Enhancing Text-Specific Inpainting Using Diffusion Models and OCR

Rutuja Chavan
Department of Computer Engineering
Pimpri Chinchwad College of Engineering
Pune, India
rutuja.chavan231@pccoepune.org

Sahil Hirve
Department of Computer Engineering
Pimpri Chinchwad College of Engineering
Pune, India
sahil.hirve23@pccoepune.org

Prof. Swati Shinde
Department of Computer Engineering
Pimpri Chinchwad College of Engineering
Pune, India
swa.shinde@pccoepune.org

Prof. Bal Virdee
Center for Communication Technology
London Metropolitan University
London,
United Kingdom
b.virdee@londonmet.ac.uk

Ashish Khanna
Department of Computer Science
Maharaja Agarsen Institute of Technology
Delhi, India
ashish.khanna@mait.ac.in

Abstract— **Text-specific inpainting—masking and replacing particular words and phrases in the different images. It has significant potential for applications such as document redaction, privacy protection, and automated image editing. While previous research has struggled with this task. This paper presents an approach combining Optical Character Recognition (OCR) and diffusion models to address these challenges. We use pytesseract for text detection and Stable Diffusion for inpainting, after using that we aim to accurately replace specific words in images. Our experiments show promising results in simpler cases but reveal limitations when handling intricate backgrounds and fonts. Based on these findings, we suggest improvements to enhance the robustness of the method, specifically in handling complex image environments.**

Keywords— Text-specific inpainting, OCR (Optical Character Recognition), Pytesseract, Stable Diffusion, Document redaction, Privacy protection, Automated image editing.

I. INTRODUCTION

Text-specific inpainting is when parts of the text in an image are selectively replaced or masked, mainly for reasons of privacy preservation, the eradication of sensitive information, or modification of content. With a current high interest in the development of machine learning models and image processing, there appears to be lots of movement toward the development of even further generalized inpainting tools that could fill large areas within an image based on their surroundings' contents. But text replacement within an image with a detection of its structure turns out to be slightly tricky because it demands two things: one is the detection along with preserving the structure of the text within the image itself.

Most previous work on text[1] replacement or inpainting of text[4] regions relied on either CNNs or GAN but are[7] severely[3] limited by this weakness in handling the pasted text on complicated or cluttered backgrounds.

Such diffusion models open up new possibilities for inpainting and display quite fantastic results in the content generation when large parts of an image are masked [1]. Text-specific inpainting along with OCR detection to ensure precise text localization remains a rather uncharted territory.

This work attempts to bridge this gap by exploiting both the power of OCR and diffusion-based inpainting models for text replacement based on the work of earlier research, which attempted similar tasks but were not able to pull it off due to a variety of reasons [2]. Most previous work on text replacement or inpainting of text regions relied on either CNNs or GAN but are severely limited by this weakness in handling the pasted text on complicated or cluttered backgrounds [3].

II. RELATED WORK

A. OCR (Optical Character Recognition)

There has been a fair share of focus on OCR technology which allows machines to interpret images with text present in them. It is a common occurrence to see OCR tools such as Pytesseract used for extracting texts from images. The accuracy of OCR, however, tends to be quite weak especially in cases where a particular text is intricately woven into a design, has varying fonts or is low in contrast. So, it can be said that although OCR works efficiently with respect to high contrast text, it is comparatively weaker in text with more complex visual structures[10]. This weakness presents a case for methods that can aid in improving the performance of OCR in difficult circumstances [2].

B. Diffusion Models for Image Inpainting

Inpainting methods seek to cover gaps in images where a portion has been masked out such that the final result is seemingly seamless[15]. Before, inpainting was concentrated around some patches & CNNs which have made many advances whereas of late, diffusion models such as the stable diffusion have made huge progress in producing realistic content in versatile visual contexts [1, 4]. They perform that by progressively removing noise from a randomly generated image while following a specific textual description to create an image conforming to the desired context. Only the most recent development has focused on text inpainting but for the most part, diffusion techniques developed for scene painting have been inscriptive[1, 3, 5,14].

III. METHODOLOGY

The approach developed in the course of this work is composed of two components. The first part addresses the issues pertaining to OCR-based text localization while the second one focuses on inpainting with a diffusion model. Each of these will be elaborated in the following section.

A. Text Region Detection and Masking

This time we are going to focus on images and OCR software Tesseract. To do so, we take the image and first create its black and white, gray scale, alternative. This changes the contrast, and so minimizes the noise, easing the OCR in use further, making it a lot more effective. After, Tesseract does not serve only for isolation and recognition of text but also for retrieval of the location of every word in the respective area. Hence the OCR tool returns a dictated list of 'words' which each contain the box coordinates in the form of (x,y,w,h), where; x is the ordinate of x axis of the corresponding bounding box and y is the ordinate of the y axis of the corresponding bounding box. w is the width of the respective bounding box t and h is the height of the respective bounding box. Moreover, we create a binary mask of the particular region in the image as well. This mask is the same size as the input image and contains only white colored pixels (255) which correspond to the bounding box of the word, which indicates the region that we want to conceal to perform inpainting. The following is the mathematical formula for the creation of the binary mask

$$\underset{255}{Mask[i,j]} = \begin{cases} \text{if}(x \leq i < x + w) \text{ and} \\ \quad (y \leq j < y + h) \end{cases}$$

where (i,j) denotes a pixel of a character in the image.

Otherwise

where (i,j) denotes a pixel of a character in the image.

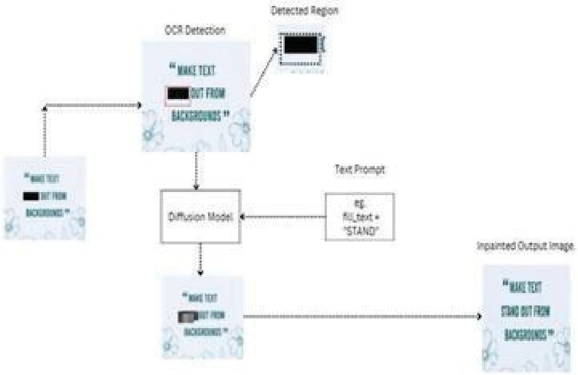

Fig. 1. The figure illustrates an OCR-guided text inpainting process using a diffusion model.

It detects and masks text regions, then fills the missing text based on a user-specified prompt to produce an inpainted output image with enhanced text completion. task of interpolating the missing portions cropping inside of a defined area/id mask in an image of consideration. The model in this case may be – The inpainting is needed for an image I with a mask M and a prompt P in this case, we may express it as $I = StableDiffusion(I,\ M,\ P)$ in which I is the outcome produced after the inpainting stage. The inpainted area could, using a diffusion model that employs a mask, commence to paint the inpainted area in a hierarchy – initially to fix the area and then paint it in so that it is hidden in the subsequent image.

B. Peak Signal to Noise Ratio as a measure of image quality (PSNR)

For this purpose, inpainting images, here used the Peak Signal-to-Noise Ratio (PSNR), which has been one of the most commonly used measures for reconstruction tasks.

How is it calculated? The measure is calculated between the original picture and the inpainting picture to determine how close the inpainting procedure was to the original. It is computed as PSNR is defined as: design.complex backgrounds, different font types, and varied text sizes[7]. Each image was preprocessed to extract the target word and apply inpainting with the prompt.

C. The Development of Inpainting Inference Based on Stable

$$\text{PSNR} = 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{\text{MSE}}}\right)$$

MAX_I for an 8-bit image can reach 255 and MSE (Mean Squared Error) is the statistic obtained when the difference of the original and inpainting images is squared and averaged:

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(I_{\text{original}}(i,j) - I_{\text{inpainted}}(i,j) \right)^2$$

Why is PSNR Important? PSNR indicates the likeness to what extent the original and inpainting images might have the same structure at the most bottom level which is a pixel. This indicates that images of higher PSNR values have been reconstructed and of a better quality. Therefore, In this study, we expect to obtain PSNR values greater than 30 dB standard, which are good in terms of inpainting results.

D. Implementation Details

The entire pipeline was conceived in Python, where we also applied Tesseract OCR and diffusers libraries from hysterical face to access the pre-trained models of stable diffusion[8] for the full UI. In addition, the Inpainting process was made faster by the use of the NVIDIA GRAPHICS CARD GPU which handled the running of the images. Boot up was done with 1 batch size at the inference steps and diffusion scores were adjusted to 50 for an improved speed-quality ratio.

IV. EXPERIMENTS AND RESULTS

A. Experimental Setup

To evaluate the effectiveness of our method, we tested it on several images with varying complexity. We used both simple and complex backgrounds, different font types, and varied text sizes. Each image was preprocessed to extract the target word and apply inpainting with the prompt.

B. Results

Successful Cases: In images where the text was clearly distinguishable from the background (e.g., high contrast and simple fonts), the model successfully masked the target word and generated plausible inpainted results. For example, in a scenario where the word "STAND" was displayed on a white background, the inpainting seamlessly replaced the text "STAND" while maintaining the surrounding context. inpainted images. Inpainting effectively restores the inpainted images.

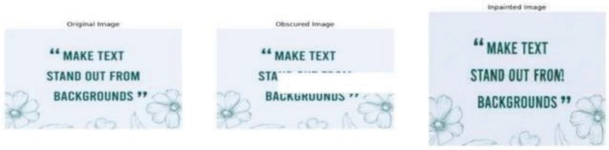
Original Image
" MAKE TEXT STAND OUT FROM BACKGROUNDS "
Obscured Image
Inpainted Image
" MAKE TEXT STAND OUT FRON! BACKGROUNDS "

Fig. 2. Comparison of original, obscured, and obscured text, maintaining readability and visual coherence with the background.

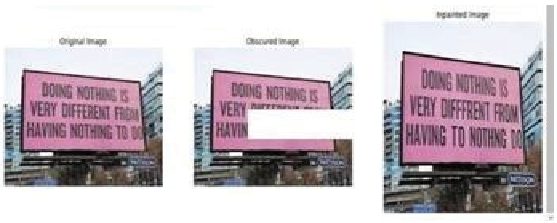
Original Image
DOING NOTHING IS VERY DIFFERENT FROM HAVING NOTHING TO DO
Obscured Image
Inpainted Image
DOING NOTHING IS VERY DIFFFRENT FROM HAVING TO NOTHING DO

Fig. 3. Comparison of original, obscured, and inpainted billboard images. Inpainting successfully restores the obscured text, maintaining readability and alignment with the original design.

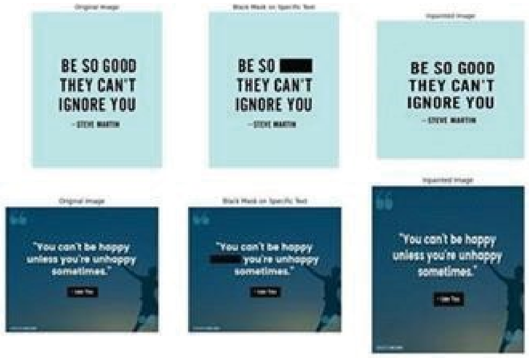
Original Image
BE SO GOOD THEY CAN'T IGNORE YOU
—STEVE MARTIN
Black Mask on Specific Text
Inpainted Image
"You can't be happy unless you're unhappy sometimes."

Fig. 4. Comparison of original, masked, and inpainted quote image. Inpainting reconstructs the masked portion of the text, achieving a readable result, though with minor inaccuracies in spelling

Challenging Cases: For images with complex backgrounds, such as those intricate textures, colored backgrounds, or distorted fonts, the OCR step often failed to detect the correct boundaries of the text. This led to poor mask generation and suboptimal inpainting. In some cases, the Stable Diffusion model created artifacts or failed to blend the new content with the background appropriately.

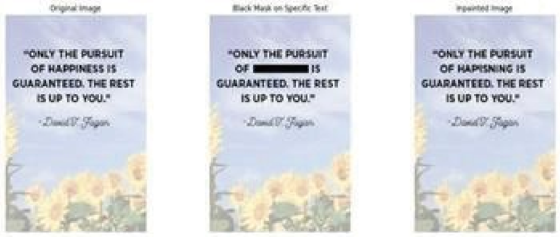
Original Image
"ONLY THE PURSUIT OF HAPPINESS IS GUARANTEED. THE REST IS UP TO YOU."
Black Mask on Specific Text
Inpainted Image
"ONLY THE PURSUIT OF HAPISNING IS GUARANTEED. THE REST IS UP TO YOU."

Fig. 5. figure shows due to colored background performance of model is reduced and it is not performing up to the mark.

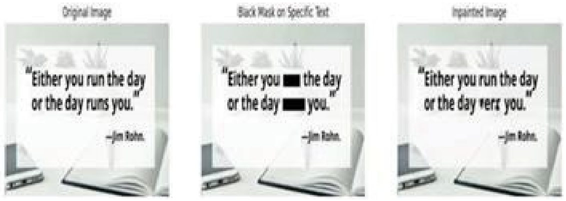
Original Image
"Either you run the day or the day runs you."
—Jim Rohn.
Black Mask on Specific Text
Inpainted Image
"Either you run the day or the day rers you."

Fig. 6. figure shows due to similar word and colored background performance of model is reduced and it is not performing up to the mark.

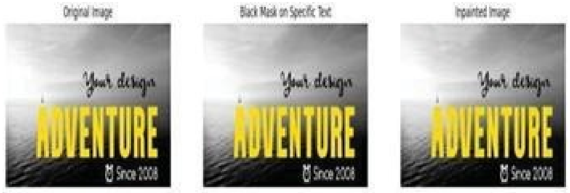

Fig. 7. figure shows due to texture background the model is unable to recognize the target word.

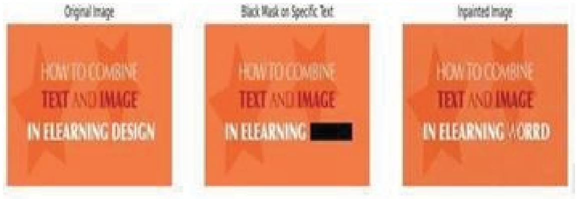
Original Image
HOW TO COMBINE TEXT AND IMAGE IN ELEARNING DESIGN
Black Mask on Specific Text
Inpainted Image
HOW TO COMBINE TEXT AND IMAGE IN ELEARNING WORRD

Fig. 8. Figure shows due to colored background performance of model is reduced and it is not performing up to the mark.

TABLE I. OCR PERFORMANCE ON DIFFERENT IMAGE TYPES

Image Type	OCR Detection Accuracy(%)	Common Issue
Simple Background	95	Minimal detection errors
Textured Background	78	Boundary misdetections
Colored Background	70	Low contrast affects Accuracy
Complex Background	60	Missed text,incorrect bounding boxes

TABLE II. INPAINTING RESULTS FOR VARIOUS BACKGROUNDS

Background Type	Inpainting Success Rate (%)	Common Issue
Simple Background	92	Minor alignment issue
Textured Background	75	Edge blending issue
Colored Background	50	Color mismatch, partial text visibility
Complex Background	30	Distortion, incomplete text integration

TABLE III. CHALLENGES WITH OCR AND INPAINTING IN DIFFERENT SCENARIOS

Scenario	OCR Challenge	Inpainting Challenge
High-Contrast Text	High accuracy, Minimal issues	Seamless inpainting, No major artifacts
Low-Contrast, Colored Background	OCR misdetects or skips text	Colour matching issues, Visible artifacts
Complex Background (Patterns)	Misaligned Bounding boxes	Blending problems, Text distortion
Distorted or Fancy Fonts	OCR fails to Recognize characters	Inaccurate text rendering

C. Constraints

1) The Limitations of Optical Character Recognition:

The performance of pytesseract depends directly on how good the image in question is. This means that text-bearing images that have a lot of background noise and detail, varying colors, or.strange fonts were misdetected and poorly edited.[11]

2) Inpainting Models Limitations:

Stable Diffusion completes very well inpainting tasks, however, it is not for the inpaint of text in this case [6].

Because of this, it often failed to generate any coherent text in areas that required text to be inserted in a very particular manner, or where the shapes of the letters needed to be defined accurately.[8][9]

V. DISCUSSION

The integration of OCR and diffusion inpainting models is a work in progress with numerous possibilities. The inherent difficulties presented by OCR in environments with convoluted backdrops point to advanced preprocessing techniques and improvements to OCR models as prerequisite proposals. In addition, while diffusion models can synthesize photorealistic images, text inpainting cannot simply be a matter of using a camera-ready model without training it for specific images containing text, or inducing other artifacts or misalignment of the inpainted text elements.

A. Improvements and Future Work

OCR Preprocessing Techniques Improvement: There exists a way to augment the efficiency of pytesseract with even more sophisticated methods of pre-operational processes such as contrast sharpening, removing any background or surrounding noise as well as text area segmentation improvement.

Texta Dependent Models of Probabilistic Diffusion: Finally the other task was paid attention to – text generation through using diffusion probabilistic models. Text parameters, such as font and font placement, will most likely be included during the training process.

1) Mixed Techniques:

Apparently, employment of attention mechanisms or even transformer models in their entirety will be incorporated in the diffusion models so as to enhance the performance of textual information detection and reconstruction.

VI. CONCLUSION

In this paper, we explored a technique for editing an image with text utilising OCR and Stable Diffusion. Unfortunately, although the technique showed great potential in controlled settings, it was limited by the presence of complex image backgrounds. By overcoming these hurdles and enhancing both the OCR and inpainting phases, we aspire to assist in more cutting-edge methods for image text elimination and alteration.

REFERENCES

[1] R. Dhariwal et al., "Diffusion Models Beat GANs on Image Synthesis," arXiv preprint arXiv:2105.05233, 2021.

[2] R. Smith, "An Overview of the Tesseract OCR Engine," Proc. Int'l Conf. Document Analysis and Recognition, pp. 629–633, 2007.

[3] "Image Inpainting for Irregular Holes Using Partial Convolutions" by Liu, G., Reda, F. A., Shih, K. J., Wang, T. C., Tao, A., & Catanzaro, B. 2018.

[4] "Deep Image Prior" by Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2018).

[5] "Image Inpainting Using Generative Adversarial Networks" by Yeh, R. A., Chen, C., Lim, T. Y., Schwing, A. G., Hasegawa-Johnson, M., & Do, M. N. (2017)

[6] J. Chen, Y. Huang, T. Lv, L. Cui, Q. Chen, and F. Wei, “TextDiffuser: Diffusion Models as Text Painters,” in Proc. NeurIPS, 2023.

[7] S. Zhu, P. Fang, C. Zhu, Z. Zhao, Q. Xu, and H. Xue, “Text Image Inpainting via Global Structure-Guided Diffusion Models,” in Proc. AAAI Conf. Artificial Intelligence, 2024. [Online]. Available: arXiv:2401.14832

[8] J. Santoso, “On Manipulating Scene Text in the Wild with Diffusion Models,” in Proc. Winter Conf. Appl. Computer Vision (WACV), 2024.

[9] S. Pathak, V. Kaushik, and B. Lall, “DiffSTR: Controlled Diffusion Models for Scene Text Removal,” arXiv:2410.21721, Oct. 2024. [Online]. Available: https://arxiv.org/abs/2410.21721

[10] W. Yang, H. Liu, and N. Liu, “STRDD: Scene Text Removal with Diffusion Probabilistic Models,” in Communications in Computer and Information Science, ISAIR, 2022.

[11] S. Tang, Y. Cao, S. Liang, Z. Jin, and K. Lai, “Scene Text Recognition That Eliminates Background and Character Noise Interference,” Applied Sciences, vol. 15, no. 7, p. 3545, 2025.

[12] P. Dhariwal and A. Nichol et al., “Diffusion Models Beat GANs on Image Synthesis,” in Proc. NeurIPS, 2021. [Online]. Available: arXiv:2105.05233

[13] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-Resolution Image Synthesis with Latent Diffusion Models,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2022.

[14] E. R. Lugmayr, K. Zhang, O. Wang, A. G. Schwing, and M. Nießner, “RePaint: Inpainting using Denoising Diffusion Probabilistic Models,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2022.

[15] G. Liu, F. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, “Image Inpainting for Irregular Holes Using Partial Convolutions,” in Proc. Eur. Conf. Computer Vision (ECCV), 2018, pp. 85–100.

[16] M. Kalbhor, S. Shinde, D. E. Popescu, and D. J. Hemanth, “Hybridization of deep learning pre-trained models with machine learning classifiers and fuzzy min–max neural network for cervical cancer diagnosis,” Diagnostics, vol. 13, no. 7, p. 1363, 2023. doi: 10.3390/diagnostics13071363

[17] S. Shinde, U. Kulkarni, D. Mane, and A. Sapkal, “Deep learning-based medical image analysis using transfer learning,” in Health Informatics: A Computational Perspective in Healthcare, Springer, 2021, pp. 19–42. doi: 10.1007/978-981-33-4367-2_2

[18] M. M. Kalbhor and S. V. Shinde, “Cervical cancer diagnosis using convolution neural network: feature learning and transfer learning approaches,” Soft Computing, pp. 1–11, 2023. doi: 10.1007/s00500-023-08599-1