Scene Image Synthesis from Natural Sentences Using Hierarchical Syntactic Analysis

Tetsuaki Mano
Graduate School of Information Science and Technology
The University of Tokyo
mano@mi.t.u-tokyo.ac.jp

Hiroaki Yamane
Graduate School of Information Science and Technology
The University of Tokyo
yamane@mi.t.u-tokyo.ac.jp

Tatsuya Harada
Graduate School of Information Science and Technology
The University of Tokyo
harada@mi.t.u-tokyo.ac.jp

ABSTRACT

Synthesizing a new image from verbal information is a challenging task that has a number of applications. Most research on the issue has attempted to address this question by providing external clues, such as sketches. However, no study has been able to successfully handle various sentences for this purpose without any other information. We propose a system to synthesize scene images solely from sentences. Input sentences are expected to be complete sentences with visualizable objects. Our priorities are the analysis of sentences and the correlation of information between input sentences and visible image patches. A hierarchical syntactic parser is developed for sentence analysis, and a combination of lexical knowledge and corpus statistics is designed for word correlation. The entire system was applied to both a clip-art dataset and an actual image dataset. This application highlighted the capability of the proposed system to generate novel images as well as its ability to succinctly convey ideas.

CCS Concepts

•**Information systems** → **Multimedia content creation;**
•**Computing methodologies** → **Image processing;**

Keywords

Syntactic Abstraction, Word Mapping, Image Synthesis

1. INTRODUCTION

What if a visual expression can be successfully synthesized from linguistic information? The relevant technology would spark groundbreaking applications, such as the automatic generation of illustrated pictures, language translation with images as intermediaries, and so forth.

Our objective is to achieve image synthesis solely from natural language sentences. A natural language sentence is defined here as a complete sentence of a reasonable length to describe a scene with non-conceptual objects that can be visualized. The target of the outcome image is set as the image of a scene with the necessary and sufficient information derived from the input sentence.

Figure 1: Examples of synthesized images.

Most research on automatic image synthesis exploits powerful clues, such as sketches [2], in addition to from linguistic information. Only a few studies have addressed syntheses that can be obtained solely from verbal information. There are two state-of-the-art proposals, that have considered image synthesis. One study set out to add a word to an image [7] through the learning of spatial, scale-related, and appearance-based contexts; the other attempted to describe short sentences containing fewer than 10 words [16]. As far as we know, no work to date has been able to successfully generate images of scenes by only using sentences, without any other hints, due to the complexity of the representation of information in both sentences and images.

Three steps are needed to convert a sentence into an image: (I) analysis of the input sentence, (II) correlation between words in the sentence and image patch labels, and (III) allocation of image patches over a canvas. Since people can at least interpret the gist of the linguistic description of a given scene with appropriate objects in a sentence at suitable positions, even if the corresponding images have aberrant brightness or odd camera angles, we focus on the first two steps. Examples of synthesized images are shown in Figure 1. Our contributions are fourfold:

- Implementation of a system for scene image synthesis by only using natural language sentences.
- Development of a hierarchical syntactic parser for sentence analysis.

MM '16, October 15-19, 2016, Amsterdam, Netherlands

DOI: http://dx.doi.org/10.1145/2964284.2967193

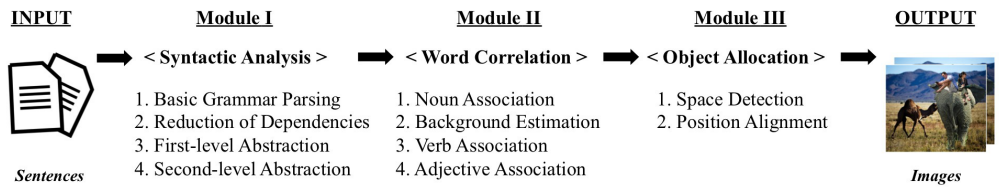

Figure 2: Overview of the proposed system.

- Design of an assignment algorithm for word correlation.
- Demonstration of the capability of our system to produce novel images and succinctly convey ideas.

2. SCENE IMAGE SYNTHESIS

An overview of our system is provided in Figure 2. A module is used to deal with each of the three steps stated in Section 1.

2.1 Syntactic Analysis (Module I)

The purpose of this module is to understand the syntactic structure of input sentences. In general, there is great variety in the patterns of grammars; thus, this module attempts to abstract fundamental grammatical structures from sentences. We assume that all essential information for constructing an interpretable image is summarized into three variables: objects, their states, and interactions among them. Furthermore, we assume that all objects are described by nouns, all states are expressed by either adjectives or verbs, and all interactions are either relative locations signified by prepositions, or interactions expressed through verbal actions.

An overview of this module is shown in Figure 3. Words in red represent objects, those in green express states of objects, and words in blue represent interactions among objects. In addition, words in pink describe body parts.

Fundamental Grammar Extraction. The input sentence is initially adjusted to eliminate ambiguity, and a parser [3] is applied to extract the fundamental grammar. At this stage, the input sentence is represented as a tree, where each node represents a word with order index within the sentence, and each edge is tagged by one of 50 possible dependencies.

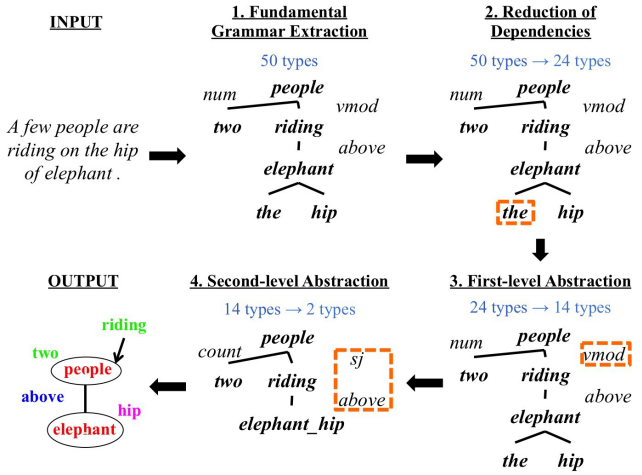

Figure 3: Overview of functions in Module I.

Reduction of Dependencies. Prior to abstraction, 26 dependencies are eliminated due to low occurrence rates (less than 1%) in a mixo-corpus [6, 14, 10], which explains the dispensable nature of rules and minor parts of speech.

First-level Abstraction. The remaining 24 types of dependencies G^{L0} are organized into the 14 varieties G^{L1} shown in Table 1. Here, G^{L0} represents a group of dependencies at hierarchical level 0. Types in G^{L1} can be bundled into three categories: subjective or objective G^{L1}_{sjoj}, and location G^{L1}_{loc} and modification G^{L1}_{mod}. This procedure is carried out by using the rules listed in Table 2, which depend on part of speech and scales of objects. In the tables, the † can take any form within the context of the grammatical pattern or word.

Second-level Abstraction. By leveraging the three categories, those in G^{L1} finally fall into two types of G^{L2}, that is, state M and interaction R. The possible operations are summarized in Table 3.

Table 1: 14 types in G^{L1}.

Group	Type
G^{L1}_{sjoj}	*subjective (sj), objective (oj)*
G^{L1}_{loc}	*inside (in), property (with), nearby (by), on (above) in front of (ifo), far away (away), below (below) back (behind), upward (over), downward (under)*
G^{L1}_{mod}	*adjective (adj), count (count)*

Table 2: Patterns of first-level abstraction.

Target	Operation	Example
Edge	shift to *sj*	$(agent, VP, N) \rightarrow (sj, VP, N)$
	shift to *oj*	$(acomp, V, J) \rightarrow (oj, V, J)$
	shift to *adj*	$(amod, N, J) \rightarrow (adj, N, J)$
	shift to parent's	$(conj\_and, J, J) \rightarrow (parent's, J, J)$
	shift to G^{L1}_{loc}	$(prep\_in, N_1, N_2) \rightarrow \begin{cases} (in, N_1, N_2) & f_{\text{size}}(N_1) \leq f_{\text{size}}(N_2) \\ (with, N_1, N_2) & f_{\text{size}}(N_1) > f_{\text{size}}(N_2) \end{cases}$
Node	integrate	$(nn, N_1, N_2), (\dagger, \dagger, N_2) \rightarrow (\dagger, \dagger, N_1\_N_2)$
	lift up	$(npadvmod, R, N) \rightarrow (\dagger, groundparent's, N)$

Table 3: Patterns of second-level abstraction.

Role	Operation	
	Target	Flow
State	$g \in G^{L1}_{mod}$	$(adj, N, \dagger) \Rightarrow M_N = M_N \cup \dagger$
	words used to detect relations	$(sj, \dagger, N_1), (oj, \dagger, N_2) \Rightarrow M_{N_1} = M_{N_1} \cup \dagger$
Relation	*sj* and *oj* sharing the same word	$(sj, \dagger, N_1), (oj, \dagger, N_2) \Rightarrow R_{N_1,N_2} = \dagger$
	sj and $g \in G^{L1}_{loc}$ sharing the same word	$(sj, \dagger, N_1), (g, \dagger, N_2) \Rightarrow R_{N_1,N_2} = g$
	$g \in G^{L1}_{loc}$ which connects two nouns	$(g, N_1, N_2) \Rightarrow R_{N_1,N_2} = g$

Algorithm 1 Noun Association

Require: $w_1, w_2, ..., w_N, t$
  for $j = 1, 2, ..., N$ **do**
    Get a list of synsets $s_1, s_2, ...$ for w_j.
    for $i = 1$ **to** $\min(\text{listsize}, \theta_{\text{syn}})$ **do**
      Trace back from s_i to the nearest keystone k_i.
      Get confidence $c^{\text{k}_\text{i}} = q_{\text{all}}(\text{s}_\text{i}, \text{k}_\text{i}, t)$ via (1)$\sim$(4).
    end for
    • **Domain**
    Set domain $d_j = \underset{i}{\text{argmax}}\, c^{k_i}$.
    • **Patch**
    Get a set of visualizable patches P_j that belongs to d_j.
    Set object patch $p_j \in P_j$.
    • **Scale**
    Trace back from p_j to the nearest keystone k_j with scale.
    Set object scale u_j to the scale of k_j.
  end for

2.2 Word Correlation (Module II)

The goal of this module is to associate nouns, verbs, and adjectives in the input sentence with those in the image dataset. The size of objects and the background images are estimated as well. As is often the case in synthesis, it is difficult to collect a wide variety of image patches. Therefore, we need to correlate a word in the sentences with a patch of the closest concept within a small number of choices. The mapping is based on three lexical pyramids for each sentence, where statistical measurements are used to make judgments. While nouns are explored within WordNet [5], verbs are used from VerbNet [12]. Adjectives are used only when they express emotions or sizes.

In this module, word similarity is measured by word2vec [11], and sentence similarity is obtained by a method proposed by Li et al. [9] with an alteration to use word2vec to calculate word similarity. $\text{sim}_{\text{syn-sent}}(\text{s}, \text{t})$, which is the similarity between synset s and sentence t, is given by computing the average of sentence similarities between synset's example and the given sentence, and synset's definition and the given sentence, respectively. Synset similarity $\text{sim}_{\text{syn-syn}}(\text{s}_1, \text{s}_2)$ is the average of sentence similarities for computable pairs.

Noun Association. We specify a set of major nodes (keystones) that must contain at least one visualizable image class with a certain scale of object on their leaves. Keystones are collected by grouping words in WordNet until accumulated frequency in a certain subset exceeds 1% for general objects, humans, and backgrounds. Scales are only attached to leaf nodes in the keystones.

Given a synset s, its nearest keystone k, and system input sentence t, we define a score to evaluate the accuracy of assigning a proper synset as:

$$q_{\text{all}}(\text{s}, \text{k}, t) = q_{\text{path}} + q_{\text{sim}} + q_{\text{freq}}, \quad (1)$$

$$q_{\text{path}}(\text{s}, \text{k}) = \frac{1}{2}\Big(\frac{1}{\theta_{\text{path}} + r_{\text{rel}}(\text{s}, \text{k})} + \frac{r_{\text{abs}}(\text{s})}{\max_{\text{s}\in\text{s}^{\text{all}}} r_{\text{abs}}(\text{s})}\Big), \quad (2)$$

$$q_{\text{sim}}(s, t) = \text{sim}_{\text{syn-sent}}(s, t), \quad (3)$$

$$q_{\text{freq}}(s) = \frac{1}{\theta_{\text{syn}} + f_{\text{synind}}(s)}, \quad (4)$$

where r is a function that measures either absolute depth or relative depth, s^{all} represents all synsets available in WordNet, and function f_{synind} returns the order index of synsets.

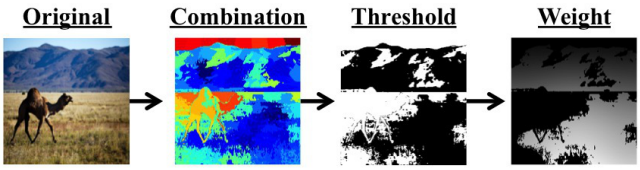

Figure 4: An example of spacious area computation.

θ_{path}, which adjusts the score weights of paths, is set to 3, and θ_{syn}, which is the maximum number of synsets to be considered as candidates, is set to 5. The entire process is described in Algorithm 1.

Background Estimation. A background is eventually estimated. We define the background through scene estimation, and domains are selected based on the nouns associated. That is, a set of these domains D is used to estimate a background among available scenes B as follows:

$$\underset{b\in B}{\text{argmax}} \sum_{d_j \in D} \Big(\text{sim}_{\text{syn-syn}}(b, d_j) + \text{sim}_{\text{syn-sent}}(b, t)\Big). \quad (5)$$

Verb and Adjective Association. Compared with noun association, these operations are more straightforward, and are grounded on the appearance of images. All verbs are classified into six self-contained activities and four interactive activities spanning both sites of action. Adjectives are simply assigned to abstract classes with the closest meaning.

2.3 Object Allocation (Module III)

The role of this module is to synthesize image patches over the estimated background. Two things this function needs to do is detect spacious areas in the background and combine the result with knowledge from the Module II.

Spacious Area Computation. An example of this procedure is shown in Figure 4. A saliency map [8] (S) is first computed to find rough candidates for the spacious region. A segmentation map (L) based on SLIC [1] is then obtained to specify object boundaries. The outcomes are mingled by:

$$\forall i, \boldsymbol{C}[x, y] = \frac{1}{N_i} \sum_{\acute{x}, \acute{y} \in L_i} \boldsymbol{S}[\acute{x}, \acute{y}], \quad (6)$$

where x, y is the position of the image, $S[x, y], C[x, y]$ represents the value at that position, i is the index of segmentation, and N_i is the number of pixels belonging to the label group of L_i. A threshold $\delta * \mu$ is used to render the combined map C more sophisticated, where μ denotes the mean of the combined map. From this threshold map T, small areas the ratio of active pixels of which to the area are smaller than η are removed. The area size is 0.08% of the image size, and we set δ, η to 0.9, 0.6, respectively. Finally, a Gaussian filter is applied to reduce areas near the outer edge, and a linear filter weighs either the foreground or the background to yield the final weighted map W.

Position Adjustment. The system prepares a canvas and places the center of the main object at a point that can maximize the summation of values within the object area. All other objects are sequentially connected to it along the lines of rules in Figure 5, which assumes three patterns of relations: vertical links, horizontal links, and immediate surroundings. Following linkage, patches are considered to be a single chunk. In case the chunk expands to the top row

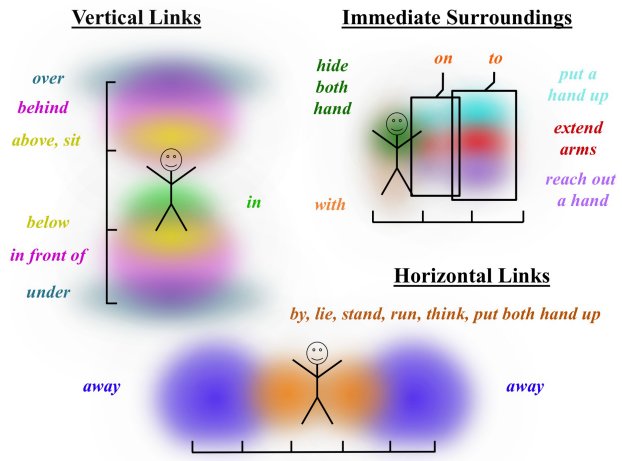

Figure 5: Positioning of all potential links.

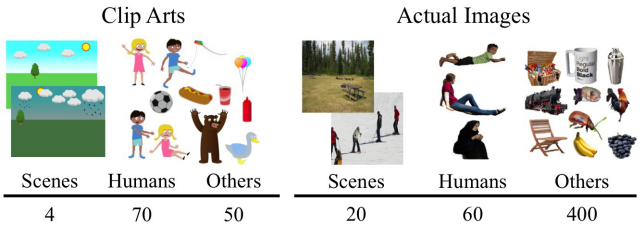

Figure 6: Examples of image patches.

Table 4: Experimental Tasks

Task	Method	# of subjects	# of sentences	Comparison				Time Limit
				ours	w/o M1	w/o M2	original	
1A	Scoring	100	20	✓	✓			-
1B	Scoring	100	15	✓		✓		-
1C	Scoring	100	20	✓		✓		-
2	Ranking	200	60	✓	✓	✓	✓	-
3	Q&A	100	20	✓			✓	4 sec.

of the canvas or the left, it is transferred or scaled as needed. The patches are finally synthesized over the background.

3. EXPERIMENT

We applied our system to the clip-art dataset [16] and an actual image dataset, which was a mixo-dataset of existing datasets. Scenes were sampled from Places205 [15] and general objects were excerpted from ImageNet [4]. Fifty images classified with the highest confidence scores in the VGG16 network [13] were chosen as candidates, and a few images were manually selected. Images of people in different poses were collected manually from the Internet. The examples are shown in Figure 6.

Input sentences are sampled from those datasets with images and corresponding sentences. Examples of the outcome are shown in Figure 1. The supplemental material provides more examples. Output images are evaluated through three tasks by crowd-sourced workers. Experimental setting is provided in Table 4. Results are shown in Figure 9 on the next page.

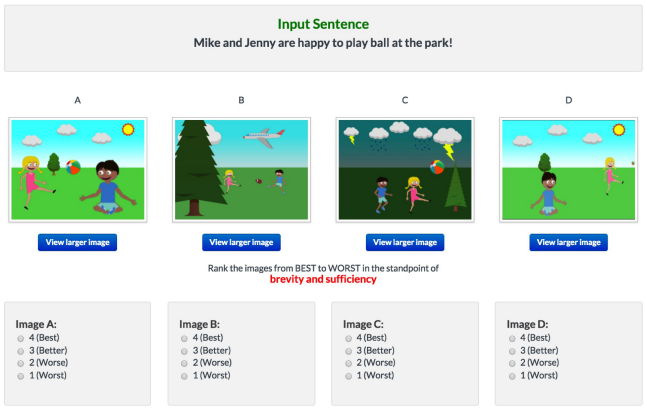

Figure 7: Experimental design of Task 2.

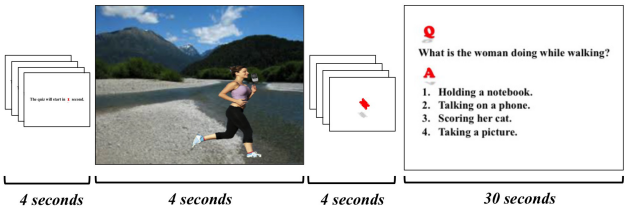

Figure 8: Experimental design of Task 3.

Task 1. The modules' effectiveness in terms of relation extraction (A), object correlation (B), and scale decision (C) were verified by absolute scoring. The synthesized images and their corresponding sentences, from either the entire system or partially deficient systems, were randomly ordered. Subjects were asked to score these using an integer from 4 (Very Good) to 1 (Very Poor) according to quality. The results indicated that all three aspects exerted a favorable influence on outcomes.

Task 2. Figure 7 depicts overall performance with relative ranking, obtained by comparing the performances with and without modules M1 and M2. Given a sentence and four images, including the original image, the contributors were asked to rank images from 4 (Best) to 1 (Worst). Three images were produced by the algorithm using both Modules I and II, using only Module II (without Module I), and using only Module I (without Module II). The other image is the original image taken from the dataset. These four images are randomly mixed in each session. The result showed that the original images were generally superior to the ones generated by our system. However, a few synthesized images were valued higher than the originals.

Task 3. The ability to convey the gist was measured by quickly asking questions of the subjects relating to parts of a scene. As shown in Figure 8, the subjects watched short videos that began with a blank screen for four seconds, was followed by an image that was shown for four seconds, another blank screen that lasted four seconds, and followed by a question related to the theme of the image. The respondents had four choices of answers to choose from, and these choices were formulated using the input sentences. There was a clear, significant difference in the average recall accuracy of the images, indicating that concise images without unnecessary depictions could more efficiently convey important ideas in sentences under the designated time limit.

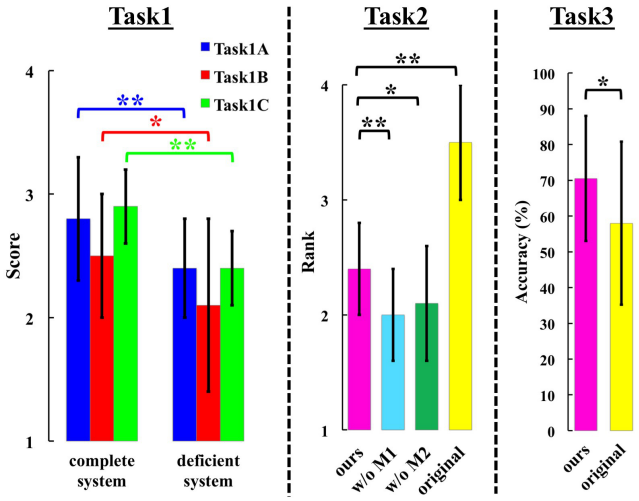

Figure 9: The result of experiments.

4. CONCLUSION

We implemented a system composed of three modules to achieve scene image synthesis using only natural language sentences. We focused on two modules.

Module I conducts syntactic analysis, for which a sentence parser using hierarchical syntactic analysis was developed. While it can handle common grammatical patterns, it is somewhat limited in dealing with collocations, time series, and words with various choices of parts of speech. Module II is responsible for word correlation, and we proposed a combination of lexical knowledge and corpus statistics to this end. This module solves an inherited problem of fewer image patches. Because its performance relies on a variety of elements, such as the chosen WordNet synsets, prepared word domains, and collected image patches, each image patch selection can fail when incorrect.

Through experiments, we showed that the two modules play an important role in image synthesis from natural language sentences. Another major insight of our experiments was the effectiveness of concise images as a means of delivering gists of scenes in a short time. Our future work will consist of the analysis of more complex grammars, collection of a greater number of image patches, and the comprehension of multiple sentences.

5. ACKNOWLEDGEMENT

This work was funded by the ImPACT Program of the Council for Science, Technology, and Innovation (Cabinet Office, Government of Japan).

6. REFERENCES

[1] R. Achanta, S. Appu, S. Kevin, L. Aurelien, F. Pascal, and S. Sabine. Slic superpixels compared to state-of-the-art superpixel methods. *PAMI*, 34(11):2274–2282, 2012.

[2] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu. Sketch2photo: internet image montage. *ACM TOG*, 28(5):124:1–10, 2009.

[3] M.-C. De Marneffe, B. MacCartney, C. D. Manning, et al. Generating typed dependency parses from phrase structure parses. In *ICLRE*, 2006.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[5] C. Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[6] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, 47:853–899, 2013.

[7] S. Inaba, A. Kanezaki, and T. Harada. Automatic image synthesis from keywords using scene context. In *ACMMM*, pages 1149–1152, 2014.

[8] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *PAMI*, 20(11):1254–1259, 1998.

[9] Y. Li, D. McLean, Z. Bandar, J. D. O'shea, K. Crockett, et al. Sentence similarity based on semantic nets and corpus statistics. *TKDE*, 18(8):1138–1150, 2006.

[10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[12] K. K. Schuler. *Verbnet: A Broad-coverage, Comprehensive Verb Lexicon*. PhD thesis, Philadelphia, PA, USA, 2005. AAI3179808.

[13] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *ACL*, 2:67–78, 2014.

[15] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, pages 487–495, 2014.

[16] C. L. Zitnick, D. Parikh, and L. Vanderwende. Learning the visual interpretation of sentences. In *ICCV*, 2013.