Informe técnico - TP1_P_1

Generado automáticamente a partir del notebook: /mnt/data/TP1_P_1.ipynb

Informe técnico - TP1_P_1
Generado automáticamente a partir del notebook

Resumen ejecutivo

Informe inferido a partir del notebook. El cuaderno contiene código que parece implementar una o varias soluciones programáticas para un ejercicio práctico. A partir del análisis estático se identificaron:

- Líneas de código aproximadas: 126
- Principales módulos importados: cv2, numpy, matplotlib.pyplot, os
- Número de funciones definidas (ejemplos): ecualizacion_local

Análisis del código

Líneas de código (aprox.): 126
Módulos importados (top):
- cv2 (apariciones: 1)
- numpy (apariciones: 1)
- matplotlib.pyplot (apariciones: 1)
- os (apariciones: 1)
Funciones definidas:
- ecualizacion_local: sin docstring
Llamadas / funciones más usadas (top):
- print (veces: 6)
- imshow (veces: 4)
- set_title (veces: 3)
- axis (veces: 3)
- show (veces: 2)
- range (veces: 2)
- ValueError (veces: 2)
- subplots (veces: 2)
- tight_layout (veces: 2)
- imread (veces: 1)
- FileNotFoundError (veces: 1)
- exit (veces: 1)
- figure (veces: 1)
- title (veces: 1)
- zeros_like (veces: 1)
Estructuras sintácticas detectadas:
- try/except: 2
- if: 5
- raise: 3
- for: 3

Problemas detectados y observaciones

No se detectaron problemas obvios mediante análisis estático.

Técnicas utilizadas

Técnicas detectadas y estilo de programación:
- Uso de NumPy para computación numérica/arrays.
- Visualización con matplotlib.
- Iteraciones con bucles for.
- Manejo de excepciones con try/except.

Conclusiones

Conclusiones y recomendaciones:
- Modularizar el código: definir funciones con docstrings y tests unitarios para cada componente.
- Añadir manejo explícito de excepciones y validación de entradas donde aplique.
- Documentar el objetivo del notebook en una celda markdown inicial (objetivo, datos de entrada, salida esperada).
- Evitar imports no usados; revisar dependencias y requisitos.
- Añadir ejemplos de ejecución y, si corresponde, celdas que muestren resultados y visualizaciones claramente etiquetadas.

Apéndice

Contenido extraído del notebook (resumen):

--- Celdas Markdown (primeras 5) ---
[MD 1]:
INFORME DE OBJETOS REVELADOS

El análisis se realiza contrastando la imagen original con las imágenes ecualizadas localmente:

Detalles Revelados por la Ecualización Local (LHE):

Al aplicar LHE, el contraste se amplifica solo dentro de la pequeña ventana de análisis, revelando los detalles que tenían valores de intensidad muy cercanos al fondo local.

1) Cuadrante Superior Izquierdo: Se revela un cuadrado pequeño centrado.

2) Cuadrante Inferior Izquierdo: Se revelan cuatro líneas horizontales paralelas.

3) Cuadrante Superior Derecho: No se revela ningún objeto (el cuadrante se mantiene uniforme o revela ruido uniforme).

4) Cuadrante Inferior Derecho: Se revela un círculo relleno o contorneado centrado.

5) Centro de la Imagen: Se revela la letra 'a' que estaba oculta en la zona central.

Análisis de la Ventana:

Una ventana pequeña (ej. 5x5) genera mucho ruido (granulado) pero máximo detalle.

Una ventana media (ej. 15x15) ofrece el mejor balance, revelando los objetos sin exce

--- Fragmentos de código (primeros 3) ---
[CODE 1]:
```
import cv2
import numpy as np
import matplotlib.pyplot as plt
import os
```

[CODE 2]:
```
# --- CONFIGURACIÓN Y CARGA DEL ARCHIVO REAL ---

IMAGEN_FILE = 'Imagen_con_detalles_escondidos.tif'

try:
# Cargar la imagen en escala de grises (necesario para la ecualización)
img_original = cv2.imread(IMAGEN_FILE, cv2.IMREAD_GRAYSCALE)
if img_original is None:
raise FileNotFoundError(f"El archivo '{IMAGEN_FILE}' no se pudo cargar. Verifique el nombre y que sea un archivo de imagen válido.")
print(f"Imagen '{IMAGEN_FILE}' cargada exitosamente.")

except Exception as e:
print(f"ERROR al cargar la imagen: {e}")
# Terminamos la ejecución si no se puede cargar la imagen real
exit()
```

[CODE 3]:
```
# Veamos la imagen original...
plt.figure(figsize=(10, 4))
```

```
plt.title('Imagen Original')
plt.imshow(img_original, cmap='gray')
plt.show()
```