Optical character recognition

Aplicatia prezentata este o solutie OCR (Optical Character Recognition) pentru extragerea textului din imagini. OCR este o tehnologie care permite citirea automata a textului din imagini, precum documente, capturi de ecran sau alte imagini.

Aplicatia permite utilizatorului sa selecteze o imagine de pe computer prin intermediul unei ferestre de dialog "filedialog", care permite utilizatorului sa navige prin folderele sale si sa selecteze imaginea dorita. De asemenea, utilizatorul poate introduce limba textului din imagine prin intermediul unei ferestre de dialog "simpledialog", in care este intrebat sa introduca limba textului din imagine (pentru limba Romana, utilizatorul trebuie sa scrie "ron"). Apoi, utilizand biblioteca Pytesseract, textul este extras din imagine si afisat in consola.

Acesta este un exemplu de utilizare: cand utilizatorul selecteaza imaginea "image.jpg" si introduce limba "ron", aplicatia va extrage si afisa textul din imagine in limba romana. Acest lucru poate fi util pentru citirea automata a textului din documente, capturi de ecran sau alte imagini. Aplicatia poate fi utilizata pentru a automatiza procesele de citire si prelucrare a textului din imagini, economisind timp si efort pentru utilizator.

In cazul in care utilizatorul nu selecteaza o imagine sau nu introduce limba textului din imagine, aplicatia va afisa un mesaj de eroare.

In ansamblu, aplicatia prezentata este o solutie rapida si eficienta pentru extragerea textului din imagini, fiind utila pentru automatizarea proceselor de citire si prelucrare a textului din imagini. Acest lucru poate economisi timp si efort pentru utilizator.

In acest fisier, sunt folosite trei biblioteci diferite:

Pytesseract: Aceasta este o biblioteca de recunoastere a caracterelor optice (OCR) care permite extragerea textului din imagini. Utilizeaza un motor OCR open-source, Tesseract, pentru a recunoaste textul din imagini. Functia "pytesseract.image\_to\_string" este utilizata pentru a extrage textul din imagine si este pasul principal pentru extragerea textului.

PIL (Python Imaging Library): Aceasta biblioteca este utilizata pentru deschiderea imaginii selectate de utilizator. Functia "Image.open" este utilizata pentru a deschide imaginea si a pregati-o pentru procesul OCR.

Tkinter: Aceasta este o biblioteca standard Python pentru interfata grafica utilizator (GUI). Este folosita pentru a crea ferestre de dialog pentru selectarea imaginii si introducerea limbii textului. Functiile "filedialog" si "simpledialog" sunt utilizate pentru acest scop, permitandu-i utilizatorului sa selecteze o imagine de pe computer si sa introduca limba textului din imagine, respectiv.

\*\*\*\*\* "Ron" este folosit ca valoare implicita pentru limba textului din imagine. Aceasta este pentru ca "ron" este codul pentru limba romana, si acesta este setat ca valoare implicita in cazul in care utilizatorul nu specifica o limba diferita. In cazul in care utilizatorul introduce o alta limba, aceasta va fi utilizata in locul "ron" ca parametru pentru functia "pytesseract.image\_to\_string", permitandu-i motorului OCR sa recunoasca textul din imagine in limba selectata. De exemplu, daca utilizatorul introduce "eng" ca limba, motorul OCR va recunoaste textul in limba engleza.

Aici putem verifica daca Tesseract suporta limba Romana si care este acronimul pentru aceasta:

https://github.com/tesseract-ocr/tessdata/

https://tesseract-ocr.github.io/tessdoc/Data-Files-in-different-versions.html

Pasi instalare Tesseract:

Instalati biblioteca Pytesseract folosind pip. Deschideti un terminal sau o linie de comanda si tastati urmatorul comanda:

pip install pytesseract

Instalati biblioteca PIL (Python Imaging Library) folosind pip. Tastati urmatorul comanda in terminal sau linia de comanda:

pip install pillow

Descarcati fisierul de date "ron.traineddata" pentru limba romana de pe site-ul oficial al Tesseract OCR sau de pe alte surse de internet si salvati-l intr-un folder numit "tessdata" .

Setati variabila de mediu TESSDATA\_PREFIX ca sa pointeze catre folder-ul "tessdata" . In cazul in care utilizati Windows, puteti face acest lucru prin intermediul Control Panel - System and Security - System - Advanced system settings - Environment Variables. Adaugati un nou PATH si setati-l catre folder-ul "tessdata" .

Verificati daca bibliotecile si fisierul de date sunt instalate corect ruland un exemplu de cod care foloseste Pytesseract.

Importam biblioteca Pytesseract, care este utilizata pentru a extrage textul din imagini.

Exemplu:

text = pytesseract.image\_to\_string(image, lang='ron')

Importam biblioteca PIL (Python Imaging Library), care este utilizata pentru a deschide si manipula imagini.

Exemplu:

image = Image.open(image\_path)

Importam biblioteca Tkinter, care este utilizata pentru a crea interfete grafice pentru utilizator.

Exemplu:

root = tk.Tk()

Importam functia filedialog din Tkinter, care este utilizata pentru a permite utilizatorului sa selecteze o imagine de pe computer.

Exemplu:

file\_path = filedialog.askopenfilename()

Importam functia simpledialog din Tkinter, care este utilizata pentru a permite utilizatorului sa introduca o valoare (in acest caz, limba textului din imagine).

Exemplu:

lang = simpledialog.askstring("Language", "Introduceti limba textului din imagine (pentru limba Romana scrieti 'ron'):")

Definim o functie numita ocr\_extractor, care ia doua argumente: image\_path (calea catre imagine) si language (limba textului din imagine).

Exemplu:

def ocr\_extractor(image\_path, language='ron'):

Deschidem imaginea utilizand metoda open din PIL.

Exemplu:

image = Image.open(image\_path)

Utilizam metoda image\_to\_string din Pytesseract pentru a extrage textul din imagine si a specifica limba textului prin parametrul lang.

Exemplu:

text = pytesseract.image\_to\_string(image, lang='ron')

Returnam textul extras.

Exemplu:

return text

Definim o functie numita select\_image, care nu ia niciun argument.

Exemplu:

def select\_image():

Cream o fereastra Tkinter care este ascunsa prin apelarea metodei withdraw.

Exemplu:

root.withdraw()

Utilizam metoda askopenfilename din filedialog pentru a permite utilizatorului sa selecteze o imagine de pe computer.

Exemplu:

file\_path = filedialog.askopenfilename()

Verificam daca utilizatorul a selectat o imagine. Daca da, intrebam utilizatorul pentru limba textului din imagine prin apelarea metodei askstring din simpledialog.

Exemplu:

if file\_path:

lang = simpledialog.askstring("Language", "Introduceti limba textului din imagine (pentru limba Romana scrieti 'ron'):")

Utilizam functia ocr\_extractor pentru a extrage textul din imagine, specificand calea catre imagine si limba textului ca argumente.

Exemplu:

text = ocr\_extractor(file\_path, lang)

Afisam textul extras.

Exemplu:

print(text)

Daca utilizatorul nu a selectat o imagine, afisam un mesaj de eroare.

Exemplu:

else:

print("Error: No file selected.")

Apelam functia select\_image pentru a incepe procesul de extragere a textului din imagine.

Exemplu:

select\_image()

In ansamblu, acest cod permite utilizatorului sa selecteze o imagine de pe computer, sa introduca limba textului din imagine si apoi sa extraga textul din imagine folosind biblioteca Pytesseract si sa afiseze textul extras.