**data\_yasudaとpy\_yasudaディレクトリには、安田が作成したファイルが入っている。**

* data\_yasuda … サンプルデータが入っている
* py\_yasuda … pythonファイルが入っている
* # 変更箇所と書かれている部分の値を適宜変更する

**～再構成の精度を評価（実験１：SIRENを通して出来た学習済みモデル（model\_final.pth）に、好きな座標数をいれて画像を再構成する。その画像をオリジナル画像と比較する。）（全体画像をSIRENに入力する）～**

1. resized\_no\_resize.py

* 1024×1024の画像を縮小する（奇数行列だけの値を抽出する）方法
  + Opencvのresizeではない
* 実行コード：python resized\_no\_resize.py
* 出力データ（サンプルデータ）：240613\_2204\_EGFP\_resized\_512x512\_no\_resize.tiff

1. train\_img.py

* 上記の画像をSIRENに通す。座標数は1024とする。
* 実行コード：

python experiment\_scripts/train\_img.py --model\_type=sine --experiment\_name=experiment\_240613\_2204\_EGFP\_resized\_512x512\_no\_resize\_sine\_10000\_w30

* + --experiment\_nameは好きなディレクトリ名を記入する
* 出力データ（サンプルデータ）：experiment\_240613\_2204\_EGFP\_resized\_512x512\_no\_resize\_sine\_10000\_w30
  + SIRENを通して出来た学習済みモデル（model\_final.pth）等

1. explore\_siren\_ipynb\_eval5.py

* 上記の学習済みモデル（model\_final.pth）を用いて、好きな座標数を入力し、画像を再構成する
* 勾配情報無
* 実行コード：python explore\_siren\_ipynb\_eval5.py
* 出力データ（サンプルデータ）：

trained\_siren\_240613\_2204\_EGFP\_resized\_512x512\_no\_resize\_sine\_10000\_w30\_1024

* + 再構成された画像：1024\_model\_output.tiff

1. crop1.py

* 画像を切り取って保存する
  + (ex1) 上記で作成された1024×1024の画像を128ピクセルに切り取り保存する
  + (ex2) 1024×1024のオリジナル画像を128ピクセルに切り取り保存する
* 実行コード：python crop1.py
* 出力データ（サンプルデータ）：
  + (ex1) 240613\_2204\_EGFP\_resized\_512x512\_no\_resize\_1024\_128\_212\_668.tiff
  + (ex1) 240613\_2204\_EGFP\_resized\_512x512\_no\_resize\_1024\_128\_235\_831.tiff
  + (ex2) 240613\_2204\_EGFP\_128\_212\_668.tiff
  + (ex2) 240613\_2204\_EGFP\_128\_235\_831.tiff

1. psnr.py

* 再構成された画像とオリジナル画像のPSNR値を出力する
* (ex1)上記で作成された1024×1024の画像（1024\_model\_output.tiff）と、オリジナル1024×1024の画像（240613\_2204\_EGFP.tiff）を比較する
* (ex2) 1024×1024ではなく、128×128での比較を行う。再構成128×128の画像（240613\_2204\_EGFP\_resized\_512x512\_no\_resize\_1024\_128\_212\_668.tiff）と、オリジナル128×128の画像（240613\_2204\_EGFP\_128\_212\_668.tiff）を比較する
* 実行コード：python psnr.py

**～再構成の精度を評価（実験１：SIRENを通して出来た学習済みモデル（model\_final.pth）に、好きな座標数をいれて画像を再構成する。その画像をオリジナル画像と比較する。）（一部画像をSIRENに入力する）～**

1. resized\_no\_resize\_128.py

* 1024×1024の画像を縮小する（奇数行列だけの値を抽出する）方法
  + Opencvのresizeではない
* 1024ピクセル中の128ピクセルだけ抽出
* 実行コード：python resized\_no\_resize\_128.py
* 出力データ（サンプルデータ）：
  + 240613\_2204\_EGFP\_resized\_64x64\_no\_resize\_212\_668.tiff
  + 240613\_2204\_EGFP\_resized\_64x64\_no\_resize\_235\_831.tiff

1. train\_img.py・dataio.py・modules.py

* 上記の画像をSIRENに通す。
* 実行コード：

python experiment\_scripts/train\_img.py --model\_type=sine --experiment\_name=experiment\_240613\_2204\_EGFP\_resized\_64x64\_no\_resize\_212\_668\_sine\_10000\_w30

* + --experiment\_nameは好きなディレクトリ名を記入する
* 出力データ（サンプルデータ）：experiment\_240613\_2204\_EGFP\_resized\_64x64\_no\_resize\_212\_668\_sine\_10000\_w30
  + SIRENを通して出来た学習済みモデル（model\_final.pth）等

1. explore\_siren\_ipynb\_eval5.py

* 上記の学習済みモデル（model\_final.pth）を用いて、好きな座標数を入力し、画像を再構成する
* 勾配情報無
* 実行コード：python explore\_siren\_ipynb\_eval5.py
* 出力データ（サンプルデータ）：
  + trained\_siren\_240613\_2204\_EGFP\_resized\_64x64\_no\_resize\_212\_668\_sine\_10000\_w30\_128
  + 再構成された画像：128\_model\_output.tiff

1. psnr.py

* 再構成された画像とオリジナル画像のPSNR値を出力する
* (ex1)上記で作成された128×128の画像と、オリジナル128×128の画像を比較する
* 実行コード：python psnr.py

1. line\_evaluation\_vertical5.py（垂直：縦）

* 4つの画像比較
* 輝度値修正なし
* 全体画像の比較
* 4つの画像の赤線あり＋輝度値グラフを表示する

～学会にて使用したpythonファイル～

* eval4.py
  + 超解像のために作成
* line\_evaluation\_horizontal3.py（水平：横）作成
  + ２つの画像比較
  + 輝度値修正なし
  + クリックした部分の128ピクセル分だけを抽出する
  + ２つの画像の赤線あり＋輝度値グラフを表示する
* line\_evaluation\_horizontal4.py（水平：横）
  + ２つの画像比較
  + ２番目の画像だけ輝度値修正あり。
  + クリックした部分の128ピクセル分だけを抽出する
  + ２つの画像の赤線あり＋輝度値グラフを表示する
* line\_evaluation\_vertical2.py（垂直：縦）　⇐（沖縄学会・中間報告会の）実験２にて使用
  + ２つの画像比較
  + ２番目の画像だけ輝度値修正あり
  + クリックした部分の全体（縦軸）を抽出する
  + ２つの画像の赤線あり＋輝度値グラフを表示する
* line\_evaluation\_vertical3.py（垂直：縦）　⇐（沖縄学会・中間報告会の） 実験１にて使用
  + ３つの画像比較
  + 輝度値修正なし（コメントアウト部分を外すと輝度値修正もできる）
  + クリックした部分の128ピクセル分だけを抽出する
  + ３つの画像の赤線あり＋輝度値グラフを表示する
* line\_evaluation\_vertical4.py（垂直：縦）　⇐（沖縄学会・中間報告会の） 実験１にて使用
  + 4つの画像比較
  + 輝度値修正なし
  + [ クリックした部分 or 座標入力 ]の128ピクセル分だけを抽出する
  + 4つの画像の赤線あり＋輝度値グラフを表示する
* line\_evaluation\_vertical5.py（垂直：縦）
  + 4つの画像比較
  + 輝度値修正なし
  + 全体画像の比較
  + 4つの画像の赤線あり＋輝度値グラフを表示する