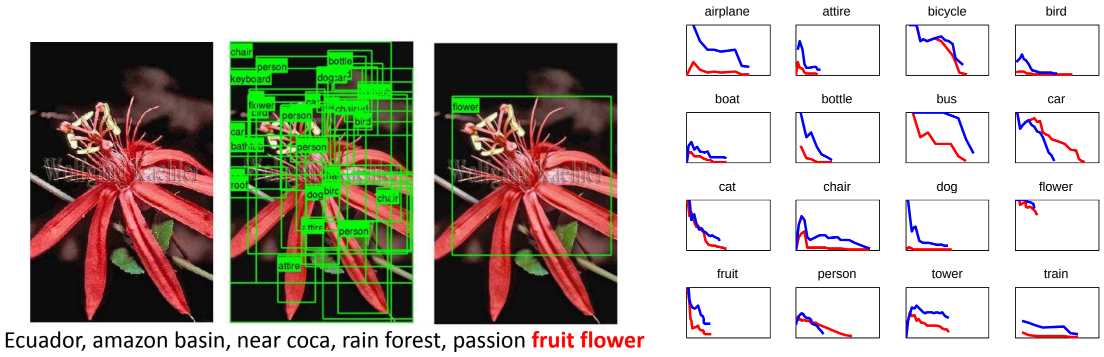


Figure 3.1: **Left:** Blindly running many object detectors on an image produces very noisy results. Running object detectors mentioned in a caption can produce much cleaner results. **Right:** Improvement in detection is measured with precision-recall (red shows raw detector performance, blue shows caption triggered). For some categories (e.g., airplane, dog) performance is greatly improved, for others not as much (e.g., cat, chair).

Figure 3.1 shows precision-recall curves for raw detectors in red and caption triggered detectors in blue for 1000 images from the SBU Dataset covering a balanced number of categories. We specifically collected bounding box annotations for this set of images to perform this evaluation. Detection is greatly improved for some categories (e.g., bus, airplane, dog), and less improved for others (e.g. cat, bicycle, person). From the million photo database we obtain a large pool of (up to 20k) highly confident object detections for each object category.

Image parsing: Image parsing is used to estimate regions of background elements in each database image. Six categories are considered: sky, water, grass, road, tree, and building, using detectors (Ordonez et al., 2011) which compute color, texton, HoG (Dalal and Triggs, 2005) and Geometric Context (Hoiem et al., 2005) as input features to a sliding window based SVM classifier. These detectors are run on all database images for retrieval.

Scene Classification: The scene descriptor for each image consists of the outputs of classifiers for 26 common scene categories. The features, classification method and training data are from the SUN dataset (Xiao et al., 2010). The descriptor is useful for capturing and matching overall global scene appearance for a wide range of scene types. Scene descriptors are computed on 700,000 images from the database to obtain a large pool of scene descriptors for retrieval.

Caption Parsing: The Berkeley PCFG parser (Petrov et al., 2006; Petrov and Klein, 2007) is used to obtain a hierarchical parse tree for each caption. From this tree we gather constituent phrases, (e.g., noun phrases, verb phrases, and prepositional phrases) referring to each of the above kinds of image content in the database.

3.1.2 Retrieving Phrases

For a query image, we retrieve several types of relevant phrases: noun-phrases (NPs), verb-phrases (VPs), and prepositional-phrases (PPs). Several different kinds of features measure visual similarity: **Color** – LAB histogram, **Texture** – histogram of vector quantized responses to a filter bank (Leung and Malik, 1999), **SIFT Shape** – histogram of vector quantized dense SIFT descriptors (Lowe, 2004), **HoG Shape** – histogram of vector quantized densely computed HoG descriptors (Dalal and Triggs, 2005), **Scene** – vector of classification scores for 26 common scene categories. The first 4 features are computed locally within a region of interest (object or stuff) and the last feature is computed globally.

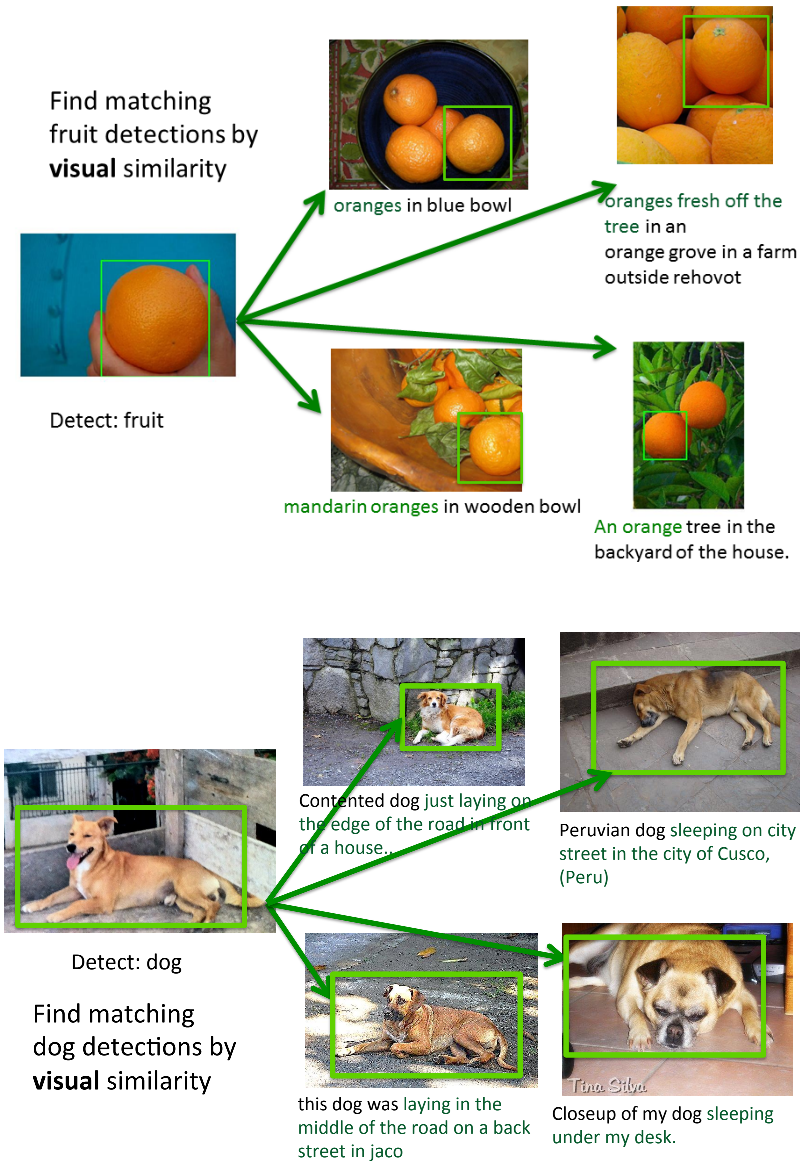

Find matching fruit detections by visual similarity
Detect: fruit
oranges in blue bowl
oranges fresh off the tree in an orange grove in a farm outside rehovot
mandarin oranges in wooden bowl
An orange tree in the backyard of the house.
Contented dog just laying on the edge of the road in front of a house..
Peruvian dog sleeping on city street in the city of Cusco, (Peru)
Detect: dog
Find matching dog detections by visual similarity
this dog was laying in the middle of the road on a back street in jaco
Tina Silva
Closeup of my dog sleeping under my desk.

Figure 3.2: **Top:** For a query "fruit" detection, we retrieve similar looking "fruit" detections (including synonyms or holonyms) from the database and transfer the referring noun-phrase (NP). **Bottom:** For a query "dog" detection, we retrieve similar looking "dog" detections (including synonyms or holonyms) from the database and transfer the referring verb-phrase (VP).

Retrieving Noun-Phrases (NPs): For each proposed object detection in a query image, we retrieve a set of relevant noun-phrases from the database. For example, if "fruit" is detected in the query, then we retrieve NPs from database image captions with

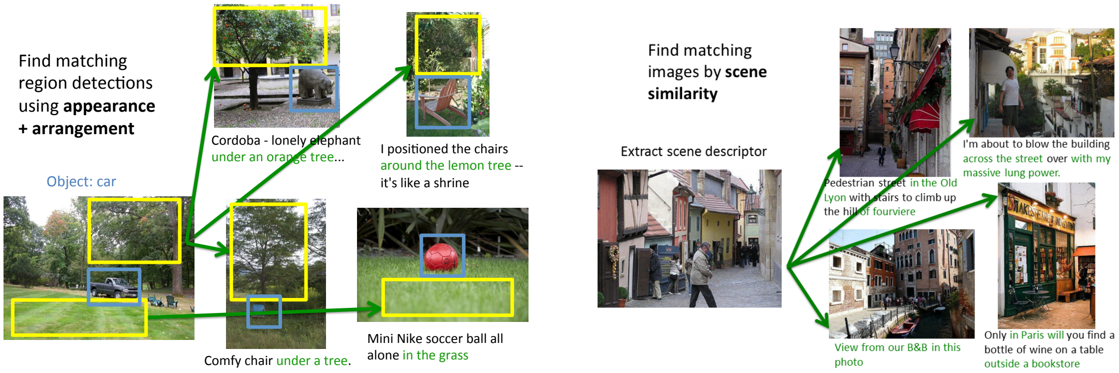

Find matching region detections using **appearance** + **arrangement**
Object: car
Cordoba - lonely elephant under an orange tree...
I positioned the chairs around the lemon tree -- it's like a shrine
Comfy chair under a tree.
Mini Nike soccer ball all alone in the grass
Find matching images by **scene similarity**
Extract scene descriptor
Pedestrian street in the Old Lyon with stairs to climb up the hill of fourviere
I'm about to blow the building across the street over with my massive lung power.
View from our B&B in this photo
Only in Paris will you find a bottle of wine on a table outside a bookstore

Figure 3.3: **Left:** For query object-stuff detection pairs, e.g.,"car" and "tree," we retrieve relevant object-stuff detections from the database using visual and geometric configuration similarity (where the database match can be e.g., "any object" and "tree" pair) and transfer the referring prepositional-phrase (PP). **Right:** We use whole image scene classification descriptors to transfer contextual scene prepositional-phrases (PPs).

visually similar "fruit" detections (including synonyms or holonyms, e.g. "apples" or "oranges"). This process is illustrated in Fig 3.2, left, where a query fruit detection is matched to visually similar database fruit detections (and their referring NPs in green). Visual similarity is computed as an unweighted combination of color, texton, SIFT, and HoG similarity, and produces visually similar and conceptually relevant NPs for a query object.

Retrieving Verb-Phrases (VPs): For each proposed object detection in a query image, we retrieve a set of relevant verb-phrases from the database. Here we associate VPs in database captions to object detections in their corresponding database images if the detection category (or a synonym or holonym) is the head word in an NP from the same sentence (e.g. in Fig 3.2 bottom right dog picture, "sleeping under my desk" is associated with the dog detection in that picture). Our measure of visual similarity is again based on equally weighted combination of color, texton, SIFT and HoG feature

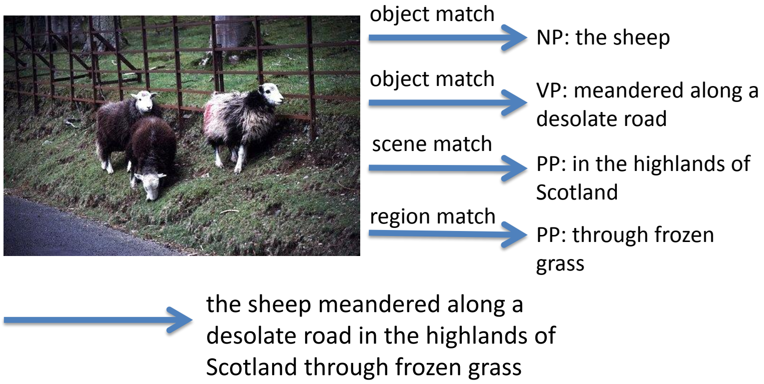

object match
NP: the sheep
object match
VP: meandered along a desolate road
scene match
PP: in the highlands of Scotland
region match
PP: through frozen grass
the sheep meandered along a desolate road in the highlands of Scotland through frozen grass

Figure 3.4: For a query image, we take a data-driven approach to retrieve (and optionally rerank) a set of visually relevant phrases based on local and global image content estimates. We can then construct an image caption for the query using phrasal description generation. Our optimization approach to generation maximizes both visual similarity and language-model estimates of sentence coherence. This produces captions that are more relevant, and human-sounding than previous approaches.

similarities. As demonstrated in Fig 3.2 (left), this measure often captures similarity in pose. Note that here we consider as our pool of objects only those instances that have VPs associated. This effectively changes the kind of similar matching objects that we find.

Retrieving Image parsing-based PPs: For each proposed object detection and for each background element detection in a query image, we retrieve relevant PPs according to visual and spatial relationship similarity (illustrated on the left in Fig 3.3 for car plus tree and grass detections). Visual similarity between a background query region and background database regions is computed based on color, texton, and SIFT co-sine similarity. Spatial relationship similarity is computed based on the similarity in geometric configuration between the query object-background pair and object-background pairs observed in the database (where the object in the database pairs need not be the same

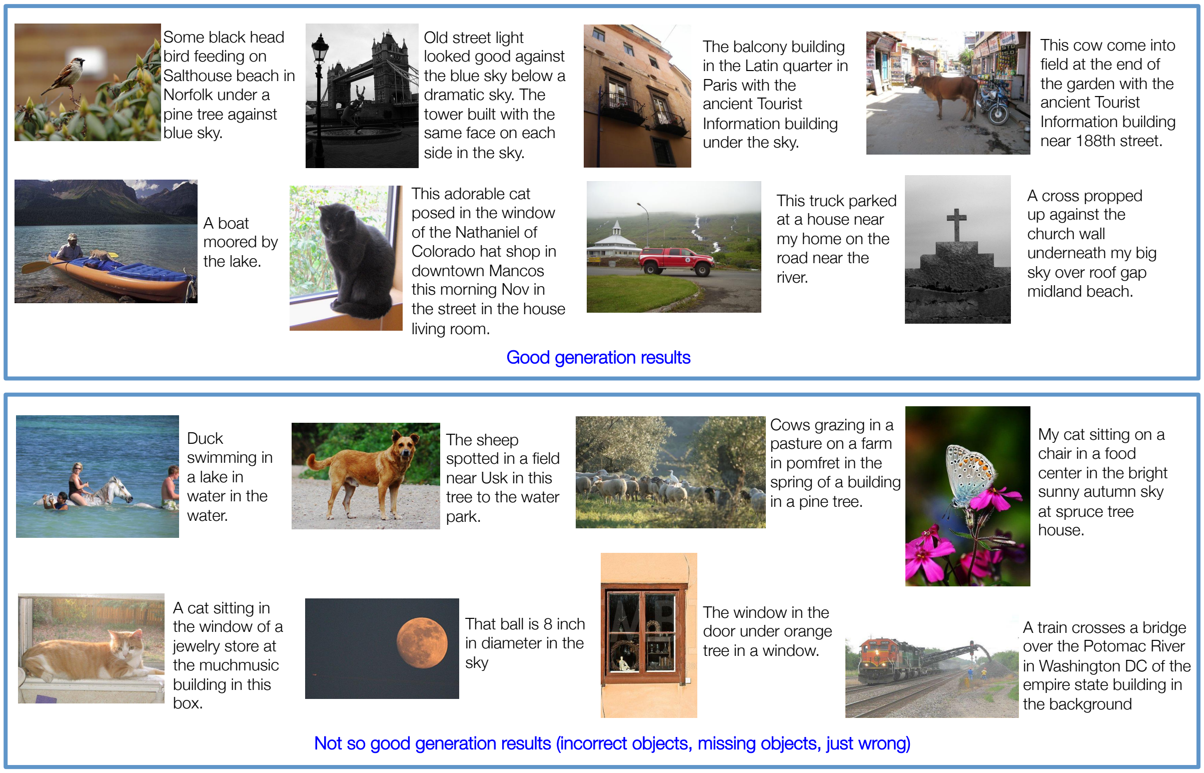

Some black head bird feeding on Salthouse beach in Norfolk under a pine tree against blue sky.
Old street light looked good against the blue sky below a dramatic sky. The tower built with the same face on each side in the sky.
The balcony building in the Latin quarter in Paris with the ancient Tourist Information building under the sky.
This cow come into field at the end of the garden with the ancient Tourist Information building near 188th street.
A boat moored by the lake.
This adorable cat posed in the window of the Nathaniel of Colorado hat shop in downtown Mancos this morning Nov in the street in the house living room.
This truck parked at a house near my home on the road near the river.
A cross propped up against the church wall underneath my big sky over roof gap midland beach.
Good generation results
Duck swimming in a lake in water in the water.
The sheep spotted in a field near Usk in this tree to the water park.
Cows grazing in a pasture on a farm in pomfret in the spring of a building in a pine tree.
My cat sitting on a chair in a food center in the bright sunny autumn sky at spruce tree house.
A cat sitting in the window of a jewelry store at the muchmusic building in this box.
That ball is 8 inch in diameter in the sky
The window in the door under orange tree in a window.
A train crosses a bridge over the Potomac River in Washington DC of the empire state building in the background
Not so good generation results (incorrect objects, missing objects, just wrong)

Figure 3.5: Using our retrieved, reranked phrases for description generation (Sec 3.2.1). Reasonably good results are shown on top and less good results (with incorrect objects, missing objects, or just plain wrong descriptions) are shown on right.

object as the query). This spatial relationship is measured in terms of the normalized distance between the foreground object and the background region, the normalized overlap area between the foreground object and the background region, and the absolute vertical position of the foreground object. Visual similarity and geometric similarity measures are given equal weights and produce appealing results (Fig 3.3).

Retrieving Scene-based PPs: For a query image, we retrieve PPs referring to the overall setting or scene by finding the most similar global scene descriptors from the database. Here we retrieve the last PP in a sentence since it is most likely to describe the scene content. As shown on the right in Fig 3.3, useful matched phrases often correspond to places (e.g., “in Paris”) or general scene context (e.g., “under water”).

3.1.3 Reranking Phrases

Given a set of phrases retrieved independently for a query image, we would like to rerank these phrases using collective measures computed on the entire set of retrieved results. Related reranking strategies have been used for other retrieval systems. (Sivic and Zisserman, 2003) retrieve images using visual words and then rerank them based on a measure of geometry and spatial consistency. (Torralba et al., 2008) retrieve a set of images using a reduced representation of their feature space and then perform a second refined reranking phase on top matching images to produce exact neighbors.

In our case, instead of reranking images, our goal is to rerank retrieved phrases such that the relevance of the top retrieved phrases is increased. Because each phrase is retrieved independently in the phrase retrieval step, the results tend to be quite noisy. Spurious image matches can easily produce irrelevant phrases. The wide variety of Flickr users and contexts under which they capture their photos can also produce unusual or irrelevant phrases.

As an intuitive example, if one retrieved phrase describes a dog as “the brown dog” then the dog *may* be brown. However, if several retrieved phrases describe the dog in similar ways, e.g., “the little brown dog”, “my brownish pup”, “a brown and white mutt”, then it is much more likely that the query dog is brown and the relevance for phrases describing brown attributes should be increased.

In particular, for each type of retrieved phrase (see Sec 3.1.2), we gather the top 100

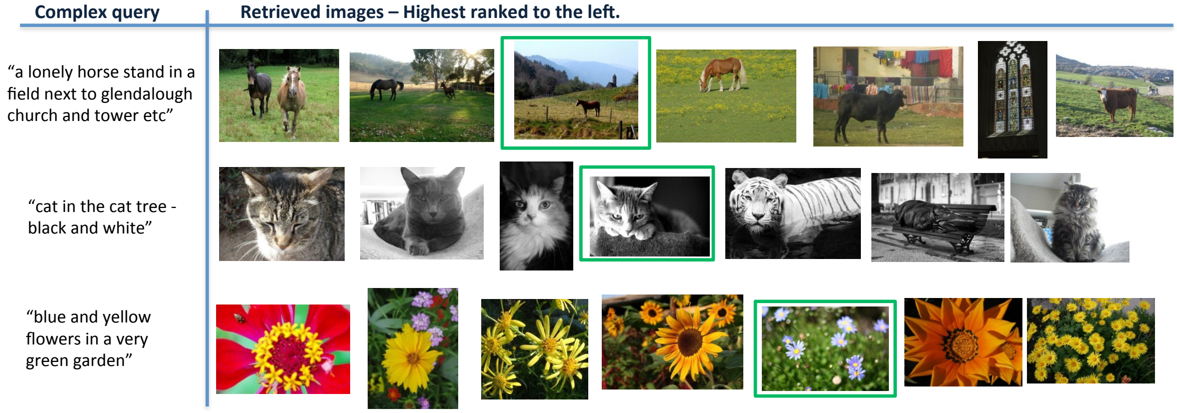

Complex query
Retrieved images – Highest ranked to the left.
"a lonely horse stand in a field next to glendalough church and tower etc"
"cat in the cat tree - black and white"
"blue and yellow flowers in a very green garden"

Figure 3.6: Complex query image retrieval. For a complex natural language text query (left), we retrieve images displaying relevant content (right). The image originally associated with the complex text query is highlighted in green.

best matches based on visual similarity. Then, we perform phrase reranking to select the best and most relevant phrases for an image (or part of an image in the case of objects or regions). We evaluate two possible methods for reranking: 1) PageRank based reranking using visual and/or text similarity, 2) Phrase-level TFIDF based reranking.

PageRank Reranking

PageRank (Brin and Page, 1998) computes a measure for the relative importance of items within a set based on the random walk probability of visiting each item. The algorithm was originally proposed as a measure of importance for web pages using hyperlinks as connections between pages (Brin and Page, 1998), but has also been applied to other tasks such as reranking images for product search (Jing and Baluja, 2008). For our task, we use PageRank to compute the relative importance of phrases within a retrieved set on the premise that phrases displaying strong similarity to other phrases within the retrieved set are more likely to be relevant to the query image.

We construct four graphs, one for each type of retrieved phrase (NP, VP, PPStuff, or PPScene), from the set of retrieved phrases for that type. Nodes in these graphs correspond to retrieved phrases (and the corresponding object, region, or image each phrase described in the SBU database). Edges between nodes are weighted using visual similarity, textual similarity, or an unweighted combination of the two – denoted as Visual PageRank, Text PageRank, or Visual + Text PageRank respectively. Text similarity is computed as the cosine similarity between phrases, where phrases are represented as a bag of words with a vocabulary size of approximately 100k words, weighted by term-frequency inverse-document frequency (TFIDF) score (Roelleke and Wang, 2008). Here IDF measures are computed for each phrase type independently rather than over the entire corpus of phrases to produce IDF measures that are more type specific. Visual similarity is computed as cosine similarity of the visual representations used for retrieval (Sec 3.1.2).

For generating complete image descriptions (Sec 3.2.1), the PageRank score can be directly used as a unary potential for phrase confidence.

Phrase-level TFIDF Reranking

We would like to produce phrases for an image that are not only relevant, but specific to the depicted image content. For example, if we have a picture of a cow, a phrase like “the cow” is always going to be relevant to any picture of a cow. However, if the cow is mottled with black and white patches then “the spotted cow” is a much better description for the particular example. If both of these phrases are retrieved for the image, then we would prefer to select the second one over the first.

To produce phrases with high description specificity, we define a phrase-level measure of TFIDF. This measure rewards phrases containing words that occur frequently within the retrieved phrase set, but infrequently within a larger set of phrases – therefore giving higher weight to phrases that are specific to the query image content (e.g., "spotted"). For object and stuff region related phrases (NPs, VPs, PPStuff), IDF is computed over phrases referring to that object or stuff category (e.g., the frequency of words occurring in a noun phrase with "cow" in the example above). For whole image related phrases (PPScene), IDF is computed over all prepositional phrases. To compute TFIDF for a phrase, the TFIDF for each word in the phrase is calculated (after removing stop words) and then averaged. Other work that has used TFIDF for image features (we use it for text associated with an image) include (Sivic and Zisserman, 2003), (Chum et al., 2008) and (Ordonez et al., 2011).

For composing image descriptions (Sec 3.2.1), we use phrase-level TFIDF to rerank phrases and select the top ten phrases. The original visual retrieval score (Sec 3.1.2) is used as phrase confidence score, effectively merging ideas of visual relevance with phrase specificity (denoted as Visual + TFIDF).

3.2 Applications of Phrases

Once we have retrieved (and reranked) phrases related to an image we can use the associated phrases in a number of applications. Here we demonstrate two potential applications: phrasal generation of image descriptions (Sec 3.2.1), and complex query image search (Sec 3.2.2).

3.2.1 Phrasal Generation of Image Descriptions

We model caption generation as an optimization problem in order to incorporate two different types of information: the confidence score of each retrieved phrase provided by the original retrieval algorithm (Sec 3.1.2) or by our reranking techniques (Sec 3.1.3), and additional pairwise compatibility scores across phrases computed using observed language statistics. Our objective is to select a set of phrases that are visually relevant to the image and that together form a reasonable sentence, which we measure by compatibility across phrase boundaries.

Let $X = \{x_{obj},\ x_{verb},\ x_{stuff},\ x_{scene}\}$ be a candidate set of phrases selected for caption generation. We maximize the following objective over possibilities for X:

$$E(X) = \Phi(X) + \Psi(X), \tag{3.1}$$

where $\Phi(X)$ aggregates the unary potentials measuring quality of the individual phrases:

$$\Phi(X) = \phi(x_{obj}) + \phi(x_{verb}) + \phi(x_{stuff}) + \phi(x_{scene}), \tag{3.2}$$

and $\Psi(X)$ aggregates binary potentials measuring pairwise compatibility between phrases:

$$\Psi(X) = \psi(x_{obj}, x_{verb}) + \psi(x_{verb}, x_{stuff}) + \psi(x_{stuff}, x_{scene}). \tag{3.3}$$

Unary potentials, $\phi(x)$, are computed as the confidence score of phrase x determined by the retrieval and reranking techniques discussed in Sec 3.1.3. To make scores across different types of phrases comparable, we normalize them using Z-score (subtract mean and divide by standard deviation). We further transform the scores so that they fall in the [0,1] range.

Binary potentials: N-gram statistics are used to compute language naturalness – a frequent n-gram denotes a commonly used, "natural", sequence of words. In particular, we use n-gram frequencies provided by the Google Web 1-T dataset (Brants and Franz., 2006), which includes frequences up to 5-grams with counts computed from text on the web. We use these counts in the form of normalized point-wise mutual information scores to incorporate language-driven compatibility scores across different types of retrieved phrases. The compatibility score $\psi(x_i, x_j)$ between a pair of adjacent phrases x_i and x_j is defined as follows: $\psi(x_i, x_j) = \alpha \cdot \psi^{\mathrm{L}}_{ij} + (1 - \alpha) \cdot \psi^{\mathrm{G}}_{ij}$. Where ψ^{L}_{ij} and ψ^{G}_{ij} are the local and the global cohesion scores defined below.[1]

Local Cohesion Score: Let L_{ij} be the set of all possible n-grams ($2 \leq n \leq 5$) across the boundary of x_i and x_j. Then we define the n-gram local cohesion score as:

$$\psi^{\mathrm{L}}_{ij} = \frac{\sum\limits_{l \in L_{ij}} \mathrm{NPMI}(l)}{\|L_{ij}\|}, \tag{3.4}$$

where $\mathrm{NPMI}(v) = (\mathrm{PMI}(v) - a)/(b - a)$ is a normalized point-wise mutual information (PMI) score where a and b are normalizing constants computed across n-grams so that the range of $\mathrm{NPMI}(v)$ is between 0 and 1. This term encourages smooth transitions between consecutive phrases. For instance the phrase "The kid on the chair" will fit

[1] The coefficient α can be tuned via grid search, and scores are normalized $\in [0, 1]$.

better preceding “sits waiting for his meal” than “sleeps comfortably”. This is because the words at the end of the first phrase including “chair” are more compatible with the word“sit” at the beginning of the second phrase than with the word “sleep” at the beginnining of the third phrase.

Global Cohesion Score: These local scores alone are not sufficient to capture semantic cohesion across very long phrases, because Google n-gram statistics are limited to 5 word sequences. Therefore, we also consider compatibility scores between the head word of each phrase, where the head word corresponds semantically to the most important word in a given phrase (last word or main verb of the phrase). For instance the phrase “The phone in the hall” is more compatible with the phrase “rings loudly all the time” than with the phrase “thinks about philosophy everyday” because the head word “phone” is more compatible with the head word “rings” than with the head word “thinks”. Let h_i and h_j be the head words of phrases x_i and x_i respectively, and let $f_\Sigma(h_i, h_j)$ be the total frequency of all n-grams that start with h_i and end with h_j. Then the global cohesion is computed as:

$$\psi_{ij}^{\mathrm{G}} = \frac{f_\Sigma(h_i, h_j) - \min(f_\Sigma)}{\max(f_\Sigma) - \min(f_\Sigma)}. \tag{3.5}$$

Inference by Viterbi decoding: Notice that the potential functions in the objective function (Equations 3.1 & 3.3) have a linear chain structure. Therefore, we can find the argmax, $X = \{x_{obj}, x_{verb},\, x_{stuff}, x_{scene}\}$, efficiently using Viterbi decoding.[2]

[2] An interesting but non-trivial extension to this generation technique is allowing re-ordering or omission of phrases (Kuznetsova et al., 2012).

3.2.2 Complex Query Image Search

Image retrieval is beginning to work well. Commercial companies like Google and Bing produce quite reasonable results now for simple image search queries, like "dog" or "red car". Where image search still has much room for improvement is for complex search queries involving appearance attributes, actions, multiple objects with spatial relationships, or interactions. This is especially true for more unusual situations, that cannot be mined directly by looking at the meta-data and text surrounding an image, *e.g.*, "little boy eating his brussels sprouts".

We demonstrate a prototype application, showing that our approach for finding descriptive phrases for an image can be used to form features that are useful for complex query image retrieval. We use 1000 test images (described in Sec 3.3) as a dataset. For each image, we pick the top selected phrases from the vision+text PageRank algorithm to use as a complex text descriptor for that image – note that the actual human-written caption for the image is not seen by the system. For evaluation we then use the original human caption for an image as a complex query string. We compare it to each of the automatically derived phrases for images in the dataset and score the matches using normalized correlation. We then sort the scores and record the rank of the correct image – the one for which the query caption was written. If the retrieved phrases matched the actual human captions well, then we expect the query image to be returned first in the retrieved images. Otherwise, it will be returned later in the ranking. Note that this is only a demo application performed on a very small dataset of images. A real image

Method	Noun Phrases $K = 1, 5, 10$	Verb Phrases $K = 1, 5, 10$	Prepositional Phrases(stuff) $K = 1, 5, 10$	Prepositional Phrases(scenes) $K = 1, 5, 10$
No reranking	$0.24, 0.24, 0.23$	$0.15, 0.14, 0.14$	$0.30, 0.29, 0.27$	$0.28, 0.26, 0.25$
Visual PageRank	$0.23, 0.23, 0.23$	$0.13, 0.14, 0.14$	$0.28, 0.28, 0.27$	$0.26, 0.25, 0.25$
Text PageRank	$0.30, 0.29, 0.28$	$0.20, 0.19, 0.17$	$0.38, 0.37, 0.36$	$0.34, 0.30, 0.27$
Visual+Text PageRank	$0.28, 0.27, 0.26$	$0.17, 0.17, 0.16$	$0.32, 0.30, 0.28$	$0.27, 0.28, 0.27$
TFIDF Reranking	$0.29, 0.28, 0.27$	$0.19, 0.19, 0.18$	$0.38, 0.37, 0.36$	$0.40, 0.36, 0.32$

Table 3.1: Average BLEU score for the top K retrieved phrases against Flickr captions.

Method	Noun Phrases	Verb Phrases	Prepositional Phrases(stuff)	Prepositional phrases(scenes)
No reranking	0.2633	0.0759	0.1458	0.1275
Visual PageRank	0.2644	0.0754	0.1432	0.1214
Text PageRank	0.3286	0.1027	0.1862	0.1642
Visual + Text PageRank	0.2262	0.0938	0.1536	0.1631
TFIDF Reranking	0.3143	0.1040	0.2096	0.1912

Table 3.2: Average BLEU score evaluation K=10 against MTurk written descriptions.

retrieval application would have access to billions of images.

3.3 Evaluation

We perform a thorough experimental evaluation on our phrase retrieval and reranking (Sec 3.3.1), phrase based description generation (Sec 3.3.2), and phrase based complex query image search (Sec 3.3.3).

For all phrase based evaluations (except where explicitly noted) we use a test set of 1000 query images, selected to have high detector confidence scores. Random test images could also be sampled, but for images with poor detector performance we expect the results to be much the same as for our baseline global generation methods. Therefore, we focus here on evaluating performance for images where detection is more likely to have produced reasonable estimates of local image content.

3.3.1 Phrase Retrieval & Reranking Evaluation

We calculate BLEU scores (without length penalty) for evaluating the retrieved phrases against the original human associated captions from the SBU Dataset (Ordonez et al., 2011). Scores are evaluated for the top K phrases for $K = 1, 5, 10$ for each phrase type in Table 3.1. We can see that except for Visual PageRank all other reranking strategies yield better BLEU scores than the original (unranked) retrieved phrases. Overall, Text PageRank and TFIDF Reranking provide the best scores.

One possible weakness in this initial evaluation is that we use single caption as reference – the captions provided by the owners of the photos – which often include contextual information unrelated to visual content. To alleviate this effect we further collect 4 additional human written descriptions using Amazon Mechanical Turk for a subset of 200 images from our test set (care was taken to ensure workers were located in the US and filtered for quality control). In this way we obtain good quality sentences referring to the image content, but we also notice some biases like rich noun-phrases while very few verb-phrases within those sentences. Results are provided in Table 3.2, further supporting our previous observations (TFIDF and Text PageRank demonstrate the most increase in BLEU score performance over the original retrieved ranking).

Method	No Reranking	Visual PageRank	Text PageRank	Visual + Text PageRank	Visual + TFIDF Rerank
BLEU	0.1192	0.1133	0.1257	0.1224	**0.1260**
ROUGE	0.2300	0.2236	0.2248	**0.2470**	0.2175

Table 3.3: BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) score evaluation of full image captions generated using HMM decoding with our strategies for phrase retrieval and reranking.

Method	Percentage
Text PageRank **vs.** No Reranking	54%/46%
Visual + Text PageRank **vs** No Reranking	57%/43%
Visual + TFIDF Reranking **vs** No Reranking	61%/39%
Text + Visual PageRank **vs** Visual + TFIDF Reranking	49%/51%
Text + Visual PageRank **vs** Global Description Generation	71%/29%

Table 3.4: Human forced-choice evaluation between various methods.

3.3.2 Application 1: Description Generation Evaluation

We can also evaluate the quality of our retrieved set of phrases indirectly by using them in an application to compose novel full image descriptions (Sec 3.2.1). Automatic evaluation is computed again using BLEU score (Papineni et al., 2002) (including length penalty), and we additionally compute ROUGE scores (Lin, 2004) (analog to BLEU scores, ROUGE scores are a measure of recall that is also used in machine translation problems). The original associated captions from Flickr are used as reference descriptions. Table 3.3 shows results. All of our reranking strategies except visual PageRank outperform the original image based retrieval on the generation task in terms of BLEU score and Visual plus Text PageRank reranking outperforms on ROUGE. For BLEU, the best reranking method is found to be Visual similarity plus TFIDF reranking. For ROUGE, the best reranking strategy is Visual + Text PageRank.

Further, we also perform human judgment forced choice tasks on Amazon Mechanical Turk. Here users are presented with an image and two captions (each generated by a different method) and they must select the caption which better describes the image. Presentation order is randomized to remove user bias. Table 3.4 shows results. The top 3 rows show our methods are preferred over unranked phrases. Row 4 shows our top 2 methods are comparable. Finally, row 5 shows one of our methods is strongly preferred over the whole sentence baseline provided with the SBU dataset (Ordonez et al., 2011). We also show some qualitative results in Fig. 3.5 showing successful cases of generated captions and different failure cases (due to incorrect objects, missing objects, incorrect grammar or semantic inconsistencies) for our top performing method.

3.3.3 Application 2: Complex Query Image Retrieval Evaluation

We tested retrieval using 200 captions from the dataset described in Sec. 3.2.2 as queries. For 3 queries, the corresponding image was ranked first by our retrieval system. For these images the automatically selected phrases described the images so well that they matched the ground truth captions better than the phrases selected for any of the other 999 images. Overall 20% of queries had the corresponding image in the top 1% of the ranked results (top 10 ranked images), 30% had the corresponding image in the top 2%, and 43% had the corresponding image in the top 5% of ranked retrievals. In addition to being able to find the image described out of a set of 1000, the retrieval system produced reasonable matches for the captions as shown in Fig. 3.6.

3.4 Discussion

We have explored several methods for collective reranking of sets of phrases and demonstrated the results in two applications, phrase based generation of image descriptions and complex query image retrieval. Finally, we have presented a thorough evaluation of each of our presented methods through both automatic and human-judgment based measures.

Generating generic image descriptions that resemble those written by people remains a challenging problem. There have been several other proposed methods to generate descriptions since then but one less studied problem is that of task-specific descriptions. We present in Chapter 4 a study on a particular type of such descriptions known as *Referring Expressions*.

CHAPTER 4: REFERRING EXPRESSIONS FOR OBJECTS IN NATURAL SCENES

One important aspect of describing objects in natural scenes using language is deciding how to refer to such objects. For unfamiliar objects, this involves deciding what is the object name, and the set of attributes, properties, and relations that should be mentioned in a noun-phrase to identify a target or *referent* object. These type of noun phrases are called Referring Expressions. From human robot interactions, to image search, to situated language learning, and natural language grounding, there are a number of research areas that would benefit from a better understanding of how people refer to physical entities in the world.

In the previous chapters we focused on generating general image descriptions. One challenge with evaluating these types of systems is that automatic evaluation metrics like BLEU or ROUGE were designed for other tasks (Machine Translation and Text Summarization) and might not correlate well with human judgments on the image description problem (Hodosh et al., 2013; Elliott and Keller, 2014). Referring Expressions are tied to a task so we can provide a more objective evaluation compared to general image descriptions. First, they should be able to identify the referent object from its context, a person should be able to use the expression to easily find the referent object in a given image. This is a rather objective way to verify the validity of the expression. Second, an automatically generated referring expression should resemble in its mentioned attributes, properties, and relations, the type of choices that people would make. This set of choices is considerably more constrained compared to the space of possible things that could be mentioned in general image descriptions.

In the same spirit as the previous chapters, one can devise a computational recognition system that can identify all the attributes for a given object in an image. But people do not mention all attributes of an object exhaustively when trying to identify the object using referring expressions. We present in this chapter an analysis of the types of attributes that people prefer to mention for different types of objects and the individual set of words that are preferred for each attribute. Finally, we devise a technique to generate human-like referring expressions for a given input image with partial annotations so that it resembles the kind of expressions that people would use. This work was originally published in (Kazemzadeh, Ordonez et al., 2014).

4.1 Introduction

Recent advances in automatic computer vision methods have started to make technologies for recognizing thousands of object categories a near reality (Perronnin et al., 2012; Deng et al., 2012a, 2010; Krizhevsky et al., 2012). As a result, there has been a spurt of recent work trying to estimate higher level semantics, including exciting efforts to automatically produce natural language descriptions of images and video (Farhadi et al., 2010; Kulkarni et al., 2013; Yang et al., 2011; Ordonez et al., 2011; Kuznetsova et al., 2012; Feng and Lapata, 2013). Common challenges encountered in these pursuits include the fact that descriptions can be highly task dependent, open-ended, and difficult to evaluate automatically.

Previous work on REG has made significant progress toward understanding how people generate expressions to refer to objects (a recent survey of techniques is provided in (Krahmer and van Deemter, 2012)). In this chapter, we study the relatively unexplored setting of how people refer to objects in *complex photographs of real-world cluttered scenes*. One initial stumbling block to examining this scenario is lack of existing relevant datasets, as previous collections for studying REG have used relatively focused domains such as graphics generated objects (van Deemter et al., 2006; Viethen and Dale, 2008), crafts (Mitchell et al., 2010), or small everyday (home and office) objects arrayed on a simple background (Mitchell et al., 2013a; FitzGerald et al., 2013).

We present here a new large-scale corpus, currently containing 130,525 expressions, referring to 96,654 distinct objects, in 19,894 photographs of real world scenes. Some examples from our dataset are shown in Figure 4.5. To construct this corpus efficiently, we design a new two player referring expression game (ReferItGame) to crowd-source the data collection. Popularized by efforts like the ESP game (von Ahn and Dabbish, 2004) and Peekaboom (von Ahn et al., 2006b), Human Computation based games can be an effective way to engage users and collect large amounts of data inexpensively. Two player games can also automate verification of human provided annotations.

Our resulting corpus is both more real-world and much bigger than previous datasets, allowing us to examine referring expression generation in a new setting at large scale. To understand and quantify this new dataset, we perform an extensive set of analyses. One significant difference from previous work is that we study how referring expressions vary for different categories. We find that an object's category greatly influences the types of attributes used in their referring expression (e.g. people use color words to describe cars more often than mountains). Additionally, we find that references to an object are sometimes made with respect to other nearby objects, e.g. "the ball to left of the man". Interestingly, the types of reference objects (i.e. "the man") used in referring expressions are also biased toward some categories. Finally, we find that the word used to refer to the object category itself displays consistencies across people. This notion is related to ideas of entry-level categories from Psychology (Rosch, 1978). We explore this problem in more detail in Chapter 5.

Given these findings, we propose an optimization model for generating referring expressions that jointly selects which attributes to include in the expression, and what attribute values to generate. This model incorporates both visual models for selecting attribute-values and object category specific priors. Experimental evaluations indicate that our proposed model produces reasonable results for REG.

In summary, contributions include:

- A two player online game to collect and verify natural language referring expressions.

- A new large-scale dataset containing natural language expressions referring to objects in photographs of real world scenes.

- Analyses of the collected dataset, including studying category-specific variations in referring expressions.

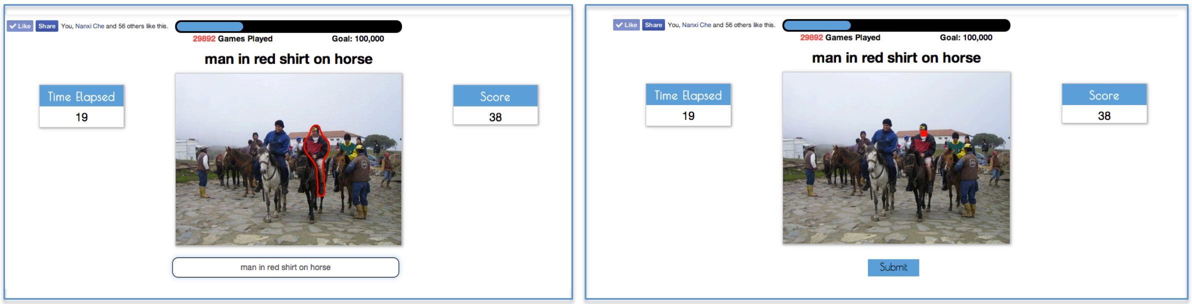

Like
Share
You, Nanxi Che and 56 others like this.
29892 Games Played
Goal: 100,000
man in red shirt on horse
Time Elapsed
19
Score
38
man in red shirt on horse
Like
Share
You, Nanxi Che and 56 others like this.
29892 Games Played
Goal: 100,000
man in red shirt on horse
Time Elapsed
19
Score
38
Submit

Figure 4.1: An example game. Player 1 (*left*) sees an image with an object outlined in red (the man) and provides a referring expression for the object ("man in red shirt on horse"). Player 2 (*right*) sees the image and the expression from Player 1 and must localize the correct object by clicking on it (click indicated by the red square). Elapsed time and current scores are also provided.

- An optimization based model to generate referring expressions for objects in real-world scenes with experimental evaluations on three labeled test sets.

The rest of this chapter is organized as follows. First we outline related work from the vision and language communities (§4.2). Then we describe our online game for collecting referring expressions (§4.3) and provide an analysis of our new ReferItGame Dataset (§4.4). Finally, we present and evaluate our model for generating referring expressions (§4.5) and discuss conclusions and future work (§4.6).

4.2 Related Work

Referring Expression Generation: There has been a long history of research on understanding how people generate referring expressions, dating back to the 1970s (Winograd, 1972). One common approach is the Incremental Algorithm (Dale and Reiter, 1995, 2000) which uses logical expressions for generation. Much work in REG follows the Gricean maxims (Grice, 1975) which provide principles for how people will behave in conversation. These include four general principles: The principle of quantity which dictates including only the minimum needed amount of information, the principle of quality which dictates that we would only include truthful information, the principle of relation which proposes including relevant information, and the principle of manner which proposes avoiding ambiguity and obscurity.

Recently, there has been progress examining other aspects of the referring expression problem such as understanding what types of attributes are used (Mitchell et al., 2013a), modeling variations between speakers (Viethen and Dale, 2010; Viethen et al., 2013; Van Deemter et al., 2012; Mitchell et al., 2013b), incorporating visual classifiers (Mitchell et al., 2011), producing algorithms to refer to object sets (Ren et al., 2010; FitzGerald et al., 2013), or examining impoverished perception REG (Fang et al., 2013). A good survey of work in this area is provided in (Krahmer and van Deemter, 2012). We build on past work, extending models to generate attributes jointly in a category specific framework.

Referring Expression Datasets: Some initial datasets in REG used graphics engines to produce images of objects (van Deemter et al., 2006; Viethen and Dale, 2008). Recently more realistic datasets have been introduced, consisting of craft objects like pipecleaners, ribbons, and feathers (Mitchell et al., 2010), or everyday home and office objects such as staplers, combs, or rulers (Mitchell et al., 2013a), arrayed on a simple background. These datasets helped to move referring expression generation research into the domain of real world objects. We seek to further these pursuits by constructing a dataset of natural objects in photographs of the real world.

Image & Video Description Generation: Recent research on automatic image description has followed two main directions. Retrieval based methods (Aker and Gaizauskas, 2010b; Farhadi et al., 2010; Ordonez et al., 2011; Feng and Lapata, 2010, 2013) retrieve existing captions or phrases to describe a query image. Bottom up methods (Kulkarni et al., 2013; Yang et al., 2011; Yao et al., 2010) rely on visual classifiers to first recognize image content and then construct captions from scratch, perhaps with some input from natural language statistics. Very recently, these ideas have been extended to produce descriptions for videos (Guadarrama et al., 2013; Barbu et al., 2012). Like these methods, we generate descriptions for natural scenes, but focus on referring to particular objects rather than providing an overall description of an image or video.

Human Computation Games: Games can be a useful tool for collecting large amounts of labeled data quickly. Human Computation Games were first introduced by Luis von Ahn in the ESP game (von Ahn and Dabbish, 2004) for image labeling, and later extended to segment objects (von Ahn et al., 2006b), collect common-sense knowledge (von Ahn et al., 2006a), or disambiguate words (Seemakurty et al., 2010). Recently, crowd games have also been introduced into the computer vision community for tasks like fine grained category recognition (Deng et al., 2013). These games can be released publicly on the web or used on Mechanical Turk to enhance and encourage *turker* (users of Mechanical Turk) participation (Deng et al., 2013). Inspired by the success of previous games, we create a game to collect and verify natural language expressions referring to objects in natural scenes.

4.3 Referring Expression Game (ReferItGame)

In this section we describe our referring expression game (ReferItGame[1]), a simple two player game where players alternate between generating expressions referring to objects in images of natural scenes, and clicking on the locations of described objects. An example game is shown in Figure 4.1.

4.3.1 Game Play

Player 1: is shown an image with an object outlined in red and provided with a text box in which to write a referring expression. *Player 2:* is shown the same image and the referring expression written by Player 1 and must click on the location of the described object (note, Player 2 does not see the object segmentation). If Player 2 clicks on the correct object, then both players receive game points and the Player 1 and Player 2 roles swap for the next image. If Player 2 does not click on the correct object then no points are received and the players remain in their current roles.

This provides us with referring expressions for our dataset and verification that the expressions are valid since they led to correct object localizations. Expressions written for games where the object was not correctly localized are kept and released with the dataset for future study, but are not included in our final dataset analyses or statistics. A game timer encourages players to write expressions quickly, resulting in more natural expressions. Also, IP addresses are filtered to prevent people from simultaneously playing both roles.

[1] Available online at http://referitgame.com

4.3.2 Playing Against the Computer

To promote engagement, we implement a single player version of the game. When a player connects, if there is another player online then the two people are paired. If there are currently no other available players, then the person plays a “canned” game against the computer. If at any point another person connects, the canned game ends and the player is paired with the new person.

To implement canned games we seed the game with 5000 pre-recorded referring expression games (5 referring expressions and resulting clicks for each of 1000 objects) collected using Amazon’s Mechanical Turk service. Implementing an automated version of Player 1 is simple; we just show the person one of the pre-collected referring expressions and they click as usual.

Automating the role of Player 2 is a bit more complicated. In this case, we compare the person’s written expression against the pre-recorded expressions for the same object. For this comparison we use a parser to lemmatize the words in an expression and then compute cosine similarity between expressions with a bag of words representation. Based on this measure the closest matching expression is determined. If there is no similarity between the newly generated expression and the canned expressions, the expression is deemed incorrect and a random click location (outside of the object) is generated. If there is a successful match with a previously generated expression, then the canned click from the most similar pre-recorded game is used. More complex similarities could be used, but since we require real-time performance in our game setting we use this simple implementation which works well for our expressions.

4.4 ReferItGame Dataset

In this section we describe the ReferItGame dataset[2], including images and labels, processing the dataset, and analysis of the collection.

4.4.1 Images and Labels

We build our dataset of referring expressions on top of the ImageCLEF IAPR image retrieval dataset (Grubinger et al., 2006). This dataset is a collection of 20,000 images available free of charge without copyright restrictions, depicting a variety of aspects of everyday life, from sports, to animals, to cities, and landscapes. Crucial for our purposes, the SAIAPR TC-12 expansion (Escalante et al., 2010) includes segmentations of each image into regions indicating the locations of constituent objects. 238 different object categories are labeled, including animals, people, buildings, objects, and background elements like grass or sky. This provides us with information regarding object category, object location, and object size, as well as the location and categories of other objects present in the same image.

4.4.2 Collecting the Dataset

From the ImageCLEF dataset, we created a total of over 100k distinct games (one per object labeled in the dataset). For the games we imposed an ordering to allow for

[2] Available at http://tamaraberg.com/referitgame

collecting the most interesting expressions first. Initially we prioritized games for objects in images with multiple objects of the same category. Once these games were completed, we prioritized ordering based on object category to include a comprehensive range of objects. Finally, after successfully collecting referring expressions from the prioritized games, we posted games for the remaining objects. In order to evaluate consistency of expression generation across people, we also include a probability of repeating previously played games during collection.

To date, we have collected 130,525 successfully completed games. This includes 10,431 canned games (a person playing against the computer, not including the initial seed set) and 120,094 real games (two people playing). We recorded at least 1,115 users contributing with referring expressions. 96,654 distinct objects from 19,984 photographs are represented in the dataset. This covers almost all of the objects present in the IAPR corpus. The remaining objects from the collection were either too small or too ambiguous to result in successful games.

For data collection, we posted the game online for anyone on the web to play and encouraged participation through social media and the survey section of reddit. In this manner we collected over 4 thousand referring expressions over a period of 3 weeks. To speed up data collection, we also posted the game on Mechanical Turk (MT). Turkers were paid upon completion of 10 correct games (games where Player 2 clicks on the correct object of interest). Turkers were pre-screened to have approval ratings above 80% and to be located in the US for language consistency. At the end, due to the time efficiency of crowdsourcing we collected almost 95% of the referring expressions from MT.

$$
\begin{aligned}
S \; &::= subject\_word \\
color\_word' \; &::= rel(S, color\_word)_{color\_word'=color\_word} \; | \\
&\quad\; prep\_in(S, color\_word)_{color\_word'=color\_word} \\
size\_word' \; &::= rel(S, size\_word)_{size\_word'=size\_word} \\
abs\_loc\_word' \; &::= rel(S, abs\_loc\_word)\ _{abs\_loc\_word'=abs\_loc\_word} | \\
&\quad\; prep\_on(S, orientation\_word) \;\wedge \neg prep\_of(S, \_)_{abs\_loc\_word'=on+orientation\_word} \\
rel\_loc\_word' \; &::= RL \\
RL \; &::= prep\_rel\_loc\_word(S, object\_word)_{RL=rel\_loc\_word} \; | \\
&\quad\; prep\_on(S, orientation\_word) \;\wedge\; prep\_of(S, object\_word)\ _{RL=on\_orientation\_word} | \\
&\quad\; prep\_to(S, orientation\_word) \;\wedge\; prep\_of(S, object\_word)\ _{RL=to\_orientation\_word} | \\
&\quad\; prep\_at(S, orientation\_word) \;\wedge\; prep\_of(S, object\_word)\ _{RL=at\_orientation\_word} \\
generic\_word' \; &::= amod(S, generic\_word)
\end{aligned}
$$

Figure 4.2: Templates for parsing attributes from referring expressions (§4.4.3).

4.4.3 Processing the Dataset

Because of the size of the dataset, hand annotation of all referring expressions is prohibitive. Therefore, similar to past work (FitzGerald et al., 2013), we design an automatic method to pre-process the expressions and extract object and attribute mentions. These automatically processed expressions are used only for analysis and model training. We also fully hand label portions of the dataset for evaluation (§4.5.2).

By examining the expressions in the collected dataset, we define a set of attributes with broad coverage of the attribute types used in the referring expressions. We define the set of attributes for a referring expression as a 7-tuple $R = \{r_1, r_2, r_3, r_4, r_5, r_6, r_7\}$:

- r_1 is an entry-level category attribute,
- r_2 is a color attribute,

- r_3 is a size attribute,
- r_4 is an absolute location attribute,
- r_5 is a relative location relation attribute,
- r_6 is a relative location object attribute,
- r_7 is a generic attribute,

Color and *size* attributes refer to the object color (e.g. "blue") and object size (e.g. "tiny") respectively. *Absolute location* refers to the location of the object in the image (e.g. "top of the image"). *Relative location relation* and *relative location object* attributes allow for referring expressions that localize the object with respect to another object in the picture (e.g. "the car to the left of the tree"). *Generic attributes* cover all less frequently observed attribute types (e.g. "wooden" or "round").

The *entry-level category attribute* is related to the concept of entry-level categories first proposed by Psychologists in the 1970s (Rosch, 1978) and explored in Chapter 5. The idea of entry-level categories is that an object can belong to many different categories; an indigo bunting is an oscine, a bird, a vertebrate, a chordate, and so on. But, a person looking at a picture of one would probably call it a bird (unless they are very familiar with ornithology). Therefore, we include this attribute to capture how people name object categories in referring expressions.

Parsing the referring expressions: We parse the expressions using the most recent version of the StanfordCoreNLP parser (Socher et al., 2013). We begin by traversing the parse tree in a breadth-first manner and selecting the head noun of the sen-

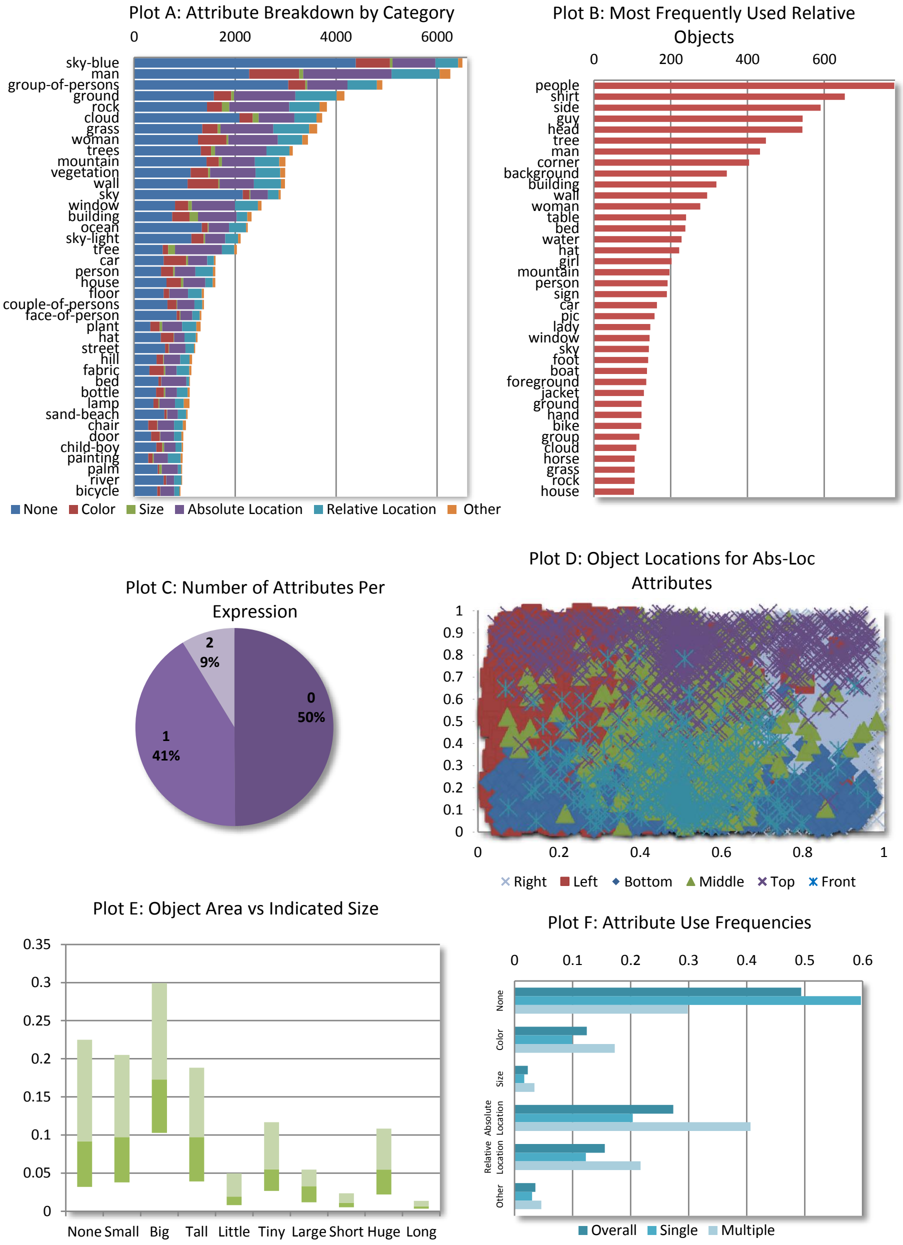


Figure 4.3: Analyses of the ReferItGame Dataset. **Plot A** shows frequency and attribute occurrence for common object categories. **Plot B** shows objects frequently used as reference points, ie "to the left of the man". **Plot C** shows frequencies of using 0, 1 or 2 attributes within the same expression. **Plot D** shows object locations vs location words used. **Plot E** shows normalized object size vs size words used (bars show 1^{st} through 3^{rd} quartiles). **Plot F** shows the frequency of usage of each attribute type for images containing either a *single* instance of the object category or *multiple* instances of the category.

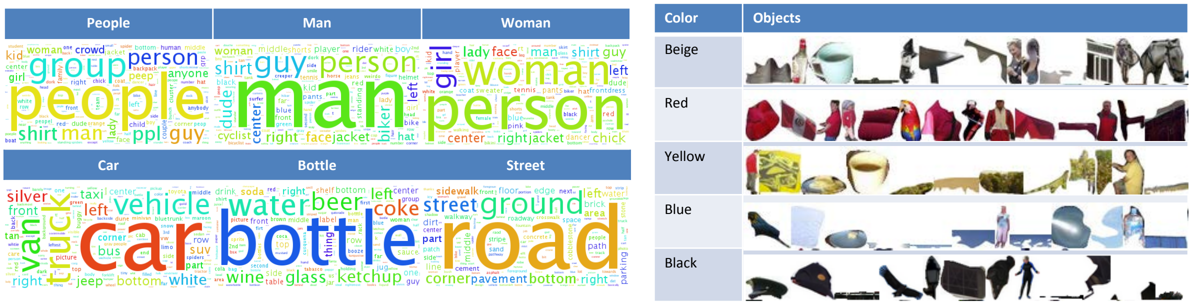


Figure 4.4: **Left:** Tag clouds showing entry-Level category words used in referring expressions to name various object categories, with word size indicating frequency. For example, this indicates that "streets" are often called "road", sometimes "ground", sometimes "roadway", etc. **Right:** example objects predicted to portray some of our color attribute values. Note sometimes our color predictor is quite accurate, and sometimes it makes mistakes (see the man in a red shirt predicted as "yellow").

tence to determine the object of the referring expression, denoted as *subject_word*. We pre-define a dictionary of attribute-values (*color_word*, *size_word*, *abs_location_word*, *rel_location_word*) for each of the attributes based on the observed data using a combination of POS-tagging and manual labeling.

We then apply a template-based approach on the collapsed dependency relations to recover the set of attributes (the main template rules are shown in Figure 4.2). The relationship *rel* indicates any linguistic binary relationship between the subject word S and another word, including the *amod* relationship. *Orientation_word* captures the words like left, right, top and bottom. For *generic_word* we consider any modifier words other than those captured by our other attributes (color, size, location).

Using this template-based parser we can for instance parse the following expression: "Red flower on top of pedestal". The first rule would match the $prep(S, color\_word)$ relation, effectively recovering the attribute $color\_word'$ as "red". The second rule would match the $prep\_on(S, orientation\_word) \wedge prep\_of(S, object\_word)$ relations, recovering *rel_loc_word′* as “on top of ” and *object_word* as “pedestal”.

The accuracy of our parser based processing is 91%. This was evaluated on 4,500 expressions that were manually parsed by a human annotator.

4.4.4 Dataset Analysis

In the resulting dataset, we have a range of coverage over objects. For 10,304 of the objects we have 2 or more referring expressions while for the rest of the objects we have collected only one expression. This creates a dataset that emphasizes breadth while also containing enough data to study speaker variation.

Multiple attribute analyses are provided in Figure 4.3. We find that most expressions use 0, 1, or 2 attributes (in addition to the entry-level attribute object word), with very few expressions containing more than 2 attributes (frequencies are shown in Fig 4.3c). We also examine what types of attributes are used most frequently, according to object category in Fig 4.3a, and when associated with single or multiple occurrences of the same object category in an image in Fig 4.3f. The frequency of attribute usage in images containing multiple objects of the same type increases for all types, compared to single object occurrences. Perhaps more interestingly, the use of different attributes is highly category dependent. People use more attribute words overall to describe some categories, like “man”, “woman”, or “plant”, and the distribution of attribute types also varies by category. For example, color attributes are used more frequently for categories like “car” or “woman” than for categories like “sky” or “rock”.

We also examine which objects are most frequently used as points of reference, e.g.,“the chair next to the *man*” in Fig 4.3b. We observe that people and some background categories like “tree” or “wall” are often used to help localize objects in referring expressions. Additionally, we provide plots showing the relationship between object location in the image and use of absolute location words, Fig 4.3d, as well as size words vs object area, Fig 4.3e.

Finally, we study entry-level category attribute-values to understand how people name objects in referring expressions. Tag clouds indicating the frequencies of words used to name various object categories are provided in Fig 4.4 (left). Objects like “street” are usually referred to as “road”, but sometimes they are called “ground”, “roadway”, etc. “Bottles” are usually called “bottle”, but sometimes referred to as “coke” or “beer”. Interestingly, “man” is usually called “man” while “woman” is most often called “person” in the referring expressions.

4.5 Generating Referring Expressions

In this section we describe our proposed generation model and provide experimental evaluations on three test sets.

4.5.1 Generation Model

Given an input tuple $I = \{P, S\}$, where P is a target object and S is a scene (image containing multiple objects), our goal is to generate an output referring expression, R. For instance, the representation R for the referring expression: *The big old white cabin beside the tree* would be $R = \{cabin, white, big, \varnothing, beside, tree, old\}$.

To generate referring expressions we construct vocabularies V_{r_i} with candidate values for each attribute $r_i \in R$, where attribute vocabulary V_{r_i} contains the set of words observed in our parsed referring expressions for attribute r_i plus an additional ε value indicating that the attribute should be omitted from the referring expression entirely.

In this way, our framework can jointly determine which attributes to include in the expression (e.g.,“size” and “color”) and what attribute values to generate (e.g.,“small” and “blue”) from the list of all possible values. We enforce a constraint to always include an “entry-level category” attribute (e.g. “boy”) so that we always generate a word referring to the object.

We pose our problem as an optimization where we map a tuple $\{P, S\}$ to a referring expression R^* as:

$$\begin{aligned} R^* &= \underset{R}{\operatorname{argmax}} E(R, P, S) \\ &\text{s. t. } f_i(R) \leq b_i \end{aligned} \tag{4.1}$$

Where the objective function E is decomposed as:

$$\begin{aligned} E(R, P, S) = \alpha &\sum_{i=2}^{6} \phi_i(r_i, P, S) \\ &+ \beta \sum_{i=1}^{7} \psi_i(r_i, type(P)) \\ &+ \sum_{i>j} \psi_{i,j}(r_i, r_j) \end{aligned} \tag{4.2}$$

Where ϕ_i is the compatibility function between an attribute-value for r_i and the properties of the observed scene S and object P (described in §4.5.1). The terms ψ_i and $\psi_{i,j}$ are unary and pairwise priors computed based on observed co-occurrence statistics of attribute-values for r_i with categories (where $type(P)$ denotes the type or category of an object) and between pairs of attribute-values (described in §4.5.1). Attributes r_1 and r_7 are modeled only in the priors since we do not have visual models for these attributes.

The constraints $f_i(R) \leq b_i$ are restricted to be linear constraints and are used to impose hard constraints on the solution. The first such constraint is used to control the verbosity (length) of the generated referring expression using a constraint function that imposes a minimum attribute length requirement by restricting the number of entries r_i that can take value ε in the solution.

$$\sum_i \mathbb{1}[r_i = \varepsilon] \leq 7 - \gamma(P, S), \tag{4.3}$$

where $\mathbb{1}[.]$ is the indicator function and $\gamma(P, S)$ is a term that allows us to change the length requirement based on the object and scene (so that images with a larger number of objects of the same type have a larger length requirement).

Finally we add hard constraints such that $r_5 = \varepsilon \iff r_6 = \varepsilon$, so that relative location and relative object attributes are produced together.

Content-based potentials

Potentials ϕ_i are defined for attributes r_2 to r_6. Attribute r_7 represents a variety of different attributes, e.g. material or shape attributes, but we lack sufficient data to train visual models for these infrequent attribute terms. Therefore, we model these attributes using only prior statistics-based potentials (§4.5.1). Visual recognition models for recognizing entry-level object categories could also be incorporated for modeling r_1, but we leave this as future work.

Color attribute:

$$\phi_2(r_2 = c_k, P, S) = sim(hist_{c_k}, hist(P)),$$

where $hist(P)$ is the HSV color histogram of the object P. We compute similarity sim using cosine similarity, and $hist_{c_k}$ is the mean histogram of all objects in our training data that were referred to with color attribute-value $c_k \in V_{r_2}$.

Size attribute:

$$\phi_3(r_3 = s_k, P, S) = \frac{1}{\sigma_{s_k}\sqrt{2\pi}} e^{-\left(size(P)-\mu_{s_k}\right)^2 \big/ 2\sigma_{s_k}^2}, \tag{4.4}$$

where $size(P)$ is the size of object P normalized by image size. We model the probabilities of each size word $s_k \in V_{r_3}$ as a Gaussian learned on our training set.

Absolute-location attribute:

$$\phi_4(r_4 = a_k, P, S) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_{a_k}|}} e^{-\frac{1}{2}(loc(P)-\mu_{a_k})^T {\Sigma_{a_k}}^{-1}(loc(P)-\mu_{a_k})}, \tag{4.5}$$

where $loc(P)$ are the 2-dimensional coordinates of the object P normalized to be $\in [0-1]$. Parameters μ_{a_k} and Σ_{a_k} are estimated from training data for each absolute location word

$a_k \in V_{r_4}$.

Relative-location and Relative object:

$$\phi_5(r_5 = l_k, P, S) = \mathbb{1}[l_k = \varepsilon] \cdot g(count(type(P), S)). \tag{4.6}$$

If there are a larger number of objects of the same type in the image we find that the probability of using a relative-location-object increases (e.g., "the car to the right of the man"). For images where P was the only object of that category type, the probability of using a relative-location-object is 0.12. This increases to 0.22 when there were two objects of the same type and further increases to 0.26 for additional objects of the same type. Therefore, we model the probability of selecting relative location value $l_k \in V_{r_5}$ as a function g, where $count(type(P), S)$ counts the number of objects in the scene S of the same category type as the object P.

$$\phi_6(r_6 = o_k, P, S) = \mathbb{1}[o_k \in objectsnear(location(P), S)]. \tag{4.7}$$

The above expression filters out potential relative objects $o_k \in V_{r_6}$ that are not located in sufficient proximity to object P or are not present in the image at all.

Prior statistics-based potentials

Prior statistics-based potentials are modeled for all of the attributes r_1 - r_7. Note that these potentials do not depend on specific attribute-values but only on the given object category *type*(P).

Unary prior potentials ψ_i are defined as:

$$\psi_i(r_i, \mathit{type}(P)) \;=\; \frac{\sum_{j=1}^{|D|} \mathbb{1}[(r_i^{(j)} \neq \epsilon) \;\wedge\; (\mathit{type}(P^{(j)}) = \mathit{type}(P))]}{\sum_{j=1}^{|D|} \mathbb{1}[\mathit{type}(P^{(j)}) = \mathit{type}(P)]} + \frac{\sum_{j=1}^{|D|} \mathbb{1}[r_i^{(j)} \neq \epsilon]}{|D|} + \lambda,$$

where $D = \{P^{(j)}, S^{(j)}, R^{(j)}\}$ is our training dataset and λ is a small additive smoothing term. The two terms in the above expression represent *category-specific* counts and *global* counts of the number of times a given attribute r_i was output in a referring expression in training data. Pairwise prior potentials $\psi_{i,j}$ are defined as:

$$\sum_{i<j} \psi_{i,j}(r_i, r_j) = \sum_{i<j} \psi_{i,j}^{(1)}(r_i, r_j) + \psi_{5,6}^{(2)}(r_5, r_6),$$

$$\psi_{i,j}^{(1)}(r_i, r_j) = \begin{cases} 1 & \text{if } r_i = r_j = \varepsilon \\ C + \lambda & o.w., \end{cases}$$

$$\psi_{5,6}^{(2)}(r_5 = a, r_6 = b) = \frac{\sum_{t=1}^{|D|} \mathbb{1}[(r_5^{(t)} = a) \;\wedge\; (r_6^{(t)} = b)]}{|D|}, \tag{4.8}$$

where $C = \frac{\sum_{t=1}^{|D|} \mathbb{1}[(r_i^{(t)} \neq \epsilon) \;\wedge\; (r_j^{(t)} \neq \epsilon)]}{|D|}$. The pairwise potential $\psi_{i,j}^{(1)}$ captures the pairwise statistics of how frequently people use pairs of attribute types. For instance how frequently people use both color and size attributes to refer to an object. The pairwise potential $\psi_{i,j}^{(2)}$ produces a cohesion score between relative-location words and relative-object words based on global dataset statistics.

Source	Prec(%)	Recall(%)
Baseline - A	27.92	43.27
Full Model - A	**36.28**	**53.44**
Baseline - B	29.87	50.57
Full Model - B	**36.68**	**59.80**
Baseline - C	28.85	37.41
Full Model - C	**37.73**	**48.54**

Table 4.1: Baseline Model & Full Model performance on the three test sets (A,B,C).

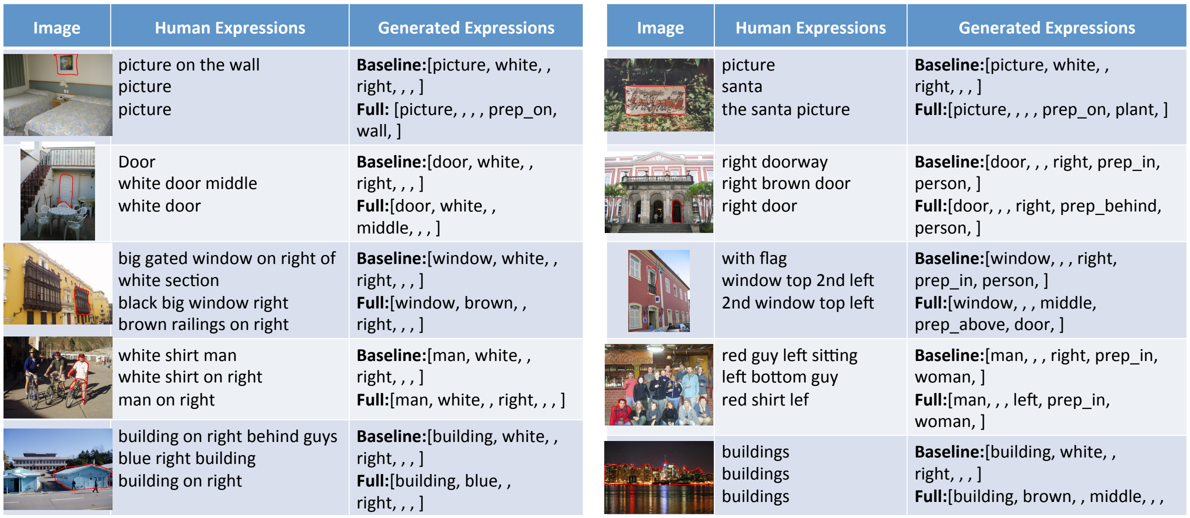


Image	Human Expressions	Generated Expressions
	picture on the wall picture picture	**Baseline:**[picture, white, , right, , ,] **Full:** [picture, , , , prep_on, wall,]
	Door white door middle white door	**Baseline:**[door, white, , right, , ,] **Full:**[door, white, , middle, , ,]
	big gated window on right of white section black big window right brown railings on right	**Baseline:**[window, white, , right, , ,] **Full:**[window, brown, , right, , ,]
	white shirt man white shirt on right man on right	**Baseline:**[man, white, , right, , ,] **Full:**[man, white, , right, , ,]
	building on right behind guys blue right building building on right	**Baseline:**[building, white, , right, , ,] **Full:**[building, blue, , right, , ,]

Image	Human Expressions	Generated Expressions
	picture santa the santa picture	**Baseline:**[picture, white, , right, , ,] **Full:**[picture, , , , prep_on, plant,]
	right doorway right brown door right door	**Baseline:**[door, , , right, prep_in, person,] **Full:**[door, , , right, prep_behind, person,]
	with flag window top 2nd left 2nd window top left	**Baseline:**[window, , , right, prep_in, person,] **Full:**[window, , , middle, prep_above, door,]
	red guy left sitting left bottom guy red shirt lef	**Baseline:**[man, , , right, prep_in, woman,] **Full:**[man, , , left, prep_in, woman,]
	buildings buildings buildings	**Baseline:**[building, white, , right, , ,] **Full:**[building, brown, , middle, , ,

Figure 4.5: Example results, including human generated expressions, baseline and full model generated expressions. For some images the model does well at mimicking human expressions (left). For others it does not generate the correct attributes (right).

4.5.2 Experiments

We implement the proposed model using the binary integer linear programming software (IBM ILOG CPLEX). This requires introducing a set of indicator variables for each of our multi-valued attributes and another set of indicator variables to model pairwise interactions between our variables, as well as incorporating additional consistency constraints between variables. Model parameters (α and β) are tuned on data randomly sampled from our training set consisiting on the entire dataset excluding the images used in the test sets. Another consideration is that we only use to train our models the referring expressions that were validated by the opponent player in the game by successfully finding the referent object. Note that our validation step allows grammar errors as long as the referring expression still includes enough information to identify the referent. This is not critical for the content planning stage but a full system that includes surface realization should take this in consideration when trying to learn models from these expressions, or use external text data.

Test Sets: We evaluate our model on three test sets, each containing 500 objects. For each object in the test sets we collect 3 referring expressions using the ReferItGame and manually label the attributes mentioned in each expression. We find human agreement to be 72.31% on our dataset (where we measure agreement as mean matching accuracy of attribute values for pairs of users across images in our test sets). The three test sets are created to evaluate different aspects of our data.

Test Set A contains objects sampled randomly from the entire dataset. This test set is meant to closely resemble the full dataset distribution. The goal of the other two test sets is to sample expressions for "interesting" objects. We first identify categories that are mainly related to background content elements, e.g. "sky, ground, floor, sand, sidewalk, etc". We consider these categories to be potentially less interesting for study than categories like people, animals, cars, etc. *Test Set B* contains objects sampled from the most frequently occurring object categories in the dataset, selected to contain a balanced number of objects from each category, excluding the less interesting categories. *Test Set C* contains objects sampled from images that contain at least 2 objects of the same category, excluding the less interesting categories.

Results: *Qualitative examples* are shown in Fig 4.5 comparing our results to the human produced expressions. For some images (left) we do quite well at predicting the correct attributes and values. For others we do less well (right). We also show example objects predicted for some color words in Fig 4.4 (right). We see that our model can fail in several ways, such as generating the wrong attribute-value due to inaccurate predictions by visual models or selecting incorrect attributes to include in the generated expression.

Quantitative results: precision and recall measures for the 3 test sets are reported in Table 4.1, including evaluation of a baseline version of our model which incorporates only the prior potentials (Section 4.5.1) without any content based estimates. We see that our model performs reasonably on both measures, and outperforms the baseline by a large margin on all test sets, with highest performance on the broadly sampled interesting category test set. Note that our problem is somewhat different than traditional REG where the input is often attribute-value pairs and the task is to select which pairs to include in the expression. Our goal is to jointly select which attributes to include and what values to predict from a list of all possible values for the attribute.

4.6 Discussion

In this chapter we have introduced a new game to crowd-source referring expressions for objects in natural scenes. We have used this game to produce a new large-scale dataset. We have also proposed an optimization based model for Referring Expression Generation and performed experimental evaluations. Generating the right set of attributes and values for each attribute in referring expressions is a challenging problem. The first principle in the gricean maxims suggests that referring expressions should not be more informative than required, yet we observe in our data that people are purposefully redundant in many instances. This redundancy can take many forms while not being ambiguous enough so that a referring expression stops being efficient. Because if there is too much redundancy in a referring expression, it might create an unnecessarily high cognitive load in the recipient. We model this in our REG approach by looking at the distribution of attributes for each type of object in our dataset. In our current model, we only encourage a larger set of attributes to be used when there are many distractor objects. It is still left to model more complex relationships where on occasions one might need to refer to an object in relation to the distribution of attributes of another object, or set of objects.

The amount of attributes and the specificity of the words used as values for those attributes also have a direct relationship with our working vocabulary. For instance, if we are dealing with a picture depicting three animals and we have words in our vocabulary to uniquely identify each animal, we might prefer to use one such word instead of other properties like size, location, or color. But assigning the name that people are likely to use for categorizing any given object is a challenging task on itself. We specifically address this problem in the context of basic-level and entry-level categories in Chapter 5.

CHAPTER 5: PREDICTION OF ENTRY-LEVEL CATEGORIES

In this section we focus our attention to a more basic problem that also tries to address the disparity between what computational visual recognition systems output and the visual descriptions of people in the more constrained context of object categorization. This work was originally published in (Ordonez et al., 2013a) and an expanded version in (Ordonez et al., 2015).

5.1 Introduction

Algorithms have now advanced to the point where they can recognize or localize thousands of object categories with reasonable accuracy (Deng et al., 2010; Perronnin et al., 2012; Krizhevsky et al., 2012; Dean et al., 2013; Simonyan and Zisserman, 2014; Szegedy et al., 2014). (Russakovsky et al., 2014) present an overview of recent advances in classification and localization for up to 1000 object categories. While one could predict any one of many relevant labels for an object, the question of "What *should* I actually call it?" is becoming important for large-scale visual recognition. For instance, if a classifier were lucky enough to get the example in Figure 5.1 correct, it might output *grampus griseus*, while most people would simply call this object a *dolphin*. We propose to develop categorization systems that are aware of these kinds of human naming choices.

This notion is closely related to ideas of *basic and entry-level categories* formulated by psychologists such as Eleanor Rosch (Rosch, 1978) and Stephen Kosslyn (Jolicoeur

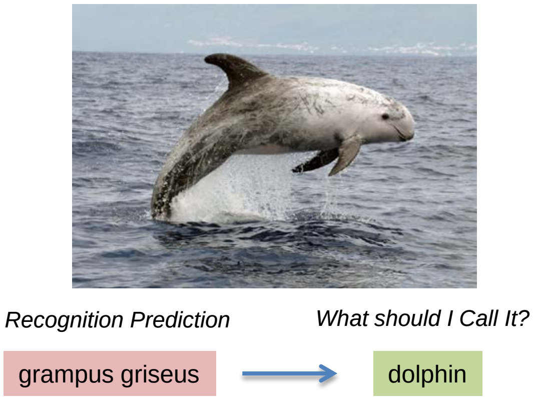


Figure 5.1: Example translation between a WordNet based object category prediction and what people might call the depicted object.

et al., 1984). Rosch defines *basic-level categories* as roughly those categories at the highest level of generality that still share many common attributes and have fewer distinctive attributes. An example of a basic level category is *bird* where most instances share attributes like having feathers, wings, and beaks. Super-ordinate, more general, categories such as *animal* will share fewer attributes and demonstrate more variability. Subordinate, more specific categories, such as *American Robin* will share even more attributes like shape, color, and size. Rosch studied basic level categories through human experiments, e.g. asking people to enumerate common attributes for a given category. The work of (Jolicoeur et al., 1984) further studied the way people identify categories, defining the concept of *entry-level categories*. Entry level categories are essentially the categories that people naturally use to identify objects. The more prototypical an object, the more likely it will have its entry point at the basic-level category. For less typical objects the entry point might be at a lower level of abstraction. For example an *American robin* or a *penguin* are both members of the same basic-level *bird* category. However, the *American*

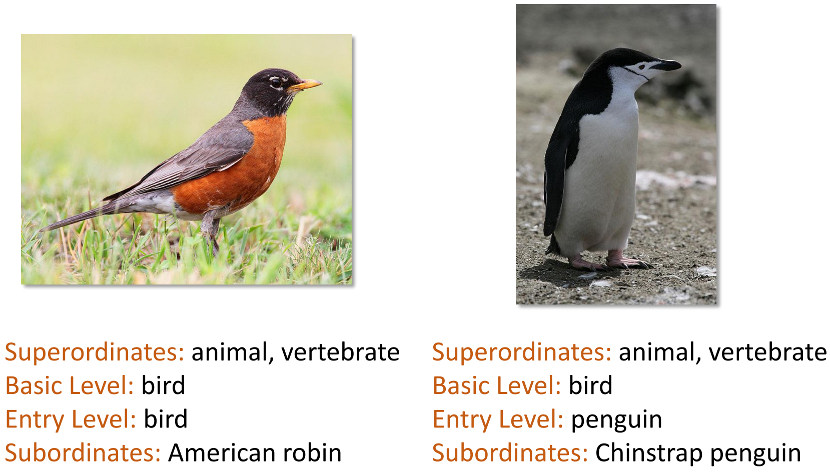

Superordinates: animal, vertebrate
Basic Level: bird
Entry Level: bird
Subordinates: American robin
Superordinates: animal, vertebrate
Basic Level: bird
Entry Level: penguin
Subordinates: Chinstrap penguin

Figure 5.2: An *American Robin* is a more prototypical type of bird hence its *entry-level category* coincides with its *basic level category* while for penguin which is a less prototypical example of bird, the *entry-level category* is at a lower level of abstraction.

robin is more prototypical, sharing many features with other birds and thus its entry-level category coincides with its basic-level category of *bird*, while the entry-level category for a *penguin* would be at a lower level of abstraction (see Figure 5.2).

So, while objects are members of many categories – e.g. Mr Ed is a palomino, but also a horse, an equine, an odd-toed ungulate, a placental mammal, a mammal, and so on – most people looking at Mr Ed would tend to call him a *horse*, his entry level category (unless they are fans of the show). This chapter focuses on the problem of object naming in the context of *entry-level categories*. We consider two related tasks: 1) learning a mapping from *fine-grained* / encyclopedic categories – *e.g.*, leaf nodes in WordNet (Fellbaum, 1998) – to what people are likely to call them (*entry-level categories*) and 2) learning to map from outputs of thousands of noisy computer vision classifiers/detectors evaluated on an image to what a person is likely to call a depicted object.

Evaluations show that our models can effectively emulate the naming choices of human observers. Furthermore, we show that using noisy vision estimates for image content, our system can output words that are significantly closer to human annotations than either raw visual classifier predictions or the results of using a state of the art hierarchical classification system (Deng et al., 2012b) that can output object labels at varying levels of abstraction from very specific terms to very general categories.

5.1.1 Insights into Entry-Level Categories

At first glance, the task of finding the entry-level categories may seem like a linguistic problem of finding a *hypernym* of any given word. Although there is a considerable conceptual connection between entry-level categories and hypernyms, there are two notable differences:

1. Although *"bird"* is a hypernym of both *"penguin"*, and *"sparrow"*, *"bird"* may be a good entry-level category for *"sparrow"*, but not for *"penguin"*. This phenomenon — that some members of a category are more prototypical than others — is discussed in *Prototype Theory* (Rosch, 1978).

2. Entry-level categories are not confined by (inherited) hypernyms, in part because encyclopedic knowledge is different from common sense knowledge. For example *"rhea"* is not a kind of *"ostrich"* in the strict taxonomical sense. However, due to their visual similarity, people generally refer to a *"rhea"* as an *"ostrich"*. Adding to the challenge is that although extensive, WordNet is neither complete nor prac-

tically optimal for our purpose. For example, according to WordNet, *"kitten"* is not a kind of *"cat"*, and *"tulip"* is not a kind of *"flower"*.

In fact, both of the above points have a connection to visual information of objects, as visually similar objects are more likely to belong to the same entry-level category. In this work, we present the first extensive study that (1) characterizes entry-level categories in the context of translating encyclopedic visual categories to natural names that people commonly use, and (2) provides methods to predict entry-level categories for input images guided by semantic word knowledge or by using a large-scale corpus of images with text.

5.1.2 Chapter Overview

This chapter is divided as follows. Section 5.2 presents a summary of related work. Section 5.3 introduces a large-scale image categorization system based on convolutional network activations. In Section 5.4 we learn translations from subordinate concepts to entry-level concepts. In Section 5.5 we propose two models that can take an image as input and predict entry-level concepts. Finally, in Section 5.6 we provide experimental evaluations.

5.2 Related work

Questions about *entry-level categories* are directly relevant to recent work on the connection between computer vision outputs and (generating) natural language descriptions of images (Farhadi et al., 2010; Ordonez et al., 2011; Kuznetsova et al., 2012; Mitchell et al., 2012; Gupta et al., 2012; Kulkarni et al., 2013; Hodosh et al., 2013; Ramnath et al., 2014; Mason and Charniak, 2014; Kuznetsova et al., 2014). Previous works have not directly addressed naming preference choices for entry-level categories when generating sentences. Often the computer vision label predictions are used directly during surface realization (Mitchell et al., 2012; Kulkarni et al., 2013), resulting in choosing non-human like namings for constructing sentences even when handling a relatively small number of categories (i.e. Pascal VOC categories like potted-plant, tv-monitor or person). For these methods, our entry-level category predictions could be used to generate more natural names for objects. Other methods handle naming choices indirectly in a data-driven fashion by borrowing human references from other visually similar objects (Kuznetsova et al., 2012, 2014; Mason and Charniak, 2014).

Our work is also related to previous works that aim to discover visual categories from large-scale data. The works of (Yanai and Barnard, 2005) and (Barnard and Yanai, 2006) learn models for a set of categories by exploring images with loosely associated text from the web. We learn our set of categories directly as a subset of the WordNet (Fellbaum, 1998) hierarchy, or from the nouns used in a large set of carefully selected image captions that directly refer to images. The more recent works of (Chen et al., 2013) and (Divvala et al., 2014) present systems capable of learning any type of visual concept from images on the web, including efforts to learn simple common sense relationships between visual concepts (Chen et al., 2013). We provide a related output in our work, learning mappings between *entry-level categories* and subordinate/leaf-node categories. The recent work of (Feng et al., 2015) proposes that entry-level categorization can be viewed as lexical semantic knowledge, and presents a global inference formulation to map all encyclopedic categories to their entry-level categories collectively.

On a technical level, our work is related to (Deng et al., 2012b) that tries to “hedge” predictions of visual content by *optimally* backing off in the WordNet hierarchy. One key difference is that our approach uses a reward function over the WordNet hierarchy that is non-monotonic along paths from the root to the leaves. Another difference is that we have replaced the underlying leaf node classifiers from (Deng et al., 2012b) with recent convolutional network activation features. Our approach also allows mappings to be learned from a WordNet leaf node, l, to natural word choices that are not along a path from l to the root, “entity”. In evaluations, our results significantly outperform those of (Deng et al., 2012b) because although optimal in some sense, they are not optimal with respect to how people describe image content.

Our work is also related to the growing challenge of harnessing the ever increasing number of pre-trained recognition systems, thus avoiding “starting from scratch” whenever developing new applications. It is wasteful not to take advantage of the CPU weeks (Felzenszwalb et al., 2010; Krizhevsky et al., 2012), months (Deng et al., 2010, 2012b), or even millennia (Le et al., 2012) invested in developing recognition models for increasingly large labeled datasets (Everingham et al., 2010; Russell et al., 2008; Xiao et al., 2010; Deng et al., 2009; Torralba et al., 2008). However, for any specific end-user application, the categories of objects, scenes, and attributes labeled in a particular dataset may not be the most useful predictions. One benefit of our work can be seen as exploring the problem of translating the outputs of a vision system trained with one vocabulary of labels (WordNet leaf nodes) to labels in a new vocabulary (commonly used visually descriptive nouns).

Our proposed methods take into account several sources of structure and information: the structure of WordNet, frequencies of word use in large amounts of web text, outputs of a large-scale visual recognition system, and large amounts of paired image and text data. In particular, we use the SBU Captioned Photo Dataset (Ordonez et al., 2011), which consists of 1 million images with natural language descriptions, and Google n-gram frequencies collected for all words on the web. Taking all of these resources together, we are able to study patterns for choice of entry-level categories at a much larger scale than previous psychology experiments.

5.3 A Large-Scale Image Categorization System

Large-scale image categorization has improved drastically in recent years. The computer vision community has moved from handling 101 categories (Fei-Fei et al., 2007) to 100,000 categories (Dean et al., 2013) in a few years. Large-scale datasets like ImageNet (Deng et al., 2009) and recent progress in training deep layered architectures (Krizhevsky et al., 2012) have significantly improved the state-of-the-art. We leverage a system based on these as the starting point for our work.

For features, we use activations from an internal layer of a convolutional network, following the approach of (Donahue et al., 2013). In particular, we use the pre-trained reference model from the Caffe framework (Jia et al., 2014) which is in turn based on the model from (Krizhevsky et al., 2012). This model was trained on the 1,000 ImageNet categories from the ImageNet Large Scale Visual Recognition Challenge 2012. We compute the 4,096 activations in the 7th layer of this network for images in 7,404 leaf node categories from ImageNet and use them as features to train a linear SVM for each category. We further use a validation set to calibrate the output scores of each SVM with Platt scaling (Platt, 1999). There is a potential here for increased performance by using more powerful convolutional network architectures that have been proposed more recently. For instance (Simonyan and Zisserman, 2014) propose an architecture based on a larger number of layers with convolution operations involving smaller receptive fields.

5.4 Translating Encyclopedic Concepts to Entry-Level Concepts

Our objective in this section is to discover mappings between subordinate encyclopedic concepts (ImageNet leaf categories, e.g. Chlorophyllum molybdites) to output concepts that are more *natural* (e.g. mushroom). In Section 5.4.1 we present an approach that relies on the WordNet hierarchy and frequency of words in a web scale corpus. In Section 5.4.2 we follow an approach that uses visual recognition models learned on a paired image-caption dataset.

5.4.1 Language-based Translation

We first consider a translation approach that relies only on language-based information: the hierarchical semantic structure from WordNet (Fellbaum, 1998) and text statistics from the Google Web 1T corpus (Brants and Franz., 2006). We posit that the frequencies of terms computed from massive amounts of text on the web reflect the "naturalness" of concepts. We use the n-gram counts of the Google Web 1T corpus (Brants

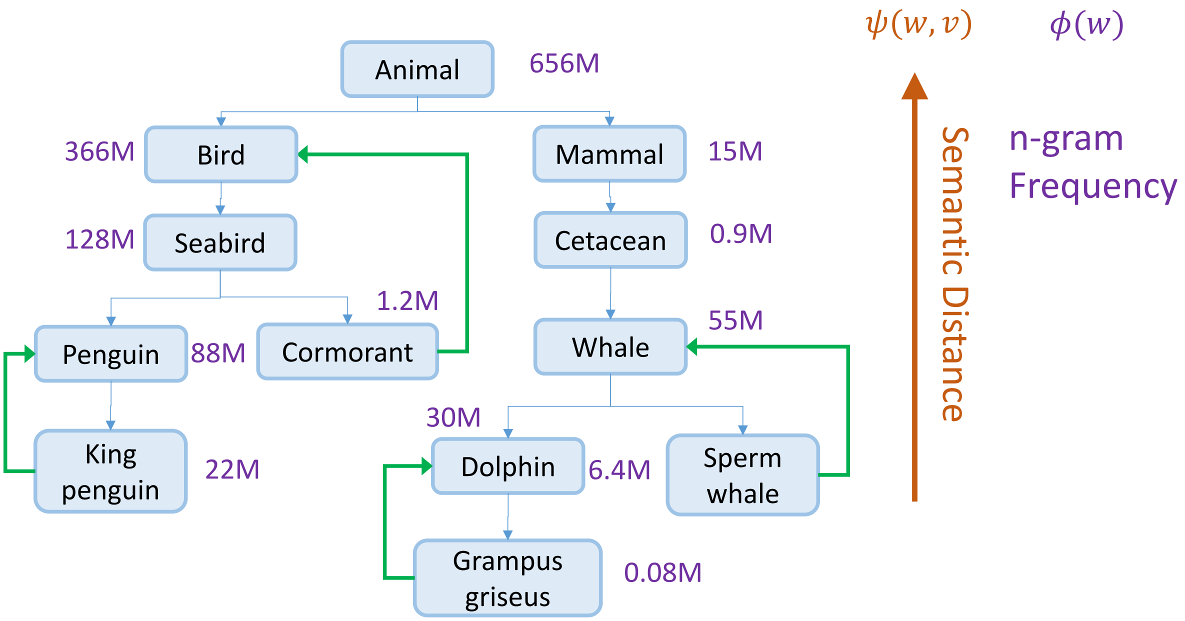


Figure 5.3: Our first categorical translation model uses the WordNet hierarchy to find an hypernym that is close to the leaf node concept (*semantic distance*) and has a large naturalness score based on its n-gram frequency. The green arrows indicate the ideal category that would correspond to the entry-level category for each leaf-node in this sample semantic hierarchy.

and Franz., 2006) as a proxy for naturalness. Specifically, for a synset w, we quantify naturalness as, $\phi(w)$, the log of the count for the most commonly used synonym in w. As possible translation concepts for a given category, v, we consider all nodes, w in $v's$ inherited hypernym structure (all of the synsets along the WordNet path from w to the root).

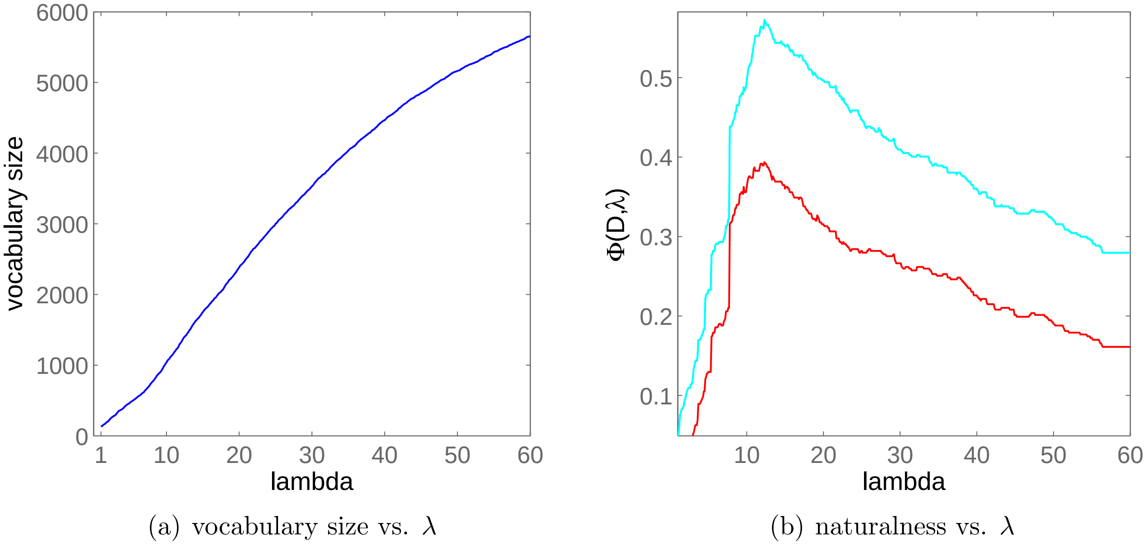
We define a translation function, $\tau(v, \lambda)$, that maximizes a trade-off between naturalness, $\phi(w)$, and semantic proximity, $\psi(w, v)$, measuring the distance between leaf node v and node w in the WordNet hypernym structure:

$$\tau(v, \lambda) = \arg\max_{w} [\phi(w) - \lambda\psi(w, v)], w \in \Pi(v), \tag{5.1}$$

where $\Pi(v)$ is the set of (inherited) hypernyms from v to the root, including v. For instance, given an input category $v =$ *King penguin* we consider all categories along its set of inherited hypernyms, e.g. *penguin, seabird, bird, animal* (see Figure 5.3). An ideal prediction for this concept would be *penguin*. To control how the overall system trades off naturalness vs semantic proximity, we perform line search to set λ. For this purpose we use a held out set of subordinate-category, entry-level category pairs (x_i, y_i) collected using Amazon Mechanical Turk (MTurk) (for details refer to Section 5.6.1). Our objective is to maximize the number of correct translations predicted by our model:

$$\Phi(D, \lambda) = \sum_i \mathbb{1}[\tau(x_i, \lambda) = y_i], \tag{5.2}$$

where $\mathbb{1}[\cdot]$ is the indicator function. We show the relationship between λ and vocabulary



(a) vocabulary size vs. λ

(b) naturalness vs. λ

Figure 5.4: **Left:** shows the relationship between parameter λ and the target vocabulary size. **Right:** shows the relationship between parameter λ and agreement accuracy with human labeled synsets evaluated against the most agreed human label (red) and any human label (cyan).

size in Figure 5.4(a), and between λ and overall translation accuracy, $\Phi(D, \lambda)$, in Figure 5.4(b). As we increase λ, $\Phi(D, \lambda)$ increases initially and then decreases as too much generalization or specificity reduces the naturalness of the predictions. For example, generalizing from *grampus griseus* to *dolphin* is good for "naturalness", but generalizing all the way to "entity" decreases "naturalness". In Figure 5.4(b) the red line shows accuracy for predicting the most agreed upon word for a synset, while the cyan line shows the accuracy for predicting any word collected from any user. Our experiment also supports that *entry-level categories* seem to lie at a certain level of abstraction where there is a discontinuity. Going beyond this level of abstraction suddenly makes our predictions considerably worse (see Figure 5.4(b)). (Rosch, 1978) indeed argues in the context of basic level categories that basic cuts in categorization happen precisely at these discontinuities where there are bundles of information-rich functional and perceptual attributes.

5.4.2 Visual-based Translation

Next, we try to make use of pre-trained visual classifiers to improve translations between input concepts and entry-level concepts. For a given leaf synset, v, we sample a set of $n = 100$ images from ImageNet. For each image, i, we predict some potential entry-level nouns, N_i, using pre-trained visual classifiers that we will describe later in Section 5.5.2. We use the union of this set of labels $N = N_1 \cup N_2 ... \cup N_n$ as keyword annotations for synset v and rank them using a TFIDF information retrieval measure. We consider each category v as a document for computing the *inverse document frequency* (IDF) term. We pick the most highly ranked noun for each node, v, as its entry-level

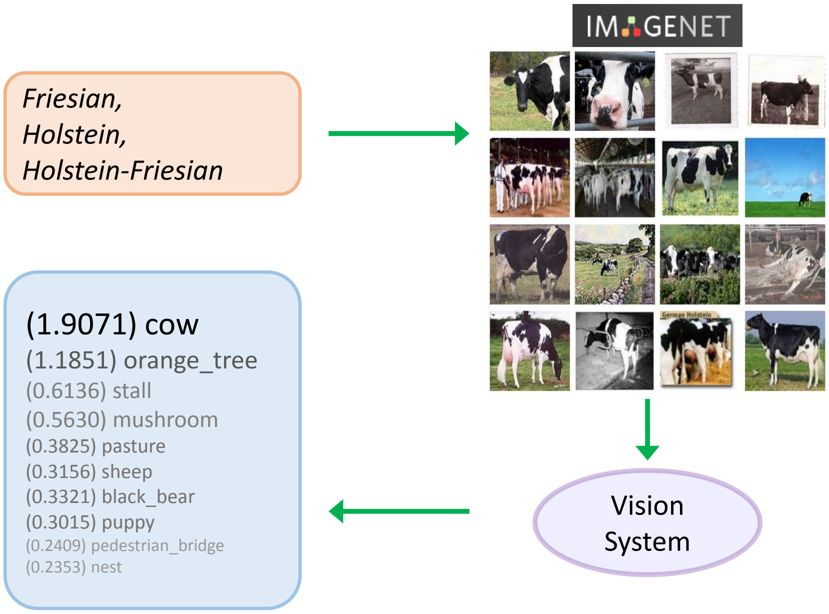
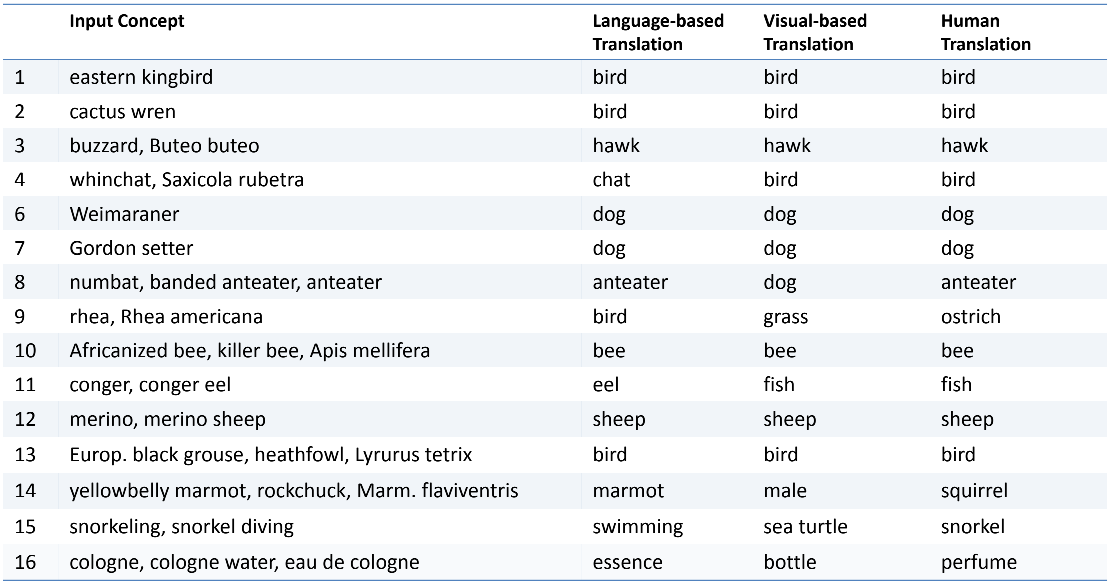


Figure 5.5: We show the system instances of the category *Friesian, Holstein, Holstein-Friesian* and the vision system pre-trained with candidate entry-level categories ranks a set of candidate keywords and outputs the most relevant, in this case *cow*.

categorical translation (see an example in Figure 5.5).

5.5 Predicting Entry-Level Concepts for Images

In Section 5.4 we proposed models to translate between one linguistic concept, e.g. *grampus griseus*, to a more natural concept, e.g. *dolphin*. Our objective in this section is to explore methods that can take an image as input and predict entry-level labels for the depicted objects. The models we propose are: 1) a method that combines "naturalness" measures from text statistics with direct estimates of visual content computed at leaf nodes and inferred for internal nodes (Section 5.5.1) and 2) a method that learns visual models for entry-level category prediction directly from a large collection of images with associated captions (Section 5.5.2).



	Input Concept	Language-based Translation	Visual-based Translation	Human Translation
1	eastern kingbird	bird	bird	bird
2	cactus wren	bird	bird	bird
3	buzzard, Buteo buteo	hawk	hawk	hawk
4	whinchat, Saxicola rubetra	chat	bird	bird
6	Weimaraner	dog	dog	dog
7	Gordon setter	dog	dog	dog
8	numbat, banded anteater, anteater	anteater	dog	anteater
9	rhea, Rhea americana	bird	grass	ostrich
10	Africanized bee, killer bee, Apis mellifera	bee	bee	bee
11	conger, conger eel	eel	fish	fish
12	merino, merino sheep	sheep	sheep	sheep
13	Europ. black grouse, heathfowl, Lyrurus tetrix	bird	bird	bird
14	yellowbelly marmot, rockchuck, Marm. flaviventris	marmot	male	squirrel
15	snorkeling, snorkel diving	swimming	sea turtle	snorkel
16	cologne, cologne water, eau de cologne	essence	bottle	perfume

Figure 5.6: Translations from ImageNet leaf node synset categories to *entry-level categories* using our automatic approaches from Sections 5.4.1 (left) and 5.4.2 (center) and crowd-sourced human annotations from Section 5.6.1 (right).

5.5.1 Linguistically-guided Naming

We estimate image content for an image, I, using the pre-trained models from Section 5.3. These models predict presence or absence of 7,404 leaf node concepts in ImageNet (WordNet). Following the approach of (Deng et al., 2012b), we compute estimates of visual content for internal nodes by hierarchically accumulating all predictions below a node:[1]

$$f(v, I) = \begin{cases} \hat{f}(v, I), & \text{if } v \text{ is a leaf node,} \\ \sum_{v' \in Z(v)} \hat{f}(v', I), & \text{if } v \text{ is an internal node,} \end{cases} \tag{5.3}$$

[1] This function might bias decisions toward internal nodes. Other alternatives could be explored to estimate internal node scores.

where $Z(v)$ is the set of all leaf nodes under node v and $\hat{f}(v, I)$ is a score predicting the presence of leaf node category v from our large scale image categorization system introduced in Section 5.3. Similar to our approach in Section 5.4.1, we define for every node in the ImageNet hierarchy a trade-off function between "naturalness" ϕ (ngram counts) and specificity $\tilde{\psi}$ (relative position in the WordNet hierarchy):

$$\gamma(v, \hat{\lambda}) = [\phi(w) - \hat{\lambda}\tilde{\psi}(w)], \tag{5.4}$$

where $\phi(w)$ is computed as the log counts of the nouns and compound nouns in the text corpus from the *SBU Captioned Dataset* (Ordonez et al., 2011), and $\tilde{\psi}(w)$ is an upper bound on $\psi(w, v)$ from equation (5.1) equal to the maximum path in the WordNet structure from node v to node w. We parameterize this trade-off by $\hat{\lambda}$.

For entry-level category prediction in images, we would like to maximize both "naturalness" and estimates of image content. For example, text based "naturalness" will tell us that both *cat* and *dog* are good entry-level categories, but a confident visual prediction for *German shepherd* for an image tells us that *dog* is a much better entry-level prediction than *cat* for that image.

Therefore, for an input image, we want to output a set of concepts that have a large prediction for both "naturalness" and content estimate score. For our experiments we output the top K WordNet synsets with the highest f_{nat} scores:

$$f_{nat}(v, I, \hat{\lambda}) = f(v, I)\gamma(v, \hat{\lambda}). \tag{5.5}$$

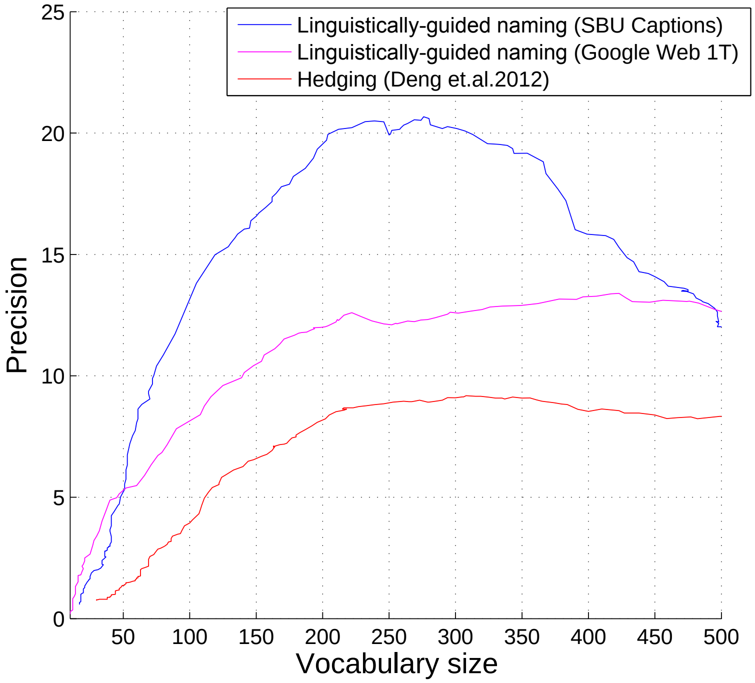


Figure 5.7: Relationship between average precision agreement and working vocabulary size (on a set of 1000 images) for the hedging method (Deng et al., 2012b) (red) and our linguistically-guided naming method that uses text statistics from the generic Google Web 1T dataset (magenta) and from the SBU Caption Dataset (Sec. 5.5.1). We use $K = 5$ to generate this plot and a random set of 1000 images from the SBU Captioned Dataset.

As we change $\hat{\lambda}$ we expect a similar behavior as in our language-based concept translations (Section 5.4.1). We can tune $\hat{\lambda}$ to control the degree of specificity while trying to preserve "naturalness" using n-gram counts. We compare our framework to the "hedging" technique of (Deng et al., 2012b) for different settings of $\hat{\lambda}$. For a side by side comparison we modify hedging to output the top K synsets based on their scoring function. Here, the working vocabulary is the unique set of predicted labels output for each method on this test set. Results demonstrate (Figure 5.7) that under different parameter settings we *consistently* obtain much higher levels of precision for predicting entry-level categories than hedging (Deng et al., 2012b). We also obtain an additional gain in performance than in our previous work (Ordonez et al., 2013a) by relying on the dataset-specific text-statistics of the *SBU Captioned Dataset* rather than the more generic *Google Web 1T* corpus.

5.5.2 Visually-guided Naming

In the previous section we rely on WordNet structure to compute estimates of image content, especially for internal nodes. However, this is not always a good measure of content because: 1) The WordNet hierarchy doesn't encode knowledge about some semantic relationships between objects (i.e. functional or contextual relationships), 2) Even with the vast coverage of 7,404 ImageNet leaf nodes we are missing models for many potentially important entry-level categories that are not at the leaf level.

As an alternative, we can directly train models for entry-level categories from data where people have provided entry-level labels – in the form of nouns present in visually descriptive image captions. We postulate that these nouns represent examples of entry-level labels because they have been naturally annotated by people to describe what is present in an image. For this task, we leverage the SBU Captioned Photo Dataset (Ordonez et al., 2011), which contains *1 million* captioned images. We transform this dataset into a set $D = \{X^{(j)}, Y^{(j)} \mid X^{(j)} \in \mathbf{X}, Y^{(j)} \in \mathbf{Y}\}$, where $\mathbf{X} = [0\text{–}1]^s$ is a vector of estimates of visual content for $s = 7,404$ ImageNet leaf node categories and $\mathbf{Y} = [0,1]^d$ is a set of binary output labels for d target categories.

Input content estimates are provided by the SVM content predictors based on convolutional network activation features described in Section 5.3. We run these SVM predictors over the whole image as opposed to the max-pooling approach over bounding boxes from our initial work as presented in (Ordonez et al., 2013a) so that we have a more uniform comparison to our linguistically-guided naming approach (Section 5.5.1) which does the same. There was some minor drop in performance when running our models exclusively on the whole image. Compared to our previous work, our visually-guided naming approach still has a significant gain from using the *ConvNet* features introduced in section 5.3.

For training our d target categories, we obtain labels Y from the million captions by running a POS-tagger (Bird, 2006) and defining $Y^{(j)} = \{y_{ij}\}$ such that:

$$y_{ij} = \begin{cases} 1, & \text{if caption for image } j \text{ has noun } i, \\ 0, & \text{if otherwise.} \end{cases} \tag{5.6}$$

The POS-tagger helps clean up some word sense ambiguity due to polysemy, by only selecting those instances where a word is used as a noun. d is determined experimentally from data by learning models for the most frequent nouns in this dataset. This provides us with a target vocabulary that is both likely to contain entry-level categories (because we expect entry-level category nouns to commonly occur in our visual descriptions) and to contain sufficient images for training effective recognition models. We use up to 10,000 images for training each model. Since we are using human labels from real-world data, the frequency of words in our target vocabulary follows a power-law distribution. Hence

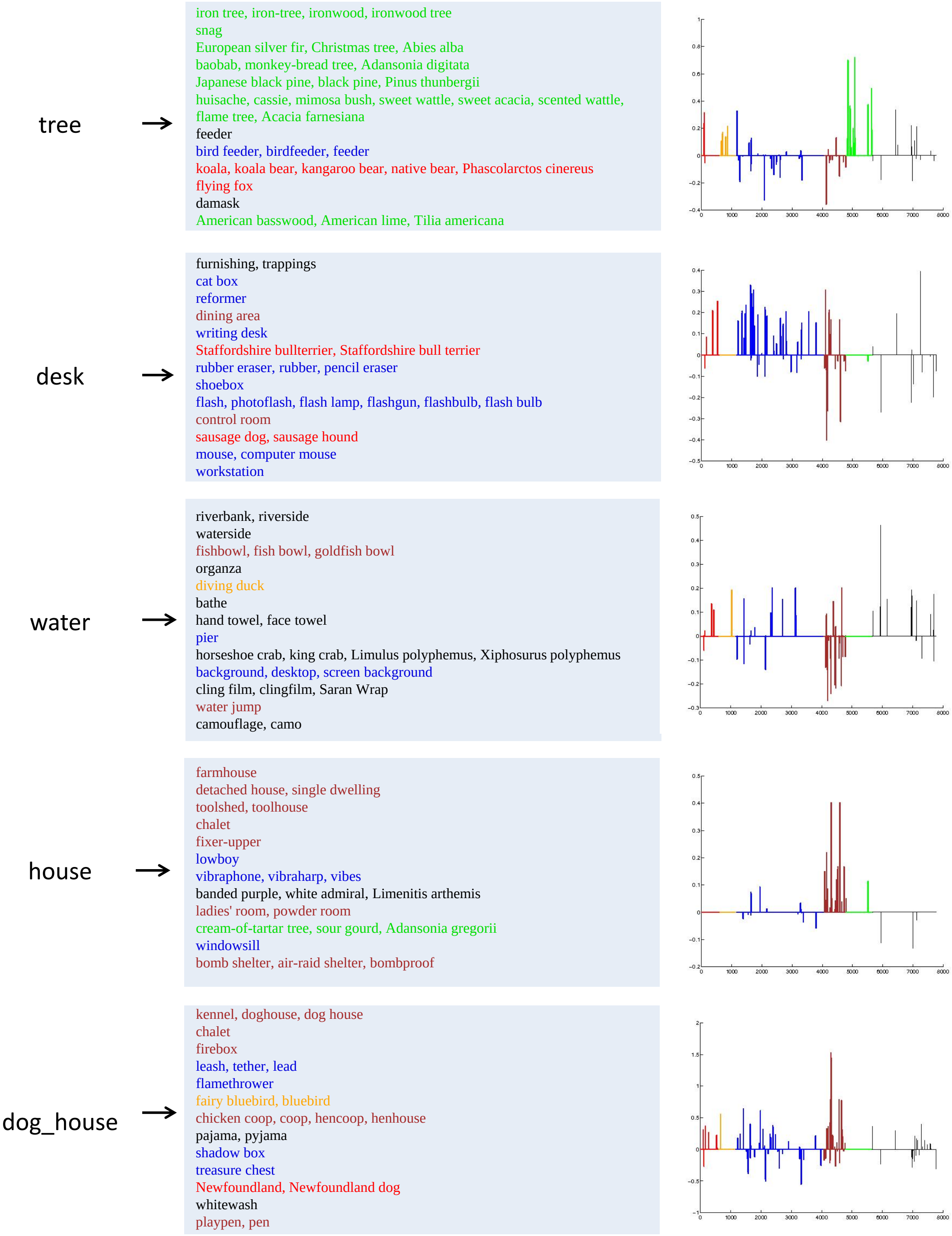


Figure 5.8: Entry-level categories with their corresponding top weighted leaf node features after training an SVM on our noisy data and a visualization of weights grouped by an arbitrary categorization of leaf nodes. vegetation(green), birds(orange), instruments(blue), structures(brown), mammals(red), others(black).

we only have a very large amount of training data for a few most commonly occurring noun concepts. Specifically, we learn linear SVMs followed by Platt scaling for each of our target concepts. We keep $d = 1,169$ of the best performing models. Our scoring function f_{svm} for a target concept v_i is then:

$$f_{svm}(v_i, I, \theta_i) = \frac{1}{1 - exp(a_i \theta_i^{\top} X + b_i)}, \qquad (5.7)$$

where θ_i are the model parameters for predicting concept v_i, and a_i and b_i are Platt

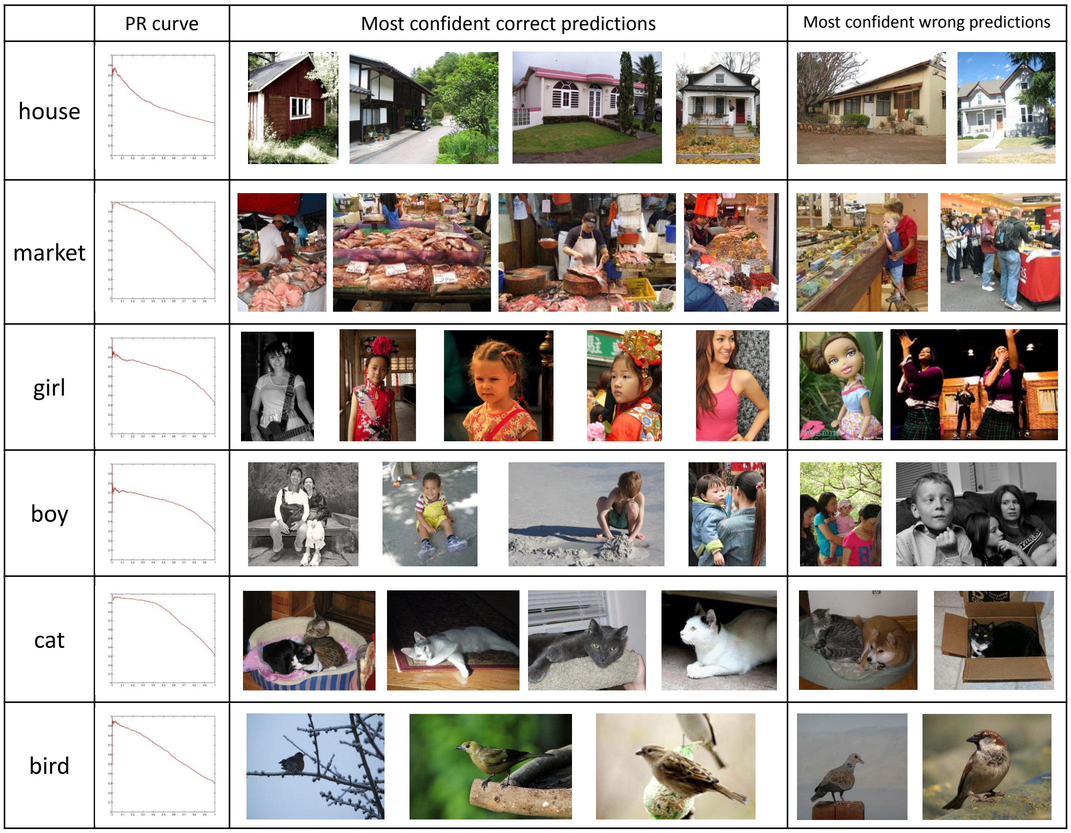

PR curve
Most confident correct predictions
Most confident wrong predictions
house
market
girl
boy
cat
bird

Figure 5.9: Sample predictions from our experiments on a test set for each type of category. Note that image labels come from caption nouns, so some images marked as correct predictions might not depict the target concept whereas some images marked as wrong predictions might actually depict the target category.

scaling parameters learned for each target concept v_i on a held out validation set.

$$R(\theta_i) = \frac{1}{2}\|\theta_i\| + c\sum_{j=1}^{|D|} max(0, 1 - y_{ij}\theta_i^\top X^{(j)})^2. \tag{5.8}$$

We learn the parameters θ_i by minimizing the squared hinge-loss with ℓ_1 regularization (eqn 5.8). The latter provides a natural way of modeling the relationships between the input and output label spaces that encourages sparseness (examples in Figure 5.8). We find $c = 0.01$ to yield good results for our problem and use this value for training all individual models.

One of the drawbacks of using the ImageNet hierarchy to aggregate estimates of visual concepts (Section 5.5.1) is that it ignores more complex relationships between concepts. Here, our data-driven approach to the problem implicitly discovers these relationships. For instance a concept like *tree* has a co-occurrence relationship with *bird* that may be useful for prediction. A chair is often occluded by the objects sitting on the chair, but evidence of those types of objects, e.g. *people* or *cat* or co-occurring objects, e.g. *table* can help us predict the presence of a chair. See Figure 5.8 for some example learned relationships.

Given this large dataset of images with noisy visual predictions and text labels, we manage to learn quite good estimators of high-level content, even for categories with relatively high intra-class variation (e.g. girl, boy, market, house). We show some results of images with predicted output labels for a group of images in Figure 5.9.

5.6 Experimental Evaluation

We evaluate two results – models that learn general translations from encyclopedic concepts to entry-level concepts (Section 5.6.1) and models that predict entry-level concepts for images (Section 5.6.2). We additionally provide an extrinsic evaluation of our naming prediction methods by using them for a sentence retrieval application (Section 5.6.3).

5.6.1 Evaluating Translations

We obtain translations from ImageNet synsets to entry-level categories using Amazon Mechanical Turk (MTurk). In our experiments, users are presented with a 2x5 array of images sampled from an ImageNet synset, x_i, and asked to label the depicted concept. Results are obtained for 500 ImageNet synsets and aggregated across 8 users per task. We found agreement (measured as at least 3 of 8 users in agreement) among users for 447 of the 500 concepts. We show a plot of the distribution of the number of users agreeing for various categories in Figure 5.10, indicating that even though there are many potential labels for each synset (e.g. *Sarcophaga carnaria* could conceivably be labeled as fly, dipterous insect, insect, arthropod, etc) people have a strong preference for particular categories. We denote our resulting set of reference translations as: $D = \{(x_i, y_i)\}$, where each element pair corresponds to a translation from a leaf node x_i to an entry-level word y_i.

We show sample results from each of our methods to learn concept translations in

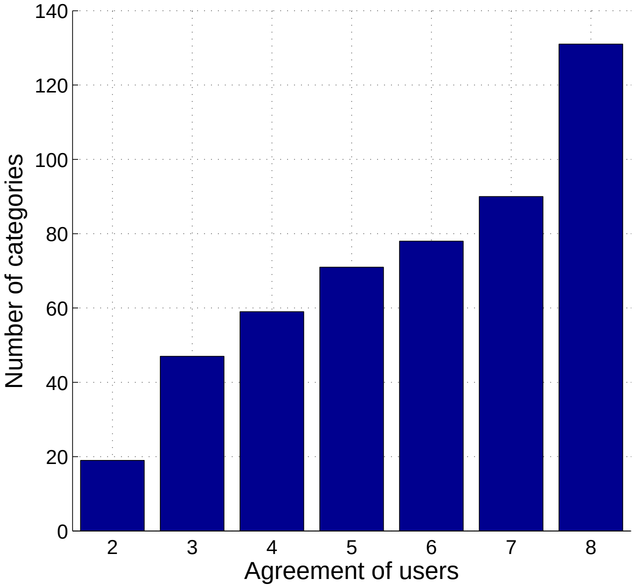


Figure 5.10: Here we show the distribution of the number of annotators agreeing among 8 users in the naming task for a group of 500 categories. Note that for more than 120 categories all users had an unanimous (all 8 users) preferred category. There is also a considerable gap between 2 and 3, and between 7 and 8.

Figure 5.6. In some cases language-based translation fails. For example, *whinchat* (a type of bird) translates to "chat" most likely because of the inflated counts for the most common use of "chat". Visual-based translation fails when it learns to weight context words highly, for example "snorkeling" → "water", or "African bee" → "flower" even when we try to account for common context words using TFIDF. Finally, even humans are not always correct, for example "Rhea americana" looks like an ostrich, but is not taxonomically one. Even for categories like "marmot" most people named it "squirrel". Overall, our language-based translation (Section 5.4.1) agrees 37% of the time with human supplied translations and the visual-based translation (Section 5.4.2) agrees 33% of the time, indicating that translation learning is a non-trivial task. Our

visual-based translation benefits significantly from using *ConvNet* features (Section 5.3) compared to the 21% agreement that we reported in (Ordonez et al., 2013a). Note that our visual-based translation unlike our language-based translation does not use the WordNet semantic hierarchy to constrain the output categories to the set of inherited hypernyms of the input category.

This experiment expands on previous studies in psychology (Rosch, 1978; Jolicoeur et al., 1984). Readily available and inexpensive online crowdsourcing enables us to gather these labels for a much larger set of (500) concepts than previous experiments and to learn generalizations for a substantially larger set of ImageNet synsets.

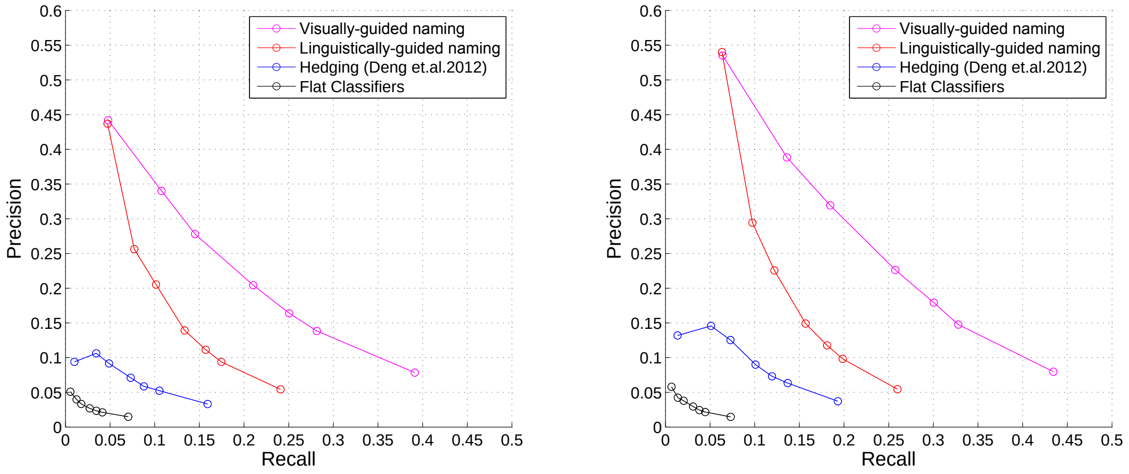
5.6.2 Evaluating Image Entry-Level Predictions

We measure the accuracy of our proposed entry-level category prediction methods by evaluating how well we can predict nouns freely associated with images by users on Amazon Mechanical Turk. We initially selected two evaluation image sets. **Dataset A:** contains 1000 images selected at random from the million image dataset. **Dataset B:** contains 1000 images selected from images displaying high confidence in concept predictions. We additionally collected annotations for another 2000 images so that we can tune trade-off parameters in our models. Both sets are completely disjoint from the sets of images used for learning. For each image, we instruct 3 users on MTurk to write down any nouns that are relevant to the image content. Because these annotations are free associations we observe a large and varied set of associated nouns – 3,610 distinct nouns total in our evaluation sets. This makes noun prediction extremely challenging!

For evaluation, we measure how well we can predict all nouns associated with an image by Turkers (Figure 5.11) and how well we can predict the nouns commonly associated by Turkers (assigned by at least 2 of 3 Turkers, Figure 5.12). For reference we compute the precision of one human annotator against the other two and found that on Dataset A humans were able to predict what the previous annotators labeled with 0.35 precision and with 0.45 precision for Dataset B.

Results show precision and recall for prediction on each of our Datasets, comparing: leaf node classification performance (flat classifier), the outputs of hedging (Deng et al., 2012b), and our proposed entry-level category predictors (linguistically guided naming (Section 5.5.1) and visually guided naming (Section 5.5.2)). Qualitative examples for Dataset A are shown in Figure 5.14 and for Dataset B in Figure 5.15. Performance at this task on Dataset B is in general better than performance on Dataset A. This is unsurprising since Dataset B contains images which have confident classifier scores. Surprisingly their difference in performance is not extreme and performance on both sets is admirable for this challenging task. When compared to our results reported in (Ordonez et al., 2013a) that rely on SIFT + LLC features, we found that the inclusion of *ConvNet* features provided a significant improvement in the performance for the visually-guided naming predictions but it did not improve the results using the WordNet semantic hierarchy for both Hedging (Deng et al., 2012b) and our linguistically-guided naming method.

On the two datasets we find the visually-guided naming model to perform better (Section 5.5.2) than the linguistically-guided naming prediction (Section 5.5.1). In addition, we outperform both leaf node classification and the hedging technique (Deng et al.,



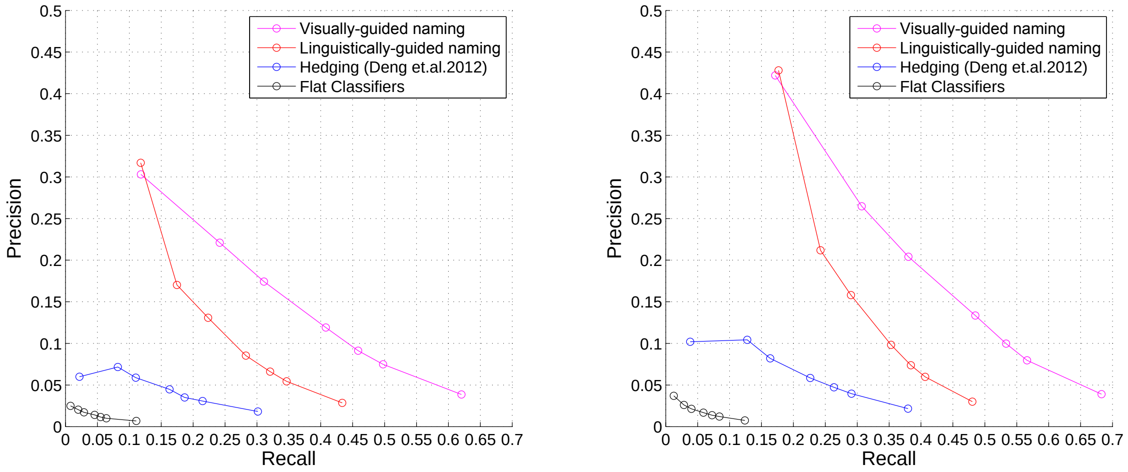
(a) Dataset A

(b) Dataset B

Figure 5.11: Precision-recall curves for different entry-level prediction methods when using the top K categorical predictions for $K = 1, 3, 5, 10, 15, 20, 50$. The ground truth is the union of labels from all users for each image.

2012b).

We additionally collected a third test set Dataset C consisting of random ImageNet images belonging to the 7,404 categories represented in our leaf node classifiers. We make sure not to include those images in the training of our leaf node classifiers. These images are more object-centric, often displaying a single object. This resulted in a smaller number of unique labels provided by users for each image with an average of 2 unique labels per image. We report the precision and recall at $K = 1, 2, 3$ for all of our methods in this dataset in Table 5.1. We observe that at $K = 1$ there is a small advantage of our linguistically-guided naming method compared to the visually-guided naming approach. Both methods surpass the flat mapping classifiers and the Hedging approach. In this different dataset the entry-level category predictors using our visually-guided naming approach still offer better performance than the linguistically-guided naming approach



(a) Dataset A

(b) Dataset B

Figure 5.12: Precision-recall curves for different entry-level prediction methods when using the top K categorical predictions for $K = 1, 3, 5, 10, 15, 20, 50$. The ground truth is the set of labels where at least two users agreed.

Method	Precision	Recall
	$K = 1, 2, 3$	$K = 1, 2, 3$
Flat classifier	$4.40, 4.00, 3.43$	$2.10, 3.82, 4.87$
Hedging	$9.00, 9.55, 10.25$	$4.90, 11.72, 19.64$
Linguist.-guided	$26.70, 16.15, 12.90$	$17.59, 19.52, 22.25$
Visually-guided	$25.80, 17.95, 13.73$	$17.50, 22.76, 25.73$

Table 5.1: Here we show results on Dataset C which consists of images from ImageNet. The human labels for each image are the union of the labels collected from different Mechanical Turk users.

at $K = 2, 3$. Note that our linguistically-guided naming does not require expensive retraining of visual models like our visually-guided naming. Also, the gap between our two naming approaches is smaller than in the previous experiments on Datasets A and B.

5.6.3 Evaluating Image Entry-Level Predictions for Sentence Retrieval

Entry-level categories are also the natural categories that people use in casual language. We evaluate our produced naming predictions indirectly by using them to retrieve image descriptions. Our sentence retrieval approach works as follows: We predict entry-level categories with $K = 5$ and use them as keywords to retrieve a ranked list of sentences from the entire 1 million image descriptions in the *SBU Captioned Dataset*. We use cosine similarity on a bag-of-words model for representation and ranking.

The images in our test Dataset A and Dataset B in the previous section come from the *SBU Captioned Dataset* and therefore already have one image description associated with each of them. This image description was written by the owner of each picture. Note that these "ground truth" image descriptions for each of our test images are included in the pool of 1 million captions. We use the rank of the ground truth image description for each image as a measure of performance in this task. We report on Table 5.2 the number of images for which its "ground truth" description was ranked within the top 1% and the top 10% for the various methods compared here and for each test set. Although our evaluation uses a rough metric of performance, we observed that the top 5 sentences retrieved for images that had its original sentence ranked within the top 1% were also often very good descriptions for the query image. We show some qualitative examples in Figure 5.13.

	Dataset A		Dataset B	
Method	Top 1%	Top 10%	Top 1%	Top 10%
Flat classifier	40	80	48	93
Hedging	62	172	92	266
Linguistically-guided	71	310	104	416
Visually-guided	162	516	210	617

Table 5.2: Here we show the number of images (for each dataset and method) for which we could retrieve its original image description within the top 1% and the top 10%. Note that each dataset has 1000 images in total.

5.7 Discussion

Results indicate that our inferred concept translations are meaningful and that our models are able to predict entry-level categories—the words people use to describe image content—for images. Our models managed to leverage a large scale visual categorization system to make new types of predictions. These methods could apply to a wide range of end-user applications that require recognition outputs to be useful for human consumption, including some of the tasks related to description generation studied in the previous chapters of this thesis. We also presented an experiment on this direction for image description using a sentence retrieval approach.

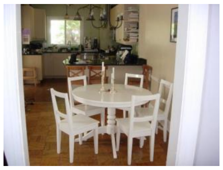
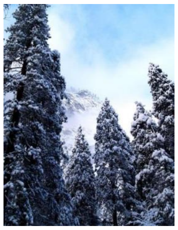
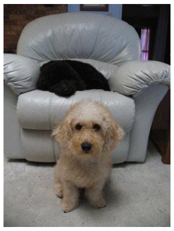
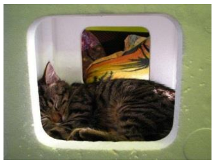
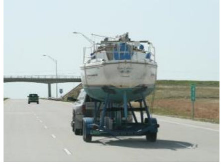
Method	Images	Original Caption	Top 5 Retrieved Sentences
Visually-guided Naming		(808) “dining area in great room open to kitchen opens to seat 8 people”	(1) [table area beside kitchen] (2) [work table sitting area in separate room bathroom kitchen area sleeping area] (3) [dining table in kitchen area] (4) [by the kitchen table area] (5) [dining room table in kitchen]
Visually-guided Naming		(1105) “fresh snow on pine trees in yosemite national park”	(1) [pine trees forest under snow] (2) [pine tree in snow] (3) [pine tree in snow] (4) [snow in pine tree] (5) [pine tree in snow]
Visually-guided Naming		(60747) “theres no room in the chair for me so i am sitting in daddys spot on the floor”	(1) [dog and cat in chair] (2) [dog and cat in chair] (3) [bear in a chair poor chair bear] (4) [dog in cat] (5) [cat in chair]
Linguistically-guided Naming		(519) “cat in the box”	(1) [cat in box cat on box] (2) [cat in the cat box] (3) [obligatory cat in box picture] (4) [cat in cats] (5) [cat in box upside down cat]
Linguistically-guided Naming		(37153) “we were wondering where you could sail a boat in colorado we passed this boat about 4 times”	(1) [car under boat] (2) [car in truck] (3) [car in car mirror] (4) [portable car toy box in cars and trucks] (5) [car in car mirror bw]

Figure 5.13: Good examples of retrieved sentences describing image content. We show the original sentence for each image with its corresponding rank in parenthesis. We also show the top 5 retrieved sentences for each image. We are showing here only images that ranked highly the original caption (within the top 10%) .

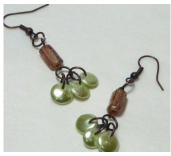
	Images	Labels	Flat Classifier	Hedging [Deng et.al.2012]	Linguistically-guided Naming	Visually-guided Naming
Results in the top 25%		bird, duck feather fin, fur goose, lake pond, swan water, wing	pen cob whooper cygnet Cygnus	**swan** aquatic **bird** anseriform waterfowl	**swan** **bird** snow **duck** pen	**swan** **duck** **pond** **bird** **water**
		boat, crowd flag, harbor lake, ocean people, wave race, sail ship, warf	drill container **harbor** clipper seaside	vessel craft transport vehicle **ship**	**ship** tree coast shore **boat**	**boat** **ship** beach **sail** **harbor**
		car, home house, land power line road, sky street, tree truck, wire	secondhand chair golf power **car**	transport wheel structure self-propelled motor	**tree** building **car** **house** tower	bus **street** **car** cable car **road**
		bag, basket bike, boy, hat man, person sidewalk, jacket stone, street tire, tree	pannier dirt ice skateboard push-bike	wheel container vehicle transport cover	**bike** **bag** **basket** dog building	**bike** mountain bike seat **boy** girl
		big ben building clock, roof sky, street light tower wall	jigsaw integrate chatelaine turret masjid	structure **building** **tower** circuit house	**building** **tower** house home tree	clock tower **building** **tower** castle church
		bathroom cabinet doorway faucet mirror, sink towel, vanity	console credence armoire Murphy **vanity**	furniture furnish room area box	box room area table **cabinet**	**bathroom** **sink** **cabinet** room floor
Results in the bottom 25%		fence, junk sign stop sign street sign trash can tree	jigsaw gift trophy display comic	outlet place establishment store structure	store place building box window	market shop bar street book
		circle earring hook jewel jewelry make up stone	skeleton clasp toggle pull corkscrew	constraint fix implement device chain	chain bottle bit tree flower	bead silver chain sterling glass

Figure 5.14: Example translations on Dataset A (random images). 1^{st} col shows images. 2^{nd} col shows MTurk associated nouns. These represent the ground truth annotations (entry-level categories) we would like to predict (colored in blue). 3^{rd} col shows predicted nouns using a standard multi-class flat-classifier. 4^{th} col shows nouns predicted by the method of (Deng et al., 2012b). 5^{th} col shows our n-gram based method predictions. 6^{th} col shows our SVM mapping predictions and finally the 7^{th} column shows the labels predicted by our joint model. Matches are colored in green. Figures 5.11,5.12 show the measured improvements in recall and precision.

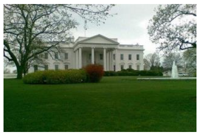
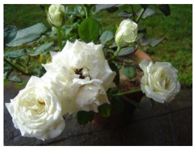
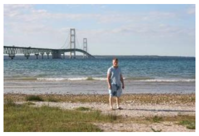
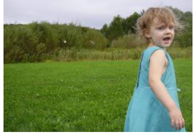
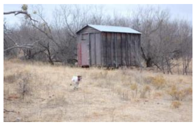
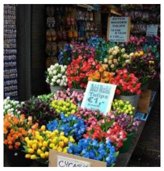
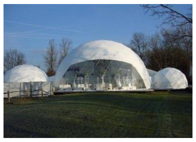
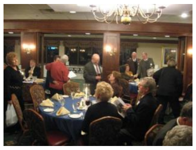
	Images	Labels	Flat Classifier	Hedging [Deng et.al.2012]	Linguistically-guided Naming	Visually-guided Naming
Results in the top 25%		building bush, field fountain grass, home house, window manor, sky, tree, yard, white house	summer farmhouse background detach tombstone	**home** **building** **house** housing structure	**building** **house** **home** **tree** country	**house** barn **field** hill **home**
		dirt, flower grass, leaf petal, plant pot, rain rise, rose stem, white	cauliflower terrarium gypsophilum West mash	vegetable solid food produce matter	**flower** dog tree fruit **white**	**flower** **plant** **rose** **grass** **pot**
		beach, beach sand bridge, cloud coast, grass, water man, ocean, weed sand, shirt shorts, structure	seaside oceanfront strand **sand** waterside	formation shore elevation psychological event	**bridge** shore **water** **coast** side	**beach** **sand** boat **bridge** **water**
		blue dress bush, dress girl, child grass, plant sky, tree	frame tudung Frisbee raglan skirt	wear good consumer cover garment	**dress** woman **tree** **dress** shirt	**grass** shirt **dress** **girl** field
		animal, barn brown, building cabin, dirt, dog farm, field, grass meadow, shack shed, tree, turkey	barnyard corncrib farmhouse sod frame	housing home structure **building** house	**building** house home **tree** area	**barn** **field** horse **farm** truck
		brick, building door, flower market product sign, table window	**window** jigsaw florist gift afghan	outlet place establishment store structure	**flower** store place **market** tree	**market** **flower** fruit pot street
Results in the bottom 25%		architecture bench, dome fence, field grass, sky stage structure tent, tree	geodesic planetarium mosque **dome** observation	**structure** building protection **dome** roof	building roof bridge tower **dome**	water tower tower bridge building background
		beam, chair chandelier gathering, wine glass, indoor light, napkin party, people silverware suit, table	control conference game war conference	structure room area building restaurant	room building area restaurant store	bar pizza shirt **table** office

Figure 5.15: Example translations on Dataset B (images with high response to visual models). 1st col shows images. 2nd col shows MTurk associated nouns. These represent the ground truth annotations (entry-level categories) we would like to predict (colored in blue). 3rd col shows predicted nouns using a standard multi-class flat-classifier. 4th col shows nouns predicted by the method of (Deng et al., 2012b). 5th col shows our n-gram based method predictions. 6th col shows our SVM mapping predictions and finally the 7th column shows the labels predicted by our joint model. Matches are colored in green. Figures 5.11,5.12 show the measured improvements in recall and precision.

CHAPTER 6: DISCUSSION AND FUTURE WORK

6.1 Summary of Contributions

In this thesis we have proposed, implemented, and evaluated systems that can output automatic image descriptions that are closer to the visual descriptions provided by humans using natural language at various level of detail (full sentences, short phrases, and names). In contrast, traditional computer vision systems have focused on producing outputs in the form of labels, locations, and segmentation masks indicating the presence or absence of individual semantic entities like objects, attributes or scenes.

For the data-driven image captioning approach presented in Chapter 2, we constructed a dataset of images with captions that was several magnitudes larger than the previous existing dataset for this task (Farhadi et al., 2010). This allowed us to really take advantage of our method since we showed that performance depends largely on the availability of such large scale dataset. There have also been later attempts to collect captioned image datasets, most notably the Microsoft COCO Dataset (Lin et al., 2014). One important difference is that we relied on careful filtering of already existing captions on the web, thus bypassing the use of expensive crowdsourcing. Our dataset has been used to train and evaluate other caption generation systems (Mitchell et al., 2012; Mason and Charniak, 2014; Vinyals et al., 2014), to learn multimodal image-sentence embeddings (Gong et al., 2014; Kiros et al., 2014), and in our own work on prediction of entry-level categories (Ordonez et al., 2015).

Chapter 3 proposed an idea that is still relevant today: Building a new sentence by stitching together multiple phrases referring to various aspects of the image. This approach is in contrast to some of the previous attempts at generating image descriptions that either used templates, tried to construct new descriptions word by word, or attempted to retrieve entire sentences (Farhadi et al., 2010; Kulkarni et al., 2011; Yang et al., 2011; Mitchell et al., 2012). Some of the most recent caption generation systems leverage convolutional networks for visual representation and recurrent neural networks for language modeling (Karpathy and Fei-Fei, 2014; Vinyals et al., 2014; Chen and Zitnick, 2014; Donahue et al., 2014). Closer to the approach presented in Chapter 3 is the recent work of (Lebret et al., 2015) that generates captions by composing them from individual phrases while still producing high quality descriptions. Our phrase-based retrieval captioning approach and our proposed phrase-based representation for images can potentially take advantage of better image representations as well.

The dataset and language generation approach presented in Chapter 4 deals with a task-specific type of descriptions: Referring Expressions. This is in contrast to some of the previous and current work that focused on generating generic image descriptions. It also sets apart from the previous work on referring expressions (Krahmer and van Deemter, 2012; van Deemter et al., 2006; Viethen and Dale, 2008; Mitchell et al., 2010, 2013a; FitzGerald et al., 2013) by dealing with objects in the context of real world complex scenes. Moreover, we present a dataset that is also several magnitudes larger and considerably different than previously available datasets. One key contribution was the formulation of a purpose-driven game to collect and verify referring expressions. Our associated referring expression generation (REG) approach is also the first proposed in this scenario. Our method attempts the standard goal to predict the set of attributes that people would use to refer to a particular object, and additionally the specific set of values that people would use for each selected attribute.

Chapter 5 brings a modern perspective to entry-level categorization using computational visual recognition. Entry-level categories and basic-level categories were studied in the past in the context of Psychology and the principles of categorization. The availability of large scale image databases like Imagenet (Deng et al., 2009) allowed us to scale experiments on entry-level categorization to a much larger number of categories than those studies. We also propose *naming*, or entry-level category prediction as a complementary component for large scale image categorization systems. We showed experimentally that such systems can readily take advantage of ideas of entry-level categorization to better predict the namings produced by people. Our work is perhaps the first to have this consideration in the categorization problem, learning what is the right level abstraction that people use when categorizing and naming objects in the real world.

Finally, I expect this study to influence further analysis in the connections between images and visually descriptive text of various types, especially increased attention to the referring expression generation problem and entry-level category prediction.

6.2 Future Directions

My vision is that future research will take advantage of advances in both computational visual recognition and natural language understanding to create systems that can solve these problems for practical applications. Significant progress has been made recently by other research groups to address the full sentence generation problem using deep layered architectures especially to leverage better visual representations (Vinyals et al., 2014; Kiros et al., 2014; Donahue et al., 2014; Lebret et al., 2015). We have also presented here a system that takes advantage of deep convolutional network representations to improve entry-level category predictions. Similar gains could be obtained in the referring expression problem that can in turn be used for robotics or human computer interaction applications. The increasing availability of higher quality datasets (i.e. Microsoft COCO (Lin et al., 2014)) will also have a big impact in terms of what will be possible but also our ability to leverage the existing visual data that is already annotated with text of various forms; examples include captioned images like the ones we used in this thesis, images embedded in webpages or video with closed captioning and other types of annotations.

I also envision in the future more research geared toward a unified view of knowledge as opposed to attacking visual and language input as disparate sources of information. We will move from recognizing objects and categories and we will start to recognize complex human activities in visual data that require information that goes beyond what pixel or sensor data can provide.

BIBLOGRAPHY

Aker, A. and Gaizauskas, R. (2010a). Generating image descriptions using dependency relational patterns. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 1250–1258. Association for Computational Linguistics.

Aker, A. and Gaizauskas, R. (2010b). Generating image descriptions using dependency relational patterns. In *Association for Computational Linguistics (ACL)*.

Barbu, A., Bridge, A., Burchill, Z., Coroian, D., Dickinson, S. J., Fidler, S., Michaux, A., Mussman, S., Narayanaswamy, S., Salvi, D., Schmidt, L., Shangguan, J., Siskind, J. M., Waggoner, J. W., Wang, S., Wei, J., Yin, Y., and Zhang, Z. (2012). Video in sentences out. In *Uncertainty in Artificial Intelligence (UAI)*.

Barnard, K., Duygulu, P., Forsyth, D., De Freitas, N., Blei, D. M., and Jordan, M. I. (2003). Matching words and pictures. *The Journal of Machine Learning Research*, 3:1107–1135.

Barnard, K. and Yanai, K. (2006). Mutual information of words and pictures. *Information Theory and Applications*, 2.

Berg, T. L., Berg, A. C., Edwards, J., Maire, M., White, R., Teh, Y.-W., Learned-Miller, E., and Forsyth, D. A. (2004). Names and faces in the news. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–848. IEEE.

Bird, S. (2006). Nltk: the natural language toolkit. In *COLING/ACL Association for Computational Linguistics*.

Bourdev, L., Maji, S., Brox, T., and Malik, J. (2010). Detecting people using mutually consistent poselet activations. In *European Conference on Computer Vision (ECCV)*.

Brants, T. and Franz., A. (2006). Web 1t 5-gram version 1. In *Linguistic Data Consortium (LDC)*.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. In *World Wide Web Conference (WWW)*.

Chen, X., Shrivastava, A., and Gupta, A. (2013). NEIL: Extracting Visual Knowledge from Web Data. In *International Conference on Computer Vision (ICCV)*.

Chen, X. and Zitnick, C. L. (2014). Learning a recurrent visual representation for image caption generation. *CoRR*, abs/1411.5654.

Chum, O., Philbin, J., and Zisserman, A. (2008). Near duplicate image detection: min-hash and tf-idf weighting. In *British Machine Vision Conference (BMVC)*.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR)*.

Dale, R. and Reiter, E. (1995). Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive Science (CogSci)*, 19.

Dale, R. and Reiter, E. (2000). Building natural language generation systems. In *Cambridge University Press*.

Dean, T., Ruzon, M. A., Segal, M., Shlens, J., Vijayanarasimhan, S., and Yagnik, J. (2013). Fast, accurate detection of 100,000 object classes on a single machine. In *Computer Vision and Pattern Recognition (CVPR)*.

Deng, J., Berg, A., Satheesh, S., Su, H., Khosla, A., and Li, F.-F. (2012a). Large scale visual recognition challenge. In *http://www.image-net.org/challenges/LSVRC/2012/index*.

Deng, J., Berg, A. C., Li, K., and Li, F.-F. (2010). What does classifying more than 10,000 image categories tell us? In *European Conference on Computer Vision (ECCV)*.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *Computer Vision and Pattern Recognition (CVPR)*.

Deng, J., Krause, J., Berg, A. C., and Fei-Fei, L. (2012b). Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In *Computer Vision and Pattern Recognition (CVPR)*.

Deng, J., Krause, J., and Fei-Fei, L. (2013). Fine-grained crowdsourcing for fine-grained recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Divvala, S., Farhadi, A., and Guestrin, C. (2014). Learning everything about anything: Webly-supervised visual concept learning.

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

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. (2013). Decaf: A deep convolutional activation feature for generic visual recognition. *CoRR*, abs/1310.1531.

Duygulu, P., Barnard, K., de Freitas, J. F., and Forsyth, D. A. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Computer VisionECCV 2002*, pages 97–112. Springer.

Efros, A. A. and Freeman, W. T. (2001). Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346. ACM.

Elliott, D. and Keller, F. (2014). Comparing automatic evaluation measures for image description. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 452–457.

Escalante, H. J., Hernandez, C. A., Gonzalez, J. A., Lopez-Lopez, A., Montes, M., Morales, E. F., Sucar, L. E., Villasenor, L., and Grubinger, M. (2010). The segmented and annotated iapr tc-12 benchmark. *Computer Vision and Image Understanding (CVIU).*

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338.

Fang, R., Liu, C., She, L., and Chai, J. (2013). Towards situated dialogue: Revisiting referring expression generation. In *Empirical Methods on Natural Language Processing (EMNLP).*

Farhadi, A., Endres, I., Hoiem, D., and Forsyth, D. A. (2009). Describing objects by their attributes. In *Computer Vision and Pattern Recognition (CVPR).*

Farhadi, A., Hejrati, M., Sadeghi, A., Young, P., Rashtchian, C., Hockenmaier, J., and Forsyth, D. A. (2010). Every picture tells a story: generating sentences for images. In *European Conference on Computer Vision ((ECCV).*

Fei-Fei, L., Fergus, R., and Perona, P. (2007). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70.

Fellbaum, C., editor (1998). *WordNet: an electronic lexical database.* MIT Press.

Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645.

Feng, S., Ravi, S., Kumar, R., Kuznetsova, P., Liu, W., Berg, A. C., Berg, T. L., and Choi, Y. (2015). Refer-to-as relations as semantic knowledge. In *AAAI.*

Feng, Y. and Lapata, M. (2010). How many words is a picture worth? automatic caption generation for news images. In *Association for Computational Linguistics (ACL).*

Feng, Y. and Lapata, M. (2013). Automatic caption generation for news images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(4):797–812.

FitzGerald, N., Artzi, Y., and Zettlemoyer, L. (2013). Learning distributions over logical forms for referring expression generation. In *Empirical Methods on Natural Language Processing (EMNLP).*

Gong, Y., Wang, L., Hodosh, M., Hockenmaier, J., and Lazebnik, S. (2014). Improving image-sentence embeddings using large weakly annotated photo collections. In *Computer Vision–ECCV 2014*, pages 529–545. Springer.

Grice, H. P. (1975). Logic and conversation. *Syntax and Semantics*, 3:41–58.

Grubinger, M., Clough, P. D., Muller, H., and Deselaers, T. (2006). The iapr benchmark: A new evaluation resource for visual information systems. In *Proceedings of the International Workshop OntoImage (LREC)*.

Guadarrama, S., Krishnamoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R., Darrell, T., and Saenko, K. (2013). Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *International Conference on Computer Vision (ICCV)*.

Gupta, A., Verma, Y., and Jawahar, C. (2012). Choosing linguistics over vision to describe images. In *Conference on Artificial Intelligence (AAAI)*.

Hays, J. and Efros, A. A. (2008). im2gps: estimating geographic information from a single image. In *Computer Vision and Pattern Recognition (CVPR)*.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.*, 47(1):853–899.

Hoiem, D., Efros, A. A., and Hebert, M. (2005). Geometric context from a single image. In *International Conference on Computer Vision (ICCV)*.

Hoiem, D., Efros, A. A., and Hebert, M. (2007). Recovering surface layout from an image. *Int. J. Comput. Vision*, 75:151–172.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM.

Jing, Y. and Baluja, S. (2008). Pagerank for product image search. In *World Wide Web Conference (WWW)*.

Jolicoeur, P., Gluck, M. A., and Kosslyn, S. M. (1984). Pictures and names: making the connection. cognitive psychology. *Cognitive Psychology*, 16:243–275.

Karpathy, A. and Fei-Fei, L. (2014). Deep visual-semantic alignments for generating image descriptions. *CoRR*, abs/1412.2306.

Kazemzadeh, Ordonez, Matten, M., and Berg, T. L. (2014). Referit game: Referring to objects in photographs of natural scenes. In *EMNLP*.

Kiros, R., Zemel, R., and Salakhutdinov, R. (2014). Multimodal neural language models. In *International Conference on Machine Learning (ICML)*.

Krahmer, E. and van Deemter, K. (2012). Computational generation of referring expressions: A survey. In *Computational Linguistics*, volume 38.

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. In *Neural Information Processing Systems (NIPS)*.

Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A., and Berg, T. (2011). Baby talk: Understanding and generating simple image descriptions. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1601–1608.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A., and Berg, T. (2013). Babytalk: Understanding and generating simple image descriptions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2891–2903.

Kuznetsova, P. (2014). *Composing Image Descriptions in Natural Language*. PhD thesis, Stony Brook University.

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

Kuznetsova, P., Ordonez, V., Berg, T., and Choi., Y. (2014). Treetalk: Composition and compression of trees for image descriptions. *Transaction of the Association for Computational Linguistics (TACL)*.

Kwatra, V., Schödl, A., Essa, I., Turk, G., and Bobick, A. (2003). Graphcut textures: image and video synthesis using graph cuts. In *ACM Transactions on Graphics (ToG)*, volume 22, pages 277–286. ACM.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching. In *Computer Vision and Pattern Recognition (CVPR)*.

Le, Q., Ranzato, M., Monga, R., Devin, M., Chen, K., Corrado, G., Dean, J., and Ng, A. (2012). Building high-level features using large scale unsupervised learning. In *International Conference on Machine Learning (ICML)*.

Lebret, R., Pinheiro, P. O., and Collobert, R. (2015). Phrase-based image captioning. *arXiv preprint arXiv:1502.03671*.

Leung, T. K. and Malik, J. (1999). Recognizing surfaces using three-dimensional textons. In *International Conference on Computer Vision (ICCV)*.

Li, L.-J., Su, H., Xing, E. P., and Fei-Fei, L. (2010). Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *Neural Information Processing Systems (NIPS)*.

Liang, L., Liu, C., Xu, Y.-Q., Guo, B., and Shum, H.-Y. (2001). Real-time texture synthesis by patch-based sampling. *ACM Transactions on Graphics (ToG)*, 20(3):127–150.

Lin, C. Y. (2004). Rouge: A Package for Automatic Evaluation of Summaries. In *Association for Computational Linguistics (ACL)*, Barcelona, Spain.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755. Springer.

Lowe, D. G. (2004). Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision (IJCV)*.

Maji, S., Bourdev, L., and Malik, J. (2011). Action recognition from a distributed representation of pose and appearance. In *Computer Vision and Pattern Recognition (CVPR)*.

Mason, R. and Charniak, E. (2014). Nonparametric method for data-driven image captioning. *Association for Computational Linguistics (ACL)*, pages 592–598.

Mitchell, M., Dodge, J., Goyal, A., Yamaguchi, K., Sratos, K., Han, X., Mensch, A., Berg, A., Berg, T. L., and III, H. D. (2012). Midge: Generating image descriptions from computer vision detections. In *European Chapter of the Association of Computational Linguistics (EACL)*.

Mitchell, M., Reiter, E., and van Deemter, K. (2013a). Typicality and object reference. In *Cognitive Science (CogSci)*.

Mitchell, M., van Deemter, K., and Reiter, E. (2010). Natural reference to objects in a visual domain. In *International Natural Language Generation Conference (INLG)*.

Mitchell, M., van Deemter, K., and Reiter, E. (2011). Two approaches for generating size modifiers. In *European Workshop on Natural Language Generation*.

Mitchell, M., van Deemter, K., and Reiter, E. (2013b). Generating expressions that refer to visible objects. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Ordonez, V., Deng, J., Choi, Y., Berg, A. C., and Berg, T. L. (2013a). From large scale image categorization to entry-level categories. In *International Conference on Computer Vision (ICCV)*, pages 2768–2775. IEEE.

Ordonez, V., Han, X., Kuznetsova, P., Kulkarni, G., Mitchell, M., Yamaguchi, K., K. Stratos, A. G., Dodge, J., Mensch, A., III, H. D., Berg, A., Choi, Y., and Berg, T. (2013b). Large scale retrieval and generation of image descriptions. In *Technical Report / Accepted to the International Journal of Computer Vision - Special Issue on Big Visual Data*.

Ordonez, V., Kulkarni, G., and Berg, T. L. (2011). Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*.

Ordonez, V., Liu, W., Deng, J., Choi, Y., Berg, A. C., and Berg, T. L. (2015). Predicting entry-level categories. In *International Journal of Computer Vision (IJCV) - Marr Prize Special Issue*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics (ACL).

Perronnin, F., Akata, Z., Harchaoui, Z., and Schmid, C. (2012). Towards good practice in large-scale learning for image classification. In *Computer Vision and Pattern Recognition (CVPR)*.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning accurate, compact, and interpretable tree annotation. In *COLING/Association for Computational Linguistics (ACL)*.

Petrov, S. and Klein, D. (2007). Improved inference for unlexicalized parsing. In *HLT- North American Chapter of the Association for Computational Linguistics (NAACL)*.

Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in large margin classifiers*.

Ramnath, K., Baker, S., Vanderwende, L., El-Saban, M., Sinha, S., Kannan, A., Hassan, N., Galley, M., Yang, Y., Ramanan, D., Bergamo, A., and Torresani, L. (2014). Autocaption: Automatic caption generation for personal photos. In *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, pages 1050–1057.

Ren, Y., Van Deemter, K., and Pan, J. Z. (2010). Charting the potential of description logic for the generation of referring expressions. In *International Natural Language Generation Conference (INLG)*.

Roelleke, T. and Wang, J. (2008). Tf-idf uncovered: a study of theories and probabilities. ACM SIGIR International Conference, pages 435–442, New York, NY, USA. ACM.

Rosch, E. (1978). Principles of categorization. *Cognition and Categorization*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Andrej, Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2014). Imagenet large scale visual recognition challenge.

Russell, B. C., Torralba, A., Murphy, K. P., and Freeman, W. T. (2008). Labelme: a database and web-based tool for image annotation. *International Journal of Computer Vision (IJCV)*, 77:157–173.

Seemakurty, N., Chu, J., von Ahn, L., and Tomasic, A. (2010). Word sense disambiguation via human computation. In *Human Computation Workshop*.

Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *ArXiv e-prints*.

Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)*.

Socher, R., Bauer, J., Manning, C. D., and Ng, A. Y. (2013). Parsing With Compositional Vector Grammars. In *Association for Computational Linguistics (ACL)*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2014). Going Deeper with Convolutions. *ArXiv e-prints*.

Tighe, J. and Lazebnik, S. (2010). Superparsing: Scalable nonparametric image parsing with superpixels. In *European Conference on Computer Vision (ECCV)*.

Torralba, A., Fergus, R., and Freeman, W. (2008). 80 million tiny images: a large dataset for non-parametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30.

Van Deemter, K., Gatt, A., van Gompel, R. P., and Krahmer, E. (2012). Toward a computational psycholinguistics of reference production. In *Topics in Cognitive Science*, volume 4(2).

van Deemter, K., van der Sluis, I., and Gatt, A. (2006). Building a semantically transparent corpus for the generation of referring expressions. In *International Conference on Natural Language Generation (INLG)*.

Viethen, J. and Dale, R. (2008). The use of spatial relations in referring expression generation. In *International Natural Language Generation Conference (INLG)*.

Viethen, J. and Dale, R. (2010). Speaker-dependent variation in content selection for referring expression generation. In *Australalasian Language Technology Workshop*.

Viethen, J., Mitchell, M., and Krahmer, E. (2013). Graphs and spatial relations in the generation of referring expressions. In *European Workshop on Natural Language Generation*.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2014). Show and tell: A neural image caption generator.

von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. In *ACM Conf. on Human Factors in Computing Systems (CHI)*.

von Ahn, L., Kedia, M., and Blum, M. (2006a). Verbosity: A game for collecting commonsense knowledge. In *ACM Conference on Human Factors in Computing Systems (CHI)*.

von Ahn, L., Liu, R., and Blum, M. (2006b). Peekaboom: A game for locating objects in images. In *ACM Conference on Human Factors in Computing Systems (CHI)*.

Winograd, T. (1972). Understanding natural language. *Cognitive Psychology*, 3(1).

Xiao, J., Hays, J., Ehinger, K., Oliva, A., and Torralba, A. (2010). Sun database: Large-scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition (CVPR)*.

Yanai, K. and Barnard, K. (2005). Probabilistic web image gathering. In *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 57–64. ACM.

Yang, Y., Teo, C. L., III, H. D., and Aloimonos, Y. (2011). Corpus-guided sentence generation of natural images. In *Empirical Methods on Natural Language Processing (EMNLP)*.

Yao, B. Z., Yang, X., Lin, L., Lee, M. W., and Zhu, S.-C. (2010). I2t: Image parsing to text description. *Proceedings of IEEE*, 98(8).