

Lab Notebook

Pupil Reconstruction

Daniel Carmona

Contents

1 Amplitude and Phase Reconstruction	2
2 PSF Reconstruction	23

1 Amplitude and Phase Reconstruction

16 February 2024

- Create new github branch: `AmpPhaseReconstructionRetraining`
 - Download the following files from Morgana:
 - `superK_slmcube_20230625_complsines-01sp_07`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file00`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file01`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file02`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file03`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file04`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file05`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file06`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file07`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file08`
 - `slmcube_20230625_complsines-01sp_07_PSFWFs_file09`
 - Data processing for Fully Convolutional NN training
 - One fast experiment for each file: very good results, around 0.05 validation mse
-

19 February 2024

- Remove normalization from amplitude and phase for the fully connected experiments
-

EXPERIMENT NewFC10000-Processed 2

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :
```

```
-Fully Connected  
-Input shape: 1320  
-Output shape: (2, 96, 96)  
-Hidden layers: [2000, 2000, 2000, 2000]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False , 0.2
```

```
*COMPILEATION HYPERPARAMETERS :
```

```
-Optimizer: ADAM lr=0.001 , beta_1=0.9 , beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

*** TRAINING HYPERPARAMETERS :**

- Epochs: 200
- Batch size: 64
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:*** RESULTS :**

- Train MSE: 0.035130929201841354
- Validation MSE: 0.0786573588848114

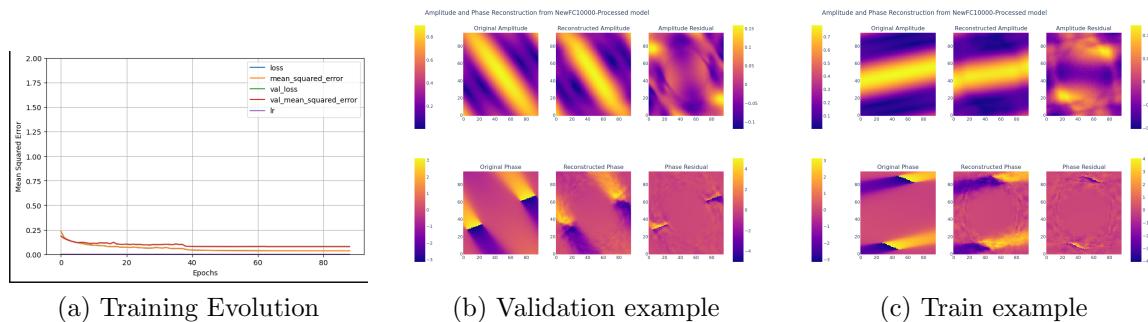


Figure 1: Results of training the model NewFC10000-Processed 2

EXPERIMENT NewFC30000-Processed-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS:

- Fully Connected
- Input shape: 1320
- Output shape: (2, 96, 96)
- Hidden layers: [2000, 2000, 2000, 2000]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

*COMPILE HYPERPARAMETERS:

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

*TRAINING HYPERPARAMETERS:

- Epochs: 200
- Batch size: 64
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:

*RESULTS :

-Train MSE: 0.0357743538916111

-Validation MSE: 0.057852283120155334

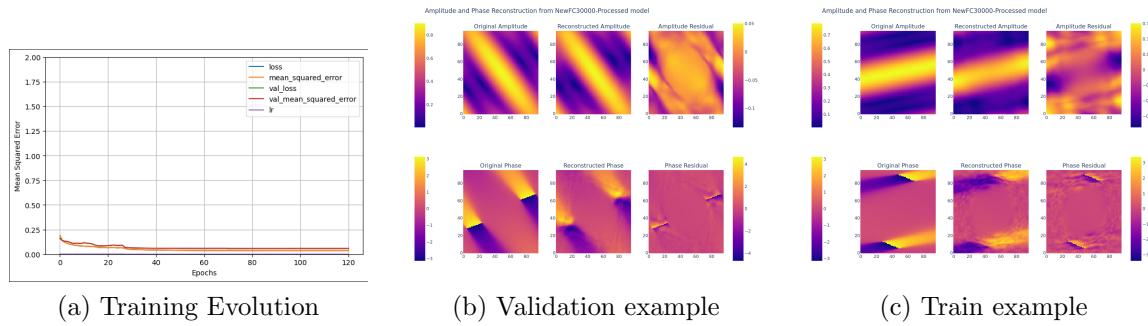


Figure 2: Results of training the model NewFC30000-Processed-1

EXPERIMENT NewFC80000-Processed-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS :

- Fully Connected
- Input shape: 1320
- Output shape: (2, 96, 96)
- Hidden layers: [2000, 2000, 2000, 2000]
- Regularizer: None
- Hidden Layers Activation: relu

```
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False , 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001 , beta_1=0.9 , beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 200  
-Batch size: 64  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.03301804140210152  
-Validation MSE: 0.04487497732043266
```

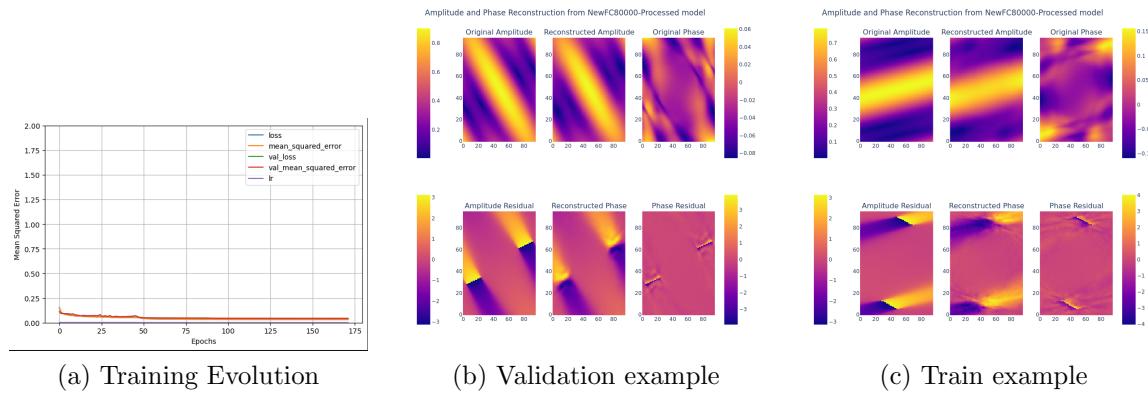


Figure 3: Results of training the model NewFC80000-Processed-1

- Normalize and split fluxes in train, validation and test files
- Stack amplitude and phase arrays and save in train, validation and test files

EXPERIMENT NewConv10000-1

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS :
-Convolutional
-Input shape: (55, 24, 1)
-Output shape: (96, 96, 2)
-Convolutional Layers: [128, 256, 512]
-Convolutonal Kernels: [(3, 3), (3, 3), (3, 3)]
```

```
-Fully Connected Hidden layers: [4096, 2048, 2048, 1024, 1024, 1024  
-Regularizer: None  
-Convolutional Activation: relu  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: True  
  
*COMPILE HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE  
  
* TRAINING HYPERPARAMETERS:  
-Epochs: 200  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 15 x0.1  
-Early Stop: MSE 50
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.031087761744856834  
-Validation MSE: 0.09376049041748047
```

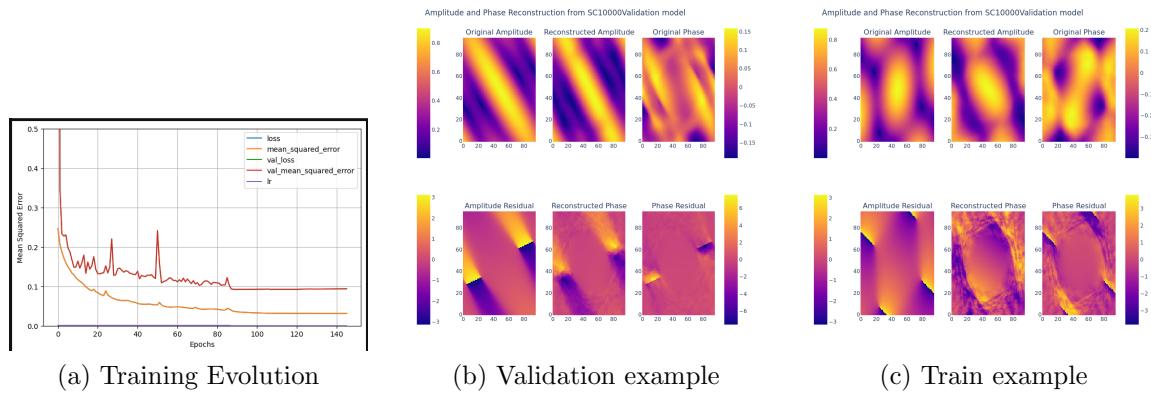


Figure 4: Results of training the model NewConv10000-1

20 February 2024

EXPERIMENT NewConv30000-1

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS :
  -Convolutional
  -Input shape: (55, 24, 1)
  -Output shape: (96, 96, 2)
  -Convolutional Layers: [128, 256, 512]
  -Convolutonal Kernels: [(3, 3), (3, 3), (3, 3)]
```

```
-Fully Connected Hidden layers: [4096, 2048, 2048, 1024, 1024, 1024  
-Regularizer: None  
-Convolutional Activation: relu  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: True  
  
*COMPILE HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE  
  
* TRAINING HYPERPARAMETERS:  
-Epochs: 200  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 15 x0.1  
-Early Stop: MSE 50
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.03819489851593971  
-Validation MSE: 0.06443119794130325
```



Figure 5: Results of training the model NewConv30000-1

EXPERIMENT NewConv30000-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS :

- Convolutional
- Input shape: (55, 24, 1)
- Output shape: (96, 96, 2)
- Convolutional Layers: [128, 256, 512]
- Convolutonal Kernels: [(3, 3), (3, 3), (3, 3)]
- Fully Connected Hidden layers: [4096, 2048, 2048, 1024, 1024, 1024]
- Regularizer: None
- Convolutional Activation: relu
- Hidden Layers Activation: relu

```
-Output Layer Activation: linear
```

```
-Batch Normalization: True
```

*COMPILE HYPERPARAMETERS:

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
```

```
-Loss Function: MSE
```

```
-Metric: MSE
```

* TRAINING HYPERPARAMETERS:

```
-Epochs: 200
```

```
-Batch size: 32
```

```
-Callbacks:
```

```
-ReduceLROnPlateau: MSE 15 x0.1
```

```
-Early Stop: MSE 50
```

VISUALIZATION:

*RESULTS:

```
-Train MSE: 0.030630357563495636
```

```
-Validation MSE: 0.08355200290679932
```



Figure 6: Results of training the model NewConv80000-1

- Normalize flux data and add padding for flux autoencoder
- Stack amplitude and phase and add padding for encoder+convolutional model

EXPERIMENT NewFluxAutoencoder10000-1

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS :
    -Autoencoder
    -Input shape: (56, 24, 1)
    -Convolutional Layers: [512, 128, 64, 8]
        (Inverse in the decoder)
    -Convolutonal Kernels: [(3, 3), (3, 3), (3, 3), (3, 3)]
```

```
(Inverse in the decoder)  
-Convolutional Activation: relu  
-Output Layer Activation: linear  
-Padding: same  
-Use Batch Normalization: True
```

***COMPIILATION HYPERPARAMETERS :**

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS :**

```
-Epochs: 75  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 8 x0.1  
-Early Stop: MSE 15
```

VISUALIZATION:

```
*RESULTS :  
-Train MSE: 0.0018673702143132687  
-Validation MSE: 0.005482238717377186
```



Figure 7: Results of training the model NewFluxAutoencoder10000-1

21 February 2024

EXPERIMENT NewFluxAutoencoder80000-1

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS :
- Autoencoder
- Input shape: (56, 24, 1)
- Convolutional Layers: [512, 128, 64, 8]
  (Inverse in the decoder)
```

```
-Convolutonal Kernels: [(3, 3), (3, 3), (3, 3), (3, 3)]  
    (Inverse in the decoder)  
  
-Convolutional Activation: relu  
  
-Output Layer Activation: linear  
  
-Padding: same  
  
-Use Batch Normalization: True  
  
*COMPILE HYPERPARAMETERS:  
  
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE  
  
*TRAINING HYPERPARAMETERS:  
  
-Epochs: 75  
-Batch size: 32  
-Callbacks:  
    -ReduceLROnPlateau: MSE 8 x0.1  
    -Early Stop: MSE 15
```

VISUALIZATION:

```
-Train MSE: 0.024667566642165184  
-Validation MSE: 0.0158506091684103
```



Figure 8: Results of training the model NewFluxAutoencoder10000-1

EXPERIMENT NewEncConv10000-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS :

- Encoder + Convolutional
- Convolutional Layers: [1024, 512, 256, 256]
- Convolutonal Kernels: [(3, 3), (3, 3), (3, 3), (3, 3)]
- Convolutional Activation: relu
- Output Layer Activation: linear

*COMPILEATION HYPERPARAMETERS :

- Optimizer: ADAM lr=0.0001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE

```

-Metric: MSE

* TRAINING HYPERPARAMETERS:

-Epochs: 100
-Batch size: 32
-Callbacks:
-ReduceLROnPlateau: MSE 8 x0.1
-Early Stop: MSE 15

```

VISUALIZATION:

```

*RESULTS:

-Train MSE: 0.011328332126140594
-Validation MSE: 0.06024651601910591

```

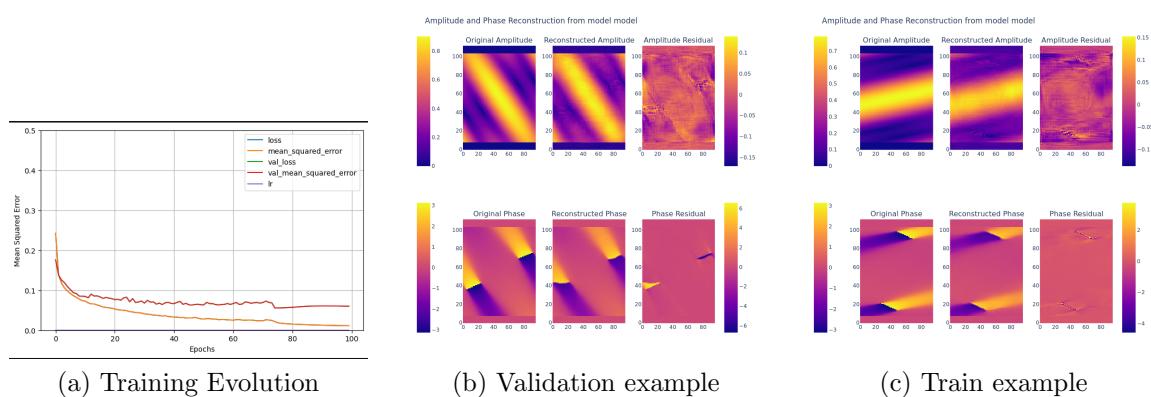


Figure 9: Results of training the model NewEncConv10000-1

EXPERIMENT NewEncConv30000-1

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS:  
  -Encoder + Convolutional  
  -Convolutional Layers: [1024, 512, 256, 256]  
  -Convolutonal Kernels: [(3, 3), (3, 3), (3, 3), (3, 3)]  
  -Convolutional Activation: relu  
  -Output Layer Activation: linear  
  
* COMPILEATION HYPERPARAMETERS:  
  -Optimizer: ADAM lr=0.0001, beta_1=0.9, beta_2=0.999  
  -Loss Function: MSE  
  -Metric: MSE  
  
* TRAINING HYPERPARAMETERS:  
  -Epochs: 100  
  -Batch size: 32  
  -Callbacks:  
    -ReduceLROnPlateau: MSE 8 x0.1  
    -Early Stop: MSE 15
```

VISUALIZATION:

```
* RESULTS:  
  -Train MSE: 0.016221819445490837
```



Figure 10: Results of training the model NewEncConv30000-1

22 February 2024

EXPERIMENT NewEncConv80000-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :
-Encoder + Convolutional
-Convolutional Layers: [1024, 512, 256, 256]
```

```
-Convolutonal Kernels: [(3, 3), (3, 3), (3, 3), (3, 3)]  
-Convolutonal Activation: relu  
-Output Layer Activation: linear  
  
*COMPILEATION HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.0001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE  
  
* TRAINING HYPERPARAMETERS:  
-Epochs: 100  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 8 x0.1  
-Early Stop: MSE 15
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.0163  
-Validation MSE: 0.0369
```



Figure 11: Results of training the model NewEncConv80000-1

2 PSF Reconstruction

20 February 2024

- Create custom dynamic dataloader for PL output flux and PSF complex field for PSF reconstruction

26 February 2024

- Create a dataloader to dynamically load flux and psf
 - Normalize electric fields, real and imaginary parts independently
 - Normalize PL output fluxes
 - Train a fully connected nn with horrible results, train mse keeps stable at 1, will have to look at the data
-

27 February 2024

- Create plotting functions to show amplitude, phase and intensity from an electric field
 - Normalize electric field, this time dividing the complex number matrix by a constant (50.000 in this case)
 - Redo output flux calculation with the normalized electric fields through the transfer matrix of the 19 fibre PL
-

28 February 2024

EXPERIMENT PSF-FC-Reconstructor-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS :

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [2000, 2000, 2000, 2000]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

*COMPILEATION HYPERPARAMETERS :

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

*TRAINING HYPERPARAMETERS :

- Epochs: 200

```
-Batch size: 64
-Callbacks:
-ReduceLROnPlateau: MSE 10 x0.1
-Early Stop: MSE 25
```

VISUALIZATION:

```
-Train MSE: 0.03701553866267204
-Validation MSE: 0.03701810911297798
```

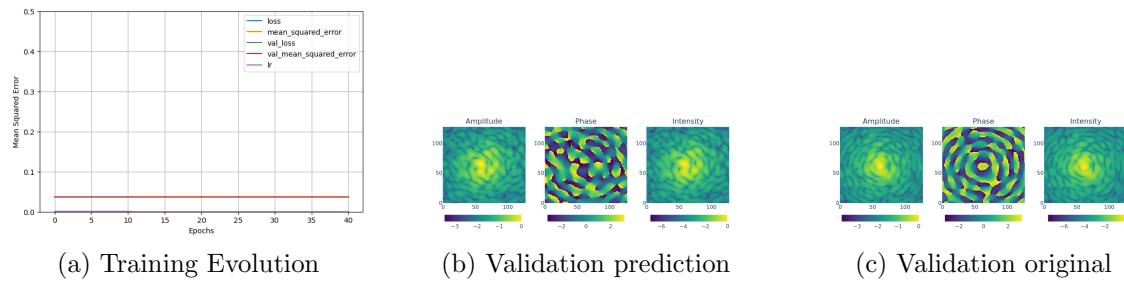


Figure 12: Results of training the model PSF-FC-Reconstructor-1

29 February 2024

- Create a function to plot amplitude and phase predictions and residuals from a complex field
- Create evaluation notebook
- Since in yesterdays experiment the model does not learn at all, I design two different experiments, to check if there is a problem with the network:

1. A model that sums the output flux of the PL
 2. A model that sums the scalars of the complex numbers array containing the electric fields at the pupil.
-

EXPERIMENT Flux-Sum-1

This model performs a sum of all the output fluxes from the PL

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:  
-Fully Connected  
-Input shape: 19  
-Output shape: 1  
-Hidden layers: [100, 100]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

```
*COMPILEATION HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.0001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

```
* TRAINING HYPERPARAMETERS :
- Epochs: 200
- Batch size: 64
- Callbacks:
  - ReduceLROnPlateau: MSE 10 x0.1
  - Early Stop: MSE 25
```

VISUALIZATION:

```
* RESULTS:
- Train MSE: 0.00018273142632097006
- Validation MSE: 0.00018727070710156113
```



Figure 13: Results of training the model Flux-Sum-1

EXPERIMENT ElectricField-Sum-1

This model performs a sum of all the output fluxes from the PL

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:  
-Fully Connected  
-Input shape: 32768  
-Output shape: 1  
-Hidden layers: [100, 100, 100]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

```
*COMPILE HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.0001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

```
*TRAINING HYPERPARAMETERS:  
-Epochs: 200  
-Batch size: 64  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

***RESULTS :**

-Train MSE: 0.050156451761722565

-Validation MSE: 0.8189939856529236



Figure 14: Results of training the model ElectricField-Sum-1

01 March 2024

- Today I decide to train a model with just one datapoint and see if is able to learn the PSF from one flux and one electric field

EXPERIMENT TestWith1DataPoint-1

HYPERPARAMETERS:

***ARCHITECTURE HYPERPARAMETERS :**

```
-Fully Connected  
-Input shape: 19  
-Output shape: 32768  
-Hidden layers: [1000, 1000, 1000, 1000]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.01, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 1000  
-Batch size: 64  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

```
-Train MSE: 0.03701553866267204  
-Validation MSE: 0.03701810911297798
```

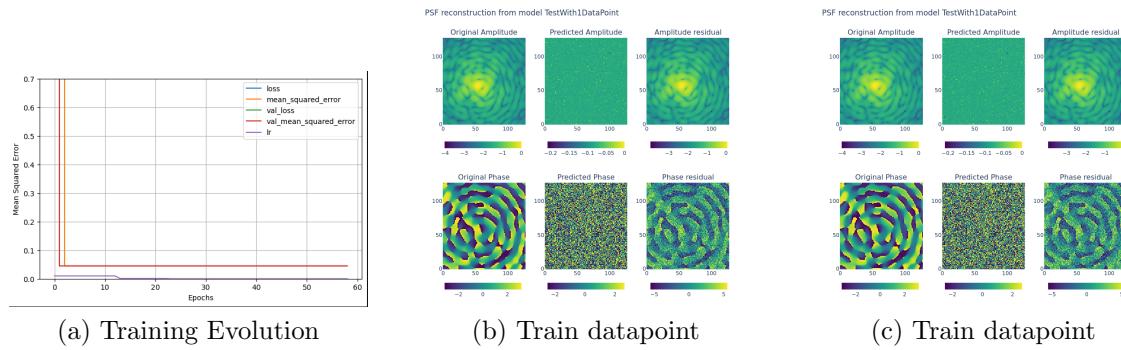


Figure 15: Results of training the model TestWith1DataPoint-1

- Something is wrong, going try with a bigger NN, don't think this will change anything but I have to try just in case.

EXPERIMENT TestWith1DataPoint-2

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS :
  -Fully Connected
  -Input shape: 19
  -Output shape: 32768
  -Hidden layers: [128, 128, 128, 128, 256,
                  256, 512, 2000, 4000]
  -Regularizer: None
```

```
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.01, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 1000  
-Batch size: 64  
-Callbacks:  
    -ReduceLROnPlateau: MSE 10 x0.1  
    -Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:  
    -Train MSE: 0.04261719062924385  
    -Validation MSE: 0.04261719062924385
```



Figure 16: Results of training the model TestWith1DataPoint-2

04 March 2024

- After seeing that with 1 datapoint the model does not learn I decide to make some simpler tests, I will train a model that from a 19 element array predicts images with 2 channels and see if the input is too small for the model to predict anything. Three experiments will be performed:
 - From a 19 elements array to a 2x128x128 image
 - From a 19 elements array to a 2x64x64 image
 - From a 19 elements array to a 2x32x32 image

EXPERIMENT FromArrayToImage128-1

HYPERPARAMETERS:

***ARCHITECTURE HYPERPARAMETERS:**

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [128, 128, 128, 128, 256, 256, 512, 2000, 4000]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False , 0.2

***COMPILE HYPERPARAMETERS:**

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

***TRAINING HYPERPARAMETERS:**

- Epochs: 10000
- Batch size: 1
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:***RESULTS:**

```
-Train MSE: 0.00010900569031946361
```

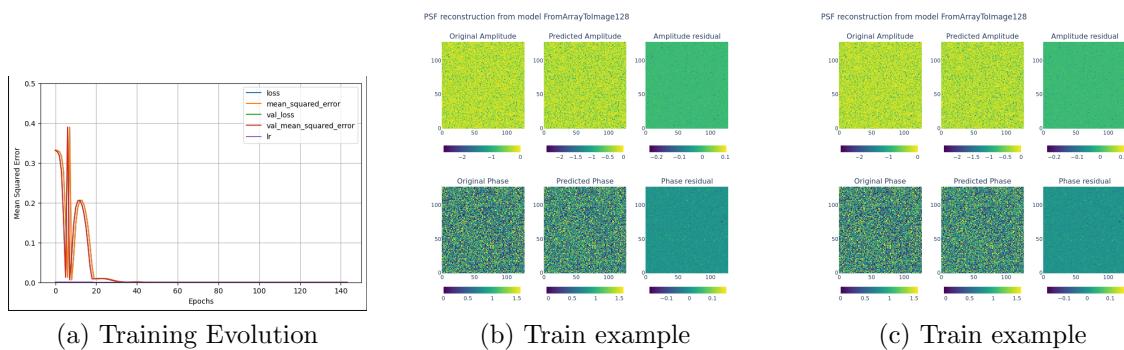


Figure 17: Results of training the model FromArrayToImage128-1

- From the previous experiment it is clear that a 19 element array as input is enough to reconstruct a random image of 2x128x128 resolution, what is happening for the psf reconstruction?.

I will perform the other 2 stated experiments because it is quite fast.

EXPERIMENT FromArrayToImage64-1

HYPERPARAMETERS:

```
* ARCHITECTURE HYPERPARAMETERS :
-Fully Connected
```

```
-Input shape: 19  
-Output shape: 8192  
-Hidden layers: [128, 128, 128, 128, 256, 256, 512, 2000, 4000]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILATION HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 10000  
-Batch size: 1  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:***RESULTS:**

```
-Train MSE: 0.00012627203250303864
```



Figure 18: Results of training the model FromArrayToImage64-1

EXPERIMENT FromArrayToImage32-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:
-Fully Connected
-Input shape: 19
-Output shape: 2048
-Hidden layers: [128, 128, 128, 128, 256, 256, 512, 2000, 4000]
-Regularizer: None
-Hidden Layers Activation: relu
-Output Layer Activation: linear
-Batch Normalization: False
-Dropout: False , 0.2
```

***COMPILE HYPERPARAMETERS :**

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

***TRAINING HYPERPARAMETERS :**

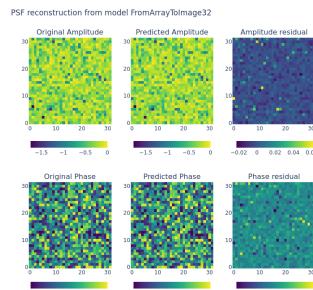
- Epochs: 10000
- Batch size: 1
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:***RESULTS :**

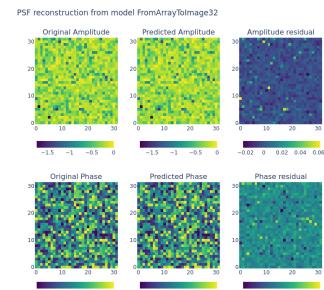
- Train MSE: 5.556200267164968e-05



(a) Training Evolution



(b) Train example



(c) Train example

Figure 19: Results of training the model FromArrayToImage32-1

- The results are satisfactory, I am going to try the exact same configuration for one data point of the PSF dataset and see what happens
-

EXPERIMENT TestWith1DataPoint-3

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:  
-Fully Connected  
-Input shape: 19  
-Output shape: 32768  
-Hidden layers: [128, 128, 128, 128, 256, 256, 512, 2000, 4000]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

```
*COMPILATION HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

```
* TRAINING HYPERPARAMETERS :
-Epochs: 10000
-Batch size: 1
-Callbacks:
-ReduceLROnPlateau: MSE 10 x0.1
-Early Stop: MSE 25
```

VISUALIZATION:

```
* RESULTS :
-Train MSE: 1.2409615010255948e-05
```



Figure 20: Results of training the model TestWith1DataPoint-3

- The result has improved incredibly compared to the TestWith1DataPoint-1 and TestWith1DataPoint-2. The only difference I can see are the learning rates and the batch size. I will perform a series of experiments varying batch size: 100 datapoints with 64, 32 and 16 batch size

EXPERIMENT PSFReconstructor-100-64-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:
```

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [128, 128, 128, 128, 256,
256, 512, 2000, 4000]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

```
*COMPILEATION HYPERPARAMETERS:
```

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

```
*TRAINING HYPERPARAMETERS:
```

- Epochs: 10000
- Batch size: 64

```
-Callbacks:
-ReduceLROnPlateau: MSE 10 x0.1
-Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:
-Train MSE: 1.7399888747604564e-05
```



Figure 21: Results of training the model PSFReconstructor-100-64-1

05 March 2024

EXPERIMENT PSFReconstructor-100-32-1

HYPERPARAMETERS:

***ARCHITECTURE HYPERPARAMETERS:**

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [128, 128, 128, 128, 256, 256, 512, 2000, 4000]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

***COMPILE HYPERPARAMETERS:**

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

***TRAINING HYPERPARAMETERS:**

- Epochs: 10000
- Batch size: 32
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:

*RESULTS :

-Train MSE: 0.00033666446688584983

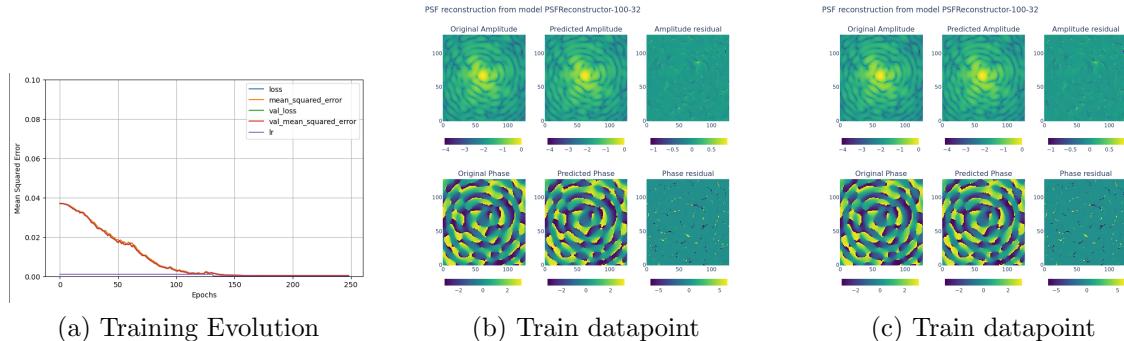


Figure 22: Results of training the model PSFReconstructor-100-32-1

EXPERIMENT PSFReconstructor-100-16-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS :

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [128, 128, 128, 128, 256, 256, 512, 2000, 4000]
- Regularizer: None

```
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 10000  
-Batch size: 16  
-Callbacks:  
    -ReduceLROnPlateau: MSE 10 x0.1  
    -Early Stop: MSE 25
```

VISUALIZATION:***RESULTS:**

```
-Train MSE: 0.0003098844608757645
```



Figure 23: Results of training the model PSFReconstructor-100-16-1

- Apart from the mse, no significant difference can be appreciated for different batch sizes using 100 datapoints.
- The goal for today is to study the correlation between the L2 norm of the pupil plane electric field and the L2 norm of the LP modes complex coefficients.
- First compute the L2 norm of the LP modes complex coefficients from the validation psf electric fields.
- Done!, here it is what it looks like

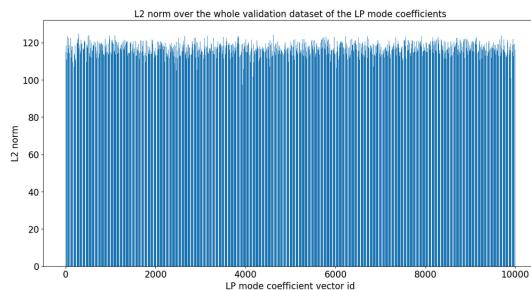


Figure 24: L2 norms for the overlap integral mode coefficients of the validation PSFs

- Let's do the same for the electric fields of the PSF
- Done!, here are the results:

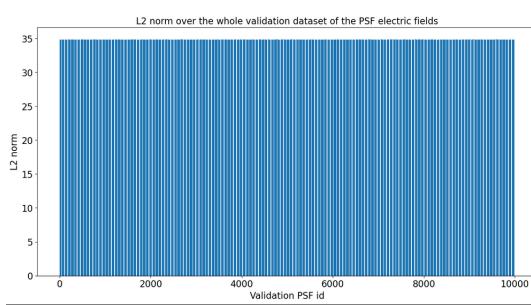


Figure 25: L2 norms of the validation PSFs electric fields

- After doing a correlation analysis the result is the following:

```
Pearson correlation coefficient: 0.1451513895198259
```

Which means that there is no correlation

06 March 2024

- Today I will be looking at the code from the `lantern_fiber_utils.py` file looking for the place where the LP mode coefficients are normalized

- Found it, in the function `make_fiber_modes()` used to compute de coefficients, there is a parameter called `normtosum` that is set to True by default, it looks as it normalizes the coefficients after the overlap integral is computed.
 - Set `normtosum` to False, and compute 10000 output fluxes.
-

07 March 2024

- I will test a model with the new generated unnormalized output fluxes
-

EXPERIMENT UnnormalizedCoefficients-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :  
    -Fully Connected  
    -Input shape: 19  
    -Output shape: 32768  
    -Hidden layers: [256, 256, 256]  
    -Regularizer: None  
    -Hidden Layers Activation: relu  
    -Output Layer Activation: linear  
    -Batch Normalization: False
```

-Dropout: False , 0.2

*COMPILEATION HYPERPARAMETERS :

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

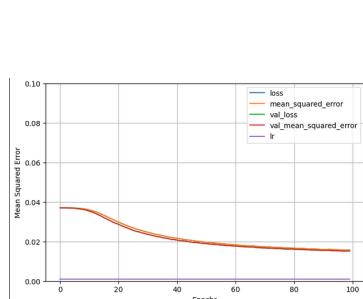
*TRAINING HYPERPARAMETERS :

- Epochs: 100
- Batch size: 64
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

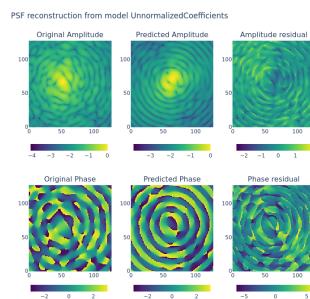
VISUALIZATION:

*RESULTS :

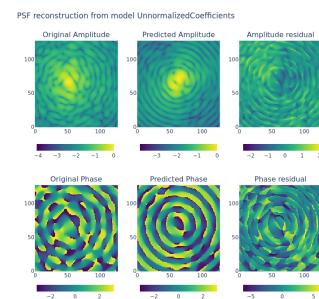
-Train MSE: 0.01565057598054409



(a) Training Evolution



(b) Train datapoint



(c) Train datapoint

Figure 26: Results of training the model UnnormalizedCoefficientsS-1

-
- Looks like the problem with the data was in the normalization of the coefficients, I will reprocess the output fluxes again and then test with a bigger dataset.
 - While, the processing is running I create two new functions to plot, amplitude-phase-intensity and output flux of the PL.
-

08 March 2024

- FUCK, When processing the data I forgot to disable the `normtosum` variable. Have to repeat the process again
-

11 March 2024

- Finally got the output fluxes from the unnormalized mode coefficients
-

EXPERIMENT PSFReconstructorFC10000-1

HYPERPARAMETERS:

***ARCHITECTURE HYPERPARAMETERS:**

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [256, 256, 256]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

***COMPIILATION HYPERPARAMETERS:**

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

***TRAINING HYPERPARAMETERS:**

- Epochs: 100
- Batch size: 64
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:***RESULTS:**

```
-Train MSE: 0.017938978970050812
-Validation MSE: 0.055591899901628494
```

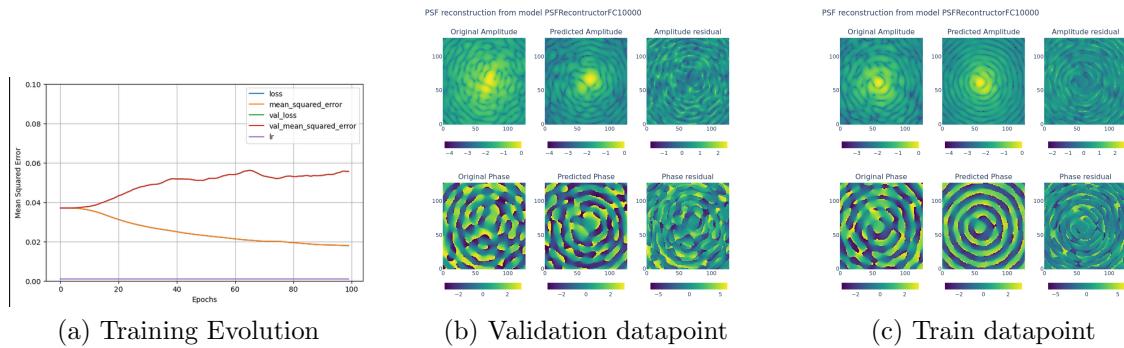


Figure 27: Results of training the model PSFReconstructorFC10000-1

EXPERIMENT PSFReconstructorFC30000-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:
-Fully Connected
-Input shape: 19
-Output shape: 32768
-Hidden layers: [256, 256, 256]
-Regularizer: None
-Hidden Layers Activation: relu
-Output Layer Activation: linear
```

```
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 100  
-Batch size: 64  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.03696389123797417  
-Validation MSE: 0.03704820200800896
```

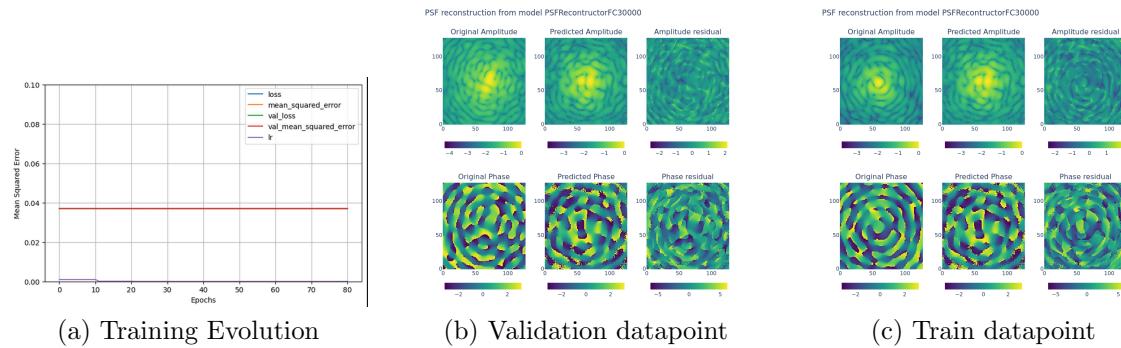


Figure 28: Results of training the model PSFReconstructorFC30000-1

- Again, flatline, and same output for any input. I will try with all the train dataset and then start using different batch sizes, see how this works

EXPERIMENT PSFReconstructorFC70000-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :
-Fully Connected
-Input shape: 19
-Output shape: 32768
-Hidden layers: [256, 256, 256]
-Regularizer: None
-Hidden Layers Activation: relu
```

```
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False , 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001 , beta_1=0.9 , beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 100  
-Batch size: 64  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.03700976073741913  
-Validation MSE: 0.0370212197303772
```

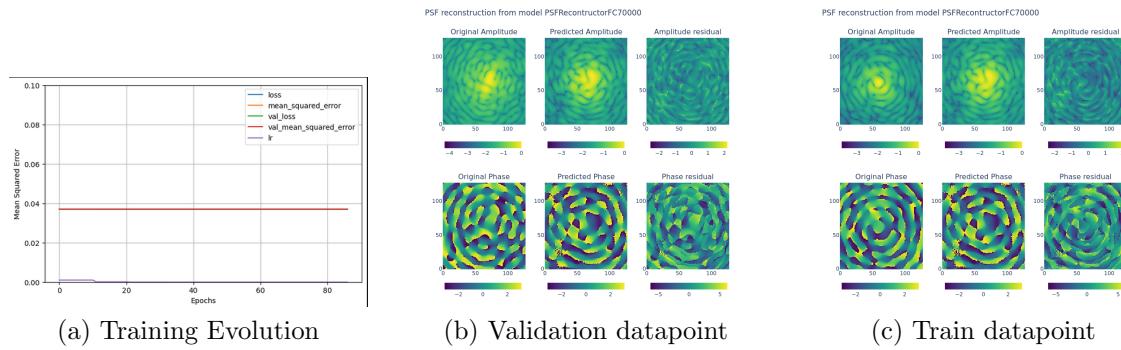


Figure 29: Results of training the model PSFReconstructorFC70000-1

- Batch size for these experiments is 64, maybe it is too big , let's try with 32

EXPERIMENT PSFReconstructorFC10000-32-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :
-Fully Connected
-Input shape: 19
-Output shape: 32768
-Hidden layers: [256, 256, 256]
-Regularizer: None
-Hidden Layers Activation: relu
-Output Layer Activation: linear
```

```
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 100  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.03432031720876694  
-Validation MSE: 0.038697339594364166
```

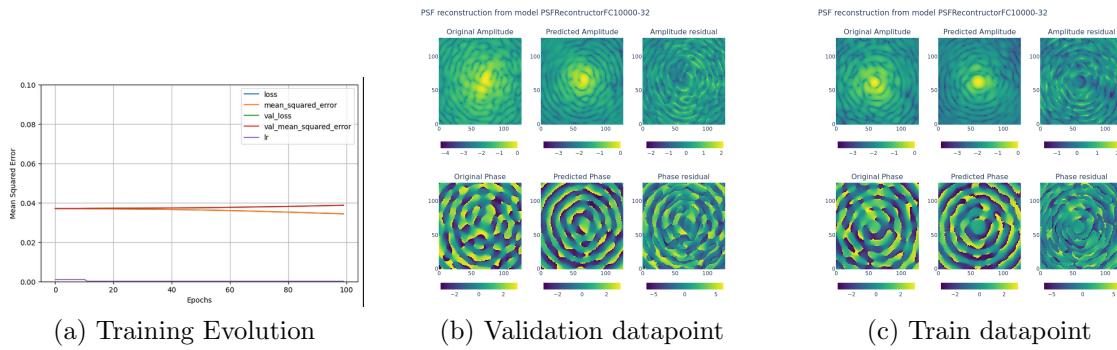


Figure 30: Results of training the model PSFReconstructorFC10000-32-1

- Interesting, slower convergence but less overfitting

EXPERIMENT PSFReconstructorFC30000-32-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :
-Fully Connected
-Input shape: 19
-Output shape: 32768
-Hidden layers: [256, 256, 256]
-Regularizer: None
-Hidden Layers Activation: relu
-Output Layer Activation: linear
```

```
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS:**

```
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE
```

***TRAINING HYPERPARAMETERS:**

```
-Epochs: 100  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 10 x0.1  
-Early Stop: MSE 25
```

VISUALIZATION:

```
*RESULTS:  
-Train MSE: 0.03701305016875267  
-Validation MSE: 0.03701600059866905
```

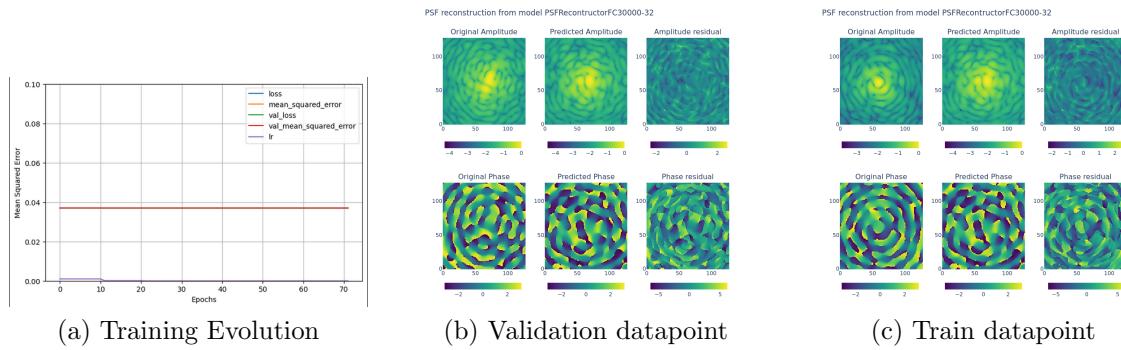


Figure 31: Results of training the model PSFReconstructorFC30000-32-1

EXPERIMENT PSFReconstructorFC70000-32-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:
-Fully Connected
-Input shape: 19
-Output shape: 32768
-Hidden layers: [256, 256, 256]
-Regularizer: None
-Hidden Layers Activation: relu
-Output Layer Activation: linear
-Batch Normalization: False
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS :**

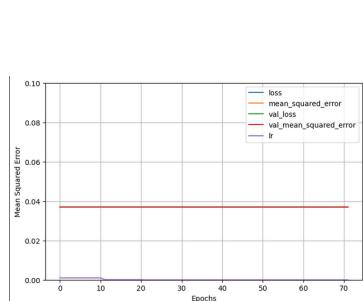
- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

***TRAINING HYPERPARAMETERS :**

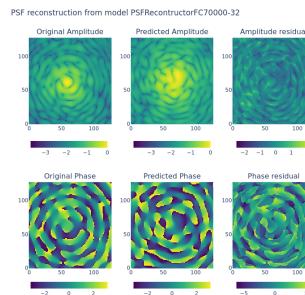
- Epochs: 100
- Batch size: 32
- Callbacks:
 - ReduceLROnPlateau: MSE 10 x0.1
 - Early Stop: MSE 25

VISUALIZATION:***RESULTS :**

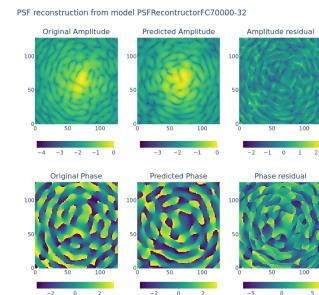
- Train MSE: 0.03701675683259964
- Validation MSE: 0.037016723304986954



(a) Training Evolution



(b) Validation datapoint



(c) Train datapoint

Figure 32: Results of training the model PSFReconstructorFC70000-32-1

- So no good results, what I saw is that the lower the batch size, the longer it takes to converge. Options now are:
 - Bigger NN
 - Bigger Batch size
 - Longer training

I will start with 10000 datapoints and a bigger NN

EXPERIMENT PSFReconstructorBigFC10000-32-1

HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS :  
-Fully Connected  
-Input shape: 19  
-Output shape: 32768  
-Hidden layers: [256, 256, 256, 256, 256, 256]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2
```

***COMPILE HYPERPARAMETERS :**

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

***TRAINING HYPERPARAMETERS :**

- Epochs: 100
- Batch size: 32
- Callbacks:
 - ReduceLROnPlateau: MSE 20 x0.1
 - Early Stop: MSE 50

VISUALIZATION:***RESULTS :**

- Train MSE: 0.012548761442303658
- Validation MSE: 0.0527174137532711

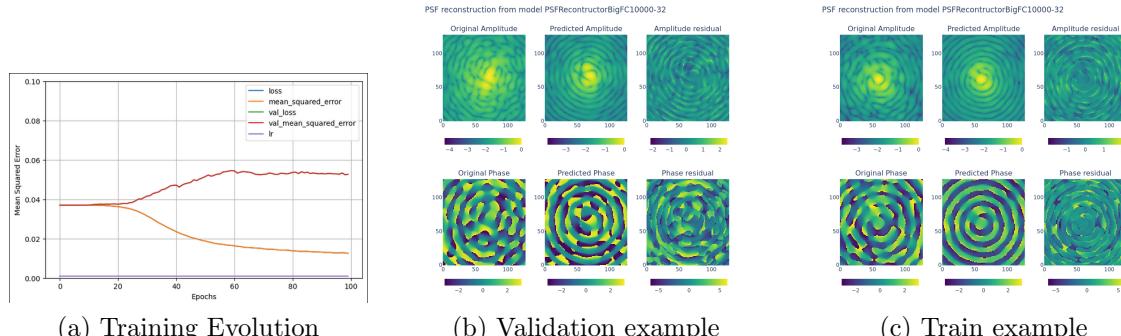


Figure 33: Results of training the model psf-PSFReconstructorBigFC10000-32-1

EXPERIMENT PSFReconstructorBigFC30000-32-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS:

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [256, 256, 256, 256, 256, 256]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

*COMPILEATION HYPERPARAMETERS:

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

*TRAINING HYPERPARAMETERS:

- Epochs: 200
- Batch size: 32

-Callbacks:

- ReduceLROnPlateau: MSE 20 x 0.1
- Early Stop: MSE 50

VISUALIZATION:

*RESULTS:

- Train MSE: 0.025507930666208267
- Validation MSE: 0.04988955706357956

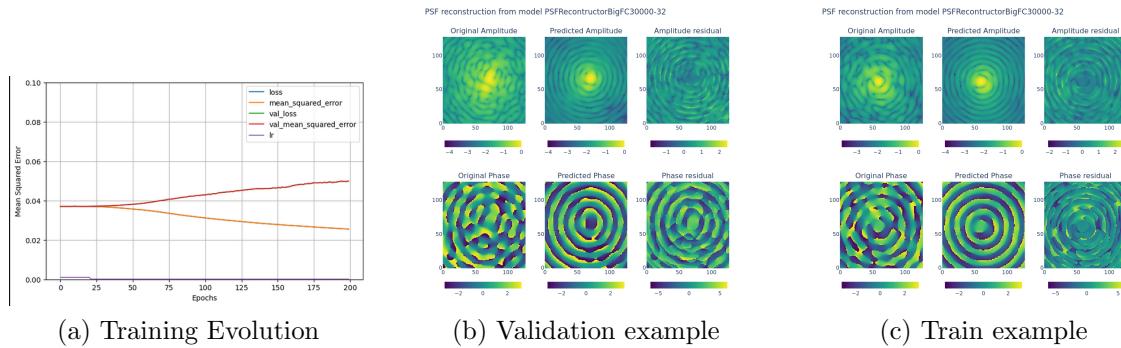


Figure 34: Results of training the model PSFReconstructorBigFC30000-32-1

- With respect to the last two experiments, a bigger neural network seems to work better, with more datapoints the learning is delayed a bit, and the overfitting slightly improves, let's check with 70000 datapoints.

12 March 2024

EXPERIMENT PSFReconstructorBigFC70000-32-1

HYPERPARAMETERS:

*ARCHITECTURE HYPERPARAMETERS :

- Fully Connected
- Input shape: 19
- Output shape: 32768
- Hidden layers: [256, 256, 256, 256, 256, 256]
- Regularizer: None
- Hidden Layers Activation: relu
- Output Layer Activation: linear
- Batch Normalization: False
- Dropout: False, 0.2

*COMPILEATION HYPERPARAMETERS :

- Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999
- Loss Function: MSE
- Metric: MSE

*TRAINING HYPERPARAMETERS :

- Epochs: 200

```
-Batch size: 32
-Callbacks:
-ReduceLROnPlateau: MSE 20 x0.1
-Early Stop: MSE 50
```

VISUALIZATION:

*RESULTS :

- Train MSE: 0.025293584913015366
- Validation MSE: 0.04955973103642464

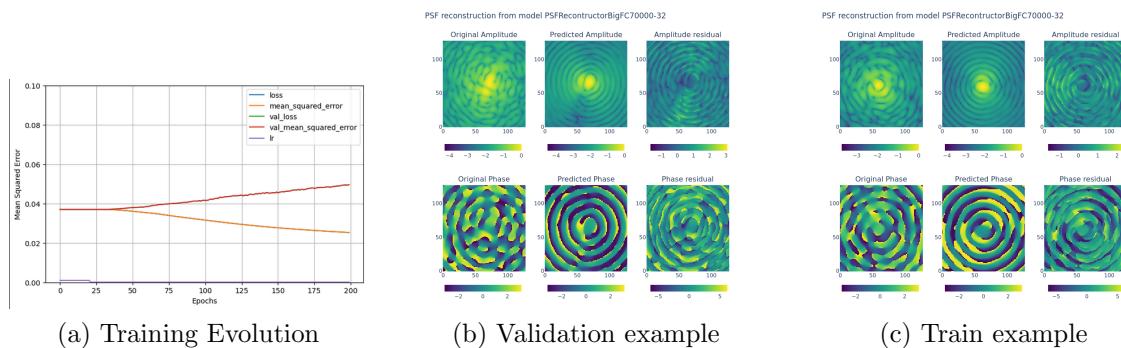


Figure 35: Results of training the model PSFReconstructorBigFC70000-32-1

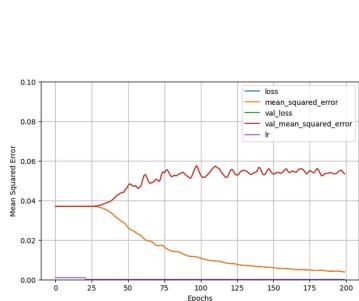
- There is barely any difference in training with 30000 and 70000 datapoints. I think I will make a final set of experiments with an even bigger neural network before addressing the overfitting.

EXPERIMENT PSFReconstructorSuperBigFC10000-1

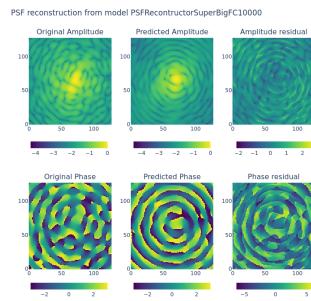
HYPERPARAMETERS:

```
*ARCHITECTURE HYPERPARAMETERS:  
-Fully Connected  
-Input shape: 19  
-Output shape: 32768  
-Hidden layers: [1024, 1024, 1024, 1024, 1024, 1024]  
-Regularizer: None  
-Hidden Layers Activation: relu  
-Output Layer Activation: linear  
-Batch Normalization: False  
-Dropout: False, 0.2  
  
*COMPILE HYPERPARAMETERS:  
-Optimizer: ADAM lr=0.001, beta_1=0.9, beta_2=0.999  
-Loss Function: MSE  
-Metric: MSE  
  
*TRAINING HYPERPARAMETERS:  
-Epochs: 200  
-Batch size: 32  
-Callbacks:  
-ReduceLROnPlateau: MSE 20 x0.1  
-Early Stop: MSE 50
```

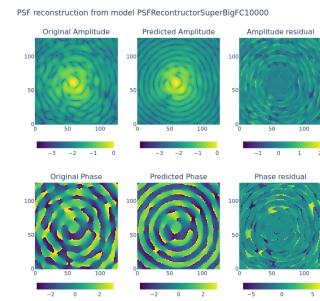
VISUALIZATION:

RESULTS :*-Train MSE: 0.00384897249750793****-Validation MSE: 0.05339088663458824**

(a) Training Evolution



(b) Validation example



(c) Train example

Figure 36: Results of training the model PSFReconstructorSuperBigFC10000-1