In [509…
```
import urllib.request
import ssl
ssl._create_default_https_context = ssl._create_unverified_context
from tkinter import *
from tkinter.font import Font
import cv2
import numpy as np
import tkinter as tk
```

In [510…
```
#https://www.photofunny.net/img/before-after/64_before.jpg
def negationofimage():
    global a
    foto = cv2.imread('./DIPFoto' + str(a) + '.jpg',3)
    fotonew = cv2.resize(foto, (400,400))
    negationfoto = 255-foto
    negationnew = cv2.resize(negationfoto, (400,400))

    yanyananegation = np.concatenate((fotonew, negationnew), axis=1)

    cv2.imshow('NegationFoto', yanyananegation)

    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

In [511…
```
def gausskernel():
    global a
    img = cv2.imread('./DIPFoto' + str(a) + '.jpg', 3)
    imgnew = cv2.resize(img, (400,400))
    blur = cv2.GaussianBlur(img, (3,3),0)
    blurnew = cv2.resize(blur, (400,400))

    yanyanagaussian = np.concatenate((imgnew, blurnew), axis=1)

    cv2.imshow('GaussianFoto', yanyanagaussian)

    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

In [512…
```
#https://assets.markallengroup.com/article-images/235348/girl-4388310_1920.jpg
#https://cc-prod.scene7.com/is/image/CCProdAuthor/gaussian-blur_P3a_690x450?$pjpeg$&jpegSize=200&wid=690
def average():
    global a
    foto = cv2.imread('./DIPFoto' + str(a) + '.jpg', 3)
    fotonew = cv2.resize(foto, (400,400))

    kernel = np.ones((5,5),np.float32)/25

    kernelfoto = cv2.filter2D(foto,-1,kernel)
    kernelnew = cv2.resize(kernelfoto, (400,400))

    yanyanaconvolution = np.concatenate((fotonew, kernelnew), axis=1)
    cv2.imshow('AverageFilter', yanyanaconvolution)
    cv2.waitKey(0)
    cv2.destroyAllWindows()

#https://www.mathworks.com/help/examples/images/win64/CompareResultsOfAveragingFilterAndMedianFilterExample_02.
#https://roflmaostc.github.io/Noise.jl/dev/images/awg_img_color_noise.png
def median():
    global a
    foto = cv2.imread('./DIPFoto' + str(a) + '.jpg', 3)
    fotonew = cv2.resize(foto, (400,400))
    median = cv2.medianBlur(foto,3)
    mediannew = cv2.resize(median, (400,400))

    yanyanamedian = np.concatenate((fotonew,mediannew), axis=1)
    cv2.imshow('MedianFilter', yanyanamedian)
    cv2.waitKey(0)
    cv2.destroyAllWindows()

#https://softwarebydefault.files.wordpress.com/2013/05/laplacian_of_gaussian.jpg
def laplacian():
    b=3
    global a
    kernel1 = np.array([[-1,-1,-1],
                        [-1,8,-1],
                        [-1,-1,-1]])

    foto = cv2.imread('./DIPFoto' + str(a) + '.jpg', 3)
    fotonew = cv2.resize(foto, (400,400))
    filter1 = cv2.filter2D(foto, -1, kernel1)
    #filteredfoto = foto-filter1
    fotoson = cv2.resize(filter1, (400,400))
    yanyanalaplacian = np.concatenate((fotonew,fotoson), axis=1)
    cv2.imshow('Laplacian', yanyanalaplacian)
    cv2.waitKey(0)
    cv2.destroyAllWindows()
```

In [513…
```
### Kullanıcıdan input olarak fotoğraf urlsi alan ve bunu klasörde isimlendirerek kaydeden kısım.
a = 0
def linklistesi():
    opener = urllib.request.build_opener()
    opener.addheaders = [('User-agent',
    'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/5
    urllib.request.install_opener(opener)
    urlstack = []
    urlstack.append(str(getlink.get()))
    print(urlstack)
    for url in urlstack:
        global a
        a = a + 1
        urllib.request.urlretrieve(url, 'DIPFoto' + str(a) + '.jpg')

def selectfoto():
    global a

    a = getphotoid.get()

def delete():
    getlinkentry.delete(0,'end')


window = Tk()
window.geometry("420x280")
window.title("Digital Image")
buttonfont = Font(family="Arial", size=10, weight="bold", slant="italic")

Label(window, text="Photograph Link: ").grid(row=0, column=0, pady=5, sticky=E)
getlink = StringVar()
getlinkentry = Entry(window, width=40, textvariable=getlink)
getlinkentry.grid(row=0, column=1)

mybutton = Button(window, text = "Delete", command = delete, font=buttonfont)
mybutton.place(x=367, y=3)


Label(window, text="Photograph Number: ").grid(row=2, column=0, pady=5, sticky=E)
getphotoid = StringVar()
getphotoid = Entry(window, width=20, textvariable=getphotoid)
getphotoid.place(x=122, y=35)

photoid = Button(window, text="Photo ID Select", command=selectfoto, width=15, height=1, font=buttonfont)
photoid.place(x=260, y=35)

findphoto = Button(window, text="Download Photo", command=linklistesi, width=30, height=1, font=buttonfont)
findphoto.place(x=100, y=75)

gaussfiltre = Button(window, text="Gaussian Blur", command=gausskernel, width=20, height=2, font=buttonfont)
gaussfiltre.place(x=30, y=120)

medianfiltre = Button(window, text="Median Filtre", command=median, width=20, height=2, font=buttonfont)
medianfiltre.place(x=220, y=120)

averagefilter = Button(window, text="Average Filter", command=average, width=20, height=2, font=buttonfont)
averagefilter.place(x=30, y=170)

negationfilter = Button(window, text="Negation", command=negationofimage, width=20, height=2, font=buttonfont)
negationfilter.place(x=220, y=170)

laplacianfilter = Button(window, text="Laplacian", command=laplacian, width=20, height=2, font=buttonfont)
laplacianfilter.place(x=120, y=220)
window.mainloop()
```

```
['https://affinity.help/photo/shared/filter_gaussianblur_after.jpg']
['https://cc-prod.scene7.com/is/image/CCProdAuthor/gaussian-blur_P3a_690x450?$pjpeg$&jpegSize=200&wid=690']
```

In [ ]: