# Image Inpainting for Object Removal using Diffusion-Based Technique

Sakina Jambughodawala
Prina Patel
Aneri Maniar
Nidhee Adesara

***Abstract*—Image inpainting is a process for filling the regions of the missing region in digitally produced images. The method follows the joint interpolation of the gradient/Isophotes directions (Isophotes is greek word where "Iso" means equal and "photes" means light) and the image gray levels, in which the algorithms will reach empty areas by predicting the light flow and what the structure would look like in the missing area. This approach can be computed by implementing partial differential equation. Experimental results demonstrate that diffusion-based inpainting performs effectively for small or narrow damaged regions such as scratches, text removal, and minor cracks.**

***Index Terms*—Image Inpainting, Diffusion-Based Inpainting, PDE, Image Restoration, Variational Method, Digital Image Processing, Edge Preservation, Laplacian Operator**

## I. Introduction

Digital images often suffer from damage, unwanted objects, or missing information due to factors such as aging, noise, or transmission errors. Image inpainting aims to restore such areas by filling in missing pixels so that the result looks natural to the human eye.

There are various inpainting techniques: diffusion-based and Exempler based (patch-based). Among them, diffusion-based methods rely on partial differential equations (PDEs) to spread pixel intensity values from the known areas into the missing regions, similar to the way heat diffuses over a surface. This project focuses on the diffusion-based approach for small missing regions and produces visually smooth restorations.

## II. Methodology

### A. Input and Mask Creation

The damaged image is loaded, and a binary mask is created where:

- Mask value = 1: Missing region (to be filled)
- Mask value = 0: Known region (unchanged)

### B. Diffusion Equation

The diffusion-based inpainting process is mathematically modeled using partial differential equations (PDEs) and energy minimization principles. The main formulations are summarized below.

*1) PDE-Based Diffusion Equation:*

$$\frac{\partial I(x,y,t)}{\partial t} = \nabla \cdot (D(x,y)\nabla I(x,y,t)) \tag{1}$$

This equation diffuses pixel values from known to missing regions, similar to heat flow, ensuring smooth filling.

**Where:**

- $I(x,y,t)$ – Image intensity at position $(x,y)$ and time $t$
- $D(x,y)$ – Diffusion coefficient (controls smoothness)
- $\nabla I$ – Image gradient (direction of intensity change)
- $\nabla\cdot$ – Divergence operator (measures spread)

*2) Variational / Energy Minimization Formulation:*

$$E(I) = \int_\Omega |\nabla I|^2 \, dx \, dy \tag{2}$$

Minimizes the total image energy to obtain the smoothest and most natural reconstruction.

**Where:**

- $E(I)$ – Total image energy
- $\nabla I$ – Gradient of image intensity
- $\Omega$ – Image domain (region being processed)

*3) Geometry-Guided Diffusion:*

$$\frac{\partial I}{\partial t} = \nabla \cdot \left(\frac{\nabla I^\perp}{|\nabla I|}\right) \tag{3}$$

Diffusion occurs *along* image edges, not across them, preserving important boundaries and structures (anisotropic diffusion).

**Where:**

- $\nabla I^\perp$ – Gradient perpendicular to edges
- $|\nabla I|$ – Magnitude of gradient (edge strength)

The inpainting process is governed by the heat flow or smoothness equation, that is diffusion partial differential equation (PDE):

$$\frac{\partial I}{\partial t} = \nabla \cdot (\nabla I) \tag{4}$$

where $I$ represents the image intensity. This equation gradually updates missing pixel values based on neighboring known pixels.

### C. Algorithm Steps

1) Input: Original Image and mask.
2) Convert to grayscale; normalize pixel intensities.
3) The masked area is given initial values, by interpolating or blurring nearby pixels.
4) Identify boundary pixels and calculate isophotes (gradient directions) from known regions.
5) Apply PDE-based updates (e.g., Laplacian or anisotropic diffusion) to spread pixel values smoothly along isophotes into the masked region.
6) Iteratively update pixel values until changes become minimal or a maximum iteration is reached.
7) Output: Unpainted image.

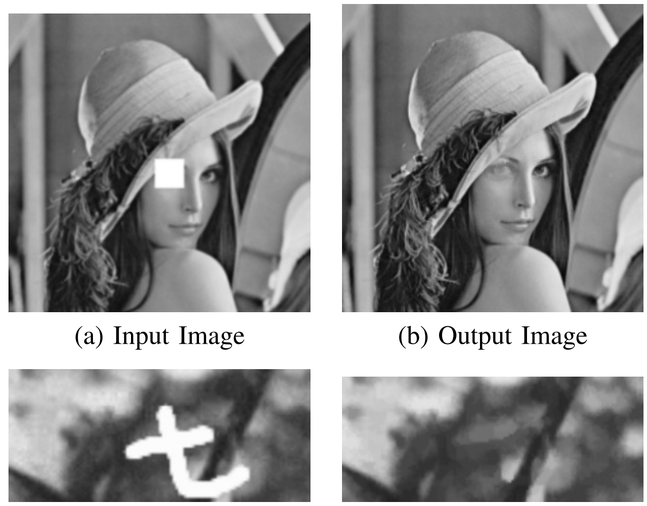
### D. Pseudocode

```
# Load image and mask
image = load_image('input.png')
mask = load_mask('mask.png')
# 1 for missing, 0 for known

# Convert to grayscale and normalize
image = convert_to_grayscale(image)
image = normalize(image)

# Initialize masked region
image[mask == 1] =
initial_fill(image, mask)

# Set parameters
diffusion_rate = 0.1
time_steps = 500

for t in range(time_steps):
    # Compute image gradient / Laplacian
    laplacian = compute_laplacian(image)

    # Update only missing pixels using
    PDE-based diffusion
    image[mask == 1] +=
    diffusion_rate * laplacian[mask == 1]

    # Optional: check for convergence and
    break if minimal change

# Output inpainted image
save_image(image, 'output.png')
```

## III. Results

The diffusion-based inpainting successfully filled missing parts of test images in the following way:

The results show that PDE-based diffusion performs well for small and narrow damaged areas, producing smooth and natural reconstructions.



(a) Input Image (b) Output Image

(c) Input Image (d) Output Image

Fig. 1. Results of diffusion-based image inpainting.

## IV. Discussion

Diffusion-based method is mathematically simple but good in repairing minor losses. It provides edge maintaining a smoothness without training data. It however has difficulties with big gaps because it does not have the contextual awareness.

Potential solutions are:

- Anisotropic diffusion to enhance edge directionality.
- Geometry-directed diffusion of improved structural continuation.

## V. Conclusion

This project shows that the method of diffusion-based inpainting that we used works well for restoring the damaged or destroyed images. It works by filling out the missing parts smoothly and naturally by spreading nearby pixel values using mathematical equations (PDEs). The method is simple, keeps edges clear, and is useful for fixing scratches, removing text, or repairing old photos.

## References


[1] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE Transactions on Image Processing*, vol. 10, no. 8, pp. 1200–1211, Aug. 2001, doi: 10.1109/83.935036.

[2] T. F. Chan and J. Shen, "Morphologically invariant PDE inpaintings," University Digital Conservancy, 2001. [Online]. Available: https://hdl.handle.net/11299/3602.

[3] Future Engineering Journal, "Volume 2 Issue 2," [Online]. Available: https://digitalcommons.aaru.edu.jo/cgi/viewcontent.cgi?article=1019&context=fej.