



Rapporti Tecnici INAF INAF Technical Reports

Number	266
Publication Year	2023
Acceptance in OA@INAF	2023-02-20T08:49:13Z
Title	Progetto di tirocinio laurea triennale in Fisica: Rilevamento automatico di scariche atmosferiche registrate negli spettrogrammi CALLISTO tramite tecniche machine-learning/deep-learning
Authors	MARASSI, Alessandro; Perin, Nicola; Milan, Andrea; Longo, Francesco
Affiliation of first author	O.A. Trieste
Handle	http://hdl.handle.net/20.500.12386/33542 ; https://doi.org/10.20371/INAF/TechRep/266

Progetto di tirocinio laurea triennale in Fisica

Rilevamento automatico di scariche atmosferiche registrate negli spettrogrammi 'CALLISTO' tramite tecniche di machine-learning/deep-learning

Alessandro Marassi¹, Nicola Perin², Andrea Milan², Francesco Longo²

¹Istituto Nazionale di Astrofisica – Osservatorio Astronomico di Trieste

²Università di Trieste

Abstract

Un solar radio burst (SRB) è un' intensa emissione radio solare spesso correlata a un brillamento solare. Il rilevamento di SRBs può essere effettuato tramite radiometri e spettrometri terrestri. CALLISTO è uno spettrometro a basso costo progettato presso il Politecnico di Zurigo. Gli strumenti CALLISTO attivi in questo momento sono 67, distribuiti in tutto il mondo e permettono il monitoraggio delle emissioni radio solari 24h/7 tramite la rete e-Callisto <http://www.e-callisto.org/>. La Stazione Osservativa di Basovizza di INAF – Osservatorio Astronomico di Trieste è equipaggiata con tre spettrometri tipo CALLISTO operanti rispettivamente nelle bande VHF, UHF, L. È attivo anche un sistema sperimentale di rilevamento automatico di SRBs.

I radiospettri acquisiti dagli spettrometri sono purtroppo affetti da disturbi ed interferenze a radio frequenza (RFI) che a volte vengono interpretati dal sistema come burst/falsi positivi. Tra questi le scariche atmosferiche, rilevate dagli spettrometri come emissioni lineari su tutto lo spettro. Il loro riconoscimento ed eliminazione può contribuire in maniera importante a diminuire il numero di falsi positivi rilevati. Tra i vari possibili metodi di riconoscimento, tenuto anche conto del numero di eventi a disposizione per costruire un training set di numerosità adeguata, si è pensato di usare un sistema basato sul deep learning. Il lavoro effettuato nell'ambito del tirocinio, dopo l'acquisizione delle necessarie conoscenze di base (solar radio burst, documentazione CALLISTO, reti neurali, deep-learning) ha consentito la costruzione di training/test set di eventi ed una sperimentazione pilota su dati limitati con sviluppo ed esecuzione di modelli Keras su una virtual machine Virtualbox Ubuntu64 18.04 LTS, precedentemente installata ed attrezzata con librerie TensorFlow/Keras, propedeutica ad una sperimentazione completa su una Deep-Learning Machine fisica dedicata (CPU Xeon + GPU NVIDIA Quadro P4000).

Introduzione

Un solar radio burst (SRB) è un' intensa emissione radio solare spesso correlata a un brillamento solare; è uno degli eventi meteorologici spaziali che possono influenzare la ionosfera terrestre e la propagazione dei segnali, le comunicazioni wireless, le reti elettriche e i sistemi di navigazione. I principali eventi meteorologici spaziali come i brillamenti solari (solar flares) e le espulsioni di massa coronale (coronal mass ejections – CME) sono solitamente accompagnati da emissioni radio utilizzabili per previsioni meteorologiche spaziali in tempo reale [White, 2007]; il rilevamento di SRBs può essere molto utile ad esempio nella mitigazione di effetti diretti sui sistemi satellitari di navigazione (global navigation satellite systems - GNSS) e come precursore di eventi quali tempeste geomagnetiche intense con potenziali importanti effetti sui sistemi tecnologici terrestri quali geomagnetically induced currents (GIC) su reti elettriche e strutture interrate come gasdotti/oleodotti.

Il rilevamento di SRBs può essere effettuato tramite radiometri e spettrometri terrestri.

CALLISTO (vedi fig.1) è uno spettrometro a basso costo progettato presso il Politecnico di Zurigo [Benz et al. 2005]. Gli strumenti CALLISTO attivi in questo momento sono 67, distribuiti in tutto il mondo e permettono il monitoraggio delle emissioni radio solari (vedi esempio in fig.2) 24h/7 tramite la rete e-Callisto <http://www.e-callisto.org/>



Fig. 1: Spettrometro CALLISTO

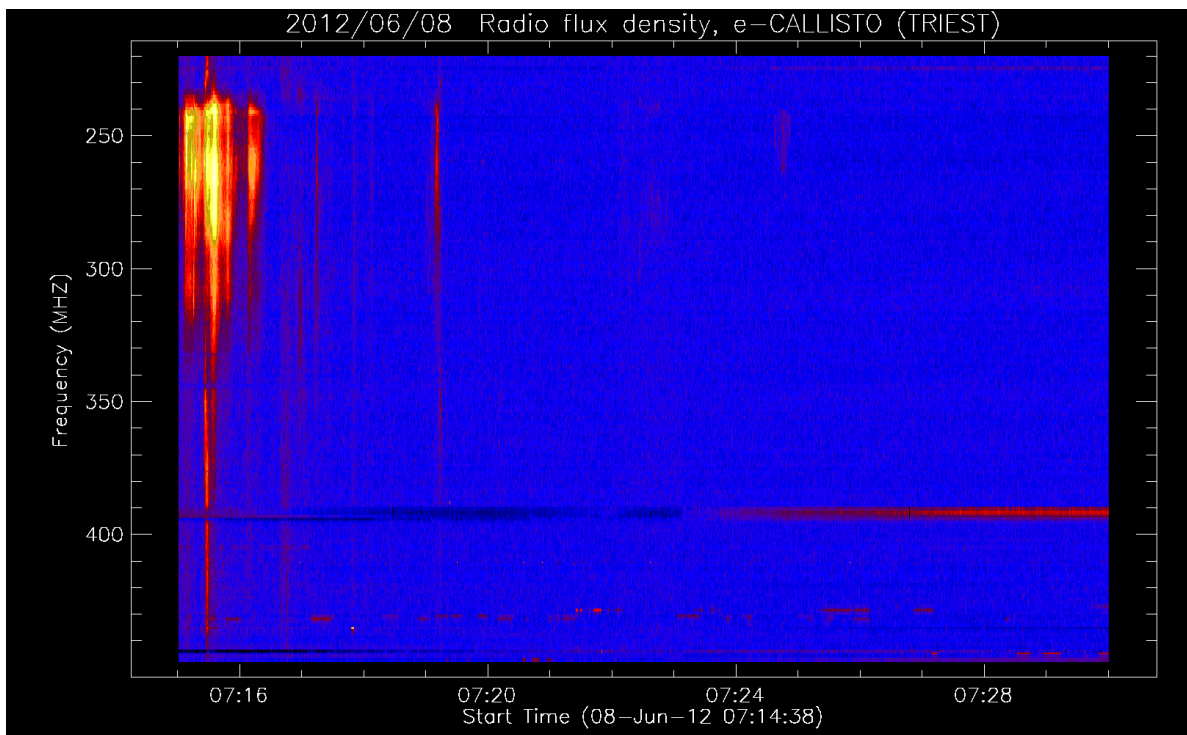


Fig. 2: Burst rilevati dalla stazione Callisto di INAF - Osservatorio Astronomico di Trieste

La Stazione Osservativa di Basovizza di INAF – Osservatorio Astronomico di Trieste e' equipaggiata con tre spettrometri tipo CALLISTO <https://e-callisto.org/> operanti rispettivamente nelle bande di frequenza 45-80 MHz, 220-420 MHz e banda L (905-1730 MHz) con un periodo di campionamento di 0.25s [Marassi and Monstein, 2022].

E' attivo anche un sistema sperimentale di rilevamento automatico di SRBs basato sullo *spectrogram intensity profile plot* (vedi fig.3) di ogni spettrogramma (200 frequenze x 3600 campioni temporali). Se questo supera la media aritmetica calcolata su tutte le 3600 colonne, il sistema rileva un burst.

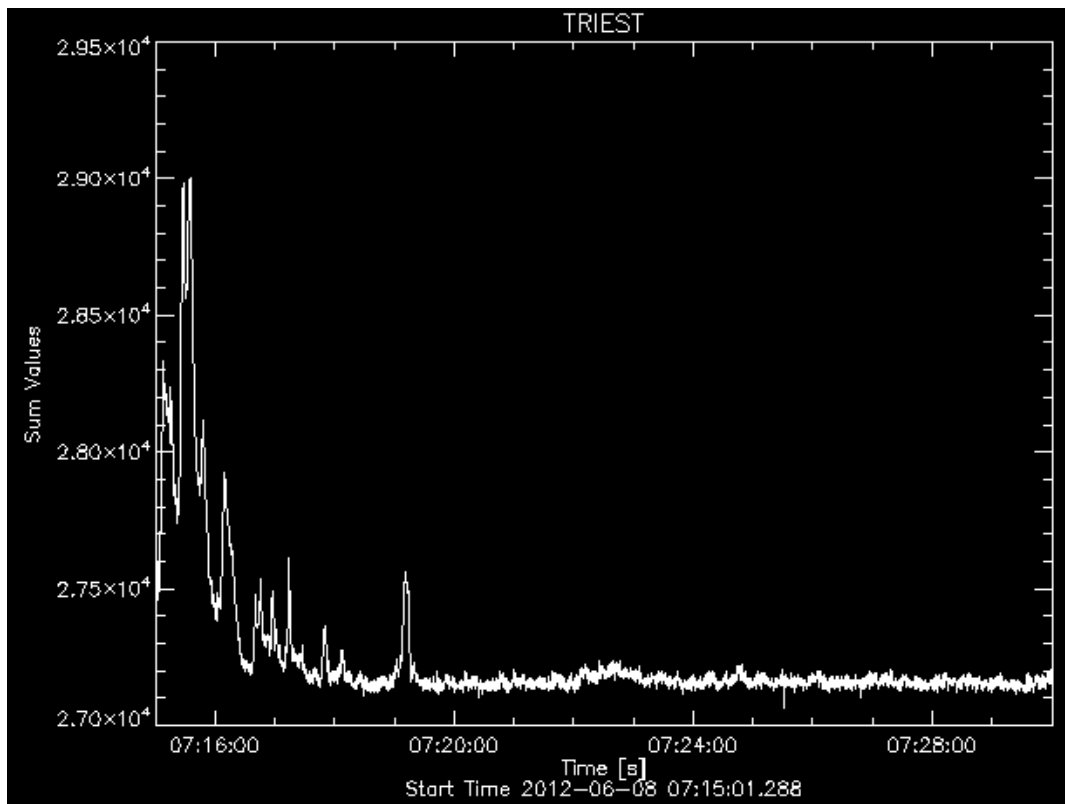


Fig. 3: Spectrogram intensity profile plot in the frequency band 220–440 MHz

I radiospettri acquisiti dagli spettrometri sono purtroppo affetti da disturbi ed interferenze a radio frequenza (RFI) che a volte vengono interpretati dal sistema come burst/falsi positivi. Tra questi le scariche atmosferiche che costituiscono una fonte importante di rumore elettromagnetico; sono rilevati dagli spettrometri come emissioni lineari su tutto lo spettro durante manifestazioni temporalesche locali e remote (vedi fig.4).

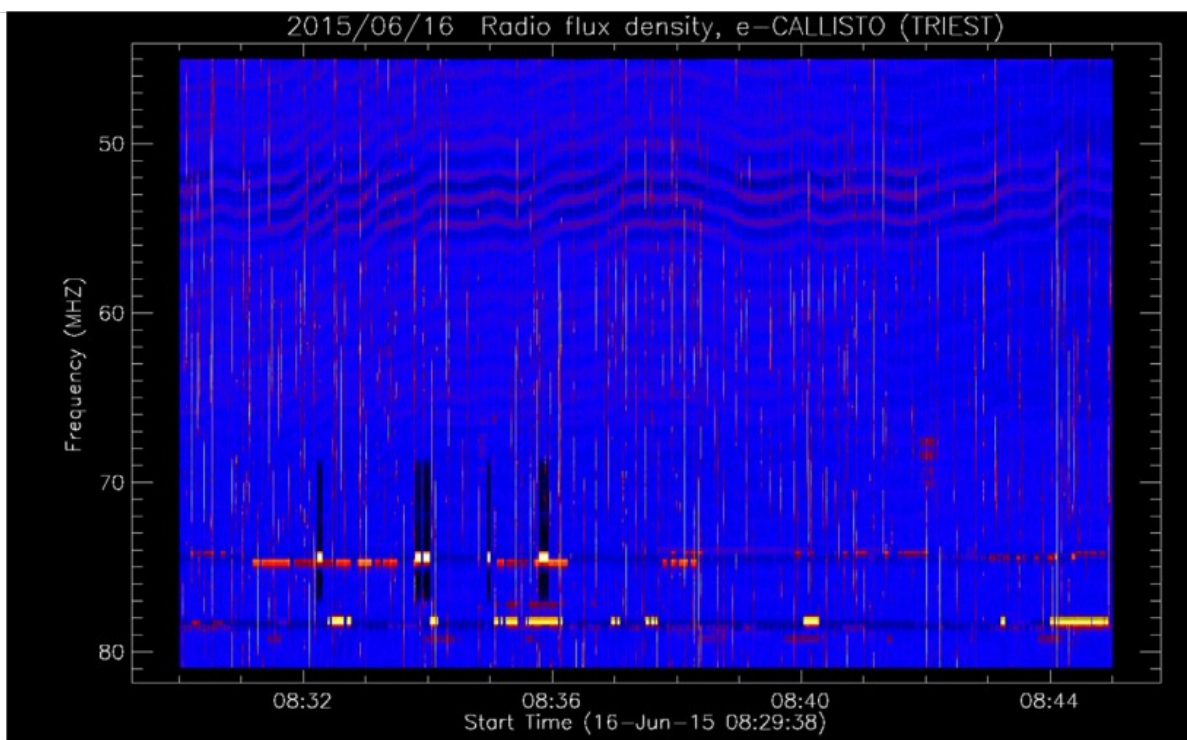


Fig. 4: Scariche atmosferiche visibili nello spettro 45-80 MHz

Il loro riconoscimento ed eliminazione puo' contribuire in maniera importante a diminuire il numero di falsi positivi rilevati.

Tra i vari metodi di rilevamento automatico di pattern vi sono quelli basati su tecniche di machine-learning e deep-learning, oggetto di questo progetto formativo.

Intelligenza artificiale

L'IA è quella disciplina, appartenente all'informatica, che studia i fondamenti teorici, le metodologie e le tecniche che permettono di progettare sistemi hardware e sistemi di programmi software capaci di fornire all'elaboratore elettronico delle prestazioni che, a un osservatore comune, sembrerebbero essere di pertinenza esclusiva dell'intelligenza umana [Somalvico 1987].

In questo ambito si possono distinguere due teorie distinte:

Intelligenza Artificiale Forte, secondo cui le macchine sono in grado di sviluppare una coscienza di sé, che studia sistemi in grado di replicare l'intelligenza umana;

Intelligenza Artificiale Debole, la quale ritiene possibile sviluppare macchine in grado di risolvere problemi specifici senza avere però coscienza delle attività svolte. Obiettivo di tale teoria non è dunque quello di realizzare macchine dotate di un'intelligenza umana, ma di avere sistemi in grado di svolgere una o più funzioni umane complesse.

Le applicazioni di Intelligenza Artificiale Debole possono essere molteplici e riguardare diversi ambiti quali la visione artificiale, il riconoscimento automatico di pattern/immagini/parole, sistemi text-to-speech, la comprensione del linguaggio naturale.

Lo sforzo di automatizzare task propri dell'intelletto umano include diversi approcci tra i quali i **sistemi esperti**, basati su regole logiche e manipolazione di simboli, il **machine learning** in cui il sistema, avuti in input dati da elaborare (features relative al problema da risolvere come ad esempio immagini di caratteri scritti a mano) e le risposte desiderate (tag associati a ciascuna immagine), ricava autonomamente le regole da applicare, il **deep learning** e' un sottoinsieme del machine learning in cui un modello impara ad eseguire dei task di classificazione direttamente dai dati (immagini, testo, suono). Sia il machine learning che il deep learning vengono di solito implementati con un'architettura tipo rete neurale. L'aggettivo 'deep' si riferisce al numero di layer della rete; mentre le reti neurali tradizionali hanno tipicamente 2 o 3 layer, una rete 'deep' ne puo' avere centinaia. La tecnologia 'deep learning' e' stata resa possibile dalla disponibilita' di dataset taggati massivi (come ImageNet o PASCAL VoC), dall'aumento della potenza computazionale (GPU ad alte prestazioni capaci di ridurre i tempi di calcolo da settimane ad ore) e da modelli pre-addestrati da esperti da riaddestrare per eseguire nuovi task usando una tecnica chiamata **transfer learning** (AlexNet e' stato addestrato usando 1.3 milioni di immagini ad alta risoluzione per riconoscere 1000 oggetti ed un suo ulteriore addestramento (transfer learning) puo' essere effettuato con dataset molto piu' piccoli).

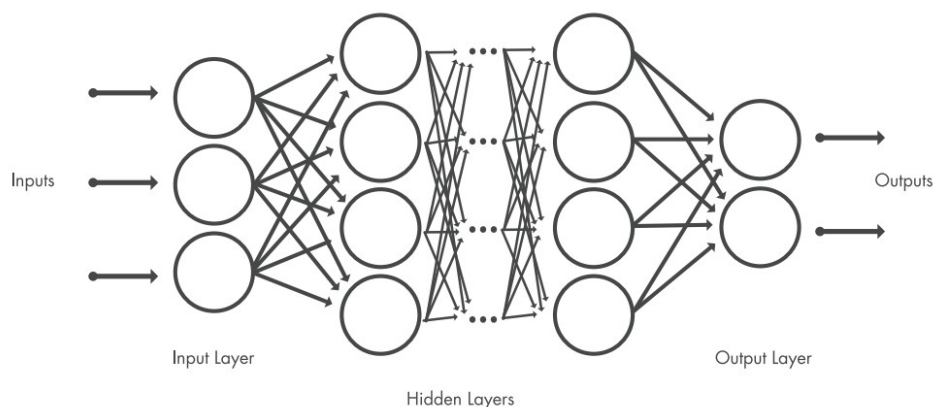


Fig. 5: Neural network

Una rete neurale è costituita da un certo numero di strati (layer) di elementi semplici (neuroni) operanti in parallelo in analogia al funzionamento dei sistemi nervosi biologici. Vi è sempre un input layer, diversi strati cosiddetti 'nascosti' (hidden layers), ed un output layer che usano l'output del layer precedente come input (vedi fig.5).

Deep learning e riconoscimento di immagini

E' possibile addestrare una rete a riconoscere automaticamente immagini appartenenti ad N classi tramite un training set di campioni dotati di label. La rete può imparare autonomamente ad associare feature specifiche alle diverse classi senza alcun intervento umano.

Un algoritmo di deep learning molto usato nel riconoscimento di immagini è rappresentato dalla cosiddetta *convolutional neural network* (CNN o ConvNet) in cui si possono distinguere *feature detection layers*, dedicati alla selezione delle features tramite filtri convolutivi nonché alla riduzione del numero di parametri e *classification layers* dedicati alla classificazione; l'output layer sarà in questo caso un vettore di dimensione N, contenente le probabilità di classificazione di ciascuna delle N classi.

Deep-learning workstation e virtual machine

In previsione della sperimentazione di tecniche ed algoritmi di deep-learning per il riconoscimento di pattern/immagini, si era in precedenza provveduto all'installazione di una *deep-learning workstation* dotata di CPU Xeon e GPU NVIDIA Quadro P4000, basata su Ubuntu64 18.04 e completa di Python scientific suite (Numpy e SciPy), CUDA drivers e cuDNN, TensorFlow, Theano e Keras.

Per poter lavorare anche fuori sede su macchine diverse durante i lockdown, si e' inoltre installata e configurata una *deep-learning virtual machine* basata su Virtualbox con la stessa configurazione di base della deep-learning workstation fisica. Pur con le ovvie limitazioni in termini prestazionali, la *deep-learning virtual machine* si e' dimostrata molto utile anche nell'ambito del tirocinio consentendo la sperimentazione concreta delle tecniche di deep-learning senza avere a disposizione una macchina fisica dedicata.

Descrizione del progetto/attività svolta

Il lavoro effettuato nell'ambito del tirocinio, dopo l'acquisizione delle necessarie conoscenze di base (definizione e caratterizzazione di solar radio burst, documentazione CALLISTO, formato e tools di visualizzazione in falsi colori di dati FITS, reti neurali, deep-learning: concetti, metodi e librerie quali TensorFlow e Keras), codifica di script Python per la visualizzazione ed elaborazione dei dati FITS CALLISTO con l'utilizzo di varie librerie (SunPy, Matplotlib, Numpy, Cv2, SciPy), e' proseguito con l'individuazione ed estrazione dei file contenenti numeri rilevanti di scariche atmosferiche dal database web della stazione Callisto di Trieste. Cio' sulla base dell'evidenza di dati ricavati in precedenza (e memorizzati in file in formato testo che riportano data e ora degli eventi) dal sistema sperimentale di rilevamento automatico di SRBs attivo da luglio 2015. Di seguito estratto del file HF.dat contenente tutti gli eventi rilevati automaticamente dal sistema nella banda 45-80 MHz. Imponendo una soglia minima al numero di eventi/data si sono trovate le giornate con un grande numero di falsi positivi dovuti per l'appunto alle scariche atmosferiche.

2016/08/22
6:49:23
9:1:58
9:11:14
17:26:51

Estratto del file HF.dat con ora di 4 eventi rilevati in data 22 agosto 2016 nella banda 45-80 MHz

E' stato quindi valutato l'andamento qualitativo tempo/frequenza degli spettri in corrispondenza delle scariche atmosferiche, rilevando in particolare che:

- nella banda 45-80 MHz si ha il massimo di intensità tra i 45 e i 50 MHz e poi l'andamento è discendente;
- nella banda 220-420 MHz c'è un minimo in corrispondenza dell'intervallo 325-360, seguito da un picco intorno ai 390MHz, mentre il resto dello spettro è circa costante;
- per quanto riguarda la lightcurve, ci sono dei picchi molto stretti di intensità in corrispondenza di ogni singola scarica, intervallati da un fondo costante;
- L'andamento dei singoli picchi al variare del canale non è costante, anche se varia lentamente. Isolando un picco, si può osservare che ad esempio scende fino a scomparire per poi ricomparire. Osservando lo spettro corrispondente, ciò coincide con il minimo di intensità in funzione della frequenza. A volte il picco principale è preceduto o seguito da altri minori, a distanza di pochi secondi.

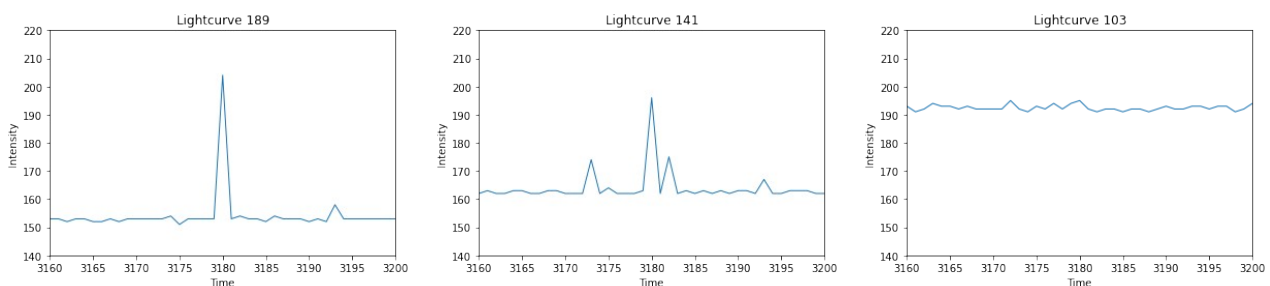


Fig.6: Andamento intensità'/tempo di uno stesso evento a tre diverse frequenze in banda 220-420 MHz

In fig.6 immagini di uno stesso picco visto in tre diverse frequenze delle 200 registrate ogni 0.25s: nella prima è isolato, nella seconda è accompagnato da altri e nella terza è praticamente assente.

Sono state calcolate medie e varianze dell'insieme di valori che ogni picco di intensità assume al variare delle 200 frequenze per confrontare picchi diversi alle varie frequenze e rilevare eventuali comportamenti simili. La cosa, applicata su un campione di numerosità limitata (400), non ha permesso di trovare informazioni utili per un loro successivo impiego nel riconoscimento degli eventi.

Immaginando di voler rilevare le scariche atmosferiche registrate negli spettri con tecniche di 'deep-learning', dovremmo poter disporre di un congruo numero di campioni (training-set) da dare in pasto ad una piattaforma di 'deep-learning' in un formato quale una matrice bidimensionale di dimensione 10x200 (in questo caso 10x0.25s per 200 frequenze) contenente solo un evento. La macchina prima deve venire addestrata a riconoscere questi eventi e può quindi essere usata per nuovi eventi (test-set). E' necessario quindi procedere con la costruzione di un training-set e test-set di eventi.

A scopo meramente didattico, il primo test ha considerato 800 campioni (matrici 10x200) relative a 'fette' di spettrogrammi di dimensioni 10x200 contenenti scariche atmosferiche, rispettivamente 400 nella banda 220-420 e altri 400 nella banda 45-80 MHz, tutti con label 1 (= 'fulmine').

Sono stati quindi creati, per ciascuna delle due bande di frequenza, un training-set con label 0 (= 'non fulmine') e un test-set di spezzoni di spettro di dimensioni uguali a quelli già costruiti per i fulmini (10x200), ma non contenenti eventi.

I 400 campioni contenenti eventi ed i 400 campioni non contenenti eventi sono stati a loro volta suddivisi in training-set e test-set nella proporzione 80:20, quindi 640 campioni train e 160 campioni test.

Si è proceduto quindi con la costruzione di una rete con due layer di tipo *Dense*, funzione di attivazione rispettivamente *relu* e *sigmoid*, loss function *binary_crossentropy*, considerato il problema di classificazione tra due classi:

```
network = models.Sequential()
network.add(layers.Dense(512, activation='relu', input_shape=(200 * 10,)))
network.add(layers.Dense(2, activation='sigmoid'))
```

```
network.compile(optimizer='rmsprop',
                 loss='binary_crossentropy',
                 metrics=['accuracy'])
```

effettuando uno reshaping dei dati di input (immagini 200x10 di tipo uint8 con valori nell'intervallo [0, 255]) in tensori Numpy, rispettivamente di shape (640, 2000) (training-data) e (160, 2000) (test-data), formattati come 'float32' con valori compresi tra 0 e 1.

```
train_data = train_data.reshape((640, 200 * 10))
train_data = train_data.astype('float32') / 255.0
test_data = test_data.reshape((160, 200 * 10))
test_data = test_data.astype('float32') / 255.0
```

codifica delle label per categoria
from keras.utils import to_categorical

```
train_labels = to_categorical(train_labels)
test_labels = to_categorical(test_labels)
```

addestramento della rete

```
network.fit(train_data, train_labels, epochs=5, batch_size=4)
```

esecuzione della rete sul test-set

```
test_loss, test_acc = network.evaluate(test_data, test_labels)
```

I risultati per gli eventi relativi alla banda 45-80 MHz sono stati i seguenti:

```
Training:
Epoch 1/5 loss: 0.7044 - accuracy: 0.6395
Epoch 2/5 loss: 0.5033 - accuracy: 0.7241
Epoch 3/5 loss: 0.2666 - accuracy: 0.9216
Epoch 4/5 loss: 0.1468 - accuracy: 0.9498
Epoch 5/5 loss: 0.0630 - accuracy: 0.9749
Test:
loss: 0.0156 - accuracy: 1.0000
```


con tempi di esecuzione per l'addestramento dell'ordine di 20 secondi.

Si e' quindi applicata la rete addestrata a spettrogrammi contenenti anche burst, considerando una 'sliding-window', finestra ampia 10 unita' di tempo, che scorre sullo spettrogramma (contenente anche burst) una unita' di tempo alla volta; ogni spettrogramma, di dimensioni 3600x200, implica 3600-10 finestre diverse, che iniziano all'istante 0 e terminano all'istante 3599 (0-9, 1-10, 2-11, ..., 3590-3599). Specificatamente e' stato considerato lo spettrogramma delle 12.00 del 6 settembre 2017 (in fig.7).

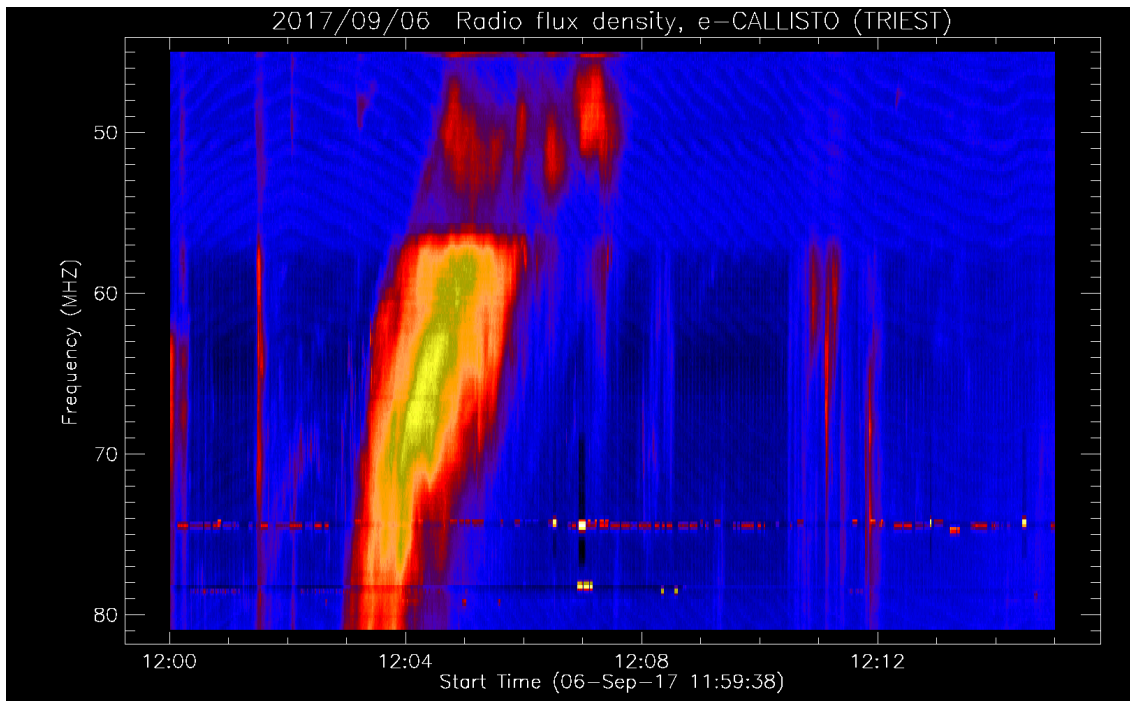


Fig. 7: Spettro contenente bursts

Il risultato dell'applicazione della rete addestrata e' stato:

loss: 0.6766 - accuracy: 0.8638

La rete quindi sembra classificare correttamente i burst come 'non fulmini', anche se con funzione di loss piuttosto alta ed accuratezza bassa. Cio' e' sicuramente imputabile anche alla scarsa numerosita' del training-set considerato.

La principale differenza tra i burst e le scariche atmosferiche e' la durata. Il fulmine si esaurisce nelle 10 unita' temporali della fetta di spettrogramma che lo contiene e quindi e' evidente la differenza tra i valori agli estremi (piu' bassi) e il centro. I burst invece hanno una durata molto maggiore e i 10 valori di intensita' sono molto simili tra loro, come nel fondo usato per l'addestramento.

E' stato fatto un tentativo di addestrare una rete al riconoscimento di burst sulla base di features tra cui media e deviazione standard. Sono state effettuate diverse prove, anche variando la struttura della rete (numero di neuroni e di hidden layer) e le funzioni di attivazione, ma in nessun caso la rete e' riuscita a convergere.

Conclusioni

Durante il progetto di tirocinio abbiamo affrontato con risultati positivi il riconoscimento di scariche atmosferiche in spettrogrammi CALLISTO con un approccio deep-learning

supervisionato: approfittando della relativa numerosita' delle scariche atmosferiche/fulmini e' stato possibile costruire un training-set ed un test-set utilizzati per addestramento e test.

La rete addestrata sembra in grado di distinguere correttamente i burst come 'non fulmini', anche se con funzione di loss piuttosto alta ed accuratezza bassa. E' chiaramente necessario aumentare la numerosita' del training-set per ottenere risultati migliori.

Il riconoscimento dei burst (meno numerosi e con caratteristiche piuttosto variabili) dal fondo rimane un problema aperto.

In letteratura si trovano varie proposte di sistemi per il rilevamento automatico di radio burst basati su varie tecniche: uso di trasformate di Radon [Lobzin et al. 2009,2010; Bonnin et al. 2011], di Hough [Carley et al. 2015; Zhang et al. 2018], mathematical morphology [Jones and Richards 2014], Hidden Markov Model (HMM) [Gunes and Erdol 2006], adaptive constant-false-alarm rate (CFAR like detector) [Salmane et al. 2018]. Alcuni autori hanno proposto approcci tipo deep-learning [Chen et al. 2015, Guo et al. 2022, Scully et al. 2021], ma tali approcci non sono risolutivi e si scontrano con la necessita' di molti dati attualmente non disponibili.

In questa fase potrebbe essere tentato un approccio machine-learning tradizionale una volta identificate delle feature significative da usare in 'supervised learning' con classificatori tipo ANN.

L'approccio impiegato dal punto di vista didattico con l'utilizzo di una virtual machine Virtualbox Linux Ubuntu, completa di librerie Tensorflow e Keras, ha consentito di sperimentare le tecniche di deep-learning con un numero limitato di campioni su un comune netbook, anche senza avere a disposizione una macchina fisica attrezzata con acceleratori hardware tipo GPU.

Bibliografia

- Benz, A.O., Monstein, C., Meyer, H., 2005. Callisto a new concept for solar radio spectrometers. *Sol. Phys.* 226, 143–151. <https://doi.org/10.1007/s11207-005-5688-9>
- Bonnin X, Abouadarham J, Fuller N, Renie C, Perez-Suarez D, Gallagher P, Higgins P, Krista L, Csillaghy A, Bentley R. 2011. Automated detection and tracking of solar and heliospheric features in the frame of the European project HELIO. SF2A-2011: Proc. Annu. Meeting French Soc Astron Astrophys 373: 377.
- Carley EP, Reid H, Vilmer N, Gallagher PT. 2015. Low frequency radio observations of bidirectional electron beams in the solar corona. *A&A* 581: A100
- Chen, Zhuo & Ma, Lin & xu, Long & Weng, Ying & Yan, Yihua. (2015). Multimodal Learning for Classification of Solar Radio Spectrum. 1035-1040. 10.1109/SMC.2015.187.
- Chollet F 2018. Deep Learning with Python. Manning Publications Co. Shelter Island, NY
- Gunes T, Erdol N 2006. HMM based spectral frequency line tracking: Improvements and new results. In: Acoustics, Speech and Signal Processing, 2006, ICASSP 2006 Proceedings. 2006 IEEE International Conference on, 2, IEEE.
- Guo, J. C., Yan, F. B., Wan, G., Hu, X. J., & Wang, S. (2022). A deep learning method for the recognition of solar radio burst spectrum. *PeerJ. Computer science*, 8, e855. <https://doi.org/10.7717/peerj-cs.855>
- Jones J, Richards GP. 2014. Automated recognition of type III solar radio bursts using mathematical morphology. In: Advanced Maui Optical and Space Surveillance Technologies Conference.

- Lobzin VV, Cairns IH, Robinson PA, Steward G, Patterson G. 2009. Automatic recognition of type III solar radio bursts: automated radio burst identification system method and first observations. *Space Weather* 7(4).
- Lobzin VV, Cairns IH, Robinson PA, Steward G, Patterson G. 2010. Automatic recognition of coronal type II radio bursts: the automated radio burst identification system method and first observations. *ApJ* 710(1): L58.
- Marassi, A. and Monstein, C., 2022. Trieste CALLISTO station setup and observations of solar radio bursts. *Advances in Space Research*, vol. 69, no. 6, pp. 2589–2600, 2022. doi:10.1016/j.asr.2021.12.043.
- Scully, Jeremiah & Flynn, Ronan & Carley, Eoin & Gallagher, Peter & Daly, Mark. (2021). Type III solar radio burst detection and classification: A deep learning approach.
- Somalvico, M 1987. *L'Intelligenza Artificiale*. Rusconi Editore, Milano.
- White, S.M., 2007. Solar radio bursts and space weather. *Asian J. Phys.* 16, 189–207