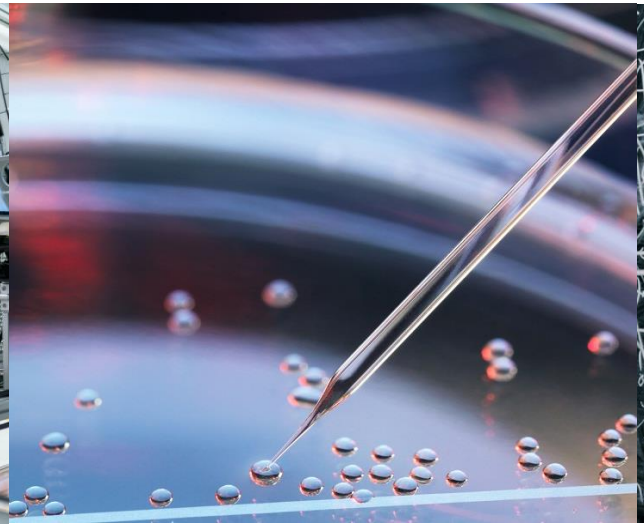
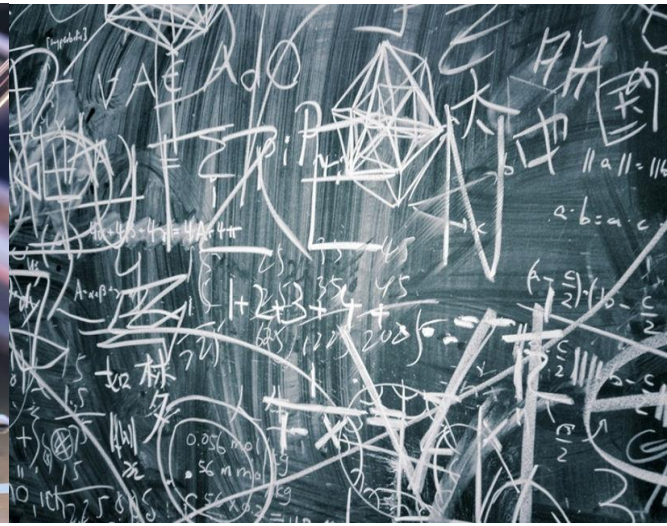
CDCS
CENTER FOR DATA AND COMPUTING
IN NATURAL SCIENCES

What's new with galaxies

Lennart Rustige

CDCS JC, 28/06/2022

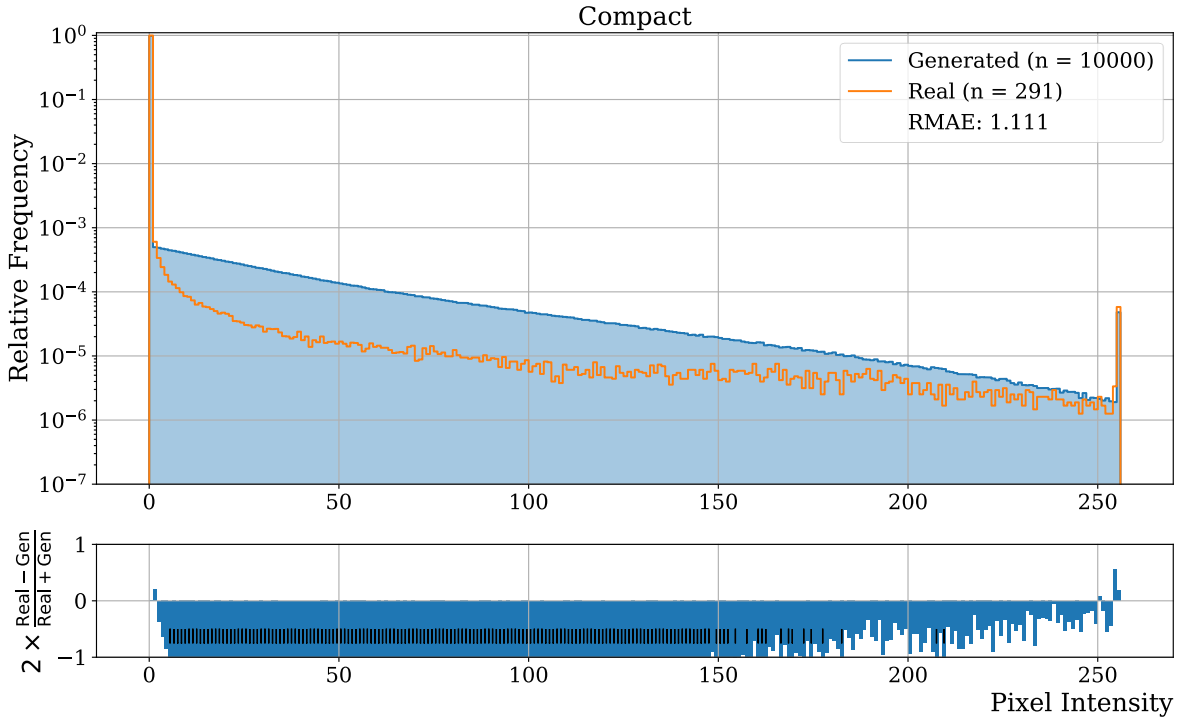
Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

DESY.

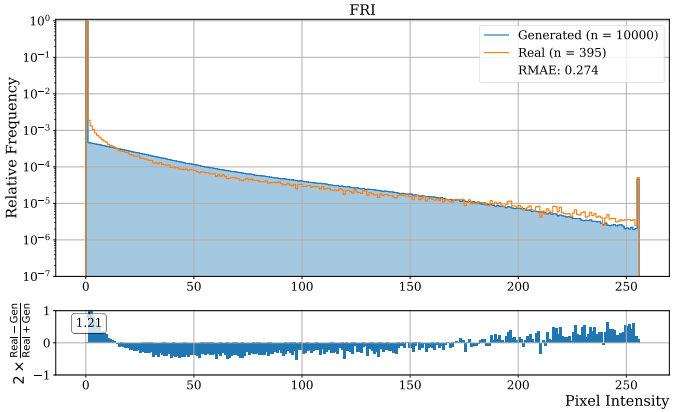
TUHH
Technische Universität Hamburg-Harburg

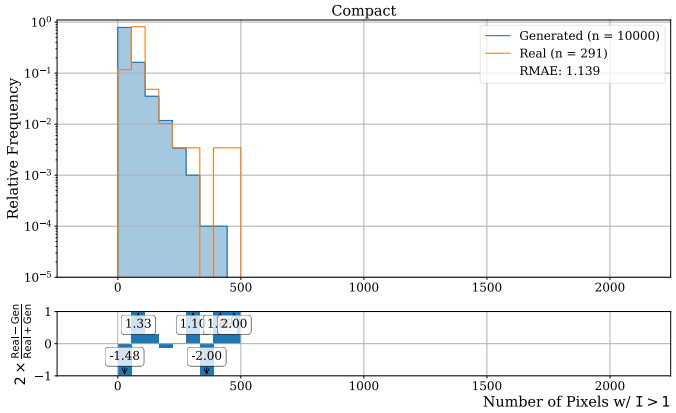
Generation

- We weren't great at generating all classes..

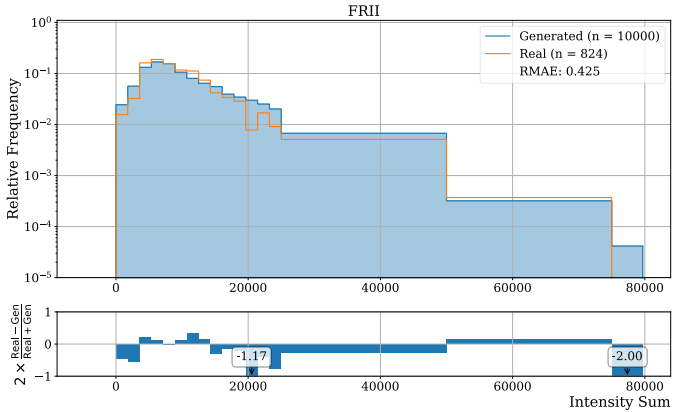


How did it get there?





- Best generator checkpoint chosen on these three histogram types
- Previously:
  - Compute RMAE for each histogram (4 classes x 3 histograms) and average over all of them
  - Take four best performing iterations and choose one randomly for generation
- Now:
  - Compute RMAE for each histogram, average only over the three histogram types
  - Take best performing checkpoint per class and generate the corresponding class only with this checkpoint



Average of

level_1	Bent	Compact	FRI	FRII
iteration				
28750	0.774832	1.452667	0.700856	0.606045
36250	0.271016	1.512272	0.402949	0.460576
37750	0.298019	1.563613	0.398343	0.402971
39750	0.270917	1.560187	0.393322	0.404278

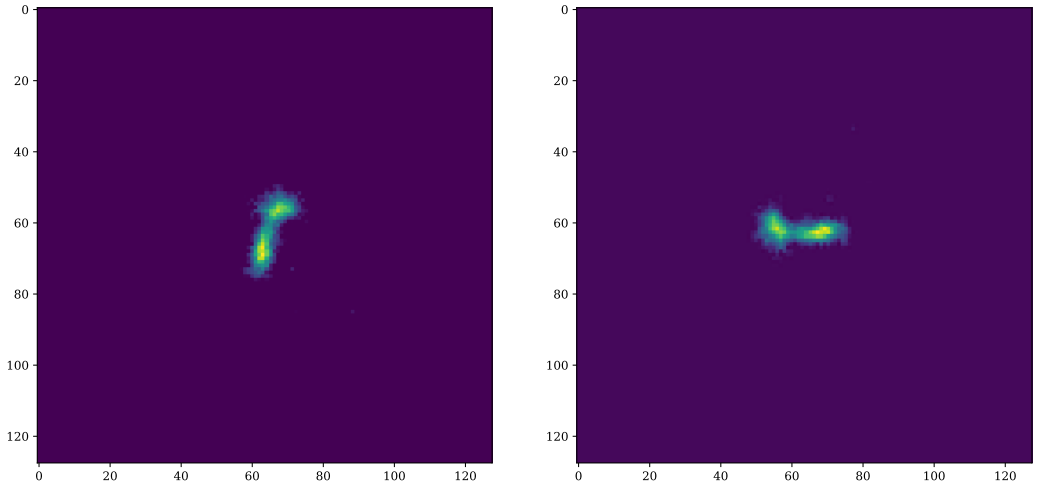
	old	new	new/old
Bent	0.403696	0.271	0.671297
Compact	1.522185	0.874	0.574175
FRI	0.473868	0.328	0.692177
FRII	0.468467	0.352	0.751386

Old setup using four ‘overall best’ gen iterations

- Image quality clearly improved
- Impact on classifier still under investigation

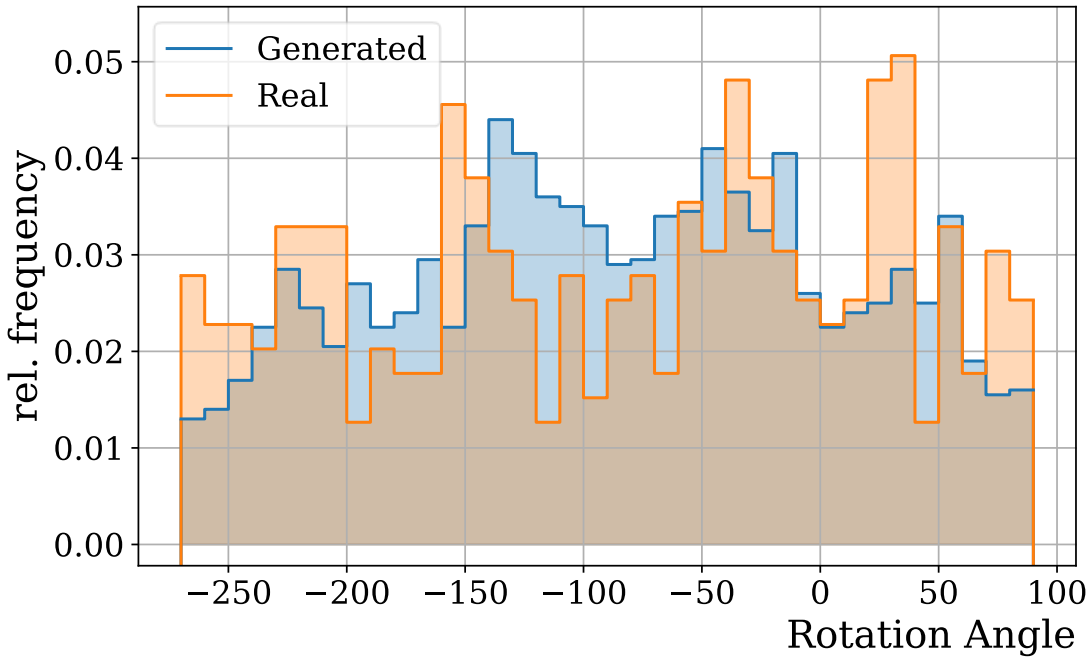
- Try to find best matching pair of real and generated images to show maximum realness achievable by the generator.
- In order to use a pixelwise metric, images need to be aligned
- Use PCA and rotate the first component

Generated

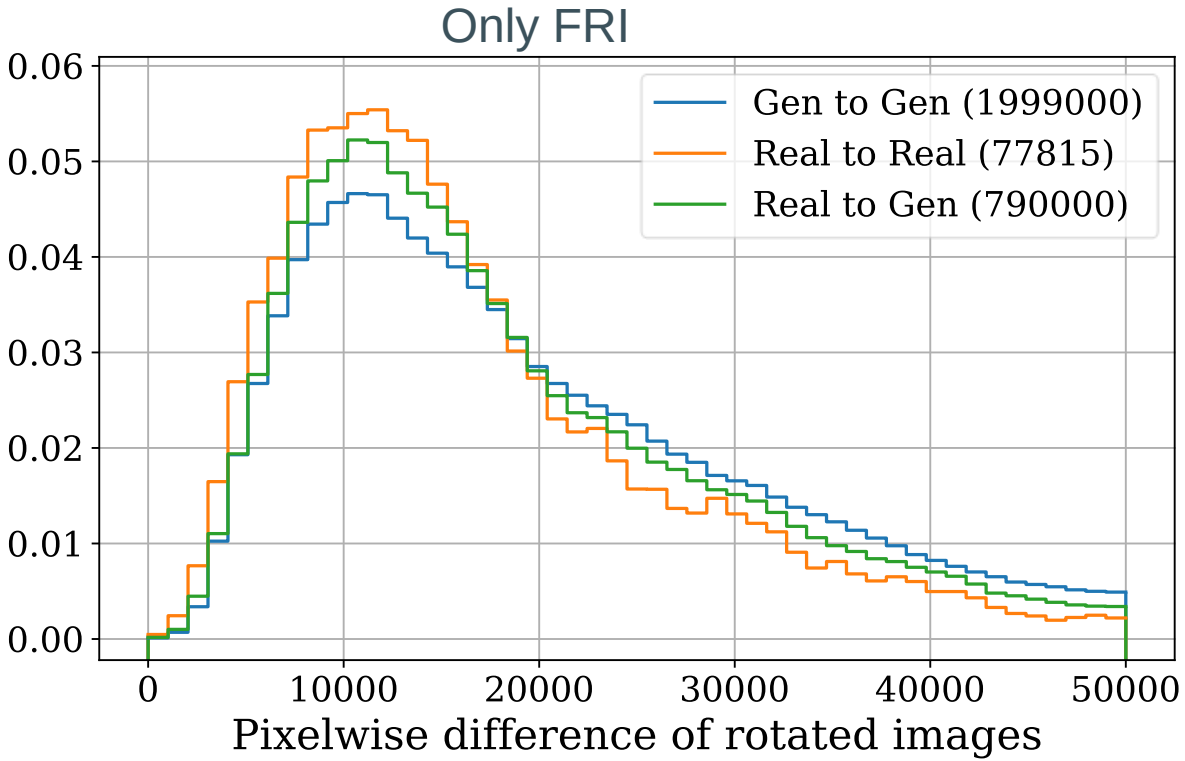


original rotated

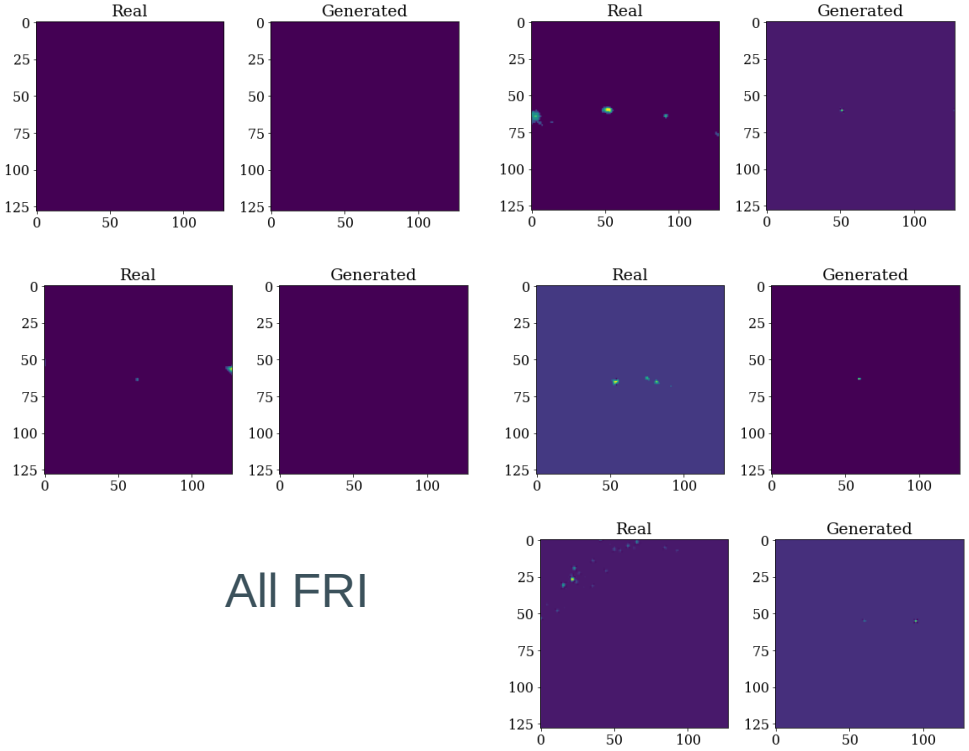
- The range is not clear at all
- The structures are not understood yet
- We had expected something uniform



- Mean roughly the same
- Real images are slightly more similar to each other

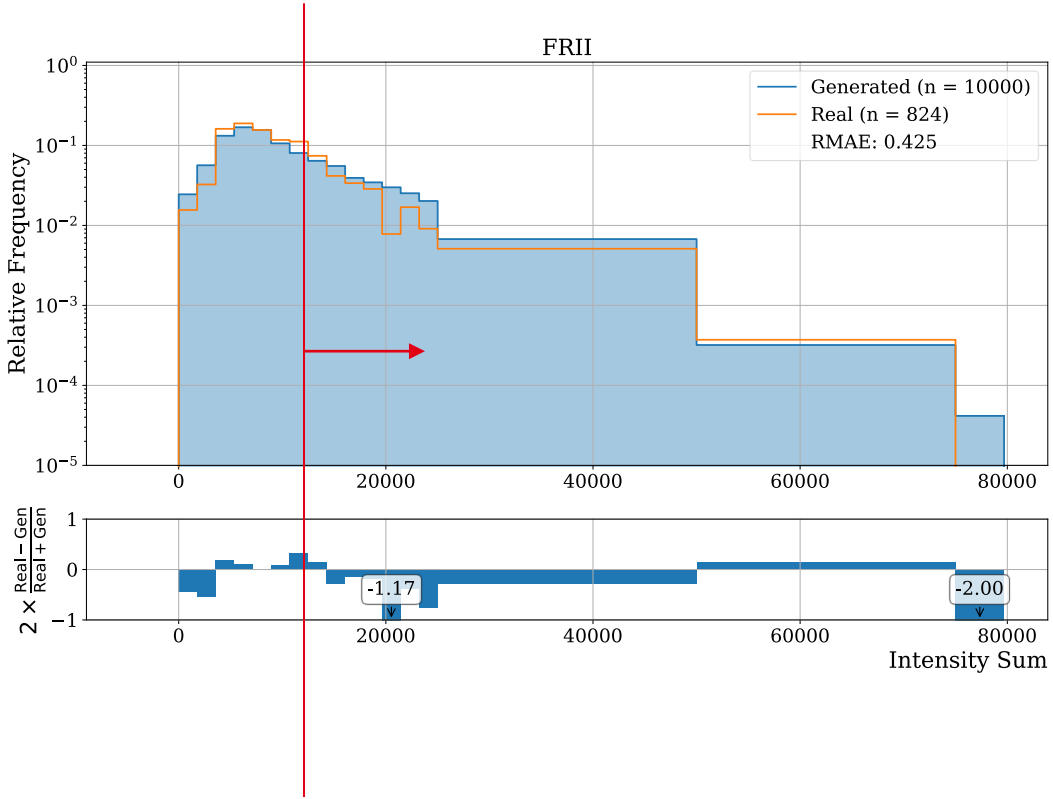


- 5 best matching pairs of real and generated images
- Yeah!! We can accurately reconstruct empty images...



All FRI

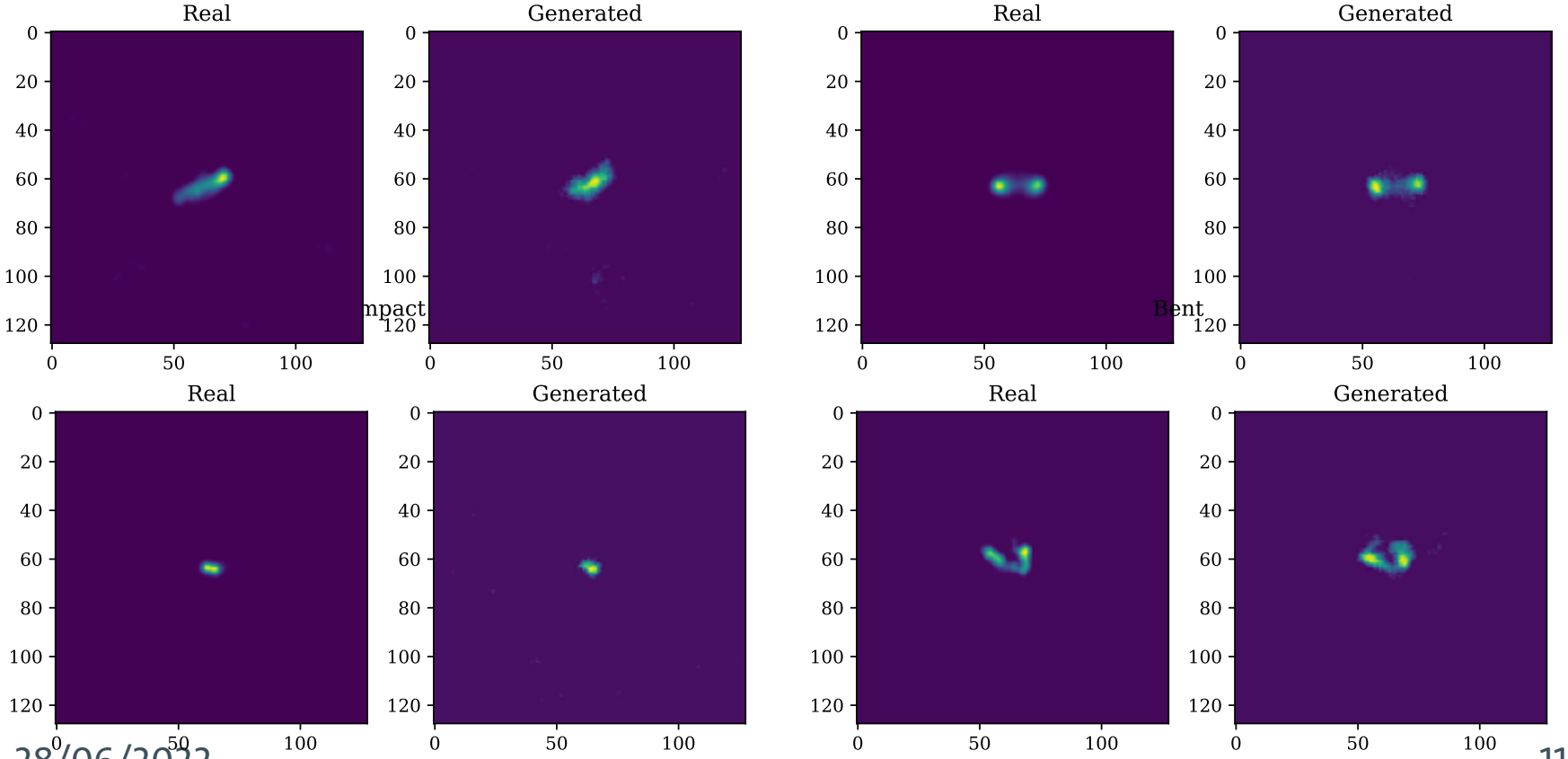
- Only select images with a minimal activity
- Threshold is arbitrary and per class



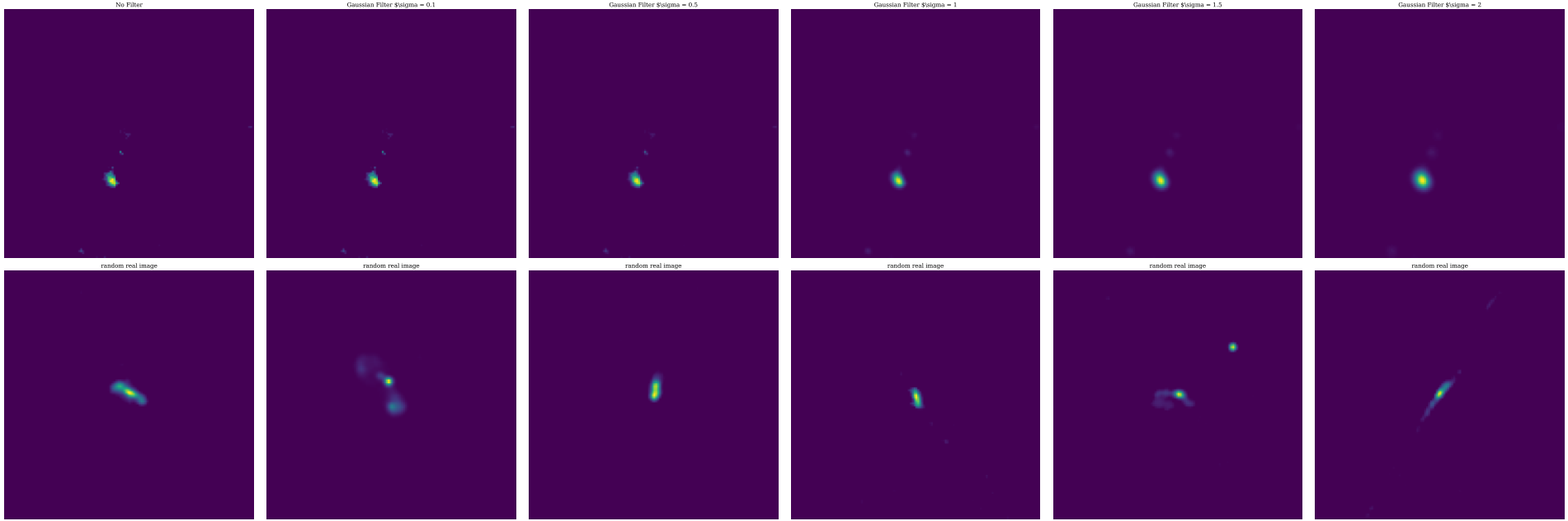
Best matching pairs

FRI

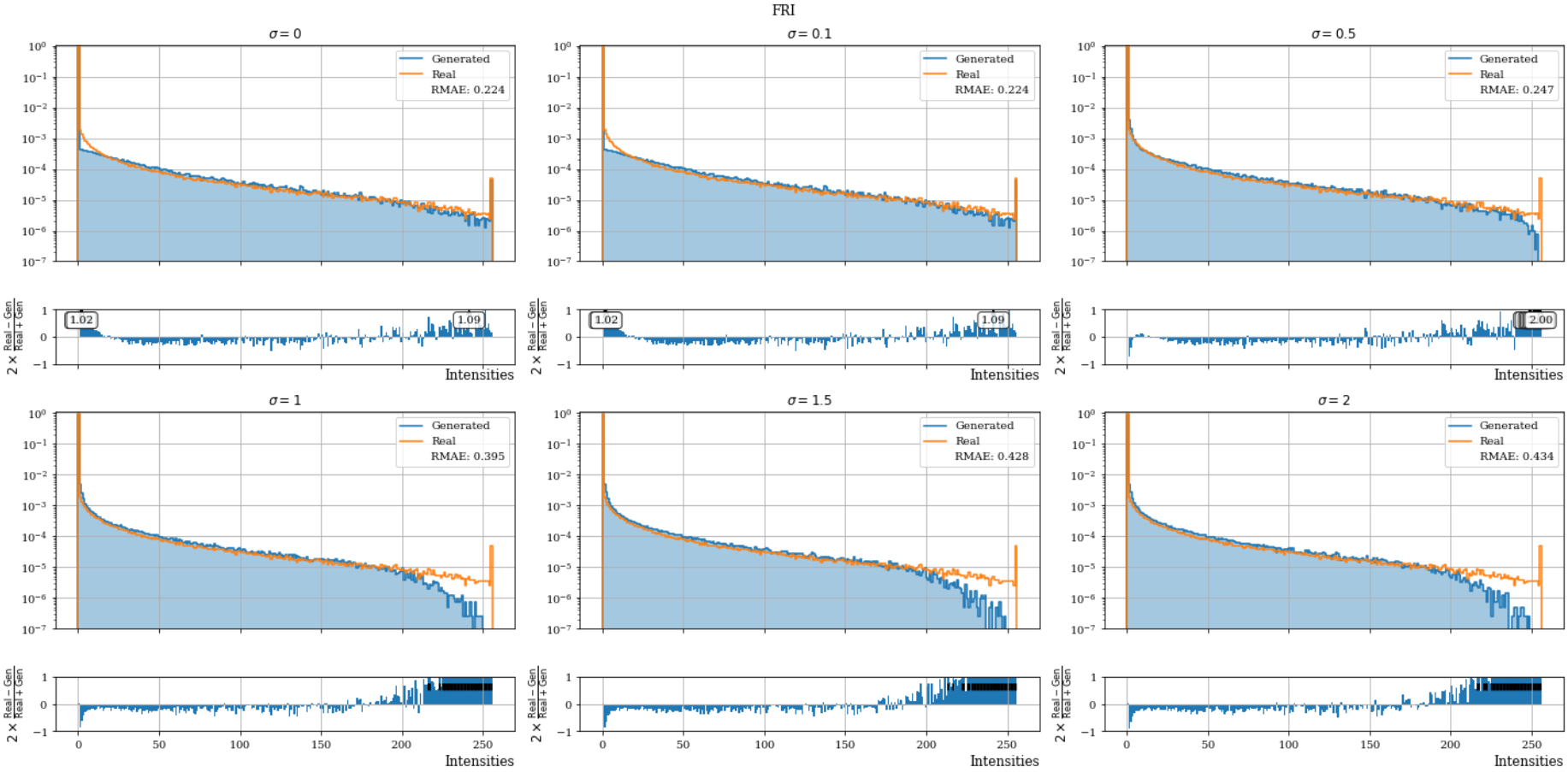
FRII

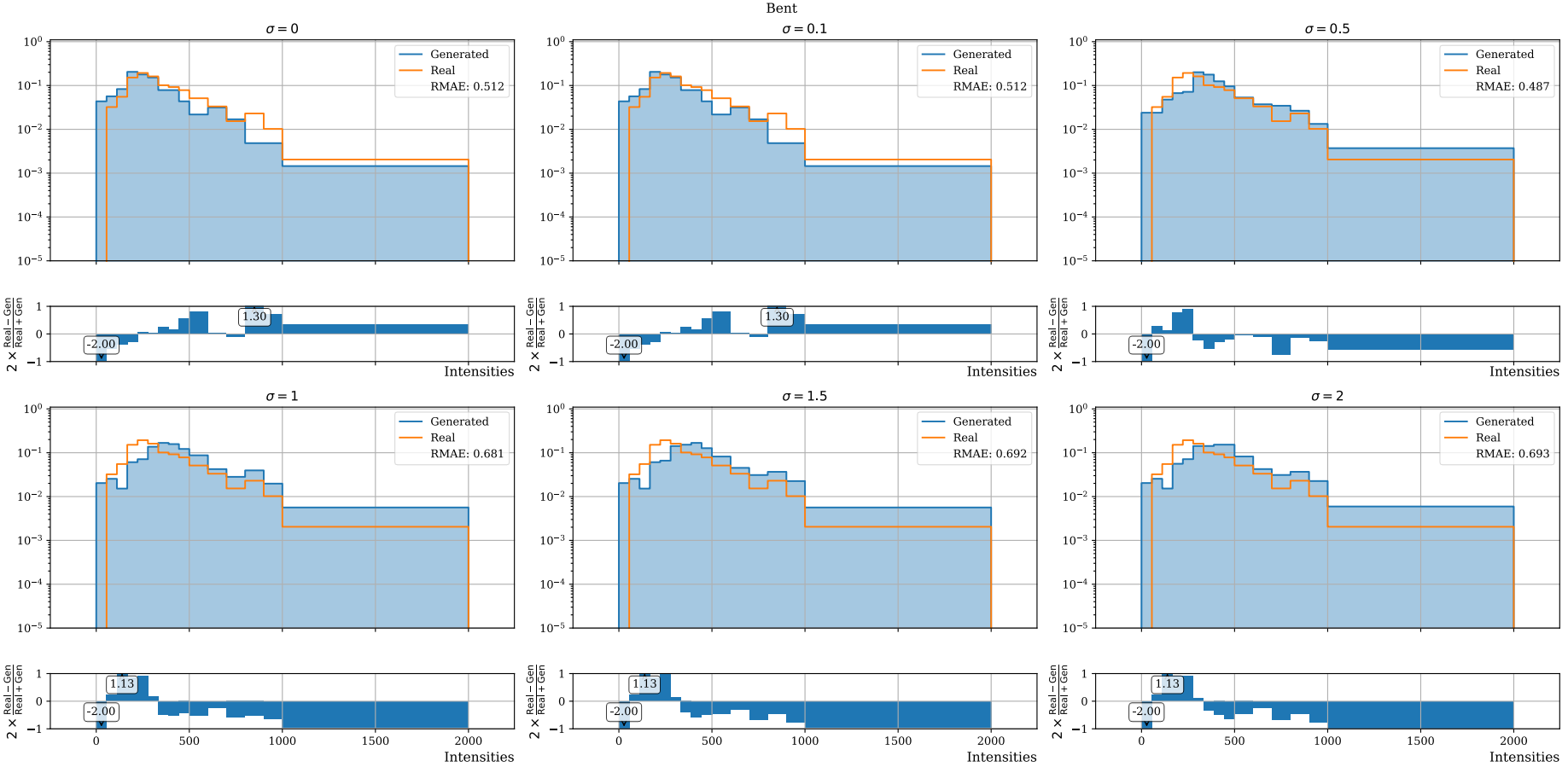


- Looking closer, real images seem to be less edgy / more blurry than generated ones. Maybe quality can be further improved through smearing?
- ->Apply Gaussian smearing with 3x3 kernel and different values for sigma to see effect


No Filter
Gaussian Filter $\sigma = 0.1
Gaussian Filter $\sigma = 0.5
Gaussian Filter $\sigma = 1
Gaussian Filter $\sigma = 1.5
Gaussian Filter $\sigma = 2
random real image
random real image
random real image
random real image
random real image
random real image

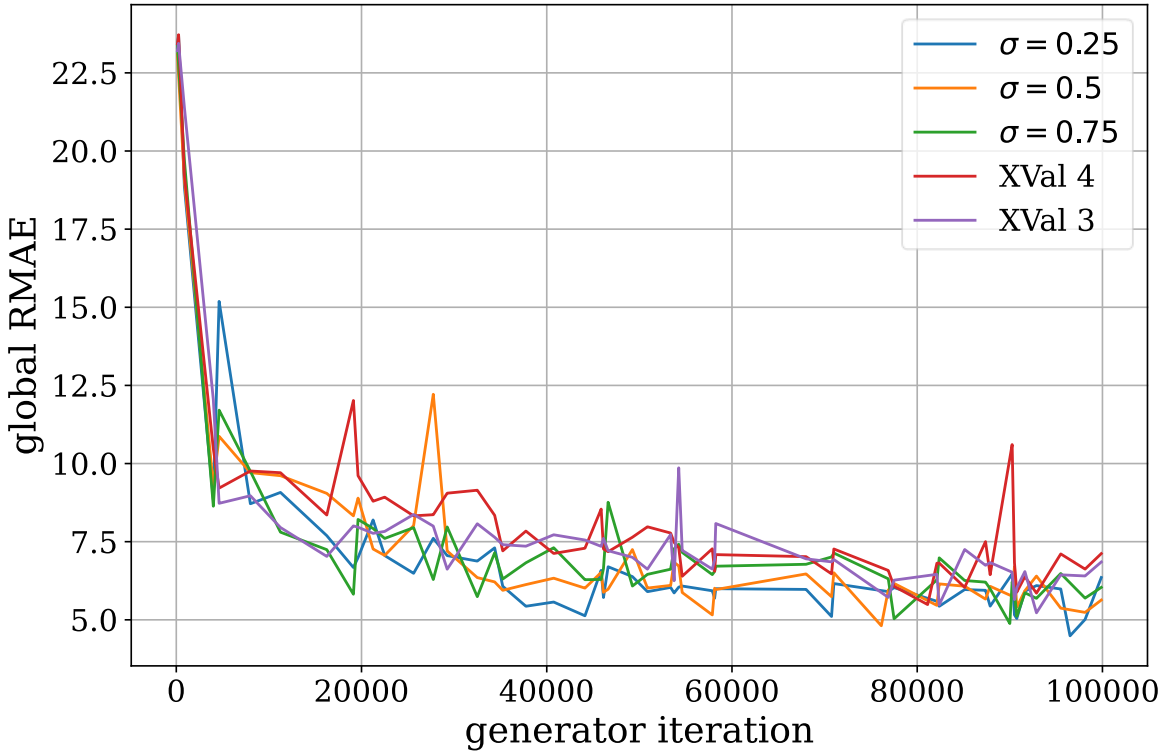
- Does smearing also have a global effect on image distributions?


FRI
σ = 0
Generated
Real
RMAE: 0.224
1.02
1.09
σ = 0.1
Generated
Real
RMAE: 0.224
1.02
1.09
σ = 0.5
Generated
Real
RMAE: 0.247
2.00
σ = 1
Generated
Real
RMAE: 0.395
σ = 1.5
Generated
Real
RMAE: 0.428
σ = 2
Generated
Real
RMAE: 0.434
2 × Real − Gen / Real + Gen
Intensities


Bent
σ = 0
Generated
Real
RMAE: 0.512
σ = 0.1
Generated
Real
RMAE: 0.512
σ = 0.5
Generated
Real
RMAE: 0.487
2 × Real − Gen / Real + Gen
1.30
-2.00
Intensities
σ = 1
Generated
Real
RMAE: 0.681
σ = 1.5
Generated
Real
RMAE: 0.692
σ = 2
Generated
Real
RMAE: 0.693
1.13
-2.00
0
500
1000
1500
2000

- SumI stays constant so of course smearing shifts intensity from high intensity pixels to lower intensity pixels
- Not ideal as this deteriorates the signal (brightest peaks)
- -> introduce smearing already at generator level?
  - The generator is a priori allowed to generate impossible images i.e. pixel values outside of [0,255]
  - Those are then clipped to the allowed range
  - Introduce smearing before clipping so that ‘additional’ intensities are available

- If you're really optimistic, smeared trainings perform sliiiightly better than benchmark runs
- Would need systematic study



- Introduced X-validation (X=5) to our setup
  - Split whole data set in training and test
  - Categorise training in 5 blocks (X=1...5)
  - Train 5 generators each on all blocks bar one (which is used as validation)
  - Determine four best checkpoints for each of the generators
  - Train 5 classifiers, one on each block using the corresponding generator to augment the data set
  - Fully leverage statistical properties and thus have a better estimate on uncertainties while gaining a larger training and test set

Conclusion

- Still trying to improve the setup in many ways
- Accepted to ml.Astro – now doing camera ready version of the abstract
- Writing up publication

Classifier Learning Rate

- Because of a bug, the ‘wrong’ learning rate settings had been set in one of the latest runs
- Turns out that the ‘wrong’ setting might be the better setting as it’s significantly smoother
  - We’re looking for a potentially small effect of improvement
  - Smaller fluctuations might render that more significant
- -> Scan for learning rates

- 1e-5 considerably smoother but might not reach same performance
- Done here for CNN classifier -> FCN in the loop