POLITECNICO DI TORINO

Corso di Laurea Magistrale in
Ingegneria del Cinema e dei Mezzi di Comunicazione

Tesi di Laurea Magistrale

Movimenti labiali reali e sintetici in ambienti virtuali immersivi per l'Audio Space Lab del Politecnico di Torino: integrazione e analisi dei contributi ai fini dell'intelligibilità del parlato

Relatori

Prof. Arianna ASTOLFI

Dott. ing. Angela GUASTAMACCHIA

Prof. Andrea BOTTINO

Prof. Louena SHTREPI

Candidato

Andrea GALLETTO

Luglio 2024

Si ringraziano per la partecipazione:

Irene: nel ruolo dell'esuberante compagna
MammaMari e PapaGio: nel ruolo dei genitori rassegnati
Paola: nel ruolo della sorella mai sazia
Aldo: nel ruolo del coinquilino
Alessandro: nel ruolo del confidente
Elisa: nel ruolo della creatura
Paolone, Luigi, Flavio, Davide, Luca, Gian, Frank, Paolino e Richard: nel ruolo dei (ex)colleghi nonché amici (molto pazienti)
Annalisa: nel ruolo dell'antropologa
Pippo: nel ruolo di Piii il consulente sporadico
Piero e la consorte Laura: nel ruolo dei golosi di "lacet"
Stefano: nel ruolo del cugino goloso di "lacet"
Tutte le comparse che hanno partecipato nel ruolo di ipoudente
E tutti coloro che in qualche modo hanno contribuito all'opera

Con: Arianna nel ruolo della protagonista

Assistenti alla fotografia:
Andrea Bonvissuto e Daniele De Rossi

Fonico:
Andrea Galletto

Consulente fonico:
Antonio Servetti

Sceneggiature di:
Arianna Astolfi, Angela Guastamacchia, Andrea Galletto

Attrezzature tecniche: Visionary Lab, LABINF, ACSLAB

Assistenti alla regia:
Angela Guastamacchia, Arianna Astolfi, Andrea Bottino, Louena Shtrepi

Regia e fotografia:
Andrea Galletto

Bravi tutti! ...clap!...clap!...clap!...clap!!

Sommario

I disturbi dell'udito sono malattie invalidanti, influenzano le capacità cognitive e portano chi ne soffre all'isolamento sociale. Gli apparecchi acustici possono mitigare tali problemi, ma il 20% dei portatori è restìa al loro uso. Le cause risiedono nei metodi di test dei dispositivi che non rispecchiano scenari acustici realistici come una conversazione in un bar o aeroporto: situazioni critiche per un orecchio sano con alle spalle migliaia di anni di evoluzione biologica, vera e propria sfida per un dispositivo acustico che non gode di pari esperienza evolutiva. Tali metodi si concentrano sull'acustica tralasciando elementi visivi che concorrono alla comprensione, come il labiale dell'interlocutore. La ricerca ha iniziato a integrare la realtà virtuale nei test uditivi per ricreare ambienti AudioVisivi (AV) immersivi, testare più fedelmente la resa degli apparecchi e valutare i fattori che influenzano la comprensione del parlato. Tuttavia, tali ambienti sono basati su simulazioni acustiche e rendering video 3D aventi avatar come interlocutori, condizioni non del tutto realistiche per le investigazioni.

Questa tesi si propone di valutare l'influenza del labiale sulla comprensione tramite test immersivi creati con registrazioni AV 3D. I test AV di intelligibilità del parlato, composti da filmati 3D 360° con audio ambisonico del 3° ordine, già disponibili presso l'Audio Space Lab (ASL) del Politecnico di Torino, sono stati integrati con video dove fosse visibile il labiale del parlatore target del test. Per un'indagine completa e attuale, sono stati confrontati l'apporto del labiale di una persona reale e di un avatar realistico animato con intelligenza artificiale (IA). Tre gli scenari selezionati, ambientati in una sala conferenze altamente riverberante con parlatore target frontale, parlatore interferente a 120°, 180° o assente e rapporto segnale rumore di -5 dB.

Un'attrice ha fornito il labiale per il discorso target rappresentato dal test di intelligibilità "Italian Matrix Sentence Test", aiutata da un sistema di gobbo creato ad-hoc. Le riprese sono state girate in modalità 3D 360° e in post-processing il soggetto è stato integrato negli scenari; tramite analisi audio sono state scelte le clip dove il labiale era sincrono con l'audio dei relativi enunciati. L'avatar è stato animato con le tracce audio dello stesso test e integrato negli stessi scenari, nella medesima posizione dell'attrice.

La sperimentazione presso l'ASL ha coinvolto 20 soggetti normoudenti, divisi in due gruppi, ognuno sottoposto a una diversa condizione di test: con labiale reale o sintetico. Le percentuali di intelligibilità raggiunte sono state inoltre confrontate con quelle dei test pregressi per gli stessi scenari privi dell'informazione visiva fornita dal labiale. Nei test con parlatore interferente, l'aggiunta del labiale del parlatore target, sintetico e non, ha significativamente migliorato l'intellegibilità;

in particolare, i test con labiale reale hanno raggiunto la migliore percentuale di intellegibilità media, nell'ordine 78%, 68,5% e 58,5%.

Il labiale è importante per l'intelligibilità, ma sebbene il labiale sintetico la migliori, al momento, non compete con quello reale che ha un apporto informativo superiore. Includere nei test il labiale reale risulta essenziale per ricreare situazioni verosimili ed essere funzionale al miglioramento delle protesi, mentre la continua evoluzione dell'IA potrebbe portare il labiale sintetico a competere con quello reale e aprire nuovi scenari di sperimentazione.

Indice

Introduzione

Health is an investment in the future: the cost of doing nothing is one we cannot afford

Dr. Tedros Adhanom Ghebreyesus, WHO Director-General, 2020

Una visione d'insieme del contesto generale e del progetto complessivo, a cui il presente lavoro contribuisce, è doverosa al fine di comprenderne le ragioni, gli scopi, e immaginarne le future evoluzioni.

Per capire l'entità e la gravità del problema, che il progetto nel suo complesso intende affrontare, è sufficiente citare alcune delle cifre stimate dall'Organizzazione Mondiale della Sanità (OMS) riguardanti le disfunzioni uditive, ampiamente diffuse e in continua crescita. Le ragioni sono da ricercare, oltre che nella tendenza demografica globale in costante aumento, anche nell'allungamento delle aspettative di vita. Secondo l'OMS, infatti, in ragione di questo trend di crescita della popolazione mondiale, entro il 2050 approssimativamente 2.5 miliardi di persone (circa 1 ogni 4) soffriranno di ipoacusia e di queste circa 700 milioni subiranno una perdita dell'udito da moderata a grave nel loro orecchio migliore [1, p. 141]. Questi dati guidano gli stati membri dell'OMS affinché intervengano con azioni di prevenzione e cura della salute dell'orecchio e dell'udito.

I disturbi uditivi sono purtroppo condizioni invalidanti che influiscono significativamente sulla qualità della vita, con conseguenze negative su apprendimento e vita lavorativa, portando chi ne soffre a una condizione di isolamento sociale. Migliorare le loro condizioni è importante sia per il benessere del singolo che per quello della società. L'ipoacusia è nella maggioranza dei casi irreversibile, ciò rende necessaria una riabilitazione continuativa in grado di migliorare la capacità di ascolto e offrire ai soggetti una migliore qualità della vita. Gli approcci alla riabilitazione variano in base alla gravità e alle possibilità di intervento e comprendono: (i) sistemi tecnologici come apparecchi acustici e impianti cocleari; (ii) linguaggio dei segni e altri metodi comunicativi di sostituzione sensoriale come Tadoma (metodo di comunicazione utilizzato dalle persone sordocieche che prevede di appoggiare le dita sulle labbra, sulle guance e sulla gola di chi parla) [2]; (iii)

terapie riabilitative come Total Communication (metodi di comunicazione in cui alla lingua parlata vengono associati altri tipi di supporti e indicazioni visive) [1, pp. 96, 109]. Per quanto riguarda le tecnologie di supporto all'udito, l'OMS e lo studio Global Burden of Deseas, guidato dall'IHME (Institute for Health Metrics and Evaluation) dell'Università di Washington, stimano che, sebbene a livello globale circa 400 milioni di persone trarrebbero beneficio dall'uso di apparecchi acustici, ben l'83% percento non ne fa uso [1, p. 178]. Anche nei paesi a più alto reddito, dove il loro utilizzo è maggiormente diffuso, tale percentuale è piuttosto elevata e si attesta intorno al 77%: oltre i tre quarti dei soggetti che trarrebbero beneficio da tali dispositivi non li utilizza [1, p. 178]. A questi dati si aggiunge un ulteriore elemento negativo: circa il 20% tra coloro che detengono un apparecchio acustico non lo impiega [3]. Le ragioni di questo atteggiamento sono molteplici: lo stigma dell'indossare il dispositivo, il comfort e l'aspetto estetico, fattori sociali e circostanze di utilizzo, difficoltà nell'uso, nella cura e nella manutenzione. Tra tutte, le motivazioni principali risultano essere legate all'efficacia dell'ausilio: i soggetti lamentano che il dispositivo è di scarso aiuto, non fornisce benefici rilevanti, presenta difficoltà di ascolto in situazioni rumorose e in presenza di rumore di fondo, oltre a una bassa qualità del suono e scarsa direttività.

Una delle cause di questo insuccesso è da ricercarsi nelle metodologie impiegate nei consueti test audiologici: gli scenari acustici cui i pazienti sono sottoposti risultano alquanto semplificati e non si avvicinano alle condizioni reali che essi affrontano nella vita quotidiana. Le procedure tradizionali prevedono l'analisi della sensibilità ai toni puri e il riconoscimento di parole in condizioni di quiete e di rumore stazionario [4], [5]. Risulta evidente come queste condizioni approssimate non rappresentino adeguatamente le complesse situazioni di ascolto giornaliere in cui un soggetto con ipoacusia si può ritrovare. Oltretutto, gli stimoli visivi correlati a quelli uditivi e fondamentali ai fini della percezione e della comprensione del parlato sono tradizionalmente tralasciati, nonostante il loro contributo sia tutt'altro che trascurabile.

Alla luce di queste osservazioni, è emersa l'idea di sviluppare nuove tipologie di test che considerino, oltre all'aspetto acustico, anche gli stimoli visivi finora trascurati. Questo approccio consente di effettuare tarature e test più rigorosi degli apparecchi acustici e diagnosticare in modo più efficace i problemi uditivi. Per raggiungere questo obiettivo, si stanno sviluppando ambienti di test in realtà virtuale che, tramite l'uso di visori e sistemi audio in grado di riprodurre ambienti acustici tridimensionali, diano al soggetto l'impressione di ritrovarsi in una situazione reale e poter così reagire in modo più naturale e spontaneo. La valenza ecologica delle sperimentazioni, ossia il loro essere aderenti alla realtà, è ciò cui la ricerca sta cercando di ricondursi con tali tipologie di test. Questo approccio permette di fare un'analisi più scrupolosa e attenta sia delle patologie dei soggetti che del loro comportamento e sia del funzionamento dei dispositivi. Un soggiorno riverberante

e un paesaggio innevato [6] sono solo alcuni esempi di ambienti virtuali ricreati ai fini di indagini acustiche che seguono tale modello.

Tuttavia, al momento gli studi esistenti vengono condotti utilizzando ambientazioni basate su simulazioni acustiche e rendering 3D di modelli virtuali, in cui eventuali interlocutori sono rappresentati da avatar. Ne consegue che queste condizioni non sono del tutto realistiche per le indagini in contesti con un'ampia validità ecologica verso cui la ricerca si sta orientando.

Questa tesi mira a investigare l'influenza che il labiale esercita sulla comprensione del parlato tramite test audiovisivi immersivi con un'alta valenza ecologica. I test di intelligibilità, costituiti da registrazioni AudioVisive (AV) 3D 360° e audio ambisonico del 3° ordine, attualmente disponibili presso l'Audio Space Lab (ASL) del Politecnico di Torino, sono stati integrati con filmati in cui fosse visibile il labiale del parlatore target del test. Per svolgere un'indagine completa che prendesse in considerazione anche le promettenti tecnologie di intelligenza artificiale (IA), l'oratore target è stato rappresentato oltre che da una persona reale, anche da un avatar realistico, il cui movimento labiale è stato creato con tecniche di animazione basate su IA.

Tra le ambientazioni disponibili, sono stati selezionati tre scenari registrati nella sala conferenze del Museo Egizio di Torino, caratterizzata da un elevato tempo di riverbero (3,2 s). In ciascuno scenario, il parlante target del test si trovava a circa 4 metri di distanza, ancora sufficiente per un buon riconoscimento dei movimenti labiali. I tre scenari prevedevano tutti un parlante target frontale al soggetto e, rispettivamente, nessun parlante interferente, un parlante interferente a 120° e uno a 180°, con un rapporto segnale rumore pari a $-5\ dB$.

La tesi è stata sviluppata presso l'Audio Space Lab, uno spazio opportunamente attrezzato per poter sviluppare e svolgere test di ascolto in realtà virtuale. Il laboratorio, realizzato in una sala trattata acusticamente, permette la riproduzione di audio ambisonico del terzo ordine (una tecnologia surround che verrà descritta successivamente) grazie a un array di 16 altoparlanti sincronizzati con un visore Meta Quest 2 utilizzato per la riproduzione degli stimoli visivi in realtà virtuale.

La tesi si articola in 8 capitoli così strutturati.

- **Capitolo 1 - Nozioni di base**
  Presenta richiami di teoria acustica e delle grandezze coinvolte, nozioni su tecniche e formati di registrazione e sulle modalità sperimentali in ambito audiologico.

- **Capitolo 2 - Stato dell'arte**
  Espone lo stato dell'arte inerente alle ricerche su percezione e intelligibilità del parlato.

- **Capitolo 3 - Integrazione del labiale reale**
  Descrive il flusso di lavoro seguito per creare i filmati con il labiale reale e la loro successiva integrazione negli scenari di test.
- **Capitolo 4 - Integrazione del labiale sintetico**
  Descrive il flusso di lavoro seguito per creare i filmati con il labiale di sintesi e la loro successiva integrazione negli scenari di test.
- **Capitolo 5- Test di intelligibilità**
  Illustra la fase di sperimentazione dei test di intelligibilità condotti presso l'Audio Space Lab del Politecnico di Torino.
- **Capitolo 6 - Risultati**
  Discute i risultati ottenuti nella fase sperimentale.
- **Capitolo 7 - Conclusioni e prospettive future**
  Espone una valutazione complessiva del lavoro svolto e i futuri sviluppi.
- **Premi e riconoscimenti**
  Presenta i premi e i riconoscimenti a cui questo lavoro ha contribuito.

Capitolo 1

Nozioni di base

In questa sezione vengono introdotte le grandezze acustiche, le terminologie e i concetti fondamentali utili alla comprensione del presente lavoro; viene esposta l'importanza degli stimoli visivi al fine della percezione del parlato e per ultimo vengono descritti i sistemi di riproduzione più adeguati alla creazione di ambientazioni audiovisive realistiche, con particolare riferimento alla parte audio.
In tutto il testo, per quanto possibile si cercherà di utilizzare una corretta terminologia italiana, tuttavia, si farà ricorso anche di terminologia inglese, comunque descritta nei suoi corrispondenti termini in italiano, per rimanere aderenti alla documentazione anglofona maggiormente usata nell'ambito della ricerca audiologia.

1.1 Grandezze acustiche e terminologia

- Pressione sonora $p(t)$ [7]
  È la variazione tra il valore istantaneo della pressione, perturbato dall'onda sonora, e il punto di equilibrio "p0" costituito dalla pressione atmosferica. A causa della sua natura oscillatoria, per ottenere un valore che meglio rappresenti la percezione umana del fenomeno acustico e per avere un valore misurabile sperimentalmente, si fa molto spesso riferimento al valore RMS o efficace della pressione sonora, definito dalla seguente relazione:

$$p_{eff} = \sqrt{\frac{1}{T}\int_0^T p(t)^2\,dt} \qquad Pa \tag{1.1}$$

- Livello di pressione sonora - Sound Pressure Level (SPL o L_p) [7]
  É 20 volte il logaritmo del rapporto tra i valori efficaci della pressione sonora del fenomeno analizzato e della pressione di riferimento della soglia di udibilità (valore prefissato a 20 μPa alla frequenza di 1 kHz):

$$L_p = 20 \log \left(\frac{p}{p_0} \right) \qquad dB(SPL) \tag{1.2}$$

- Livello di pressione sonora ponderato A [7]
  È il livello di pressione sonora a cui è applicata la curva di ponderazione di tipo A. Questo tipo di ponderazione fornisce risultati analoghi alla sensibilità dell'orecchio umano che è variabile nell'intervallo di frequenze udibili. L'unità di misura $dB(A)$ o dBA esprime i valori ottenuti con la curva di ponderazione A.

- Rapporto segnale-rumore - Signal To Noise Ratio (SNR) [8]
  È la differenza espressa in dB tra il livello di pressione sonora del segnale considerato (L_S) e quello del rumore (L_N):

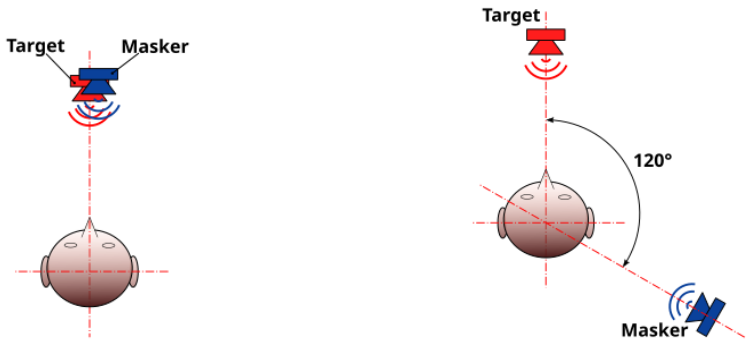
$$SNR = L_S - L_N \qquad dB \tag{1.3}$$

- Soglia di riconoscimento del parlato - Speech Recognition Threshold (SRT) [9]
  É il minimo valore SNR al quale un individuo è in grado di riconoscere una determinata percentuale di parole da un insieme di test. SRT50 si riferisce a una percentuale del 50%, SRT80 dell'80%.

- Segnale Target
  In audiologia si intende un segnale acustico, solitamente vocale, che un soggetto sottoposto a test di ascolto deve riconoscere.

- Segnale mascherante
  In audiologia si intende un segnale di diversa natura usato intenzionalmente come disturbo durante i test di ascolto. A volte indicato con *masker, segnale interferente o mascheratore.*

- Rilascio spaziale da mascheramento - Spatial Release from Masking (SRM) [10]
  Differenza tra il valore di SRT in condizioni di co-locazione tra il segnale target e il segnale mascherante entrambi in asse con la testa dell'ascoltatore (0° fronte, 180° retro), e il valore SRT in cui il segnale target risulta ancora in asse e il segnale mascherante separato di una certa angolazione azimutale. Per esempio, se in figura 1.1a l'SRT80 è 10 dB e in 1.1b di 7 dB, allora l'SRM120° vale $10 - 7 = 3$ dB



(a) *Target e masker sono co-locati.* **(b)** *Masker a 120° azimuth dal target.*

Figura 1.1: Esempio di SRM 120°.

- Tempo di riverberazione (T_{60}) [7]
  È definito come il tempo necessario affinché il livello sonoro in un punto dell'ambiente decada di 60 dB dall'istante di spegnimento di una sorgente sonora che emette un segnale stazionario. Poiché un decadimento di 60 dB è un valore piuttosto elevato, si preferisce utilizzare livelli di decadimento inferiori e successivamente estrapolare il valore a 60 dB considerando un andamento lineare: si parla in questo caso di T_{30} o T_{20}, con livelli di decadimento rispettivamente di 30 e 20 dB.

- Intelligibilità del parlato - Speech intelligibility (SI) [11]
  È la percentuale di parole o frasi comprese rispetto totalità di quelle pronunciate durante una comunicazione verbale. La SI è influenzata da svariati fattori tra cui: caratteristiche vocali del segnale sorgente, distanza tra ascoltatore e parlatore, livello del rumore di fondo, presenza di eventuali segnali disturbanti, caratteristiche acustiche dell'ambiente. Nel corso del tempo sono stati definiti diversi indici di intelligibilità per consentire di caratterizzare gli ambienti in base alla loro destinazione d'uso.

- Speech Shaped Noise (SSN)
  Rumore bianco il cui spettro in frequenza è analogo allo spettro di un segnale vocale parlato di lunga durata.

- Dummy-Head
  È una testa *modello* che simula una testa umana utilizzata per effettuare misure acustiche.

- Head Related Transfer Functions (HRTF) [12] [13]
  È una funzione che descrive le variazioni che un suono, proveniente da una

determinata direzione, subisce a causa della conformazione della testa e del sistema uditivo. Le HRTF sono specifiche per ogni soggetto, tuttavia la loro misurazione avviene tramite microfoni inseriti nei condotti uditivi di appositi manichini o Dummy-Head.

- Risposta all'impulso dell'ambiente - Room Impulse Response (RIR) [14]
  È la funzione di trasferimento tra un segnale audio emesso da una specifica posizione in una stanza e la sua versione ricevuta in un'altra specifica posizione della stessa stanza. Rappresenta l'impronta acustica dell'ambiente, ogni stanza ha una propria RIR e ogni punto all'interno della stanza presenta una RIR unica.

- Digital Audio Workstation (DAW)
  Sistema elettronico destinato a registrazione, elaborazione, montaggio e riproduzione audio. Oggigiorno sono costituite da potenti workstation dotate di apposite schede audio e software dedicati.

- Ambiente sonoro virtuale - Virtual Sound Environment (VSE)
  Campo acustico riprodotto in modo tale da offire all'ascoltatore l'impressione di immersività spaziale.

- Ambisonico - Ambisonics
  Tecnologia in grado di registrare e di riprodurre con differenti livelli di fedeltà un campo sonoro immersivo, permettendo di creare un VSE. La successiva sezione 1.6 approfondirà l'argomento.

- Normal Hearing (NH)
  Soggetti con udito nella norma.

- Hearing Impaired (HI)
  Soggetti ipoacusici.

- Visual cue
  Stimoli, elementi, indizi visivi di qualsivolgia genere che possono o meno stimolare la percezione visiva dei soggetti sottoposti a test.

1.2 Accenni sui test di ascolto

Nei test di ascolto riguardanti la comprensione del parlato generalmente viene chiesto al soggetto di ascoltare un parlatore target (in cuffia o tramite coppie di altoparlanti) e ripetere una determinata sillaba o gruppo di sillabe, parole o frasi e, sulla base delle esatte ripetizioni, viene assegnato un punteggio. Un recente test di intelligibilità per la lingua italiana è "Italian Matrix Sentence Test" (ITAMatrix)

[15]. Questo tipo di test, sviluppato nel 2015 per scopi audiologici e di ricerca, prevede una matrice di 50 parole suddivise in 5 insiemi (nomi di persona, verbi, numerali, sostantivi e aggettivi) di 10 parole ciascuno. Le singole parole sono state selezionate tra quelle di uso più comune in modo da minimizzare eventuali influenze dovute alle competenze linguistiche dei soggetti e, inoltre, anche in base alla distribuzione dei singoli fonemi della lingua italiana. Selezionando una parola da ogni insieme della matrice così creata è possibile formare fino a 10^5 frasi differenti, corrette dal punto di vista sintattico, ma semanticamente non prevedibili, in modo da minimizzare eventuali intuizioni da parte dei soggetti. A puro titolo di esempio si riportano un paio di frasi:

Andrea trascina molte matite normali

Sara vede nove bottiglie utili

1.3 Test di ascolto ecologici

Negli ultimi anni la ricerca si è sempre più orientata verso approcci *olistici* per affrontare problemi e trovare soluzioni più efficaci ed efficienti nei diversi ambiti. Tale approccio si è rivelato particolarmente adatto per quelle scienze e per quei campi di ricerca in cui alle grandezze fisiche si affiancano i sensi e in generale i comportamenti umani che, come noto per esperienza diretta di ogni individuo, rispetto a un particolare stimolo possono essere facilmente influenzati da diverse condizioni e fattori esterni, siano essi fortemente o debolmente correlati allo stimolo in esame. Anche la ricerca audiologica si sta orientando verso tale tipo di approccio [16], in modo da coinvolgere fattori che altrimenti sarebbero esclusi dalle valutazioni.
Occorre dunque tenere in considerazione questi elementi di influenza, i cui effetti giocano un ruolo importante. Nel mondo della ricerca si parla di *ecological validity* o *validità ecologica*, l'enciclopedia Britannica ne propone la seguente definizione *"in psychology, a measure of how test performance predicts behaviours in real-world settings"* (in psicologia, una misura del modo in cui le prestazioni nei test predicono i comportamenti nel mondo reale) [17]; di conseguenza, quando nei test di ricerca sono coinvolti in qualche maniera i sensi e la sfera psicologico-cognitiva, vengono svolti quelli che in gergo si definiscono "test ecologici", ossia test che cercano di riprodurre al meglio la situazione reale in cui il soggetto andrebbe a trovarsi nel suo quotidiano. Si cerca, cioè, di sviluppare ambienti di test, anche detti scenari, con un'alta ecological validity in modo da ottenere una ricostruzione verosimile della situazione reale e allo stesso tempo un'ambiente di test controllato e ripetibile, che induca a risposte dei soggetti più veritiere e, di conseguenza, risultati più attendibili [18].
Nell'ambito della ricerca audiologica, i test ecologici risultano particolarmente

efficaci; l'ambiente sonoro è una realtà dinamica molto complessa in cui diversi elementi come rumori ambientali, interferenze, geometria del luogo, persone, apportano il proprio contributo. Tale complessità può essere ben modellata attraverso l'uso dei test ecologici: essi consistono nel ricreare in laboratorio il campo audio, e sempre più spesso anche video, che circonda il soggetto; permettono di combinare la necessità di controllo delle misure sperimentali di laboratorio con narrazioni più coinvolgenti e affini alla vita reale [19].
Precedenti studi dimostrano che questo approccio garantisce risultati più attendibili nella valutazione delle performance dei dispositivi acustici rispetto a test di ascolto più elementari [20]. Grazie a queste tipologie di test, i soggetti ipoudenti possono valutare i benefici apportati dall'uso degli apparecchi acustici in situazioni simili al reale facilitandone lo sviluppo e la loro messa a punto, superando le difficoltà delle precedenti metodologie che prevedevano dei semplici test di ascolto attraverso l'utilizzo di cuffie o altoparlanti.
Il recente studio [21] valuta la SI in condizioni controllate e realistiche attraverso l'utilizzo di un VSE grazie a un sistema ambisonico. La riproduzione sonora spaziale, tramite un array sferico a 64 canali, è del 7° ordine ambisonico. Lo scenario acustico è costituito da una registrazione ambisonica di una riunione d'ufficio, a cui vengono sovrapposte le frasi di test. I risultati ottenuti si sono dimostrati più aderenti alla realtà rispetto ai tradizionali test in cuffia: gli SRT su ipoacusici e normoudenti ottenuti erano, in media, 2-3 dB più alti (situazione peggiore) rispetto alla condizione di riferimento basata su cuffie, differenze dovute principalmente al riverbero provocato dalla spazializzazione del rumore di fondo. Gli autori, in conclusione, propongono che tale tipo di approccio possa essere ampliato fornendo anche informazioni visive in modo da aumentare ulteriormente il realismo dell'ambiente simulato.
È noto da tempo come l'intelligibilità del parlato migliori quando la sollecitazione sonora è accompagnata dal corrispondente stimolo visivo dell'oratore, sia in soggetti normoudenti che in soggetti audiolesi [22]. La vista gioca dunque un ruolo determinante nel processo di comprensione del parlato [23] e questo fattore non deve essere trascurato. L'aggiunta di immersività, grazie a registrazioni video 360° o ambientazioni VR, non può che aiutare il soggetto a ricondursi a una situazione reale rendendo più naturale la sua risposta agli stimoli e incrementando il livello di ecological-validity del test e di conseguenza le aspettative sulla correttezza e coerenza degli esiti delle prove.
Cospicue sono le ricerche con lo scopo di definire le caratteristiche dei test ecologici al fine di ottenere un buon livello di ecological-validity nel campo della ricerca audiologica, si citano a titolo di esempio [24, 25, 26].

1.4 Visual cues: quali sono significativi e impatto del labiale rispetto a visual cues più statici

La scoperta, nel recente passato, dell'effetto McGurk [23] ha dimostrato che la comunicazione verbale non è solo un fenomeno prettamente acustico e che la percezione del parlato dipende anche dagli stimoli visivi. Durante la comunicazione vengono scambiate, oltre a informazioni attraverso il canale uditivo, anche un'ampia gamma di indicazioni visive, come le espressioni del viso, il movimento labiale, lo sguardo, i gesti delle mani e altri tipi di linguaggio del corpo [27, pp. 477–485]. Per tenere conto di questo, nel corso del tempo sono stati messi a punto numerosi test con differenti tipologie di visual cue, come ad esempio contestualizzazioni visive dell'ambiente, visualizzazioni del volto dell'oratore, o riproduzioni più o meno complete di scenari complessi. Capire quali contributi visivi sono stati maggiormente utilizzati e valutati e quali possono essere i più efficaci e promettenti aiuta nella realizzazione di scenari di test con una migliore validità ecologica.
Tra i visual cue meno noti si evidenzia l'articolazione della lingua: recenti studi dimostrano la possibilità di apprendimento della lettura dell'articolazione della lingua, anche se meno preponderante rispetto a quella labiale, e una certa attitudine alla sua lettura in caso di stimolo acustico fortemente degradato o addirittura assente [28]. In questo studio, lo stimolo audiovisivo era rappresentato da un modello virtuale 3D di una testa. Con l'utilizzo di un dispositivo di articolografia elettromagnetica sono stati registrati, contemporaneamente all'audio, i movimenti articolatori dell'oratore. Successivamente, le registrazioni di questi movimenti sono state utilizzate per pilotare l'animazione della testa virtuale. Il segnale audio era costituito dai suoni naturali originali del parlato. Il test è stato svolto presentando ai singoli soggetti uno stimolo audiovisivo con audio in cuffia. Il compito del soggetto era quello di identificare il segnale audio che veniva loro presentato attraverso la visualizzazione della testa parlante in diverse configurazioni di "trasparenza" in modo da rendere visibile anche l'articolazione della lingua e i movimenti interni del tratto orale. L'audio veniva deteriorato con differenti valori di SNR, unica elaborazione sonora applicata. I risultati confermano la possibilità di apprendimento della lettura della lingua e, in particolare, una certa capacità naturale di leggere la lingua quando il segnale audio è fortemente degradato o assente: i soggetti hanno riferito che nella condizione di solo video (SNR = 1), la lingua era molto utile per riconoscere la consonante. Tuttavia, non viene evidenziata una differenza significativa tra la presentazione in spaccato della testa con la lingua visibile e la resa più naturale del volto completo.
La capacità di lettura della lingua può essere sfruttata per terapie logopediche e riabilitazioni di soggetti con problemi uditivi. A tal fine si cita la *testa parlante "Baldi"* [29], una sorta di tutor del linguaggio rappresentato dall'animazione di una

testa parlante in grado di mostrare l'articolazione esterna, interna e i movimenti interni del tratto orale grazie alla possibilità di rendere la pelle trasparente. Test di percezione e produzione del parlato in soggetti problematici hanno portato a risultati sodisfacenti confermando l'importanza dell'articolazione della lingua come visual cue.
Tuttavia, tra i visual cue maggiormente utili alla comprensione del parlato, il labiale, come dimostrato in [30], riveste un ruolo determinante essendo fortemente correlato e congruente allo stimolo acustico. L'abilità più o meno sviluppata delle persone nella lettura del labiale è una naturale dimostrazione di ciò. Il riconoscimento labiale viene sfruttato da sistemi per migliorare la comunicazione dei pazienti sottoposti a terapia intensiva [31] e svariati sistemi di lettura labiale automatica (ALR), anche noti come Visual Speech Recognition (VSR), sono stati sviluppati nel corso del tempo [32]. Pazienti affetti da disfunzioni uditive fanno uso dei visual cue congruenti per compensare il disturbo [33].
Lo studio [34] analizza i comportamenti di coppie di individui durante una conversazione in presenza di rumore e i loro accorgimenti messi in atto per compensare il disturbo. La ricerca indaga le strategie adottate nel parlato, nei movimenti e negli sguardi degli interlocutori. Per quanto riguarda lo sguardo, gli ascoltatori si sono concentrati sul volto dell'interlocutore per una media dell'88% di ogni prova. Le persone hanno trascorso una percentuale diversa di tempo concentrandosi sulla bocca rispetto agli occhi e la quantità di tempo dedicata alla bocca variava in base al livello di rumore: all'aumentare del livello di rumore, i partecipanti hanno dedicato meno tempo agli occhi del partner e più tempo alla bocca. Questi risultati sono ulteriore conferma dell'importanza dei visual cue e in particolare del labiale.
Nello studio "Auditory-visual scenes for hearing research" [24] vengono presentati tre ambienti audiovisivi virtuali che sono riproduzioni di altrettanti spazi reali (stazione della metropolitana, pub e soggiorno). Per ogni ambiente reale sono state generate due tipologie di modelli geometrici tridimensionali: uno accurato per la simulazione visiva e un modello geometrico semplificato adatto alla simulazione acustica. Inoltre, in ogni ambiente reale sono state effettuate delle registrazioni acustiche in modo che le scene possano essere ricreate anche dal punto di vista sonoro; vengono forniti tre tipi di misurazioni acustiche (omnidirezionali, ambisoniche e registrazioni con Dummy-Head). Questi scenari rappresentano situazioni di vita quotidiana in cui il riconoscimento del parlato può risultare difficoltoso. Nello studio non sono stati effettuati test di ascolto, la prerogativa era quella di presentare un framework in grado di consentire la simulazione di ambienti audiovisivi realistici e fornire visual cue capaci di trasmettere ai soggetti una idonea contestualizzazione dell'azione. L'inclusione delle misure acustiche garantisce la riproducibilità per gli studi di ricerca e la confrontabilità dei dati tra diversi enti.
I visual cue non solo influenzano la SI ma anche il modo in cui le persone muovono testa e sguardo. Questo fatto risulta particolarmente importante per i soggetti

portatori di protesi per i quali il movimento e la direzione della testa possono influenzare notevolmente la resa dei dispositivi. Queste influenze e i movimenti che ne conseguono sono stati analizzati nello studio [35], in cui i soggetti sono stati sottoposti a diverse situazioni di ascolto per confrontarne e valutarne il comportamento: solo audio, video di persone reali, personaggi animati con diversi modelli di sincronizzazione labiale e di sguardo. Lo scopo principale dello studio era di valutare se nei soggetti vi fossero differenze di comportamento tra le tipologie di stimoli e validare la possibilità di utilizzare animazioni in sostituzione di registrazioni reali. I risultati sono stati incoraggianti, hanno dimostrato l'importanza degli stimoli visivi nel rendere gli scenari più realistici e nell'indurre i soggetti a comportamenti più naturali, inoltre anche l'idoneità degli stimoli visivi artificiali sono stati convalidati. Una parte dello studio ha esaminato gli effetti dei visual cue sulla localizzazione della sorgente e sulla SI. Gli esiti hanno evidenziato che le condizioni visive in grado di aiutare la localizzazione della sorgente hanno influenzato positivamente la SI: a parità di SNR rispetto a situazioni in cui la localizzazione era difficoltosa, si è ottenuto un miglioramento di circa il 35% sulle risposte corrette.
Inoltre, l'importanza dei visual cue e in particolare delle espressioni facciali ha portato allo studio e all'implementazione di tecniche di analisi con diverse finalità, come per esempio il miglioramento di segnali audio degradati [36] e il riconoscimento delle emozioni [37].

1.5 Sistemi di registrazione e riproduzione idonei per test ecologici

Per quanto visto sull'importanza dei test ecologici e dei visual cue, la creazione e la riproduzione degli scenari utilizzati nei test di ascolto rivestono un ruolo fondamentale al fine di garantire una elevata validità ecologica [26]. Le ambientazioni immersive possono essere create attraverso la computer grafica o registrazioni video 360°, entrambe con pregi e difetti.
La computer grafica consente di creare scenari che possono essere riutilizzati, modificati e adattati a differenti scopi, offre la possibilità di poter generare delle ambientazioni complesse in cui ad esempio il soggetto può muoversi e interagire all'interno dell'ambiente. La fedeltà di riproduzione può variare notevolmente, da ambientazioni stile cartoon a situazioni indistinguibili dalla realtà, ovviamente con tutte le complicazioni, difficoltà e tempistiche del caso.
La registrazione video 360°, facilmente ottenibile anche a basso costo, restituisce un'alta fedeltà di riproduzione visiva con pochi artefatti eliminabili in post-produzione, ma non consente una reale interazione con l'ambiente. Il punto di vista soggettivo può ruotare in tutte le direzioni per consentire all'utilizzatore di

osservare l'ambiente circostante, ma al soggetto non è data piena libertà di movimento e interazione. Anche in caso di movimenti di camera permane la staticità del punto di presa che limita l'interazione: il fruitore non potrà far altro che seguire l'eventuale movimento di camera deciso dal regista, ma con la libertà di poter ruotare il proprio sguardo in ogni direzione. Una eventuale modifica comporta una nuova registrazione dello scenario. L'esperienza interattiva è dunque meno completa, ma la fedeltà di riproduzione visiva è di fatto assoluta. Nell'ambito di ricerca audiologica, tuttavia, questo limite non costituisce un problema, i test di ascolto rappresentano generalmente situazioni più o meno complesse ma statiche dal punto di vista del movimento del soggetto.
Entrambe le soluzioni possono essere facilmente fruite dal soggetto attraverso l'uso di dispositivi Head Mount Display (HMD) come i visori per sistemi di realtà virtuale.
Per quanto riguarda la registrazione e la riproduzione del campo sonoro è doveroso tenere in considerazione lo scenario proposto e adottare adeguati accorgimenti affinché il risultato sia adatto allo scopo e i visual cue coinvolti coerenti al contesto. Ad esempio, la registrazione del parlato in camera anecoica può essere usata, con le opportune elaborazioni, per simulazioni in ambienti riverberanti, ma se si desidera riprodurre uno scenario rumoroso questo segnale non è più idoneo. In situazioni rumorose, infatti, un oratore è soggetto all'effetto Lombard che lo induce a modificare il tono della voce e a enfatizzare il labiale [38], effetto che solo attraverso opportuni accorgimenti può essere indotto durante registrazioni in camera anecoica [26].
Il campo acustico è un'entità complessa che circonda completamente l'ascoltatore. La tradizionale capsula microfonica non è adeguata alla registrazione del campo audio e della sua spazialità: per catturare in toto l'ambiente sonoro occorre utilizzare particolari microfoni (array microfonici) e far uso di tecniche e formati audio adeguati. Nell'ambito della ricerca, il formato ambisonico è ampiamente diffuso e consente di registrare e riprodurre spazialmente il campo sonoro.

1.6 Formato ambisonico

Il formato ambisonico è un formato di codifica audio multicanale in grado di rappresentare il campo acustico con le sue caratteristiche spaziali nelle tre dimensioni. Il suo sviluppo si deve in particolare a Michael Gerzon e ai suoi studi sulla percezione uditiva e sulla spazializzazione sonora a partire dagli anni '70 del '900 [39]. La finalità della codifica ambisonica è di registrare un campo acustico e di riprodurlo in modo da ricreare un'esperienza immersiva completa per l'ascoltatore. L'approccio adottato da questo formato per la spazializzazione acustica è ritenere il punto di ascolto, detto *sweet-spot* in cui si ha una corretta riproduzione, al centro di una ipotetica sfera che rappresenta il campo sonoro.

I sistemi di diffusone spaziale più tradizionali, come ad esempio stereo, surround 5.1 o surround 7.1, sono limitati al piano orizzontale e si basano su uno schema fisso di altoparlanti ognuno dei quali ha un proprio segnale audio associato. Diversamente, la codifica ambisonica si basa sui principi fisici del campo acustico e sulla percezione uditiva. In ambisonico il campo sonoro viene scomposto in varie componenti che non sono segnali diretti agli altoparlanti, ma informazioni relative a determinate proprietà fisiche del campo che si desidera riprodurre, come pressione e gradiente di pressione [40]. Questa caratteristica rende la codifica ambisonica indipendente dal layout di altoparlanti usati per la diffusione e consente la riproduzione su svariati sistemi, siano essi stereofonici, cuffie o array di altoparlanti, ma rende necessaria una fase di decodifica per creare gli opportuni segnali con cui pilotare i diffusori. Maggiori sono i dettagli spaziali del campo acustico che si desidera registrare e riprodurre, maggiore è il numero di componenti in cui è necessario scomporre il campo. Data la sua rappresentazione sferica, la scomposizione del campo acustico avviene sulla base dello sviluppo delle funzioni armoniche sferiche. Il livello di tale sviluppo determina ciò che si definisce ordine ambisonico. Una più ampia espansione delle armoniche sferiche determina un elevato ordine ambisonico, cioè un elevato numero di segnali e di conseguenza una maggiore fedeltà del campo sonoro codificato. Con una certa semplificazione, queste componenti possono essere interpretate come i segnali acquisiti da microfoni altamente direzionali posizionati nel centro della sfera e orientati nello spazio secondo lo sviluppo delle funzioni armoniche sferiche.

1.6.1 Armoniche sferiche

Per completezza si introduce il concetto delle funzioni armoniche sferiche.
Le funzioni armoniche sferiche sono funzioni a tre dimensioni definite in un sistema di coordinate sferico e formanti una base ortonormale. Una combinazione lineare di tali funzioni consente di descrivere una generica funzione definita su una superfice sferica, in modo analogo come avviene con lo sviluppo in serie di Fourier per le funzioni periodiche nello spazio bidimensionale.
La figure 1.2 e 1.3 illustrano lo sviluppo delle funzioni armoniche sferiche fino al terzo ordine, corrispondente anche al medesimo ordine ambisonico. La prima è una rappresentazione come mappa di calore, in cui il colore rappresenta, su una superfice sferica, il valore assunto dalla funzione; la seconda è una rappresentazione tramite coordinate, dove il raggio assume il valore della funzione.

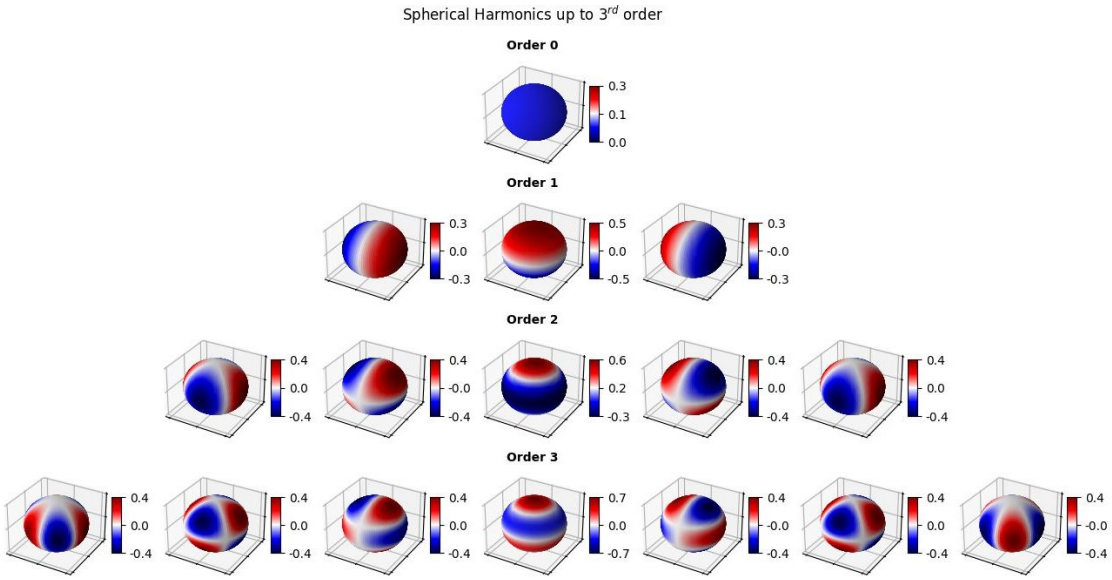


Figura 1.2: Rappresentazione a mappa di calore di armoniche sferiche su una superficie sferica.

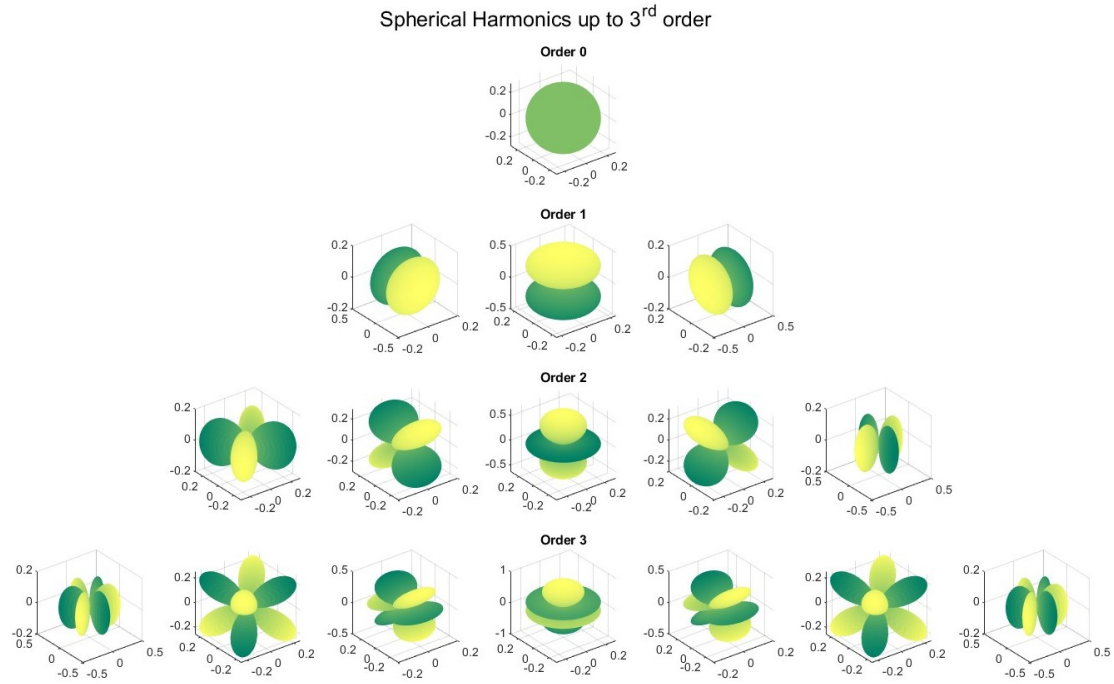


Figura 1.3: Rappresentazione come variazione di raggio di armoniche sferiche.

1.6.2 Ordine ambisonico

- **Ordine *0***
  Possiede un solo canale di codifica, chiamato W, che rappresenta le informazioni sulla pressione sonora del campo nel centro della sfera o punto di origine; di fatto è l'equivalente di una registrazione effettuata con un microfono omnidirezionale. L'armonica sferica corrispondente è la prima in alto in entrambe le figure 1.2 e 1.3.

- **Ordine *1*** o *First-Order Ambisonics (FOA)*
  Oltre al canale W possiede tre canali aggiuntivi X, Y, Z che contengono informazioni relative al gradiente della pressione acustica lungo le tre dimensioni spaziali. Equivale a una registrazione effettuata aggiungendo tre microfoni a forma di otto lungo le tre direzioni cartesiane. Questo è l'ordine Ambisonics maggiormente diffuso per applicazioni VR e video 360° (dove generalmente non è richiesta una elevata direzionalità). La codifica FOA di un segnale $s(t)$, avente una direzione angolare (ϕ, δ) rispetto al centro della sfera, nei canali W, X, Y, Z avviene in base alle seguenti relazioni [40]:

$$W(t) = \frac{s(t)}{\sqrt{2}} \tag{1.4}$$

$$X(t) = s(t) \cos\phi \cos\delta \tag{1.5}$$

$$Y(t) = s(t) \sin\phi \cos\delta \tag{1.6}$$

$$Z(t) = s(t) \sin\delta \tag{1.7}$$

  Il segnale $s(t)$ viene scomposto nella sua parte omnidirezionale $W(t)$ e nelle sue componenti lungo le tre direzioni principali $X(t)$, $Y(t)$ e $Z(t)$. Dalle immagini 1.2 e 1.3 risulta evidente come le armoniche sferiche del primo ordine si sviluppino secondo le tre principali direzioni spaziali.
  Questo modo di codificare i canali FOA è denominato formato-B. Esistono altri formati di codifica, come ad esempio il formato-A in cui i canali contengono il segnale grezzo registrato da un array microfonico le cui quattro capsule sono disposte ai vertici di un tetraedro.
  La figura 1.4 illustra il setup per realizzare una registrazione FOA con l'uso di 4 microfoni, uno omnidirezionale per il canale W e tre a forma di "8"per i restanti canali.

- **Ordine *2*** o *superiori*
  In questo caso si parla di Formato Ambisonico di Ordine Superiore - High Order Ambisonics (HOA): i segnali contengono informazioni sulle derivate di ordine superiore del campo acustico e descrivono informazioni maggiormente

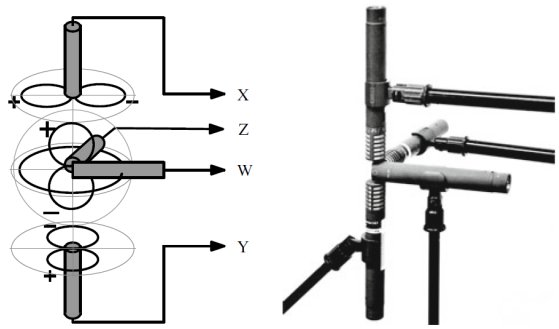


Figura 1.4: Registrazione ambisonica nativa 3D del primo ordine con un microfono omnidirezionale e tre microfoni a figura di otto allineati agli assi cartesiani X, Y, Z [41].

direzionali in grado di fornire una riproduzione spaziale più accurata del campo al crescere del numero di ordine. Il numero di canali di codifica è funzione dell'ordine l ambisonico in base alla relazione $(l+1)^2$, HOA di ordine 2 (2OA) avrà 9 canali, HOA di ordine 3 (3OA) avrà 16 canali e così via [40].

Una riproduzione fedele del campo sonoro dipende dunque dall'accuratezza della codifica, cioè dall'ordine ambisonico utilizzato in fase di registrazione; tuttavia, un'accurata emissione è funzione anche della geometria dell'array di altoparlanti utilizzato e della quantità di diffusori. Nonostante Ambisonics sia indipendente dal layout di riproduzione, un array sferico con disposizione regolare degli altoparlanti consente una decodifica più semplice rispetto a un array con forma arbitraria. La decodifica dei canali ambisonici in segnali adatti a pilotare i diffusori è resa necessaria dal fatto che il formato ambisonico è svincolato dal sistema di riproduzione. Tali segnali si ottengono attraverso opportuni filtraggi dei canali in funzione del sistema di diffusione. La numerosità di altoparlanti che dovrà costituire l'array di riproduzione è funzione dell'ordine ambisonico e, come per i canali di codifica, dipende dalla relazione $(l+1)^2$. Questa relazione non è ovviamente valida per una riproduzione in cuffia, la quale necessita di una decodifica più complessa al fine di ottenere il segnale binaurale adeguato [40, p. 23].

1.6.3 Registrazioni audio

La registrazione audio 360° è diffusa principalmente tra gli addetti ai lavori. Viene effettuata utilizzando molteplici capsule microfoniche organizzate in modo da costituire un unico dispositivo di registrazione. Queste tipologie di array microfonici hanno capsule orientate secondo diverse direzioni per consentire la registrazione del campo acustico circostante al punto di presa; un sistema di signal processing restituisce infine il segnale in formato ambisonico.

Registrazioni ambisoniche di primo ordine vengono effettuate con microfoni Soundfield in grado di restituire i 4 canali FOA. In genere sono costituiti da quattro capsule direzionali posizionate sui vertici di un tetraedro. Il formato grezzo registrato dalle capsule di questi dispositivi è definito "formato A". Tale formato viene convertito internamente al microfono, o tramite dispositivi esterni, in formato B per le successive elaborazioni [40, p. 10]
Registrazioni in HOA sono effettuate con dispositivi dotati di molteplici capsule microfoniche direzionali disposte su una superfice sferica. Anche in questo caso il formato di uscita è codificato nei canali ambisonici internamente o tramite l'uso di dispositivi esterni.
A puro titolo di esempio si citano alcuni dispositivi in commercio: Core Sound TetraMic™ (FOA), Core Sound OctoMic™ (2OA), Zylia PRO™ (3OA), em64 Eigenmike™ (6OA).

(a) *Core Sound TetraMic™ FOA.*

(b) *Eigenmike® em64 6° OA.*

Figura 1.5: Due array microfonici ambisonici.

1.6.4 Registrazioni video

Oggigiorno le registrazioni video 360 sono piuttosto diffuse, esistono sul mercato decine di soluzioni commerciali, dalla camera amatoriale a quella più professionale in grado di registrare il campo visivo più o meno nella sua interezza. Sono dispositivi costituiti da più camere che si occupano di registrare ognuna una porzione di spazio limitata; in seguito, il software tramite algoritmi di computer vision effettua l'operazione di *stitching* che si occupa di riunire tra loro le vari immagini registrate e correggerne le deformazioni. Il risultato finale è una "*volta celeste sferica*" su cui è rappresentato l'ambiente ripreso attorno al punto di osservazione. La figura 1.6 mostra la camera Insta360 Pro2™.
A puro titolo di esempio citiamo alcuni dispositivi in commercio: PIXPRO SP360

Action Camera™, GoPro MAX™, GoPro Fusion™, Insta360 Pro2™, Insta360 Titan™.

Figura 1.6: Insta360 Pro2™.

Capitolo 2

Stato dell'arte

Il presente capitolo espone lo stato dell'arte sulla ricerca della percezione del parlato. Vengono confrontati tra loro diversi studi, sviluppati con differenti metodologie e livelli di validità ecologica, con lo scopo di individuare sia i metodi di test più adeguati, sia in quale direzione si stia muovendo il campo della ricerca.

2.1 Confronti tra stimoli visivi e loro contributo all'intelligibilità del parlato

I visual cue utilizzati nei test di ascolto possono essere suddivisi in quattro macro-categorie, ognuna delle quali in grado di dare un contributo più o meno importante alla comprensione del parlato e più in generale alla ecological validity dei test. Sulla base di tale contributo è possibile individuare visual cue che:

- (i) permettono di contestualizzare l'ambiente di test e dare al soggetto il senso di immersività spaziale, oltre a una generica indicazione sulla qualità acustica dell'ambiente;
- (ii) consentono di localizzare la sorgente sonora permettendo al soggetto modificare il proprio orientamento e la propria attenzione;
- (iii) permettono la lettura del labiale;
- (iv) uniscono alla lettura labiale anche il linguaggio non verbale (del corpo nel suo insieme e non limitatamente alla sola mimica del volto) supportando maggiormente la comprensione.

Risulta chiaro come queste categorie possano essere presenti sia singolarmente, che in configurazioni intermedie in base alle finalità della ricerca e della validità ecologica che si desidera ottenere.

Per quanto riguarda la prima categoria, al momento non sono presenti studi che affrontano l'intelligibilità del parlato con test in cui l'unico stimolo visuale è costituito dall'immersività spaziale visiva. Quando presente, la contestualizzazione spaziale visiva è abbinata in contorno ad altri visual cue al fine di rafforzare l'aspetto ecologico del test. Studi in cui solo la contestualizzazione spaziale dell'ambiente viene considerata riguardano principalmente la percezione del riverbero e altri parametri ambientali come fatto in [42] ma, al momento, non l'intelligibilità del parlato.
Stimoli visivi di categoria (i) e (ii) vengono utilizzati nello studio [6]. SI, percezione della distanza e altre caratteristiche psicoacustiche sono state valutate in due scenari AV simulati che riproducono un ambiente anecoico e uno riverberante.
Per la simulazione audio è stato utilizzato il software di simulazione acustica RAZR [43] [44] mentre la parte visiva è stata realizzata con Unreal Engine 4 [45]. In entrambi gli scenari virtuali il soggetto si trovava al centro di una disposizione circolare di 8 altoparlanti, renderizzati nell'ambiente virtuale per rappresentare visivamente la posizione delle sorgenti sonore reali. L'ambiente riverberante era costituito da una stanza di grandi dimensioni scarsamente arredata, mentre l'ambiente anecoico era costituito da un paesaggio innevato. L'ambiente reale in cui i test sono stati svolti era costituito da un array sferico tridimensionale di 86 altoparlanti con al centro il soggetto; gli 8 altoparlanti che il soggetto vedeva nella scena virtuale coincidevano esattamente con 8 altoparlanti dell'anello orizzontale principale dell'array fisico. Il soggetto indossava un dispositivo HMD per la parte visiva. Secondo i ricercatori questa particolare configurazione e l'esiguo numero di altoparlanti virtuali simulati offre la facoltà di poter riprodurre realmente gli scenari proposti fornendo l'opportunità di un confronto tra reale e virtuale in un futuro ampliamento dello studio. Dai risultati si osserva che il riverbero ha un significativo effetto sulle misure percettive analizzate: riduzione dell'intelligibilità, minor accuratezza nella valutazione della distanza, minore capacità di valutare l'intensità acustica e un maggiore sforzo di ascolto rispetto a condizioni anecoiche. Il riverbero, dunque, contribuisce in modo importante alla complessità della scena sonora; inoltre, si osserva che la conoscenza a priori della direzione del target migliora le prestazioni dell'intelligibilità. In aggiunta è stata valutata l'influenza dell'utilizzo degli HMD sulle performance delle percezioni acustiche, concludendo che indossare questi dispositivi non apporta effetti negativi sulle prestazioni psicoacustiche. Tuttavia, nonostante la rilevanza di questi risultati, lo studio non affronta l'influenza del labiale sulla SI, aspetto di notevole importanza.
Nello studio [46] vengono indagate eventuali correlazioni tra la SI e il comportamento del movimento dei soggetti. I test si sono svolti in camera anecoica dove è stato simulato un ambiente moderatamente riverberante ($T_{20} = 900\ ms$) tramite un array di altoparlanti disposti a forma quadrata con lato $4{,}8m$. Il soggetto, posto al centro dell'array, indossava un cappello rigido che permetteva il tracciamento del

movimento grazie a un sistema di telecamere; davanti agli altoparlanti sono stati srotolati 4 schermi, trasparenti dal punto di vista acustico, per creare un ambiente immersivo a 360°. Gli stimoli visivi proiettati sugli schermi rappresentavano figure intere di persone intente a osservare verso il centro dell'array. Le figure, generate tramite il software MakeHuman [47], erano visualizzate alla naturale dimensione di una persona distante circa 2,2m dal centro dell'array, la loro posizione corrispondeva a quella da cui provenivano le frasi target. I soggetti sono stati sottoposti a tre condizioni sperimentali di ascolto: audiovisiva (AV), audio (A), e statica. Nelle prime due condizioni il soggetto era libero di muoversi senza però allontanarsi troppo dalla posizione centrale dell'array per non uscire dallo sweet-spot acustico, con la differenza che nella condizione audio non venivano proiettate le immagini sugli schermi. Nella condizione statica, anch'essa senza stimoli visivi, doveva stare al centro e non poteva muovere la testa. I risultati mostrano un leggero miglioramento nella SI quando il target è a 90° rispetto al segnale mascherante, dovuto principalmente alla separazione spaziale, ma non si nota un decisivo contributo migliorativo tra la condizione AV rispetto ad A. Tuttavia, nella condizione AV i partecipanti orientavano la testa direttamente verso il target, mentre nella condizione A la localizzazione risultava meno efficiente. Inoltre, i dati non suggeriscono che il movimento sia dovuto principalmente alla ricerca di un migliore SNR ai fini dell'intelligibilità, ma piuttosto che i modelli di movimento possano essere guidati principalmente dalla localizzazione della sorgente invece che dai processi di comprensione. Probabilmente questo comportamento può essere causato dalla qualità degli stimoli visivi utilizzati, lo studio difatti non tiene in considerazione la potenzialità del labiale che potrebbe avere un ruolo determinate sia sui modelli di comportamento che sulla SI.
Nello studio [48] i visual cue appartengono alla seconda categoria. Il soggetto è seduto al centro di un array circolare di 24 altoparlanti visibili, è a conoscenza del diffusore dal quale proviene il segnale target e può solamente ruotare la testa per variare la propria capacità di ascolto. Il livello del mascherante è fisso, mentre il livello del target viene diminuito di 7.5 dB ogni minuto partendo da un livello pari a 7.5 dB SNR. Il soggetto è libero di orientarsi come meglio preferisce ed è istruito a indicare quando non riesce più a seguire il discorso target. Il valore SNR raggiunto ha permesso di determinare una misura soggettiva di SRT correlato all'orientamento della testa. I valori ottenuti e opportunamente mediati sono stati utilizzati per validare il modello binaurale per l'intelligibilità del parlato nel rumore di Jelfs et al. (2011) [49]. Nonostante le misurazioni effettuate riguardassero la SI, quest'ultima non era l'argomento principale della ricerca.
Nello studio [50] l'esperimento 1 è privo di contestualizzazione spaziale e i visual cue utilizzati rientrano nelle categorie (ii) e (iii). Lo studio mira a dimostrare che una modesta rotazione della testa di 30° apporta un miglioramento dell'SRM rispetto a un parlato target frontale e contemporaneamente non influisce negativamente

sulla lettura labiale, consentendo anzi di ottenere benefici cumulativi. Il test si è svolto all'interno di due stanze insonorizzate, una all'Università di Cardiff e una all'University College di Londra. I partecipanti erano costituiti da un gruppo di giovani normoudenti, un gruppo di utilizzatori di BCI (impianto cocleare bilaterale) e uno di UCI (impianto cocleare unilaterale). Il soggetto è seduto su una sedia girevole, posizionata in modo che la testa si trovi al centro di un array di 4 altoparlanti disposti secondo i punti cardinali (rispetto al soggetto), inoltre è in grado di vedere l'altoparlante frontale da cui proviene il parlato target; l'ascoltatore può anche osservare il labiale dell'oratore grazie a un monitor da 17" che trasmette la registrazione del volto dell'oratore a grandezza naturale, posizionato immediatamente sotto all'altoparlante frontale. Per evitare che lo stimolo potesse divenire fastidioso, durante tutta la durata del test il livello sonoro complessivo è stato mantenuto a 65 dBA: l'aumento del SNR è stato ottenuto aumentando simultaneamente il livello del target e diminuendo il livello del mascherante. Per i test sono state selezionate queste cinque configurazioni spaziali: H_0M_0, H_0M_{180}, $H_{30}M_{180}$, H_0M_{90} e $H_{30}M_{90}$, dove i pedici indicano gli azimut della testa (H) e del mascheratore (M) rispetto al discorso target. Per ogni configurazione sono stati misurati valori SRT audio e audiovisivi. Il beneficio della lettura labiale, calcolato come SRT audiovisivo sottratto all'SRT solo audio nelle varie configurazioni, è stato di 3 dB per gli ascoltatori NH e di 5 dB per gli utilizzatori di Impianti Cocleari (IC), questi ultimi leggono meglio le labbra e dipendono maggiormente dalle indicazioni visive. L'aspetto più rilevante di questo studio è che una rotazione della testa di 30° non ha avuto alcun effetto negativo sulla lettura del labiale sui differenti gruppi, ma ha permesso di ottenere un beneficio cumulativo nella comprensione del parlato, con un incremento dei valori SRM sia nei soggetti normodotati che nei portatori di IC [48].

Anche nello studio [51] sono stati sfruttati stimoli visivi che consentono la lettura del labiale e la localizzazione della sorgente. Tuttavia, lo studio si incentra principalmente sulla localizzazione spaziale delle sorgenti sonore e sugli effetti che la localizzazione induce sulla percezione del parlato, ma è possibile osservare alcuni risultati interessanti ai fini dell'intelligibilità. Lo studio ha utilizzato un paradigma di realtà mista in cui gli stimoli uditivi sono stati presentati da un array di altoparlanti e gli stimoli visivi sono stati forniti attraverso un visore di realtà virtuale. L'array di altoparlanti era costituito da 26 dispositivi montati in un arco di 300° che circondava il partecipante a un raggio di 1 m e disposti in tre file di elevazione, ma per lo studio sono stati utilizzati solo quattro altoparlanti a 0° di elevazione rispetto al piano orizzontale delle orecchie e a -90°, -36°, 36° o 90° di azimut. Il test prevedeva la necessità di poter alterare la capacità dei soggetti di localizzare gli stimoli acustici. Tale effetto si è ottenuto facendo uso di un algoritmo di elaborazione in tempo reale [52] in grado di creare un'immagine spaziale distorta della posizione azimutale del segnale audio di riferimento.

Il sistema era costituito da microfoni binaurali montati su auricolari indossati dai partecipanti; i segnali sonori in arrivo dalle due orecchie sono stati acquisiti grazie ai microfoni binaurali, elaborati e poi presentati al soggetto attraverso gli auricolari. Gli stimoli visivi consistevano in quattro finestre quadrate, disposte orizzontalmente nello spazio virtuale del visore, su ognuna veniva visualizzato un filmato del volto dell'oratore. Grazie a un sistema di tracciamento della testa le quattro finestre virtuali sono state allineate con i quattro altoparlanti utilizzati per la diffusione audio. I partecipanti sono stati istruiti a girarsi e a rivolgersi verso l'oratore il più rapidamente e accuratamente possibile per ogni prova, ma è stato loro permesso di continuare a cercare di orientarsi verso il bersaglio durante l'intera prova. Nel primo esperimento il segnale target è stato mascherato da un rumore SSN co-locato e derivato dall'insieme delle frasi di riferimento in modo da avere uno spettro compatibile. Per valutare il contributo degli stimoli visivi i test sono stati divisi in due parti: una parte solo audio in cui gli stimoli visivi erano assenti: le finestre virtuali rappresentavano uno schermo nero e servivano solo come riferimento spaziale; l'altra in cui un filmato dell'oratore coerente con il target audio era visualizzato in una delle finestre virtuali, mentre nelle restanti tre venivano visualizzati alcuni video del medesimo oratore, ma scorrelati dall'audio. Entrambe le tipologie di test sono state svolte con valori SNR di $-15\ dB$ e $-10\ dB$. Nei soggetti in grado di localizzare correttamente il video target e di orientarsi verso la finestra virtuale corrispondente, il beneficio degli stimoli visivi ha portato a un incremento nel riconoscimento del parlato con una maggiore percentuale di parole individuate correttamente rispetto ai test solo audio: nel caso di SNR minore ($-15\ dB$) gli stimoli visivi apportano un maggiore contributo alla comprensione: i risultati sono migliori di circa il 35%; mentre con SNR pari a $-10\ dB$ la percentuale si abbassa a circa 25%.
Il secondo esperimento dello studio segue lo stesso paradigma di realtà mista e la stessa configurazione dell'ambiente di test, ma il segnale interferente è costituito da tre frasi pronunciate dal medesimo oratore del segnale target. L'esperimento consta di due parti. Nella prima parte il segnale audio di riferimento e i tre segnali interferenti sono co-locati, mentre per ogni finestra virtuale viene visualizzato un filmato relativo a uno degli stimoli acustici in esame. Nella seconda parte i segnali audio non sono più co-locati, ma distribuiti tra i vari altoparlanti, i video nelle finestre virtuali sono coerenti con l'audio nella rispettiva posizione, in modo da rappresentare una reale situazione con parlanti multipli. Anche questi due test sono stati effettuati in modalità solo audio e audio-visiva, entrambe con SNR di $0\ dB$ e $-10\ dB$. Anche in questa seconda parte, nei soggetti in grado di localizzare correttamente il video target e di orientarsi verso la finestra virtuale corrispondente, gli stimoli visivi hanno apportato benefici alla comprensione del parlato. Nel caso di sonoro co-locato la comprensione risulta migliore di circa il 35% per valori SNR di $0\ dB$, mentre per SNR $-10\ dB$ non si sono ottenuti risultati significativi. Nel

caso di suoni interferenti spazialmente separati la comprensione risulta migliore di circa il 12% con SNR di 0 dB, mentre con SNR -10 dB di circa il 25%.
Lo studio [53] sfrutta i primi tre tipi di stimoli visivi: contestualizzazione ambientale (tipo i), localizzazione della sorgente sonora (tipo ii) e lettura labiale (tipo iii). La ricerca è mirata a convalidare il funzionamento del sistema SEAT per test audiologici, un sistema che tiene conto del movimento della testa del soggetto; l'intelligibilità non è l'obiettivo primario. Il segnale audio veniva elaborato tramite una HRTF con modello di testa sferica e distribuito in cuffia. Lo stimolo visivo era fornito attraverso un HMD in cui veniva riprodotta una registrazione 360° di un interno di una caffetteria; tre monitor di computer erano posizionati su un tavolo all'interno della caffetteria. Per i test che prevedevano stimoli visivi del discorso target, il video dell'oratore veniva presentato sul monitor centrale, con gli altri due monitor spenti. I risultati ottenuti seguono le aspettative e sono coerenti con studi precedenti [54]. Per un segnale mascherante tipo SSN si ottiene un miglioramento dell'SRT di circa 4 dB, sia in caso di mascheratore co-locato che spazializzato (±40°). Nel caso di rumore mascherante costituito da due oratori il miglioramento risulta circa 8 dB nel caso di co-locazione e di 7 dB nel caso di mascherante spazializzato.
La ricerca continua a indagare nuovi e più efficaci metodi per migliorare la validità ecologica degli ambienti virtuali e degli ambienti di test utili allo sviluppo e allo studio di protesi acustiche. Il recente studio [55] dimostra che il movimento della testa apporta benefici alla comprensione del parlato in ambienti virtuali. Il test valutava lo sforzo di apprendimento di una serie di conversazioni intermediate da un sistema di VR, in cui tre soggetti fisicamente distanti interagivano attraverso un monitor e ascolto in cuffia. Il compito dei partecipanti era di ingaggiare una conversazione generica. L'ambiente acustico era elaborato tramite il software TASCAR [56] per riprodurre l'acustica di un pub. Il test prevedeva l'interazione attraverso quattro livelli di realismo visivo. La condizione con il livello più basso di realismo era costituita dalle teste statiche degli avatar stilizzati dei soggetti; nel secondo livello le teste degli avatar erano animate dalle battute del discorso dei tre interlocutori; il livello successivo di realismo è stato raggiunto animando gli avatar direttamente tramite i movimenti della testa dei soggetti; infine, il livello più alto di realismo è stato ottenuto attraverso una ripresa video dell'interlocutore inserita direttamente nella scena VR. Ai soggetti è stato chiesto di indicare su una scala di valutazione a 7 valori la loro difficoltà nella comprensione del discorso durante i vari esperimenti: i risultati ottenuti indicano che lo sforzo di comprensione del discorso è stato via via più semplice con l'aumentare del realismo degli stimoli visivi.
Uno degli approcci più innovativi al problema della SI è affrontato nello studio [57]. La ricerca propone l'uso di riprese 360° 3D integrate con registrazioni audio ambisoniche 3OA, questa scelta permette di offrire al soggetto una immersività piuttosto spinta e più aderente alla realtà sia dal punto di vista visivo che sonoro. Lo

scenario selezionato è una tipica sala conferenze con un elevato tempo di riverbero ($T_{20} = 3{,}2s$ circa). Sono state definite sette scene con diverse posizioni di ascolto, diverse posizioni del parlato target e diverse posizioni delle sorgenti di rumore di mascheramento; per ognuna delle sette scene AV sono stati registrati i video omnidirezionali stereoscopici (3D) nelle diverse posizioni di ascolto e registrate le corrispondenti RIR 3OA spaziali. Per aumentare l'immersività alcuni attori erano presenti come spettatori all'interno della sala conferenze. I test di ascolto sono stati eseguiti presso il laboratorio Audio Space Lab del Politenico di Torino, una piccola sala insonorizzata in cui è installato un sistema di riproduzione audio 3OA sincronizzato con un visore HMD per creare ambientazioni VR immersive. Il sistema audio è costituito da 16 altoparlanti disposti in modo omogeneo a formare un array sferico che circonda la posizione di ascolto a 1,2 m di distanza e da 2 subwoofer per le basse frequenze. La sincronizzazione della riproduzione audiovisiva è coordinata da uno script Matlab® che pilota due ulteriori software: la DAW Reaper [58] per la diffusione audio Ambisonics e il motore grafico Unreal Engine [45] per la riproduzione video attraverso il visore Oculus Meta Quest 2. I test di SI sono stati somministrati sia modalità solo audio che audiovisiva e durante la prestazione i soggetti non potevano muovere la testa per non variare la condizione di ascolto. In entrambe le condizioni sono state presentate le sette scene previste, facendo ascoltare per ogni scena 14 frasi di test. Le frasi di test, costituite da 3 parole, provenivano dal test validato "Simplified Italian Matrix Sentence Test (SiIMax)" [59], una versione semplificata del più completo test ITAMatrix [15]. I risultati ottenuti sono in linea con i fondamenti della letteratura e dimostrano la validità del metodo proposto dallo studio; tuttavia, i ricercatori concordano sul fatto che la metodologia di test debba essere estesa consentendo ai soggetti una più ampia capacità di movimento e, in particolare, migliorando la parte visiva con l'introduzione degli stimoli labiali, che in questa fase sono stati tralasciati per semplificare la ricerca.

2.2 Stimoli visivi sintetici

Da quanto si è potuto osservare, gli stimoli labiali hanno assunto costantemente una maggiore importanza negli studi sull'intelligibilità: come conseguenza si è introdotta un'ulteriore complessità che sarebbe opportuno considerare nella gestione dei test di ascolto. In particolare, i vari visemi possono presentare differenze, seppur minime, sia in uno stesso oratore che tra oratori differenti, introducendo una variabilità non sempre desiderata per un rigoroso approccio scientifico ai test di ascolto. A tal proposito, un aiuto potenzialmente molto promettente potrebbe essere fornito da software in grado di sintetizzare i movimenti labiali: questo approccio eliminerebbe di fatto la variabilità tra singoli visemi e offrirebbe l'opportunità per una certa

automazione dei contenuti, consentendo sia di migliorare la ripetitività dei test e sia di ampliarne la varietà, a beneficio dei risultati e dei pazienti.
Allo stato attuale sono disponibili diversi software in grado di generare movimenti labiali; a partire da un contenuto audio vocale questi software, basati su tecniche di intelligenza artificiale, sono in grado di manipolare l'immagine o i fotogrammi video di un volto, o la fisionomia di un modello 3D garantendo una corretta sincronia tra il fonema e il visema generato. Per la loro complessità molti sono prodotti commerciali, ma esistono anche soluzioni disponibili senza oneri.
Una breve rassegna degli strumenti open-source o comunque liberamente utilizzabili per scopi non commerciali o di ricerca, mette in evidenza le caratteristiche dello stato dell'arte in questo ambito.

2.2.1 Wav2Lip

In questo studio [60], pubblicato nel 2020, gli autori affrontano la sincronia labiale di un generico volto, umano o umanoide, rispetto a un generico segnale vocale in una generica lingua; il software è in grado di individuare e seguire la posizione del volto e di sincronizzare i visemi con i fonemi della traccia vocale e può operare sia su video che su immagini, creando in quest'ultimo caso un video ex novo. Il video o l'immagine processati vengono manipolati nella zona labiale, lasciando inalterata la parte restante.
Il funzionamento è basato su una rete neurale antagonista generativa (GAN Generative Adversarial Network) addestrata sul dataset pubblico LRS2 [61]. Il dataset ha una durata complessiva di 225 ore e contiene migliaia di frasi, pronunciate da differenti oratori in differenti inquadrature, tratte dai programmi della televisione britannica BBC, ma nonostante ciò il sistema è in grado di operare con qualsiasi linguaggio. La bontà delle immagini generate è garantita dall'architettura della rete che prevede due distinti discriminatori: uno dedicato alla valutazione della sincronia e della forma dei visemi generati (lip-sync discriminator), l'altro dedicato alla valutazione dell'immagine nel suo complesso (visual quality discriminator) in modo da minimizzare la presenza di artefatti nell'immagine di sintesi migliorandone la qualità.
In base ai risultati, i video generati con Wav2Lip sono preferiti per oltre il 90% rispetto a quelli creati con altri metodi [60] e il modello Wav2Lip supera nelle metriche quantitative gli altri approcci con un ampio margine [60]. Inoltre, sempre secondo gli autori la sincronia labiale ottenuta è paragonabile a quella dei video reali.
Le potenziali applicazioni sono molteplici: i doppiaggi di film e serie TV possono avere una migliore sincronia labiale, rendendo più semplice e aderente all'originale il doppiaggio e più gradevole la visione nelle altre lingue; i discorsi pubblici e

le conferenze stampa in diretta possono subire una traduzione simultanea e contemporaneamente subire l'aggiornamento dei movimenti labiali rispetto alle varie traduzioni migliorandone coinvolgimento e comprensione. Sul sito web del progetto sono presenti una serie di video dimostrativi [62].
Il codice è liberamente disponibile sul repository GitHub del progetto [63]. In questa versione le reti neurali sono addestrate con immagini a bassa risoluzione, di conseguenza anche le elaborazioni finali lo sono. Modelli neurali migliorati e con risoluzioni più elevate sono disponibili commercialmente [64].

2.2.2 Live Speech Portraits

Questa ricerca [65] pubblicata nel 2021 propone una architettura di deep learning per affrontare il problema della sincronizzazione labiale tra un generico segnale vocale e il volto di un generico oratore, sia esso un'immagine statica che un video. Anche in questo caso il funzionamento è indipendente dalla lingua del segnale vocale.
Rispetto a [60] in cui viene esclusivamente manipolata la zona labiale, in questo caso l'immagine finale viene rigenerata in toto. Secondo gli autori questo approccio evita possibili conflittualità tra la parte visiva e il nuovo contenuto audio, permettendo una maggiore libertà nella scelta della nuova traccia vocale usata per la sintesi. Inoltre, per cercare di mantenere inalterate le dinamiche di conversazione specifiche della persona, viene utilizzato un breve filmato di 3 minuti della persona dal quale vengono ricavate le pose della testa e della parte superiore del corpo, in modo da aumentare il realismo del video finale.
Nel complesso si individuano tre fasi, governate da reti neurali. La prima consente di estrarre le caratteristiche del segnale vocale. La seconda fase predice, sulla base delle caratterisitche individuate, l'intera dinamica del movimento ed è costituita da due reti neurali: una dedicata al movimento labiale e una alla posa della testa e della parte superiore del corpo. La terza fase si occupa della sintesi finale del fotogramma sulla base delle precedenti predizioni e di un set di immagini candidate, derivate dal breve filamto di 3 minuti. Il risultato finale è un flusso di animazione fotorealistico del volto che include espressioni facciali e movimenti sia della testa che della parte superiore del corpo, guidate dall'audio e in tempo reale. Da notare che i fotogrammi vengono in ogni caso ricreati ex novo sintetizzando sia la persona che lo sfondo, anche nel caso in cui venga elaborato un video e non una semplice immagine della persona.
L'apprendimento è stato effettuato usando la sezione in lingua cinese mandarina del dataset multilingua Common Voice [66], per un totale di 889 oratori e 26 ore complessive.
Rilevante è la capacità del sistema di mantenere lo stile espressivo della persona, la qual cosa non è presente in altri lavori, come dichiarato dagli autori.

Un aspetto per certi versi negativo è la necessità di un pre-addestramento della rete con il video di 3 minuti della persona, azione invece non necessaria in [60]. Anche in questo caso le possibilità applicative riflettono quanto accennate in [60]. Il codice implementativo è liberamente accessibile al sito web del progetto [67].

2.2.3 Audio2Face

Audio2Face [68] è un software commerciale *production-ready* sviluppato da NVIDIA, utilizzato da molte industrie dell'intrattenimento per l'animazione facciale dei personaggi, che, a differenza dei precedenti, manipola una mesh 3D del volto e non i fotogrammi di un filmato. Applicazioni legate a traduzione simultanea e videoconferenze non rappresentano il suo appropriato dominio applicativo, ma la sua peculiarità di operare su modelli 3D apre altri ambiti di utilizzo, come ad esempio VFX cinematografici e applicazioni in ambienti immersivi virtuali. Alla base del progetto vi è la seguente ricerca [69], il cui scopo era quello di generare animazioni facciali 3D plausibili ed espressive basate unicamente su una traccia audio vocale. Per ottenere movimenti facciali realistici il sistema manipola tutto il volto, non solamente la parte labiale, grazie all'uso di una rete neurale convoluzionale (CNN Convolutional Neural Network) opportunamente addestrata.
Data una breve finestra della traccia audio, il compito della rete è quello di dedurre l'espressione facciale al centro della finestra; le espressioni vengono rappresentate come differenza della posizione dei vertici della mesh 3D a partire da una sua posa neutrale. La rete è suddivisa concettualmente in tre parti. La prima, *formant analysis network*, analizza la finestra vocale di input e individua le caratterisitiche del segnale come accenti, fonemi, intonazioni, utili alla guida dell'articolazione labiale. La seconda, *articulation network*, analizza l'evoluzione temporale delle caratteristiche individuate dalla sezione precedente e determina la posa facciale al centro della finestra temporale di ingresso. Inoltre, questa sezione accetta come input secondario un *descrittore dello stato emotivo*, composto da una serie di parametri, utile a discriminare eventuali ambiguità tra diverse espressioni e stili di parola. La terza sezione, *output*, ha il compito di produrre le posizioni 3D finali dei vertici della mesh.
L'addestramento è stato condotto ricreando le mesh 3D del volto di due attori attraverso un sistema di registrazione con 9 telecamere, in modo da catturare con alta qualità tutti i movimenti facciali durante la prestazione. Data la complessità di questa fase la durata delle animazioni è stata limitata a 5 e 3 minuti per attore. Il loro compito era di enunciare un pangramma[1], in lingua inglese, con differenti stili emotivi in modo da coprire una buona gamma di espressioni facciali; questo

[1]Breve composizione di testo in cui sono presenti tutte le lettere dell'alfabeto.

ha permesso alla rete di apprendere i parametri del descrittore dello stato emotivo. Anche in questo caso l'addestramento in lingua inglese non ha precluso la capacità del sistema di operare in altri idiomi.
Il software dispone di numerosi parametri di configurazione per raggiungere l'espressività desiderata nei movimenti facciali, e la sua capacità di trattare anche la parte emotiva è sicuramente un punto di forza. In aggiunta, Audio2Face è in grado di analizzare le emozioni di una performance audio e applicarle alle animazioni facciali del personaggio.

2.3 Osservazioni conclusive

Allo stato attuale non esistono studi approfonditi che integrano filmati 360° 3D, stimoli visivi completi di labiale e registrazioni di audio spazializzato. Questo abbinamento consente di riprodurre scenari di test con un'elevata validità ecologica, più aderenti alle condizioni reali, in grado di offrire ai soggetti la libertà di un comportamento più naturale.
Questa tesi si propone di estendere la ricerca [57] descritta nella sezione 2.1 integrandola con la parte labiale, tralasciata in un primo momento per semplicità di studio. In questa ricerca il soggetto si trova immerso in una sala conferenze caratterizzata da una difficile acustica a causa di un elevato tempo di riverbero (T_{20} 3,2s); il suo compito è ripetere le parole di un parlatore target situato frontalmente a 4 e 8 metri di distanza; le condizioni di test sono variabili e prevedono sia l'ascolto in quiete che con parlatori interferenti situati a 120° e 180°.
Degli scenari descritti, quelli in cui la distanza tra soggetto e parlante target è pari a 4 m saranno arricchiti, tramite compositing video, sia con un'attrice che con un avatar realistico che enunceranno le frasi del test ITAMatrix fornendo così lo stimolo labiale desiderato. L'avatar verrà animato tramite il software Audio2Face.
I test aggiornati saranno successivamente sottoposti a una popolazioe stocasticamente compatibile con quella impiegata per le indagini precedenti, in cui gli stimoli labiali erano assenti. Infine, saranno confrontati i risultati delle indagini, in assenza e in presenza di labiale, in modo da avere un riscontro del contributo apportato dalle espressioni facciali e dall'articolazione delle labbra nella comprensione del parlato, sia per quanto riguarda il labiale reale, sia per quanto riguarda il labiale di sintesi.
Sebbene l'uso di [60] o di [65] possa sembrare a prima vista più immediato e adatto allo scopo, una serie di motivi hanno fatto propendere all'uso di Audio2Face. [60] nella sua versione non commerciale lavora a una bassa risoluzione, mentre [65] necessita di un addestramento preventivo con un breve video del personaggio che si desidera animare, la qual cosa potrebbe essere non sempre disponibile; inoltre entrambi necessitano di un video esistente su cui operare la manipolazione labiale.

Audio2Face ha la caratteristica di essere *production-ready* e quindi industrialmente diffuso, il fatto di lavorare su modelli 3D lo rende versatile e particolarmente adatto ad applicazioni per ambienti immersivi con la possibilità di poter creare pressoché infinite situazioni senza necessità di filmati pregressi o realizzati ex-novo, semplificando il processo di produzione.

Capitolo 3

Integrazione del labiale reale

Il capitolo descrive come sono stati realizzati i test con labiale reale, dal concetto base alle riprese, dalla post-produzione al taglio delle singole clip per garantire un'appropriata sincronia tra audio e video, fino alla creazione dei filmati complessivi per la somministrazione dei test di ascolto.

3.1 Quadro generale

Le scene ecologiche che saranno integrate con gli stimoli labiali sono state registrate presso la sala conferenze del Museo Egizio grazie al precedente lavoro di tesi [70]. In questo scenario, i pregressi test di intelligibilità, tutti con audio ambisonico e video stereoscopico 360°, prevedono diverse configurazioni del punto di presa, che si ricorda essere il punto di vista del soggetto sotto esame, rispetto all'ipotetico parlatore target. Per le finalità del presente studio sono state utilizzate le ambientazioni in cui la distanza tra il punto di presa e l'oratore target è pari a 4,1 m: nella ambientazioni in cui la distanza è maggiore la comprensione dello stimolo labiale risulterebbe praticamente impossibile vanificando lo scopo. Si è dunque deciso di effettuare una ripresa alla stessa distanza di 4,1 m e una a distanza ravvicinata di 1.8 m che risultasse utile sia per lo scenario che riproduce l'aula universitaria 1T del Politecnico di Torino, anch'essa disponibile grazie al lavoro di tesi precedente [70], sia per ulteriori ambientazioni.
L'idea di base per ottenere i movimenti labiali è stata di operare una sorta di "doppiaggio" sulle frasi del test ITAMatrix. Ogni frase del test è stata riprodotta per quattro volte, la prima riproduzione consentiva all'attrice di ascoltare le parole e assimilarne la cadenza, le tre successive fungevano da base di riferimento affinché l'attrice potesse ripetere la frase parlandoci sopra al medesimo ritmo. Delle tre ripetizioni è stata scelta quella ritenuta più sincrona con l'audio originale. Le frasi su cui si è lavorato erano un sottoinsieme di 120 frasi tra tutte quelle disponibili

nel set ITAMatrix. Per aiutare ulteriormente l'attrice, le frasi venivano di volta in volta visualizzate su uno schermo a parete tramite un proiettore.
Per selezionare il movimento labiale più conforme all'audio originale, si sono dovuti affrontare due problemi: (i) valutare quello ritenuto maggiormente coincidente; (ii) selezionare ed estrarre dal filmato solo la parte utile. Per (i), ciascuna delle tre persone attive sul set (esclusa l'attrice) aveva il compito di osservare l'esecuzione dell'attrice e annotare quale delle tre ripetizioni riteneva maggiormente credibile: quella che otteneva il punteggio maggiore era selezionata, in caso di parità veniva selezionata la prima esecuzione; (ii) è stato risolto con un'elaborazione audio della traccia sonora ottenuta in fase di registrazione. Per l'elaborazione è stato scritto un apposito programma Python [71].

3.2 Produzione

Per l'integrazione del labiale sono state individuate diverse macro-fasi di seguito elencate:

- analisi degli scenari esistenti, studio delle inquadrature e dei set di ripresa e equipaggiamento;
- analisi degli stimoli sonori esistenti, studio di un sistema "prompter" in grado di agevolare il compito del locutore e di consentire la successiva sincronizzazione tra le clip video e le tracce audio utilizzate per i test;
- realizzazione delle riprese audiovisive, valutazione del miglior *doppiaggio* per ogni frase;
- post-produzione e effetti visivi (VFX) per la realizzazione degli scenari comprensivi di oratore;
- selezione delle clip video con l'apporto labiale conforme e loro taglio dai filmati complessivi;
- creazione delle sequenze per i test di ascolto a partire dalle singole clip video.

A tutto questo va aggiunto un certo livello di automazione, esigenza particolarmente sentita per la fase di estrazione delle singole clip. Questo consentirebbe di ottenere una maggiore omogeneità dei tagli, una migliore sincronia audio/video oltre ad agevolare futuri ampliamenti degli scenari.

3.2.1 Scenari, set di ripresa, equipaggiamento

I set di ripresa sono stati studiati in modo da mantenere coerenza visiva tra il locutore e gli scenari in cui sarà aggiunto.

Le riprese video sono state effettuate con la videocamera Insta360 Pro 8K messa a disposizione dal Visionary Lab del Politecnico di Torino: è la stessa utilizzata per le registrazioni degli scenari in modo da garantire uniformità dei nuovi filmati con quelli esistenti. Il Visionary Lab ha fornito anche il resto del materiale e della strumentazione.
Idealmente, per ogni scena sarebbe opportuno mantenere le stesse configurazioni di ripresa originali, lasciando inalterate distanze e posizioni tra camera e soggetti, in modo da evitare alterazioni prospettiche degli elementi nel compositing delle scene. Un leggero ingrandimento o riduzione dell'elemento da integrare può essere accettabile per variarne la percezione di distanza nella nuova scena; tuttavia, se ad esempio l'elemento si trova su un piano orizzontale differente rispetto a quello di presa, cioè si trova più un alto o in basso rispetto all'obiettivo, tale posizione deve essere rispettata pena una percezione distorta da parte dell'osservatore. Tenendo conto di ciò, si è cercato di configurare i set di ripresa in modo da ottenere filmati idonei a essere integrati nei differenti scenari con trascurabili alterazioni prospettiche.
Le riprese sono state effettuate in due sole configurazioni, la prima adatta allo scenario della sala conferenze del Museo Egizio e la seconda adatta all'aula 1T. Nella configurazione per il Museo Egizio il parlante target si trova su un palco rialzato di 30 cm e a 4,1 m di distanza rispetto all'ascoltatore in platea che deve rivolgere lo sguardo leggermente verso l'alto per osservarlo. Nella configurazione prevista per l'aula 1T invece, target e soggetto si trovano alla stessa altezza a una distanza di circa 2 m, il soggetto osserva il parlante target dritto di fronte a sé, senza la necessità di elevare lo sguardo. Quest'ultima impostazione è adatta anche per gli ulteriori scenari previsti: una caffetteria e una camera anecoica [70].
Occorre precisare che è proprio questa differenza nelle altezze tra parlante target e soggetto che ha reso necessaria una doppia configurazione del set di ripresa. Questa scelta evita anomalie nelle prospettive tra l'ambiente esistente e il parlante target da integrare, che avrebbero influito negativamente sulla percezione visiva del soggetto sotto test. Sul piano orizzontale, invece, la distanza tra target e soggetto è meno influente sulla percezione della prospettiva, e l'immagine del locutore si può adattare in post-produzione operando, entro certi limiti, un ridimensionamento.
Per lo studio dei set si è fatto uso dello strumento on-line SketchUp [72]. Le figure 3.1 e 3.2 illustrano rispettivamente il layout per il set che andrà a integrare lo scenario *Museo Egizio* e lo scenario *caffetteria*.

Un limbo green screen portatile ha funto da sfondo. La configurazione della camera Insta360 Pro è stata impostata in maniera analoga alle riprese esistenti per ottenere un video 3D stereoscopico 6K (6400x6400) a 29,97fps, successivamente ridotto alla risoluzione di 4K (3840x3840) a causa di limitazioni del visore. Come si può notare dai due layout, i set differiscono nell'altezza della camera e nella sua distanza dal soggetto. Le immagini rappresentano un layout di massima, i due fari messi a lato del soggetto, per riprodurre l'illuminazione degli scenari originali

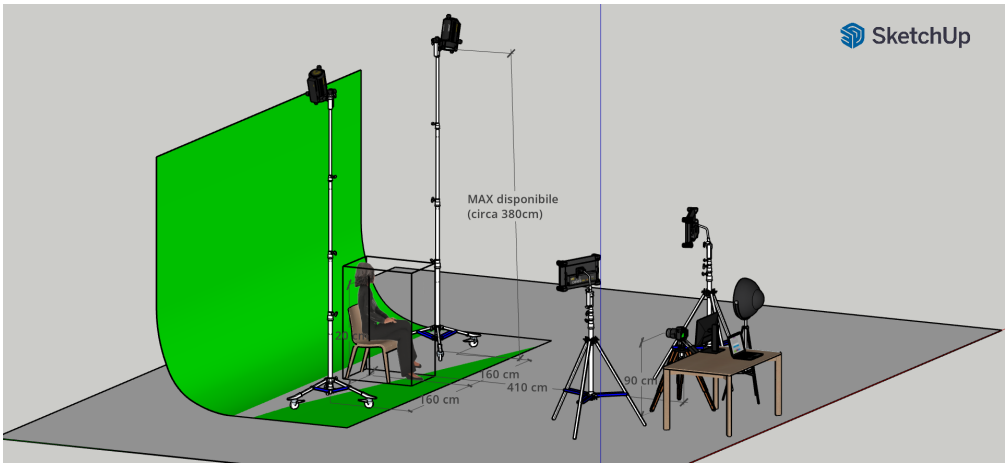


Figura 3.1: Schema set di riprese per scenario *Museo Egizio*.

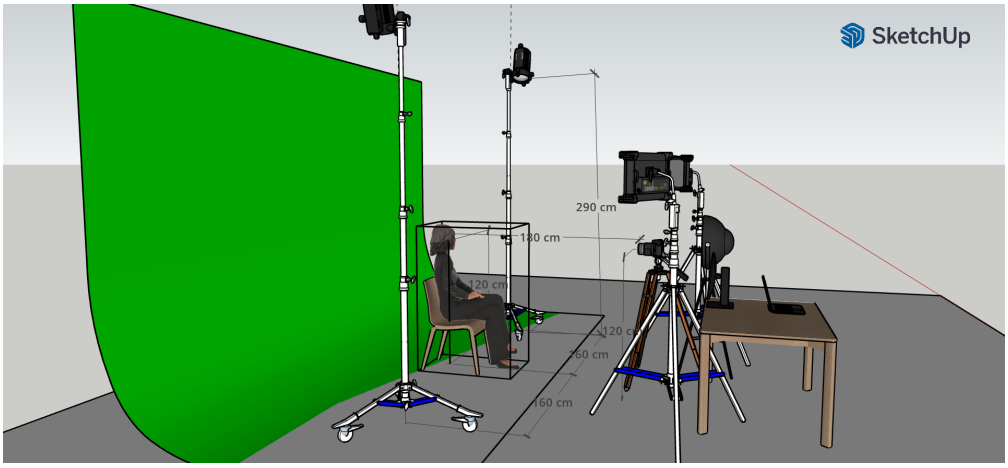


Figura 3.2: Schema set di riprese per scenario *caffetteria*.

non erano funzionanti al momento delle riprese, ciononostante la luce della sala è risultata sufficiente per ottenere un discreto risultato. Inoltre, per visualizzare le frasi da enunciare si è utilizzato, in sostituzione del monitor, il sistema di proiezione disponibile nell'aula 8N del Politecnico di Torino dove si sono svolte le riprese.

3.2.2 File audio ITAMatrix: trascrizione e sistema di prompting

Lo studio del sistema di prompting ha rappresentato una fase critica del processo. Oltre a facilitare il compito dell'attrice doveva consentire, nella fase di post-produzione, il corretto taglio delle scene per garantire la sincronia tra labiale

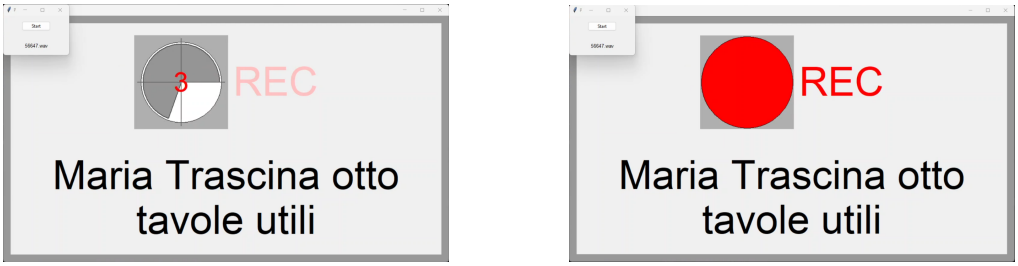
e sonoro. L'obiettivo è stato ottenuto operando su due fronti, visivo e acustico, in modo da avere una ridondanza dei segnali di riferimento. Per la trascrizione delle tracce audio e la codifica del sistema di prompting è stato utilizzato Python, linguaggio dotato di numerose librerie e adatto per una prototipazione rapida.

Trascrizione tracce audio

Complessivamente erano disponibili 120 tracce anecoiche, un sottoinsieme del set ITAMAtrix. Per la trascrizione delle singole frasi, necessarie per il sistema di prompting, si è codificato il programma Python *speech2text.py* consultabile in appendice A.1. Il programma fa uso della rete neurale "OpenAI-Whisper" [73], un modello di riconoscimento vocale generico in grado di operare in modalità offline.

Sistema di prompting

La parte grafica del sistema di prompting è stata creata tramite la libreria standard Tkinter [74]. L'animazione aveva inizio visualizzando il testo e un conto alla rovescia di 3 secondi per introdurre la riproduzione della traccia audio di riferimento; alla sua conclusione veniva visualizzata la sequenza di un conto alla rovescia di 3 secondi al termine del quale veniva mostrato un pallino rosso e la scritta *REC*, in modo da rendere evidente il momento in cui ripetere il testo e avere un chiaro riferimento visivo sincronizzato con l'audio; questa parte veniva ripetuta per tre volte. Dopodiché si passava alla frase successiva. Per la parte audio si è scelto di riprodurre, al termine di ogni conto alla rovescia, un tono puro a 443 Hz di durata 0.3 secondi seguito da 0.15 secondi di silenzio dopo il quale aveva inizio la riproduzione della traccia audio anecoica. La riproduzione del tono era sincrona con la visualizzazione del pallino rosso per la parte visiva. L'aggiunta del tono 443 Hz ha consentito di avere un punto di riferimento sonoro facilmente individuabile per le successive elaborazioni. Per questa parte si è fatto uso della libreria audio Librosa versione 0.10.1 [75]. Per non affaticare troppo l'attrice, il prompter è stato realizzato per riprodurre le frasi a gruppi di 12 per un totale di 10 sessioni di ripresa per set. Ogni sessione prevedeva 12 locuzioni riprodotte 4 volte per un totale di 48 proposizioni e una durata di circa 5 minuti. Le figure 3.3a e 3.3b mostrano due fasi del prompter.



(a) *Fase di conto alla rovescia.* **(b)** *Fase di registrazione.*

Figura 3.3: Due schermate del prompter.

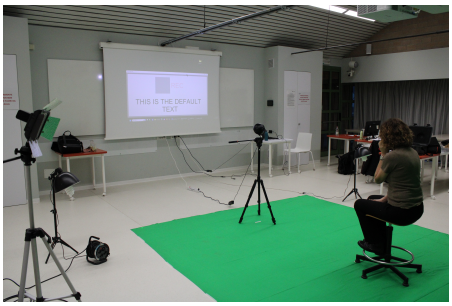
3.2.3 Esecuzione delle riprese

In mancanza di uno teatro di posa le registrazioni sono state effettuate presso l'aula 8N del Politecnico di Torino; oltre all'attrice, sul set erano presenti 3 persone per lo svolgimento delle riprese e di tutte le attività connesse. Le figure 3.4a e 3.4b illustrano il set di ripresa.
A causa del malfunzionamento di una coppia di fari, il green screen è stato posizionato in modo da sfruttare al meglio la luce presente in aula. Come green screen è stato utilizzato un limbo portatile di dimensioni 6x3m, posizionato in modo tale che l'attrice potesse osservare lo schermo a muro del videoproiettore.
L'impianto di videoproiezione multimediale è stato utilizzato sia per la parte visiva che per quella audio del sistema di prompting. Non essendo necessaria una registrazione sonora di alta qualità, per l'audio si è fatto uso dei microfoni interni della Insta360 Pro.
La preparazione del set, l'esecuzione delle riprese e lo smontaggio del set hanno richiesto circa quattro ore.
Durante le riprese le tre persone presenti sul set avevano il compito di osservare attentamente l'esecuzione del doppiaggio e indicare su un modulo la performance ritenuta maggiormente in sincronia con le tracce anecoiche riprodotte.
Complessivamente sono stati registrati 20 filmati ottenendo un centinaio di minuti di girato.

3.3 Post-produzione

Per quanto riguarda la post-produzione si possono identificare le seguenti fasi: stitching, VFX per realizzazione chroma-key e compositing sullo scenario, selezione e taglio delle clip, realizzazione del video finale.

(a) *Particolare del limbo portatile.*



(b) *Particolare del sistema di prompting.*

Figura 3.4: Due viste del set di ripresa.

3.3.1 Stitching

La procedura di stitching, letteralmente *cucitura*, è un processo gestito attraverso diversi parametri dal software *Insta360 STITCHER* fornito dal produttore della camera. Lo stitching consiste nell'unire tra loro le porzioni video riprese dai sei obiettivi fisheye del dispositivo per ottenere un video a 360°. Il video risultante è nel formato "equirettangolare" con rapporto di aspetto 2:1, editabile dai tradizionali editor video NLE (Non Linear Editing). Inoltre, il software consente di creare video stereoscopici fruibili tramite visore, costituiti da due formati equirettangolari, uno con la proiezione per l'occhio destro e uno con la proiezione per l'occhio sinistro. In questo caso si può decidere se sovrapporre le proiezioni e ottenere un video con rapporto 1:1 (configurazione Top-and-Bottom) o metterle adiacenti e avere un video con rapporto 4:1 (configurazione Side-by-Side). Dato che il filmato deve essere inserito in un video esistente sono state mantenute le stesse impostazioni: la proiezione per singolo occhio ha risoluzione 3840x1920 e le proiezioni sono disposte in modalità Top-and-Bottom, con la proiezione sinistra sopra a quella destra ottenendo un video finale a risoluzione 3840x3840. L'immagine 3.5 mostra un fotogramma di una sessione di riprese, ottenuto a seguito dell'operazione di stitching, limitato al solo occhio sinistro.

3.3.2 VFX e compositing

Per la parte di editing è stato utilizzato il software DaVinci Resolve Studio 18 [76]. In prima istanza si è realizzato il chroma-key dell'attrice. La porzione di sfondo esterna al green screen e una parte di esso sono stati eliminati attraverso una maschera di garbage-matte. La parte di video restante, cioè l'attrice su fondo verde, è stata trattata a zone: testa, busto, gambe, per ottimizzare la chiave colore e i parametri di selezione del green screen; tutta l'area rimossa è stata resa trasparente.

Figura 3.5: Fotogramma monoscopico di una sessione di riprese a seguito dell'operazione di stitching.

Infine, i video sono stati renderizzati con il codec intermedio Voukoder ProRes 4444 XQ, disponibile in DaVinci come plug-in esterno, in modo da non perdere qualità per il compositing successivo. Nella figura è illustrato lo schema a nodi di DaVinci Resolve Fusion usato per l'editing descritto. La simmetria che si nota è dovuta alle due proiezioni per occhio destro e sinistro che vengono trattate con gli stessi effetti.

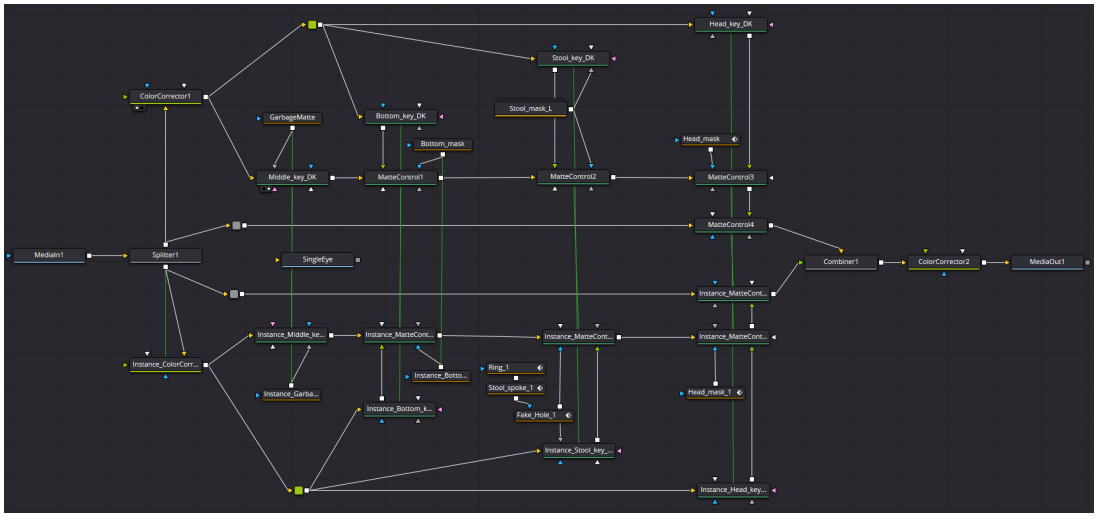


Figura 3.6: Flusso di elaborazione del Node Editor di Fusion.

(a) *Porzione del fotogramma da trattare con chroma-key.*

(b) *Immagine pronta per il compositing nello scenario.*

Figura 3.7: L'attrice, prima (a) e dopo (b) l'elaborazione del chroma-key.

L'ultimo step per realizzare i test di intelligibilità è stato di integrare i 10 video così ottenuti nelle tre differenti scene del Museo Egizio in cui locutore e soggetto si trovano a una distanza di 4,1 m. Sebbene la distanza di ripresa sia la stessa, le tre scene variano tra loro per alcuni particolari legati all'ambiente sonoro che si è voluto simulare. Nel primo scenario SC1M non esistono rumori mascheranti e viene semplicemente simulato l'audio della sala senza alcun disturbo; nel secondo scenario SC2M il rumore mascherante proviene da un'angolazione di 120° rispetto all'azimut del soggetto; nel terzo scenario SC3M il disturbo proviene dalle spalle del soggetto, con azimuti di 180°. Nelle ultime due scene il rumore mascherante è rappresentato dalla presenza del manichino Dummy-Head.
La parte di compositing finale ha richiesto le seguenti operazioni: rimozione del diffusore NTi Audio Talkbox usato per diffondere le frasi target e del corrispondente stativo; reframe del video dell'oratore per posizionarlo in maniera adeguata; creazione di un mascheramento per il tavolo, affinché l'oratore risultasse seduto dietro; creazione di un effetto "riflessione" a causa del vetro presente nella parte bassa del tavolo; estensione della durata complessiva dello scenario per adattarlo ai video dell'attrice che enuncia le frasi target; color-correction per uniformare i filmati. Infine, rendering di produzione con codifica video H.264 per l'importazione su Unreal Engine. La traccia audio registrata in fase di ripresa è rimasta invariata. Il risultato finale è costituito da 30 video, 10 per ogni scenario. Ogni scenario comprende i labiali delle 120 frasi di test.

3.3.3 Selezione e taglio delle clip video

Insieme al sistema di prompting ha costituito una fase piuttosto critica, dato che la sincronia finale tra labiale e audio dipendeva profondamente da entrambe. Il problema è stato affrontato facendo uso di Python, le librerire Librosa e FFmpeg [77], e PowerShell [78]. Per semplicità, le librerie FFmpeg non sono state utilizzate direttamente nel codice Python, ma attraverso la loro interfaccia a linea di comando PowerShell. Il programma Python è stato utilizzato per analizzare i filmati, ricercare le esecuzioni ritenute più sincrone con le tracce anecoiche di base e individuare i corretti tempi di taglio e, infine, per creare gli script PowerShell contenenti i comandi FFmpeg con gli opportuni parametri per il taglio delle varie clip.

Il listato del programma *seek_lips_sync.py* è disponibile in appendice A.2. Di seguito viene descritto il suo funzionamento generale.

A ognuna delle due configurazioni di ripresa (4,1 m e 1,8 m) viene associata una matrice 10x12, le 10 righe rappresentano le 10 sessioni di registrazione, le 12 colonne le 12 frasi per sessione. Gli elementi della matrice contengono l'indice della ripetizione, fra i tre tentativi di doppiaggio, ritenuta con la migliore sincronia labiale per ognuna delle 12 frasi anecoiche. Questo valore è stato ottenuto semplicemente a maggioranza confrontando le tre valutazioni effettuate durante le riprese, in caso di pareggio è stata selezionata la prima ripetizione. Nonostante il codice sia generico, al momento vengono processate solo le tre scene del Museo Egizio SC1M, SC2M, SC3M. Per ogni sessione di ripresa, a partire dai valori della matrice viene calcolato l'indice sequenziale del corrispondente tono di riferimento a 443 Hz che precede la ripetizione scelta, con la formula:

$$tone_{idx} = 4 \cdot sentence_i + best_{rec} \tag{3.1}$$

dove:

$tone_{idx}$: numero sequenziale del tono 443 Hz da ricercare

$sentence_i$: frase i-esima della sessione di registrazione (colonna della matrice)

$best_{rec}$: numero della ripetizione con migliore sincronia fra i tre tentativi

Attraverso il calcolo della STFT vengono individuati sull'asse temporale i 48 toni di riferimento a 443 Hz e in seguito i loro fronti di salita. Successivamente, con la formula 3.1 vengono selezionati i fronti di salita corrispondenti alle 12 frasi ritenute con la migliore sincronia labiale.

Le figure 3.8 e 3.9 mostrano rispettivamente lo spettrogramma e un suo dettaglio della traccia audio registrata dal microfono della Insta360 Pro. Il rettangolo rosso mette in evidenza l'area intorno al tono di riferimento a 443 Hz, quello giallo la zona dei 200 Hz dove si nota lo spettrogramma relativo alle varie enunciazioni. La parte bassa sopra 0 Hz rappresenta rumore.

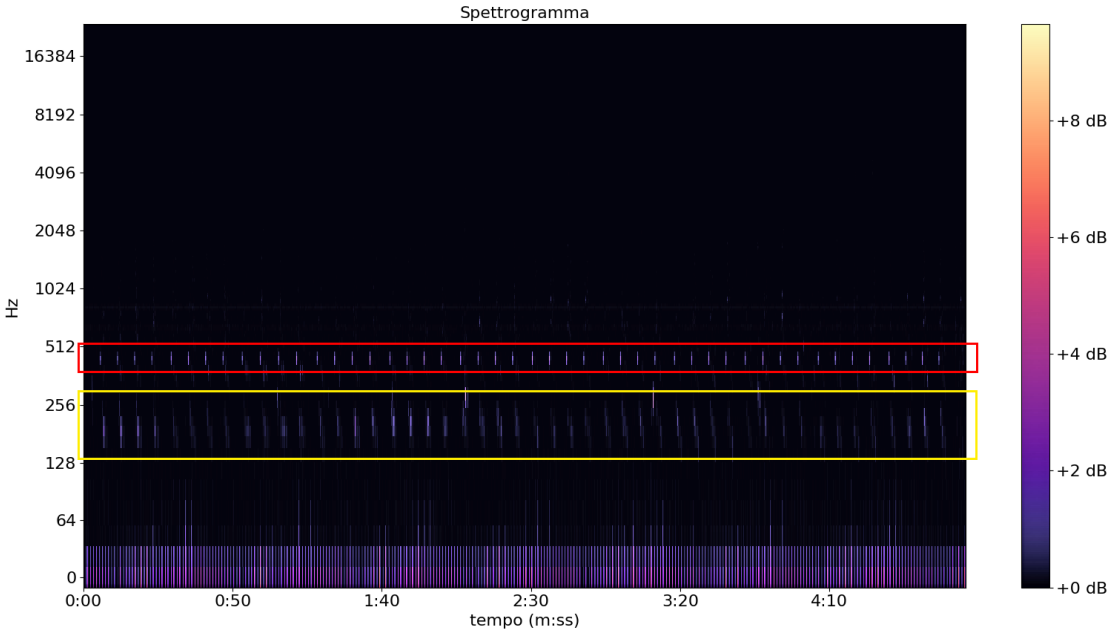


Figura 3.8: Spettrogramma della traccia audio di una ripresa.

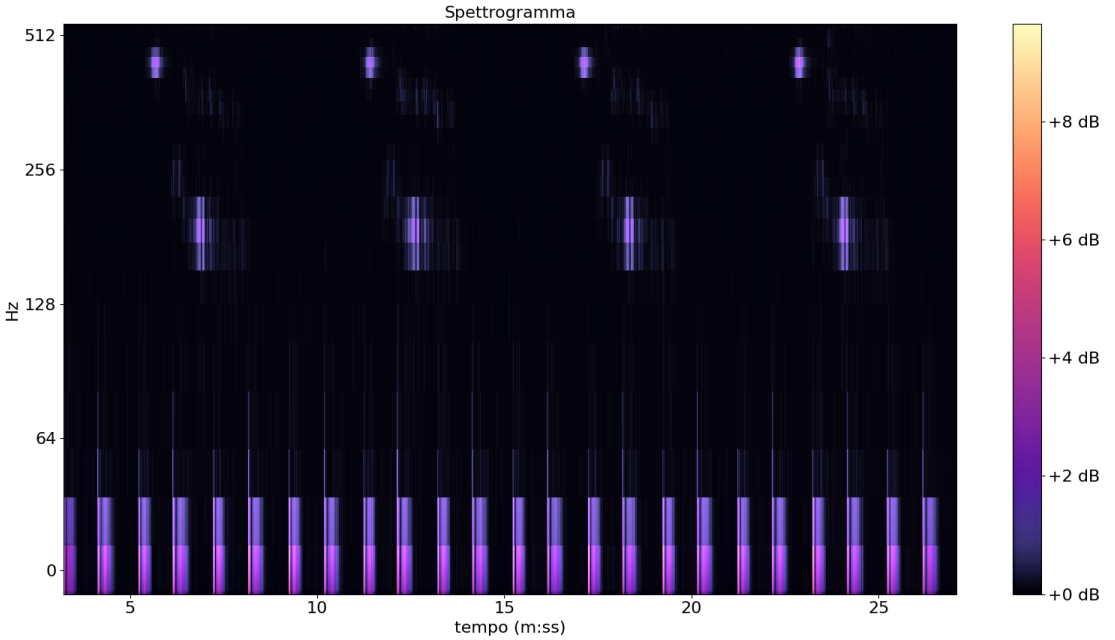


Figura 3.9: Spettrogramma, dettaglio. Nella parte alta è visibile lo spettro di 4 toni di riferimento a 443 Hz, intorno ai 200 Hz lo spettro di 4 enunciazioni.

Si sono scelti come riferimento i fronti di salita perché sono risultati più affidabili rispetto, ad esempio, al picco del tono: il segnale audio analizzato è quello registrato dai microfoni della camera InstaPro 360, pertanto il tono ricercato non è la sinusoide pura con ampiezza costante generata dal sistema di prompting, ma è un tono che ha subìto inevitabili distorsioni dovute alla catena elettroacustica e che contiene le riflessioni del tono stesso dovute all'ambiente, oltre al rumore. Il contributo di tutte le riflessioni si somma ed è massimo verso la fine del tono, generando un picco di massimo, ma ciò non è sempre verificato, a causa del rumore possono esserci spike spuri anche nella parte iniziale o mediana. Invece, il fronte di salita trattandosi di una ripida variazione risulta di più facile individuazione e meno soggetto ai disturbi da rumore e alle riflessioni.

La figura 3.10 mostra l'andamento della STFT per la banda contenente il tono di riferimento a 443 Hz, mentre la 3.11 raffigura un dettaglio: fronte di salita, picco del tono e picco selezionato sono messi in evidenza dai rispettivi marker.

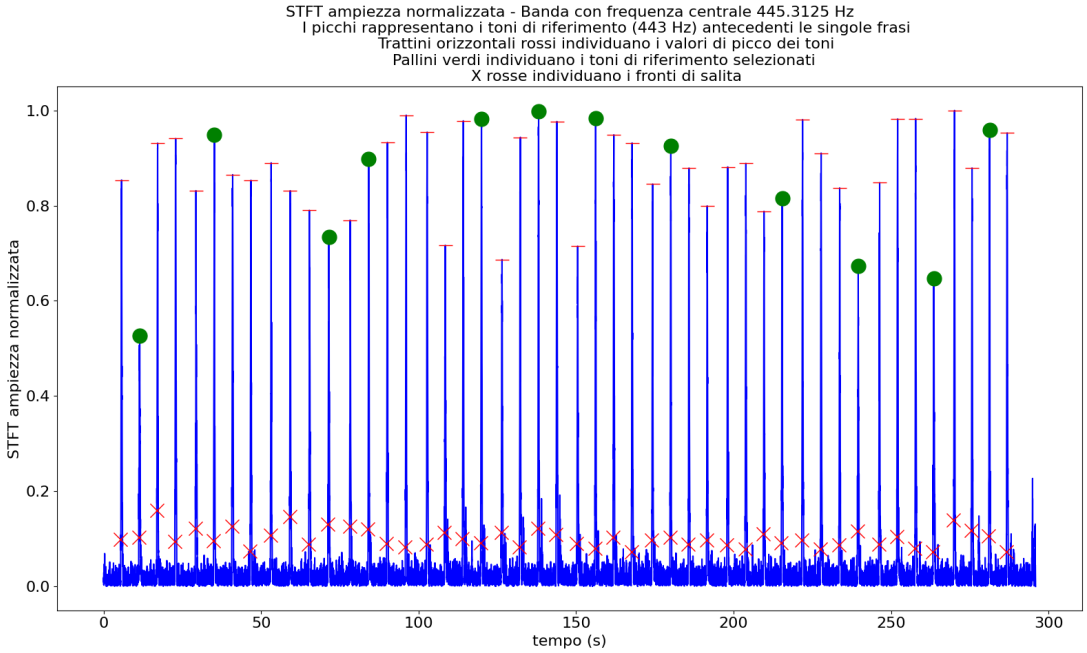


Figura 3.10: STFT banda tono di riferimento.

Individuati gli istanti dei fronti di salita, sulla base alle tempistiche definite nel sistema di prompting, vengono determinati i tempi di inizio delle corrispondenti tracce anecoiche; successivamente si ricavano il corretto punto di taglio iniziale sottraendo 2 secondi dall'istante del tempo di salita e, per il punto di taglio finale, aggiungendo la durata della traccia anecoica più ulteriori 2 secondi. I due secondi iniziali e finali sono dettati dalle tracce auralizzate ambisoniche: il segnale target somministrato ai soggetti è sia preceduto che seguito da 2 secondi di silenzio, o di

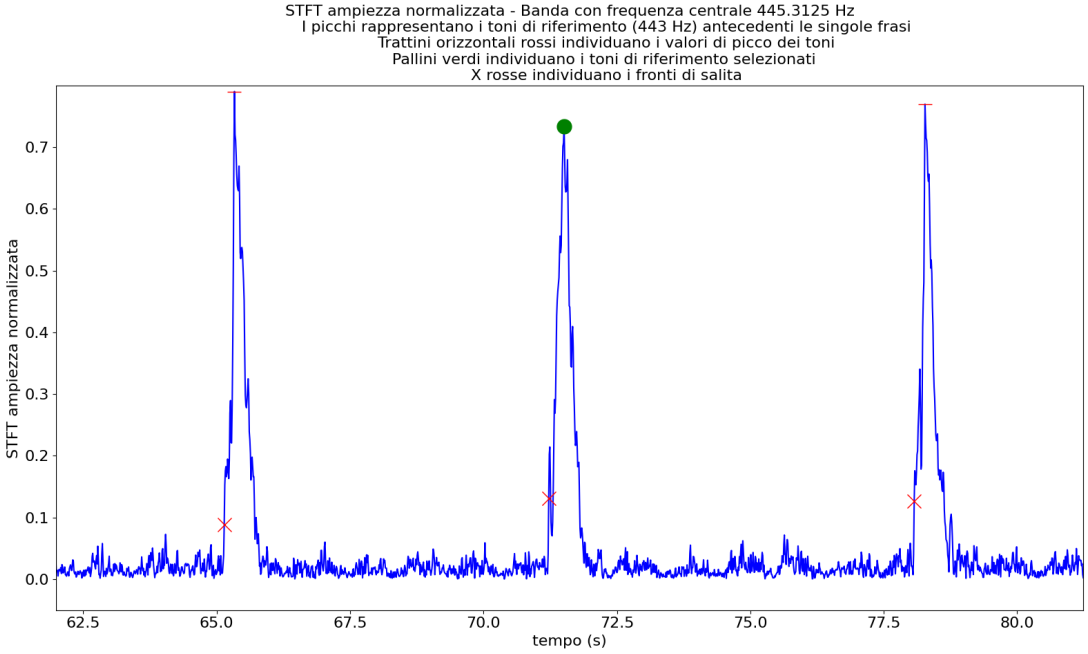


Figura 3.11: STFT banda tono di riferimento, dettaglio.

rumore nel caso lo scenario lo preveda. Dai valori ottenuti vengono composti gli opportuni comandi FFmpeg che saranno visualizzati in output.

Il taglio effettivo, per ottenere le singole clip adatte ai test di ascolto, è stato fatto eseguendo uno script PowerShell creato dall'output del programma Python. Questa soluzione può sembrare alquanto macchinosa, ma consente di avere un maggiore controllo sulle operazioni in atto e un debug più agevole rispetto all'esecuzione diretta dal codice Python. Uno stralcio del codice PowerShell è consultabile in appedice A.3, il codice riporta il taglio delle prime due frasi.

Al termine del processamento, per ognuna delle scene SC1M, SC2M, SC3M, si sono ottenute 120 clip, una per frase, per un totale complessivo di 360 spezzoni video.

3.3.4 Video finale

Il video finale usato per la somministrazione dei test di ascolto è stato realizzato concatenando tra loro alcuni degli spezzoni ottenuti precedentemente. La scelta delle clip è stata effettuata in base agli attuali test esistenti, costituiti da una serie predefinita di frasi differenti per le tre scene. Per concatenare tra loro le singole clip è stato utilizzato uno script PowerShell e le librerie FFmpeg. Per completezza si riporta di seguito il listato del codice PowerShell e non del file di testo che è un semplice elenco di file comprensivi di percorso.

```
$ordine_scene_conference_hall = 'E:/compositing/410_SC2M_comps/
    SC2M_conferece_hall_clips_list.txt'

ffmpeg -f concat -safe 0 -i $ordine_scene_conference_hall -c:v copy
    SC2M_full_test_ffmpeg.mp4
```

Infine, i filmati finali sono stati integrati con gli opportuni metadata video 360, necessari affinché vengano correttamente riconosciuti e riprodotti dai vari player e dispositivi. Questa operazione è stata eseguita tramite il programma liberamente disponibile *Spatial Media Metadata Injector* [79]. Le figure 3.12 e 3.13 mostrano rispettivamente il risultato finale in formato monoscopico e un suo dettaglio.

Figura 3.12: Risultato finale in formato monoscopico dello scenario SC2M con attrice e rumore mascherante a 120° impersonato dal manichino Dummy-Head.

Figura 3.13: Dettaglio del risultato finale con attrice.

Capitolo 4

Integrazione del labiale sintetico

Il capitolo descrive come sono stati realizzati i test con labiale di sintesi, dalla creazione dell'avatar alla sua animazione, dalla sua integrazione negli scenari al taglio delle singole clip per ottenere la corretta sincronia tra audio e video, fino alla creazione dei filmati complessivi per la somministrazione dei test di ascolto.

4.1 Quadro generale

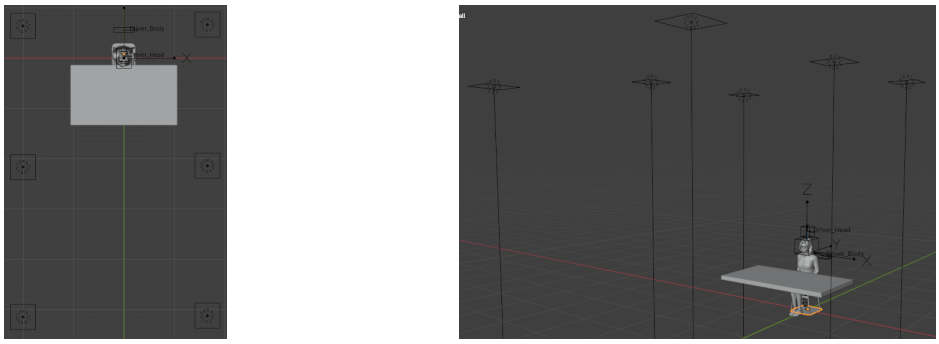
Le stesse tre scene ecologiche, ambientate nella sala conferenze del Museo Egizio, viste nel cap. 3 saranno integrate con stimoli labiali di sintesi grazie all'animazione di un avatar. Il segnale vocale è stato composto concatenando tra loro le singole tracce anecoiche. Tramite Audio2Face e Blender [80] l'avatar è stato animato, rispettivamente per l'articolazione delle labbra e per i movimenti secondari di testa e busto. In Blender è stato realizzato un set virtuale, coerente con la sala conferenze del Museo Egizio e renderizzata l'animazione dell'avatar su sfondo trasparente. In seguito, si è provveduto al compositing nelle tre scene e successivamente al taglio delle singole clip tramite una serie di script Python. Infine si è passati a realizzare i video finali per la somministrazione dei test di ascolto.

4.2 Produzione

4.2.1 Set virtuale

Il set virtuale ricrea un contesto, coerente con sala conferenze del museo, in cui l'avatar è collocato in posizione seduta. Questa ambientazione è utile per generare un'illuminazione congruente con la corrispondente parte reale: luci e ombre saranno

esportate unitamente all'avatar per il successivo compositing affinché l'avatar risulti *naturalmente* presente nella scena finale. Il tutto è stato modellato in scala per rispettare le proporzioni reali, inoltre solo gli effettivi elementi che concorrono alla generazione delle luci e delle ombre sono stati modellati nell'ambiente virtuale.



(a) *Pianta.* **(b)** *Prospettiva.*

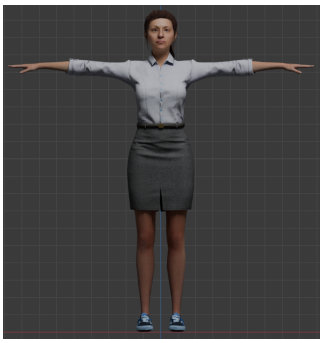
Figura 4.1: Set virtuale in Blender: le sagome quadrate rappresentano le luci, la parte rettangolare sotto la sedia l'unica area del pavimento di interesse per la proiezione delle ombre.

4.2.2 Realizzazione dell'avatar e animazione

Per mantenere uniformità con i test esistenti si è cercato di creare un avatar che riprendesse le sembianze dell'attrice. On-line sono disponibili diversi servizi per la sua realizzazione: MetaHuman [81] e Avaturn [82] sono quelli che offrono un migliore fotorealismo e inoltre sono in grado di realizzare avatar pronti per essere animati. Come anticipato, l'avatar è usato in sostituzione dell'attrice che nelle scene reali si trova a una distanza di poco più di 4 metri dal punto di presa, corrispondente al punto di vista del soggetto sotto test; questo rende trascurabili le differenze qualitative tra avatar realizzati con i due servizi. Anche se offre un'estetica finale meno ricca di dettagli, la scelta è ricaduta su Avaturn per via della sua maggiore praticità. Per la realizzazione dell'avatar l'attrice ha semplicemente scattato 3 foto del suo volto: una frontale, una laterale destra e una laterale sinistra e il software ha creato in pochi istanti l'avatar fotorealistico, con la possibilità di personalizzare abiti e capigliatura. Il modello è stato infine importato in Blender e leggermente personalizzato.

Articolazione labiale

Per creare l'animazione labiale, la mesh della testa, comprensiva di occhi, lingua e dentatura, è stata esportata da Blender in Audio2Face grazie a un plug-in realizzato



(a) *Avatar in T-pose.*

(b) *Avatar in posizione finale.*

Figura 4.2: Modello dell'avatar.

da NVIDIA. In Audio2Face, il volto dell'avatar è stato associato, attraverso una serie di punti caratteristici come gli angoli della bocca, i canti mediali e laterali, la punta del naso, ecc., al volto di una generica mesh usata dal software come riferimento.

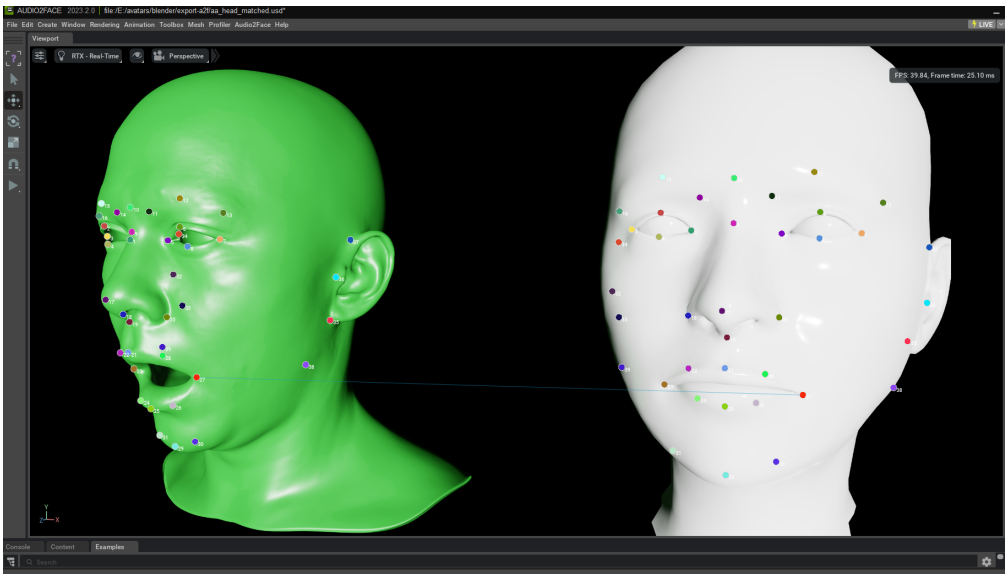


Figura 4.3: Associazione dei punti caratteristici tra mesh da animare (a destra) e mesh di riferimento (a sinistra).

Una volta definita l'associazione e fornito il segnale vocale desiderato, la mesh di riferimento viene animata e il software provvede a trasferire tali movimenti alla mesh dell'avatar. Una serie di parametri permettono di definire alcune caratteristiche

del movimento labiale e anche l'aspetto emotivo del viso, come ad esempio stupore o rabbia, la qual cosa però non riscuote interesse ai fini del presente lavoro. Per rendere più chiara e ampia l'articolazione labiale sono stati modificati alcuni parametri in modo da enfatizzare i movimenti.

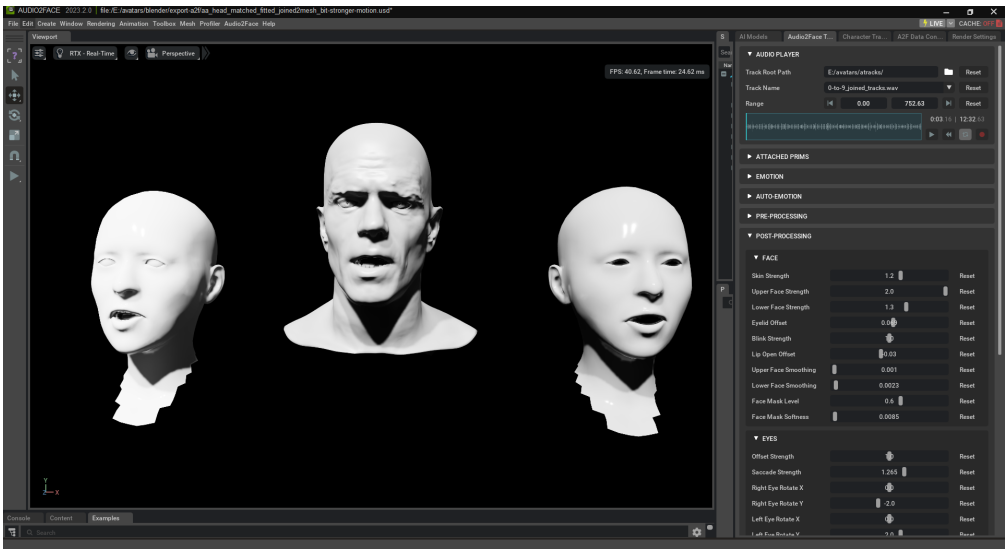


Figura 4.4: Fase di animazione, al centro la mesh di riferimento, a destra si possono vedere alcuni dei parametri configurabili per la modifica dell'articolazione labiale.

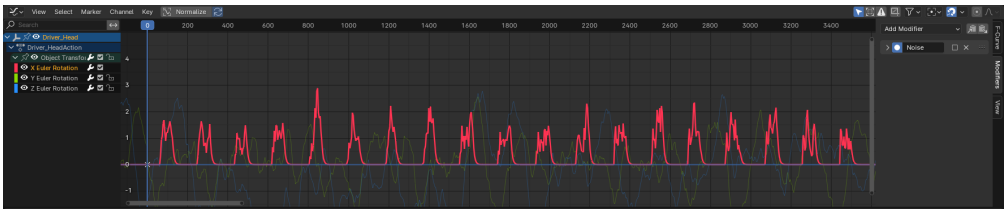
Ottenuto il risultato desiderato, l'animazione è stata esportata tramite un file di cache, reimportato in Blender e applicato alle mesh dell'avatar.

Movimenti secondari

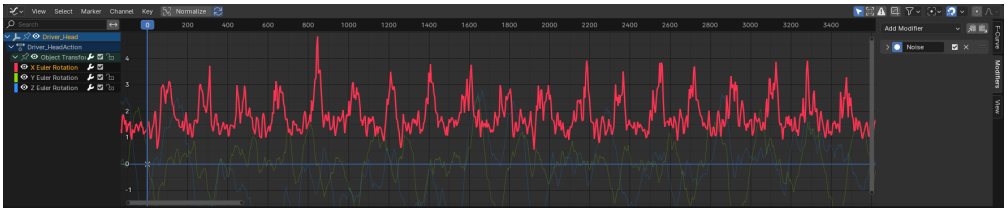
La presenza dei movimenti secondari è di fondamentale importanza per dare quel tocco di realismo in più, necessario per un'animazione verosimile. Sono stati individuati 2 movimenti secondari: (i) quello della testa, (ii) quello del busto. Anche in questo frangente, la distanza dal punto di presa limita per l'osservatore la capacità di riconoscere questi tipi di movimento, che, anche se variabili da persona a persona, sono solitamente piuttosto limitati se il parlatore non è condizionato da particolari stati emotivi. In questo caso l'oratore ha un tono neutro e si può ragionevolmente ritenere che i movimenti siano circoscritti e di piccola entità. Una conferma a quanto ipotizzato è fornita dalle clip registrate con l'attrice: è possibile osservare che i movimenti secondari risultano contenuti. Inoltre è possibile notare che per una persona seduta, sia (i) che (ii) possono essere approssimati con piccoli movimenti di rotazione.
I due tipi di movimento sono realizzati con piccole rotazioni casuali attorno a un

osso del collo e a uno del busto. Al movimento della testa inoltre, è stato aggiunto un moto di rotazione in avanti legato alla traccia vocale: i punti del segnale audio in cui l'ampiezza oltrepassava una certa soglia sono stati usati come valori da sommare ai valori casuali, in modo da dare una sorta di accento e di ritmo al movimento nel suo complesso.

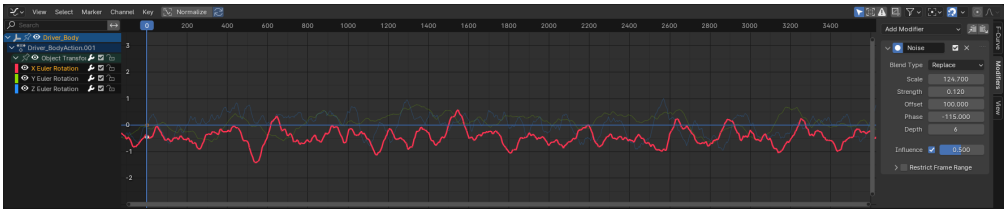
Per ottenere il risultato desiderato sono state utilizzate le *F-curve*, ossia delle curve, che possono essere generiche funzioni, con cui poter far variare un determinato attributo, in questo caso la rotazione attorno agli assi X,Y,Z delle due ossa del'armatura.



(a) *F-curve con i picchi della traccia audio per il movimento della testa.*



(b) *F-curve con rumore random sovrapposto ai picchi della traccia audio per il movimento della testa.*



(c) *F-curve per il movimento del busto, si nota la minore frequenza e ampiezza rispetto alle F-curve della testa.*

Figura 4.5: F-curve per le animazioni secondarie. In evidenza l'andamento per l'asse X, le curve più chiare rappresentano gli altri due assi.

4.2.3 Traccia vocale

La traccia vocale è semplicemente costituita dalla sequenza di tutti i singoli file audio anecoici utilizzati anche nella prima parte del lavoro. Ogni segnale audio è preceduto e seguito da 2 secondi di silenzio. Questa traccia complessiva è stata utilizzata sia per l'animazione del labiale che per l'animazione secondaria della testa. Con Python sono stati generati:

- uno script Powershell che usando FFmpeg concatenava i singoli file audio aggiungendo i 2 secondi di silenzio prima e dopo ogni traccia;
- un file di testo con i timestamp di inizio e fine delle singole tracce (comprensivi dei secondi di silenzio), utile nella successiva fase di taglio.

La durata totale della traccia è di 12'32".
Il listato *joined_tracks_maker.py* è disponibile in appendice A.4.

4.2.4 Rendering

Il rendering finale è stato realizzato suddividendo i fotogrammi dell'animazione, di durata complessiva pari a 12'32", tra diversi calcolatori, in modo da procedere in parallelo e ridurre le tempistiche. Allo scopo di illustrare la complessità computazionale dell'operazione, si elencano i sistemi utilizzati:

- n.1 PC i7 9500 48GB RAM con GPU RTX3060 12GB RAM;
- n.1 VM 48 vCPU, 56GB RAM con n. 2 GPU Tesla T4 16GB RAM;
- n.1 VM 48 vCPU, 56GB RAM;
- n.90 PC i5 9500 16GB RAM.

A eccezione del primo, gli altri sistemi sono stati messi a disposizione dei laboratori LABINF e ACSLAB afferenti al Dipartimento di Automatica e Informatica dell'Ateneo. La distribuzione del carico di lavoro tra i PC del laboratorioe e la successiva raccolta dei risultati è stata eseguita attraverso il sistema di gestione Ansible [83] di cui si omettono gli script.

Indicativamente sono stati necessari 4 giorni di calcolo.

4.3 Post-produzione

La post-produzione è costituita dalle seguenti fasi: VFX e compositing, taglio delle clip, realizzazione del video finale.

4.3.1 VFX e compositing

Le operazioni seguono indicativamente quanto fatto per le riprese reali, con la differenza che il chroma-key non è stato necessario.

Il video è stato ritagliato intorno alla figura dell'avatar e renderizzato con il codec intermedio Voukoder ProRes 4444 XQ; successivamente è stato integrato nelle tre scene del Museo Egizio come descritto nel cap. 3.

4.3.2 Taglio delle clip

Anche in questo caso, il taglio delle singole clip a partire dal filmato complessivo è stato eseguito attraverso una serie di script. Con Python, a partire dal file contenente le marche temporali delle singole tracce, è stato generato uno script PowerShell che tramite FFmpeg provvedeva a estrarre le corrispondenti porzioni di video.
Per ognuna delle scene SC1M, SC2M, SC3M, si sono ottenute 120 clip, una per frase, per un totale complessivo di 360 spezzoni video.
Il listato del programma *seek_lips_sync_by_timestamp.py* è disponibile in appendice A.5.

4.3.3 Video finale

Per questa parte si rimanda a quanto scritto in precedenza nel capitolo 3 alla sezione 3.3.4. Al fine di poter effettuare un confronto dei risultati, i video deI test sono stati realizzati utilizzando le clip con la medesima traccia dei test precedenti. Le figure 4.6 e 4.7 mostrano rispettivamente il risultato finale in formato monoscopico e un suo dettaglio.

Figura 4.6: Risultato finale in formato monoscopico dello scenario SC2M con avatar e rumore mascherante a 120° impersonato dal manichino *Dummy-Head*.

Figura 4.7: Dettaglio del risultato finale con avatar.

Capitolo 5

Test di intelligibilità

Il capitolo illustra gli scenari usati per i test di ascolto e la metodologia di sperimentazione.

5.1 Partecipanti

I partecipanti ai test, tutti di madrelingua italiana, di età compresa tra 22 e 46 anni (media 26,2 anni, deviazione standard 5,9), sono stati selezionati su base volontaria; non era prevista retribuzione e i soggetti sono stati ricompensati con semplici gadget quali penne, block-notes e borracce. Tutti i partecipanti hanno confermato di non avere evidenti problemi di udito; l'uso degli occhiali non pregiudicava lo svolgimento del test.

5.2 Set-up sperimentale

La sperimentazione è stata condotta presso Audio Space Lab del Politecnico di Torino. La sala è trattata acusticamente e insonorizzata per ottenere un livello di rumore di fondo inferiore a 38 dB nel range da 100 Hz a 10 kHz. Al suo interno è presente un sistema di riproduzione audio/video immersivo a 360° con audio 3OA e video sincronizzato con un HMD Meta Quest 2. La parte audio è costituita da un array sferico a 16 canali avente un raggio attorno allo sweet-spot di 1,2 m dove è presente una sedia girevole per i partecipanti. I 16 altoparlanti a due vie Genelec 8030B che compongono l'array sono equalizzati in modo da avere una risposta in frequenza piatta tra 40 Hz e 20 kHz nel punto di ascolto; due ulteriori diffusori Genelec 8351A sono dedicati alle basse frequenze. Una workstation di alto profilo si occupa della gestione audio e video tramite la DAW Bidule, il motore grafico Unreal Engine e MATLAB. Tutti gli altoparlanti sono pilotati dalla scheda audio Antelope Orion32 a 32 canali attraverso Bidule che gestisce la riproduzione 3OA;

il visore viene controllato con il motore grafico Unreal Engine; uno script Matlab coordina la riproduzione dei test tramite comunicazione OSC con Bidule e Unreal. La figura 5.1 mostra l'ASL durante una sessione di test, sono chiaramente visibili l'array di altoparlanti, i due subwoofer vicino alla parete di fondo e i trattamenti acustici, in particolare i bass-trap agli angoli della sala.

Figura 5.1: ASL durante una sessione di test [84].

5.3 Scenari di test

Tutti gli scenari sono ambientati nella sala conferenze del Museo Egizio di Torino. La sala ha un volume di circa 1500 m^3 e non ha trattamenti acustici; a causa di questa carenza il locale presenta un elevato riverbero con un T_{20} di 3,2 s [57], circa 2 secondi in più rispetto al valore ottimale stabilito dalle norme UNI [85].
La sala ha una platea composta da 100 sedie leggere; di fronte è presente un piccolo palco in legno alto 30 cm che ospita un tavolo, anch'esso in legno, dietro cui solitamente si siedono gli oratori; per amplificarne la voce è presente un sistema di diffusione audio costituito da due array di altoparlanti verticali. Tali diffusori sono posizionati a ridosso delle pareti laterali, tra palco e platea, a un'altezza dal pavimento di 1,7 m rispetto al loro centro.
La sperimentazione prevede 3 scenari di test rappresentanti una tipica situazione di utilizzo della sala, ognuno con una diversa configurazione spaziale delle sorgenti sonore disturbanti. In tutti gli scenari il parlante target, rappresentato in un caso dalla persona reale e nell'altro dall'avatar, è frontale al soggetto a una distanza

di 4.1 m; inoltre, per simulare fedelmente l'uso realistico della sala il discorso target è stato sempre riprodotto come trasmesso dagli altoparlanti del sistema di amplificazione.
Nel punto di sweet-spot, i segnali auralizzati, ossia resi spazializzati in base alle caratteristiche dello scenario, sono stati equalizzati in modo da ottenere un livello di pressione sonora pari a 73 $dB(A)$, valore tipicamente raggiunto all'interno della sala conferenze nella posizione di ascolto L. Per le condizioni con rumore interferente è stato impostato un SNR di -5 dB, corrispondente a una condizione acustica moderatamente impegnativa, simile a SRT80 in condizioni anecoiche [15].
Nel caso di condizione in quiete le tracce audio del segnale target erano precedute e seguite da 2 secondi di silenzio, mentre nelle condizioni con parlante interferente da 2 secondi di rumore, rappresentato da spezzoni di un brano foneticamente bilanciato, per una durata complessiva di 6-7 secondi. Questo permetteva al soggetto di preparasi all'ascolto.
Le tre scene hanno la seguente configurazione:

- **SC1M**: non sono presenti sorgenti di disturbo, il segnale audio è dovuto al solo parlante target;
- **SC2M**: il segnale di disturbo si trova a 120° rispetto all'azimut, a una distanza di 1,8 m;
- **SC3M**: il segnale di disturbo si trova a 180° rispetto all'azimut, quindi alle spalle del soggetto a una distanza di 1,8 m.

La figura 5.2 illustra la pianta della sala conferenze con le posizioni del punto di ascolto (L), del segnale target (T0°), dei diffusori (LS1, LS2) e dei segnali mascheranti (N120°, N180°). Un'immagine della sala conferenze si può vedere in figura 5.3.
Le figure 5.4a, 5.4b, 5.4c mostrano rispettivamente le scene SC1M, SC2M e SC3M. Il manichino, indicato dalle frecce rosse, che si può osservare nelle ultime due rappresenta la posizione del segnale mascherante. In 5.4c, essendo a 180° di azimut e trovandosi alle spalle del soggetto, il manichino risulta diviso nelle estremità destra e sinistra del fotogramma equirettangolare.
Nelle figure 5.5a, 5.5b, 5.5c il dettaglio rispettivamente delle scene senza parlatore target, con parlatore reale e infine con avatar.

5.3.1 Acquisizione degli scenari di contesto

Le scene audiovisive di partenza, cioè le ambientazioni nella sala conferenze, sono state acquisite in modo da poter auralizzare discorsi diversi in differenti posizioni della coppia parlante target e ascoltatore. Per questo motivo i movimenti labiali sono stati esclusi, la loro registrazione in simultanea non avrebbe offerto questa

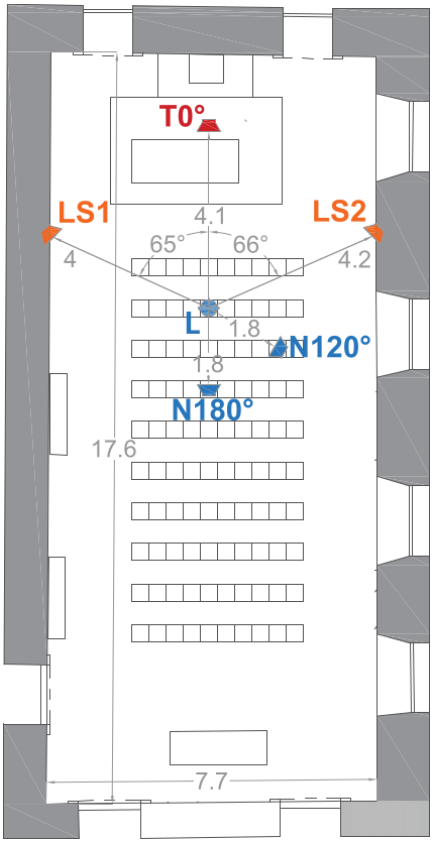


Figura 5.2: Pianta della sala conferenze con la posizione dei diffusori (LS1/LS2), della sorgente target (T0°), del rumore (N120°, N180°), del punto di ascolto (L) [84].

flessibilità. Di fatto la parte visiva di base apporta solo informazioni inerenti al contesto e alla posizione dei suoni target, attraverso l'altoparlante situato frontalmente all'ascoltatore sopra al tavolo in legno, e attraverso il manichino Dummy-Head in caso di presenza di suoni interferenti.
Le riprese video sono state effettuate utilizzando la telecamera Insta360 Pro a 360°; le registrazioni audio usando l'array di microfoni sferico Zylia ZM-1 a 19 capsule. Per maggiori dettagli sulle modalità si rimanda a [70].

5.4 Protocollo di esecuzione

I 20 soggetti sono stati suddivisi in due gruppi di egual numero, la prima metà è stata sottoposta ai test con parlatore reale, l'altra ai test con avatar. Ai singoli soggetti è stato descritto il compito da eseguire, cioè ascoltare l'oratore di fronte

Figura 5.3: La sala conferenze vista dal palco [84].

a loro e ripetere le parole che erano riusciti a comprendere; inoltre sono stati informati di restare seduti sulla sedia girevole e tenere la testa ferma, senza girarsi, per mantenere costante la configurazione spaziale del discorso target e del rumore di mascheramento rispetto all'ascoltatore. Prima del test vero e proprio, per familiarizzare con il compito da svolgere, i singoli partecipanti sono stati sottoposti a una prova generale per ogni scenario, composta al più da sei enunciati del test, differenti da quelli della sperimentazione reale; quando il compito era chiaro il training veniva interrotto e si passava alla condizione di test successiva. Entrambi i gruppi sono stati addestrati usando gli scenari con la persona reale.
Il test effettivo era costituito dalle tre ambientazioni descritte in 5.3, ognuna delle quali formata da 20 frasi tratte da un elenco distinto del test di intelligibilità ITAMatrix. La sperimentazione iniziava sempre con lo scenario in cui il parlatore interferente era assente; i successivi due venivano proposti in ordine alterno per bilanciare l'effetto di eventuali bias derivanti dal mantenimento dello stesso ordine delle scene. I test SI sono stati condotti in un formato aperto, in cui gli ascoltatori ripetevano verbalmente le parole comprese e lo sperimentatore registrava le risposte corrette. Il test è durato circa 20 minuti per ogni partecipante. La procedura sperimentale ha ricevuto l'approvazione etica (riferimento 100993/2023).
La stessa modalità è stata seguita per i test di intelligibilità pregressi, condotti su un gruppo distinto di 10 individui, ambientati nei medesimi scenari ma sprovvisti dei movimenti labiali; qui il parlante target era sostituito da un altoparlante, il cui scopo era di dare un riferimento visivo relativo alla provenienza del suono.

(a) *SC1M.*

(b) *SC2M.*

(c) *SC3M.*

Figura 5.4: Tre fotogrammi estratti delle scene con parlate target reale. Le frecce rosse indicano il *Dummy-Head* per il suono mascherante.

(a) *Scenario SC1M originale senza integrazione del parlante target.*

(b) *Scenario SC1M con integrazione del parlante target reale.*

(c) *Scenario SC1M con integrazione del parlante target avatar*

Figura 5.5: Dettagli del palco per le tre condizioni di test. Dall'alto in basso: versione originale senza parlante target, versione con l'integrazione del parlante reale, versione con l'integrazione dell'avatar.

Capitolo 6

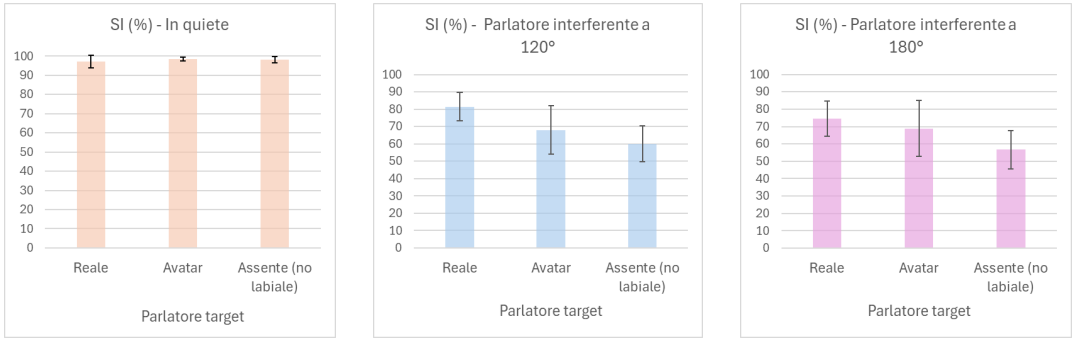
Risultati

Il capitolo illustra e discute i risultati ottenuti facendo un confronto con gli esiti del pregresso test di intelligibilità del parlato che fornisce solo il contesto visivo, inteso come presentazione visiva del luogo in cui avviene l'ascolto e della posizione delle sorgenti sonore.

6.1 Risultati

I grafici nella figura 6.1 illustrano la media e la deviazione standard dei punteggi di intelligibilità del parlato per ciascuna configurazione di ascolto, mentre nella tabella 6.1 sono presentati i corrispondenti valori. Per tutte le condizioni di test (i.e., labiale: reale, sintetico, assente) nello scenario di quiete la percentuale si accosta al 100%, mentre cala drasticamente nelle altre configurazioni. In tutte le condizioni di rumore il labiale naturale fornisce un apporto informativo maggiore: per ovvie ragioni rispetto ai test in assenza di labiale, mentre per quanto riguarda il labiale sintetico a causa della qualità dell'animazione non ancora del tutto in grado di rappresentare correttamente i differenti visemi; sebbene la rete neurale possa funzionare adeguatamente in diversi linguaggi, l'addestramento è stato svolto utilizzando la lingua inglese, fattore che potrebbe pregiudicare la corretta rappresentazione di visemi in italiano.
Dai grafici inoltre si può notare come le percentuali di SI siano pressoché invariate, a parità di condizioni di test, nelle due configurazioni di rumore a 120° e 180°, suggerendo che il labiale annulli l'effetto della SRM. Tuttavia, non sono presenti differenze statisticamente significative tra parlante mascherante a 120° e 180° per tutte le condizioni di test. Per questa ragione, per ogni condizione di test i risultati ottenuti dalle scene con rumore sono stati accorpati, considerando come unica configurazione la presenza di rumore, indipendentemente dalla sua posizione. I confronti tra le medie così ricavate hanno mostrato differenze statisticamente

significative tra le diverse condizioni di test. La migliore intelligibilità si ottiene con il labiale reale, che raggiunge una media pari a 78%, seguita dal labiale sintetico con 68,5%. Infine, il risultato peggiore si ha in assenza di labiale con una intelligibilità media di 58,5%. Il grafico 6.2 illustra i risultati ottenuti e la tabella 6.2 i corrispondenti valori.



(a) *SI in condizione di quiete.* **(b)** *SI con rumore a 120°.* **(c)** *SI con rumore a 180°.*

Figura 6.1: Confronti SI tra condizioni di labiale reale, sintetico e in sua assenza.

Tabella 6.1: Esiti test SI: media e deviazione standard.

		In quiete	**Rumore 120°**	**Rumore 180°**
Labiale reale	Media	97,3%	81,4%	74,6%
	DS	3,30%	8,14%	10,25%
Labiale sintetico	Media	98,6%	68%	68,9%
	DS	0,97%	13,96%	16,03%
No Labiale	Media	98,2%	60%	56,8%
	DS	1,55%	10,33%	11,04%

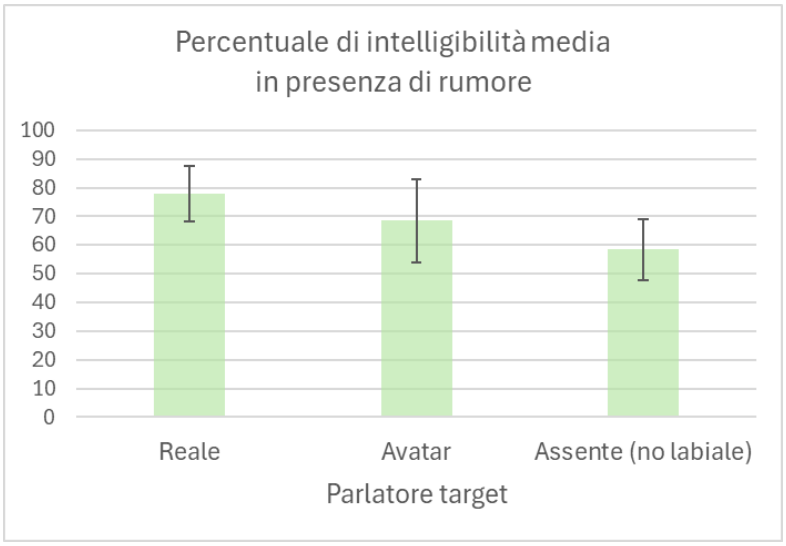


Figura 6.2: Intelligibilità media in condizione di rumore per le condizioni di labiale reale, sintetico e in sua assenza.

Tabella 6.2: Percentuale di intelligibilità media e deviazione standard in presenza di rumore per le varie condizioni di test.

	Labiale reale	**Labiale sintetico**	**No labiale**
Media	78,0%	68,5%	58,5%
DS	9,6%	14,6%	10,6%

Sui risultati ottenuti sono state fatte analisi statistiche, il test U di Mann-Whitney è stato effettuato per ogni condizione di test tra rumore a 120° e 180°. Valori di p-value ≤ 0.05 rigettano l'ipotesi nulla e indicano, con una probabilità del 95%, che le differenze rilevate nei valori medi siano dovute a variazioni tra le due condizioni e non a fattori casuali. Con i valori ottenuti, indicati in tabella 6.3 non si può di rifiutare l'ipotesi nulla. Poiché le medie in posizione 180° sono inferiori a quelle in 120°, non potendo rifiutare l'ipotesi H_0, è da escludere che il contributo labiale con rumore a 180° sia maggiore del contributo a 120°. Pertanto, si può affermare che i due valori sono uguali.

Tabella 6.3: p-value per il test U di Mann-Whitney per condizioni di test e posizioni di rumore. Valori di p-value ≤ 0.05 rifiutano l'ipotesi nulla.

$\mathbf{H_0}$	**Labiale Reale**	**Labiale Sintetico**	**No Labiale**
$180° \geq 120°$	0,913	0,842	0,231

Inoltre, per verificare il contributo all'intelligibilità del labiale sintetico rispetto a quello reale, è stato applicato il test U di Mann-Whitney a tutte le condizioni di test tra i due tipi di labiale, sia per la configurazione in quiete che per le due posizioni del parlante interferente. In aggiunta, si è voluto verificare il contributo delle sole informazioni contestuali rispetto al labiale sintetico. Nella tabella 6.4 sono indicate le ipotesi nulle e i relativi p-value risultanti dalle analisi statistiche. Valori di p-value ≤ 0.05 rigettano l'ipotesi nulla, indicando una probabilità del 95% che le differenze riscontrate nei valori medi siano effettivamente dovute a variazioni tra le due condizioni e non a fattori casuali. In entrambe le condizioni con rumore interferente, l'intellegibilità con il labiale sintetico risulta, dai valori medi, inferiore rispetto a quello reale, infatti in base ai p-value, che in tabella sono sottolineati, si rifiuta l'ipotesi nulla. Inoltre, anche l'ulteriore ipotesi nulla, specificata nell'ultima riga della tabella, si rifiuta, indicando che il labiale sintetico apporta un contributo, rispetto alla sua assenza, come ci si aspetterebbe.

Tabella 6.4: p-value per il test U di Mann-Whitney. Valori di p-value $\leq 0{,}05$ sono sottolineati e indicano il rifiuto dell'ipotesi nulla.

$\mathbf{H_0}$	**In quiete**	**N120°**	**N180°**
Sintetico $\geq$ Reale	0,923	<u>0,000</u>	<u>0,009</u>
No labiale $\geq$ Sintetico	0,296	<u>0,001</u>	<u>0,000</u>

In conclusione, dai risultati ottenuti, si può affermare con certezza che il labiale reale fornisce il maggiore apporto informativo all'intelligibilità del parlato: i risultati sono infatti migliori rispetto alle altre due condizioni esaminate; l'intelligibilità è superiore di circa il 10% rispetto al labiale sintetico e di circa il 20% rispetto all'assenza di labiale. Anche il labiale sintetico, nonostante le sue imperfezioni, produce risultati migliori rispetto all'assenza di labiale, con un miglioramento del 10%.

Capitolo 7

Conclusioni e prospettive future

Nonostante il giovamento che un ipoudente può trarre dall'indossare un apparecchio acustico, spesso il suo utilizzo viene evitato per via delle scarse prestazioni che non apportano i benefici auspicati dal paziente. La messa a punto degli apparecchi è limitata dai vigenti test audiometrici che non tengono in considerazione né i complessi aspetti del campo acustico, come un eccessivo riverbero, né tutti quegli elementi che concorrono alla riuscita di una conversazione, come il labiale dell'interlocutore.

Per far fronte a queste criticità, si stanno sviluppando test audiometrici in grado di coinvolgere e il senso dell'udito e quello della vista: il fine è creare quello che nell'ambito della ricerca si definisce test ecologico, cioè creare dei test in cui le condizioni di sperimentazioni siano il più realistiche possibili per il soggetto sottoposto al test, in modo da stimolare le sue reazioni più naturali e franche. Questo è quanto si sta cercando di portare avanti presso l'ASL del Politecnico di Torino: sviluppare dei test di intelligibilità del parlato in ambiente altamente immersivo dotato di audio ambisonico e video 360° 3D.

Le ricerche fatte in letteratura hanno mostrato una carenza negli studi sull'intelligibilità del parlato in ambienti di test altamente ecologici. Questa tesi vuole colmare tali lacune analizzando il ruolo del labiale e il suo apporto informativo alla comprensione del parlato.

Per attualizzare maggiormente lo studio sono state previste due tipologie di movimenti labiali: un labiale naturale, fornito da una persona reale; un labiale sintetico, fornito da un avatar realistico animato tramite il software Audio2Face basato su IA. Tre scenari video 360° 3D, già utilizzati in precedenti test di ascolto presso l'ASL, sono stati integrati con le due tipologie di stimoli labiali. La prima ambientazione è in quiete, ossia è presente solo il parlante target, rappresentato

dalla persona reale o dall'avatar, che enunciano le frasi del test di intelligibilità del parlato femminile standard per la lingua italiana "Italian Matrix Sentence Test"; gli ulteriori due scenari prevedono, in aggiunta, un parlatore interferente di disturbo posizionato rispettivamente a 120° e 180°, rappresentato da un brano foneticamente bilanciato; in questo caso il valore SNR tra segnale target e rumore era pari a $-5\ dB$.

La sperimentazione ha coinvolto due gruppi di soggetti normoudenti, il primo sottoposto ai test con labiale reale, l'altro ai test con labiale sintetico. I risultati ottenuti sono stati confrontati con gli esiti dei test precedenti sprovvisti di stimoli labiale.

I risultati rivelano che il labiale è di fondamentale importanza per l'intelligibilità del parlato: includere il labiale reale nei test è indispensabile per riprodurre scenari con alta validità ecologica. Sebbene il labiale sintetico migliori la comprensione, attualmente non raggiunge il livello di quello naturale, il cui apporto informativo è superiore. Tuttavia, la continua evoluzione dell'IA, potrebbe far raggiungere al labiale sintetico la stessa espressività di quello reale allargando gli scenari di sperimentazione e di ricerca. Per questo motivo, sarebbe opportuno ripetere un'indagine analoga quando l'IA avrà compiuto ulteriori progressi rispetto alla situazione attuale.

Inoltre, sarebbe interessante proporre lo stesso confronto utilizzando strumenti di animazione labiale commerciali, perché tipicamente più prestanti, ma esclusi da questo studio; ciò potrebbe fornire un'ulteriore conferma delle conclusioni ricavate o, al contrario, una smentita. Capire se il labiale sintetico possa sostituire quello reale potrebbe risultare estremamente utile nel caso in cui si vogliano utilizzare altri parlatori target o inserire il labiale anche per i parlatori mascheranti negli stessi test. L'uso del software per la generazione del labiale sintetico eviterebbe la necessità di preparare set di ripresa, registrare video e affrontare la relativa post-produzione, operazioni di certo più onerose rispetto a quelle richieste in ambienti virtuali. In ogni caso, per una parlante donna, il discorso target del test ITAMatrix resta invariato; questo lavoro ha prodotto e reso disponibile un vasto set sia di clip già correttamente ambientate nello scenario di ascolto, sia di clip contenenti solo il parlatore target privo di sfondo, pronte per essere integrate in scenari al momento sprovvisti di stimolo labiale. In aggiunta, quanto realizzato può essere utilizzato per automatizzare lo svolgimento dei test. Essendo ora disponibile un archivio, oltre che di tracce audio, anche delle corrispondenti clip video pronte all'uso, sarebbe possibile sviluppare un sistema in grado di selezionare il materiale da riprodurre direttamente dall'archivio, senza la necessità di creare a priori un filmato composto da una sequenza di clip predefinite, svincolando e snellendo notevolmente la procedura di preparazione dei test. Questo approccio renderebbe l'intero processo più efficiente e meno dispendioso in termini di tempo e risorse.

Premi e riconoscimenti

Questo lavoro ha consentito all'autore di contribuire in qualità di coautore alla stesura dell'articolo *"Impact of contextual and lip-sync-related visual cues on speech intelligibility through immersive audio-visual scene recordingsin a reverberant conference room"*, accettato e in fase di pubblicazione per la conferenza internazionale "Internoise 2024" [84].

Inoltre, la sperimentazione e i relativi risultati sono stati fondamentali per la redazione dell'articolo dal titolo *"Impatto dell'immersione audiovisiva sull'intellegibilità del parlato"* [86], presentato al 50° Convegno Nazionale dell'Associazione Italiana di Acustica, tenutosi a Taormina dal 29 al 31 maggio 2024. L'articolo è risultato vincitore del premio "Italo Barducci[1]", assegnato a Angela Guastamacchia, dottoranda presso il Dipartimento Energia del Politecnico di Torino.

[1]Istituito nel 2004 dall'AIA con lo scopo di incoraggiare e valorizzare l'attività e la produzione scientifica di giovani studiosi e ricercatori nei vari campi dell'acustica

Appendice A

Listati

A.1 speech2text.py

content/codes/speech2transcript.py

```
'''
Author: Andrea Galletto
Date: October 2023

Spech to Transcript
This script loops on .wav files in the 'searching_path' directory,
it tries to understand the speech of file and to transcript the sentences
in a .txt file with the same name of corresponding audio file
and in the same directory.
NOTE: it overwrites the .txt files without any warning.

Whisper is a general-purpose speech recognition model.
https://github.com/openai/whisper
'''

import os
import whisper as wh

## NOTE !! specify searching path with  "/"
searching_path = "C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic"
extension_to_search_for = ".wav"
file_with_list_of_audiofile_processed = "list_of_audio_files.txt"

def get_list_of_files_by_extension(searching_path, f_extension):
    num_of_files = 0
    list_of_files = []
    for f_name in os.listdir(searching_path):
```

```
28          f_path = os.path.join(searching_path,f_name)
29
30          ## verifico se è un file con estensione voluta e carico la
    lista
31          if os.path.isfile(f_path) and f_name.endswith(f_extension):
32              num_of_files+=1
33              #print(f"Nome file={f_name}; Path={f_path}")
34              list_of_files.append(searching_path + '/' + f_name)
35
36      print(f"Total files found={num_of_files}")
37      return list_of_files
38
39  def speech2text(list_of_files):
40      print("Start OpenAI WHISPER")
41      ## Model could be [tiny|base|small|medium|large]
42      model = wh.load_model("medium")
43      list_of_sentences = []
44      for f in list_of_files:
45          print(f"processing {f}")
46          result = model.transcribe(f, fp16=False)
47          print(result["text"])
48          list_of_sentences.append(result["text"])
49      return list_of_sentences
50
51  def save_sentence_files(list_of_files, file_extension,
    list_of_sentences, dest_path):
52      print("start fun save_sentence_files")
53      ## save the list of processed files
54      f_list_txt = open(dest_path + "/" +
    file_with_list_of_audiofile_processed, "w")
55
56      for i, f_path in enumerate(list_of_files):
57          ## get the file name by splitting the full path and keep last
     field
58          f_audio_name = f_path.split("/")
59          f_audio_name = f_audio_name[-1]
60          print(f"file audio name = {f_audio_name}")
61          ## keep just the filename without the extension
62          prefix,delim,suffix = f_audio_name.rpartition(file_extension)
63          f_sentence_name = prefix if (prefix) else suffix
64          f_sentence_name = f_sentence_name + ".txt"
65          print(f"file sentence name={f_sentence_name} : {
    list_of_sentences[i].strip()}")
66          ## save the sentence to file (without leading and trailing
    spaces '.strip()')
67          f_txt = open(dest_path + "/" + f_sentence_name, "w")
68          f_txt.write(list_of_sentences[i].strip())
69          f_txt.close()
70          ## save the url of every files
```

```
71          f_list_txt.write(f_path + "\n")
72
73      f_list_txt.close()
74      print(f"saved {len(list_of_files)} text files, list available in
        {file_with_list_of_audiofile_processed}")
75      return
76
77  def main():
78      list_of_files = get_list_of_files_by_extension(searching_path,
        extension_to_search_for)
79      print(f"founded: {list_of_files}")
80      list_of_sentences = speech2text(list_of_files)
81      save_sentence_files(list_of_files, extension_to_search_for,
        list_of_sentences, searching_path)
82
83
84  if __name__ == "__main__":
85      main()
```

A.2 seek_lips_sync.py

content/codes/seek_lips_sync.py

```
1  '''
2  Author: Andrea Galletto
3  Date: December 4th 2023
4
5  This script creates the command line to execute to cut with ffmpeg
       the
6  full scene footage into a single footage for each sentence, saving
       the
7  results in a dedicated directory.
8
9  Moreover the script:
10 - substitute the original recorded with the anechoic track, trying
       to sync it with the clip.
11 - can select the amount of leading and trailing silence respectively
        at the beginning and at the end of the sentences
12
13 To run the script:
14 - update the variables:
15   - {i_footage_to_process} with the number of footage to process
16   - {footages_best_recs} with the correct matrics of the best
       records
17   - {scene_to_process} with the name of the scene, ex. "SC1M"
18   - {footage_names} # with the desired footages list to process
19 - run the script
```

```
'''

import numpy as np
import math
import scipy
import matplotlib.pyplot as plt
from scipy.signal import find_peaks
import ffmpeg
import json
import soundfile as sf
import librosa

## ==== AUDIO parameters ==== ##
file_audio_extension = ".wav"
# Path realitve to the reference audiotrack, i.e the audio track of the official test
searching_path_af = "C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic"
#file_with_list_of_audiofile_to_process = "list_of_audio_files_0.txt"
file_basename_with_list_of_audio_files = ("list_of_audio_files_", ".txt")

## ==== VIDEO parameters ==== ##
# each row represents a footage [0..9] of the scene 0 to 9 (i.e. set of senteces with the same person name);
# each element is the best sync performance of the 3 recording for each audio-test sentence
# 410cm : NOTE at idx 6 the last item is '0' cause memory full during recording ! Check the Una-Tantum session
footages_best_recs_410 = [
                [2,3,3,2,2,1,1,2,2,2,1,3],
                [2,1,1,1,3,1,3,2,2,3,3,3],
                [3,2,3,3,3,2,3,2,1,3,3,2],
                [1,1,3,1,3,2,1,1,3,3,3,2],
                [2,1,2,3,1,3,3,3,1,3,3,3],
                [2,2,3,2,2,3,1,2,3,1,2,2],
                [3,3,2,2,3,1,3,1,1,3,3,0],
                [2,3,3,1,1,1,2,2,2,2,3,2],
                [2,3,2,2,2,2,2,1,1,3,2,2],
                [3,3,1,3,3,2,2,1,2,1,3,3]]
# 180cm ** TODO **
footages_best_recs_180 = [
                [2,2,2,2,2,2,3,3,2,2,3,2],
                [3,1,2,3,3,3,3,2,2,3,2,3],
                [],
                []]

# Select the correct matrix of beste recordings
footages_best_recs = footages_best_recs_410
```

```
63 #footages_best_recs = footages_best_recs_180
64
65 # ** NOTE ** All path MUST be exist!
66 scene_to_process = "SC1M"  # SC1M, SC2M, SC3M
67 footages_path_src = f'E:/compositing/410_{scene_to_process}_comps'
68 footages_path_dst = f'E:/compositing/410_{scene_to_process}_comps'
69
70 # subdirectory in the {footages_path_dst} to keep the cutted clips
71 splitted_sentences_dst_dir = f'{scene_to_process}'  # footage number
       [0-9] will be appended (ex: SC2M_s0)
72
73 # Footages with full composite scene for the 'manual version' test (
       the real tests will have to be done with UE and RealTime
       compositing)
74 footage_names_410_SCxM = [f'{scene_to_process}_s0.mp4',
75                           f'{scene_to_process}_s1.mp4',
76                           f'{scene_to_process}_s2.mp4',
77                           f'{scene_to_process}_s3.mp4',
78                           f'{scene_to_process}_s4.mp4',
79                           f'{scene_to_process}_s5.mp4',
80                           f'{scene_to_process}_s6.mp4',
81                           f'{scene_to_process}_s7.mp4',
82                           f'{scene_to_process}_s8.mp4',
83                           f'{scene_to_process}_s9.mp4']
84
85 # Select the correct footages list to process
86 footage_names = footage_names_410_SCxM
87
88 FPS = 29.97
89 ref_tone = 443  # Hz, reference tone frequency LA, as specified in
       the prompter script
90 ref_tone_len = 0.3  # sec, as done in the prompter
91 threshold_ampl = 0.5  # treshold for tone peak amplitude to be
       detected [0.0-1.0], valid for the recorded footage (0.5 is quite
       good)
92 leading_tones_for_recording_session = 48  # for each rec session we
       have this amount of leading tone: 12 sentence repeated 4 times
93 leading_tones_for_recording_una_tantum_session = 16  # leading tone
       for the una-tantum session
94 sentence_time_lenght = 3  # sec, the average length of the sentences,
        need for cut the very last performance
95
96 # For parameters See: https://librosa.org/doc/0.10.1/generated/
       librosa.stft.html
97 # STFT parameters to seek for the leading tone
98 # The 'max_stft_freq' depends by the Sample Rate SR of the audio file
       . In our case is sr=48000
99 n_fft=2048
```

```
100 bins = 1 + n_fft / 2  # number of frequency bins (https://librosa.org
    /doc/0.10.1/generated/librosa.fft_frequencies.html#librosa.
    fft_frequencies)
101 max_stft_freq = 24000  # Hz (https://librosa.org/doc/0.10.1/generated
    /librosa.fft_frequencies.html#librosa.fft_frequencies)
102 bin_freq_hop = max_stft_freq / (bins - 1)
103
104 # Amount of seconds to retain before start to trim the footage with
    respect to the beginning of the
105 # audio track of official audiometric test (for ex. the file 00252.
    wav).
106 # i.e. for example 2 seconds before the performer start to talk.
107 # (in the actual test there are 2 sec. of silence before the subject
    can earing the sentence).
108 # So, if we put here 2 sec. we have 2 sec. more in the video during
    we can see the performer waiting to talk,
109 # otherwise we will see the performer as freeze until the talk
110 leading_silence = 2  # sec.
111 # The same as leading silence, but at the end of the sentence, with
    respet to the end of the
112 # official track of audiometric test
113 trailing_silence = 2  # sec.
114
115
116 # For each recording we get the list of the audio file playedback
    from the text file {file_basename_with_list_of_audio_file}
117 def get_list_of_audio_files(searching_path,
    file_basename_with_list_of_audio_file, i_footage_to_process):
118     #print(f'i_footage_to_process={i_footage_to_process}')
119     basename = file_basename_with_list_of_audio_file[0]
120     ext = file_basename_with_list_of_audio_file[1]
121     file_name = searching_path + "/" + basename +
    i_footage_to_process + ext
122     #print(f'file_name={file_name}')
123     list_of_audio_files = []
124     try:
125         with open(file_name, "r") as f:
126             print(f'#try reading file={file_name}')
127             list_of_audio_files = f.readlines()
128         f.close()
129     except Exception as e:
130         print(f"#ERROR reading the file {file_name}")
131         quit()
132     # We have in list_of_audio_files the following:
133         #C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea
     Galletto/Target_anechoic/00152.wav
134         #C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea
     Galletto/Target_anechoic/00264.wav
135     return list_of_audio_files
```

```
136
137
138 def get_file_name(full_path):
139     ## keep just the filename without the extension
140     # we get substring following the last "/", i.e the full file name
         with extension
141     prefix_tmp,delim_tmp,full_fname = full_path.strip().rpartition("/
        ")
142     # rid of the extension
143     fname,delim_tmp,suffix_tmp = full_fname.strip().rpartition(
        file_audio_extension)
144     return fname
145
146
147 def load_audio_track(af_path):
148     audio_data, sample_rate = sf.read(af_path)
149     return audio_data, sample_rate
150
151
152 def trim_footage(scene_to_process, footage_file, path_dest,
        start_cuts_s, best_rec_for_sentence, list_of_audio_files,
        leading_silence, trailing_silence):
153     print(f'#** trim_footage() start the dirty job **')
154     # get fps
155     info_json = ffmpeg.probe(footage_file)
156     fps = info_json['streams'][0]['r_frame_rate']
157     # ** Before start we check for the correct FPS of the footage **
158     try:
159         assert fps == "30000/1001" or \
160                fps == "29.97" or \
161                fps == "2997/100"
162     except AssertionError:
163         print('#*****')
164         print('# fps not 29.97')
165         print('# Check video format')
166         print('#*****')
167         quit()
168     ## ** End of assert ** ##
169     print('
        #-------------------------------------------------------------')
170     print(f'# *** Cut & Paste in a powershell script, then run it
        ***\n')
171     print(f'# *** start/stop video cutting ***')
172     print(f'# Scene to process = {scene_to_process}')
173     print(f'# leading_silence = {leading_silence} -> i.e. we can see
        the performer waiting to talk instead of starts immediately')
174     print(f'# trailing_silence = {trailing_silence} -> i.e. we can
        see the performer waiting after talk instead of disappear
        immediately')
```

```
175     print(f'New-Item -Path "{path_dest}" -ItemType "directory" -Force
         # the destination directory')
176     print(f'$footage = "{footage_file}"  # the file to cut')
177 
178     for i, start_s in enumerate(start_cuts_s):
179         print(f'\n# -- sentence n. {i} -- #')
180         # compute some parameters
181         afnamefp = list_of_audio_files[i].strip()  # audio file name
        full path
182         vfname = get_file_name(list_of_audio_files[i].strip())  # the
         video file keeps same name of audio track
183         vfnamefptmp1 = f'{path_dest}/{vfname}_tmp.mp4'  # temporary
        video file name full path
184         vfnamefptmp2 = f'{path_dest}/{vfname}_tmp2.mp4'  # temporary
        video file name full path
185         vfnamefp = f'{path_dest}/{scene_to_process}_{vfname}.mp4' #
        final video file name full path
186         vfnamefp_mute = f'{path_dest}/{scene_to_process}_{vfname}
        _mute.mp4' # final video file name full path
187 
188         # get the duration of audio track to compute the right ending
         cut time
189         atrack, sr = librosa.load(path=afnamefp, sr=None)
190         atrack_len = librosa.get_duration(y=atrack, sr=sr)
191         print(f'# audio track {afnamefp}, time lenghts = {atrack_len}
         sec., sr = {sr}')
192         # stopping-time of cut
193         stop_s = start_s + atrack_len + trailing_silence
194         # starting-time of cut, we need 2 more seconds earlier than
        the actual start of speech
195         start_cut_s = start_s - leading_silence
196 
197         ## *** make the dirty job *** per opzioni https://trac.ffmpeg
        .org/wiki/Encode/H.264 ##
198         ## Constructs the ffmpeg command
199         video_opts = "-c:v libx264 -preset fast -qp 0 -b:v 20M"  #
        codec di ffmpeg
200         ffmpg_cut_cmd = f'ffmpeg -i "$footage" -ss {start_cut_s:.5f}
       -to {stop_s:.5f} {video_opts} -an "$vf_tmp1"'  ## h.264 encoder
       MUTE
201         ## SWAP and SYNC audio
202         ffmpg_swap_audio_cmd = f'ffmpeg -i "$vf_tmp1" -i "$af" -
        filter_complex "[1]adelay={leading_silence * 1000}[out];[out]amix=
        inputs=1" -c:v copy -b:a 192K -map 0:v:0 -map 1:a:0 -an "$vf_tmp2"
        '  # Add leading silence, no trailing silence-pad
203         ffmpeg_pad_audio_cmd = f'ffmpeg -i "$vf_tmp2" -filter_complex
         "[0:a]apad[a1];[a1]amix=inputs=1" -c:v copy -b:a 192k -shortest "
        $vf"'  # Add trailing silence
```

```
204         ffmpeg_rem_audio_cmd = f'ffmpeg -i "$vf" -c:v copy -an "$vf_mute"' # discard the audio
205         remove_vfnametmp_cmd = f'Remove-Item -Path "$vf_tmp1","$vf_tmp2" -Force'
206 
207         # Make the actual PowerShell script
208         print(f'Write-Output "# --- Sentence n.{i}, best recorded performance={best_rec_for_sentence[i]} --- #"')
209         print(f'$vf_tmp1 = "{vfnamefptmp1}"')
210         print(f'$vf_tmp2 = "{vfnamefptmp2}"')
211         print(f'$vf = "{vfnamefp}"')
212         print(f'$vf_mute = "{vfnamefp_mute}"')
213         print(f'$af = "{afnamefp}"')
214         print(f'# Cut the original footage and create a temporary clip')
215         print(f'{ffmpg_cut_cmd}')
216         print(f'# Swap the audio with the official anechoic track and add 2 sec. leading silence (adelay={leading_silence * 1000})')
217         print(f'{ffmpg_swap_audio_cmd}')
218         print(f'# Pad the audio with silence till the end of video')
219         print(f'{ffmpeg_pad_audio_cmd}')
220         print(f'# Make the mute version of video for UE')
221         print(f'{ffmpeg_rem_audio_cmd}')
222         print(f'# Remove the temporary clips')
223         print(f'{remove_vfnametmp_cmd}')
224     return
225 
226 
227 def find_peaks_rising_edges(peaks, ref_tone_relative_amplitudes_norm, sr):
228     # 'peaks' contains the index of each peaks in the array 'ref_tone_relative_amplitudes_norm'
229     # For each peak, we search in 'ref_tone_relative_amplitudes_norm' the first item preceding the peak
230     # whose amplitude is greater then the 'threshold_rising_edge', i.e. we found where the peak start rising up.
231     # We do the search for a temporal length that is a bit greater than the tone length, starting before the peak.
232     # Find the rising edge is necessary because peaks are not all identify in the same position
233     # in the leading tone (i.e in the middle, but more presumably at the end).
234     # All this stuff under the hypothesis that the start of rising edge is more predictable than the peak position.
235     # Start of rising is a changing condition, while the peaks is a bit more costant situation, even if
236     # near the end of tone we have both the direct sound and both all the reflected sounds, the peak
237     # is not so sharp
```

```
238     peaks_rising_edges = []
239     frames_searching_interval = librosa.time_to_frames(ref_tone_len
    *1.25, sr=sr, n_fft=n_fft)
240     threshold_rising_edge = 0.07
241     print(f'#frames_searching_interval_units={
    frames_searching_interval}')
242     for i, pkidx in enumerate(peaks):
243         #print(f'peak {i}, peak index in array "
    ref_tone_relative_amplitudes_norm" = {pkidx}')
244         # Using np.nonzero() to find the indices where the elements
    are greater than the threshold;
245         # we must accessing the first (and only) array of indices
    with the last '[0]'. (the method return a tupla)
246         #rising_edge_candidates_in_subarray = np.nonzero(
    ref_tone_relative_amplitudes_norm[pkidx -
    frames_searching_interval : pkidx] >= threshold_rising_edge)[0]
247         # break-down in elementary instruction for clarity:
248         searching_subarray = ref_tone_relative_amplitudes_norm[pkidx
    - frames_searching_interval : pkidx]
249         #print(f'peak {i}, searching_subarray = {searching_subarray
    }')
250         idx_in_subarray_of_items_over_threshold = np.nonzero(
    searching_subarray >= threshold_rising_edge)
251         #print(f'peak {i}, idx_in_subarray_of_items_over_threshold={
    idx_in_subarray_of_items_over_threshold}')
252         #print(f'peak {i}, idx_in_subarray_of_items_over_threshold
    [0]={idx_in_subarray_of_items_over_threshold[0]}')
253         rising_edge_candidates_in_subarray =
    idx_in_subarray_of_items_over_threshold[0]  # we get the first
    item of tupla
254         #print(f'peak {i}, rising_edge_candidates_in_subarray (frames
    ) = {rising_edge_candidates_in_subarray}')
255         rising_edge_candidates = pkidx - frames_searching_interval +
    rising_edge_candidates_in_subarray  # position in destination
    array
256         #print(f'peak {i}, rising_edge_candidates (pos in destination
     array)= {rising_edge_candidates}')
257         #print(f'peak {i}, rising_edge_candidate  (in destination
    array)= {rising_edge_candidates[0]} ')
258         peaks_rising_edges.append(rising_edge_candidates[0])
259
260     # transform in numpy array for coherence with others variables
261     peaks_rising_edges = np.array(peaks_rising_edges)
262     return peaks_rising_edges
263
264
265 def main_processing(i_footage_to_process, fn):
266     # get the video file to process
267     footage_file = f'{footages_path_src}/{fn}'
```

```
268     print(f'# footage_file= {footage_file}')
269     # to use the original sample rate of the file, you have to
        explicitly set the the target sample rate to None: sr=None
270     y_stereo, sr = librosa.load(path=footage_file, mono=False, sr=
        None)
271     print(f'# Sample Rate={sr}')
272     y_mono = librosa.to_mono(y_stereo)
273
274     # Compute the STFT
275     short_time_FT = librosa.stft(y_mono, n_fft=n_fft)
276     #print(f'stft[100]={short_time_FT[100][1]}')
277     stft_amplitudes = np.abs(short_time_FT)
278     stft_phases = np.angle(short_time_FT)
279     print(f'# short_time_FT shape = {short_time_FT.shape}')
280     print(f'# stft_amplitudes shape = {stft_amplitudes.shape}')
281     print(f'# stft_phase shape = {stft_phases.shape}')
282     #stft_amplitudes_db = librosa.amplitude_to_db(stft_amplitudes,
        ref=np.max)
283     #print(f'stft_amplitudes_db shape = {stft_amplitudes_db.shape}')
284
285     ## *** plot the STFT *** ###
286     fig, ax = plt.subplots(nrows=1, ncols=1)
287     ### Sample Rate 'sr' must be specified, otherwiese 22050 will be
        used
288     img = librosa.display.specshow(stft_amplitudes, sr=sr, n_fft=
        n_fft, x_axis='time', y_axis='log', ax=ax)
289     ax.set_title("Spettrogramma")
290     fig.colorbar(img, ax=ax, format="%+2.0f dB")
291     plt.xlabel('tempo (m:ss)')
292     plt.show()
293
294     ## == get the time in seconds and the central frequency of the
        bins == ##
295     stft_freq_bins, stft_frames = stft_amplitudes.shape
296     print(f'# stft_freq_bins={stft_freq_bins}')  # the numbers of
        freq. bins
297     print(f'# stft_frames={stft_frames}')        # the numbers of
        frames of the audio track
298     # array of time values to match the time axis from a feature
        matrix
299     # frame_to_time_array[10] = second at frame n.10
300     frame_to_time_array = librosa.times_like(stft_amplitudes, sr=sr,
        n_fft=n_fft)
301     print(f'# frame_to_time_array.shape={frame_to_time_array.shape}')
302     print(f'<# frame_to_time_array[0:8]={frame_to_time_array[0:8]} #>
        ')
303
304     # get the array with central frequencies of bins
305     bins_freq_array = librosa.fft_frequencies(sr=sr, n_fft=n_fft)
```

```
306     print(f'# bins_freq_array len={len(bins_freq_array)}')
307     print(f'# bins_freq_array[0:5] (first 5 central frequencies Hz)=\
        n#{bins_freq_array[0:5]}')
308     print(f'# bins_freq_array[-5:] (last 5 central frequencies Hz)=\n
        #{bins_freq_array[-5:]}')
309
310     # get the bin index of the reference frequency
311     lower_bin_index = math.floor( ref_tone / bin_freq_hop) # central
        freq. of the preceding bin
312     # check if ref. tone is near the preceding bin or the next one
313     if (ref_tone - bins_freq_array[lower_bin_index]) < (bin_freq_hop
        / 2):
314         bin_index_ref = lower_bin_index
315     else:
316         bin_index_ref = lower_bin_index + 1
317     print(f'# reference tone of {ref_tone} Hz drops in bin with index
        = {bin_index_ref}')
318     print(f'# center frequency for bin #{bin_index_ref} = {
        bins_freq_array[bin_index_ref]}')
319
320     # sum column elements, i.e. get the total amplitude for each
        frame
321     stft_amplitudes_sum = np.sum(stft_amplitudes, axis=0)
322     print(f'# stft_amplitudes_sum.shape={stft_amplitudes_sum.shape}')
323
324     # When the ref. tone is played the most of the sound energy is
        concentrated in the bin of the ref.tone
325     # get the row of the reference tone amplitude, i.e for each frame
         we get the amplitude of the bin where reference tone drops
326     ref_tone_amplitudes = stft_amplitudes[bin_index_ref, :]
327     # compute the relative amplitude for the reference tone along all
         the frames
328     ref_tone_relative_amplitudes = np.divide(ref_tone_amplitudes,
        stft_amplitudes_sum)
329     # In case of silence (audio signal = 0) the previous np.divide
        return 'nan'
330     # because the signal and the sum are both equal to 0 and we try
        to compute 0/0.
331     # So we change 'nan' with '0' otherwise the following
        normalization will fail
332     ref_tone_relative_amplitudes = np.nan_to_num(
        ref_tone_relative_amplitudes)
333     # normalization betwenn [0,1]
334     ref_tone_relative_amplitudes_norm = librosa.util.normalize(
        ref_tone_relative_amplitudes)
335
336     ## ==== find the PEAK of leading tones ==== ##
337     # in the prompter the leading tone of 443Hz lenghts 0.3s, so we
        seek for
```

```
338     # peaks spaced at least of this value, but to be sure we use a
        bigger interval
339     tone_len_in_frames = librosa.time_to_frames(ref_tone_len, sr=sr,
        n_fft=n_fft)
340     # minimal horizontal (temporal) distance between neighbouring
        peaks
341     peaks_distance = tone_len_in_frames * 5
342     # find the peaks of amplitude (spaced of 'peaks_distance') in the
         bin of the reference tone
343     # 'peaks' contains the index of the peak values in the
        ref_tone_relative_amplitudes_norm array
344     peaks, _ = find_peaks(ref_tone_relative_amplitudes_norm, height=
        threshold_ampl, distance=peaks_distance)
345
346     ## == we know that for each recording session we need to find 48
        leading tones or other values, so we assert this == ##
347     try:
348         assert len(peaks) == leading_tones_for_recording_session or \
349                 len(peaks) ==
        leading_tones_for_recording_una_tantum_session or \
350                 len(peaks) == 46  ## EXCEPTION FOR footage S6 !!
        Memory full during recording!! What a fuck!
351     except AssertionError:
352         print('# *****')
353         print(f'# Found {len(peaks)} leading tones')
354         print(f'# Number of leading tones differs from {
        leading_tones_for_recording_session} or {
        leading_tones_for_recording_una_tantum_session}')
355         print('# Check threshold_ampl level and peaks_distance or
        maybe some tones are missing!')
356         print('# *****')
357         quit()
358     ## == END assert == ##
359
360     ## ==== find the RISING EDGE of leading tones ==== ##
361     # Now we search for the rising edge of each tone, so we can cut
        mor precisely
362     rising_edges_of_tones = find_peaks_rising_edges(peaks,
        ref_tone_relative_amplitudes_norm, sr)
363     ## == We do the same assert for the rising edge we found == ##
364     try:
365         assert len(rising_edges_of_tones) ==
        leading_tones_for_recording_session or \
366                 len(rising_edges_of_tones) ==
        leading_tones_for_recording_una_tantum_session or \
367                 len(peaks) == 46
368     except AssertionError:
369         print('# *****')
370         print(f'# Found {len(rising_edges_of_tones)} rising edges')
```

```
371         print(f'# Number of rising edges differs from {
    leading_tones_for_recording_session} or {
    leading_tones_for_recording_una_tantum_session}')
372         print('# Check threshold_rising_edge value or maybe some
    tones are missing!')
373         print('# *****')
374         quit()
375     ## == END assert == ##
376
377     ## == Now we calculate the position of the tone where the best
    sync recordings start == ##
378     tone_idxs_of_best_rec_in_footage = []  # a.k.a. peaks index in
    the recording footage of the best records
379     # get the row from the matrix with the index of best recording
    for each sentence in the footage
380     best_rec_for_sentence = footages_best_recs[i_footage_to_process]
381     #print(f'#best_rec_for_sentence={best_rec_for_sentence}')
382     # for each rec get the index of the reference tone
383     for sentence_i, best_rec in enumerate(best_rec_for_sentence):
384         idx = 4 * sentence_i + best_rec
385         tone_idxs_of_best_rec_in_footage.append(idx)
386     # transform in numpy array for coherence with others variables
387     tone_idxs_of_best_rec_in_footage = np.array(
    tone_idxs_of_best_rec_in_footage)
388     #print(f'#tone_idxs_of_best_rec_in_footage={
    tone_idxs_of_best_rec_in_footage}')
389
390     # from peaks array we extract the frame positions of leading tone
     where the best recordings are
391     leading_tones_of_best_rec = peaks[
    tone_idxs_of_best_rec_in_footage]
392     # we do the same for the rising edges
393     rising_edges_of_best_rec = rising_edges_of_tones[
    tone_idxs_of_best_rec_in_footage]
394
395     # In the prompter, before the speech we have a leading tone and a
     bit of silence.
396     # leading tone length is 0.3s plus 0.15s of silence, the rising
    edge is at the very beginning of
397     # the tone where we have recorded only the direct sound;
398     # so, from the rising edge we have found earlier, we wait a bit
    less than the lenght of leading tone
399     # plus the trailing silence (0.15s) before trim the footage.
400     # We assume the hypothesis that we miss 15% of the leading tone
    length, due to some noise and the poor
401     # reproduction/recording audio chain and delay in the detection,
    after some perceptive test.
402     # (we suppose that we hit the rising edge at its very beginning,
    so we wait for that amount of time before trim)
```

```
403     # First we make an array of starting frames (where the footage
        cut begin, to be in synch with the anechoic track)
404     leading_tones_recorded_length = ref_tone_len * 0.85 # sec.
405     leading_silence_time = 0.15  # sec.
406     waiting_time = leading_tones_recorded_length +
        leading_silence_time
407     print(f'# waiting_time={waiting_time} (sec.), in frame={librosa.
        time_to_frames(waiting_time, sr=sr, n_fft=n_fft)}')
408     starting_cut_frames = rising_edges_of_best_rec + librosa.
        time_to_frames(waiting_time, sr=sr, n_fft=n_fft)
409     print(f'# starting_cut_frames.shape={starting_cut_frames.shape}')
410
411     # now convert starting frames in seconds to use with ffmpeg
412     starting_cuts_sec = librosa.frames_to_time(starting_cut_frames,
        sr=sr, n_fft=n_fft)
413
414     # a bit of delay not possible with librosa due to the window
        frame step
415     # we saw that is a bit better
416     offset_start_cut = 0.005  # sec.
417     starting_cuts_sec = starting_cuts_sec + offset_start_cut
418
419     # get the list of audio file of the official test track
420     list_of_audio_files = get_list_of_audio_files(searching_path_af,
        file_basename_with_list_of_audio_files, str(i_footage_to_process))
421
422     # do the dirty job
423     path_dest = f'{footages_path_dst}/{splitted_sentences_dst_dir}_{
        leading_silence}s{trailing_silence}s'
424     trim_footage(scene_to_process, footage_file, path_dest,
        starting_cuts_sec, best_rec_for_sentence, list_of_audio_files,
        leading_silence, trailing_silence)
425
426     # plot normalized amplitudes with peaks and rising edges detected
427     ## time in seconds
428     plt.plot(frame_to_time_array, ref_tone_relative_amplitudes_norm,
        'b-')
429     plt.plot(frame_to_time_array[peaks],
        ref_tone_relative_amplitudes_norm[peaks], "r_")
430     plt.plot(frame_to_time_array[rising_edges_of_tones],
        ref_tone_relative_amplitudes_norm[rising_edges_of_tones], "rx")
431     ## time in frames
432     #plt.plot(ref_tone_relative_amplitudes_norm, 'b-')
433     #plt.plot(peaks, ref_tone_relative_amplitudes_norm[peaks], "rx")
434     plt.xlabel('tempo (s)')
435     plt.ylabel('STFT ampiezza normalizzata')
436     #plt.title(f'Normalized STFT amplitude for bin with center freq={
        bins_freq_array[bin_index_ref]} Hz\n \
```

```
437     #           Peaks are the reference tone {ref_tone} Hz played
    before each sentence\n \
438     #           Green X markers represent the {len(peaks)} peaks found
    \n \
439     #           Red circles are the reference tone of detected sync
    sentences\n \
440     #           Green triangle markers are the detected rising edge')
441
442     plt.title(f'STFT ampiezza normalizzata - Banda con frequenza
    centrale {bins_freq_array[bin_index_ref]} Hz\n \
443               I picchi rappresentano i toni di riferimento (443 Hz)
    antecedenti le singole frasi\n \
444               Trattini orizzontali rossi individuano i valori di
    picco dei toni\n \
445               Pallini verdi individuano i toni di riferimento
    selezionati \n \
446               X rosse individuano i fronti di salita')
447
448     # plot a green star on the best recording tone peaks (unit in sec
     or frames) and for rising/falling edges
449     plt.plot(frame_to_time_array[leading_tones_of_best_rec],
    ref_tone_relative_amplitudes_norm[leading_tones_of_best_rec], "go"
    )
450     #plt.plot(leading_tones_of_best_rec,
    ref_tone_relative_amplitudes_norm[leading_tones_of_best_rec], "g
    *")
451     #plt.plot(frame_to_time_array[rising_edges_of_best_rec],
    ref_tone_relative_amplitudes_norm[rising_edges_of_best_rec], "g*")
452     plt.show()
453     return
454
455
456 def main():
457     ## for debug purpose
458     #i=3
459     #fn=footage_names[i]
460     #print(f'## ===== {i}) Processing footage: ** {fn} ** ===== ##')
461     #main_processing(i, fn)
462
463     for i, fn in enumerate(footage_names):
464         print(f'## ===== {i}) Processing footage: ** {fn} ** ===== ##
    ')
465         main_processing(i, fn)
466         print(f'## ===== END of footage: {fn} ===== ##\n')
467
468     return
469
470
471 if __name__ == '__main__':
```

```
472     main()
```

A.3 cutter.ps1

content/codes/cutter.ps1

```
1  # *** start/stop video cutting ***
2  # Scene to process = SC1M
3  # leading_silence = 2 -> i.e. we can see the performer waiting to
       talk instead of starts immediately
4  # trailing_silence = 2 -> i.e. we can see the performer waiting after
        talk instead of disappear immediately
5  New-Item -Path "E:/compositing/410_SC1M_comps/SC1M_2s2s" -ItemType "
       directory" -Force  # the destination directory
6  $footage = "E:/compositing/410_SC1M_comps/SC1M_s3.mp4"  # the file to
        cut
7
8  # -- sentence n. 0 -- #
9  # audio track C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi
       Andrea Galletto/Target_anechoic/31737.wav, time lenghts =
       2.1183958333333335 sec., sr = 48000
10 Write-Output "# -- Sentence  n.0,  best recorded performance=1 -- #"
11 $vf_tmp1 = "E:/compositing/410_SC1M_comps/SC1M_2s2s/31737_tmp.mp4"
12 $vf_tmp2 = "E:/compositing/410_SC1M_comps/SC1M_2s2s/31737_tmp2.mp4"
13 $vf = "E:/compositing/410_SC1M_comps/SC1M_2s2s/SC1M_31737.mp4"
14 $vf_mute = "E:/compositing/410_SC1M_comps/SC1M_2s2s/SC1M_31737_mute.
       mp4"
15 $af = "C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea
       Galletto/Target_anechoic/31737.wav"
16 # Cut the original footage and create a temporary clip
17 ffmpeg -i "$footage" -ss 9.54633 -to 15.66473 -c:v libx264 -preset
       fast -qp 0 -b:v 20M -an "$vf_tmp1"
18 # Swap the audio with the official anechoic track and add 2 sec.
       leading silence (adelay=2000)
19 ffmpeg -i "$vf_tmp1" -i "$af" -filter_complex "[1]adelay=2000[out];[
       out]amix=inputs=1" -c:v copy -b:a 192K -map 0:v:0 -map 1:a:0 -an "
       $vf_tmp2"
20 # Pad the audio with silence till the end of video
21 ffmpeg -i "$vf_tmp2" -filter_complex "[0:a]apad[a1];[a1]amix=inputs=1
       " -c:v copy -b:a 192k -shortest "$vf"
22 # Make the mute version of video for UE
23 ffmpeg -i "$vf" -c:v copy -an "$vf_mute"
24 # Remove the temporary clips
25 Remove-Item -Path "$vf_tmp1","$vf_tmp2" -Force
26
27 # -- sentence n. 1 -- #
```

```
# audio track C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic/31843.wav, time lenghts = 2.1655 sec., sr = 48000
Write-Output "# --- Sentence n.1, best recorded performance=1 --- #"
$vf_tmp1 = "E:/compositing/410_SC1M_comps/SC1M_2s2s/31843_tmp.mp4"
$vf_tmp2 = "E:/compositing/410_SC1M_comps/SC1M_2s2s/31843_tmp2.mp4"
$vf = "E:/compositing/410_SC1M_comps/SC1M_2s2s/SC1M_31843.mp4"
$vf_mute = "E:/compositing/410_SC1M_comps/SC1M_2s2s/SC1M_31843_mute.mp4"
$af = "C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic/31843.wav"
# Cut the original footage and create a temporary clip
ffmpeg -i "$footage" -ss 33.29033 -to 39.45583 -c:v libx264 -preset fast -qp 0 -b:v 20M -an "$vf_tmp1"
# Swap the audio with the official anechoic track and add 2 sec. leading silence (adelay=2000)
ffmpeg -i "$vf_tmp1" -i "$af" -filter_complex "[1]adelay=2000[out];[out]amix=inputs=1" -c:v copy -b:a 192K -map 0:v:0 -map 1:a:0 -an "$vf_tmp2"
# Pad the audio with silence till the end of video
ffmpeg -i "$vf_tmp2" -filter_complex "[0:a]apad[a1];[a1]amix=inputs=1" -c:v copy -b:a 192k -shortest "$vf"
# Make the mute version of video for UE
ffmpeg -i "$vf" -c:v copy -an "$vf_mute"
# Remove the temporary clips
Remove-Item -Path "$vf_tmp1","$vf_tmp2" -Force
```

A.4 joined_tracks_maker.py

content/codes/joined_tracks_maker.py

```
'''
this script creates the audio track to feed the AI process Audio2Face
to animate the actress avatar: for each anachoic tracks it add 2 second
of leading and trailing silence, and join them all togheter.
To create the audio track we use ffmpeg, so this Python script constructs
the PowerShell script to make all the stuff.
'''

import librosa

## ==== AUDIO parameters ==== ##
file_audio_extension = ".wav"
# Path realitve to the reference audiotrack, i.e the audio track of the official test
```

```
14 searching_path_af = "C:/Users/andrea/OneDrive - Politecnico di Torino
       /Tesi Andrea Galletto/Target_anechoic"
15 #file_with_list_of_audiofile_to_process = "list_of_audio_files_0.txt"
16 file_basename_with_list_of_audio_files = ("list_of_audio_files_", ".
       txt")
17
18 # The suffix of the final file with all the test tracks in the
       correct order and with the 2s of leading and trailing silence
19 joined_tracks_suffix_fn = "joined_tracks.wav"
20 # The txt file with the timestamp for each track (start and stop),
       including the leading and trailing 2s
21 joined_tracks_timestamps_suffix_fn = "joined_tracks_timestamps.txt"
22 # we save the joined track in the same place of source tracks
23 joined_tracks_path_dst = searching_path_af
24
25 # Amount of seconds to retain before start to trim the footage with
       respect to the beginning of the
26 # audio track of official audiometric test (for ex. the file 00252.
       wav).
27 # i.e. for example 2 seconds before the performer start to talk.
28 # (in the actual test there are 2 sec. of silence before the subject
       can earing the sentence).
29 # So, if we put here 2 sec. we have 2 sec. more in the video during
       we can see the performer waiting to talk,
30 # otherwise we will see the performer as freeze until the talk
31 leading_silence = 2  # sec.
32 # The same as leading silence, but at the end of the sentence, with
       respet to the end of the
33 # official track of audiometric test
34 trailing_silence = 2  # sec.
35
36
37 # For each tracks set, we get the list of the audio file playedback
       from the text file {file_basename_with_list_of_audio_file}
38 def get_list_of_audio_files(searching_path,
       file_basename_with_list_of_audio_file, tracks_set_to_process):
39     #print(f'tracks_set_to_process={tracks_set_to_process}')
40     basename = file_basename_with_list_of_audio_file[0]
41     ext = file_basename_with_list_of_audio_file[1]
42     file_name = searching_path + "/" + basename +
       tracks_set_to_process + ext   #i.e. ...path.../
       list_of_audio_files_[0-9].txt
43     #print(f'file_name={file_name}')
44     list_of_audio_files = []
45     try:
46         with open(file_name, "r") as f:
47             print(f'# try reading file={file_name}')
48             list_of_audio_files = f.readlines()
49         f.close()
```

```
50      except Exception as e:
51          print(f"# ERROR reading the file {file_name}")
52          quit()
53      # We have in list_of_audio_files the following:
54          #C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic/00152.wav
55          #C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic/00264.wav
56      return list_of_audio_files
57
58
59  def get_file_name(full_path):
60      ## keep just the filename without the extension
61      # we get substring following the last "/", i.e the full file name with extension
62      prefix_tmp,delim_tmp,full_fname = full_path.strip().rpartition("/")
63      # rid of the extension
64      fname,delim_tmp,suffix_tmp = full_fname.strip().rpartition(file_audio_extension)
65      return fname
66
67
68  def tracks_joiner(tracks_set_to_process, list_of_audio_files, path_dest):
69      if (tracks_set_to_process == -1):
70          tracks_set_to_process = '0to9'
71      joined_tracks = f'{joined_tracks_path_dst}/{tracks_set_to_process}_{joined_tracks_suffix_fn}'  # the final joined file
72      joined_tracks_tmp = f'{joined_tracks_path_dst}/TMP_{tracks_set_to_process}_{joined_tracks_suffix_fn}'
73      joined_tracks_timestamps = f'{joined_tracks_path_dst}/{tracks_set_to_process}_{joined_tracks_timestamps_suffix_fn}'
74      print('#------------------------------------------------------------------------------')
75      print('#------------------------------------------------------------------------------')
76      print(f'# *** ========== Cut & Paste in a powershell script, then run it ========== ***\n')
77      print(f'# *** Tracks-set to process = {tracks_set_to_process} ***#')
78      print(f'# leading_silence = {leading_silence} -> i.e. we can see the performer waiting to talk instead of starts immediately')
79      print(f'# trailing_silence = {trailing_silence} -> i.e. we can see the performer waiting after talk instead of disappear immediately')
```

```
80      print(f'$joined_tracks = "{joined_tracks}"  # the final joined
        file')
81      print(f'$joined_tracks_tmp = "{joined_tracks_tmp}"  # the
        temporary joined file')
82      print(f'$aftmp1 = "{path_dest}/aftmp1.{file_audio_extension}"')
83      print(f'$aftmp2 = "{path_dest}/aftmp2.{file_audio_extension}"')
84
85      timestamp_list = []
86      current_start_timestamp = 0  # sec
87      # let's cycle on every track
88      for i, fn in enumerate(list_of_audio_files):
89          print(f'\n# {i}) joining track {fn.strip()}')
90          print(f'$af = "{fn.strip()}"')
91          # get the duration of audio track to compute the ending cut
        time
92          atrack, sr = librosa.load(path=fn.strip(), sr=None)
93          atrack_len = librosa.get_duration(y=atrack, sr=sr)
94          print(f'# track lenght = {atrack_len} sec., sr = {sr}')
95
96          # Create the ffmpeg commands
97          # Add 2S leading silence
98          ffmpeg_leading_silence_cmd = f'ffmpeg -y -i "$af" -af "adelay
        ={leading_silence}s:all=true" -b:a 192K "$aftmp1"'
99          # Add 2S trailing silence
100         ffmpeg_trailing_silence_cmd = f'ffmpeg -y -i "$aftmp1" -
        filter_complex "aevalsrc=0:d={trailing_silence}[silence];[0:a][
        silence]concat=n=2:v=0:a=1[out]" -map "[out]" "$aftmp2"'
101         # Set the right command for the first iteration
102         if (i==0):
103             # First iteration, nothing to join, just copy the track
104             ffmpeg_join_cmd = f'ffmpeg -y -i "$aftmp2" -c:a copy "
        $joined_tracks"'
105         else:
106             # append the current track
107             ffmpeg_join_cmd = f'ffmpeg -y -i "$joined_tracks_tmp" -i
        "$aftmp2" -filter_complex "[0:a][1:a]concat=n=2:v=0:a=1[out]" -map
         "[out]" "$joined_tracks"'
108         # update the temporary joined track with the last joined
        track
109         ffmpeg_upd_joined_track_tmp_cmd = f'ffmpeg -y -i "
        $joined_tracks" -c:a copy "$joined_tracks_tmp"'
110         remove_tempf_cmd = f'Remove-Item -Path $joined_tracks_tmp,
        $aftmp1, $aftmp2 -Force'
111
112         # Make the PowerShell script
113         print(f'Write-Output "# ------ Track n.{i} ------ #"')
114         print(f'# Add 2s leading')
115         print(f'{ffmpeg_leading_silence_cmd}')
116         print(f'# Add 2s trailing')
```

```
117             print(f'{ffmpeg_trailing_silence_cmd}')
118             print(f'# Make the join')
119             print(f'{ffmpeg_join_cmd}')
120             print(f'# Upd the temp join file for next iteration')
121             print(f'{ffmpeg_upd_joined_track_tmp_cmd}')
122 
123             # stopping timestamp (sec)
124             stop_s = current_start_timestamp + leading_silence +
    atrack_len + trailing_silence
125             fname_only = get_file_name(fn) + file_audio_extension
126             timestamp_list.append((fname_only, current_start_timestamp,
    stop_s))
127             current_start_timestamp = stop_s
128 
129         print(f'# Remove temp files')
130         print(f'{remove_tempf_cmd}')
131 
132         try:
133             with open(joined_tracks_timestamps, "w") as f:
134                 print(f'#try writing file={joined_tracks_timestamps}, the
    same content follows')
135                 # Save the timestamps
136                 for ts in timestamp_list:
137                     print(f'# {ts}')
138                     f.write(f'{ts}\n')
139             f.close()
140         except Exception as e:
141             print(f"#ERROR writing the file {joined_tracks_timestamps}")
142             quit()
143 
144         return
145 
146 
147 def main_processing(tracks_set_to_process):
148     # if i=-1 process all tracks
149     if (tracks_set_to_process == -1):
150         ## All joined tracks 0 to 9
151         list_of_audio_files = get_list_of_audio_files(
    searching_path_af, file_basename_with_list_of_audio_files, '0to9')
152     else:
153         # get the list of audio file of the official test track
154         list_of_audio_files = get_list_of_audio_files(
    searching_path_af, file_basename_with_list_of_audio_files, str(
    tracks_set_to_process))
155 
156     # do the dirty job
157     tracks_joiner(tracks_set_to_process, list_of_audio_files,
    joined_tracks_path_dst)
158     return
```

```
159
160
161 def main():
162     # NOTE: the tracks set is the set of the 12 audio tracks with the
        same protagonist (Sofia, Andrea...) from 0 to 9
163     # i.e.: it is the first char in the filename of the anechoic
        track
164
165     # for debug purpose
166     #i=0
167     #print(f'## ===== {i}) Processing the tracks set n. {i} =====
        ##')
168     #main_processing(i)
169
170     i=-1  # -1 means join all the 120 tracks togheter
171     print(f'## ===== {i}) Processing the tracks set n. {i} ===== ##')
172     main_processing(i)
173
174     #for i in range(10):
175     #    print(f'## ===== {i}) Processing tracks set: ** {i} ** =====
        ##')
176     #    main_processing(i)
177     #    print(f'## ===== END of tracks set: {i} ===== ##\n')
178
179     return
180
181
182 if __name__ == '__main__':
183     main()
```

A.5 seek_lips_sync_by_timestamps.py

content/codes/seek_lips_sync_by_timestamps.py

```
1 '''
2 this script creates the command line to execute to cut with ffmpeg
    the
3 full scene footage with avatar into a single footage for each
    sentence,
4 saving the results in a dedicated directory.
5
6 To set the cut point it uses the timestamps file [0-9]
    _joined_tracks_timestamps.txt
7
8 '''
9 import ast
10 import ffmpeg
```

```
11
12 ## ==== AUDIO parameters ==== ##
13 file_audio_extension = ".wav"
14 # Path realitve to the reference audiotrack, i.e the audio track of
       the official test
15 searching_path_af = "C:/Users/andrea/OneDrive - Politecnico di Torino
       /Tesi Andrea Galletto/Target_anechoic"
16 #file_with_list_of_audiofile_to_process = "list_of_audio_files_0.txt"
17 file_basename_with_list_of_audio_files = ("list_of_audio_files_", ".
       txt")
18
19 # The txt file with the timestamp for each track (start and stop),
       including the leading and trailing 2s
20 joined_tracks_timestamps_suffix_fn = "joined_tracks_timestamps.txt"
21 # we save the joined track in the same place of source tracks
22 joined_tracks_path_dst = searching_path_af
23
24 ## ==== VIDEO parameters ==== ##
25 # It is the whole scene with the talker uttering all the sentences of
        a set (audio file whose name starts with the same number)
26 scene_to_process = "SC1M_avatar"  # SC1M_avatar, SC2M_avatar,
       SC3M_avatar
27
28 footages_path_src = f'E:/compositing/410_{scene_to_process}_comps'
29 footages_path_dst = f'E:/compositing/410_{scene_to_process}_comps'
30
31 # subdirectory in the {footages_path_dst} to keep the cutted clips
32 splitted_sentences_dst_dir = f'{scene_to_process}'  # footage number
       [0-9] will be appended (ex: SC2M_avatar_s0)
33
34 # The footage with all the audio tracks joined together from set 0 to
        set 9
35 footage_names_410_SCxM_s0to9 = [f'{scene_to_process}_s0to9.mp4'] #
       SCxM_avatar_s0to9.mp4
36 # Select the correct footages list to process
37 footage_names = footage_names_410_SCxM_s0to9
38
39 FPS = 29.97
40 # Amount of seconds to retain before start to trim the footage with
       respect to the beginning of the
41 # audio track of official audiometric test (for ex. the file 00252.
       wav).
42 # i.e. for example 2 seconds before the performer start to talk.
43 # (in the actual test there are 2 sec. of silence before the subject
       can earing the sentence).
44 # So, if we put here 2 sec. we have 2 sec. more in the video during
       we can see the performer waiting to talk,
45 # otherwise we will see the performer as freeze until the talk
46 leading_silence = 2  # sec.
```

```
47 # The same as leading silence, but at the end of the sentence, with respet to the end of the
48 # official track of audiometric test
49 trailing_silence = 2  # sec.
50
51
52 # For each recording we get the list of the audio file playedback from the text file {file_basename_with_list_of_audio_file}
53 def get_list_of_audio_files(searching_path, file_basename_with_list_of_audio_file, i_footage_to_process):
54     #print(f'i_footage_to_process={i_footage_to_process}')
55     basename = file_basename_with_list_of_audio_file[0]
56     ext = file_basename_with_list_of_audio_file[1]
57     file_name = searching_path + "/" + basename + i_footage_to_process + ext
58     #print(f'file_name={file_name}')
59     list_of_audio_files = []
60     try:
61         with open(file_name, "r") as f:
62             print(f'# try reading file={file_name}')
63             list_of_audio_files = f.readlines()
64         f.close()
65     except Exception as e:
66         print(f"#ERROR reading the file {file_name}")
67         quit()
68     # We have in list_of_audio_files the following:
69         #C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic/00152.wav
70         #C:/Users/andrea/OneDrive - Politecnico di Torino/Tesi Andrea Galletto/Target_anechoic/00264.wav
71     return list_of_audio_files
72
73
74 def get_file_name(full_path):
75     ## keep just the filename without the extension
76     # we get substring following the last "/", i.e the full file name with extension
77     prefix_tmp,delim_tmp,full_fname = full_path.strip().rpartition("/")
78     # rid of the extension
79     fname,delim_tmp,suffix_tmp = full_fname.strip().rpartition(file_audio_extension)
80     return fname
81
82
83 def trim_footage(scene_to_process, footage_file, path_dest, timestamps_list, list_of_audio_files):
84     print(f'#** trim_footage() start the dirty job **')
85     # get fps
```

```
86      info_json = ffmpeg.probe(footage_file)
87      fps = info_json['streams'][0]['r_frame_rate']
88      print(f'#fps={fps} -- type={type(fps)}')
89
90      # ** Before start we check for the correct FPS of the footage **
91      try:
92          assert fps == "30000/1001" or \
93                 fps == "29.97" or \
94                 fps == "2997/100"
95      except AssertionError:
96          print('#*****')
97          print('# fps not 29.97')
98          print('# Check video format')
99          print('#*****')
100         quit()
101     ## ** End of assert ** ##
102     print('
    #-------------------------------------------------------------------------------
    ')
103     print(f'# *** ========== Cut & Paste in a powershell script, then
        run it ========== ***\n')
104     print(f'# *** start/stop video cutting ***')
105     print(f'# Scene to process = {scene_to_process}')
106     print(f'New-Item -Path "{path_dest}" -ItemType "directory" -Force
         # the destination directory')
107     print(f'$footage = "{footage_file}"  # the file to cut')
108
109     #for i, start_s in enumerate(start_cuts_s):
110     for i, (afn, start_s, stop_s)  in enumerate(timestamps_list):
111         print(f'\n# -- sentence n. {i} -- #')
112         # compute some parameters
113         vfname = get_file_name(list_of_audio_files[i].strip())  # the
        video file keeps same name of audio track
114         vfnamefp = f'{path_dest}/{scene_to_process}_{vfname}.mp4' #
    final video file name full path
115         vfnamefp_mute = f'{path_dest}/{scene_to_process}_{vfname}
    _mute.mp4' # final video file name full path (mute version)
116
117         ## CUT
118         video_opts = "-c:v libx264 -preset fast -qp 0 -b:v 20M"  #
    codec di ffmpeg
119         ## UE do not like clip with audio , so we remove it
120         ffmpg_cut_cmd1 = f'ffmpeg -i "$footage" -ss {(start_s):.5f} -
    to {stop_s:.5f} {video_opts} -c:a copy "$vf"'
121         ffmpg_cut_cmd2 = f'ffmpeg -i "$vf" -c:v copy -an "$vf_mute"'
122
123         # Make the PowerShell script
124         print(f'Write-Output "# -- Sentence  n.{i} -- #"')
125         print(f'$vf = "{vfnamefp}"')
```

```
126         print(f'$vf_mute = "{vfnamefp_mute}"')
127         print(f'# Cut the original footage keeping the audio')
128         print(f'{ffmpg_cut_cmd1}')
129         print(f'# Cut the original footage ridding of the audio')
130         print(f'{ffmpg_cut_cmd2}')
131     return
132 
133 
134 def main_processing(i_footage_to_process, fn):
135     # get the video file to process
136     footage_file = f'{footages_path_src}/{fn}'
137     print(f'# footage_file= {footage_file}')
138 
139     if (i_footage_to_process == -1):
140         i_footage_to_process = '0to9'
141         # get the list of audio file of the official test track
142         list_of_audio_files = get_list_of_audio_files(
    searching_path_af, file_basename_with_list_of_audio_files,
    i_footage_to_process)
143     else:
144         list_of_audio_files = get_list_of_audio_files(
    searching_path_af, file_basename_with_list_of_audio_files, str(
    i_footage_to_process))
145 
146     timestamps_list = []
147     # Load the timestamp from file
148     timesstamp_fp = f'{searching_path_af}/{i_footage_to_process}_{
    joined_tracks_timestamps_suffix_fn}'
149     try:
150         with open(timesstamp_fp, "r") as f:
151             print(f'# try reading file={timesstamp_fp}')
152             # read the file line by line
153             for r in f:
154                 # Read the line as a tupla with ast.literal_eval()
155                 tupla = ast.literal_eval(r)
156                 # append to the list
157                 timestamps_list.append(tupla)
158         f.close()
159     except Exception as e:
160         print(f"# ERROR reading the file {timesstamp_fp}")
161         quit()
162 
163     # do the dirty job
164     path_dest = f'{footages_path_dst}/{splitted_sentences_dst_dir}_{
    leading_silence}s{trailing_silence}s'
165     trim_footage(scene_to_process, footage_file, path_dest,
    timestamps_list, list_of_audio_files)
166 
167     return
```

```
168
169
170 def main():
171     # Exception for footage with all the tracks joined together
172     i=-1
173     fn=footage_names[0]
174     main_processing(i, fn)
175     return
176
177
178 if __name__ == '__main__':
179     main()
```

Bibliografia

[1] World Health Organization. *World report on hearing*. World Health Organization, 2021, xiv, 252 p. (Cit. alle pp. 1, 2).

[2] Lega del Filo d'Oro. *Area della Comunicazione*. Accesso: 14 luglio 2024. 2024. URL: https://www.legadelfilodoro.it/it/come-aiutiamo/area-della-comunicazione (cit. a p. 1).

[3] Abby McCormack e Heather Fortnum. «Why do people fitted with hearing aids not wear them?» In: *International journal of audiology* 52.5 (2013), pp. 360–368 (cit. a p. 2).

[4] Richard H Wilson e Wendy B Cates. «A comparison of two word-recognition tasks in multitalker babble: Speech Recognition in Noise Test (SPRINT) and Words-in-Noise Test (WIN)». In: *Journal of the American Academy of Audiology* 19.07 (2008), pp. 548–556 (cit. a p. 2).

[5] Richard H Wilson. «Clinical experience with the words-in-noise test on 3430 veterans: Comparisons with pure-tone thresholds and word recognition in quiet». In: *Journal of the American Academy of Audiology* 22.07 (2011), pp. 405–423 (cit. a p. 2).

[6] Stefan Fichna, Thomas Biberger, Bernhard Seeber e Stephan Ewert. «Effect of Acoustic Scene Complexity and Visual Scene Representation on Auditory Perception in Virtual Audio-Visual Environments». In: set. 2021, pp. 1–9. DOI: 10.1109/I3DA48870.2021.9610916 (cit. alle pp. 3, 22).

[7] Leo Beranek e Tim Mellow. *Acoustics: Sound Fields, Transducers and Vibration*. Academic Press, 2019. ISBN: 9780128152270 (cit. alle pp. 5–7).

[8] F. Alton Everest. *Manuale di Acustica*. Hoepli, 1996, p. 444. ISBN: 9788820322885 (cit. a p. 6).

[9] American Speech-Language-Hearing Association. *Guidelines for the Speech Reception Threshold*. Accessed: 2024-07-13. 1988. URL: https://www.asha.org/policy/gl1988-00008/ (cit. a p. 6).

[10] Ruth Litovsky. «Spatial Release from Masking». In: *Acoustics Today* 8 (gen. 2012), p. 18. DOI: 10.1121/1.4729575 (cit. a p. 6).

[11] Melissa M. Baese-Berk, Susannah V. Levi e Kristin J. Van Engen. «Intelligibility as a measure of speech perception: Current approaches, challenges, and recommendationsa)». In: *The Journal of the Acoustical Society of America* 153.1 (gen. 2023), pp. 68–76. ISSN: 0001-4966. DOI: 10.1121/10.0016806. eprint: https://pubs.aip.org/asa/jasa/article-pdf/153/1/68/16723457/68_1_online.pdf. URL: https://doi.org/10.1121/10.0016806 (cit. a p. 7).

[12] Treccani. *HRTF*. https://www.treccani.it/enciclopedia/hrtf_(Enciclopedia-della-Scienza-e-della-Tecnica)/. Accesso il: 14 luglio 2024 (cit. a p. 7).

[13] M. Kleiner. *Acoustics and Audio Technology*. Fort Lauderdale, FL: J. Ross Publishing, 2011 (cit. a p. 7).

[14] S. Barré, D. Döbler e Andy Meyer. «Room impulse response measurement with a spherical microphone array, application to room and building acoustics». In: (gen. 2014) (cit. a p. 8).

[15] Giuseppina Emma Puglisi, Anna Warzybok, Sabine Hochmuth, Chiara Visentin, Arianna Astolfi, Nicola Prodi e Birger Kollmeier. «An Italian matrix sentence test for the evaluation of speech intelligibility in noise». In: *International Journal of Audiology* 54.sup2 (2015), pp. 44–50. DOI: 10.3109/14992027.2015.1061709. URL: https://doi.org/10.3109/14992027.2015.1061709 (cit. alle pp. 9, 27, 58).

[16] Jennifer L. Campos e Stefan Launer. «From Healthy Hearing to Healthy Living: A Holistic Approach». In: *Ear and Hearing* 41 (2020). Cited by: 11; All Open Access, Hybrid Gold Open Access, 99S–106S. DOI: 10.1097/AUD.0000000000000931. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85094818361&doi=10.1097%2fAUD.0000000000000931&partnerID=40&md5=2e530bfa5bc0ec71417b5019c53b8826 (cit. a p. 9).

[17] https://www.britannica.com/science/ecological-validity.. Accessed: 2023-6-13 (cit. a p. 9).

[18] G. Keidser et al. «The Quest for Ecological Validity in Hearing Science: What It Is, Why It Matters, and How to Advance It». English. In: *Ear and hearing* 41 (2020). Cited By :61, 5S–19S. URL: www.scopus.com (cit. a p. 9).

[19] Thomas D. Parsons. «Virtual Reality for Enhanced Ecological Validity and Experimental Control in the Clinical, Affective and Social Neurosciences». In: *Frontiers in Human Neuroscience* 9 (2015). ISSN: 1662-5161. DOI: 10.3389/fnhum.2015.00660. URL: https://www.frontiersin.org/articles/10.3389/fnhum.2015.00660 (cit. a p. 10).

[20] G. Grimm, B. Kollmeier e V. Hohmann. «Spatial acoustic scenarios in multichannel loudspeaker systems for hearing aid evaluation». English. In: *Journal of the American Academy of Audiology* 27.7 (2016). Cited By :26, pp. 557–566. URL: www.scopus.com (cit. a p. 10).

[21] Naim Mansour, Marton Marschall, Tobias May, Adam Westermann e Torsten Dau. «Speech intelligibility in a realistic virtual sound environment». In: *Journal of the Acoustical Society of America* 149.4 (2021). Cited by: 3; All Open Access, Green Open Access, pp. 2791–2801. DOI: 10.1121/10.0004779. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85104636947&doi=10.1121%2f10.0004779&partnerID=40&md5=1481cc6a3fd0b19dd14836c9db7131c2 (cit. a p. 10).

[22] N. P. Erber. «Auditory visual perception of speech». English. In: *Journal of Speech and Hearing Disorders* 40.4 (1975). Cited By :208, pp. 481–492. URL: www.scopus.com (cit. a p. 10).

[23] H. Mcgurk e J. Macdonald. «Hearing lips and seeing voices». English. In: *Nature* 264.5588 (1976). Cited By :4183, pp. 746–748. URL: www.scopus.com (cit. alle pp. 10, 11).

[24] S. Van De Par et al. «Auditory-visual scenes for hearing research». English. In: *Acta Acustica* 6 (2022). Cited By :4. URL: www.scopus.com (cit. alle pp. 10, 12).

[25] K. Smeds, S. Gotowiec, F. Wolters, P. Herrlin, J. Larsson e M. Dahlquist. «Selecting Scenarios for Hearing-Related Laboratory Testing». English. In: *Ear and hearing* 41 (2020). Cited By :16, 20S–30S. URL: www.scopus.com (cit. a p. 10).

[26] G. Llorach, G. Grimm, M. M. E. Hendrikse e V. Hohmann. «Towards realistic immersive audiovisual simulations for hearing research capture, virtual scenes and reproduction». English. In: *AVSU 2018 - Proceedings of the 2018 Workshop on Audio-Visual Scene Understanding for Immersive Multimedia, Colocated with MM 2018*. Cited By :11. 2018, pp. 33–40. URL: www.scopus.com (cit. alle pp. 10, 13, 14).

[27] Marc Swerts e Emiel Krahmer. «Visual Prosody Across Cultures». In: *The Oxford Handbook of Language Prosody*. Oxford University Press, dic. 2020. ISBN: 9780198832232. DOI: 10.1093/oxfordhb/9780198832232.013.45. eprint: https://academic.oup.com/book/0/chapter/298317391/chapter-ag-pdf/44507599/book_34870_section_298317391.ag.pdf. URL: https://doi.org/10.1093/oxfordhb/9780198832232.013.45 (cit. a p. 11).

[28] P. Badin, Y. Tarabalka, F. Elisei e G. Bailly. «Can you 'read' tongue movements? Evaluation of the contribution of tongue display to speech understanding». English. In: *Speech Communication* 52.6 (2010). Cited By :55, pp. 493–503. URL: www.scopus.com (cit. a p. 11).

[29] D. W. Massaro e J. Light. «Using Visible Speech to Train Perception and Production of Speech for Individuals with Hearing Loss». English. In: *Journal of Speech, Language, and Hearing Research* 47.2 (2004). Cited By :87, pp. 304–320. URL: www.scopus.com (cit. a p. 11).

[30] W. H. Sumby e Irwin Pollack. «Visual Contribution to Speech Intelligibility in Noise». In: *The Journal of the Acoustical Society of America* 26.2 (giu. 2005), pp. 212–215. ISSN: 0001-4966. DOI: 10.1121/1.1907309. eprint: https://pubs.aip.org/asa/jasa/article-pdf/26/2/212/12193669/212_1_online.pdf. URL: https://doi.org/10.1121/1.1907309 (cit. a p. 12).

[31] H. Laux, A. Hallawa, J. C. S. Assis, A. Schmeink, L. Martin e A. Peine. «Two-stage visual speech recognition for intensive care patients». English. In: *Scientific Reports* 13.1 (2023). URL: www.scopus.com (cit. a p. 12).

[32] C. Santos, A. Cunha e P. Coelho. *A Review on Deep Learning-Based Automatic Lipreading*. English. Vol. 484 LNICST. Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST. 2023, pp. 180–195. URL: www.scopus.com (cit. a p. 12).

[33] Sebastian Puschmann, Mareike Daeglau, Maren Stropahl, Bojana Mirkovic, Stephanie Rosemann, Christiane M. Thiel e Stefan Debener. «Hearing-impaired listeners show increased audiovisual benefit when listening to speech in noise». In: *NeuroImage* 196 (2019), pp. 261–268. ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2019.04.017. URL: https://www.sciencedirect.com/science/article/pii/S1053811919303039 (cit. a p. 12).

[34] Lauren V Hadley, W Owen Brimijoin e William M Whitmer. «Speech, movement, and gaze behaviours during dyadic conversation in noise». In: *Scientific reports* 9.1 (2019), pp. 1–8 (cit. a p. 12).

[35] Maartje M.E. Hendrikse, Gerard Llorach, Giso Grimm e Volker Hohmann. «Influence of visual cues on head and eye movements during listening tasks in multi-talker audiovisual environments with animated characters». In: *Speech Communication* 101 (2018), pp. 70–84. ISSN: 0167-6393. DOI: https://doi.org/10.1016/j.specom.2018.05.008. URL: https://www.sciencedirect.com/science/article/pii/S0167639317302558 (cit. a p. 13).

[36] M. L. Iuzzolino e K. Koishida. «AV(Se)2: Audio-visual squeeze-excite speech enhancement». English. In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Vol. 2020-May. Cited By :7. 2020, pp. 7539–7543. URL: www.scopus.com (cit. a p. 13).

[37] R. Rashmi Adyapady e B. Annappa. «A comprehensive review of facial expression recognition techniques». In: *Multimedia Systems* 29.1 (2023). Cited by: 1, pp. 73–103. DOI: 10.1007/s00530-022-00984-w. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85135257001&doi=10.1007%2fs00530-022-00984-w&partnerID=40&md5=ba77939f4afc65d6183a4a0db299956f (cit. a p. 13).

[38] M. Garnier, L. Ménard e G. Richard. «Effect of being seen on the production of visible speech cues. A pilot study on Lombard speech». English. In: *13th Annual Conference of the International Speech Communication Association 2012, INTERSPEECH 2012*. Vol. 1. Cited By :11. 2012, pp. 610–613. URL: www.scopus.com (cit. a p. 14).

[39] Michael A. Gerzon. «Periphony: With-Height Sound Reproduction». In: *J. Audio Eng. Soc* 21.1 (1973), pp. 2–10. URL: http://www.aes.org/e-lib/browse.cfm?elib=2012 (cit. a p. 14).

[40] Daniel Arteaga. *Introduction to Ambisonics*. Ver. 0.7. Mag. 2023. DOI: 10.5281/zenodo.7963106. URL: https://doi.org/10.5281/zenodo.7963106 (cit. alle pp. 15, 17–19).

[41] Franz Zotter e Matthias Frank. *Ambisonics: A Practical 3D Audio Theory for Recording, Studio Production, Sound Reinforcement, and Virtual Reality*. Cham, Switzerland: Springer, 2019. ISBN: 978-3-030-17207-7. DOI: 10.1007/978-3-030-17207-7. URL: https://link.springer.com/book/10.1007/978-3-030-17207-7 (cit. a p. 18).

[42] Michael Schutte, Stephan D. Ewert e Lutz Wiegrebe. «The percept of reverberation is not affected by visual room impression in virtual environments». In: *The Journal of the Acoustical Society of America* 145.3 (mar. 2019), EL229–EL235. ISSN: 0001-4966. DOI: 10.1121/1.5093642. URL: https://doi.org/10.1121/1.5093642 (cit. a p. 22).

[43] Christoph Kirsch, Torben Wendt, Steven Van De Par, Hongmei Hu e Stephan D. Ewert. «Computationally-Efficient Simulation of Late Reverberation for Inhomogeneous Boundary Conditions and Coupled Rooms». In: *J. Audio Eng. Soc* 71.4 (2023), pp. 186–201. URL: https://www.aes.org/e-lib/browse.cfm?elib=22040 (cit. a p. 22).

[44] Christoph Kirsch, Torben Wendt, Steven Van De Par, Hongmei Hu e Stephan D. Ewert. *RAZR Engine*. Last accessed 2024-04-11. URL: https://medi.uni-oldenburg.de/razr/ (cit. a p. 22).

[45] Epic games. *Unreal Engine*. Last accessed 2024-04-11. URL: www.unrealengine.com (cit. alle pp. 22, 27).

[46] Martin Ochmann, Michael Vorländer e Janina Fels, cur. *Proceedings of the 23rd International Congress on Acoustics : integrating 4th EAA Euroregio 2019 : 9-13 September 2019 in Aachen, Germany*. 23. International Congress on Acoustics, Aachen (Germany), 9 Sep 2019 - 13 Sep 2019. Berlin, Germany: Deutsche Gesellschaft für Akustik, 9 set. 2019, 1 Online–Ressource (8256 Seiten) : Illustrationen. ISBN: 978-3-939296-15-7. URL: https://publications.rwth-aachen.de/record/767416 (cit. a p. 22).

[47] MakeHuman Community. *MakeHuman*. Last accessed 2024-04-12. URL: http://www.makehumancommunity.org (cit. a p. 23).

[48] Jacques A. Grange e John F. Culling. «The benefit of head orientation to speech intelligibility in noise». In: *The Journal of the Acoustical Society of America* 139.2 (feb. 2016), pp. 703–712. ISSN: 0001-4966. DOI: 10.1121/1.4941655. eprint: https://pubs.aip.org/asa/jasa/article-pdf/139/2/703/15315210/703_1_online.pdf. URL: https://doi.org/10.1121/1.4941655 (cit. alle pp. 23, 24).

[49] Sam Jelfs, John Culling e Mathieu Lavandier. «Revision and validation of a binaural model for speech intelligibility in noise». In: *Hearing research* 275 (dic. 2010), pp. 96–104. DOI: 10.1016/j.heares.2010.12.005 (cit. a p. 23).

[50] Jacques A. Grange e John F. Culling. «Head orientation benefit to speech intelligibility in noise for cochlear implant users and in realistic listening conditions». In: *The Journal of the Acoustical Society of America* 140.6 (dic. 2016), pp. 4061–4072. ISSN: 0001-4966. DOI: 10.1121/1.4968515. URL: https://doi.org/10.1121/1.4968515 (cit. a p. 23).

[51] Sterling W. Sheffield, Harley J. Wheeler, Douglas S. Brungart e Joshua G. W. Bernstein. «The Effect of Sound Localization on Auditory-Only and Audiovisual Speech Recognition in a Simulated Multitalker Environment». In: *Trends in Hearing* 27 (2023). PMID: 37415497, p. 23312165231186040. DOI: 10.1177/23312165231186040. URL: https://doi.org/10.1177/23312165231186040 (cit. a p. 24).

[52] Sterling W. Sheffield, Griffin D. Romigh, Patrick M. Zurek, Joshua G. W. Bernstein e Douglas S. Brungart. «A method for degrading sound localization while preserving binaural advantages for speech reception in noise». In: *The Journal of the Acoustical Society of America* 145.2 (feb. 2019), pp. 1129–1142. ISSN: 0001-4966. DOI: 10.1121/1.5090494. URL: https://doi.org/10.1121/1.5090494 (cit. a p. 24).

[53] Alastair H. Moore, Tim Green, Mike Brookes e Patrick A. Naylor. «Measuring audio-visual speech intelligibility under dynamic listening conditions using virtual reality». In: *Audio Engineering Society Conference: 2022 AES International Conference on Audio for Virtual and Augmented Reality*. Ago. 2022. URL: https://www.aes.org/e-lib/browse.cfm?elib=21876 (cit. a p. 26).

[54] Karen S. Helfer e Richard L. Freyman. «The role of visual speech cues in reducing energetic and informational masking». In: *The Journal of the Acoustical Society of America* 117.2 (gen. 2005), pp. 842–849. ISSN: 0001-4966. DOI: 10.1121/1.1836832. URL: https://doi.org/10.1121/1.1836832 (cit. a p. 26).

[55] Giso Grimm, Angelika Kothe e Volker Hohmann. «EFFECT OF HEAD MOTION ANIMATION ON IMMERSION AND CONVERSATIONAL BENEFIT IN TURN-TAKING CONVERSATIONS VIA TELEPRESENCE IN AUDIOVISUAL VIRTUAL ENVIRONMENTS». In: *Forum Acusticum*. 2023, pp. 433–435 (cit. a p. 26).

[56] Giso Grimm, Joanna Luberadzka e Volker Hohmann. «A Toolbox for Rendering Virtual Acoustic Environments in the Context of Audiology». In: *Acta Acustica united with Acustica* 105.3 (2019), pp. 566–578. ISSN: 1610-1928. DOI: doi:10.3813/AAA.919337 (cit. a p. 26).

[57] Angela Guastamacchia, Giuseppina Emma Puglisi, Andrea Albera, Louena Shtrepi, Fabrizio Riente, Masoero Marco Carlo e Arianna Astolfi. «AUDIOVISUAL RECORDING AND REPRODUCTION OF ECOLOGICAL ACOUSTICAL SCENES FOR HEARING RESEARCH: A CASE STUDY WITH HIGH REVERBERATION». In: *Forum Acusticum*. 2023 (cit. alle pp. 26, 31, 57).

[58] Cockos Incorporated. *Reaper*. Last accessed 2024-04-13. URL: https://www.reaper.fm/ (cit. a p. 27).

[59] Giuseppina Puglisi et al. «Evaluation of Italian Simplified Matrix Test for Speech-Recognition Measurements in Noise». In: *Audiology Research* 11 (feb. 2021), pp. 73–88. DOI: 10.3390/audiolres11010009 (cit. a p. 27).

[60] K R Prajwal, Rudrabha Mukhopadhyay, Vinay P. Namboodiri e C.V. Jawahar. «A Lip Sync Expert Is All You Need for Speech to Lip Generation In the Wild». In: *Proceedings of the 28th ACM International Conference on Multimedia*. MM '20. Seattle, WA, USA: Association for Computing Machinery, 2020, pp. 484–492. ISBN: 9781450379885. DOI: 10.1145/3394171.3413532. URL: https://doi.org/10.1145/3394171.3413532 (cit. alle pp. 28–31).

[61] Afouras Triantafyllos, Chung Joon Son, Senior Andrew, Vinyals Oriol e Zisserman Andrew. *The Oxford-BBC Lip Reading Sentences 2 (LRS2) Dataset*. Last accessed 2024-05-31. 2018. URL: https://www.robots.ox.ac.uk/~vgg/data/lip_reading/lrs2.html (cit. a p. 28).

[62] Prajwal K R, Mukhopadhyay Rudrabha, Namboodiri Vinay P. e C V Jawahar. *A Lip Sync Expert Is All You Need for Speech to Lip Generation In the Wild*. Last accessed 2024-05-31. 2020. URL: http://cvit.iiit.ac.in/research/projects/cvit-projects/a-lip-sync-expert-is-all-you-need-for-speech-to-lip-generation-in-the-wild (cit. a p. 29).

[63] Prajwal K R, Mukhopadhyay Rudrabha, Namboodiri Vinay P. e C V Jawahar. *A Lip Sync Expert Is All You Need for Speech to Lip Generation In the Wild - Source Code*. Last accessed 2024-05-31. 2020. URL: https://github.com/Rudrabha/Wav2Lip (cit. a p. 29).

[64] synchronicity labs inc. *SyncLab web site*. Last accessed 2024-05-31. URL: https://synclabs.so/ (cit. a p. 29).

[65] Yuanxun Lu, Jinxiang Chai e Xun Cao. «Live speech portraits: real-time photorealistic talking-head animation». In: *ACM Trans. Graph.* 40.6 (dic. 2021). ISSN: 0730-0301. DOI: 10.1145/3478513.3480484. URL: https://doi.org/10.1145/3478513.3480484 (cit. alle pp. 29, 31).

[66] Rosana Ardila et al. *Common Voice: A Massively-Multilingual Speech Corpus*. 2020. arXiv: 1912.06670 [cs.CL] (cit. a p. 29).

[67] Yuanxun Lu, Jinxiang Chai e Xun Cao. *Live speech portraits: real-time photorealistic talking-head animation*. Last accessed 2024-06-01. 2021. URL: https://yuanxunlu.github.io/projects/LiveSpeechPortraits/ (cit. a p. 30).

[68] NVIDIA Corporation. *NVIDIA Audio2Face*. Last accessed 2024-06-04. 2023. URL: https://docs.omniverse.nvidia.com/audio2face/latest/index.html (cit. a p. 30).

[69] Tero Karras, Timo Aila, Samuli Laine, Antti Herva e Jaakko Lehtinen. «Audio-Driven Facial Animation by Joint End-to-End Learning of Pose and Emotion». In: *ACM Transactions on Graphics (TOG)*. Vol. 36. 4. ACM, 2017, pp. 1–12. URL: https://research.nvidia.com/sites/default/files/publications/karras2017siggraph-paper_0.pdf (cit. a p. 30).

[70] Ignazio Ligani. «Recording of ecological audiovisual scenes for the Audio Space Lab of the Politecnico di Torino». Master thesis. Apr. 2023. URL: http://webthesis.biblio.polito.it/26920/ (cit. alle pp. 33, 35, 59).

[71] Guido van Rossum. *Python Programming Language*. Accessed: 14 luglio 2024. Python Software Foundation, 1995. URL: https://www.python.org/ (cit. a p. 34).

[72] Trimble Inc. *SketchUp*. Last accessed 2024-04-13. URL: https://app.sketchup.com (cit. a p. 35).

[73] OpenAI. *Whisper*. Last accessed 2024-04-13. URL: https://github.com/openai/whisper (cit. a p. 37).

[74] Python Software Foundation. *Tkinter Reference: a GUI for Python*. Last accessed 2024-04-13. Python Software Foundation. URL: https://docs.python.org/3/library/tkinter.html (cit. a p. 37).

[75] McFee, Brian and Raffel, Colin and Liang, Dawen and Ellis, Daniel PW and McVicar, Matt and Battenberg, Eric and Nieto, Oriol and Giannakopoulos, Thanos and Zabilansky, Justin and Zanin, Marcello Magno. *Librosa: Audio and music signal analysis in Python*. 2022. DOI: https://doi.org/10.5281/zenodo.591533. URL: https://librosa.org/doc/latest/index.html (cit. a p. 37).

[76] Blackmagic Design. *DaVinci Resolve Studio 18*. https://www.blackmagicdesign.com/products/davinciresolve/. Accessed: 14 luglio 2024. 2022 (cit. a p. 39).

[77] FFmpeg Team. *FFmpeg*. Accessed: 14 luglio 2024. 2000. URL: https://ffmpeg.org/ (cit. a p. 42).

[78] Microsoft. *PowerShell*. Accessed: 14 luglio 2024. 2006. URL: https://docs.microsoft.com/en-us/powershell/ (cit. a p. 42).

[79] Google Inc. *Spatial Media Metadata Injector*. Accessed: 2024-06-08. 2016. URL: https://github.com/google/spatial-media/releases/tag/v2.0 (cit. a p. 46).

[80] Blender Foundation. *Blender 4.1*. Accessed: 14 luglio 2024. 2023. URL: https://www.blender.org/ (cit. a p. 48).

[81] Epic Games. *MetaHuman Creator*. Accessed: 2024-06-04. 2021. URL: https://www.unrealengine.com/en-US/metahuman (cit. a p. 49).

[82] Avaturn. *Avaturn: Realistic 3D Avatar Creator*. Accessed: 2024-06-04. 2023. URL: https://avaturn.me/ (cit. a p. 49).

[83] Inc. Red Hat. *Ansible*. Accessed: 2024-06-08. 2012. URL: https://www.ansible.com/ (cit. a p. 53).

[84] Angela Guastamacchia, Andrea Galletto, Fabrizio Riente, Louena Shtrepi, Giuseppina Puglisi, Andrea Albera, Franco Pellerey e Arianna Astolfi. «Impact of contextual and lip-sync-related visual cues on speech intelligibility through immersive audio-visual scene recordingsin a reverberant conference room». In: 2024 (cit. alle pp. 57, 59, 60, 69).

[85] UNI Ente Italiano di Normazione. *UNI 11532-2:2020*. Accessed: 2024-06-20. 2020. URL: `https://store.uni.com/uni-11532-2-2020` (cit. a p. 57).

[86] Angela Guastamacchia e Andrea Albera. «IMPATTO DELL'IMMERSIONE AUDIOVISIVA SULL'INTELLEGIBILITÀ DEL PARLATO». In: *50° Convegno Nazionale AIA*. Convegno tenuto dal 29 al 31 maggio 2024, Contributo in Atti di Convegno (Proceeding). Taormina, mag. 2024 (cit. a p. 69).