DataPrepare

April 2, 2025

1 Training Data Preparation

We need to preprocess our data in order to train on them. We will reuse the previously converted images from our layout detection tasks. These are present in `../test1/yolov5/spanishdocs/train/images`.

In order for our diffusion model to learn to create similar pages as in training data, we will feed it with images of the text pages with text masked with similar color to background.

To mask the text we make use of inpainting from cv2 library. For our finetuning we are using **40** images due to the limited training resources(more on this in the generation notebook). Post processing, we store these images in `../test1/yolov5/spanishdocs/train/images`.

[8]:

```
import os
import cv2
import numpy as np
import pandas as pd
from PIL import Image
import torch

# Input and output directories
input_dir = "../test1/yolov5/spanishdocs/train/images"  # Replace with your input directory
output_dir = "./preprocessed_images"     # Replace with your output directory
os.makedirs(output_dir, exist_ok=True)

# Target size for Stable Diffusion
target_size = (512, 512)

# Prompt for all images (Can be customized per image if needed)
default_prompt = "A 17th-century Spanish book page background, aged yellow parchment with subtle ink stains, worn edges, faded texture, no text"

# List to store metadata
metadata = []

def resize_and_pad_image(image, target_size=(512, 512)):
    """
    Resize the image to fit within target_size while preserving aspect ratio,␣
 ↪and pad with a background color.
    """
    img = Image.fromarray(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
    img.thumbnail(target_size, Image.Resampling.LANCZOS)  # Resize while␣
 ↪preserving aspect ratio
    new_img = Image.new("RGB", target_size, (245, 235, 200))  # Parchment␣
 ↪yellow background
    offset = ((target_size[0] - img.size[0]) // 2, (target_size[1] - img.
 ↪size[1]) // 2)
    new_img.paste(img, offset)
    return cv2.cvtColor(np.array(new_img), cv2.COLOR_RGB2BGR)

def mask_text(image, use_inpainting=True, blur_strength=31):
    """
    Mask out text by either applying a strong blur or using inpainting to␣
 ↪approximate the background.
    """
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    _, thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY_INV + cv2.
 ↪THRESH_OTSU)
    kernel = np.ones((5, 5), np.uint8)
    dilated = cv2.dilate(thresh, kernel, iterations=3)
    if use_inpainting:
        inpainted = cv2.inpaint(image, dilated, inpaintRadius=5, flags=cv2.
 ↪INPAINT_TELEA)
        return inpainted
    else:
        blurred = cv2.GaussianBlur(image, (blur_strength, blur_strength), 0)
        mask = dilated[:, :, np.newaxis] / 255.0
        image = (1 - mask) * image + mask * blurred
        return image.astype(np.uint8)

def preprocess_images(input_dir, output_dir, mask_text_option=False,␣
 ↪use_inpainting=True):
    """
    Preprocess all images in the input directory: resize, optionally mask text,␣
 ↪and save.
    """
    for filename in os.listdir(input_dir):
        if filename.lower().endswith(('.png', '.jpg', '.jpeg', '.bmp', '.
 ↪tiff')):
            filepath = os.path.join(input_dir, filename)
            image = cv2.imread(filepath)
            if image is None:
                print(f"Failed to load {filename}")
                continue

            image_resized = resize_and_pad_image(image, target_size)
            if mask_text_option:
                image_resized = mask_text(image_resized,␣
 ↪use_inpainting=use_inpainting)

            output_filename = f"preprocessed_{filename}"
            output_filepath = os.path.join(output_dir, output_filename)
            cv2.imwrite(output_filepath, image_resized)

            metadata.append({
                "file_name": output_filename,
                "prompt": default_prompt
            })
            print(f"Processed {filename} -> {output_filename}")

    metadata_df = pd.DataFrame(metadata)
    metadata_df.to_csv(os.path.join(output_dir, "metadata.csv"), index=False)
    print(f"Saved metadata.csv with {len(metadata)} entries")

# Run the preprocessing
preprocess_images(input_dir, output_dir, mask_text_option=True,␣
 ↪use_inpainting=True)
```

```
Processed Paredes-Reglas-
generales_pdf_page_8_png.rf.434c44e37a03e4a0fb2f3aceda115aa9.jpg ->
preprocessed_Paredes-Reglas-
generales_pdf_page_8_png.rf.434c44e37a03e4a0fb2f3aceda115aa9.jpg
Processed Mendo-Principe-
perfecto_pdf_page_9_png.rf.3850b34456b1acc01130d11911c47f31.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_9_png.rf.3850b34456b1acc01130d11911c47f31.jpg
Processed
PORCONES_228_35-1636_pdf_page_4_png.rf.8106b96f8f38b9a24f9823219b3b1081.jpg -> p
reprocessed_PORCONES_228_35-1636_pdf_page_4_png.rf.8106b96f8f38b9a24f9823219b3b1
081.jpg
Processed Ezcaray-Vozes_pdf_page_5_png.rf.835565c32b999aedc1a22096ea6c8c5f.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_5_png.rf.835565c32b999aedc1a22096ea6c8c5f.jpg
Processed Buendia-
Instruccion_pdf_page_6_png.rf.210a4039b56d0e0cd72535827dca5c5f.jpg ->
preprocessed_Buendia-
Instruccion_pdf_page_6_png.rf.210a4039b56d0e0cd72535827dca5c5f.jpg
Processed
PORCONES_228_35-1636_pdf_page_12_png.rf.22bed35e44ff8aba5af8b9214c5bd0cc.jpg ->
preprocessed_PORCONES_228_35-1636_pdf_page_12_png.rf.22bed35e44ff8aba5af8b9214c5
bd0cc.jpg
```

```
Processed
PORCONES_228_35-1636_pdf_page_1_png.rf.aad5e8fce184700e8827927cbb1e876d.jpg -> p
reprocessed_PORCONES_228_35-1636_pdf_page_1_png.rf.aad5e8fce184700e8827927cbb1e8
76d.jpg
Processed Ezcaray-Vozes_pdf_page_10_png.rf.8d873497b7b8f390f573c5f30c0b4a9d.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_10_png.rf.8d873497b7b8f390f573c5f30c0b4a9d.jpg
Processed Buendia-
Instruccion_pdf_page_1_png.rf.1df35c148a953090a046fde77a5d4667.jpg ->
preprocessed_Buendia-
Instruccion_pdf_page_1_png.rf.1df35c148a953090a046fde77a5d4667.jpg
Processed
PORCONES_228_35-1636_pdf_page_6_png.rf.c1367899662a00754fdc4911b1338f39.jpg -> p
reprocessed_PORCONES_228_35-1636_pdf_page_6_png.rf.c1367899662a00754fdc4911b1338
f39.jpg
Processed Constituciones-sinodales-
Calahorra-1602_pdf_page_2_png.rf.2b9299d2753d866b234619489b1f5dd9.jpg ->
preprocessed_Constituciones-sinodales-
Calahorra-1602_pdf_page_2_png.rf.2b9299d2753d866b234619489b1f5dd9.jpg
Processed Mendo-Principe-
perfecto_pdf_page_8_png.rf.ff199ef18d2cd1c3fbaddbb4a8b642a6.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_8_png.rf.ff199ef18d2cd1c3fbaddbb4a8b642a6.jpg
Processed Buendia-
Instruccion_pdf_page_3_png.rf.6ea123f8edc0f1d20692fc7e04eee4c5.jpg ->
preprocessed_Buendia-
Instruccion_pdf_page_3_png.rf.6ea123f8edc0f1d20692fc7e04eee4c5.jpg
Processed Ezcaray-Vozes_pdf_page_7_png.rf.2be3abfacae6db608a593a4f3ce53396.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_7_png.rf.2be3abfacae6db608a593a4f3ce53396.jpg
Processed Ezcaray-Vozes_pdf_page_9_png.rf.68ebf5de27c0ec8b05a846ccdc3973fa.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_9_png.rf.68ebf5de27c0ec8b05a846ccdc3973fa.jpg
Processed Paredes-Reglas-
generales_pdf_page_5_png.rf.059677c5d4df2a8a507d0a124385b3cb.jpg ->
preprocessed_Paredes-Reglas-
generales_pdf_page_5_png.rf.059677c5d4df2a8a507d0a124385b3cb.jpg
Processed Paredes-Reglas-
generales_pdf_page_3_png.rf.5e9c1aa50adc5fccb05a821e97433e88.jpg ->
preprocessed_Paredes-Reglas-
generales_pdf_page_3_png.rf.5e9c1aa50adc5fccb05a821e97433e88.jpg
Processed Paredes-Reglas-
generales_pdf_page_4_png.rf.a5e38facb1c45d03f9e23252ab9a2f02.jpg ->
preprocessed_Paredes-Reglas-
generales_pdf_page_4_png.rf.a5e38facb1c45d03f9e23252ab9a2f02.jpg
Processed Mendo-Principe-
perfecto_pdf_page_3_png.rf.2bc9e1ff16d4dac786e36ed9a8b2f9ee.jpg ->
preprocessed_Mendo-Principe-
```

```
perfecto_pdf_page_3_png.rf.2bc9e1ff16d4dac786e36ed9a8b2f9ee.jpg
Processed Ezcaray-Vozes_pdf_page_1_png.rf.49bade769c5398eabb9719458b947302.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_1_png.rf.49bade769c5398eabb9719458b947302.jpg
Processed Ezcaray-Vozes_pdf_page_6_png.rf.c6b6e89f26fc31ce459202832ec100a6.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_6_png.rf.c6b6e89f26fc31ce459202832ec100a6.jpg
Processed
PORCONES_228_35-1636_pdf_page_5_png.rf.c7759d5cc86fe7f283fdb11b828313a1.jpg -> p
reprocessed_PORCONES_228_35-1636_pdf_page_5_png.rf.c7759d5cc86fe7f283fdb11b82831
3a1.jpg
Processed Mendo-Principe-
perfecto_pdf_page_1_png.rf.fb2da1e407a5e9480d30ea08bfd507dc.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_1_png.rf.fb2da1e407a5e9480d30ea08bfd507dc.jpg
Processed Constituciones-sinodales-
Calahorra-1602_pdf_page_5_png.rf.fc4160c79bd4587e001c046330bba395.jpg ->
preprocessed_Constituciones-sinodales-
Calahorra-1602_pdf_page_5_png.rf.fc4160c79bd4587e001c046330bba395.jpg
Processed Buendia-
Instruccion_pdf_page_2_png.rf.d13da101b998175fd5e685fda829d6fc.jpg ->
preprocessed_Buendia-
Instruccion_pdf_page_2_png.rf.d13da101b998175fd5e685fda829d6fc.jpg
Processed Ezcaray-Vozes_pdf_page_4_png.rf.bd01b23c1ad23e39176c7ac54d16572a.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_4_png.rf.bd01b23c1ad23e39176c7ac54d16572a.jpg
Processed Paredes-Reglas-
generales_pdf_page_9_png.rf.04ae70feedbdba9e5bb51ee005f0c56d.jpg ->
preprocessed_Paredes-Reglas-
generales_pdf_page_9_png.rf.04ae70feedbdba9e5bb51ee005f0c56d.jpg
Processed
PORCONES_228_35-1636_pdf_page_16_png.rf.65cd838dd8d61c7938054e5805caf593.jpg ->
preprocessed_PORCONES_228_35-1636_pdf_page_16_png.rf.65cd838dd8d61c7938054e5805c
af593.jpg
Processed Buendia-
Instruccion_pdf_page_5_png.rf.ce5b1dc3c16815571b209021efcd76ec.jpg ->
preprocessed_Buendia-
Instruccion_pdf_page_5_png.rf.ce5b1dc3c16815571b209021efcd76ec.jpg
Processed Mendo-Principe-
perfecto_pdf_page_5_png.rf.8022e7063a6683e3c397d959a88c0cb0.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_5_png.rf.8022e7063a6683e3c397d959a88c0cb0.jpg
Processed Ezcaray-Vozes_pdf_page_8_png.rf.c7b6a281b9064721b341843b23c5d145.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_8_png.rf.c7b6a281b9064721b341843b23c5d145.jpg
Processed Mendo-Principe-
perfecto_pdf_page_2_png.rf.63b76733f367d53bcd90f2b61dae6842.jpg ->
preprocessed_Mendo-Principe-
```

```
perfecto_pdf_page_2_png.rf.63b76733f367d53bcd90f2b61dae6842.jpg
Processed Mendo-Principe-
perfecto_pdf_page_7_png.rf.f99990962069deaea526789913630db9.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_7_png.rf.f99990962069deaea526789913630db9.jpg
Processed Ezcaray-Vozes_pdf_page_11_png.rf.42d40c7ba469c038f52b73e586d5d676.jpg
-> preprocessed_Ezcaray-
Vozes_pdf_page_11_png.rf.42d40c7ba469c038f52b73e586d5d676.jpg
Processed Mendo-Principe-
perfecto_pdf_page_4_png.rf.d76af10933d783618ca58e9748aa0f96.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_4_png.rf.d76af10933d783618ca58e9748aa0f96.jpg
Processed Constituciones-sinodales-
Calahorra-1602_pdf_page_3_png.rf.c33a08f90cda90a79958b9ae9d8d63f0.jpg ->
preprocessed_Constituciones-sinodales-
Calahorra-1602_pdf_page_3_png.rf.c33a08f90cda90a79958b9ae9d8d63f0.jpg
Processed
PORCONES_228_35-1636_pdf_page_15_png.rf.a3de0f5a6fb48ffd7c03736b03ca210a.jpg ->
preprocessed_PORCONES_228_35-1636_pdf_page_15_png.rf.a3de0f5a6fb48ffd7c03736b03c
a210a.jpg
Processed
PORCONES_228_35-1636_pdf_page_2_png.rf.63ecd85ff9be5cf943f2576f9714f502.jpg -> p
reprocessed_PORCONES_228_35-1636_pdf_page_2_png.rf.63ecd85ff9be5cf943f2576f9714f
502.jpg
Processed Constituciones-sinodales-
Calahorra-1602_pdf_page_6_png.rf.6aed1ac2d04b54108094ddd7b2dea5bd.jpg ->
preprocessed_Constituciones-sinodales-
Calahorra-1602_pdf_page_6_png.rf.6aed1ac2d04b54108094ddd7b2dea5bd.jpg
Processed Mendo-Principe-
perfecto_pdf_page_6_png.rf.3e82d82417c1672faf6ba0e5eee1cf54.jpg ->
preprocessed_Mendo-Principe-
perfecto_pdf_page_6_png.rf.3e82d82417c1672faf6ba0e5eee1cf54.jpg
Saved metadata.csv with 40 entries
```

1.0.1 Let's Visualize the changes

[10]:

```python
import os
from PIL import Image
import matplotlib.pyplot as plt

def display_before_after(dir_before, dir_after, num_examples=5):
    """
    Displays before-and-after images side by side in a Jupyter Notebook.

    Parameters:
        dir_before (str): Directory containing the original images.
        dir_after (str): Directory containing the preprocessed images.
        num_examples (int): Number of examples to display.
    """
    # Helper function to filter image files
    def get_image_files(directory):
        return [f for f in os.listdir(directory) if f.lower().endswith(('.png',␣
 ↪'.jpg', '.jpeg', '.bmp', '.gif'))]

    # Get image files from both directories
    before_images = sorted(get_image_files(dir_before), reverse=True)[:
 ↪num_examples]
    after_images = sorted(get_image_files(dir_after), reverse=True)[:
 ↪num_examples]

    # Set up the plot
    fig, axes = plt.subplots(num_examples, 2, figsize=(10, num_examples * 3))
    for i, (before_file, after_file) in enumerate(zip(before_images,␣
 ↪after_images)):
        # Load images
        before_img = Image.open(os.path.join(dir_before, before_file))
        after_img = Image.open(os.path.join(dir_after, after_file))

        # Display before image
        axes[i, 0].imshow(before_img)
        axes[i, 0].axis('off')
        axes[i, 0].set_title("Before")

        # Display after image
        axes[i, 1].imshow(after_img)
        axes[i, 1].axis('off')
        axes[i, 1].set_title("After")

    plt.tight_layout()
    plt.show()

display_before_after(input_dir, output_dir, num_examples=3)
```

Before

After

Before

After

Before

After

Now that we are have a dataset to train on, let's proceed to train our diffusion model.