TKINTER IMAGE VIEWER

MAY 29, 2024

JAYACHAITHANYA

Tkinter Image Viewer

About:

Tkinter image viewer is a kind of image viewer which is used to view images. With the help of the tkinter we can make our own image viewer.

Code:

```
import tkinter as tk
from tkinter import filedialog
import cv2
from PIL import ImageTk, Image
import pytesseract
```

These are used to import the things required for tkinter image viewer.

```
pytesseract.pytesseract.tesseract_cmd =
r"C:\Users\jayac\AppData\Local\Programs\Python\Python39\Scripts\pytesseract
.exe"
```

For importing pytesseract.

```
abc = tk.Tk()
abc.geometry("1200x1200")
abc.title('image browser')
my_font1 = ('times', 18, 'bold')
l1 = tk.Label(abc, text='Add Photo', width=30, font=my_font1)
l1.grid(row=1, column=1)
```

These are the code for initiating the tkinter.

```
def upload_file():
    global img, b2
    f_types = [('Jpg Files', '*.jpg')]
    filename = filedialog.askopenfilename(filetypes=f_types)
    img = cv2.imread(filename, cv2.IMREAD_GRAYSCALE)
    img = Image.fromarray(img)
    img = ImageTk.PhotoImage(img)
    b2 = tk.Button(abc, image=img)  # using Button
    b2.grid(row=3, column=1)
```

In this snippet of code, we can create a function for uploading the file.(supports .jpg format only)

```
b1 = tk.Button(abc, text='Upload File', width=20, command=upload_file)
b1.grid(row=2, column=1)
```

These lines are used to create and give the result for the function.

```
def process_image():
    global img
    # Convert the image to grayscale
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    # Apply bilateral filter
    bilateral_filtered_image = cv2.bilateralFilter(gray, 5, 175, 175)

    # Apply Canny Edge Detection
    edges = cv2.Canny(bilateral_filtered_image, 100, 200)

    # Convert the processed image into a PIL Image object to display in the
tkinter window
    processed_img = Image.fromarray(edges)
    processed_img = ImageTk.PhotoImage(processed_img)
    b3 = tk.Button(abc, image=processed_img)  # using Button
    b3.grid(row=4, column=1)
```

This code is for converting the image into biterial filter, grayscale conversion and canny edges.

```
b3 = tk.Button(abc, text='Process Image', command=process_image)
b3.grid(row=3, column=1)
```

These lines are used to display the process button.

```
abc.mainloop()
```

To loop.

Full code:

```
#  used to open images in the file manager using
tkinter.

import tkinter as tk
from tkinter import filedialog
import cv2
from PIL import ImageTk, Image
import pytesseract

pytesseract.pytesseract.tesseract_cmd =
r"C:\Users\jayac\AppData\Local\Programs\Python\Python39\Scripts\pytesseract
.exe"

abc = tk.Tk()
abc.geometry("1200x1200")
abc.title('image browser')
my_font1 = ('times', 18, 'bold')
l1 = tk.Label(abc, text='Add Photo', width=30, font=my_font1)
l1.grid(row=1, column=1)
```

```
def upload_file():
    global img, b2
    f_types = [('Jpg Files', '*.jpg')]
    filename = filedialog.askopenfilename(filetypes=f_types)
    img = cv2.imread(filename, cv2.IMREAD_GRAYSCALE)
    img = Image.fromarray(img)
    img = ImageTk.PhotoImage(img)
    b2 = tk.Button(abc, image=img)  # using Button
    b2.grid(row=3, column=1)


b1 = tk.Button(abc, text='Upload File', width=20, command=upload_file)
b1.grid(row=2, column=1)


def process_image():
    global img
    # Convert the image to grayscale
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    # Apply bilateral filter
    bilateral_filtered_image = cv2.bilateralFilter(gray, 5, 175, 175)

    # Apply Canny Edge Detection
    edges = cv2.Canny(bilateral_filtered_image, 100, 200)

    # Convert the processed image into a PIL Image object to display in the
tkinter window
    processed_img = Image.fromarray(edges)
    processed_img = ImageTk.PhotoImage(processed_img)
    b3 = tk.Button(abc, image=processed_img)  # using Button
    b3.grid(row=4, column=1)


b3 = tk.Button(abc, text='Process Image', command=process_image)
b3.grid(row=3, column=1)

abc.mainloop()
```