Diffusion Illusions: Hiding Images in Plain Sight

Ryan Burgert, Kanchana Ranasinghe, Xiang Li, Michael Ryoo

1 Introduction

In this paper, we present a method for creating optical illusions and steganography using text-to-image diffusion models. It's very similar to the technique used in Peekaboo [2]. The technique leverages a variant of Score Distillation Loss (SDL) introduced in DreamFusion [1] to optimize different optical illusions, hiding images in plain sight. We focus on three types of illusions: Flippy Illusions, Rotating Overlays, and Hidden Characters. In the following sections, we provide a mathematical explanation for the methods used to create each illusion. For more information and examples, see ryanndagreat.github.io/Diffusion-Illusions. This short paper is supplementary to that website.

2 Flippy Illusions

The goal is to create an image I that matches the prompt P_1 when viewed right-side up and matches the prompt P_2 when viewed upside down. We use a variant of Score Distillation Loss (SDL) for optimization, inspired by DreamFusion [1].

Mathematically, we want to minimize the loss function:

$$L(I) = \mathrm{SDL}(I, P_1) + \mathrm{SDL}(\mathrm{rot180}(I), P_2) \tag{1}$$

where $\mathrm{rot180}(I)$ is the image I rotated 180 degrees.

The optimization is performed using gradient descent:

$$I^{(t+1)} = I^{(t)} - \alpha \nabla L(I^{(t)}) \tag{2}$$

where α is the learning rate and t is the iteration number.

3 Rotating Overlays

The goal is to create two images, T (top) and B (bottom), such that when T is overlaid on B and rotated by 0, 90, 180, and 270 degrees, they match prompts P_1, P_2, P_3, and P_4, respectively.

We model the light filtering when overlaying two transparencies with a pixel-wise multiplication operation: $O = T \cdot B$, where $\cdot$ is element-wise multiplication.

The loss function we want to minimize is:

$$L(T, B) = \mathrm{SDL}(T \cdot B, P_1) + \mathrm{SDL}(\mathrm{rot90}(T) \cdot B, P_2) + \mathrm{SDL}(\mathrm{rot180}(T) \cdot B, P_3) + \mathrm{SDL}(\mathrm{rot270}(T) \cdot B, P_4) \tag{3}$$

Optimization is performed using gradient descent on both T and B:

$$T^{(t+1)} = T^{(t)} - \alpha \nabla L(T^{(t)}, B^{(t)}) \tag{4}$$

$$B^{(t+1)} = B^{(t)} - \alpha \nabla L(T^{(t)}, B^{(t)}) \tag{5}$$

4 Hidden Characters

The goal is to create four images I_1, I_2, I_3, and I_4, that match prompts P_1, P_2, P_3, and P_4 when viewed individually. When overlaid on top of each other, they reveal a hidden image H that matches prompt P_H. We model the light filtering when overlaying multiple transparencies with a pixel-wise multiplication operation, in the same way as Rotating Overlays.

The loss function we want to minimize is:

$$\begin{aligned} L(I_1, I_2, I_3, I_4) =& \mathrm{SDL}(I_1, P_1) + \mathrm{SDL}(I_2, P_2) + \mathrm{SDL}(I_3, P_3) \\ &+ \mathrm{SDL}(I_4, P_4) + \mathrm{SDL}(I_1 \cdot I_2 \cdot I_3 \cdot I_4, P_H) \end{aligned} \tag{6}$$

Optimization is performed using gradient descent on all four images:

$$I_1^{(t+1)} = I_1^{(t)} - \alpha \nabla L(I_1^{(t)}, I_2^{(t)}, I_3^{(t)}, I_4^{(t)}) \tag{7}$$

$$I_2^{(t+1)} = I_2^{(t)} - \alpha \nabla L(I_1^{(t)}, I_2^{(t)}, I_3^{(t)}, I_4^{(t)}) \tag{8}$$

$$I_3^{(t+1)} = I_3^{(t)} - \alpha \nabla L(I_1^{(t)}, I_2^{(t)}, I_3^{(t)}, I_4^{(t)}) \tag{9}$$

$$I_4^{(t+1)} = I_4^{(t)} - \alpha \nabla L(I_1^{(t)}, I_2^{(t)}, I_3^{(t)}, I_4^{(t)}) \tag{10}$$

5 Conclusion

We have presented a method for creating optical illusions and steganography using text-to-image diffusion models. Our technique leverages a variant of Score Distillation Loss (SDL) introduced in DreamFusion to optimize different optical illusions, hiding images in plain sight. We have focused on three types of illusions: Flippy Illusions, Rotating Overlays, and Hidden Characters, providing mathematical explanations and intuitive insights behind the methodology.

Though the multiplication model for light passing through transparencies may not perfectly capture the real-world behavior of ink and transparency materials, it remains a useful approximation for our purposes. By understanding the limitations and strengths of this model, we can create fascinating and effective optical illusions, as demonstrated by the accompanying demo video.

A Modeling Light Attenuation as Multiplication

When light passes through a transparency, its intensity is affected by the transparency's opacity. If a transparency is fully transparent (let's say at a pixel), all of the light passes through. If it's fully opaque (at a pixel), no light passes through. Partially transparent areas allow some light to pass through, attenuating the light intensity proportionally to the transparency's opacity.

When we overlay multiple transparencies, the light passes through each of them in succession. With each layer, the light intensity is further attenuated according to the opacity of the current layer. Mathematically, this can be modeled as a multiplication operation: the light intensity after passing through multiple layers is the product of the light intensities after passing through each individual layer.

In practice, the multiplication model may not perfectly capture the behavior of light passing through transparencies with ink. For instance, black ink on a white transparency will not have the same effect as black ink on a black transparency. However, as you can see on the website's videos, it works well enough.

References

[1] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3D using 2D Diffusion. *arXiv preprint arXiv:2012.01894*, 2022.

[2] Ryan Burgert, Kanchana Ranasinghe, Xiang Li, and Michael S. Ryoo. Peekaboo: Text to Image Diffusion Models are Zero-Shot Segmentors. *arXiv preprint arXiv:2211.13224*, 2022.