Autonomous Object Translation from Language

Rishub Jain
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
rishubj@andrew.cmu.edu

Abstract

Currently most object translation tasks within an image require tedious human input. We propose a system that attempts to perform this object translation task from a sentence describing the object and how it is manipulated. We explore and compare different models for this task.

1 Introduction

1.1 Problem Statement

Object translation is the task of moving an object in an image. We attempt to create a system that autonomously does this object translation within an image given a natural language sentence of the objects description and where to move it.

1.2 Related Work

There has been much work in object translation, and object manipulation in general. However, all of this work has had some human involvement in the manipulation process. Goldberg et al. proposed a system that can manipulate an image by adding features of another image. For example, with this system one can add a man to an image with a horse to create an image of the man riding that horse. However, this process requires the user to outline where the person and horse is.

Iizuka et al. proposed a system that translates objects within the image and can draw a copy of an object in the image. Margolin et al. created a system that can change the color of an object while making it look natural. Chen et al. propose a system that can modify 3D objects in an image. However, all of these methods also require a user to outline which objects they should manipulate and how they should manipulate them.

1.3 Motivation

Previous methods lack in the ability to easily manipulate images, and require a human to go through a tedious process of manually outlining objects and manipulations that they want to do. Our system takes a step towards replacing that process with a single sentence that the human needs to write describing the object and how it should be manipulated. If successful, this can greatly decrease the time it takes to manipulate objects in an image, and can even make object manipulation more accessible to non experts.

2 Dataset

For this problem, we created our own dataset. This dataset includes an original image, the sentence describing an object and how it should be manipulated, and the final image resulting from that manipulation. An example image pair and sentence is shown in Figure 1.

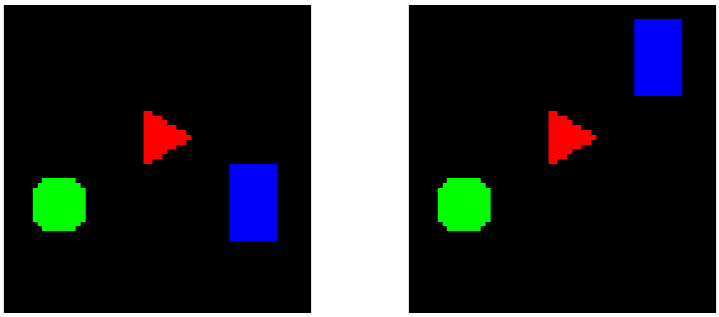

Figure 1: An example image pair in the dataset with the manipulation sentence: "Move the **blue rectangle** 30 pixels **up**". Only the bolded words change value across examples. The left image is the original image and the right image is the manipulated image.

Each original image was created by randomly selecting the positions, sizes, and colors of a circle, rectangle, and triangle, and placing them on a 64×64 image. Then, an object and direction were randomly chosen, and the manipulation sentence and final image were created from that. These random selections were chosen such that a valid, reasonable image could be created.

We used this method to generate 100,000 image pairs and sentences. For our evaluation, we used 90,000 examples as our training set, 5,000 as our validation set, and 5,000 as our test set.

3 Methods

3.1 Sentence representation

In the case of our dataset, only 3 values were manipulated: the color and shape of the object being translated, and direction of translation. Thus, we represented our sentence as a 3-hot vector of length 10 that fully encodes this sentence.

3.2 Encoder-Decoder Model

We initially created an Encoder-Decoder model for this problem. This model first takes the original image as an input and feeds it through an encoding module (a 5 layer CNN with max pooling after each layer) to produce a flattened representation. The sentence representation is then concatenated with the sentence representation. This is then fed through a 1-layer MLP, which is fed to the decoding module (a 5 layer CNN, each layer is a transpose of a convolution layer). The output of this decoding module is the final predicted translated image.

3.3 Encoder-Decoder Model with Sentence Reconstruction

Another model we had tried was the initial Encoder-Decoder model with an added sentence recreation loss. This model uses the same architecture as the previous Encoder-Decoder Model, but adds a convolution module after the final decoding module. This final convolution module takes both the original image and the reconstructed image, and feeds them through another 5 layer CNN followed by a 1-layer MLP. The final output is then the length 10 sentence encoding. For further supervision, this final output has a softmax over each of the 3 information categories: color, shape, and direction. The final loss that the network is trained on is: 80% * Manipulated Image Construction Cross Entropy Loss + 20% * Manipulated Sentence Reconstruction Cross Entropy Loss.

4 Experiments and Results

We compared the above two models, along with a baseline Encoder-Decoder model that did not use the sentence encoding at all. These results are shown in Table 1.

Table 1: Cross Entropy Loss of each model on the test set

Model	Cross Entropy Loss
Encoder-Decoder Model	0.1340
Encoder-Decoder Model without Sentence	0.0645
Encoder-Decoder with Sentence Reconstruction	0.1020

4.1 Qualitative Results

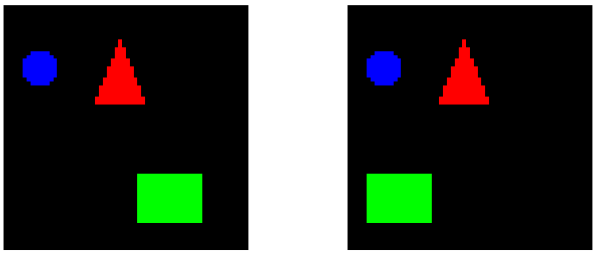

Figure 2: An example image pair in the dataset with the manipulation sentence: "Move the **green rectangle** 30 pixels **left**". The left image is the original image and the right image is the manipulated image.

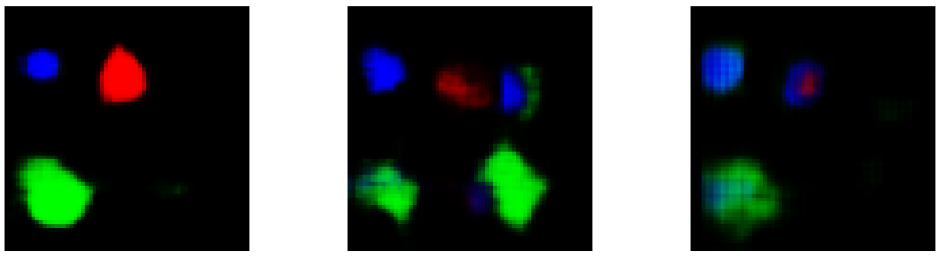

Figure 3: The results of each method on the example image pair and sentence shown in Figure 2, from left to right: Encoder-Decoder Model, Encoder-Decoder Model without Sentence, Encoder-Decoder with Sentence Reconstruction

5 Conclusion

Qualitatively, all models are able to reasonable capture the essence of the images because of the strength of the Encoder-Decoder modules. However, it is clear that the model without access to the sentence is unable to predict which object is translated. Surprisingly, it seems to be able to predict where each object is likely to be translated because it draws blue and green shapes in the only places they could be translated in this case, and recognizes that the red object could not be translated because it would then collide with the green object. Also, the Encoder-Decoder model with the sentence reconstruction seems to be performing worse than the Encoder-Decoder model.

The cross entropy loss results also show that the Encoder-Decoder model outperforms the other two models, and that having no access to the sentences leads to poor performance on the test set. The

Encoder-Decoder model with the sentence reconstruction might be performing worse because the sentence reconstruction module of the network might not be doing what it is intending to do. The information of the sentence is likely encoded in the image such that the sentence reconstruction module looks for that encoding instead of the differences between the image pairs to predict the sentence vector.

6 Future Work

One issue with the current model is that it is unable to refine the objects such that they look like the input shapes. One way to fix this could be to apply a discriminator loss on the predicted manipulated image. This discriminator could predict whether or not the input image was constructed from our model or if it came from the original dataset, and this loss could be applied to our model so it would try to create images that look similar to the dataset. We had attempted to use this but could not get reasonable results because of the difficulties of training GAN-like architectures.

If this method proves to perform well, it could be used on more complicated datasets. For example, for translating shapes, one could make the dataset more challenging by allowing for two objects in the same image to have the same color or shape, allow them to be overlapping, and allow for translations of varying amounts instead of the fixed 30 pixels used.

In addition, this could be applied to more complicated domains, such as real life images with many different real objects.

References

[1] Goldberg, C., Chen, T., Zhang, F. L., Shamir, A., & Hu, S. M. (2012, May). Data-Driven Object Manipulation in Images. In *Computer Graphics Forum* (Vol. 31, No. 2pt1, pp. 265-274). Blackwell Publishing Ltd.

[2] Iizuka, S., Endo, Y., Hirose, M., Kanamori, Y., Mitani, J., & Fukui, Y. (2014, December). Object repositioning based on the perspective in a single image. In *Computer Graphics Forum* (Vol. 33, No. 8, pp. 157-166).

[3] Margolin, R., Zelnik-Manor, L., & Tal, A. (2013). Saliency for image manipulation. *The Visual Computer*, 29(5), 381-392.

[4] Chen, T., Zhu, Z., Shamir, A., Hu, S. M., & Cohen-Or, D. (2013). 3-sweep: Extracting editable objects from a single photo. *ACM Transactions on Graphics (TOG)*, 32(6), 195.