Sepehr Kerachi

Abstract

This documentation contains the necessary changes for the lfbnet repo. Heading 1 of this document is the main folder of the LFBNet repo which contain the codes (except the data loader). Heading 2 is the name of the python script. And heading 3 is the name of the python class or function which should have the necessary changes.

The final changes are the main branch of the repo given below:
https://github.com/sepehratwork/LFBNet_SepehrKerachi.git

sepehratwork@gmail.com

Contents

Utilities 2
compute_surrogate_features.py 2
read_3D_nifti_mask_image_compute_TMTV_Dmax_save_as_csv_file.py 2
train_valid_paths.py 2
Preprocessing 3
Preprocessing.py 3
read_pet_gt_resize_crop_save_as_3d_andor_mip 3
Network Architecture 4
get_conv_blocks.py 4
UpConvLayer 4
lfbnet.py 4
get_default_config 4
LfbNet 4
Losses 6
losses.py 6
Postprocessing 7
Postprocessing.py 7
Data Loader 8
data_loader.py 8
DataLoader 8

Utilities

compute_surrogate_features.py

No changes!

read_3D_nifti_mask_image_compute_TMTV_Dmax_save_as_csv_file.py

No changes!

train_valid_paths.py

No changes!

Preprocessing

Preprocessing.py

read_pet_gt_resize_crop_save_as_3d_andor_mip

- output_resolution = [128, 128, 256] => [64, 64, 128]

For the sake of reducing the size of dataset and model.

- save_3D: bool = False => True

Changing the default value of the function's save_3D's parameter from False to True

Network Architecture

get_conv_blocks.py

UpConvLayer

- conv_upsampling: str = '2D' => '3D'

Changing the default value of conv_upsampling parameter of class UpConvLayer from string '2D' into '3D'.

- self.strides = [2, 2] => [2, 2, 2]

Changing the value of strides parameter of class UpConvLayer from [2, 2] into [2, 2, 2] so that it would be used for 3D input images

- self.kernel_size = [2, 2] => [2, 2, 2]

Changing the value of kernel_size parameter of class UpConvLayer from [2, 2] into [2, 2, 2] so that it would be used for 3D input images

lfbnet.py

get_default_config

- dimension: int = 3

Changing the value of dimension parameter of function get_default_config from 2 into 3 so that it would be used for 3D input images.

LfbNet

- input_image_shape = [128, 256, 1] => [64, 64, 128, 1]

- skipped_input.append(

  ```
  [int(decoder_input_shape[0] * (2 ** stage)), int(decoder_input_shape[1] * (2 ** stage)),
   int(base_num_features * (2 ** (num_layers - (1 + stage))))])
  ```

  =>
  skipped_input.append(

```
[int(decoder_input_shape[0] * (2 ** stage)), int(decoder_input_shape[1] * (2 ** stage)),
 int(decoder_input_shape[2] * (2 ** stage)), int(base_num_features * (2 ** (num_layers - (1 + stage))))])
```

-
  ```
  num_output_features = int(self.base_num_features * (2 ** (self.num_layers - (2 + decoder_stage))))
  current_up_conv = UpConvLayer(current_up_conv, num_output_features=num_output_features,
                  conv_upsampling="2D").Up_conv_layer()
  =>
  num_output_features = int(self.base_num_features * (2 ** (self.num_layers - (2 + decoder_stage))))
  current_up_conv = UpConvLayer(current_up_conv, num_output_features=num_output_features,
                  conv_upsampling="3D").Up_conv_layer()
  ```

-
  ```
  current_up_conv = UpConvLayer(current_up_conv, num_output_features=num_output_features,
                  conv_upsampling="2D").Up_conv_layer()
  =>
  current_up_conv = UpConvLayer(current_up_conv, num_output_features=num_output_features,
                  conv_upsampling="3D").Up_conv_layer()
  ```

Losses

losses.py

No changes!

Postprocessing

Postprocessing.py

No changes!

Data Loader

data_loader.py

DataLoader

Changes needed for get_nii_files_path function of the class DataLoader but could not be figured out. The images could not be loaded with `nib.load(path).dataobj`!