Real-time multi-stage deep neural network control for SCExAO

B. POU (1,2,3), V. DEO (4), K. AHN (4), S. VIEVARD (4), J. LOZI (4), O. GUYON (4,5,6,7), E. QUINONES (1), M. MARTIN (2), D. GRATADOUR (3)

(1) Barcelona Supercomputing Center (BSC), (2) Universitat Politècnica de Catalunya (UPC), (3) Observatoire de Paris, (4) Subaru Telescope, (5) Steward Observatory, (6) Wyant College of Optical Sciences [University of Arizona], (7) Astrobiology Center of NINS

Introduction

Motivation

- Extreme AO requires exquisite phase reconstruction and prediction in real-time
- Streamline AO pipeline (e.g., no need for gain selection)
- Experiment in a realistic environment

Proposal

- Develop plugin to handle real-time streams with ML and to connect to telescope software
  - **TensorRT**
  - **MILK**
- Test new ML methods on the SCExAO bench

Subaru Telescope and SCExAO

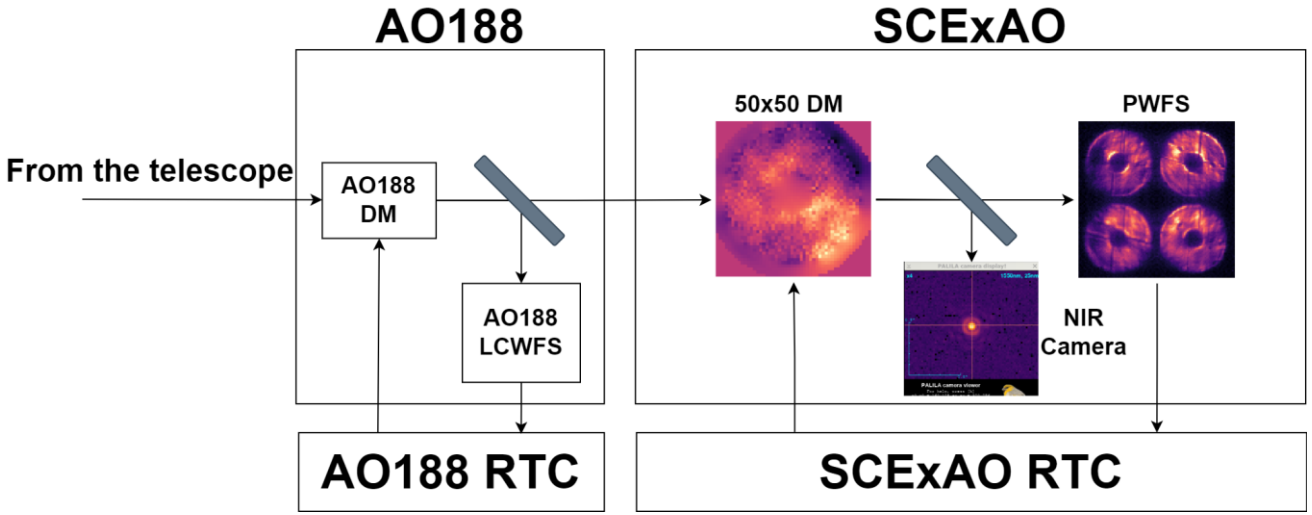
Subaru telescope

- 8.2 m diameter
- Altitude: 4139 m
- Hawaii, USA

Source: Subaru Telescope webpage

SCExAO

- 50x50 Deformable Mirror
- Visual Pyramid Wavefront Sensor
- **Hardware available**
  - AMD EPYC 7763 64-Core Processor (x2). 3.1 GHz, Max 3.5 GHz
  - GPUs: A6000, RTX 3080 Ti (x2), GPU 2080 Ti (x2)



Methods

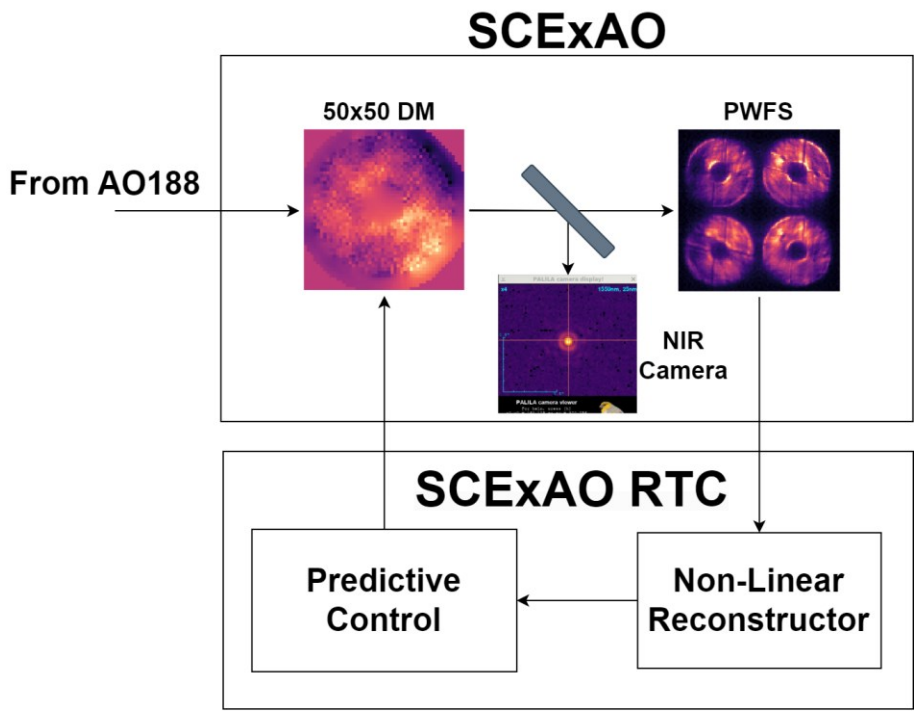
Methodology: multi-stage ML

Non-Linear Reconstruction – Supervised Learning

- P-WFS: non-linear relationship between P-WFS image and phase
- Close the loop even under strong non-linear behaviour

Predictive Control – Reinforcement Learning

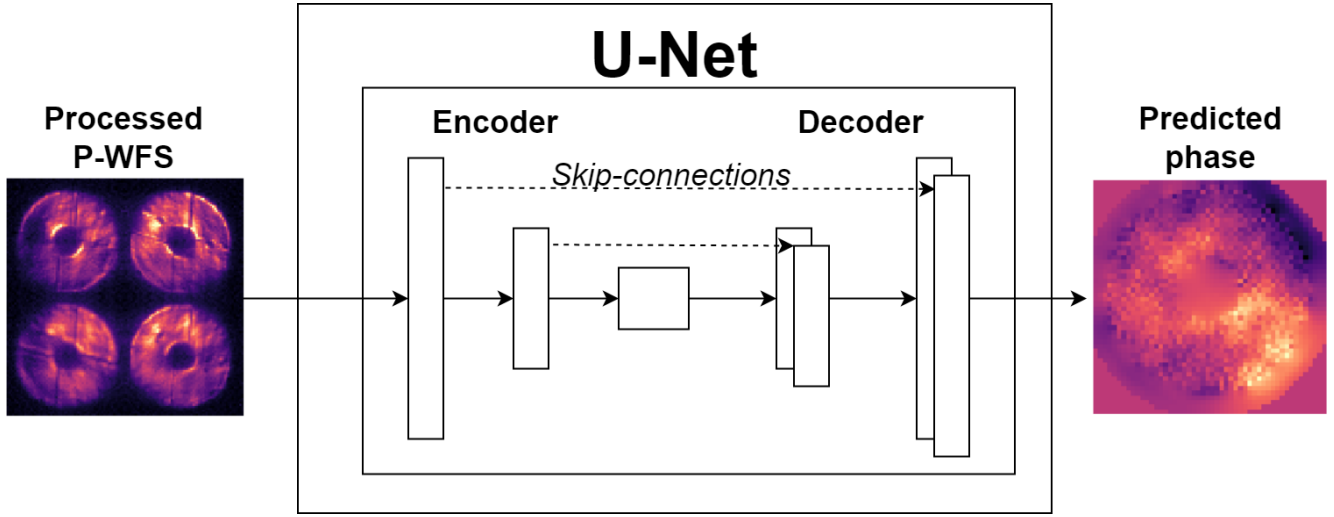
- Correct for error introduced by the system delay and atmospheric evolution



Non-linear reconstruction: U-Net

Training

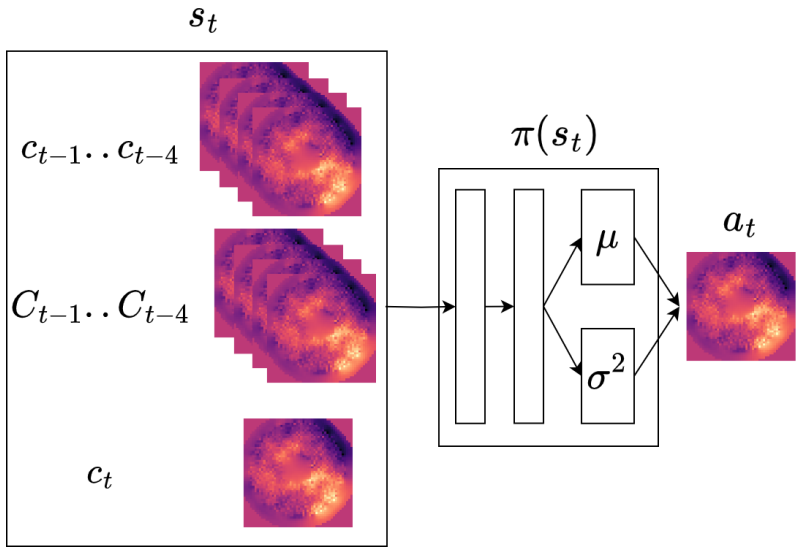
- Supervised Learning
- Gather data from the bench pushing modes/actuators randomly -> Generalize to any distribution
- U-Net with 8 layers
- L1 relative loss



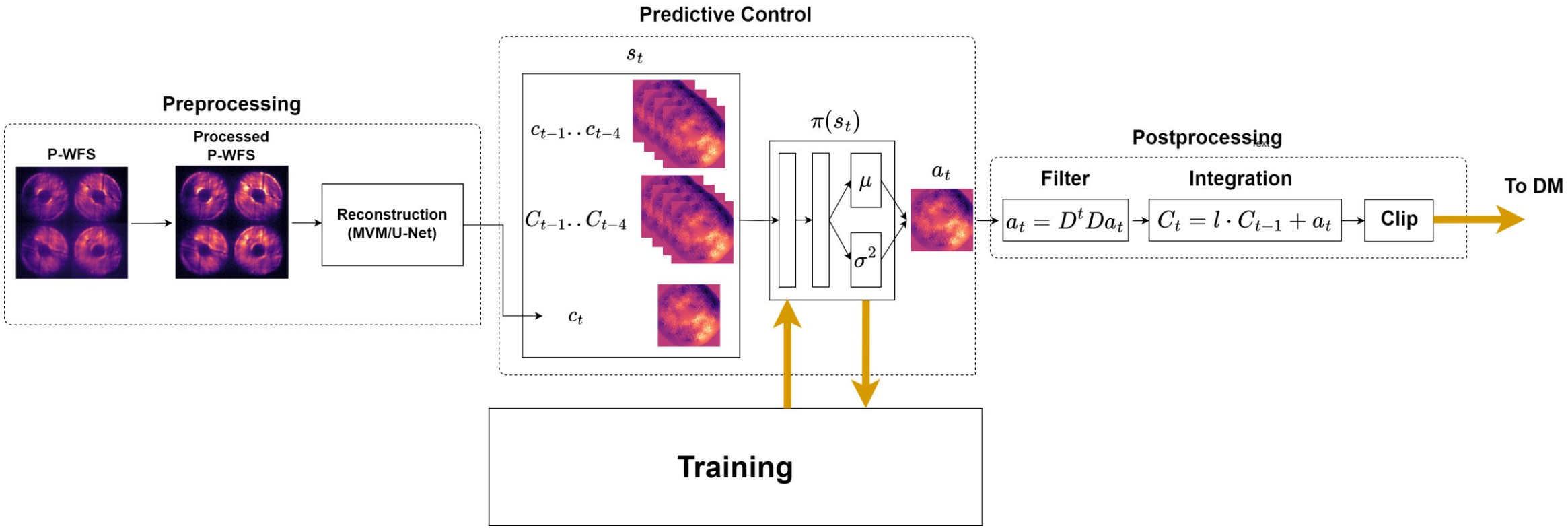
Predictive Control: Reinforcement Learning

Correct for **temporal error** with Deep **Reinforcement Learning (RL)**

- **Trial and error**
- **Policy** (π): given **state** predict **action** that maximises cumulative **reward**
  - State (s_t): reconstruction and history of reconstruction/history of commands
  - Reward (r_t): drives the policy to minimise future reconstructions
  - Action (a_t): residual command
- **Model-free RL**
- **Online training**



Pipeline

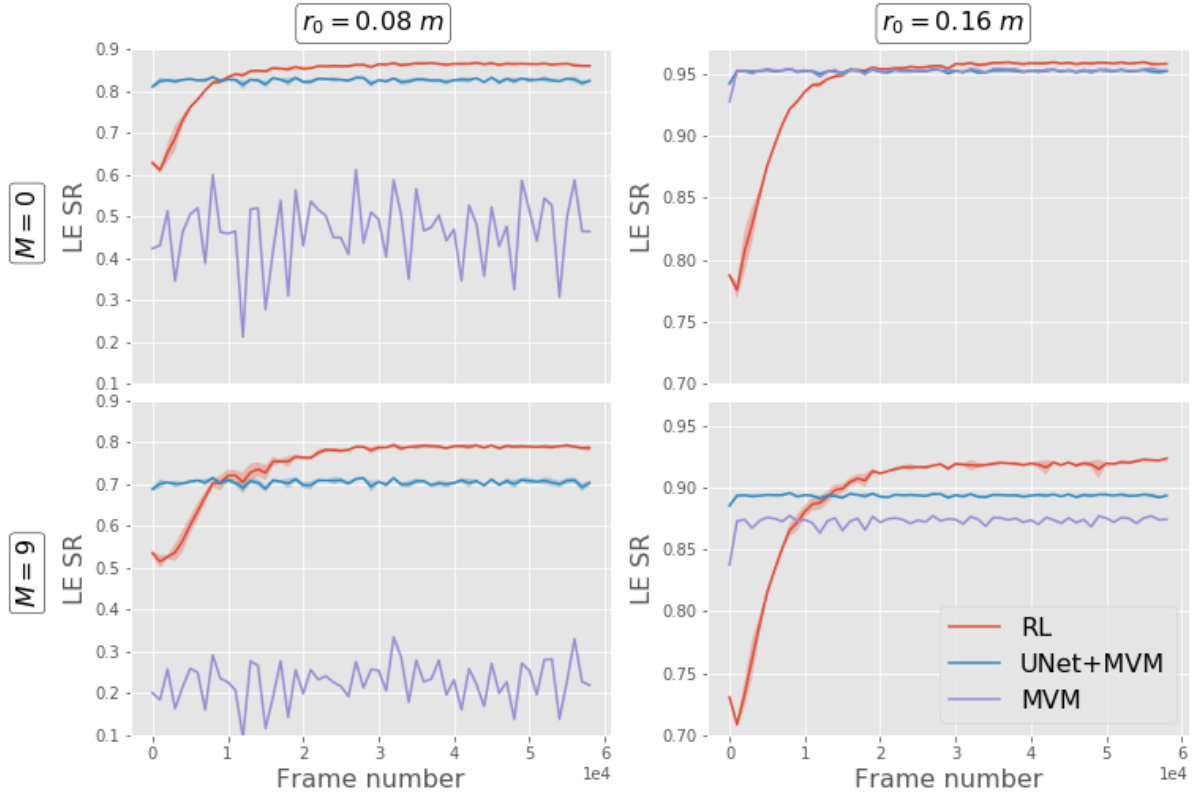

Predictive Control
s_t
Preprocessing
P-WFS
Processed P-WFS
Reconstruction (MVM/U-Net)
$c_{t-1}..c_{t-4}$
$C_{t-1}..C_{t-4}$
c_t
$\pi(s_t)$
μ
σ^2
a_t
Postprocessing
Filter
Integration
$a_t = D^t D a_t$
$C_t = l \cdot C_{t-1} + a_t$
Clip
To DM
Training

Simulation results (COMPASS)

- 40x40 DM, 56x56 P-WFS
- Various atmospheric conditions

Differences with real-world

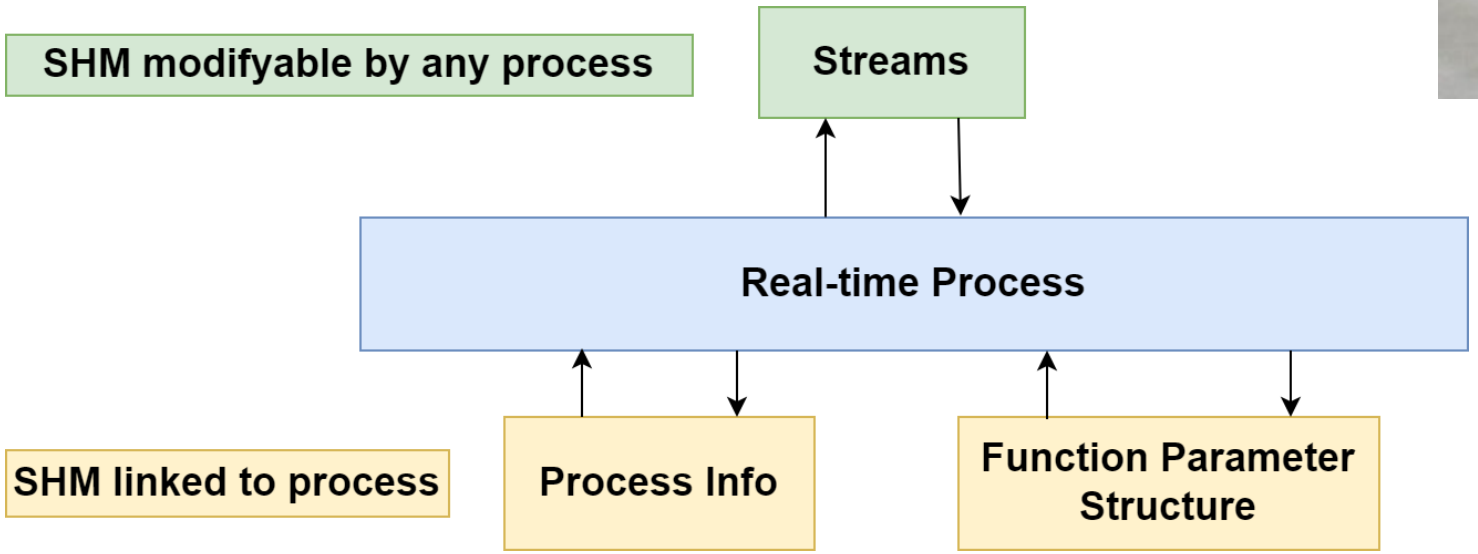
Ensure real time performance



MILK: Modular Image processing Library toolKit

Framework for high-performance image processing with shared memories (SHM)

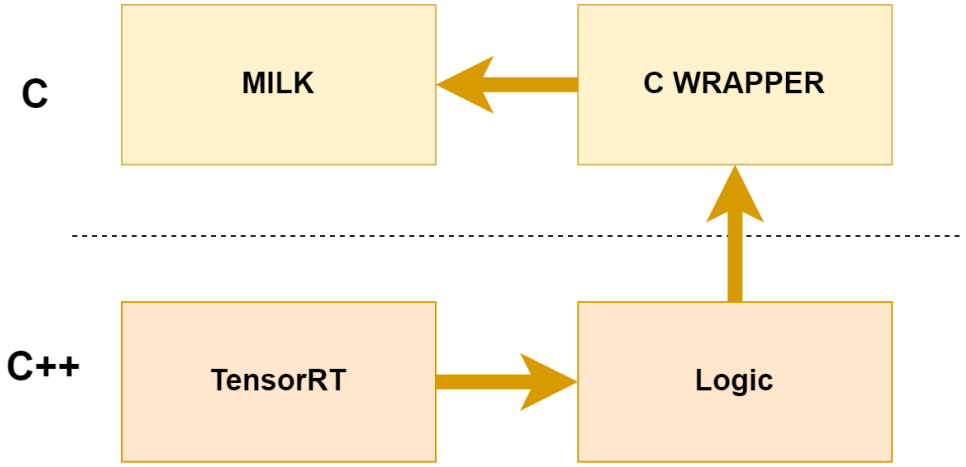
- **Main components:**
  - **Function Parameter Structure (FPS)**: Reading/writing parameters of processes
  - **Process information (ProcInfo):** Process management
  - **Streams**
- **Modularity**



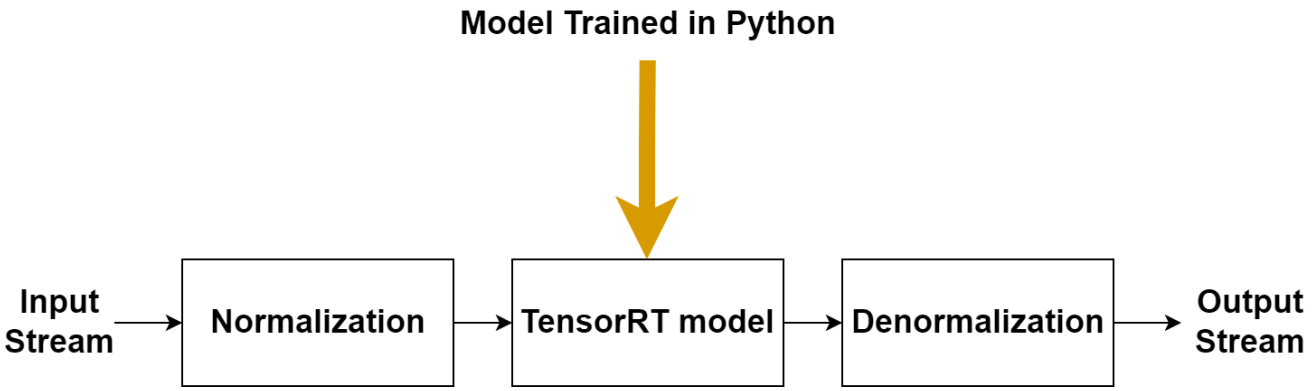
Deep neural network plugin for MILK

- Integrate MILK with high performance computing library TensorRT
  - Offline training models
  - Online training models

Code structure



Offline training models

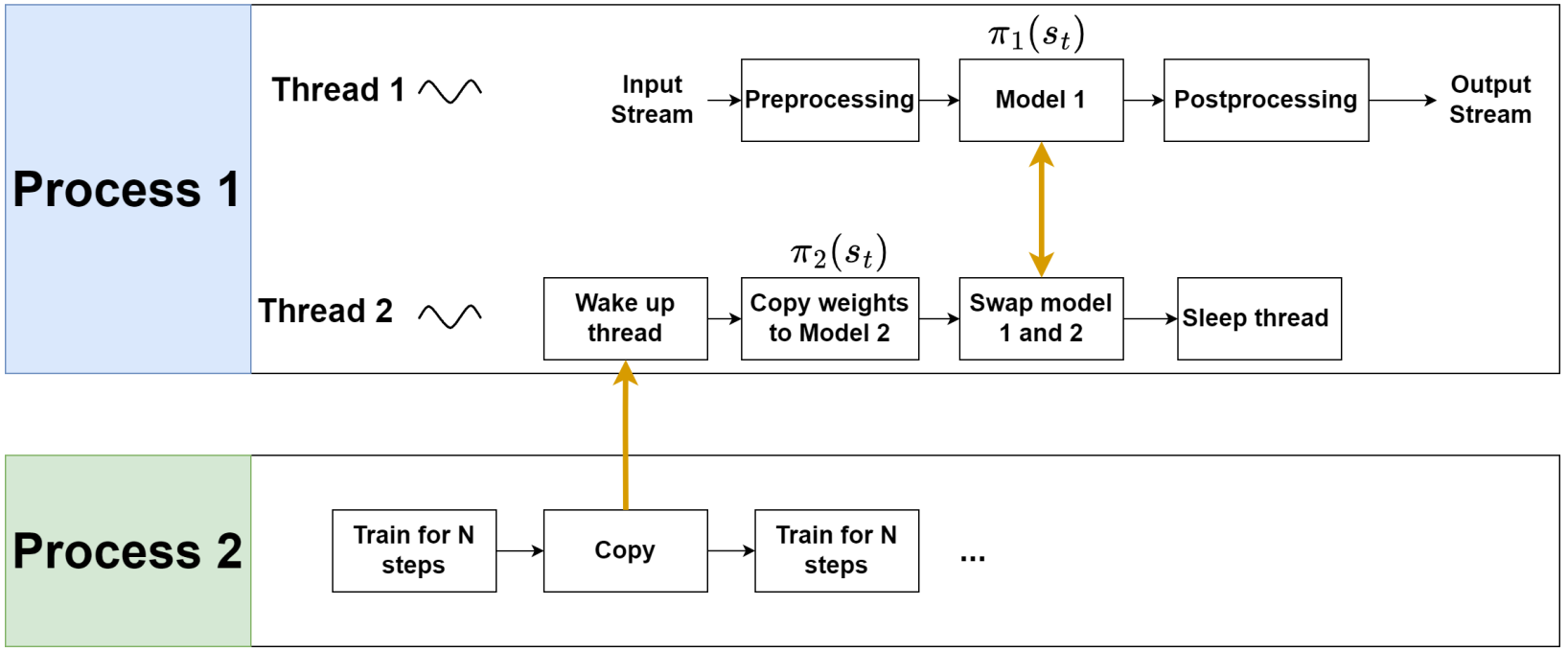


Deep neural network plugin for MILK II

Online training models

C/C++

(Hard real-time)



Python

(No hard real-time)

From ms to sub-ms!

Results

RL results: bench

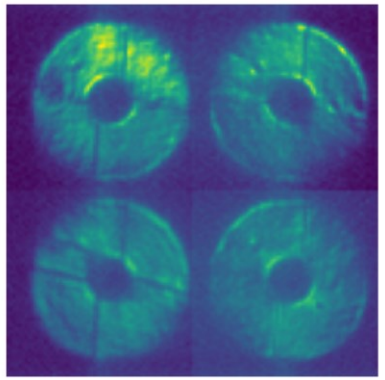
No fitting error

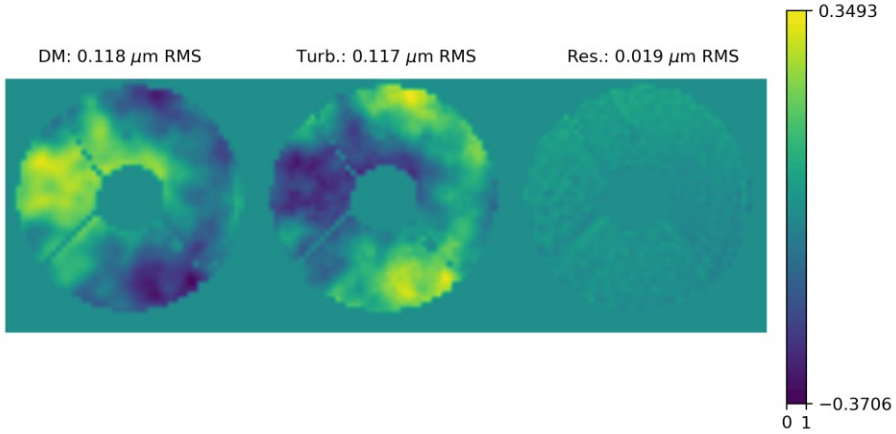
Loop parameters:

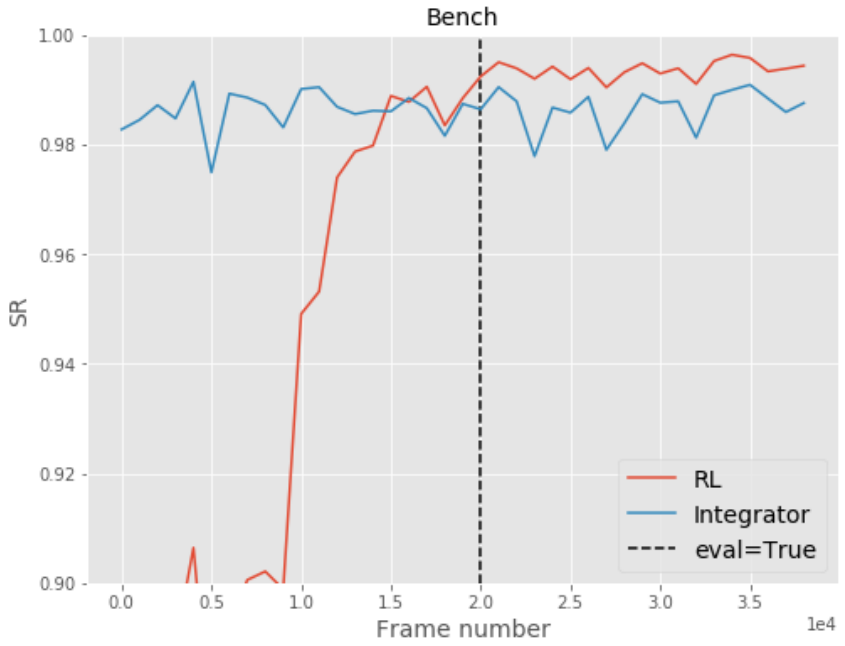
- 2 Khz
- Num. modes controlled = 500
- PWFS $r_{modulation} = 100$ mas
- Bright star

Atmos parameters:

- $A_{atmos} = 0.2$ µm
- Wind speed $= 20$ m/s
- Reduced amplitude on lower order modes (to simulate first stage)

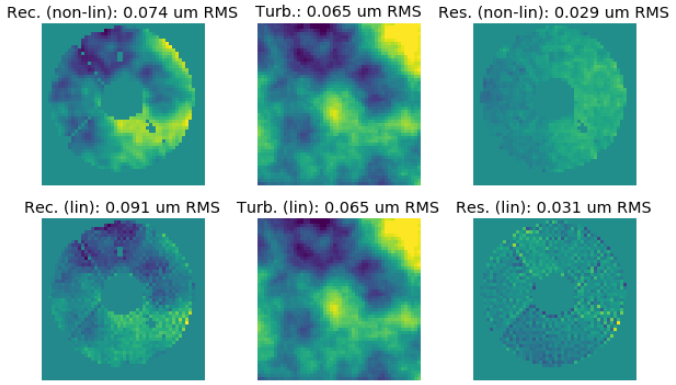




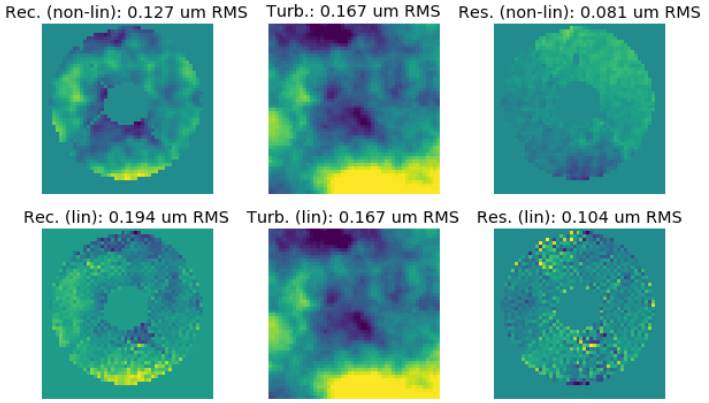


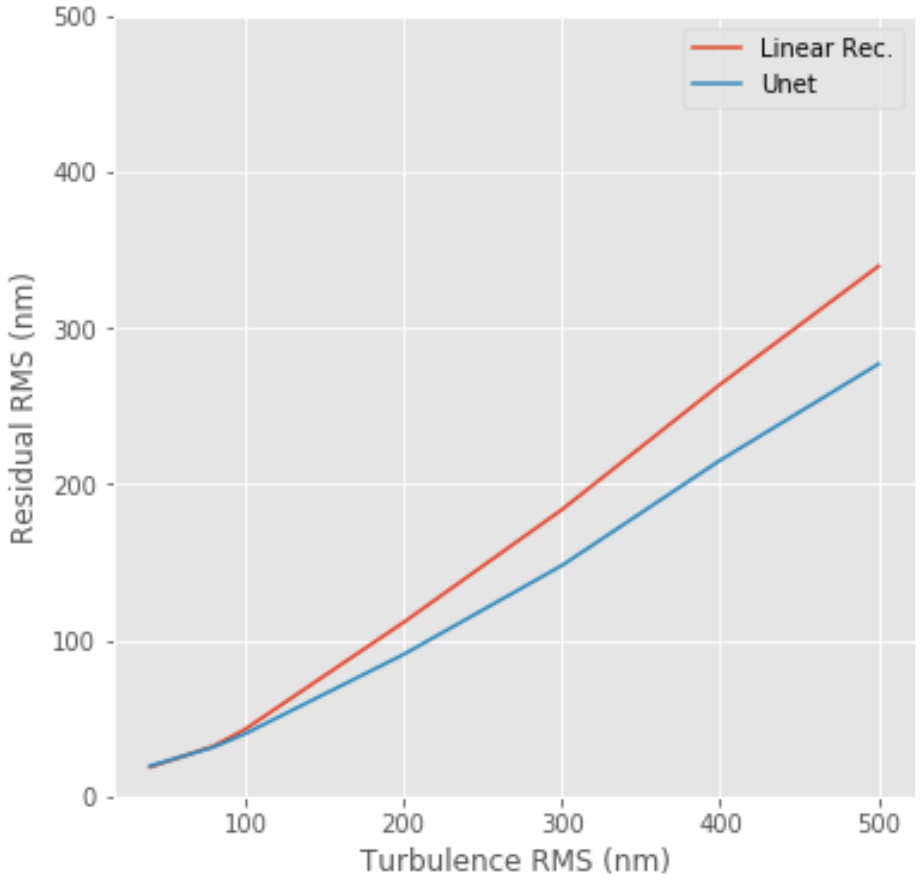
U-Net results: bench (same parameters)

Example 1



Example 2

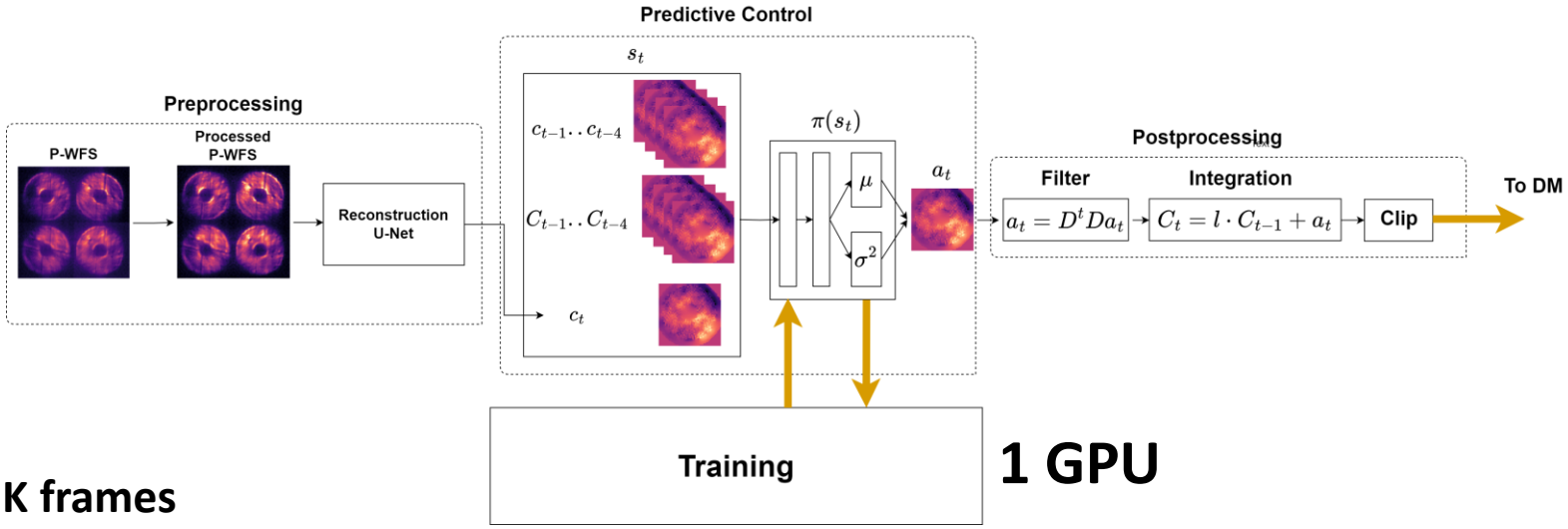




Tip tilt removed

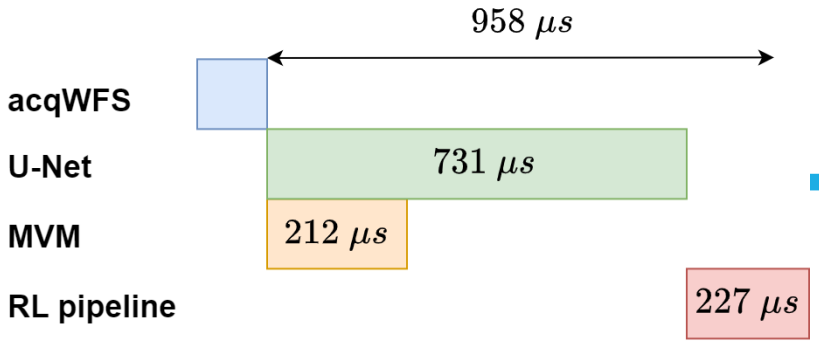
Real-time results: RL + U-Net

- Loop Frequency: 1 KHz
- Hardware:
  - Using 1 GPU A6000
  - Using 2 GPU RTX3080 Ti
  - Using 1 GPU RTX2080 Ti
  - 1 CPU Core per process



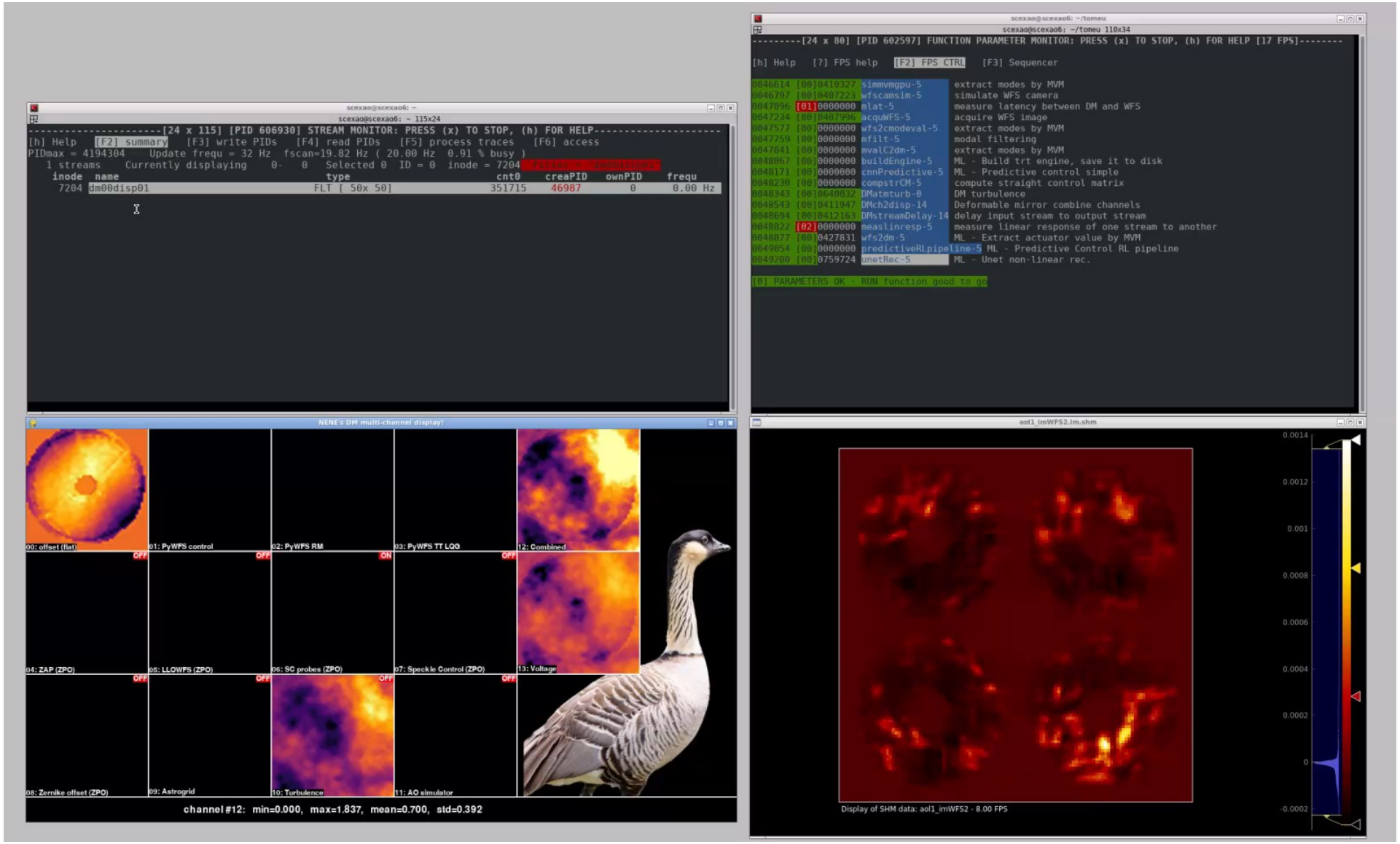
1 GPU

Average over 40 K frames



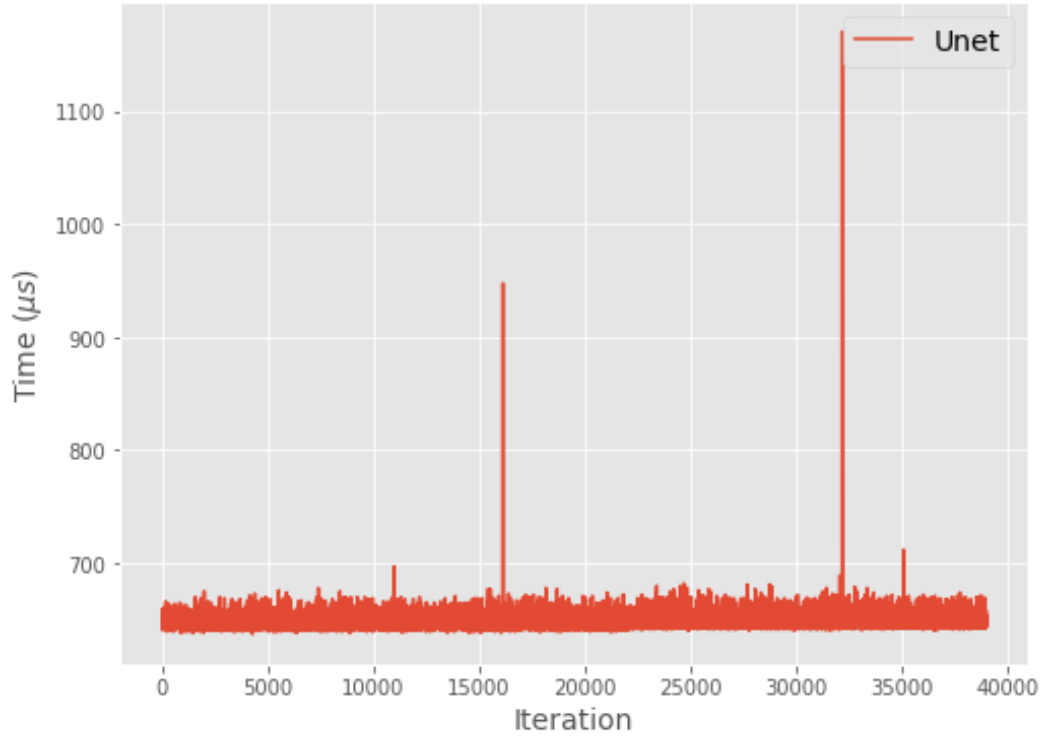
Critical Path

Could run at around 1 kHz


scexao@scexao6: ~
scexao@scexao6: ~ 115x24
[24 x 115] [PID 606930] STREAM MONITOR: PRESS (x) TO STOP, (h) FOR HELP
[h] Help [F2] summary [F3] write PIDs [F4] read PIDs [F5] process traces [F6] access
PIDmax = 4194304 Update frequ = 32 Hz fscan=19.82 Hz (20.00 Hz 0.91 % busy)
1 streams Currently displaying 0- 0 Selected 0 ID = 0 inode = 7204
inode name type cnt0 creaPID ownPID frequ
7204 dm00disp01 FLT [50x 50] 351715 46987 0 0.00 Hz
scexao@scexao6: ~/tomeu
scexao@scexao6: ~/tomeu 110x34
[24 x 80] [PID 602597] FUNCTION PARAMETER MONITOR: PRESS (x) TO STOP, (h) FOR HELP [17 FPS]
[h] Help [?] FPS help [F2] FPS CTRL [F3] Sequencer
0046614 [00]0410327 simmvmgpu-5 extract modes by MVM
0046797 [00]0407223 wfscamsim-5 simulate WFS camera
0047096 [01]0000000 mlat-5 measure latency between DM and WFS
0047234 [00]0407996 acquWFS-5 acquire WFS image
0047577 [00]0000000 wfs2cmodeval-5 extract modes by MVM
0047759 [00]0000000 mfilt-5 modal filtering
0047841 [00]0000000 mvalC2dm-5 extract modes by MVM
0048067 [00]0000000 buildEngine-5 ML - Build trt engine, save it to disk
0048171 [00]0000000 cnnPredictive-5 ML - Predictive control simple
0048230 [00]0000000 compstrCM-5 compute straight control matrix
0048343 [00]0640032 DMatmturb-0 DM turbulence
0048543 [00]0411947 DMch2disp-14 Deformable mirror combine channels
0048694 [00]0412163 DMstreamDelay-14 delay input stream to output stream
0048822 [02]0000000 measlinresp-5 measure linear response of one stream to another
0048877 [00]0427831 wfs2dm-5 ML - Extract actuator value by MVM
0049054 [00]0000000 predictiveRLpipeline-5 ML - Predictive Control RL pipeline
0049280 [00]0759724 unetRec-5 ML - Unet non-linear rec.
[0] PARAMETERS OK - RUN function good to go
NENE's DM multi-channel display!
00: offset (flat)
01: PyWFS control
02: PyWFS RM
03: PyWFS TT LQG
12: Combined
04: ZAP (ZPO)
05: LLOWFS (ZPO)
06: SC probes (ZPO)
07: Speckle Control (ZPO)
13: Voltage
08: Zernike offset (ZPO)
09: Astrogrid
10: Turbulence
11: AO simulator
channel #12: min=0.000, max=1.837, mean=0.700, std=0.392
aol1_imWFS2.im.shm
Display of SHM data: aol1_imWFS2 - 8.00 FPS
0.0014
0.0012
0.001
0.0008
0.0006
0.0004
0.0002
0
-0.0002

Real-time: Issues

Jitter



Time per update (RL)

- Time per update: 0.08 ms
- Total updates to learn from scratch: 20K
- Total time: 26 min
- Still, RL won't be learning from scratch all the time

Conclusion

SCExAO is a unique environment to test new ideas in a state-of-the-art 8m telescope

1. Demonstrated potential of new ML methods on the SCExAO bench (off-real time)
2. Highlighted the extra difficulties in real-life compared to a simulation
3. Implemented a library for real-time inference of deep neural networks which is integrated into RTC software – to be released soon …
4. Once the constraints are solved, I expect to try it on-sky with SCExAO

Integration into COSMIC

Questions?

Feel free to email me at
bartomeu.poumulet@bsc.es