Evaluating the WordsEye Text-to-Scene System: Imaginative and Realistic Sentences

Morgan Ulinski, Bob Coyne, Julia Hirschberg
Department of Computer Science
Columbia University
New York, NY, USA
{mulinski, coyne, julia}@cs.columbia.edu

Abstract

We describe our evaluation of the WordsEye text-to-scene generation system. We address the problem of evaluating the output of such a system vs. simple search methods to find a picture to illustrate a sentence. To do this, we constructed two sets of test sentences: a set of crowdsourced *imaginative sentences* and a set of *realistic sentences* extracted from the PASCAL image caption corpus (Rashtchian et al., 2010). For each sentence, we compared sample pictures found using Google Image Search to those produced by WordsEye. We then crowdsourced judgments as to which picture best illustrated each sentence. For imaginative sentences, pictures produced by WordsEye were preferred, but for realistic sentences, Google Image Search results were preferred. We also used crowdsourcing to obtain a rating for how well each picture illustrated the sentence, from 1 (completely correct) to 5 (completely incorrect). WordsEye pictures had an average rating of 2.58 on imaginative sentences and 2.54 on realistic sentences; Google images had an average rating of 3.82 on imaginative sentences and 1.87 on realistic sentences. We also discuss the sources of errors in the WordsEye system.

Keywords: text-to-scene, evaluation, corpus creation, crowdsourcing

1. Introduction

WordsEye (Coyne and Sproat, 2001) is a system for automatically converting natural language text into 3D scenes representing the meaning of that text. WordsEye supports language-based control of spatial relations, spatial properties, surface textures and colors, and cardinality; it handles simple anaphora and coreference resolution, allowing for a variety of ways to refer to objects and describe scenes. Scenes are assembled from a library of 3,000 3D objects and 10,000 2D images tied to a lexicon of 15,000 nouns. WordsEye is a web application (http://www.wordseye.com) with 27,000 registered real-world users. During the two year period from November 2015 to December 2017, approximately 2,200 users posted 14,000 finished scenes to an online gallery.

One task that WordsEye addresses is the problem of creating vs. automatically finding a picture to illustrate a sentence. Standard image search engines are limited to pictures that already exist in their databases, biasing them toward retrieving images of mundane and real-world scenarios. In contrast, a scene generation system like WordsEye can illustrate a much wider range of images, allowing users to visualize unusual and fantastical scenes. When users are freed from the normal constraints of what is possible or already exists they will often describe what they imagine – from situational, to iconic, to abstract, to fantastical. The majority of scenes created by actual users of the online WordsEye system are imaginative. One user commented “I truly enjoy watching people unleash their minds here.” Some examples of imaginative scenes that have been created in WordsEye are shown in Figure 1.

The ability to generate both realistic and imaginative scenes from text input demonstrates that text-to-scene generation systems such as WordsEye can be used to supplement the results of image search engines such as Google. In evaluating WordsEye vs. image search engines, we therefore compare *imaginative sentences* and *realistic sentences* as items to be visualized. We use crowdsourcing to collect imaginative sentences and extract realistic sentences from the PASCAL image caption corpus (Rashtchian et al., 2010).

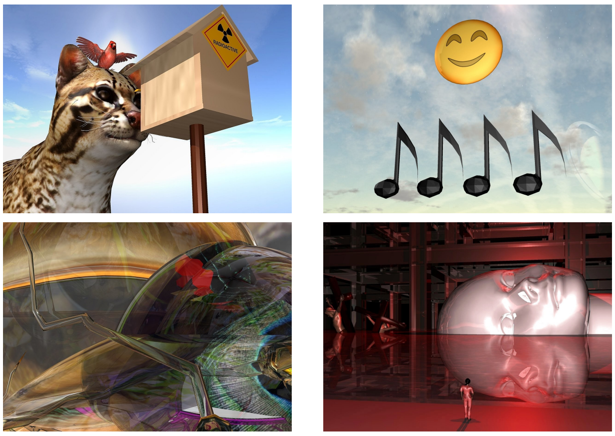


Figure 1: Imaginative Images: Situational, Iconic, Abstract, Fantastic

In Section 2. we discuss related work. In Section 3. we introduce the WordsEye text-to-scene system. In Section 4. we describe the construction of *imaginative* sentences and *realistic* sentences for system evaluation. In Section 5. we explain the collection of potential illustrations for these, using Google Image Search or WordsEye. In Section 6., we discuss the use of crowdsourcing to evaluate the illustrations. We discuss the results of the evaluation in Section 7. and conclude in Section 8..

2. Related Work

Several systems exist for producing graphics from natural language sources. Glass and Bangay (2008) describe a system for transforming text sourced from popular fiction into

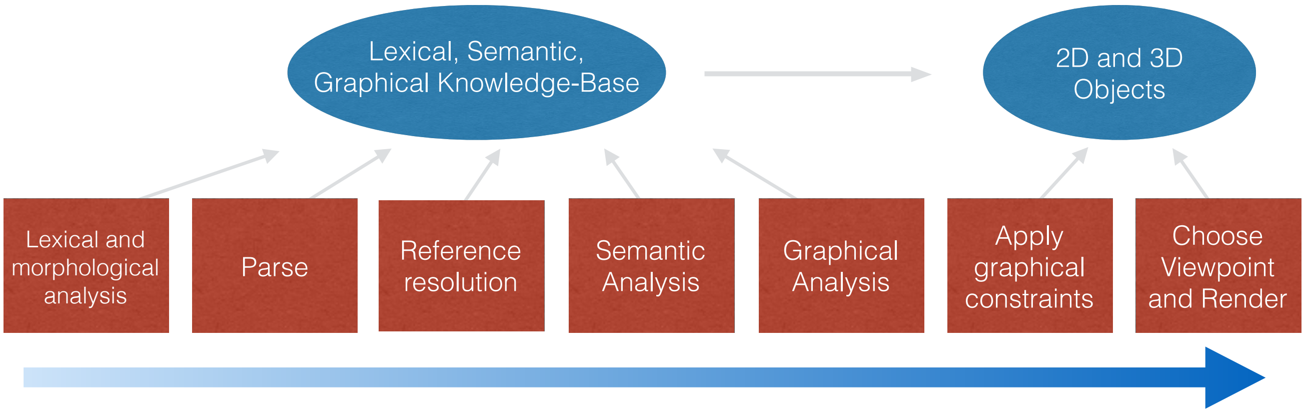

Lexical, Semantic, Graphical Knowledge-Base
2D and 3D Objects
Lexical and morphological analysis
Parse
Reference resolution
Semantic Analysis
Graphical Analysis
Apply graphical constraints
Choose Viewpoint and Render

Figure 2: WordsEye System Architecture

corresponding 3D animations without prior language simplification. 3SVD (Zeng et al., 2005) is a 3D scene creation system using story-based descriptions. Parisi et al. (2007) describe an ontology-driven generation of 3D animations for training and maintenance. CONFUCIUS (Ma, 2006) is a multimodal text-to-animation system that generates animations of virtual humans from single sentences containing an action verb. In all these systems the referenced objects, attributes, and actions are typically relatively small in number or targeted to specific pre-existing domains. Seversky and Yin (2006) is a system tailored for interactively generating 3D scenes from natural language and voice and text input where the user can adjust the scene they create. Chang et al. (2014) created a text-to-scene system focused on learning spatial relations by supplying examples of indoor scenes and letting the user teach it by adjusting the scene to match the text. A survey of these and other text-to-graphics systems is given in Hassani and Lee (2016).

Others have used crowdsourcing to collect human-generated sentences, e.g. to create image captions. This includes the PASCAL image caption corpus (Rashtchian et al., 2010), Flickr8k (Hodosh et al., 2013) and Microsoft COCO (Chen et al., 2015). Our work differs in that we want to collect sentences describing anything users can imagine, as opposed to descriptions of existing photographs.

Zitnick and Parikh (2013) crowdsourced the evaluation of their scene generation system using 2D clip art: subjects created an initial set of scenes and wrote descriptions of the scenes. Zitnick et al. (2013) used several methods to automatically generate scenes for these descriptions and asked subjects which picture matched the description better. While the pictures that the sentences describe are human-constructed scenes rather than photographs from sources like Flickr, the scenes use a fixed set of 80 objects and are limited to the domain of children playing outside. Chang et al. (2015) evaluate their text-to-scene system by asking people to rate the degree to which scenes match the text used to generate them. Their test corpus includes a much larger number of objects than Zitnick et al. (2013), but the sentences and scenes are realistic descriptions of the configuration of objects in a room.

The WordsEye text-to-scene system was previously evaluated as an educational tool, as a means of helping students develop language skills. We found that students using the system had significantly greater improvement in their literary character and story descriptions in pre- and post-test essays compared with a control. In this paper, we focus on evaluating the pictures produced by the system more directly as accurate illustrations of input sentences.

3. The WordsEye Text-to-Scene System

In this section, we provide more details on the WordsEye text-to-scene generation system. WordsEye includes a library of approximately 3,000 3D objects and 10,000 2D images tied to a lexicon of 15,000 nouns. These include a wide variety of common objects (including variations of the same basic type, such as different types of doors or chairs) and textures (e.g. wood, grass, granite). WordsEye also supports several dozen graphical primitives and properties that are used for spatial relations (different senses of “in”, “on”, lateral relations, etc.), spatial properties (absolute and relative sizes and aspect ratios), and surface properties (colors, opacity, reflectivity, etc.). These primitives in conjunction with the objects and semantic knowledge about those objects (such as defaults for size, orientation, and top surface regions) allow the scene to be composed.

The system operates by first tokenizing each input sentence into lexical items (including modifiers like contractions or possessives) and possible parts-of-speech. The tokens are parsed into a labeled syntactic dependency structure. The dependency structure is then processed for anaphora and other co-reference, which is especially important for depicting multi-sentence input. These resolved structures are converted to lexical-semantic relations using lexical valence patterns and other lexical and semantic information. The resulting semantic relations are converted to a set of graphical constraints, representing the position, orientation, size, color, texture, cardinality, and poses of objects in the scene. The graphical constraints (other than poses, which currently are ignored) are applied to construct a fully specified 3D scene which is then rendered. Throughout this process, the system relies on a knowledge-base of lexical, semantic, and graphical information and a library of 2D and 3D objects. The architecture is shown in Figure 2.

4. Elicitation and Selection of Sentences

In this section we describe our use of crowdsourcing to collect imaginative sentences and filtering the PASCAL image caption corpus (Rashtchian et al., 2010) to obtain realistic sentences.

Category	Definition	Examples
PROP	Small objects that could be held or carried	*cellphone, apple, diamond*
FIXTURE	large objects such as furniture, vehicles, plants	*couch, sailing ship, oak tree*
ANIMAL	Animals	*dolphin, chicken, llama*
SPATIAL TERM	terms representing spatial relations	*above, against, facing, on*
NUMBER	small numbers	*one, four, nine, twelve*
COLOR	common colors	*beige, green, scarlet, black*
SIZE	general size or specific dimensions	*big, tiny, thin, 5 feet long*
DISTANCE	distances	*4 inches, five meters, 10 feet*
SURFACE PROPERTY	surface properties	*opaque, shiny, transparent*
LOCATION	terms representing terrain types and locations	*field, driveway, lake, forest*
BUILDING	buildings and architectural structures	*doghouse, castle, skyscraper*

Figure 3: Categories of words in the lexicon

AMT Column Headings	WordsEye Lexical Categories	
1.Noun1, Noun2, Spatial Term, Adjective 2.Adjective, Noun1, Noun2, Spatial Term	Noun1 is PROP. Noun2 is FIXTURE.	Spatial Term is SPATIAL TERM. Adjective is SIZE, COLOR or SURFACE PROPERTY. Distance is DISTANCE. Location is BUILDING or LOCATION. Size is SIZE. Color is COLOR. Number is NUMBER.
3.Adjective, Noun1, Noun2, Spatial Term	Noun1 is ANIMAL. Noun2 is FIXTURE.	
4.Noun1, Noun2, Spatial Term, Distance, Adjective	Noun1, Noun2 are PROP, FIXTURE or ANIMAL.	
5.Noun1, Noun2, Spatial Term, Location	Noun1 is ANIMAL. Noun2 is PROP or FIXTURE.	
6.Noun1, Noun2, Spatial Term, Distance, Adjective 7.Adjective, Noun1, Noun2, Spatial Term, Distance 8.Noun1, Noun2, Spatial Term, Location 9.Noun1, Noun2, Spatial Term, Color, Size 10.Noun1, Noun2, Spatial term, Number 11.Noun1, Noun2, Spatial term, Number, Adjective 12.Adjective, Noun1, Noun2, Spatial Term, Number	Noun1, Noun2 are PROP or FIXTURE.	

Figure 4: Possible combinations of categories for the sentence construction task

4.1. Imaginative Sentences

We used Amazon Mechanical Turk to obtain imaginative sentences for our evaluation. We gave Turkers short lists of words divided into several categories and asked them to write a short sentence using at least one word from each category. The words provided to the Turkers represent the objects, properties, and relations supported by the text-to-scene system.

To help Turkers construct sentences of different types, we organized the objects, properties, and relations into a few basic categories. The categories are listed in Figure 3. We restricted the lexicon to include only commonly known words that could be easily understood and recognized visually. We excluded super-types such as "invertebrate" and sub-types such as "european elk". We omitted obscure terms such as "octahedron" or "diadem". The resulting lexicon included about 1500 terms and phrases.

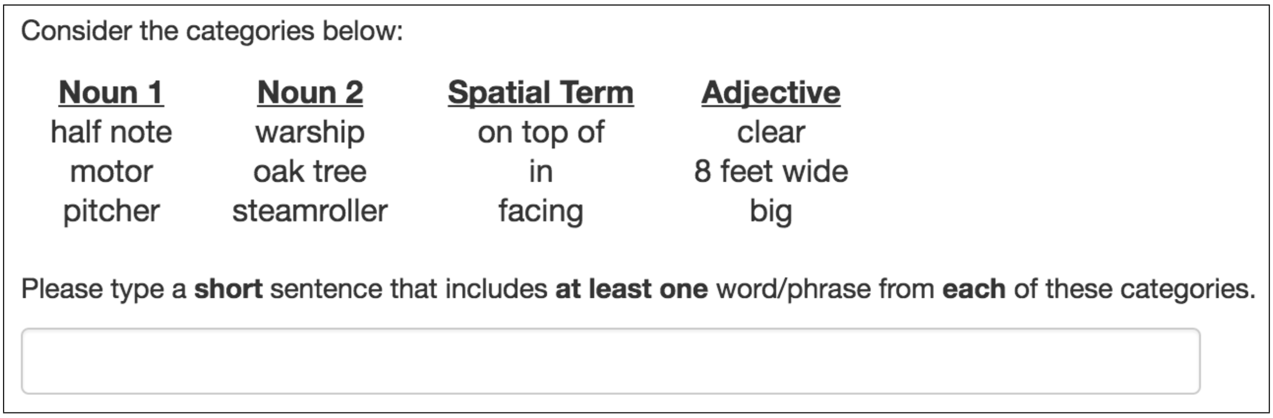
We created 12 different combinations of categories with 20 HITs per combination. Each HIT randomly presented different words for each category in order to elicit different types of sentences from the Turkers. This involved varying the types and number of categories as well as the order of the items in the categories. We wanted to encourage sentences such as "there is a blue dog on the large table" as well as different orders and constructs like "the dog on the large table is blue". Each HIT showed 4 or 5 categories, with three words per category. Figure 4 shows all the combinations of categories.

Our instructions specified that Turkers write one sentence using a maximum of 12 words. Words could be in any order as long as the resulting sentence was grammatical. We allowed the use of any form of a given word; for example, using a plural noun instead of a singular. We also allowed the use of *filler words* not listed in the categories, but asked Turkers not to add any unlisted *content words*. We defined *filler words* as words with "little meaning on their own, but that are used to make the sentence grammatical (e.g. *the*, *has*, *is*, *with*)" and *content words* as words that "refer to an object, action, or characteristic (e.g. *eat*, *shallow*, *organization*)." An example HIT is shown in Figure 5.

We restricted our task to workers who had completed at least 100 HITs previously with an approval rate of at least 98%. We paid $.04 per assignment. We started with 240 unique combinations of words and collected one sentence for each of these. After filtering out ungrammatical sentences, we ended up with a total of 209 imaginative sentences. Some examples are shown in Figure 6(a).

4.2. Realistic Sentences

We began with image captions collected by Rashtchian et al. (2010) for the PASCAL Data Set (Everingham et al., 2011), which consists of 5000 descriptive sentences, 5 cap-



Consider the categories below:

Noun 1	Noun 2	Spatial Term	Adjective
half note	warship	on top of	clear
motor	oak tree	in	8 feet wide
pitcher	steamroller	facing	big

Please type a **short** sentence that includes **at least one** word/phrase from **each** of these categories.

Figure 5: Example of sentence collection HIT

(a) Imaginative

- *The huge jewel is in front of the red rolling pin.*
- *Five pears are under the martini glass.*
- *The large prawn is on top of the stool.*
- *The red clock is three feet above the desk.*
- *Two tulip trees are close to a seashell.*

(b) Realistic

- *A brown duck and white duck stand on the grass.*
- *A man is standing next to a yellow sports car.*
- *A black dog in a grass field.*
- *The big white boat is in the ocean.*
- *A child sits in a large black leather chair.*

Figure 6: Examples of imaginative and realistic sentences

tions each for 1000 images. The images cover 20 object categories from the original PASCAL task, including *people, animals, vehicles*, and *indoor objects*. We used at most a single caption for each photograph.

To select a usable caption, we manually removed all ungrammatical sentences and fed the remaining sentences into WordsEye, which was able to create scenes for about one third of the image captions; the captions that were rejected were omitted due to current limitations of the system's lexicon, object library, or parser. Since our goal is to evaluate WordsEye we excluded these sentences which are outside the domain of the system. For example we omitted most sentences using action verbs since the system currently cannot pose characters to represent those verbs. We kept simple stative pose-related verbs such as "sit" and "stand" so the system could capture other aspects of the sentence. We also omitted sentences that could not be parsed or that had concrete nouns with no corresponding 3D object. This resulted in a total of 250 realistic sentences. Some examples are shown in Figure 6(b).

5. Collection/Generation of Illustrations

In this section, we describe how we obtained the possible illustrations for each sentence.

Google Image Search: We used each sentence as a query for Google image search. We did not strip punctuation, add quotation marks, or otherwise modify sentences. The first 4 results were downloaded and resized to a uniform width.

WordsEye Scene Generation: Since WordsEye images are rendered 3D scenes, they can be easily viewed from different angles. Normally, users can interactively change the viewpoint in the scene they are creating and choose the best view. So our approach was to automatically generate four WordsEye scenes with slightly different camera views. If one of the objects was occluded by another (and hence not visible in a front view of the scene), we automatically produced an alternate view of the scene from the back. Likewise, the elevation of the camera was varied to allow an object to potentially be more visible. We randomized the objects chosen in the scene from those compatible with the sentence. For example, Figure 7 shows the four scenes generated for the sentence "A furry dog lying on a glass table."

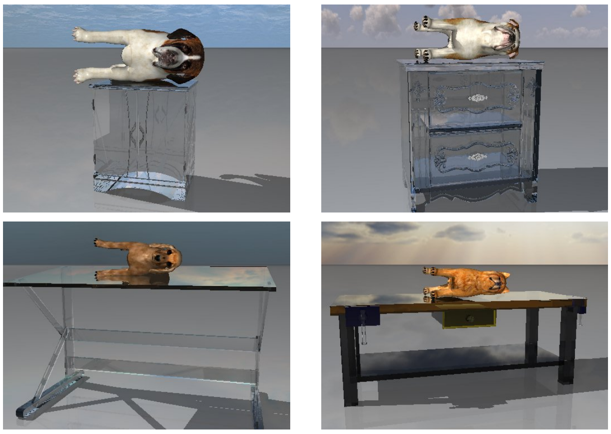
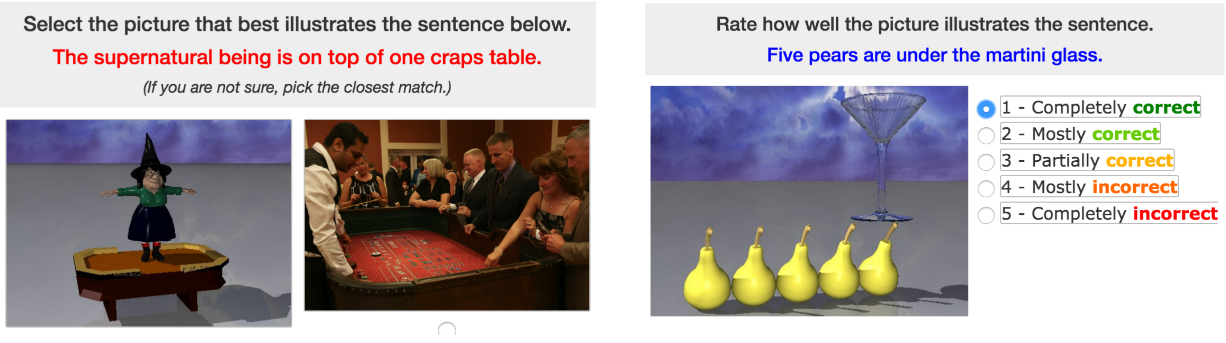


Figure 7: Generated scenes for the sentence "A furry dog lying on a glass table."

6. Evaluating Illustrations with AMT

The evaluation of the quality of the illustrations was done in two phases. In the first phase we asked Turkers to determine the best image for each sentence from the downloaded Google results and (separately) for each sentence among the Wordseye-generated images. In the second phase, Turkers evaluated the quality of the best Google image and the best WordsEye image. We did this second phase evaluation with two separate crowdsourced tasks. In the first, we asked Turkers to compare the best Google image with the best WordsEye image directly. In the second, we obtained a rating for how well each of the images illustrated the sentence. For all tasks, we required turkers to have previously



Select the picture that best illustrates the sentence below.

The supernatural being is on top of one craps table.

(If you are not sure, pick the closest match.)

○

Rate how well the picture illustrates the sentence.

Five pears are under the martini glass.

● 1 - Completely correct
○ 2 - Mostly correct
○ 3 - Partially correct
○ 4 - Mostly incorrect
○ 5 - Completely incorrect

(a) (b)

Figure 8: Examples of the second phase AMT tasks: (a) image comparison task and (b) rating task. (Google image source: `https://en.wikipedia.org/wiki/Craps`)

completed at least 500 HITs and to have a 98% approval rate. We paid $0.01 per assignment.

Each image comparison HIT showed a single sentence with the possible images below it. Turkers were asked to select the picture that best illustrated the sentence. In the first phase, we showed four pictures and collected 5 judgments for each HIT. In case of ties, we published additional assignments for that sentence until one image had more votes than any of the others. The image that received the most votes was used in the next phase, which compared the winning Google image with the winning WordsEye image. In the second phase, we collected 3 judgments for each HIT, which guaranteed no ties. A sample HIT from the second phase is shown in Figure 8(a).

For the rating task, each HIT showed a single sentence and a single image. Turkers were asked to rate how well the picture illustrated the sentence. The scale was from 1 (completely correct) to 5 (completely incorrect). We collected 3 judgments for each HIT and averaged these ratings to obtain the final rating for each picture. An example of the rating HIT is shown in Figure 8(b).

Winner (votes)	**Imaginative**	
WordsEye (3 to 0)	60.3% (126)	**85.6% (179)**
WordsEye (2 to 1)	25.4% (53)	
Google (2 to 1)	10.0% (21)	14.4% (30)
Google (3 to 0)	4.3% (9)	
Total	100.0% (209)	

(a)

Winner (votes)	**Realistic**	
WordsEye (3 to 0)	8.8% (22)	16.4% (41)
WordsEye (2 to 1)	7.6% (19)	
Google (2 to 1)	14.4% (36)	**83.6% (209)**
Google (3 to 0)	69.2% (173)	
Total	100.0% (250)	

(b)

Figure 9: Distribution of Turkers' Votes for WordsEye vs. Google Images for (a) imaginative sentences and (b) realistic sentences.

	Imaginative	**Realistic**
WordsEye	**2.581**	2.536
Google	3.825	**1.871**

(a)

Winner	**Imaginative**	**Realistic**
WordsEye	**74.6% (156)**	25.6% (64)
Tie	5.3% (11)	13.6% (34)
Google	20.1% (42)	**60.8% (152)**
Total	100.0% (209)	100.0% (250)

(b)

Figure 10: (a) Avg. ratings for WordsEye and Google images. (b) Distribution of winner based on ratings.

7. Results and Discussion

In this section, we discuss results from the second phase of evaluation. In the image comparison task, we asked 3 Turkers to choose the picture that best illustrated the sentence. The distribution of outcomes is shown in Figure 9. The winner is shown in bold for each category.

Next, we obtained a rating for each image from 1 (completely correct) to 5 (completely incorrect). Figure 10(a) shows average ratings for Google and WordsEye for each category of sentence, with the better rating in each category shown in bold. We also calculated the winning image for each category based on the ratings. For each sentence, the winning image was the one with the lower rating. These are shown in Figure 10(b), with the winner for each category shown in bold.

The trend for both votes and ratings is the same: WordsEye is superior for imaginative sentences and Google for realistic sentences. The winning image based on votes is not always the same as the winner based on rating. Figure 11 compares the distribution based on ratings and votes.

For imaginative sentences, when the Google and WordsEye ratings were tied, WordsEye tended to win the votes. Even when Google had a better rating than WordsEye, WordsEye still tended to win by votes. In particular, out of the 42 cases where the Google image received a better rating, Turkers chose the WordsEye image for 24 (more than half) of them. This pattern is reversed for the realistic sentences. For realistic sentences, when both images had the same rating, Turkers tended to choose the Google image. However,

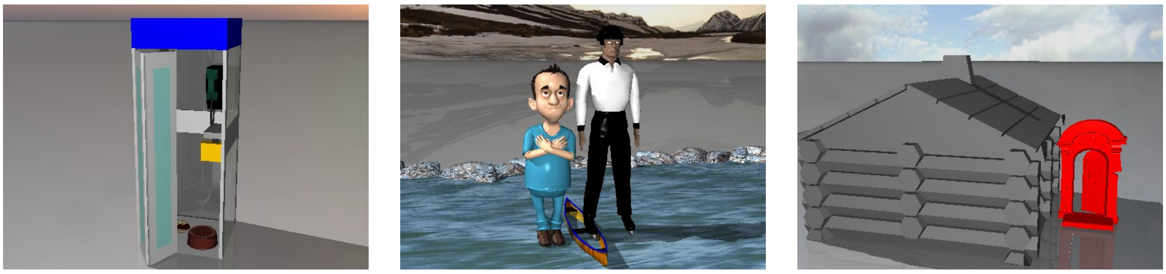
	WordsEye won rating	Tie rating	Google won rating	Total
WordsEye won votes	70.3% (147)	3.8% (8)	11.5% (24)	85.6% (179)
Google won votes	4.3% (9)	1.4% (3)	8.6% (18)	14.4% (30)
Total	74.6% (156)	5.3% (11)	20.1% (42)	100.0% (209)

(a) Imaginative sentences

	WordsEye won rating	Tie rating	Google won rating	Total
WordsEye won votes	14.0% (35)	0.8% (2)	1.6% (4)	16.4% (41)
Google won votes	11.6% (29)	12.8% (32)	59.2% (148)	83.6% (209)
Total	11.6% (64)	13.6% (34)	60.8% (152)	100.0% (250)

(b) Realistic sentences

Figure 11: Distribution of winner (WordsEye vs Google) based on ratings and votes.



(a) (b) (c)

Figure 12: Example WordsEye errors: (a) camera viewpoint partial occlusion: (*The hotdog is next to the chocolate cake in the booth.)* and (b) graphical interpretation and knowledge-base: (*Two men in a small wooden canoe on the water*) and (c) semantic interpretation: (*a gray house with a red door*)

when WordsEye had the better rating for a realistic sentence, Turkers still tended to choose the WordsEye image. Thus, while Turkers seemed to prefer to associate imaginative sentences with WordsEye-style pictures when forced to make a binary choice (even when the Google image had a lower rating), the reverse bias does not hold for realistic sentences: when an WordsEye image illustrated a realistic sentence better based on rating, the binary choices made by Turkers usually favored the WordsEye image as well.

7.1. Error Analysis

In this section we examine the sources of errors in the WordsEye system. One common cause of errors was a poorly placed camera (3D viewpoint). This was especially an issue for imaginative sentences which could involve very small objects in the same scene with large ones, making it hard to see both at the same time, given our default algorithm for positioning the camera to frame the full scene. A better strategy would be to position the camera aimed to frame the small object with the large object in the background. In other cases, one object was inside another (e.g. within an enclosed area such as a building) and the default generated camera positions were outside the building, making it impossible to see the inner object. Another source of errors was from missing graphical primitives. For example, sentences that required a person or animal to be in a particular pose (e.g. sitting) are internally represented, but the system is currently unable to actually put the 3D character into a pose. A third source of errors was in anaphora resolution in text like *A field with many black cows in it.* The WordsEye system currently processes anaphora and other co-reference across sentences but not within a sentence. Other errors occurred because of incorrect information stored in the knowledge base (e.g. incorrect real-world sizes resulting in strange relative sizes between objects in a scene) or from incorrect or unexpected semantic and graphical interpretations.

A description of the kinds of errors that could occur in each WordsEye module (see Figure 2) is presented here. Examples of some of these are shown in Figure 12.

- **Knowledge Base:** missing lexical entry or word sense; incorrect object (or part) properties.
- **Graphics Library:** missing or unrepresentative 3D object.
- **Parsing:** problem with syntax or punctuation.
- **Reference resolution:** unresolved anaphora.
- **Semantic analysis:** syntax-to-semantic conversion, including object selection and ambiguity.
- **Graphical analysis:** incorrect graphical interpretation (backgrounds, materials, spatial layout).
- **Apply graphical constraints:** spatial constraint maintenance issues, missing graphical primitives.
- **Camera/Render:** camera angle, zoom, occlusions.

We tagged each WordsEye image that had a rating worse than 2 with the type of error it exhibited and the WordsEye module where the error occurred. The WordsEye pictures for 114 imaginative sentences and 137 realistic sentences were tagged with errors. Figure 13 shows the distribution of errors per module. Note that since some pictures were tagged with errors from multiple modules, the total of each column is greater than 100%. WordsEye made

WordsEye Module	Imaginative	Realistic
Knowledge Base	10.0% (21)	2.4% (6)
Graphics Library	3.3% (7)	6.0% (15)
Parsing	2.9% (6)	2.0% (5)
Reference resolution	0.5% (1)	2.0% (5)
Semantic analysis	3.8% (8)	15.2% (38)
Graphical analysis	10.0% (21)	25.6% (64)
Apply graphical constraints	4.3% (9)	21.2% (53)
Camera/Render	42.1% (88)	6.8% (17)
No Error	45.5% (95)	45.2% (113)

Figure 13: Errors per module. Note: a given sentence could have more than one error.

more knowledge-base errors and camera errors on imaginative sentences. It made more semantic analysis, graphical analysis, and apply graphical constraints errors on realistic sentences.

8. Summary

We have described our evaluation of the WordsEye text-to-scene system; specifically, we have evaluated WordsEye's ability to create a picture that illustrates *imaginative* and *realistic* sentences, as compared to traditional image search methods (i.e. Google search). We found that WordsEye performs very similarly on both kinds of sentences (average rating of 2.581 and 2.536, respectively - on our rating scale, halfway between "mostly correct" and "partially correct"). While Google search does perform better than WordsEye on realistic sentences (average rating of 1.871 - between "completely correct" and "mostly correct"), performance breaks down when faced with imaginative sentences (average rating of 3.825 - between "partially correct" and "mostly incorrect"). Thus, we have shown that WordsEye is superior for imaginative sentences, and Google search is superior for realistic sentences. While this result is not unexpected, we can now quantify what the gap in performance actually is. In particular, while the average rating of WordsEye on realistic sentences was just 0.665 below that of Google, WordsEye's ratings on imaginative sentences was 1.244 higher than Google's. This suggests that as WordsEye and text-to-scene technology in general improve, they may become a viable alternative to image search even for realistic sentences, but it might be difficult to adapt traditional image search techniques to retrieve illustrations for imaginative sentences. In addition, as sentences get longer and more complicated (or if multiple sentences are involved), Google might be begin to have more trouble with realistic sentences as well.

Creativity is something that too often gets overlooked in technology development, and our results show that research into text-to-scene generation could play an important role in addressing the issue. Our new corpus of imaginative sentences may also have applications for other researchers studying language in a visual context or those interested in spatial language in general.

9. Bibliographical References

Chang, A. X., Savva, M., and Manning, C. D. (2014). Semantic parsing for text to 3D scene generation. In *Association for Computational Linguistics (ACL) Workshop on Semantic Parsing*.

Chang, A. X., Monroe, W., Savva, M., Potts, C., and Manning, C. D. (2015). Text to 3d scene generation with rich lexical grounding. *CoRR*, abs/1505.06289.

Chen, X., Fang, H., Lin, T., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. (2015). Microsoft COCO captions: Data collection and evaluation server. *CoRR*, abs/1504.00325.

Coyne, B. and Sproat, R. (2001). WordsEye: An automatic text-to-scene conversion system. In *SIGGRAPH*.

Everingham, M., Van Gool, L., Williams, C., Winn, J., and Zisserman, A. (2011). The pascal visual object classes challenge results (voc2011).

Glass, K. and Bangay, S. (2008). Automating the creation of 3d animation from annotated fiction text. In *IADIS 2008: Proceedings of the International Conference on Computer Graphics and Visualization 2008*, pages 3–10. IADIS Press.

Hassani, K. and Lee, W.-S. (2016). Visualizing natural language descriptions: A survey. *ACM Computing Surveys (CSUR)*, 49(1):17.

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, May.

Ma, M. (2006). *Automatic conversion of natural language to 3D animation*. Ph.D. thesis, University of Ulster.

Parisi, S., Bauch, J., Berssenbrugge, J., and Radkowski, R. (2007). Ontology-driven generation of 3d animations for training and maintenance. In *2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*, pages 608–614, April.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. (2010). Collecting image annotations using amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, pages 139–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Seversky, L. M. and Yin, L. (2006). Real-time automatic 3d scene generation from natural language voice and text descriptions. In *Proceedings of the 14th ACM International Conference on Multimedia*, MM '06, pages 61–64, New York, NY, USA. ACM.

Zeng, X., Mehdi, Q. H., and Gough, N. E. (2005). From visual semantic parameterization to graphic visualization. In *Ninth International Conference on Information Visualisation (IV'05)*, pages 488–493, July.

Zitnick, C. L. and Parikh, D. (2013). Bringing semantics into focus using visual abstraction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June.

Zitnick, C. L., Parikh, D., and Vanderwende, L. (2013). Learning the visual interpretation of sentences. In *The IEEE International Conference on Computer Vision (ICCV)*, December.