Introduction to some documents of solar energy project model

1. config.py

The config.py file contains the settings of all command parameters.

The following are the command parameters:

--cal_mean_std_path : The path of calculating the mean and standard deviation of the training set.

--mask : Whether to add a mask to the picture (0 is not added, 1 is added).

--source_train_path : path of original training set (Use when dividing data sets).

--source_test_path : path of original test set (Use when dividing data sets).

--target_train_path : training set path after dividing data set (Use when dividing data sets).

--target_val_path : validation set path after dividing data set (Use when dividing data sets).

--target_test_path : test set path after dividing data set (Use when dividing data sets).

--train_path : path of train dataset (Used during model training).

--val_path : path of validation dataset (Used during model training).

--test_path : path of test dataset (Used in model prediction).

--data_type : the data types of training and testing (GHI, DHI, DNI).

--model_name : model name that will be used (ResNet50, ResNet152).

--save_path : Model and result saving path.

--load_model : Model path to be loaded during model prediction.

--epoch : train epochs.

--batch_size : batch size (Number of pictures entered per batch).

--num_workers : Number of threads used when loading data.

--lr : learning rate of model.

--momentum : momentum of learning.

--weight_decay : Parameters for weight decay.

2. cal_mean_std.py

Calculate the mean and variance of the training set. Then output the calculation results.

Run command:

```
python cal_mean_std.py --cal_mean_std_path [Path of original data training set] --mask [0 is not added, 1 is added]
```

3. split_dataset.py

The original data set is divided into training, verification and test sets.

Run command:

```
python split_dataset.py --source_train_path [path of original training set] --source_test_path [path of original test set] --target_train_path [training set path after dividing data set] --target_val_path [validation set path after dividing data set] --target_test_path [test set path after dividing data set]
```

4. dataset.py

The dataset.py file contains the processes of central clipping, standardization, and masking of the dataset.

5. ResNet50.py

This file contains the construction of resnet50 model architecture.

6. ResNet152.py

This file contains the construction of resnet152 model architecture.

7. train.py

This file contains the process of training and saving the model.

Run command:

```
python train.py --data_type [ghi/dhi/dni] --train_path [path of train dataset] --val_path [path of validation dataset] --model_name [ResNet50/ResNet152] --save_path [path to save results] --mask [0/1] --num_workers [number of threads used when loading data]
```

8. eval_test.py

This file contains the process of loading the model, evaluating the test set and saving the results.

Run command:

```
python eval_test.py --load_model [model path to be loaded] --data_type [ghi/dhi/dni] --test_path [path of test dataset] --model_name [ResNet50/ResNet152] --save_path [path to save prediction results] --mask [0/1] --num_workers [number of threads used when loading data]
```