Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

SPECTRA-AI

Spectral Pattern Extraction for Classifying Transient Radiations in Astronomy with Artificial Intelligence

Alberto Garinei, Department of Engineering Science - Guglielmo Marconi University

Spoke 3 :

Sara Cutini - INFN PG

Francesco Longo - Università di Trieste - INFN

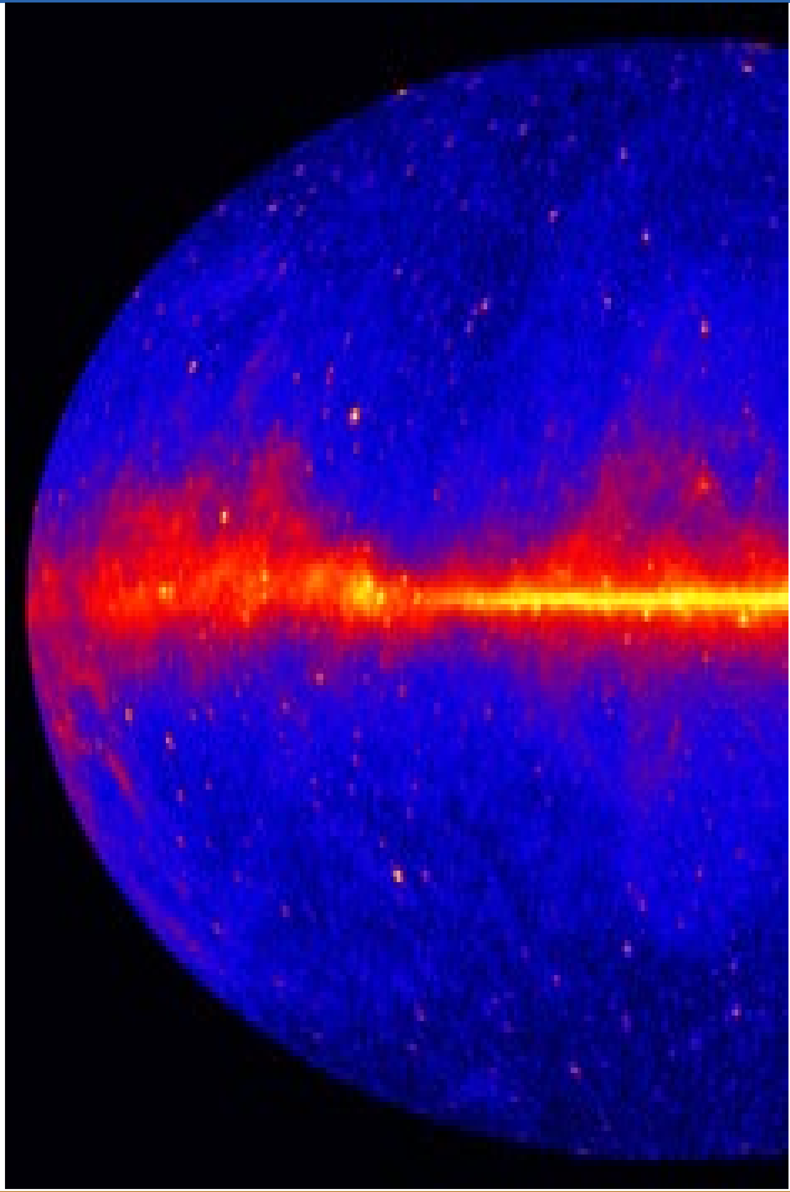
Spoke 3 Progetti Bandi a Cascata, 29/05/2025

Project Overview

The **SPECTRA-AI** project aims to innovate astronomical observation by applying Artificial Intelligence (AI) to identify and analyze transient astronomical sources using high-energy gamma ray data.

Transient sources, such as gamma ray bursts, supernovae, and solar flares, are crucial for understanding the high-energy universe.

We use data from high-energy gamma-ray experiments, in particular from the Fermi LAT.



Technical Objectives, Methodologies and Solutions

Technical objectives:

- Develop big data processing and visualization tools to manage the large volumes of data.

The Methodologies include the use of Deep Learning architectures for spatial and temporal data analysis:

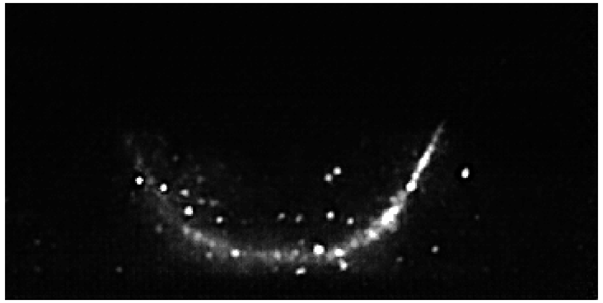
- Convolutional Neural Networks (CNN) for image analysis
- LSTMs for analyzing photon emission time series

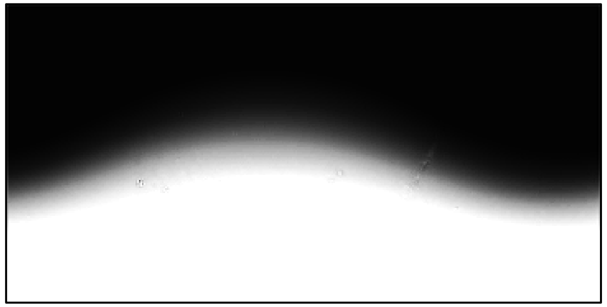
The solutions proposed are aimed at building a unified system capable of automatic detection and real-time classification of transient astronomical events.

Technical Objectives, Methodologies and Solutions

Raw Data Analysis:

- Fermipy package analysis (Python)
- Fermitools





With standard tools we get **Photon Counts** and **Exposure Map**, crucial to understand Average Flux from the Universe

↓

This is crucial to understand how the Universe "behaves"

Technical Objectives, Methodologies and Solutions

Universe Simulation and NN training:

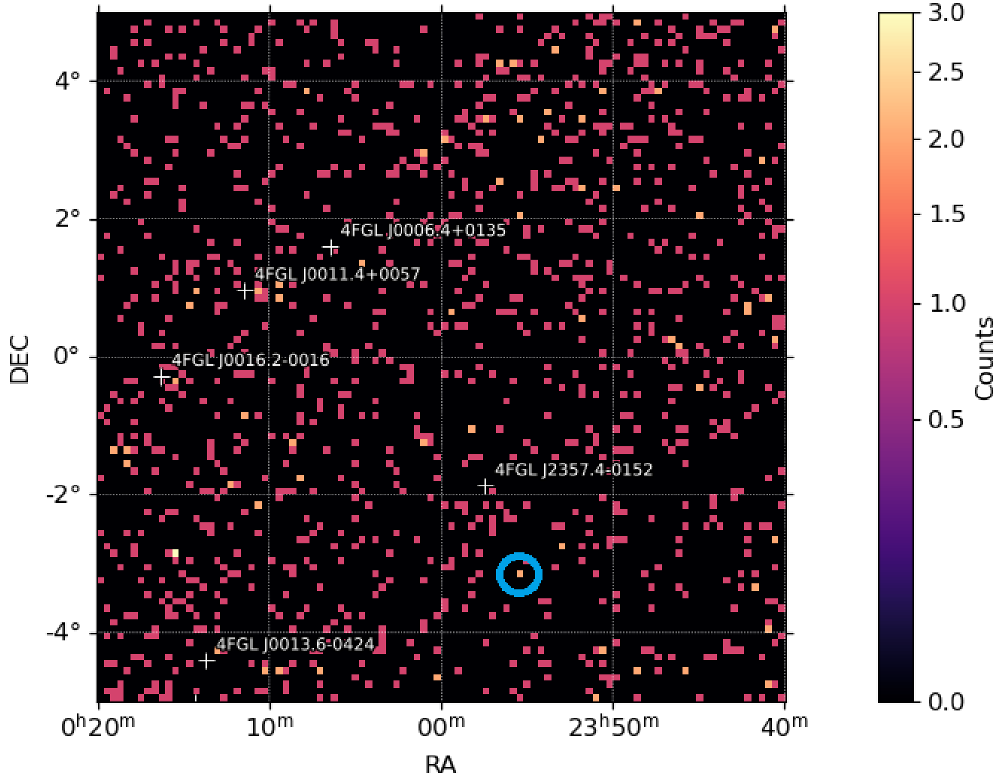
1. Catalog Model Preparation (*fermimodel*).
2. Simulation using gtobssim from NASA
3. Our Neural Network learns what average fluxes, given photon counts and LAT exposure look like

Technical Objectives, Methodologies and Solutions

Anomaly Detection

- **Pipeline Architecture**. Models:

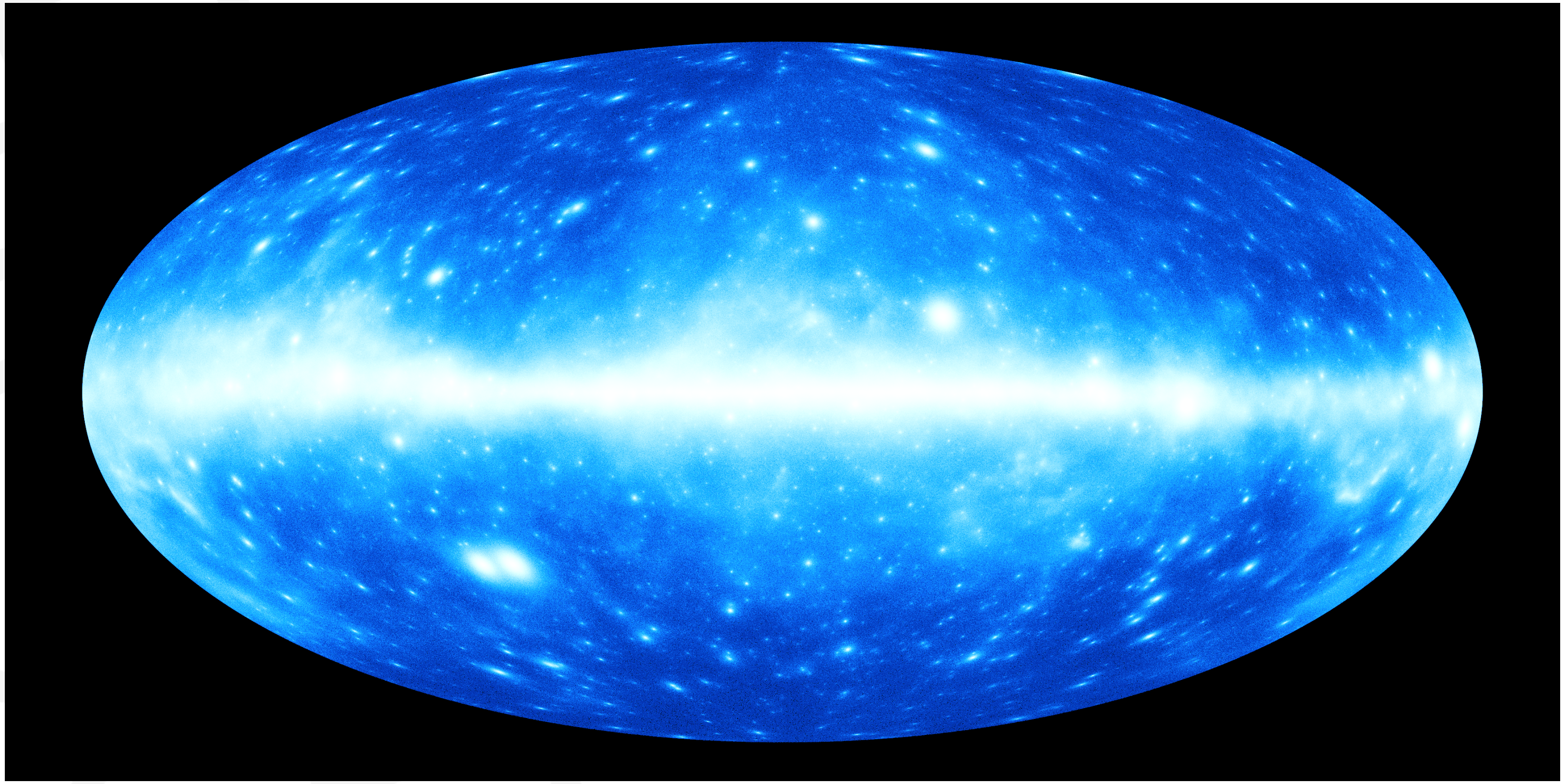
1. Autoencoder
2. ConvLSTM



Whenever **new data** is fed into the NN, the system automatically tells you whether that's **Anomaly Signal or not**.

Technical Objectives, Methodologies and Solutions

What a Ten Year-simulation looks like



Next Steps & Future Development

- **Different time scale**:
  Simulation carried out over different time windows will allow to spot different time scale anomalies
- **Different anomaly patterns**:
  Simulation of anomaly patterns to be tested against our model
- **Computational Cost Metrics**
  Develop metrics to assess the computational cost preserving model performance.
- **Dashboard**
  User friendly platform to carry out anomaly analysis

Finanziato dall'Unione europea
NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC
Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

GRAIL

Gamma-Ray Imaging with deep Learning

Alberto Garinei – Department of Engineering Science, Guglielmo Marconi University

Spoke 3 :

Fabio Gargano - INFN Bari

Spoke 3 Progetti Bandi a Cascata, 29/05/2025

Project Overview

- **Objective**:
Development of an AI-based system for onboard particle classification in space calorimeters.

- **Challenges**:

1. High data volume from calorimeters.
2. Computational limitations on satellites (power consumption < 5-10W).
3. Need for onboard data processing due to limited transmission capacity.

Key Methodologies and Achievements

1. **Data Reduction Techniques:**

- Implemented and evaluated **multiple dimensionality reduction techniques**:
  - Point Cloud, PCA (Principal Component Analysis), ICA (Independent Component Analysis), LDA (Linear Discriminant Analysis)
  - TSNE (t-Distributed Stochastic Neighbor Embedding), UMAP (Uniform Manifold Approximation and Projection), VAE (Variational Autoencoders), Clustering

- **Feature Importance with XGBoost** emerged as the most effective method: it ensures **high classification accuracy** in subsequent phases while preserving explainability of the selected features

- Assessed using reference metrics to balance data compression and information retention.

Key Methodologies and Achievements

2. Predictive Models Optimization:

- Tested and optimized **various machine learning models on the reduced datasets** obtained from all dimensionality reduction techniques:
  - CNN (Convolutional Neural Networks), XGBoost, Transformer, MLP (Multilayer Perceptron)
  - Logistic Regression, SVM (Support Vector Machine), Perceptron, GaussianNB (Gaussian Naïve Bayes)
  - Passive Aggressive Classifier, LDA (Linear Discriminant Analysis), QDA (Quadratic Discriminant Analysis)
- Evaluated based on reference metrics for **accuracy and computational efficiency**.
- While several models performed well, **XGBoost** was selected for final deployment due to its **high accuracy, fast inference time**, and its ability to provide **interpretable decision logic** through its tree-based structure

On-Edge Deployment and Real-Time Testing

- **Conversion to C for Embedded Execution:**
  - Dimensionality reduction and classification models **converted to C** for execution on embedded systems.
  - Developed a script that loads **pre-trained models,** receives **new calorimeter data** via serial, and returns **real-time predictions**.

- **Testing on Arduino MKR WAN 1310:**
  - Simulated a **low-power, resource-constrained environment.**
  - Achieved **classification in <1s,** meeting real-time **on-edge processing** requirements**.**

Next Steps & Future Development

- **Full-scale Model Training**

Train models on **100% of available data** using high-performance GPU servers, overcoming current computational limitations (models trained on ~1%).
This is expected to improve accuracy from **97% to 99%**.

- **On-Edge GPU Deployment**

Convert and test models for execution on **Jetson Orin Nano (NVIDIA)** to ensure efficient inference on low-power embedded GPUs.

- **Computational Cost Metrics**

Develop metrics to **assess the computational cost** to determine the **optimal number of features** to retain during dimensionality reduction, balancing information preservation with model performance.

Thanks!