

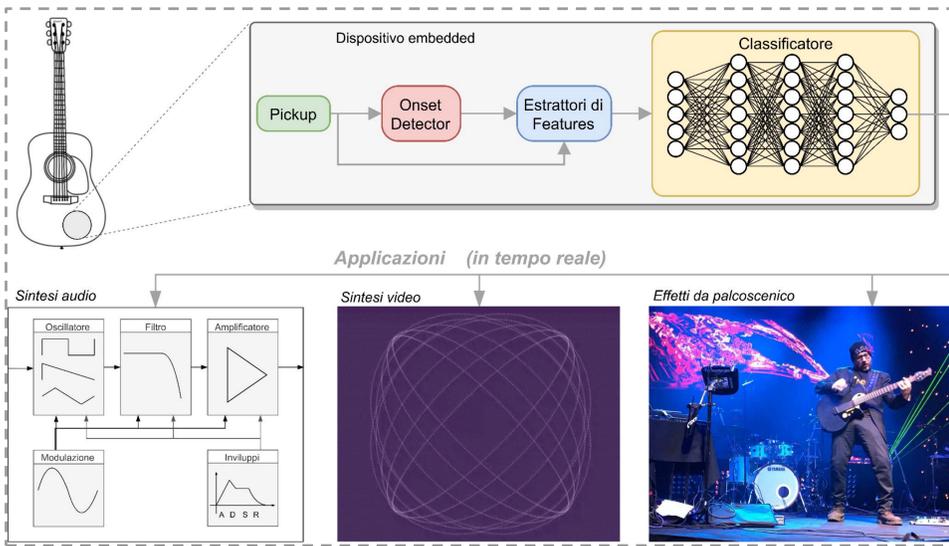
RICONOSCIMENTO IN TEMPO REALE DI TECNICHE ESPRESSIVE PER CHITARRA SU EMBEDDED COMPUTERS

Domenico Stefani

Introduzione

Le innovazioni nel campo dell'elettronica analogica degli ultimi decenni dello scorso secolo hanno portato alla creazione di una serie di **strumenti-controller** basati sulla chitarra che potevano produrre un'ampia gamma di suoni **controllando dei sintetizzatori audio**. Nonostante i vari miglioramenti apportati negli anni a questi sistemi, essi sono rimasti **limitati al tracciamento della frequenza e intensità nel tempo** delle note suonate, mancando di considerare le sottili **tecniche espressive** generalmente usate dai chitarristi per mutare il suono dello strumento. La mia ricerca si basa sulle tecniche più moderne di **deep learning** per il riconoscimento in **tempo reale** della tecnica espressiva usata su di una chitarra. Particolare attenzione è dedicata a proporre implementazioni su dispositivi embedded per poter permettere la creazione di strumenti intelligenti.

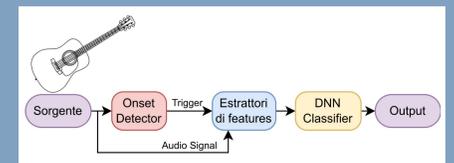
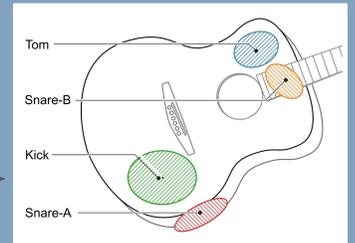
Progresso e Risultati



Primi Studi

Creazione di un dataset:

- 12 tecniche (4 percussive, 8 sulle corde)
- Percussive
- Pitched:
 - Plettro sulla buca
 - Plettro verso il ponte
 - Armonici naturali
 - Bending
 - Vibrato
 - Staccato
 - Palm-mute
 - Hammer-on
- 5 chitarristi, 5 diverse chitarre (oltre 20mila note in totale)



Sviluppo di una pipeline di classificazione monofonica:

1. Onset Detector
2. Estrattori di features timbriche
3. Classificatore neurale

Esecuzione su **Raspberry PI 4** con **Elk Audio OS**.

Latenza target di **~30ms**.

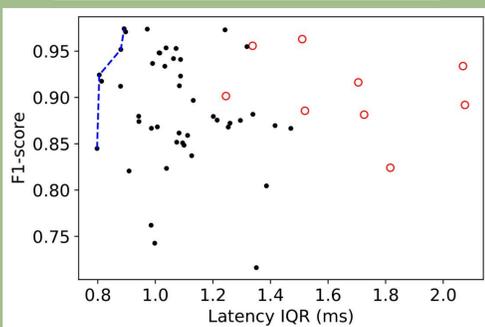
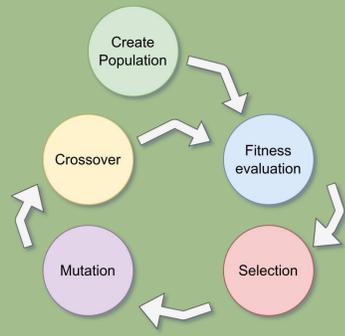
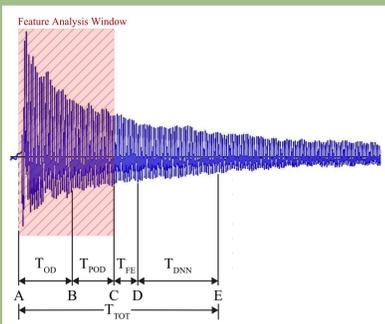


Raffinamento di Onset Detection ed estrazione delle features

L'ottimizzazione dei vari stadi della classificazione parte dall'Onset Detection.

Le performances vanno ottimizzate in termini di:

- **Accuratezza:** ridurre gli errori di individuazione (falsi positivi e negativi)
- **Latenza:** ridurre l'intervallo di tempo medio tra l'onset reale e la sua individuazione. Ridurre la variabilità della latenza permette un miglior allineamento della finestra di analisi delle features.



Altre ottimizzazioni

Uso di algoritmi di **selezione automatica** delle features più significative.

Ulteriori ottimizzazioni mostrano risultati più accurati, con una latenza media inferiore (~20 millisecondi)

Table 1: Task-A results (Percussive vs Pitched).

	Precision	Recall	F1-score
Macro avg.	97.5%	96.1%	96.8%
Accuracy	99.2%		

Table 2: Task-B results (4 percussive vs all pitched).

	Precision	Recall	F1-score
Macro avg.	92.8%	95.7%	94.2%
Accuracy	99.1%		

Table 3: Task-C (4 percussive and 4 pitched).

	Precision	Recall	F1-score
Macro avg.	63.6%	57.7%	59.8%
Accuracy	56.5%		

Table 4: Average latency of each step of the classification pipeline and total.

	Onset Detection Latency	post-Onset delay	Feature Extraction Latency	Inference Time	TOT
	19.00ms	7.77 ms	0.78 ms	3.15 ms	30.70 ms

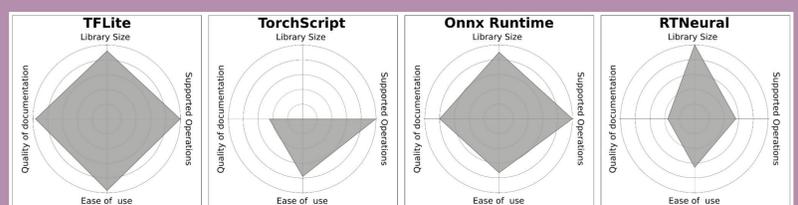
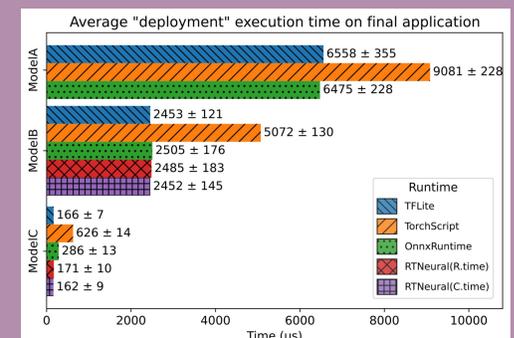


Comparazione di quattro librerie di inference per Embedded Deep Learning

- **Librerie:** 4 diverse librerie di inference per embedded devices (C++).
- **Task:** classificazione real-time di 8 tecniche espressive per chitarra.
- **Modelli:** 3 modelli di deep learning da 63 mila a oltre 2 milioni di parametri.

★ Le soluzioni commerciali **non-specializzate** sono molto flessibili e possono essere utilizzate per audio real-time.

★ Le soluzioni specializzate sono più leggere e minimaliste, utili dove la flessibilità non è necessaria.



Lavori Futuri

I risultati ottenuti fino ad ora hanno aperto alcune possibili linee di ricerca per il futuro qui elencate:

- **Miglioramento** della classificazione monofonica (accuratezza e latenza).
- Conduzione di **test tecnici e percettivi** per verificare l'accuratezza del sistema e l'impercettibilità del ritardo tra le note suonate dal musicista e i suoni sintetici generati secondo la tecnica classificata.
- Estensione alla **classificazione polifonica** della tecnica espressiva con trasduttore esafonico
- **Accelerazione hardware** come Tensor Processing Unit (TPU) per velocizzare la classificazione o permettere l'uso di modelli più complessi.
- **Transfer-learning** di modo da adattare un modello generico ad una chitarra specifica.

