Modeling the 3D Milky Way using Convolutional Neural Networks

***David Cornu[1,2],**
Julien Montillaud[2], Annie Robin[2]

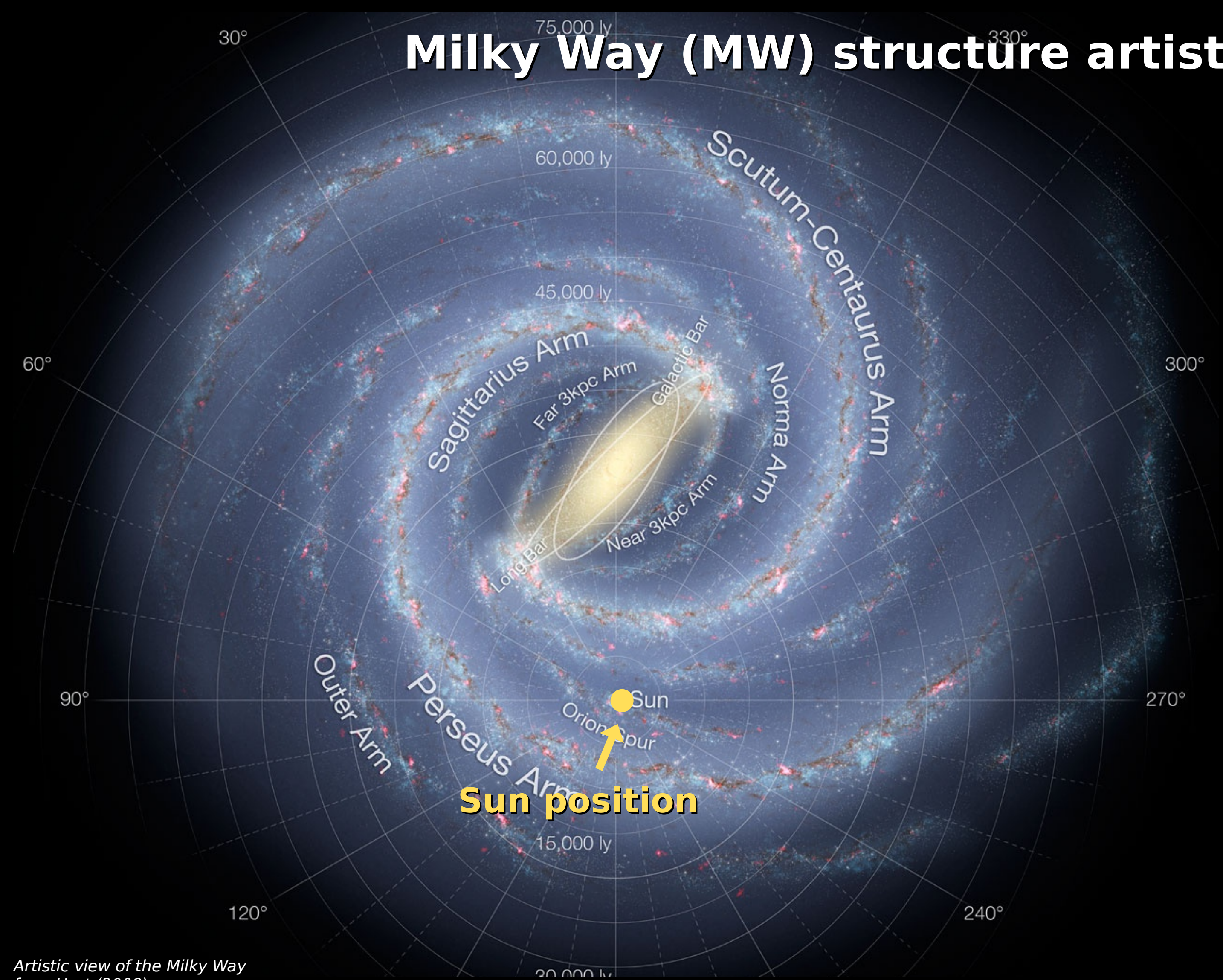
[1] ***LERMA***, *PSL Research Univ., Observatoire de Paris, Paris, France*

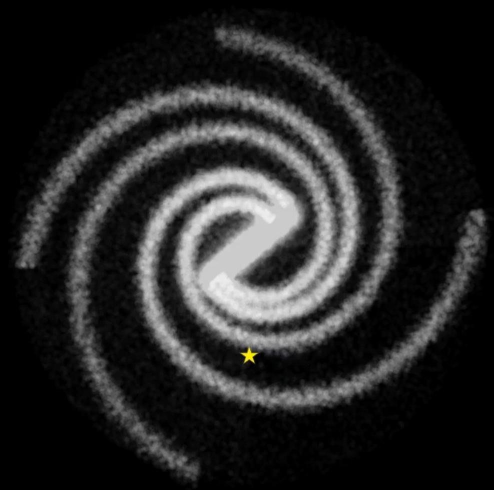
[2] ***Institut UTINAM***, *Univ. Bourgogne Franche-Comté, OSU THETA, Besançon, France*

December 3rd, JCAD 2020

Background image :
Artistic view of the Milky Way by photographer Nick Risinger (2009), adapted from Hurt (2008)

Milky Way (MW) structure artistic view

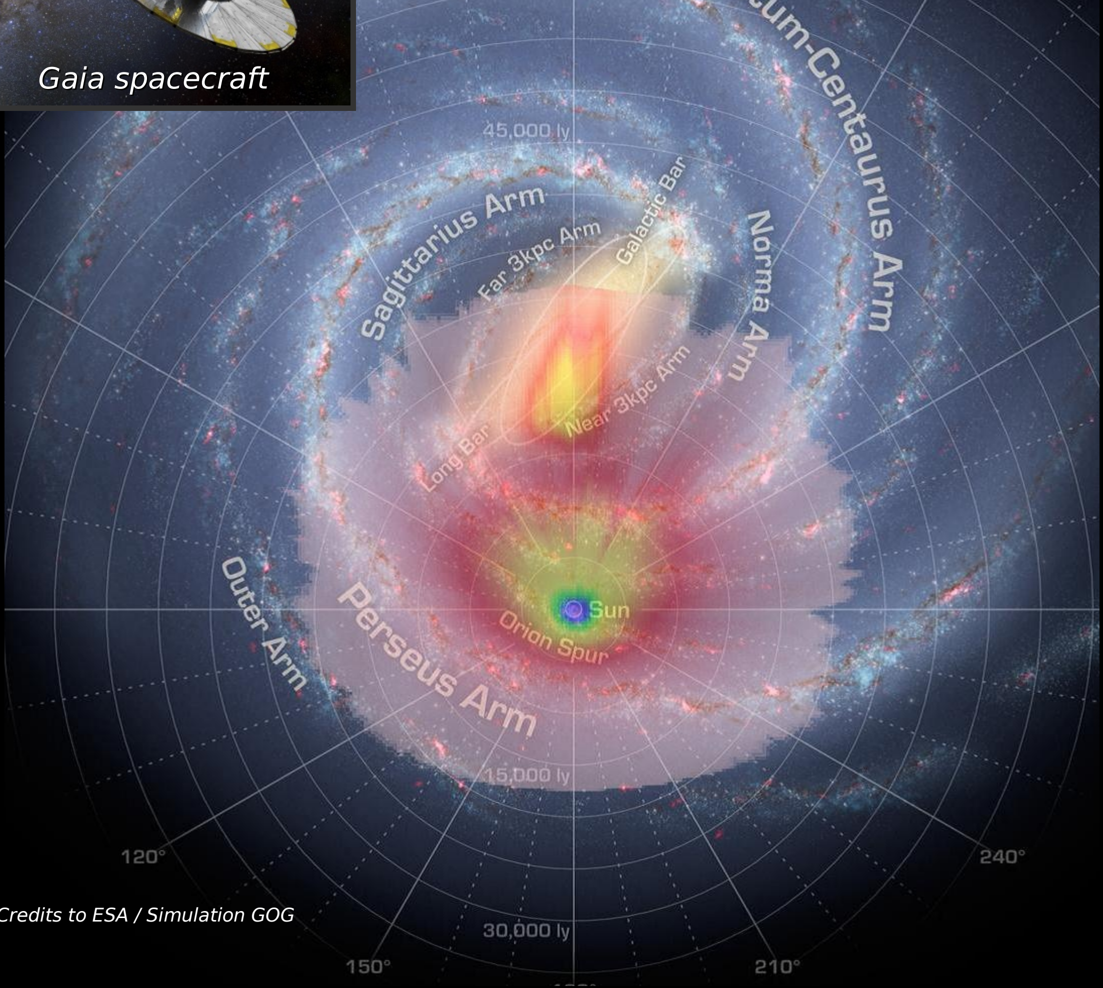




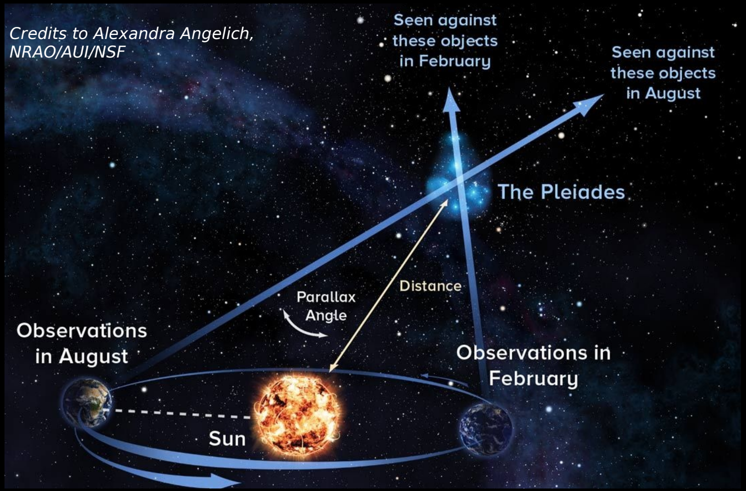
Face-on to edge-on view

Artistic view of the Milky Way from Hurt (2008)

Gaia: new Milky Way structure flagship



Credits to ESA / Simulation GOG



- Parallax measurement for 1.6 billion stars

- Unprecedented distance range for MW survey

→ Groundbreaking for MW structure studies

- Very large dataset
  → one of the first astronomical **Big Data** survey

→ Require sophisticated statistical methods like Machine Learning for data analysis

Machine Learning in Astronomy

Core concept of ML: extract statistical information about a dataset and adapt the response accordingly through a learning process

➡ **Very generic methods:** Artificial Intelligence, computer vision, numerical science, ...

These methods are able to manage:

-High dimensionality **-Big Data** **-Heterogeneous information**

Increasingly used in astronomy:

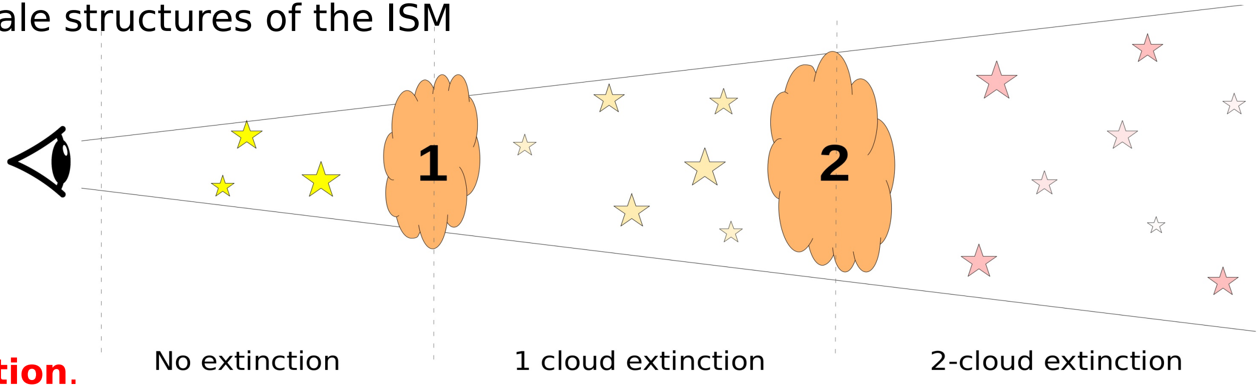
- galaxy classification (Huertas-Company et al. 2015; Welmsley et al. 2020, ...)
- computation accelerator (Grassi et al. 2011)
- ISM structure classification (Beaumont et al. 2011)
- molecular cloud clustering (Bron et al. 2018)
- Identification of ISM turbulence regime (Peek & Burkhart 2019)
- ...

Interstellar medium extinction

The extinction = absorption + scattering

Motivation: extinction traces large scale structures of the ISM

$$A_\lambda = 2.5\log\left(\frac{F^0_\lambda}{F_\lambda}\right)$$



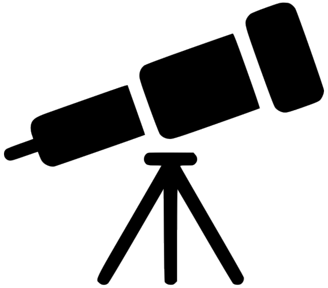
For the diffuse ISM in the Milky Way:

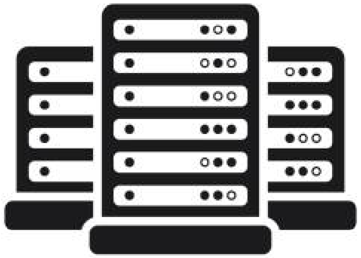
$$A_V \propto E(B-V)$$

→ Reddening directly provides extinction.

What do we have ?

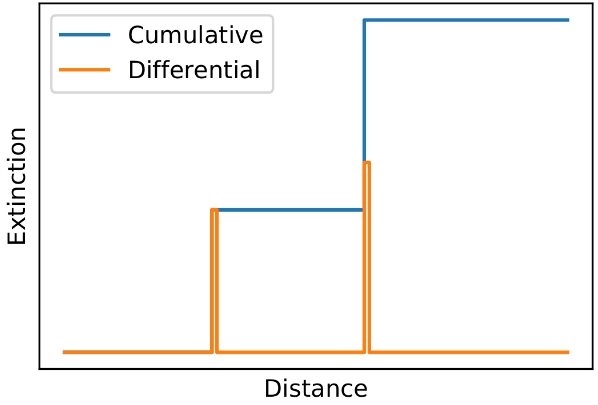
Observations And/Or Milky Way Models





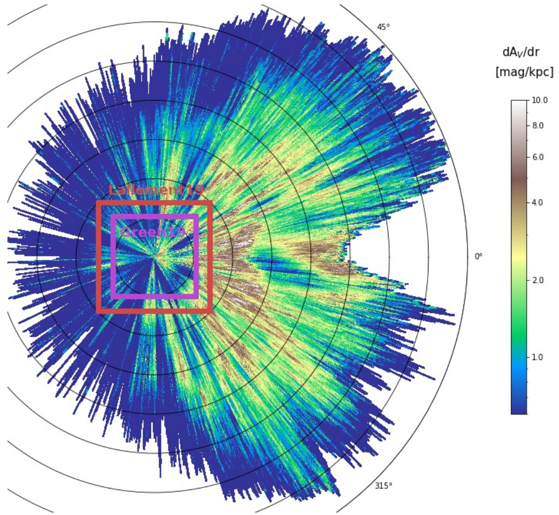
What do we want ?

An extinction profile



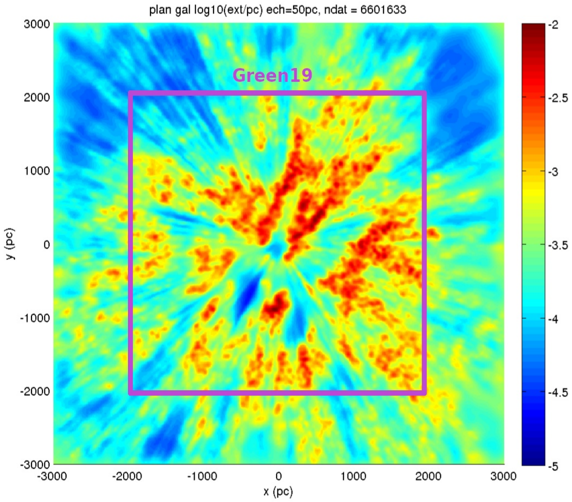
Extinction 3D structure in the Milky Way

Marshall+20



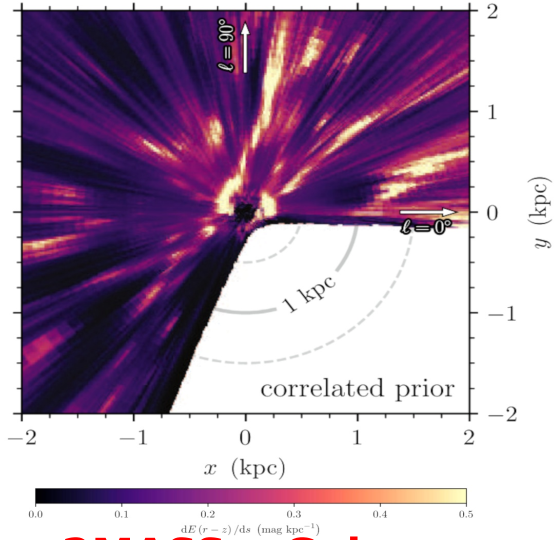
2MASS + Besançon Galaxy Model

Lallement+19



2MASS x Gaia

Green+19



2MASS x Gaia x PanSTARRS

Cross-Matching *infrared and optical surveys limits the maximum distance to the optical surveys*

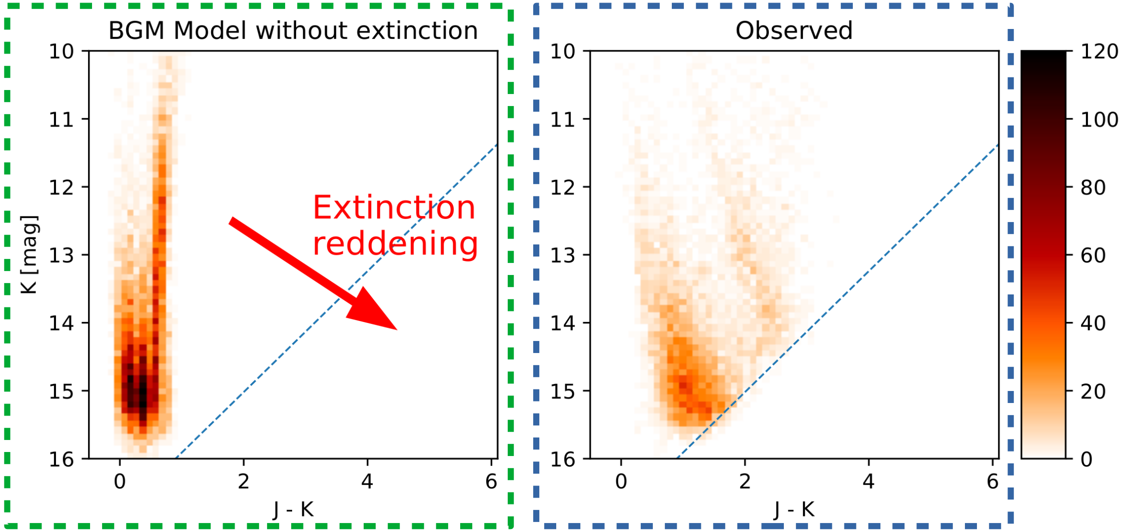
Besançon Galaxy Model (BGM, Robin+2003):

Stellar population synthesis model

- Produce star lists that are **statistically representative** of the Milky Way.
- Provide physical properties of stars and **observable quantities**.

FYI: the BGM is supported by a national astronomical service.
It can be used through a web service at *model.obs-besancon.fr/ws*

Measuring extinction with the BGM



****Using the 2MASS infrared survey observables***

Color-Magnitude Diagram (CMD):
2d histogram of observables for a large list of stars corresponding to a given sightline.

We have two informations:

1) **Observed** stars are affected by the **integrated extinction** along a sightline.

2)The BGM **modeled** stars provides distances **without extinction**

So assuming the BGM is perfect:

Ext. profile = f(observed, modeled)

Why use ML / Neural Networks?

No assumption on **HOW** to make the comparison

Can combine **heterogeneous** surveys

…

How to express it as an ML problem ?

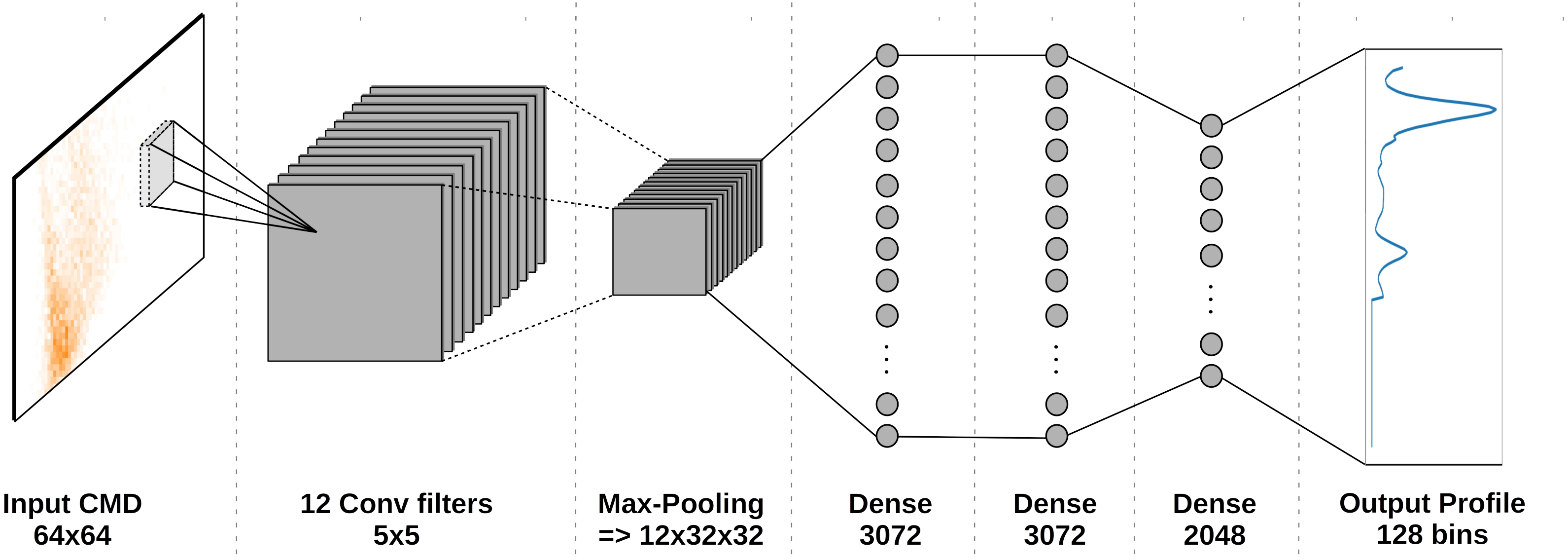
CMD ↔ image (here 64x64 pixels)

Pixel value: number of stars in a given bin

Objective: convert specific **pattern movement** into extinction information

Convolutional Neural Networks (CNNs)

CNNs are built on top of more classical ANN.
→ Convolution layers are specialized toward information distribution and redundancy



CNNs are mostly used with conventional architectures (number of layers, size, activation function, ...)
→ **In our case common architectures were inefficient.**

Training a Deep Neural Network is a difficult task due to the high number of parameters.

But CNN inner structure can be expressed as **matrix operation** or other function that are easy to parallelize and very computationally efficient.

The CIANNA framework

In the context of this study, we fully coded our own **CNN framework called CIANNA**
(Convolutional Interactive Artificial Neural Networks by/for Astrophysicists)

It is fully codded in **C and CUDA**, with support for different **compute schemes**:

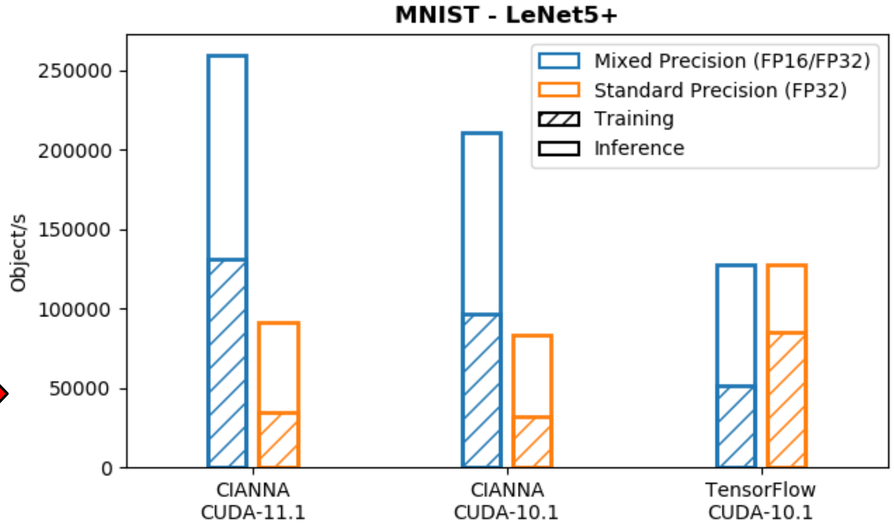
- **No dependency option** (OpenMP if needed)
- **CPU parallelism** (OpenBLAS + OpenMP)
- **GPU acceleration** (Nivida GPU)

CIANNA can be used with either a C or Python interface and presents the following capabilities:

- Arbitrary deep and thick Convolutional or Dense ANN
- Ease of use with high level interfaces
- Various ANN optimization techniques
  (Learning rate decay, momentum, mini-batch, SGD, dropout)
- Dynamical GPU memory loading
- Mixed precision (FP16/FP32) and Tensor Cores support
- …

Performance:
Due to a focus on low latency and minimal overhead,
CIANNA is presently faster than state-of-the-art frameworks
for dense-only architectures and small convolutional ones,
while being slower but still competitive on larger ones.

Benchmark using a Tesla V100 GPU

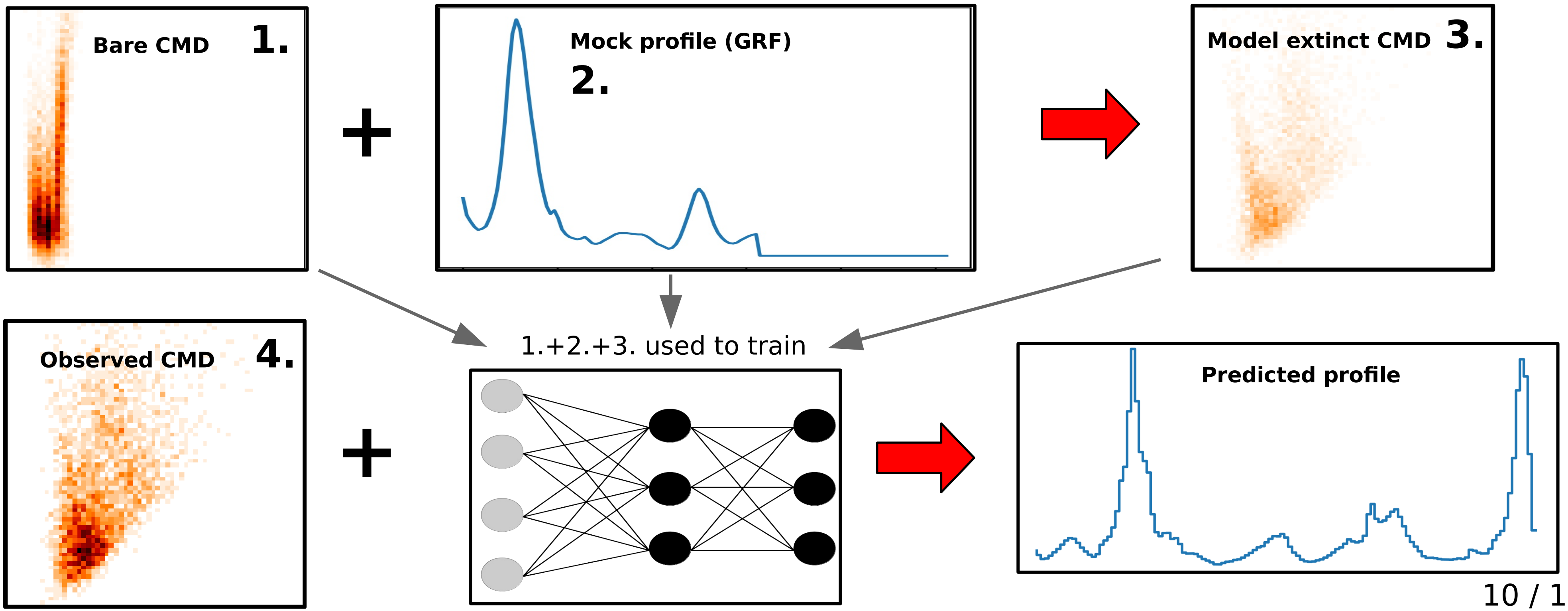


CIANNA is now Open Source (ApacheV2) and can be accessed on github.com/Deyht/CIANNA

Training a CNN from BGM+2MASS

Methodology description:

1) Use the BGM to create « bare » CMDs from 2MASS data.
2) Generate mock extinction profiles (Gaussian Random Fields, GRF).
3) Create a training sample of extincted CMDs with the « target » extinction profile.
4) Use the trained network to predict extinction profile from observed CMDs

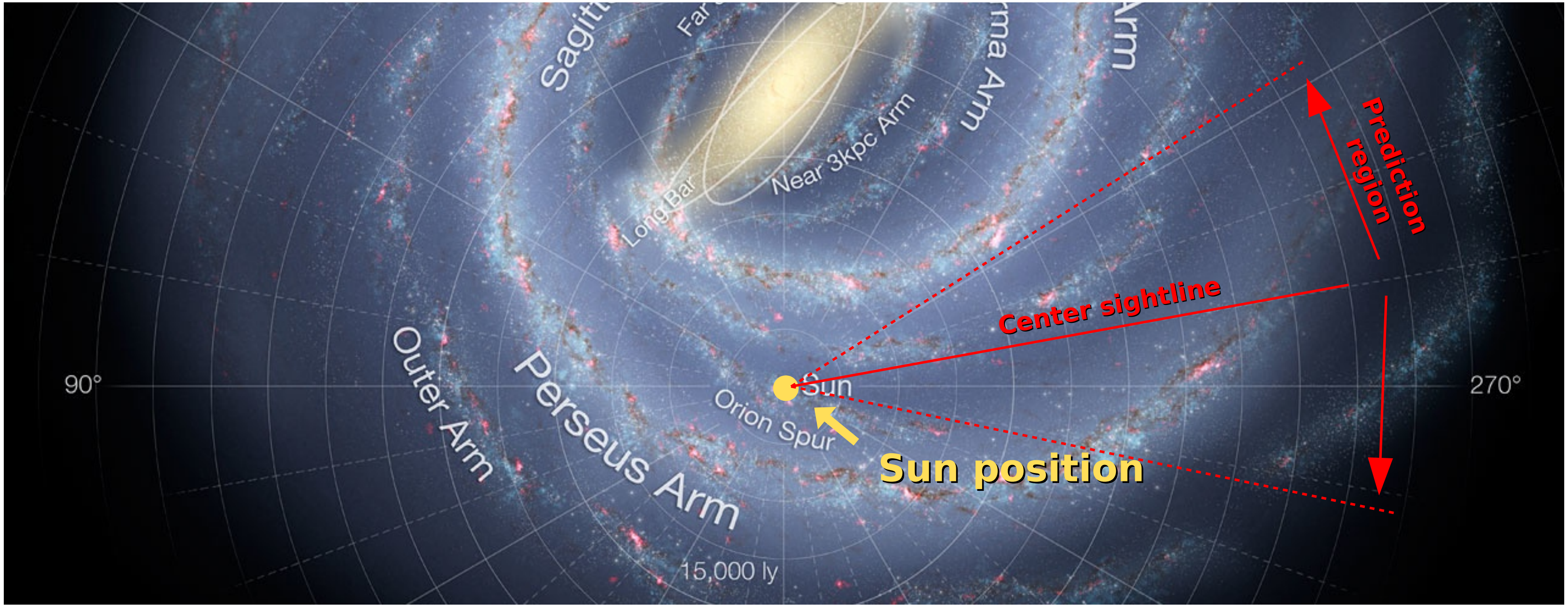


Region selection

We selected the Carina arm tangent region:

- Relatively easy to observe and close
- Important extinction dynamical range
- Other tracers of the tangent are available for comparison

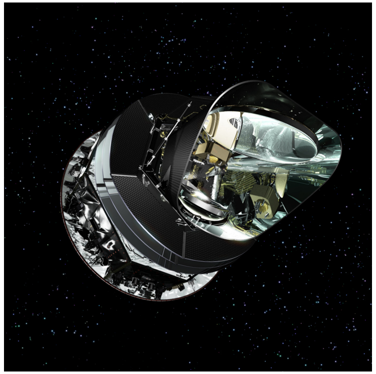
Training: 9 lines of sight **x** 200000 examples = **1.8 10^6 objects.** → 94% train, 5% valid, 1 % test



Training our CNN on this region required up to 60 Go of RAM usage for the training sample and up to 5 hours of compute on a single Tesla V100 of the UFC Mesocenter (~1000h CPU).

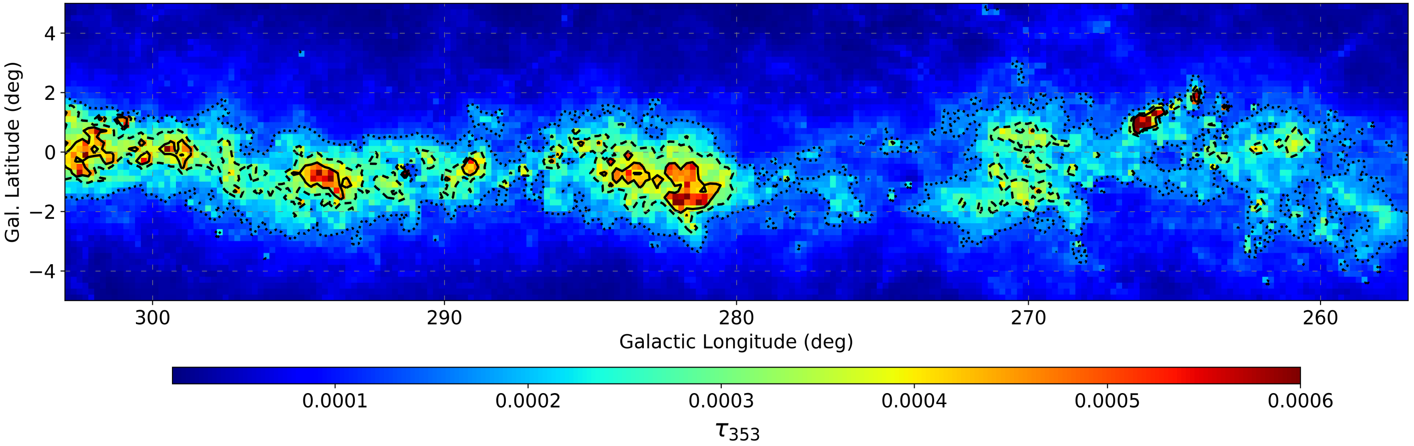
CNN prediction: plan-of-the-sky view

Morphological reference:* *Planck dust emission → proxy for dust distribution

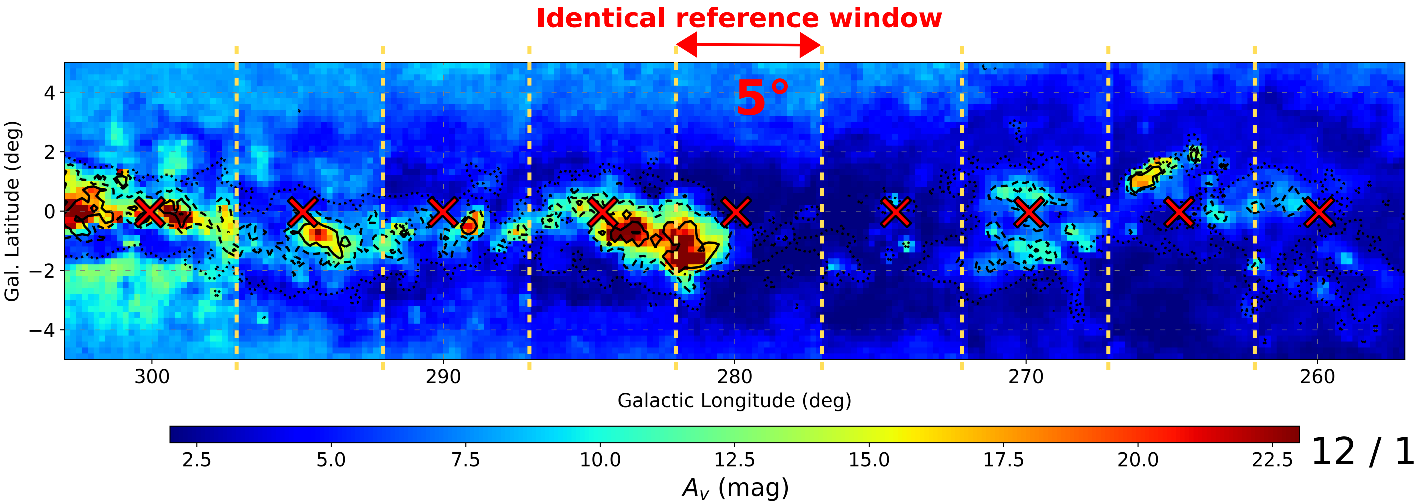


Plane-of-the-sky view → integrated quantities

Planck

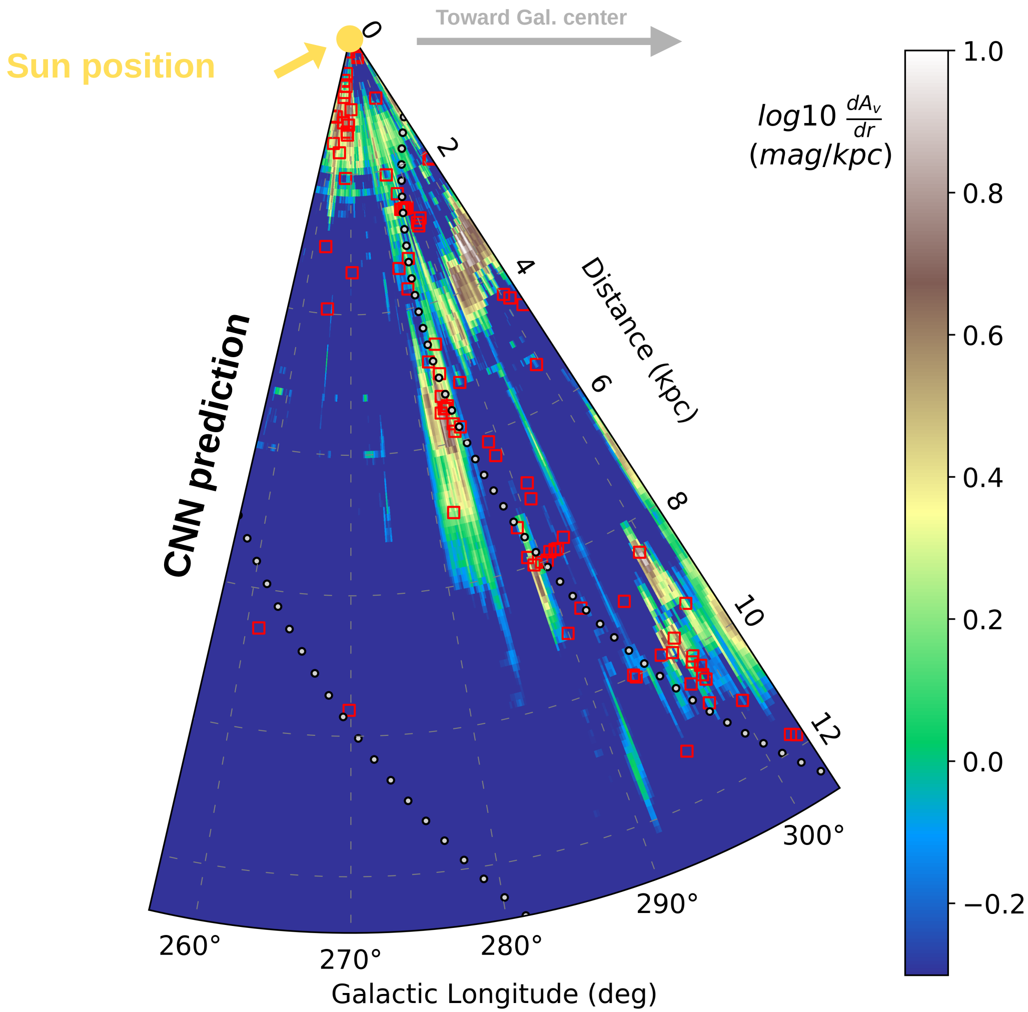


CNN prediction



- ✔ **Morphology**
- ✔ Contrast
- ~ Latitude artifacts
- ~ Tiling effect

CNN prediction: face-on view



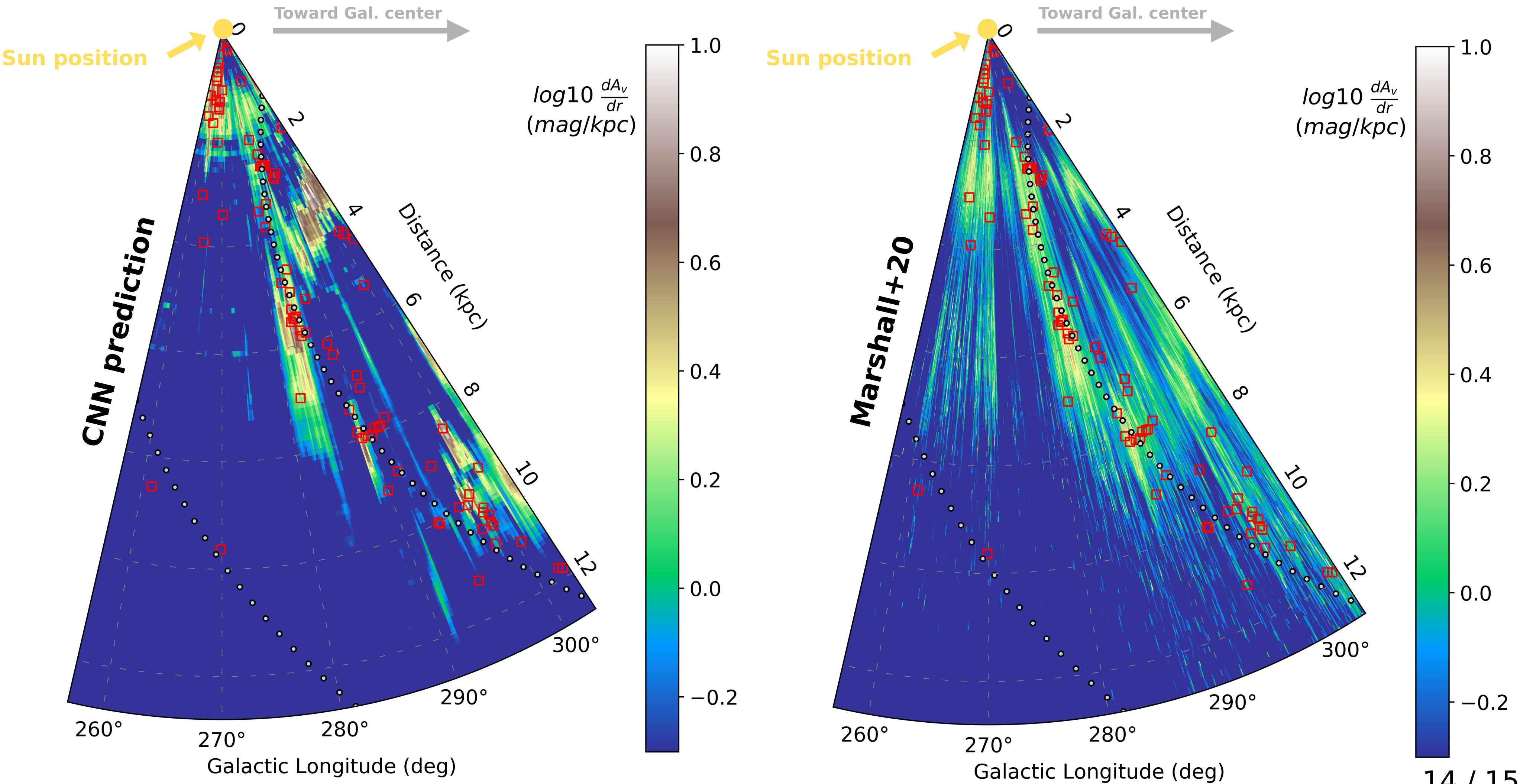
Red squares: HII regions, dense cloud tracer (Hou et Han 2014)
Grey dots: simple elliptical arm model

- ✔ Coherent structures between LOS
- ✔ Visible Carina arm tangent
- ✔ Several structures aligned with the arm
- ✔ Convincing patchy structures at d~10kpc

- ~ Nothing at high distance low longitude
  → Not enough stars ?

- ✘ Large artifact at 3-4 kpc, l=300

Maps comparison



Summary and perspectives

- **We identified that CNNs can be used to construct MW extinction maps**
  → constructed an architecture able to combine BGM and 2MASS/Gaia

- **Our solution is competitive with state of the art maps**
  → high range prediction with less distance artifact

The associated paper is in prep. → to be submitted to Astronomy and Astrophysics (A&A, Cornu et al. 2021)

A word on combining optical and infrared surveys without cross-match:
Several surveys can be added as independent input channels for the CNN!

- A lot of work remains on creating realistic Gaia diagrams to obtain convincing maps
- On modeled data, the CNN architecture showed its efficiency to combine Gaia and 2MASS

A special “Thank You”:

- To the Mésocentre Université de Franche Comté that already hosted 1500+ hours of GPU computation (Tesla V100) for this study.
- To the CompuPhys Master of the UFC that shared lighter resources (Quadro RTX 5000) that was used for easier CNN inference.