Image Description with a Goal: Building Efficient Discriminating Expressions for Images

Amir Sadovnik
Cornell University
as2373@cornell.edu

Yi-I Chiu
National Cheng Kung University
n26990021@mail.ncku.edu.tw

Noah Snavely
Cornell University
snavely@cs.cornell.edu

Shimon Edelman
Cornell University
edelman@cornell.edu

Tsuhan Chen
Cornell University
tsuhan@ece.cornell.edu

Abstract

Many works in computer vision attempt to solve different tasks such as object detection, scene recognition or attribute detection, either separately or as a joint problem. In recent years, there has been a growing interest in combining the results from these different tasks in order to provide a textual description of the scene. However, when describing a scene, there are many items that can be mentioned. If we include all the objects, relationships, and attributes that exist in the image, the description would be extremely long and not convey a true understanding of the image. We present a novel approach to ranking the importance of the items to be described. Specifically, we focus on the task of discriminating one image from a group of others. We investigate the factors that contribute to the most efficient description that achieves this task. We also provide a quantitative method to measure the description quality for this specific task using data from human subjects and show that our method achieves better results than baseline methods.

Figure 1. In this paper we develop a method for creating the most efficient textual description that can discriminate one image from a group of images. For example, if the image with the red border is our target image and the rest are distractors, an efficient description might be: *"The image with the gray oven"*, since the target image is the only one in which a gray oven exists. The white cabinets or the window would not need to be mentioned, because they exist in other images. The plant, which also only exists in the target image, might be harder to find because of its smaller size and thus does not need to be mentioned as well.

1. Introduction

Scene understanding is one of the ultimate goals of computer vision. However, coming up with methods to attain this goal is still a very hard problem. Most of the computer vision field is currently focused on trying to extract the information needed for scene understanding such as object detection, scene recognition, and 3D modeling. However, merely listing the output of these algorithms would not amount to a true understanding of the scene. To show a higher level of understanding, one may try to rank these outputs so as to describe things in a correct order and omit those that are of no import.

A visual scene may contain a large number of objects. All these objects stand in certain spatial relationships with one another, and to each one we might be able to ascribe many attributes. However, when asked to describe an image, a human viewer will obviously not choose to name all such objects one by one. Indeed, there is likely to be a great deal of information that a human will deem unnecessary to mention at all. This is partly because a genuine understanding of a scene makes certain items very important, while rendering others insignificant. In addition, different tasks might require different descriptions of the scene. A general description of a scene might be different from a description aimed at singling out one image from a group of images. In this work, we focus on the latter task. As far as we know,

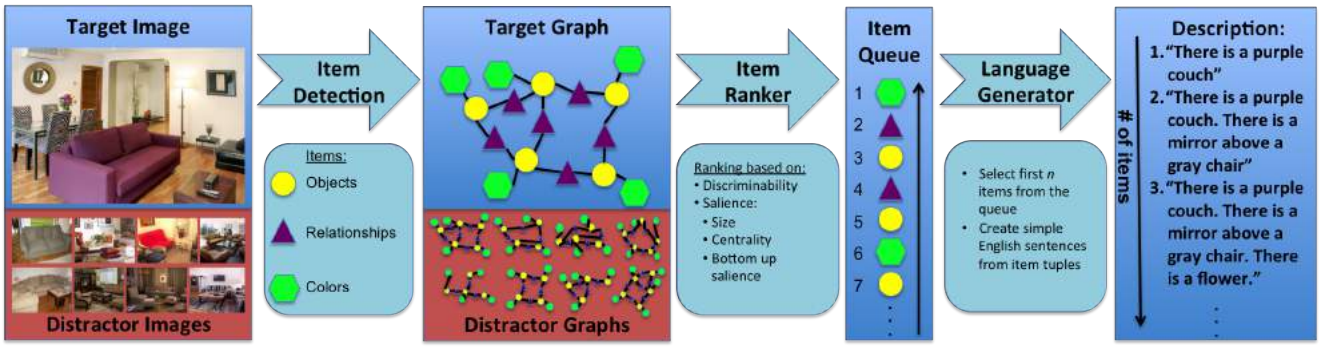

Figure 2. Our approach to building a discriminating description. Given a target image and a set of distractors, we first build a graph for each of the images with three different types of nodes: (a) objects (b) relationships (c) colors. Then, using the graphs from all the images, we rank the different items in the target image. This ranking is based on two main criteria: discriminability and salience. Finally, depending on the length of description we require, we use the top n items and submit them to a natural language generator to create the final description.

this is the first attempt to construct such discriminative description for general scenes.

Consider for example Fig. 1, where the task is to distinguish the image framed in red from the others. If we merely create a "laundry list" of all the objects in the image with their colors and relationships, we might end up with a type of description that starts as follows (we omit the ending of the description because of its length):

"There is window above a gray sink. The sink is above a white cabinet which is next to a white dishwasher. There is a gray oven below a gray stovetop next to a white drawer. There are brown chairs next to a table. . . . "

However, if our task is simply to discriminate our target from the other images, we should be able to use a description as simple as:

"There is a gray oven"

This description is much more efficient for this specific task in that it conveys the same amount of information in many fewer words. In this work, we investigate the possibility of creating such efficient descriptions automatically.

Although this is a specific task, it is useful for other tasks as well. For example, when describing an image, it is known that people tend to mention the unexpected. Therefore, this type of task in which we specifically search for what is unusual about an image as opposed to other ones that are similar to it will need to be incorporated in any system whose goal is to create natural sounding descriptions.

By choosing this specific task, we are also able to measure the effectiveness of our description in a more quantitative manner. This is in contrast to previous works ([4],[11]) in which results are mostly assessed qualitatively. We show that by ranking the candidate items according to our new metric, we are able to create shorter and more efficient descriptions. In addition, we show how the different factors we use to rank these items contribute to the performance.

Although we construct a textual description, in this work we do not focus on the sentence structure. We instead focus on using the visual data to rank the different items in the image, and so create very simple sentences based on specific rules. Although creating an appropriate grammar is also an important part of the challenge of image description, we believe that it is mostly independent of the visual data on which we concentrate. That is, given a set of items and relationships that need to be described, the description and the image are independent. Therefore, by providing the item information to a more complex natural language generation algorithm, a more realistic description can be created.

1.1. Previous Work

In recent years, there have been a few attempts to develop automatic methods for generating textual descriptions of images. For example, Farhadi et al. [4] try to use existing descriptions from the web and match them to new images. Although this method, which constitutes one of the first attempts at scene description, can associate natural sounding sentences with images, it is limited in that it can only select a description from a given database of sentences. Therefore, given a new image, the probability of finding a sentence that describes it very closely is relatively low.

Yao et al. [13] also try to create a textual image description for a scene. They use a hierarchical parsing of the image to generate the description. They try to learn a complex model in which knowledge base information from the web is used to parse the image and to create sentences. While this allows them to generate more natural-sounding sentences, they do not attempt to filter the information detected in the image, and simply end up describing everything.

Berg et al. [8] uses a conditinal random field to detect different objects/relationships/attributes in images. This CRF uses textual descriptions from different online databases to encourage the detection of commonly used objects/relationships/attributes combinations. This means that if a certain relationship was mentioned many times in the database, this relationship will be encouraged in the CRF. Although this approach does take into account the probability of mentioning certain items, it does not do so on a

per-image basis. Since all images use the same description database as potentials in the CRF, a certain item may be encouraged regardless of the specific image being described. In our approach, we attempt to tailor the description to a specific image for a specific task. For example, if in the online database no one ever mentioned a "blue cat", this attribute-object relationship will be discouraged. However, given an image with a "blue cat", this might be exactly what we would want to describe because of its unusualness.

Among other related works, Spain et al. [11] ask people to name objects in photographs, then use this informastion to build a model that tries to predict the importance of objects in novel images. Although this task resembles ours, there are two main differences. First, Spain et al. only consider objects and do not attempt to rank also attributes or relationships. In addition, subjects are asked to list the objects without a specific task in mind. Our work attempts to provide the most efficient description for a specific task.

Farhadi et al. [3] use high-level semantic attributes to describe objects. While most of their work is about object classification and category learning, they do discuss textual description, noting that focusing on unusual attributes results in descriptions similar to those generated by humans. They do not, however, try to combine objects and atttibutes into a scene description. We use the idea of highlighting unusual attributes in our description.

In natural language generation, there has been much work on referring expressions (for an extensive and recent survey see [7]). These are sentences that can refer to one and only one item among a set of items. This work is very closely related to ours, but there are some major differences, which stem from our use of visual data, instead of just a list of properties. For example, imagine trying to refer to the first scene out of the following two:

- chair, table, apple, melon, strawberries, blueberries.
- chair, table, melon, strawberries, blueberries.

The obvious choice for a referring expression generator would be to describe the apple. However, in Fig. 3 we show that this is not the best description when using visual data.

An overview of our method is shown in Fig. 2. This method allows us to create an efficient tailored description for a specific set of target/distractor images. In contrast to previous work, our description is goal-oriented and includes a quantitative estimate of its quality.

2. Item Detection

Similarly to previous work ([8],[9]), we focus on three main types of visual information that can be used to describe a scene:

1. The objects in the scene $O = \{o_1, o_2, ..., o_n\}$
2. The relationships between the objects $R = \{r_{12}, r_{13}, ..., r_{nm}\}$
3. The colors of each object $C = \{c_1, c_2, ..., c_n\}$

We refer to the unified set of O, R, C as items. In this section, we describe how we collect these items; section 3 states our approach to ranking these items.

2.1. Object Collection

The main building blocks for our description are the objects that exist in the image and their categories. We use labeled data for localizing and recognizing objects. Since our main focus is not on recognition but on the description task, we would like to be able to have as many objects as possible in an image, coming from a wide range of object categories. We use three different categories from the indoor LabelMe dataset: kitchens, bathrooms, and living rooms [10]. After cleaning up the LabelMe data, we obtain a dataset with over 150 different types of objects, and an average of 20 objects per image. Although labeled images are expensive, this gives us images with a much richer variety of categories than those used before for image description tasks, allowing us to assess the quality of our algorithm under much more interesting conditions.

2.2. Relationship Detection

We focus on three types of relationships between objects: "above", "overlapping", and "next-to". To detect these, we simply calculate the relative position $(\Delta x, \Delta y)$ and overlap (O) between all pairs of objects that are less than a certain number of pixels away from each other. We then use the following criteria to define the relationship:

1. A "overlaps" C if $\frac{O_{AB}}{BB_A} > 0.8$ where O_{AC} is the overlap area and BB_A is the bounding box area of A.
2. A is "above" C if $-0.375\pi < \tan(\frac{\Delta y}{\Delta x}) < 0.375\pi$
3. A is "next-to" C for all other objects whose distance is less than the threshold.

2.3. Color Detection

Among various possible attributes of an object, we choose to detect color, since it offers fairly reliable results. Our color classifier distinguishes among 11 different colors, using the database of [12]. As features we use a normalized binned histogram in HSV space [14]. We then use an SVM with an RBF kernel [1]. When presented with a new set of images, we use the mask of each object to extract the feature histogram. Then after running the classifier, we get a set of 11 probability values (one for each color), which signify the likelihood of the object to have that specific color.

3. Item Ranking

Our description model resembles the incremental algorithm of [2] for referring expression generation. The basic

Figure 3. An illustration of why visual saliency should be helpful. When trying to build a description for image (a) the apple is the most discriminative object. However, it is small and might be missed. On the other hand, although a chair exists in both images, it is much more salient in our target image. Therefore, if we choose to describe the chair instead of the apple, we should be able to distinguish the target image.

idea is that when people use a referring expression to describe an object, they have a certain preference in mentioning certain items over others. This preference order can be viewed as a queue in which all the items are waiting. By going through these items one by one, the speaker iteratively selects the ones that are discriminative enough under some criteria (for example, those that can eliminate more than n objects). Our goal is to construct the item queue from visual data. We do this by calculating a score for each detected item, and then sorting them in decreasing order.

3.1. Item Probability

The first property of the item we examine is its discriminability: given a set of images I, including our target image, we calculate the probability of the item being in this set. This obviously captures the discriminability of the item, since the lower the probability, the more images would be eliminated by including it. More specifically, we calculate the following probabilites:

$$p(cat_i|I) = \frac{|I_{o_i}|}{|I|}$$
$$p(rel_{r_{ij}}|I) = \frac{|I_{r_{ij}}|}{|I|}$$
$$p(col_{c_i}|I) = \frac{|o_{i_c}|}{|o_i|}$$

Where I_{o_i} is the set of images with an object from category i, $I_{r_{ij}}$ is the set of images with relationship r between objects of type i and j, o_{i_c} is the set of objects of type i with color c and o_i is the set of objects of type i.

3.2. Salience

We note that simply choosing the most discriminative item would not necessarily lead to the best discriminative description. This is because not all visual data are equal. There are many different properties of an item that might make it more or less useful in a description (cf. Fig. 3). We therefore use the following three measures for saliency inspired by Spain et al. [11]:

1. Size of the item. We normalize the size of each object by dividing by the size of the image.

2. Low level saliency of the object. A saliency map based on the work of Itti and Koch [6] implemented by [5].

3. Centrality of the item. The distance from the center normalized by the size of the image.

We calculate these values for each of the items. Since the relationship item involves two objects, we use the mean value of the two as the saliency feature for it. Taking this average can prove to be very useful under certain conditions. For example, in Fig. 3, the apple is the most discriminative item to describe image (a), since it does not appear in image (b). However, since it is very small, it might be missed. Since the apple is discriminative, the relationship "apple above table" is as discriminative, but has a much larger size score, because the size score is taken from the mean of the apple and the table. Therefore, this relationship might be ranked highest, and the description *"there is an apple above the table"* will be given. This will allow the listener to find the target image much quicker and perhaps avoid missing the apple all together.

3.3. Combining the scores

We formulate a score for each item based on its discriminativity and its salience. This score represents the importance of the specific item in the target image as related to the set of images I:

$$Score(IT_i) = (1 - p(IT_i|I)) + \alpha S(O_i) + \beta L(O_i) + \gamma C(O_i) \quad (1)$$

Where IT_i is an item, O_i is the object(s) that exist(s) in the item, $p(IT_i|I)$ is one of the probabilities as described in Sec. 3.1, and S, L, C are the size, low-level saliency, and centrality respectively. The parameters α, β, and γ are the weights given to the different saliency measures; these need to be adjusted to an optimal value. Too low a value may result in very non-salient items being chosen, which may be discriminative, yet easy to miss. At the same time, too high a value may cause the algorithm to choose items that are salient but not very discriminative. Although the users would be able to find those items quickly, they may exist in multiple images, and therefore not be of much help. We examine the effect of changing the parameters in Sec 6.2.

Our color classifier can produce erroneous results which can cause the user to make mistakes.Therefore, we use the probability score $P(c)$ that is given by the SVM to minimize these types of errors. We multiply $1 - P(c)$ by a fourth parameter δ, and subtract that from the score of the color items in equation 1. Colors for which the classifier gave

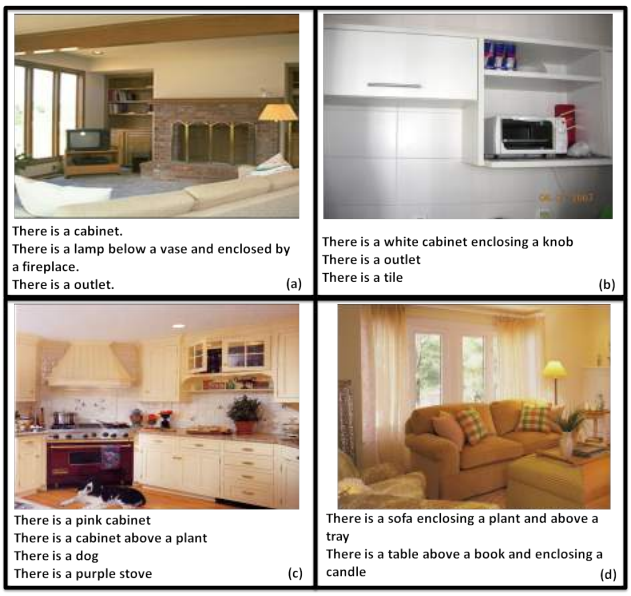

Figure 4. Four examples of the output of our discriminative description for different target images from different categories (living room (a)+(d), kitchen (b), bathroom (c)). Although the distractors are not shown here, each item described is chosen by being the most discriminative (no saliency).

a low probability (low confidence) will therefore get a low score and thus not be mentioned.

Once we have calculated all the scores, we rank the items based on the score in descending order. We then form a description of length n by choosing the n items with the top score. These items can be thought of as a set of n-tuples:

1. $\langle object \rangle$ a single for the object item
2. $\langle object, color \rangle$a double for the color item
3. $\langle object_1, relationship, object_2 \rangle$ a triple for the relationship item.

These n-tuples are then sent to a language generation algorithm in order to construct more natural english sentences.

4. Constructing the Sentences

Although we do not focus in this paper on the task of constructing perfect English sentences from the items we choose, we still need to perform some simple operations in order to make the sentences understandable and clear for experimentation. We follow a few very basic rules:

1. The first time an item is introduced, construct the sentence: *"there is a* $\langle ntuple \rangle$*"*
2. If an item has been introduced using the relationship or color item, remove the simple introduction of the item ($\langle object \rangle$) from the queue since it would be redundant if intoduced later.
3. When an object exists in more than one relationship, introduce an *"and"* between them and remove $object_1$ from the second triple.
4. Always place the color before the object it describes (even if the object has already been introduced).

There are a few obvious limitations to this approach, which can result in unnatural sentences. First, there is no notion of numbers in this method, and therefore if there are two objects of the same category it simply gets mentioned twice. In addition, there is no notion of continuity between sentences and therefore the transition between them appears unnatural. However, given all these limitations, the description is clear and concise in such a way that the necessary information is conveyed to our subjects. For an example of descriptions created by our algorithm, see Fig. 4.

5. Experiment Design

We ran an experiment with human subjects to measure how well the descriptions generated by our method can discriminate among a set of images. The experiment is conducted as follows. The computer first selects a random set of 10 images. Out of these, it then chooses a random target image which it tries to describe to the human subject. After detecting and ranking the items from the target image as discussed in sections 2 and 3, the algorithm presents the 10 images to the subject along with a description that includes only the top scored item. The subject is then required to select the correct image based on the description. If the subject is correct, the trial is over. However, if the subject selects a wrong image, the algorithm takes the next highest ranked item from the list and offers a new description that includes the top two items. This happens repeatedly until the subject selects the correct image, or there are no more items to describe in the image, or the subject has failed a certain number of times.

To examine the effects of different values for parameters α, β and γ, we needed to conduct a larger scale experiment. To that end, we adapted the experiment to work in Amazon Mechanical Turk, with a slight adjustment. The main difference is that each user only gets one chance at guessing the correct image, given a certain length of description. Whether or not the answer is correct, the next picture is then presented. Since we perform these tests with descriptions of different length, we are able to get a complete set of results from this style of testing.

To make the task more challenging and the choices less trivial, we select the distractor images to be of the same scene category as the target image. We end up using three scene categories from the indoor dataset: kitchen, bathroom and living room [10]. We use these scene categories because they contain many different object categories, as well as many object per scene. Thus, if the target image is a kitchen,

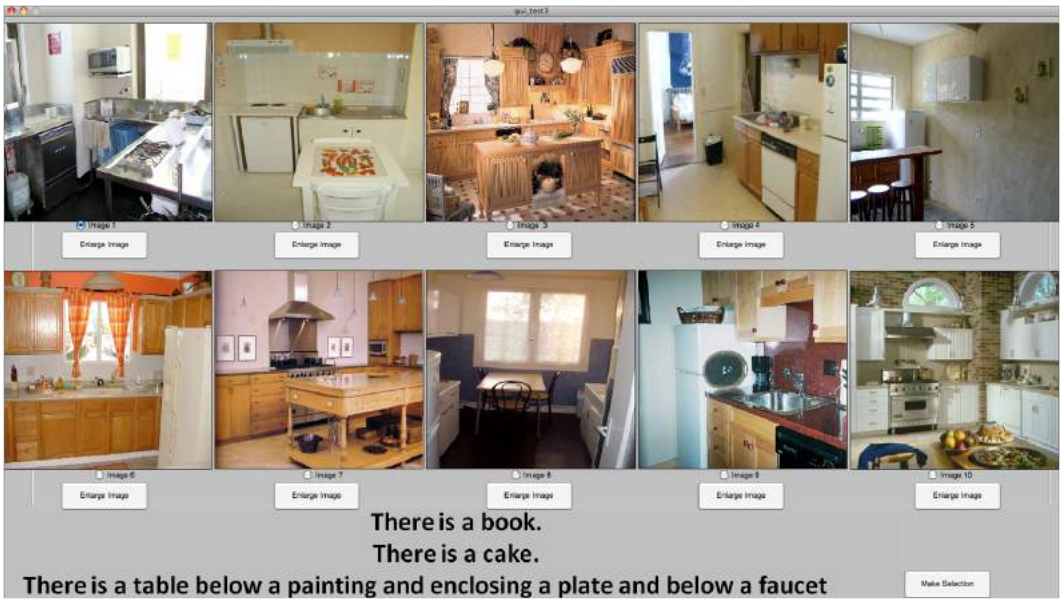

Figure 5. A screen shot of our experiment. The subject is given 10 images and instructed to select the one described by the algorithm. The subject also has the option of enlarging any of the images. In this specific case, the target image is the one on the bottom right.

all the distractors would be images of different kitchens. For a screenshot of our experiment, see Fig. 5.

6. Results & Discussion

We divide the description of our results into two sections. We first present the results of the experiments run in the lab. These experiments were used to verify that in our set-up, choosing the most discriminating items (regardless of the other parameters) for the description would allow people to choose the correct image given its length of description with higher percentage. We compare our results to a random selection, in which all the items in the image are ordered randomly in the queue and then chosen one by one.

6.1. Discriminating Description

The lab experiment was performed on 18 subjects, using the kitchen category. Each subject had 15 trials, where each trial is a set of one target image and nine distractors. On average, the subjects needed 2.5 guesses per trial. That is, since every time they guessed wrong they received a longer description for the same trial, they usually needed more than one guess.

The results, presented in Fig. 6, show that the discriminative selection yields better performance than random selection. For example, only 32.6% of subjects managed to guess the correct image given a random description with only one item, while 43.8% managed to guess correctly with our discriminative approach ($p = 0.059$). Although the curves do get closer and the difference less significant for certain numbers of items, the discriminative description always results in better performance.

There are a few reasons why the performance in the random description and the discriminative description conditions are relatively close. First, although the images we selected as distractors are all from the same category, they are usually different enough, such that even after a few items it is relatively easy to find the correct image. This is even more pronounced after the subject has already made an incorrect guess, since at that point he or she had already eliminated one of the images. For example, after a wrong first guess the subject's chances increase from 1:10 to 1:9.

In addition there is much noise in the different items, which may cause the discriminative description to be less effective than it can. One such problem stems from not all objects being labeled in all the images. For example, the object *'wall'* has not been labeled in many images, even though walls exist in all the indoor images that we use. Therefore, if the target image is the only one which has the label *'wall'*, this object will be the first to be described in our discriminative approach, even though it does not actually give any useful information to the subject.

There can also be errors in our color or relationship detector. These would cause more problems for a discriminative description than for a random description. Since an error in these detectors might create a very unlikely item, there is a high probability that it will be the first to be mentioned in the discriminative description, and might end up throwing the subject off. This is in contrast to the random approach, where this item might not be chosen to be described until later.

In Fig. 7, we also plot the average time it took for subjects to guess correctly, for each description length up to 3. From this plot, it is clear that people were able to guess the correct answer given our description 7 seconds quicker on

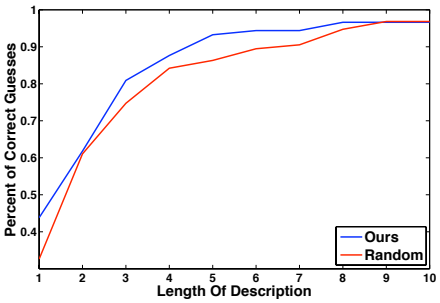

Figure 6. Results from our lab experiment. The x axis represents the number of items in a description, while the y axis represents the percentage of subjects who succeeded in guessing the correct image when less than x items were given.

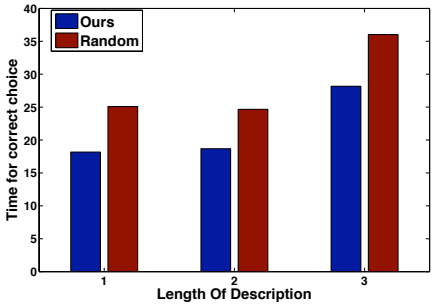

Figure 7. Average time to guess correctly. For each description length (up to three) we take only the subjects who have guessed correctly and calculate the average time they spent guessing at that description length.

average. This make sense, since if the items describing the image existed in only one or very few images, the subjects would need less time to choose the correct answer.

6.2. Parameter Evalutation

In this section we present the results from our experiments on Amazon's Mechanical Turk. During our study, we had 159 unique workers guessing 90 different sets of target/distractor images (30 for each parameter). For each target/distractor set, we created 4 different description lengths for each parameter setting. Since we examined three values for each parameter, and allowed 4 workers to work on every task, we ended up conducting a total of about $90 \times 4 \times 3 \times 4 = 4320$ tasks.

Since our focus was to test if the saliency measures we are using could improve efficiency, we conducted the following experiment. First, we selected manually images for which we expected these measures to make a difference. This allows us to show that these measures can actually be useful for discriminating between scenes. Second, instead of paying each person a constant sum we pay only \$0.01 per task, but then pay people who guess correctly an extra of \$0.02, thus tripling their reward amount.

Fig. 9 presents the results of this experiment. Although it has been conducted on a relatively small set of images,

Figure 8. Image examples of how saliency can assist in discriminating between images. The colors represent different values for the different parameters, while the graph shows the improvement in performance for each parameter for that specific image. (a) On the left of the image there is a small basket above the sink. This is very hard to notice. However, the plant next to the cabinet in the front-right of the image is much easier to see and therefore provides a 25% increase in guessing. (b) There is a cup in the middle of the image. However, since it is clear it has very few edges. Although the outlet is small it has a much higher saliency score and thus provides a 30% increase. (c) Although both the carpet and the curtain only existed in this image out of all the distractors, the curtain is centered, so it provided a 12% increase. (d) Although using the size parameter helps in choosing the carpet over the basket, if it is too high then too much weight is given to the size and it selects a non-discriminative item.

which were chosen specifically for this task, some interesting observations can be made. First, from all three graphs it is clear that all the parameters can be helpful in determining what are the most useful items to describe. This supports our initial assumption, by showing that in the case of visual scenes, mere discriminability will not always produce the best results. Each of the three factors seem to provide some benefit to the algorithm.

We examine how different people responded to the same target/distractor set given the different description, and find the ones in which the different saliency parameters made the most difference. Examples of these can be shown in Fig. 8. Fig. 9 (a) also shows an interesting effect of these parameters: if these end up being too high, they can make the description worse. This is fairly obvious, since the more weight we put on saliency, the more probable it is that a high ranked item might also appear in other images. For example, although a chair is much bigger than an apple, if it appears in many of the distractors it might reduce the

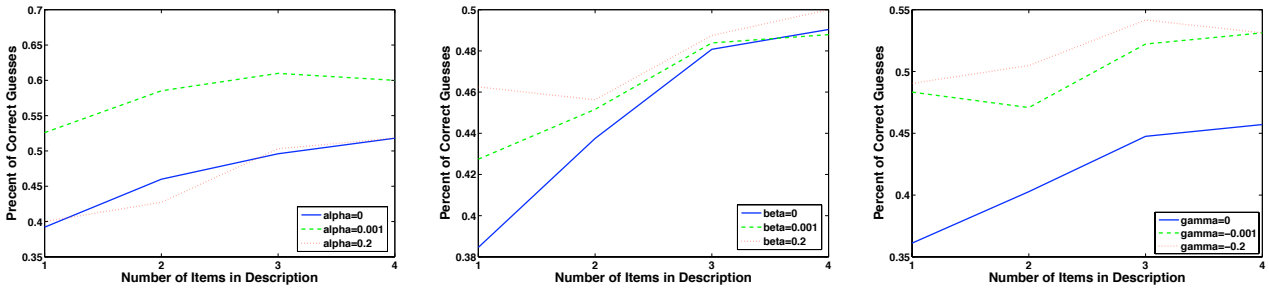

Figure 9. The results of our three Amazon Mechanical Turk experiments. In each experiment we examined the effect of one of the parameters and set the other two parameters to 0. (a) The effect of α which is the weight given to the size of the object. (b) The effect of β which is the weight given to the low level saliency of the object as described in the model by [6]. (c) The effect of γ which is the weight given to the centrality of the object.

probability of a correct guess. Although this effect does not show itself in the other two results, we expect that if we raise the parameter even higher the effect will be the same.

Another interesting observation is the performance increase for the different parameters. That is, both size and centrality seem to increase the performance around 15% while our saliency model only gives a 8% increase which reduces to just a few percent for descriptions longer than one item.

7. Future Work

Since generating discriminative descriptions for images has never been attempted before, there are many possible extensions to this work. For example, we plan to collect human-generated discriminative descriptions using Amazon's Mechanical Turk. The basic idea would be to use the same data set of labeled images in a similar setting, but instead of requiring the subjects to find the target image, they would be provided with the image, and would need to generate a description. By analyzing the statistics of what they chose to describe, in relation to the objects that appear in the image, we should be able to build a more reliable model.

An additional extension can involve looking at more general descriptions that are not task specific. It has been shown in the past that people tend to name the same objects in an image relatively consistently when not presented with a definite task [11]. It would be interesting to examine how people choose what to describe (not only objects, but relationships and colors as well) given a general task of describing an image, and then try to build a model to replicate that.

There are other properties in the image that we have not examined in this paper. On the object level, there are many more attributes that can be described. On the scene level, the scene category, lighting, and coloring might be of use. Finally, it may be possible to infer attributes such as actions or feelings from the image. How to integrate all these details into one coherent description remains an open problem.

References

[1] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011. 3

[2] R. Dale and E. Reiter. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive Science*, 1995. 3

[3] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. 2009. 3

[4] A. Farhadi, S. M. M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. A. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, 2010. 2

[5] J. Harel. A saliency implementation in matlab. http://www.klab.caltech.edu/~harel/share/gbvs.php. 4

[6] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1998. 4, 8

[7] E. Krahmer and K. van Deemter. Computational generation of referring expressions: A survey. 2011. 3

[8] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011. 2, 3

[9] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. CoNLL '11, 2011. 3

[10] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, 2009. 3, 5

[11] M. Spain and P. Perona. Some objects are more equal than others: Measuring and predicting importance. In *ECCV*, 2008. 2, 3, 4, 8

[12] J. Van De Weijer and C. Schmid. Applying color names to image description. In *ICIP*, 2007. 3

[13] B. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu. I2t: Image parsing to text description. *Proceedings of the IEEE*, 2010. 2

[14] Y. Zhang and T. Chen. Object color categorization in surveillance videos. In *ICIP*, 2011. 3