

Final project

Paolo Ostan 1189730

1 Introduzione

Nel seguente progetto è stata trattata l'implementazione di un algoritmo di *Voice Activity Detection*, che permette di distinguere l'attività vocale dell'utente dai rumori di sottofondo. Poiché il rumore di sottofondo non contribuisce alla comprensione del discorso evitare di trasmetterlo non inficia sull'esperienza dell'utente finale permettendo però un grande risparmio di *Bandwidth*. [1]

2 Implementazione

Dopo una ricerca bibliografica [2] e alcuni test con i file forniti, si è scelto di utilizzare un algoritmo che implementa due metodi per la valutazione dell'attività vocale:

- *Analisi dell'Energia dei pacchetti*: All'interno del pacchetto da valutare si analizza se l'energia complessiva sia al di sopra di una certa soglia, si adatta questa soglia utilizzando un valore valori di energia calcolata nei pacchetti di silenzio.
- *Varianza dello spettro*: Si utilizza la varianza dello spettro di frequenze su cui si sviluppa il segnale per comprendere

Si è poi scelto di implementare un metodo che permetta di allungare la coda dei momenti di parlato. Così da permettere un risultato migliore nel caso di situazioni particolarmente rumorose in cui il parlato viene coperto dal rumore di sottofondo e l'algoritmo elimina le parti finali del parlato. Così da emulare il parametro di rilascio di molti processori audio di dinamica che permettono di regolare l'aggressività dell'elaborazione sulla parti finali dell'attività vocale.

Nello specifico l'algoritmo analizza lo stream fornito in pacchetti da 20 ms, così da emulare il comportamento di un algoritmo utilizzato per il trattamento di uno stream realtime.

È composto di due componenti che gestiscono l'analisi pacchetto per pacchetto:

- *vad*: funzione che ha il compito di accumulare i valori minimi e gestire il rilascio del filtro alla fine dell'attività vocale.
- *vadBuff*: Funzione che gestisce l'analisi effettiva di energia e spettro e viene aggiornata nel corso dell'analisi eseguita.

3 Flowchart

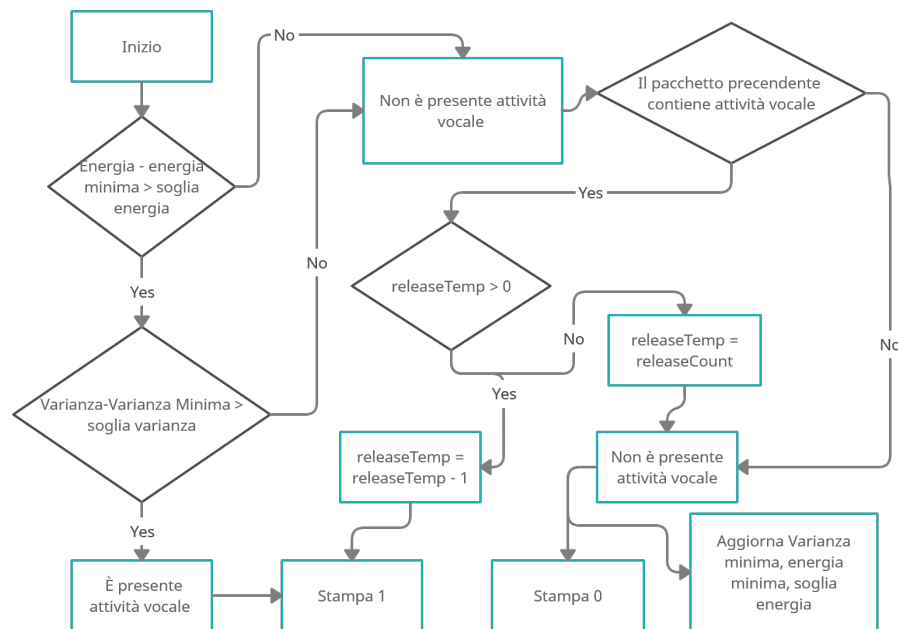


Figura 1: VAD FlowChart

- *ReleaseCount*: Valore che tiene la quantità massima di pacchetti lasciati dopo che l'algoritmo precedente determina la fine dell'attività vocale.
- *ReleaseTemp*: Valore che mantiene momentaneamente la quantità di pacchetti da lasciare attivati e reimpostato ogni volta che viene trovato.

4 Risultato simulazione

Si illustrano di seguito i risultati degli esempi forniti.

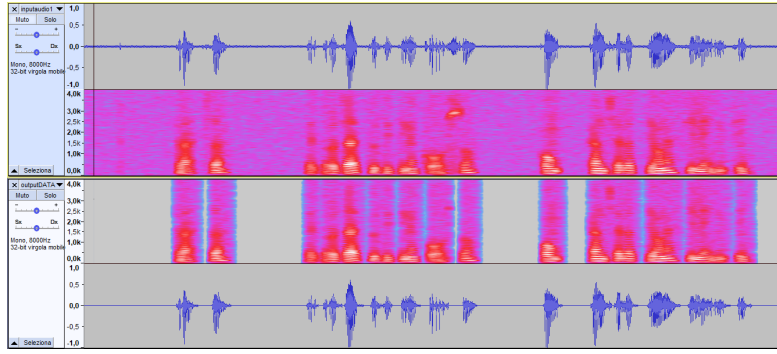


Figura 2: input/output audio 1

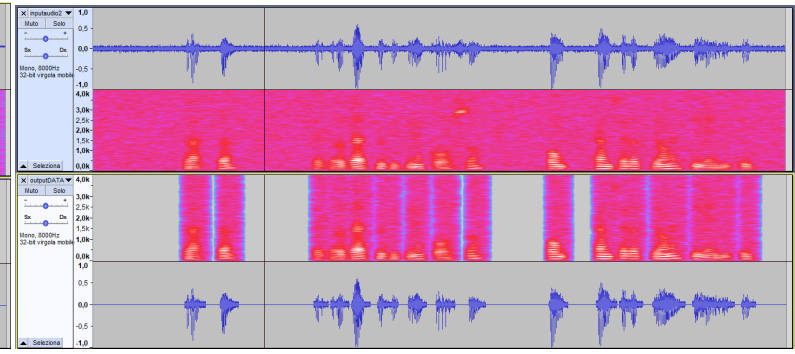


Figura 3: input/output audio 2

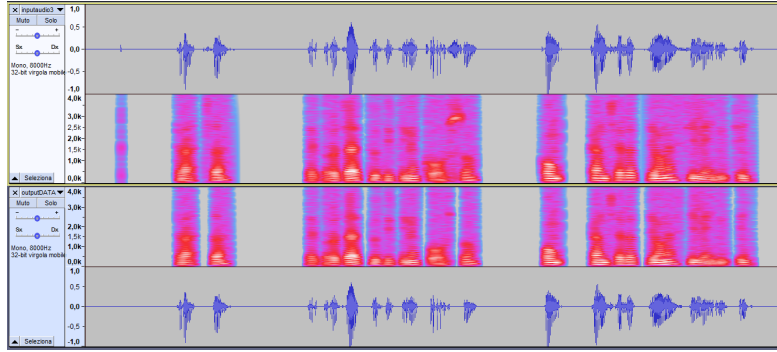


Figura 4: input/output audio 3

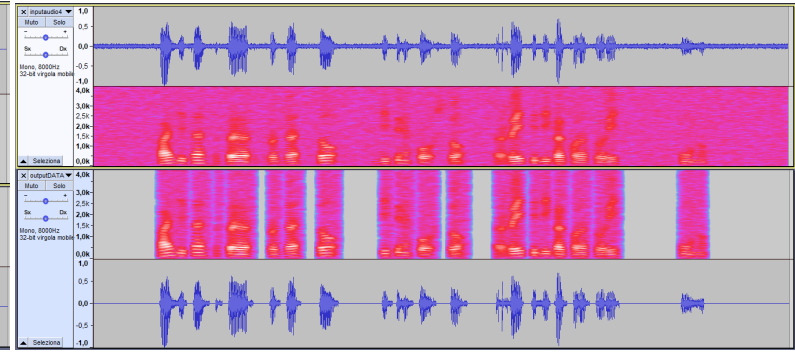


Figura 5: input/output audio 4

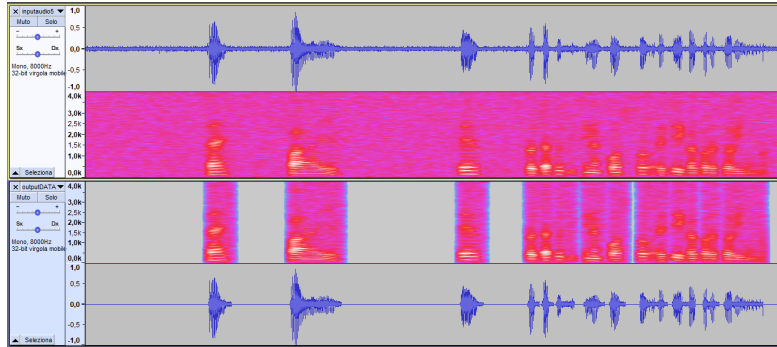


Figura 6: input/output audio 5

Possiamo notare come in ogni confronto la coppia forma d'onda/spettrogramma superiore, rappresentante la traccia fonte renda evidenti alla vista le differenze fra i momenti in cui è presente attività vocale e quelli in cui è presente solo rumore di sottofondo sia sottoforma di forma d'onda sia sottoforma di spettrogramma. La forma d'onda infatti risulta avere molta più energia nei momenti di parlato rispetto ai momenti di rumore di fondo, anche in tracce audio molto rumorose all'ascolto, come la seconda presentata. All'interno dello spettrogramma notiamo invece uno spettro molto omogeneo su tutte le frequenze per le zone di rumore di fondo, mentre nelle zone con attività vocale si notano maggiori picchi di energia, soprattutto su alcune frequenze specifiche. Notiamo infatti come, negli audio di Output l'algoritmo abbia eliminato i momenti con forma d'onda con energia bassa e spettrogramma uniforme, mantenendo quelli con una maggiore energia e uno spettrogramma vario.

Riferimenti bibliografici

- [1] Mousa Tawfiq Al-Akhras. «Quality of media traffic over Lossy internet protocol networks: Measurement and improvement.» In: (2007).
- [2] Mohammad Hossein Moattar e Mohammad M Homayounpour. «A simple but efficient real-time voice activity detection algorithm.» In: *2009 17th European Signal Processing Conference*. IEEE. 2009, pp. 2549–2553.