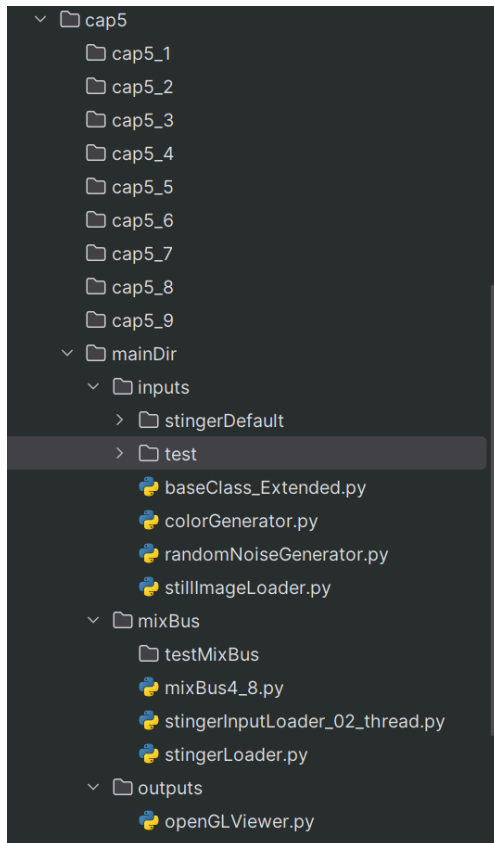


openPyVISION

A.Michelassi 2024 draft v0.5

5.0 - More Input



Nel capitolo precedente, abbiamo discusso e scritto molto codice per costruire le basi di un video mixer. Nel capitolo 5, ci concentriamo sul consolidamento e sull'organizzazione di questo codice in una struttura più definitiva, integrando nuove funzionalità per migliorare la gestione dei dati e delle risorse del programma.

La struttura del codice che abbiamo costruito è organizzata in diverse cartelle, simili a quelle del progetto **OpenPyVision Mixer**. All'interno della cartella principale, troverete cartelle dedicate agli input, che contengono i generatori che abbiamo utilizzato finora, il **synchObject** e la classe base estesa. C'è una cartella dedicata al **mixBus**, che include anche la gestione degli stinger, ovvero le animazioni di transizione utilizzate per passare da una scena all'altra. La cartella di output contiene la classe per

la visualizzazione tramite **OpenGL**, che rappresenta il risultato finale delle operazioni di miscelazione e manipolazione video.

Una delle novità introdotte in questo capitolo è la classe **DataManager**, che si occupa di gestire le posizioni di default delle cartelle contenenti risorse come gli stinger e le immagini fisse. Questa classe è progettata per caricare e salvare automaticamente queste informazioni, garantendo che il programma non vada in crash se le cartelle non vengono trovate. La **DataManager** utilizza le funzionalità costruite nel capitolo 4 per caricare gli stinger e le immagini fisse, passando poi questi elementi al **mixBus** per l'elaborazione.

Il funzionamento del **DataManager** è abbastanza semplice: all'avvio, la classe tenta di caricare i percorsi salvati da un file JSON. Se il file non esiste o è corrotto, viene creato un nuovo file con i valori di default. Il DataManager gestisce anche il caricamento degli stinger tramite un thread separato, mostrando un widget di progresso durante il caricamento per informare l'utente dello stato dell'operazione. Se qualcosa va storto, ad esempio se un percorso non viene trovato, il DataManager segnala l'errore, mantenendo il programma stabile e informando l'utente.