**Osnovna manipulacija medicinskom slikom u *Python*-u**

Ime i prezime Jovan Tanasijevic Indeks 2019/0054

**ZADACI – odgovore na pitanja upisati crvenom bojom, a slike koje su rezultat primenjenih funkcija iskopirati u ovaj templejt nakon odgovarajućeg teksta zadatka u prostoru koji je za to predviđen. Programski kôd i ovaj popunjen templejt postaviti na MS Teams (u okviru “Assignments”).**

1. Otvoriti *Spyder* okruženje.
2. Importovati *imageio* biblioteku koja omogućava čitanje i pisanje slika pomoću funkcije:

*import imageio*

1. Importovati *pyplot* funkciju biblioteke *matplotlib* koja omogućava grafički prikaz pomoću funkcije:

*import matplotlib.pyplot as plt*

1. Importovati *numpy* biblioteku za numeričke proračune pomoću funkcije:

*import numpy as np*

1. Importovati *os* biblioteku za komunikaciju sa operativnim sistemom pomoću funkcije:

*import os*

1. Zatvoriti sve eventualno otvorene grafike funkcijom *plt.close("all")*.
2. Učitati sliku *IM-0001-0001.dcm* koja je u DICOM formatu pomoću funkcije:

*im=imageio.imread('IM-0001-0001.dcm')*

1. Koji je tip podataka učitana slika? <class 'imageio.core.util.Array'>Koristiti funkciju *print(type(im)).*
2. Koja je dimenzija slike? (512, 512)Koristiti funkciju *im.shape.*
3. Koliki je *im[0,0]* element slike? -1024
4. Kako izgleda podmatrica *im[11:14,290:293]? [[ -995 -1000 -999] [-1002 -993 -1004] [ -995 -998 -1007]]*
5. Pročitati meta podatke slike, tj. podatke iz zaglavlja DICOM datoteke pomoću *im.meta*.
6. Pročitati iz zaglavlja DICOM datoteke koji sve meta podaci postoje u saglavlju koristeći *im.meta.keys()*.
7. Pročitati koji modalitet medicinskog slikanja je korišćen pri snimanju slike pomoću *im.meta['Modality']*. CT Pročitati kolika je rezolucija u milimetrima po svakoj od osa čitanjem meta podatka *sampling*. (0.9765625, 0.9765625)
8. Prikazati sliku pomoću funkcije *plt.imshow(im, cmap='gray').* Kolor mapa (cmap) je podešena ovom funkcijom na *'gray'*. Prikazati sliku u nekoj drugoj paleti, npr. *'rainbow'*.

Slika (gray)



Slika(rainbow)

A picture containing text

Description automatically generated

1. Prikazati sliku bez ispisanih brojeva po *x* i *y* osi pomoću funkcije *plt.axis('off')*.
2. Učitati slike *IM-0001-0002.dcm* i *IM-0001-0003.dcm*. Kreirati trodimenzionalni niz od slika *IM-0001-0001.dcm*, IM-0001-0002.dcm i *IM-0001-0003.dcm* pomoću funkcije np.stack(). Koja je dimenzija kreiranog steka (koristiti *shape* funkciju za odredjivanje dimenzije)? (3, 512, 512)
3. Izlistati nazive svih datoteka koje sadrže slike i nalaze se u direktorijumu *CT HEAD-NK 5.0 B30s* pomoću funkcije *os.listdir()*.
4. Učitati sve slike koje se nalaze u direktorijumu *CT HEAD-NK 5.0 B30s* pomoću funkcije:

*vol\_83=imageio.volread('CT HEAD-NK 5.0 B30s')*

1. Odrediti dimenziju promenljive vol\_83. (83, 512, 512)
2. Pročitati kolika je rezolucija u milimetrima po svakoj od osa čitanjem meta podatka *sampling* pomoću funkcije *d0,d1,d2=vol\_83.meta['sampling'].*\_d0=3.3999999999999995 d1=0.9765625 d2=0.9765625

Koliko je vidno polje modaliteta slikanja koji je korišćen? 3.2424926757812496

1. Izdeliti prozor za grafički prikaz na 3x3 segmenata pomoću funkcije:

*fig,axes=plt.subplots(nrows=3, ncols=3)*

Pomoću funkcije *axes[0,0].imshow(vol\_83[:,256,:], cmap='gray')* prikazati 256. frontalni slajs u nultom redu i nultoj koloni prozora za grafički prikaz. Isključiti oznake brojeva po osama pomoću *axes[0,0].axis('off')*. Pomoću for petlje uraditi isključivanje oznaka brojeva po osama za sve segmente. Preskalirati razliku u d0 i d2 *sampling*-u duž *z* i *x* ose i prikazati preskaliranu sliku desno od prethodne pomoću funkcije *axes[0,1].imshow(vol\_83[:,256,:], cmap='gray', aspect=d0/d2)*.

Prikazati preskalirani 40. transverzalni slajs u segmentu prozora koji je u prvom redu i prvoj koloni.

Prikazati preskalirani 256. sagitalni slajs u segmentu prozora koji je u drugom redu i prvoj koloni.

3x3 prozor sa slikama

Qr code

Description automatically generated

1. Sačuvati program kao *01\_python\_ime\_prezime.py*.
2. U *Spyder* okruženju otvoriti opciju *Tools/Preferences/IPython console/Graphics/* i pogledati šta je podešeno za *Backend*: *inline* ili *automatic*. *Inline* opcija omogućava da se grafici prikazuju unutar konzole, a *automatic* opcija omogućava da se grafici prikazuju u posebnim prozorima. Izabrati suprotnu opciju od one koja je već selektovana, restartovati *Spyder* i pokrenuti program ponovo. Uočiti razliku načinu prikaza grafika.
3. Program dopuniti čitanjem *imageinverse\_input.png* datoteke i čuvanjem nultog slajsa u jpg format pomoću funkcija *imageio.imread()* i *imageio.imwrite()*, respektivno. Novi prozor za prikaz png slike se otvara funkcijom *plt.figure()*.

Učitana PNG slika

Graphical user interface

Description automatically generated with low confidence