# Audio Restoration per modelli generativi: miglioramento dell'audio tramite rete CRNN residua

October 31, 2025

**Adrian Patrizi**

## Abstract

I modelli generativi audio di nuova generazione, come *MusicGen*, sono in grado di creare brani musicali a partire da testo, ma le loro uscite presentano spesso artefatti. Poiché la qualità percepita dell'audio è un fattore cruciale per l'adozione di tali modelli in ambito creativo e produttivo, questo progetto affronta il problema del **miglioramento dell'audio generato** con l'obiettivo di aumentarne la qualità. È stata progettata una rete basata su architettura **CRNN con connessioni residue**, addestrata sul dataset *FMA Small* opportunamente degradato attraverso artefatti sintetici generati on-the-fly. I risultati sperimentali mostrano miglioramenti in termini di metriche spettrali, dimostrando l'efficacia del modello proposto.

## 1. Miglioramento della qualità audio: contesto e approcci esistenti

Negli ultimi anni, i modelli generativi audio come **MusicGen** (Copet et al., 2023) hanno dimostrato la capacità di *sintetizzare musica coerente a partire da descrizioni testuali*, aprendo prospettive interessanti per la creatività assistita dall'intelligenza artificiale. Tuttavia, le tracce prodotte presentano ancora *artefatti percettibili*, come rumore di quantizzazione, perdita di brillantezza timbrica e limitata estensione in frequenza. Questi difetti, seppur sottili, riducono la qualità percepita e ne limitano l'uso in contesti produttivi, dove la fedeltà sonora è cruciale.

La ricerca sul **miglioramento della qualità audio** si è sviluppata lungo tre principali direzioni: *speech enhancement*, *separazione di sorgenti musicali* e *super-risoluzione*.

Nel primo ambito, **SEGAN** (Pascual et al., 2017) è stato uno dei primi modelli basati su *Generative Adversarial Networks (GAN)* a operare direttamente nel dominio del tempo per la rimozione del rumore dal parlato. La rete generativa produce un segnale "pulito" mentre un discriminatore ne valuta la naturalezza. Sebbene efficace sul linguaggio, SEGAN mostra limiti sui segnali musicali, dove la struttura armonica è più complessa e non stazionaria.

**Demucs** (Défossez et al., 2020) affronta invece il problema della separazione delle sorgenti musicali mediante una *architettura encoder-decoder di tipo U-Net*, operando nel dominio del segnale. L'uso di convoluzioni e connessioni *skip* consente di preservare dettagli locali, ma il modello è pensato per separare componenti (voce, batteria, basso, strumenti vari) e non per correggere artefatti di generazione.

Un ulteriore approccio è rappresentato da **AudioSR** (Liu et al., 2023), che impiega un'architettura basata su *Transformer* per la ricostruzione delle alte frequenze in segnali a bassa risoluzione. Questo metodo mostra ottimi risultati nella super-risoluzione audio, ma richeide dataset di grandi dimensioni e non è progettato per gestire *distorsioni non lineari* come il clipping o il riverbero artificiale.

In sintesi, questi modelli mostrano che il miglioramento della qualità audio è un problema complesso e multifattoriale: SEGAN si concentra sulla pulizia del rumore, Demucs sulla separazione, e AudioSR sulla ricostruzione spettrale. Tuttavia, nessuno di essi affronta esplicitamente gli *artefatti sintetici introdotti dai modelli generativi musicali*, come MusicGen.

Per colmare questo divario, il presente progetto propone un approccio **supervisionato** per il miglioramento dell'audio generato. È stata progettata una **rete CRNN con connessioni residue**, addestrata sul dataset **FMA Small** opportunamente degradato *on-the-fly* come verrà presentato nella sezione successiva. L'obiettivo è ottenere una ricostruzione più fedele e percettivamente coerente dei segnali originali, valutata tramite **metriche spettrali** (L1, LSD e cosine similarity).

Il codice completo è disponibile su GitHub: `https://github.com/Adrian-Patrizi/Progetto-ML`.

Email: Adrian Patrizi <patrizi.2094287@studenti.uniroma1.it>.

*Machine Learning 2025, Sapienza University of Rome*, 2nd semester a.y. 2024/2025.

## 2. Architettura e impostazione sperimentale

**Dataset e degradazioni sintetiche.** Gli esperimenti sono stati condotti sul dataset *FMA Small*, contenente estratti musicali di 30 secondi appartenenti a vari generi. Ogni clip è stata suddivisa in segmenti casuali di 4 secondi, usati come unità di addestramento. Le degradazioni vengono applicate *on-the-fly* a ogni batch, simulando difetti tipici dell'audio generato da modelli come *MusicGen*:

- **Quantizzazione a bassa profondità con dithering** (8–12 bit), che introduce rumore e riduce la dinamica;
- **Downsampling e resampling** (16–22 kHz), che comportano perdita di banda e aliasing;
- **Riverbero artificiale** tramite convoluzione con una coda esponenziale, che altera chiarezza e risposta impulsiva;
- **Clipping leggero** (0.8–0.95), che introduce distorsione non lineare.

Il 15% dei campioni è mantenuto intatto per favorire stabilità e variabilità. Le clip sono convertite in **spettrogrammi Mel** in scala logaritmica (dB), normalizzati in $[0, 1]$. Tale rappresentazione riduce la dimensionalità e modella la risposta uditiva umana, rendendo l'apprendimento più stabile e percettivamente coerente.

**Architettura del modello.** Il modello proposto è una rete **CRNN con connessioni residue**, progettata per combinare la capacità di estrazione locale delle CNN con la modellazione temporale delle RNN. **L'encoder** include tre blocchi convoluzionali con normalizzazione e dropout, seguiti da una **GRU bidirezionale** nel bottleneck, che consente di catturare pattern temporali musicali. Il **decoder** utilizza strati ConvTranspose2D con connessioni residue per recuperare i dettagli ad alta frequenza e migliorare la convergenza. Il modello può operare in due modalità:

- **Diretta:** predice lo spettrogramma Mel migliorato;
- **Residua:** stima un residuo da sommare al Mel degradato.

**Funzione di perdita.** La funzione di perdita è data da:

$$\mathcal{L} = \mathcal{L}_{charb} + 0.3\,\mathcal{L}_{HF}$$

dove $\mathcal{L}_{charb}$ è la *Charbonnier loss*, robusta al rumore, e $\mathcal{L}_{HF}$ è una loss pesata sulle alte frequenze per enfatizzare brillantezza e armoniche.

L'ottimizzazione avviene tramite AdamW (learning rate $10^{-4}$, weight decay $10^{-3}$) con scheduler di tipo *ReduceLROnPlateau* e *gradient clipping*.

**Pipeline di addestramento e validazione.** Il modello è addestrato per 20 epoche con batch size 4 e *early stopping* (pazienza 5), salvando automaticamente la miglior versione. Le prestazioni sono valutate tramite $\mathcal{L}_1$, $\mathcal{LSD}$ e *similarità coseno* tra spettrogrammi, che misura la coerenza spettrale tra l'audio migliorato e quello di riferimento.

## 3. Analisi dei risultati sperimentali

**Setup e valutazione.** Gli esperimenti sono stati condotti su una GPU NVIDIA T4 in ambiente `PyTorch`. Le prestazioni sono state valutate tramite tre metriche descritte nella sezione precedente.

**Analisi dei risultati.** Come mostrato nel notebook, le curve di training risultano stabili e prive di overfitting, con convergenza raggiunta intorno alla 15ª epoca. La Tabella 1 riporta i guadagni medi ottenuti nel dominio Mel. Entrambi gli approcci mostrano un miglioramento rispetto all'audio degradato, con un vantaggio marginale per la modalità residua.

*Table 1.* Confronto dei guadagni medi nel dominio Mel tra gli approcci *diretto* e *residuo*. I valori indicano l'incremento rispetto al segnale degradato (↑ = maggiore è meglio).

| Approccio | Gain L1s | Gain LSD↑ | Gain Cos↑ |
|---|---|---|---|
| Diretto | 0.1649 | 13.574 | 0.1087 |
| Residuo | **0.1673** | **13.763** | 0.1087 |

## 4. Conclusioni e sviluppi futuri

Il lavoro presentato ha mostrato come una rete **CRNN con connessioni residue** possa migliorare la qualità spettrale di segnali audio degradati, ottenendo guadagni nelle metriche.

Come sviluppo futuro, si prevede di estendere l'addestramento a **nuove tipologie di degradazioni** per aumentare la robustezza del modello.

Un ulteriore passo, già avviato come mostrato nel secondo notebook su GitHub, consiste nella progettazione di una **pipeline zero-shot** composta da più modelli specializzati. L'obiettivo è applicare tale pipeline direttamente alle uscite reali di *MusicGen*. Questa è articolata in quattro fasi: separazione in stem tramite *Demucs*, miglioramento dello stem vocale mediante uno *speech enhancer*, ricomposizione del mix e successiva super-risoluzione con *AudioSR*. Non è stato possibile completare l'implementazione in tempo per la consegna, ma nel notebook sono riportate le scelte progettuali e i primi tentativi di integrazione. In prospettiva, tale pipeline potrebbe essere confrontata con il modello **CRNN**, per valutare i vantaggi e i limiti dei due approcci.

## References

Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., and Défossez, A. Simple and controllable music generation. *arXiv preprint arXiv:2306.05284*, 2023. URL https://arxiv.org/pdf/2306.05284.

Défossez, A., Usunier, N., Bottou, L., and Bach, F. Music source separation in the waveform domain. *arXiv preprint arXiv:1911.13254*, 2020. URL https://arxiv.org/pdf/1911.13254.

Liu, H., Chen, K., Qiao, T., Wang, W., and Plumbley, M. D. Audiosr: Versatile audio super-resolution at scale. *arXiv preprint arXiv:2309.07314*, 2023. URL https://arxiv.org/pdf/2309.07314.

Pascual, S., Bonafonte, A., and Serrà, J. Segan: Speech enhancement generative adversarial network. 2017. URL https://arxiv.org/pdf/1703.09452.