Manuale d’Uso – Analisi dei Canti

Versione: Marzo 2025

# 1. Caricamento e Visualizzazione (main\_program\_multi\_sc2.py)

Questa sezione permette di caricare e visualizzare file WAV, selezionarli, riprodurli e aprire strumenti di analisi.

Il programma consente due modalità di caricamento dei file .wav:

1. **Caricamento di un singolo file**  
   Questa modalità si utilizza nella **prima fase del flusso di lavoro**, dedicata all’identificazione dei canti all’interno di un file audio continuo (ad esempio una registrazione lunga).  
   È utile per:
   * Esaminare visivamente il segnale.
   * Applicare zoom e selezioni.
   * Individuare e salvare i singoli canti con strumenti dedicati.
2. **Caricamento di un’intera cartella**  
   Questa modalità si impiega **nella fase successiva**, quando i canti sono già stati identificati e salvati in file separati.  
   Caricando l’intera cartella, è possibile:
   * Analizzare automaticamente o semi-automaticamente i canti.
   * Applicare analisi spettrali e parametriche sui singoli frammenti audio.
   * Salvare i risultati in formato JSON/Pickle per ulteriori elaborazioni.

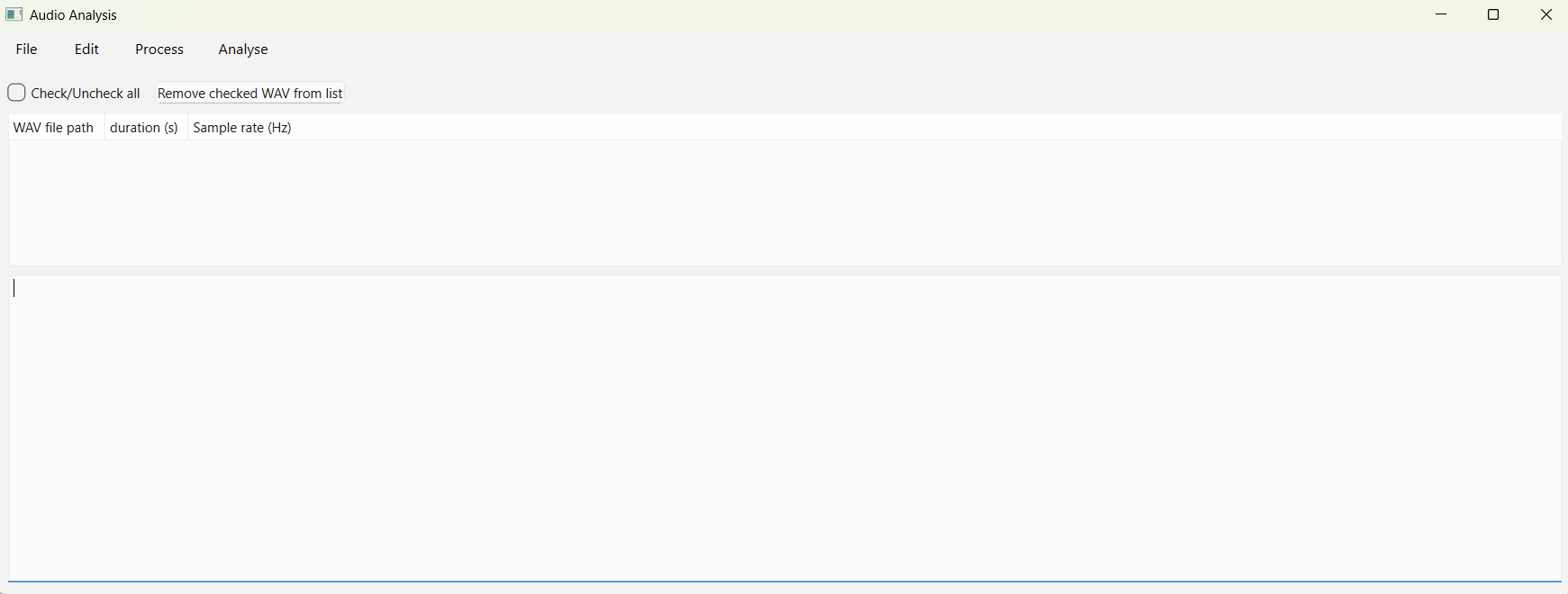


Figura 1 – Avvio del programma.

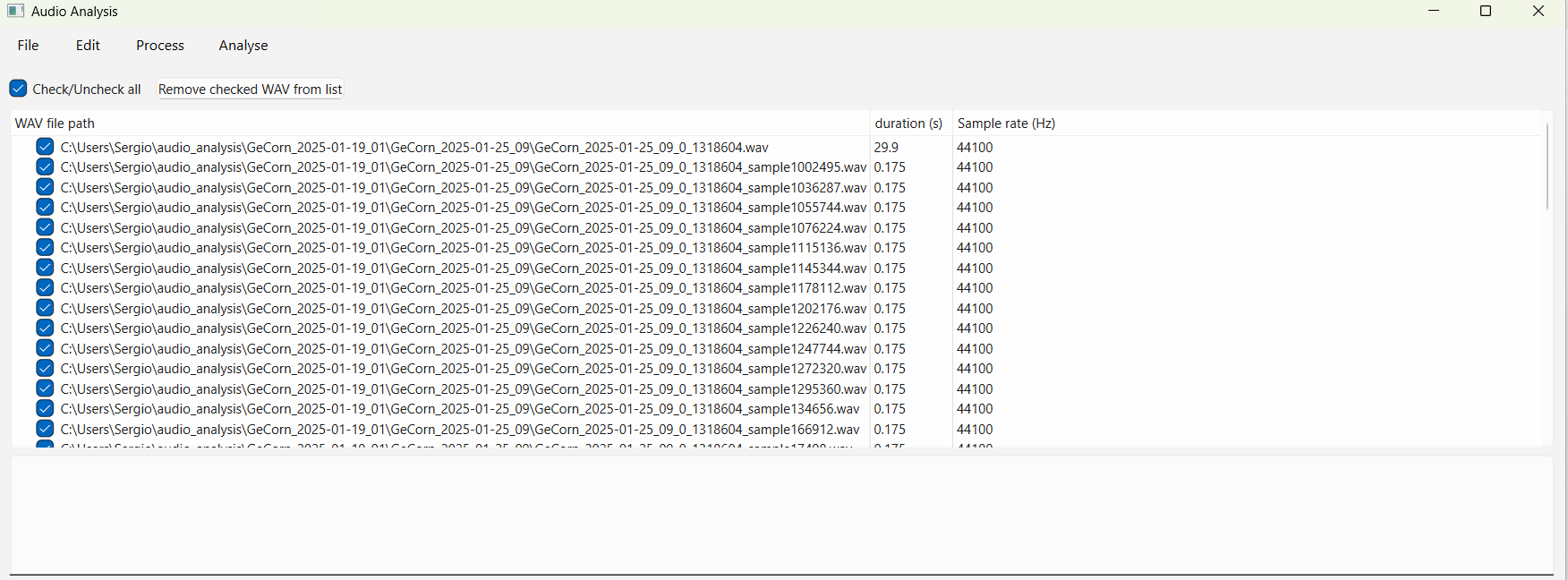


Figura 2 – Lista dei file caricati.

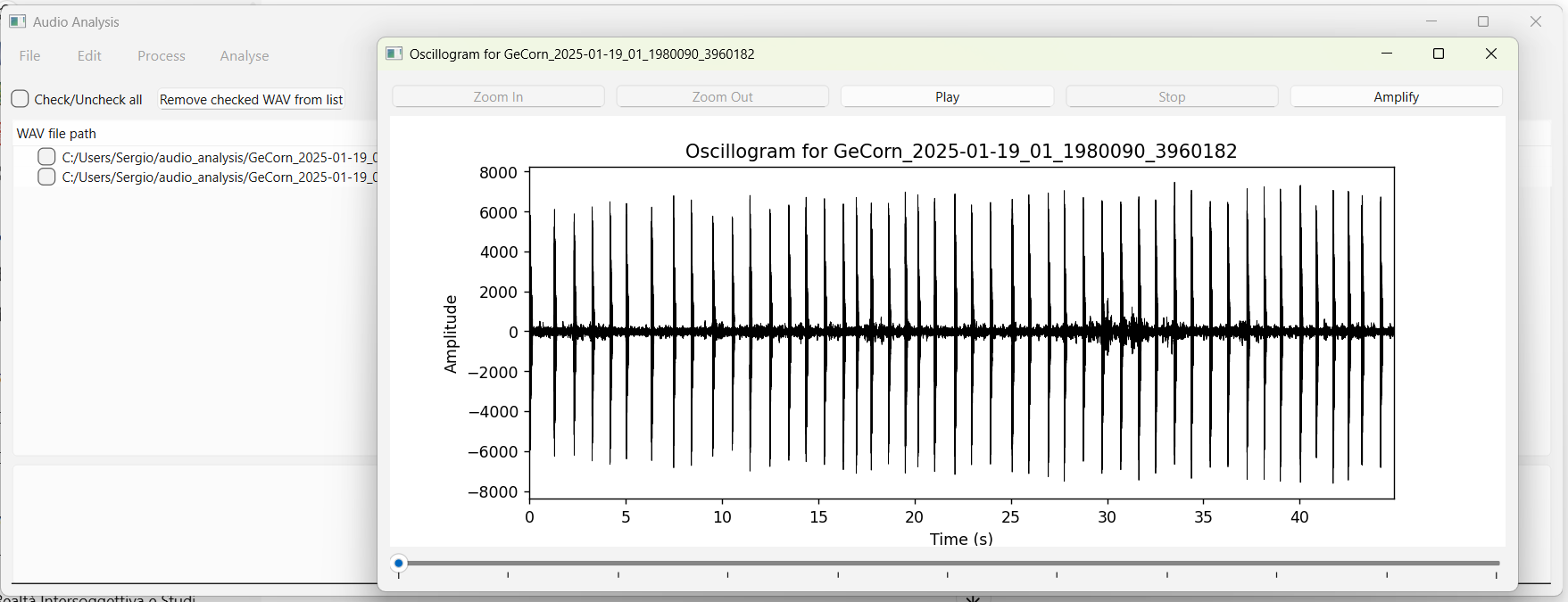


Figura 3 – Oscillogramma di un file.

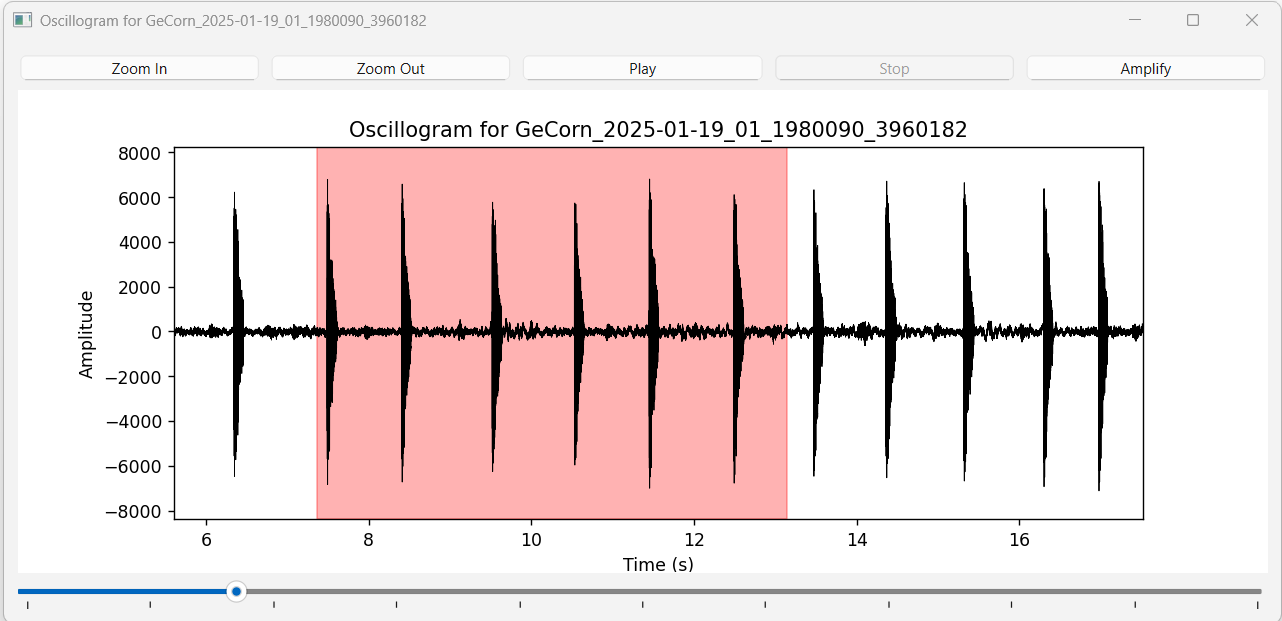


Figura 4 – Zoom sull’oscillogramma con selezione attiva.

# 2. Ritaglio in Spezzoni (wav\_cutting.py)

Questo modulo consente di suddividere file audio lunghi in segmenti di durata fissa, individuando il miglior punto di taglio nei pressi di zone silenziose.

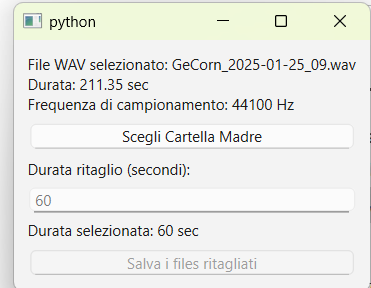


Figura 5 – Parametri per il ritaglio audio.

# 3. Rilevamento e Salvataggio dei Canti (trova\_picchi\_vs2.py)

Permette di calcolare l’inviluppo RMS, identificare i picchi (canti), stimarne la durata e salvarli come file distinti.

## Parametri per il calcolo dell’inviluppo (Envelope)

L’inviluppo rappresenta l’andamento generale dell’ampiezza nel tempo e viene ottenuto tramite una media scorrevole dei valori assoluti del segnale.

Due parametri principali ne influenzano la forma:

* Window size (finestra): Indica la durata (in millisecondi) della finestra su cui viene calcolata la media dell’ampiezza assoluta. Una finestra più ampia produce un inviluppo più smussato, utile per ignorare le variazioni interne al canto (es. impulsi) e individuare la forma generale del canto (inizio e fine).
* Overlap (passo) Indica di quanto si sposta la finestra tra un calcolo e il successivo. Un overlap più piccolo genera una curva più continua, mentre uno più grande rende l’inviluppo più “a gradini”.

Per facilitare il rilevamento automatico dei canti nel segnale, bisogna ottenere un inviluppo che segua il profilo del canto, senza evidenziare i modulazioni interne, come gli impulsi.

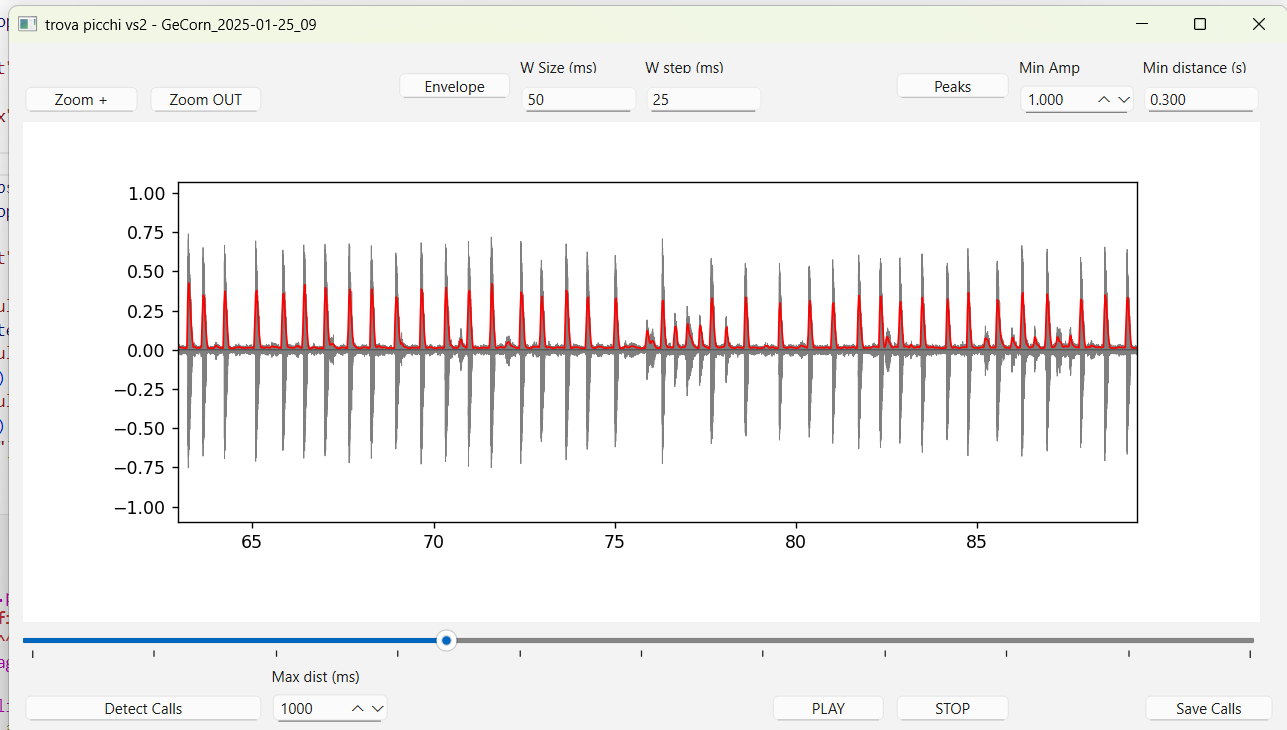


Figura 6 – Inviluppo calcolato.

## Parametri per l’individuazione dei picchi (Peaks)

Calcolato l’inviluppo, si procede con la rilevazione dei picchi, che corrispondono al singolo canto. Il rilevamento utilizza due parametri della curva di inviluppo:

* Ampiezza minima (Min Amp). Imposta una soglia e permette di filtrare i picchi sotto di essa. Questo parametro è utile per eliminare piccoli rumori di fondo o variazioni non significative.
* Distanza minima (Min Distance). Impone un intervallo minimo di tempo tra due picchi consecutivi, espresso in secondi. Evita il rilevamento di picchi troppo ravvicinati che potrebbero derivare da fluttuazioni locali dell’inviluppo all’interno di un singolo segnale.

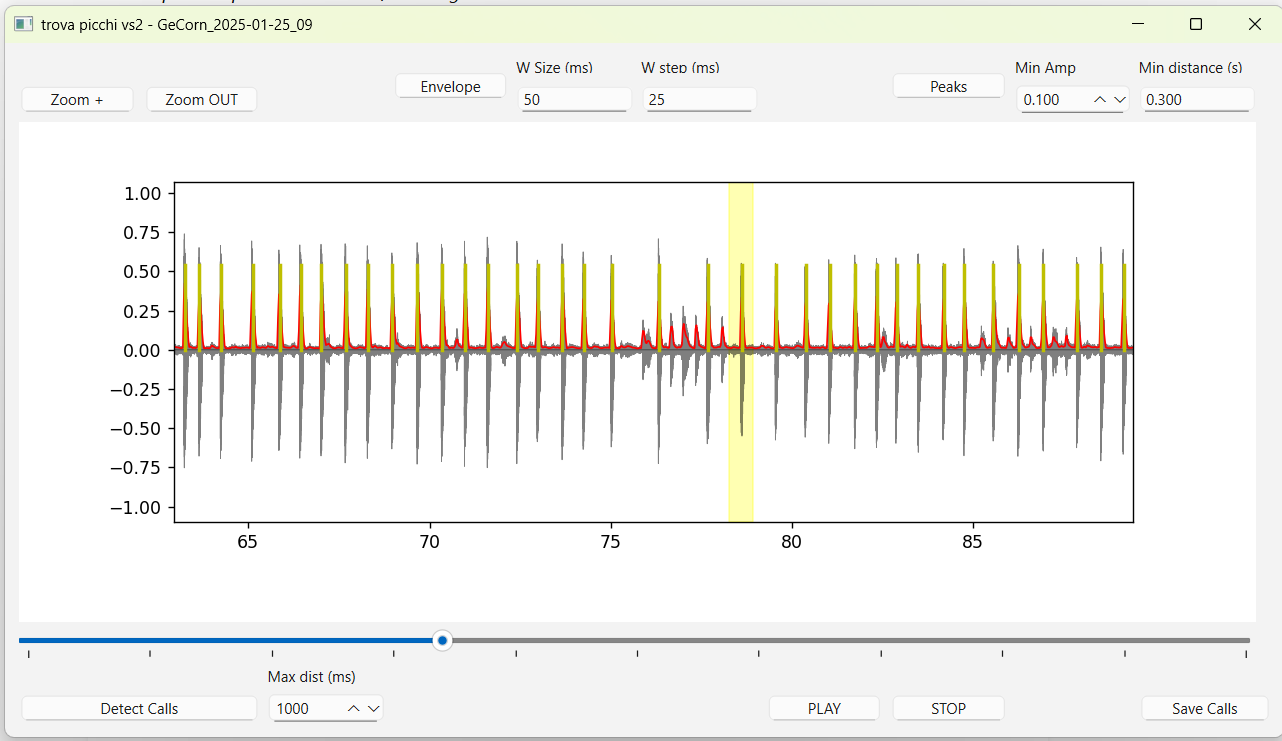


Figura 7 – Picchi rilevati.

## Parametri per definire inizio e fine della traccia contenente il singolo canto (Detect Calls)

Trovati i picchi, si utilizza la funzione Detect Calls per individuare inizio e fine di ciascun segmento contenente un canto. La funione usa il parametro “Max dist (ms)” che definisce la l’ampiezza della finestra di calcolo utilizzata dall’algoritmo. Questa finestra deve essere all’incirca lunga come (o poco inferiore alla) durata media del canto, ma meno lunga della distanza tra due picchi successivi. L’algoritmo utilizzato è il seguente:

Dove è il campione corrispondente al picco j-esimo dell’inviluppo e *K* è il numero di campioni corrispondenti alla Max\_dist impostata.

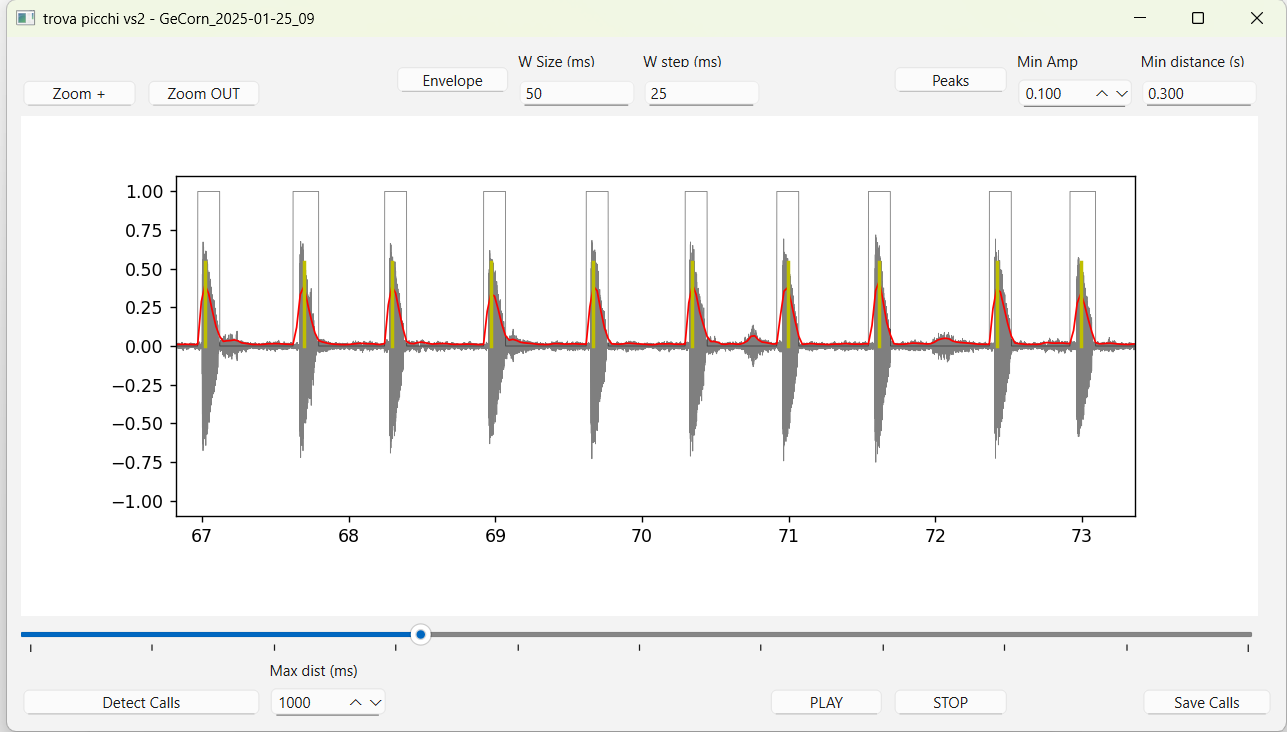


Figura 8 – Canti individuati.

Una volta individuati e controllati i segmenti, si può procedere al taglio e al salvataggio dei singoli spezzoni. Il programma chiede di scegliere una cartella in cui salvare i canti. Il programma, crea quindi una sotto-cartella, il cui nome è ??? e salva al suo interno i singoli spezzoni.

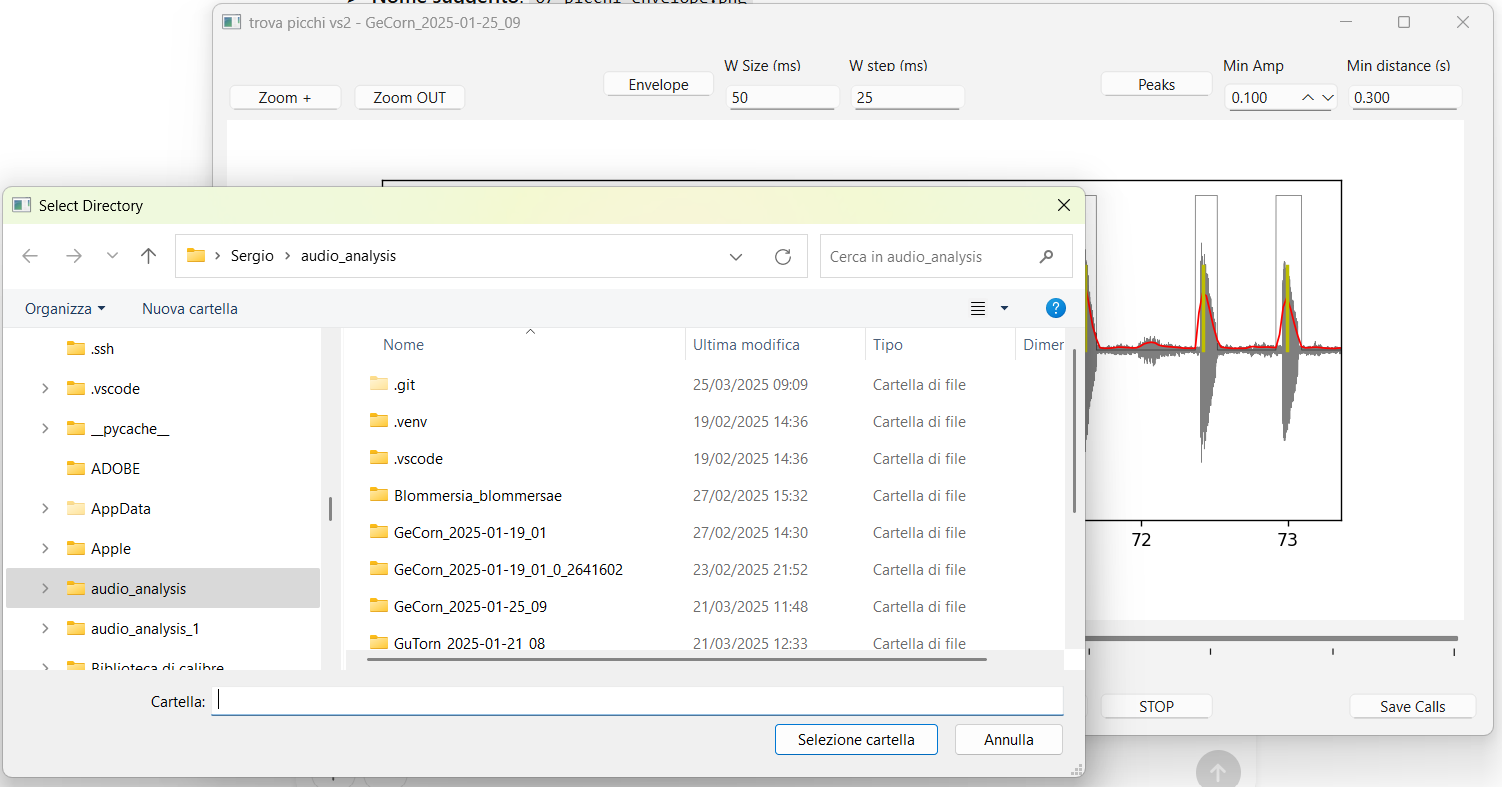


Figura 9 – Finestra di salvataggio dei canti.

# 4. Analisi dei Singoli Canti (analisi\_singoli\_canti.py)

Questo modulo elabora i canti salvati, estrae parametri come durata, numero di impulsi e spettro di frequenza, e salva i risultati in formato JSON. Il programma può procedere in modalità interattiva e non-interattiva (automatica). Nel primo caso, con i tasti [Previous file] e [Next file] si scorre la lista dei files, si modificano i parametri e, se soddisfatti, si salvano i risultati nel file JSON. Nella modalità automatica, il programma processa tutti i files applicando gli stessi parametri e salvando i dati nel file JSON, senza possibilità di modificarli.

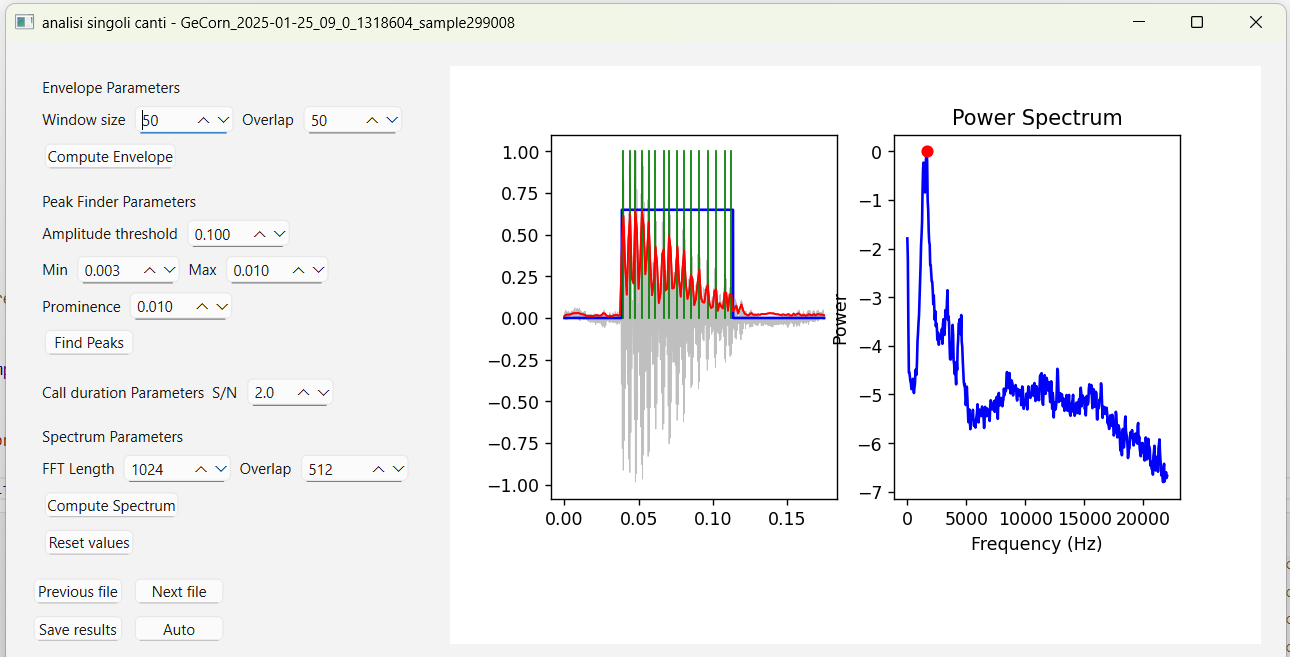


Figura 10 – Analisi di un canto (oscillogramma e spettro).

# Parametri per il calcolo dell’inviluppo (RMS)

|  |  |
| --- | --- |
| **Parametro** | **Descrizione** |
| window\_size | Lunghezza della finestra (in campioni) usata per calcolare il valore RMS. |
| overlap | Numero di campioni di sovrapposizione tra una finestra RMS e la successiva. |

# Parametri per la rilevazione dei picchi

|  |  |
| --- | --- |
| **Parametro** | **Descrizione** |
| min\_amplitude | Soglia minima di ampiezza: esclude i picchi con valori RMS troppo bassi. |
| min\_distance | Distanza minima tra due picchi consecutivi (in secondi). |
| max\_distance | Distanza massima per considerare validi due picchi consecutivi. |
| prominence | Prominenza minima richiesta per un picco: quanto si distingue dal contesto. |

# Parametro per identificare l’intero canto

|  |  |
| --- | --- |
| **Parametro** | **Descrizione** |
| signal\_to\_noise\_ratio | Rapporto segnale/rumore usato per stimare inizio e fine del canto in base al livello di fondo dell’inviluppo. |

# Parametri per lo spettro di potenza

|  |  |
| --- | --- |
| **Parametro** | **Descrizione** |
| fft\_length | Numero di campioni su cui calcolare la FFT (risoluzione dello spettro). |
| fft\_overlap | Numero di campioni condivisi tra finestre successive nella stima spettrale. |