

```

# image_sequence_to_mp4.py

import inspect
import os.path
import glob
import platform
import os
import ffmpeg_utils
import stat

# This variable will store the name of image sequences. For example,
# { 'wavey_particles_1' : 'jpg',
#   'wavey_particles_2' : 'jpg',
#   'wavey_particles_3' : 'tif' }
namesDB = {}

def get_parent_dir():
    # This is the most robust way of getting the full path to the
    # parent directory that contains a python script. This code
    # works if the script is run from Cutter or from a terminal or
    # Windows CMD window.
    scriptpath = inspect.getframeinfo(inspect.currentframe()).filename
    parent_dir = os.path.dirname(os.path.abspath(scriptpath))
    return parent_dir
#-----

def get_fullfile_names(parent_dir):
    all_images = []

    glob_pattern = parent_dir + "/*.jpg"
    jpgs = glob.glob(glob_pattern)
    if len(jpgs) > 0:
        all_images.extend(jpgs)

    glob_pattern = parent_dir + "/*.tif"
    tifs = glob.glob(glob_pattern)
    if len(tifs) > 0:
        all_images.extend(tifs)

    glob_pattern = parent_dir + "/*.png"
    pngs = glob.glob(glob_pattern)
    if len(pngs) > 0:
        all_images.extend(pngs)

    return all_images
#-----

def get_name_image_type(full_name):
    # Split the filename into its parts and grab the name
    # excluding the numeric file extension and its file type.
    filename = os.path.basename(full_name)
    parts = filename.split('.')
    name_only = parts[0]
    image_type = parts[2]
    return [name_only, image_type]
#-----

```

```

def add_name_to_DB(name, img_type):
    if name not in namesDB:
        namesDB[name] = img_type
#-----
#-----

parent_dir = get_parent_dir()
full_names = get_fullfile_names(parent_dir)

for full_name in full_names:
    name, img_type = get_name_image_type(full_name)
    add_name_to_DB(name, img_type)

if len(namesDB) == 0:
    print("Sorry cannot continue - could not find any images.")
else:
    # name_only, image_type = get_name_image_type(full_name)

    # Open a file that will contain the ffmpeg command
    # that will combine the jpegs into a single mp4 movie.
    # However, we must be careful to use the .bat ONLY
    # if we are on Windows.
    if os.name == 'posix':
        out_path = parent_dir + '/convertJPGtoMP4'
    elif os.name == 'nt':
        out_path = parent_dir + '/convertJPGtoMP4.bat'

    file_out = open(out_path, 'w')

    keys = namesDB.keys()
    for key in keys:
        name = key
        image_type = namesDB[key]
        if os.name == 'nt':
            command = ffmpeg_utils.WIN_seq_to_mpeg(name, image_type)
        else:
            command = ffmpeg_utils.POSIX_seq_to_mpeg(name, image_type)
        file_out.write(command)
    file_out.close()
    if os.name == 'posix':
        os.chmod(out_path, stat.S_IRWXU) #makes mp4 file executable

if __name__ == '__main__':
    print('The Windows ffmpeg script has been created.')

```