Enhancing Image Inpainting with GLIDE: Techniques, Applications, and Performance Evaluation

***Abstract*— Image inpainting is a crucial field in computer vision and image processing that has garnered significant attention in recent years. Advances in deep learning have propelled inpainting techniques to new levels of sophistication, with models like GLIDE (Guided Language-to-Image Diffusion for Generation and Editing) emerging as powerful tools for reconstructing missing or damaged regions in images. This paper provides an introduction to GLIDE-based inpainting methods and evaluates their performance in reconstructing images.**

I. Introduction to Image Inpainting

Have you ever wondered how damaged photographs are restored to look like new, or how satellite images fill in missing data from incomplete scans? The answer is through image inpainting, a technique for filling in missing or corrupted parts of an image by using information from surrounding areas so they blend seamlessly with the rest of the image [1]. With the development of advanced image processing techniques, image inpainting has become widely applicable across fields such as photography, medical imaging, and satellite technology. [2].

Fig. 1. Historical image inpainting examples, first row presents the original images, second row presents the inpainted images. [3].

II. Introduction to GLIDE

GLIDE (Guided Language-to-Image Diffusion for Generation and Editing) is a powerful AI model developed by OpenAI for photorealistic image generation and editing tasks [4]. The model consists of three main components: a diffusion model (ADM-G) for generating 64x64 pixel images, a transformer-based text encoder, and an upsampling model for 256x256 pixel outputs [5].

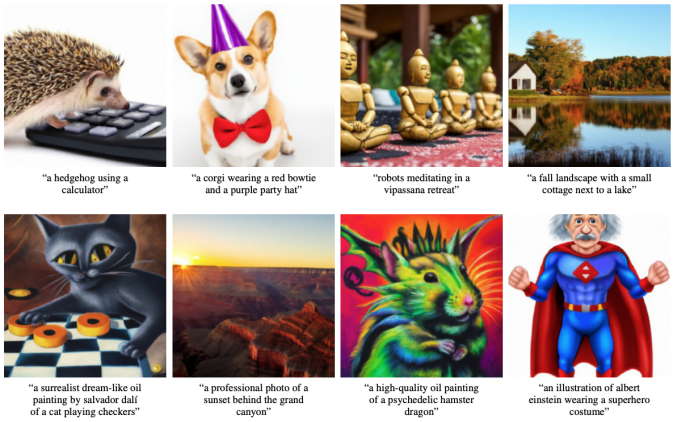

Fig. 2. Examples of images generated by GLIDE based on text prompts. Source: [4].

III. Applications of GLIDE in Image Inpainting

A. Effectiveness in Inpainting Tasks

One of GLIDE's unique capabilities is its effectiveness in image inpainting. By utilizing natural language prompts, GLIDE can generate or restore images with contextually accurate details, pushing the boundaries of image editing and restoration.

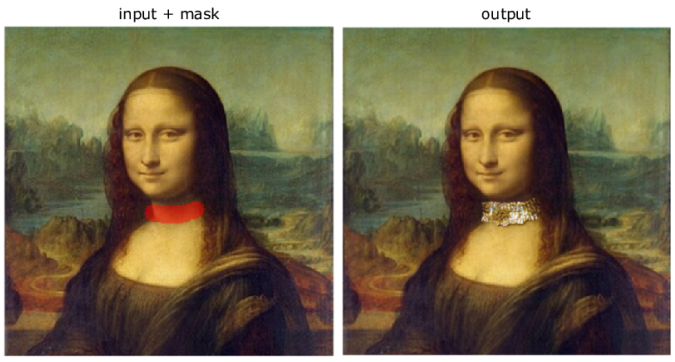

Fig. 3. An example of inpainting with GLIDE. The model receives an input as the Mona Lisa as well as a mask and a text description "a golden necklace". Then, it generates the output as Mona Lisa with a golden necklace. [6].

B. Performance Evaluation and Quantitative Analysis

To evaluate GLIDE's effectiveness in image inpainting, we use two quantitative metrics: PSNR and SSIM.

- **PSNR**: Measures pixel-level difference between the original and inpainted images, with higher values indicating better quality. The equation for PSNR is:

$$\text{PSNR} = 10 \log_{10} \left(\frac{R^2}{MSE} \right)$$

where R is the maximum possible pixel value of the image (e.g., 255 for an 8-bit image), and MSE is the Mean Squared Error between the original and inpainted images.

- **SSIM**: Assesses perceptual similarity, focusing on structural integrity, with values from -1 to 1 (1 being perfect similarity). The equation for SSIM is:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

where:

- μ_x and μ_y are the mean pixel values of the original and inpainted images, respectively.
- σ_x^2 and σ_y^2 are the variances of the original and inpainted images, respectively.
- σ_{xy} is the covariance between the original and inpainted images.
- C_1 and C_2 are small constants to stabilize the division (typically $C_1 = 0.01^2$ and $C_2 = 0.03^2$).

By using both PSNR and SSIM, we obtain a more comprehensive evaluation of GLIDE's performance in inpainting tasks, as these metrics collectively capture both pixel accuracy and structural fidelity.

IV. Concept to Code

This section provides a complete guide from the initial dataset setup to running the inpainting model using GLIDE. Each step explains the concept and provides practical instructions to implement the inpainting process in a Colab environment.

Access the code at `https://github.com/MHC-FA24-CS341CV/beyond-the-pixels-emerging-computer-vision-research-topics-fa247/blob/main/code/08-im-inpainting/GLIDE_Inpainting.ipynb`.

1) **Dataset:** This GLIDE inpainting project uses datasets containing images with specific regions masked out, simulating real-world inpainting scenarios, such as restoring parts of faces or objects in natural scenes.
2) **Source:** Images can be uploaded directly into the Colab environment. Common sources for similar tasks include CelebA (for face images) and other standard computer vision datasets for diverse objects and scenes.
3) **Key Characteristics:**
   - The dataset contains images of varied subjects, making it suitable for evaluating GLIDE's inpainting capabilities across different contexts.
   - Specific regions of each image are masked, allowing GLIDE to restore or complete these parts based on surrounding context.
4) **Clone Repository:** Clone the official GLIDE repository to access the model and inpainting functions. Use the following commands to clone and navigate to the repository:

```
!git clone https://github.com/openai/glide-text2im.git
%cd glide-text2im
```

5) **Set Up Environment:** Install the necessary libraries and set environment paths as shown in the Colab notebook. This ensures that all dependencies required for GLIDE are available.
6) **Upload and Process Image:**
   - Upload an image to Colab that you wish to use for inpainting.
7) **Run Inpainting:**
   - Adjust parameters like `batch_size` (default 1), `guidance_scale` (default 3), and `upsample_temp` (default 0.997) to control inpainting quality and style.
   - Run the model to generate inpainted outputs, filling in masked regions based on surrounding content and set parameters.
8) **Calculate Metrics (Optional):** After generating the inpainted image, calculate PSNR and SSIM by comparing it to the original image.

V. Testing and Analysis

A. Goal

The primary goal of this evaluation is to assess the performance of the inpainting model across prompts of varying complexity. The quality of the inpainted results is measured using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM).

B. Methodology

To test the model's adaptability, three prompts of increasing complexity were used:

- **Easy**: Add two chairs to the scene.
- **Medium**: Add a dog lying down under the chairs to the scene.
- **Hard**: Place a fountain in the scene, introduce ambient lighting from the top-left, and change the wall color to light blue with a floral pattern.

Each inpainted image was evaluated against the original image using PSNR and SSIM metrics to measure pixel similarity and structural accuracy.

C. Results

In Figure 4, the left side shows the original image with a dog sitting between chairs in a café-like setting, providing the context for inpainting. The right side displays the masked image, where a solid gray area covers the dog and parts of the surrounding chairs. This masked region is where inpainting occurs, with the model tasked to fill in this area based on surrounding details, seamlessly blending it with the unmasked parts of the image.

Fig. 4. Original image and masked region for testing. The gray area represents the masked section, which the model attempts to reconstruct.

Table I summarizes the PSNR and SSIM values for each prompt level. Figures 5, 6, and 7 illustrate the original and inpainted images for each prompt.

TABLE I
PSNR AND SSIM VALUES FOR EACH PROMPT COMPLEXITY

Prompt Complexity	PSNR (dB)	SSIM
Easy	29.73	0.39
Medium	29.61	0.36
Hard	29.60	0.36

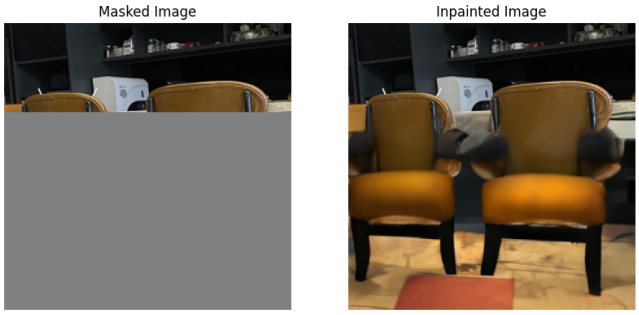

Fig. 5. Inpainting result for the Easy prompt: "Add two chairs to the scene."

For the **Easy** prompt (Figure 5), the model achieved a PSNR of 29.73 dB and an SSIM of 0.39, indicating a high level of pixel similarity. The inpainting result shows a smooth background replacement, consistent with surrounding textures, as object removal is generally simpler for the model.

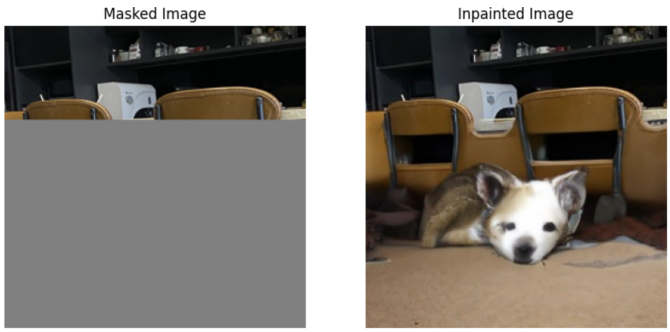

Fig. 6. Inpainting result for the Medium prompt: "Add a dog lying down to the scene."

For the **Medium** prompt (Figure 6), the PSNR was 29.61 dB with an SSIM of 0.36, reflecting a moderate structural change. The model partially succeeded by repositioning the dog's body, but the face and overall shape lack realistic detail, with some distortions present.

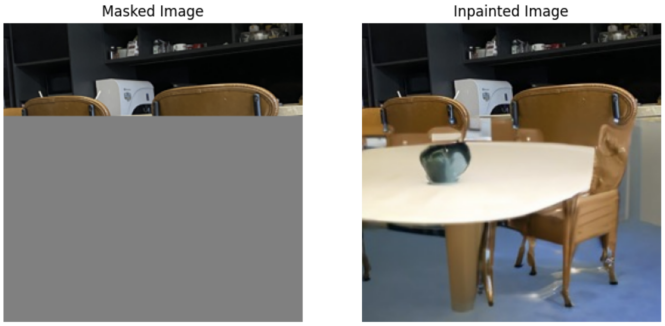

Fig. 7. Inpainting result for the Hard prompt: "Place a fountain in the scene, introduce ambient lighting from the top-left, and change the wall color to light blue with a floral pattern."

For the **Hard** prompt (Figure 7), the PSNR remained at 29.66 dB, similar to the Easy prompt, indicating minimal pixel-level change. However, the SSIM of 0.37 shows the model's difficulty in preserving structural similarity. The model produced a distorted shape instead of a realistic fountain and lacked an accurate background change, failing to apply the expected wall color and floral pattern, which reveals its limitations in handling complex, multi-layered tasks.

VI. Open Challenges and Current Limitations

In the field of image inpainting, there are several open challenges and limitations that researchers continue to address:

- **Resolution limitations:** GLIDE's output resolution is limited, with images being downsized to 64x64 during processing and only able to be upsampled to 256x256 without introducing artifacts. Higher resolution outputs are challenging to achieve without upsampling techniques [4].
- **Handling complex prompts:** While GLIDE can render a wide variety of text prompts zero-shot, it can have difficulty producing realistic images for more complex prompts [4].
- **Contextual consistency:** Maintaining contextual coherence between the inpainted region and surrounding image content remains challenging, particularly for complex images with intricate details [7].

VII. Research Idea

This research proposes enhancements to GLIDE's image inpainting capabilities, focusing on improving performance in complex scenes. Potential improvements include:

- **Dynamic Resolution Upsampling:** GLIDE's current upsampling can cause artifacts beyond 256x256 resolution. Integrating a progressive upsampling approach,

leveraging super-resolution techniques, could enhance photorealism, making GLIDE more suitable for applications like large-format photo restoration and high-res satellite imaging.

- **Evaluation Framework with Multi-Dimensional Metrics:** To better assess inpainting success, we can go beyond PSNR and SSIM by adding perceptual metrics like FID (Fréchet Inception Distance) and LPIPS (Learned Perceptual Image Patch Similarity). This multi-metric approach offers a fuller performance view, especially for applications where perceptual quality matters as much as pixel accuracy, such as in artistic restoration and autonomous driving.

These enhancements aim to address GLIDE's current limitations in handling detailed textures, maintaining contextual coherence, and ensuring consistent lighting across the inpainted regions.

REFERENCES

[1] "Image Inpainting — link.springer.com," https://link.springer.com/referenceworkentry/10.1007/0-387-30038-4_98, [Accessed 06-11-2024].

[2] R. Mitson, "Introduction to image inpainting with deep learning — wandb.ai," https://wandb.ai/wandb_fc/articles/reports/Introduction-to-image-inpainting-with-deep-learning--Vmlldzo1NDI3MjA5, [Accessed 06-11-2024].

[3] S. Zarif, I. Faye, and D. Awang Rambli, "Image completion: Survey and comparative study," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 29, p. 1554001, 12 2014.

[4] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," 2022. [Online]. Available: https://arxiv.org/abs/2112.10741

[5] "Generating and editing photorealistic images from text-prompts using OpenAI's GLIDE — blog.paperspace.com," https://blog.paperspace.com/glide-image-generation/, [Accessed 10-11-2024].

[6] A.-S. Maerten and D. Soydaner, "From paintbrush to pixel: A review of deep neural networks in ai-generated art," 02 2023.

[7] "ecva.net," https://www.ecva.net/papers/eccv_2022/papers_ECCV/papers/136770564.pdf, [Accessed 08-11-2024].