Práctica Final Inteligencia Ambiental

Javier Pita González-Campos (100348832)

30 de diciembre de 2022

Aplicación

La aplicación desarrollada es un predictor de sonidos para ayudar a personas con problemas de audición. La interfaz desarrollada, además de permitir su uso a través de una serie de botones clicables con interfaz gráfica, permite a su usuario un manejo completo* a través de un lenguaje de signos muy sencillo.

Esta herramienta es considerada una modelo de inteligencia ambiental debido a que cumple con los cuatro pasos definidos por la teoría clásica. **Detecta**, a través de la cámara, los sonidos o el ratón, los elementos de su entorno. **Razona**, a través de tres modelos diferentes de inteligencia ambiental que el usuario: se encuentra frente a la cámara, realiza gestos para usar la interfaz o se escuchan y clasifican sonidos. **Actúa**, en base a los gestos o sonidos clasificados dando feedback al usuario que **interactúa** continuando con la interfaz y haciendo ad-hoc un modelo para sus necesidades.

En la Figura 1 se puede observar la interfaz completa de la aplicación desarrollada de inteligencia ambiental. Esta se encuentra dividida en dos secciones claramente diferenciadas. La primera sección, la superior, permite al usuario hacer un seguimiento del uso de la herramienta a través de signos de la interfaz, mientras que la inferior es la parte de reconocimiento de sonidos.

Interfaz de signos

En la interfaz de signos podemos apreciar al usuario por pantalla donde este puede ver en tiempo real qué gestos está haciendo y en la parte inferior el conjunto de signos codificados para ser reconocidos por la herramienta. El conjunto de signos que la aplicación está diseñada para interpretar son: Index Finger, Middle Finger, Ring Finger y Pinky Finger. Los gestos están nombrados por el dedo que hay que unir con el dedo gordo (Thumb Finger) para ser reconocido como gesto. La figura 2 muestra un ejemplo del gesto Middle Finger siendo reconocido por la interfaz. Como podemos apreciar el elemento correspondiente con el nombre se iluminará para permitir al usuario entender que ha sido reconocido y por tanto, dependiendo de si en el estado actual el botón correspondiente clicado. Debajo de cada

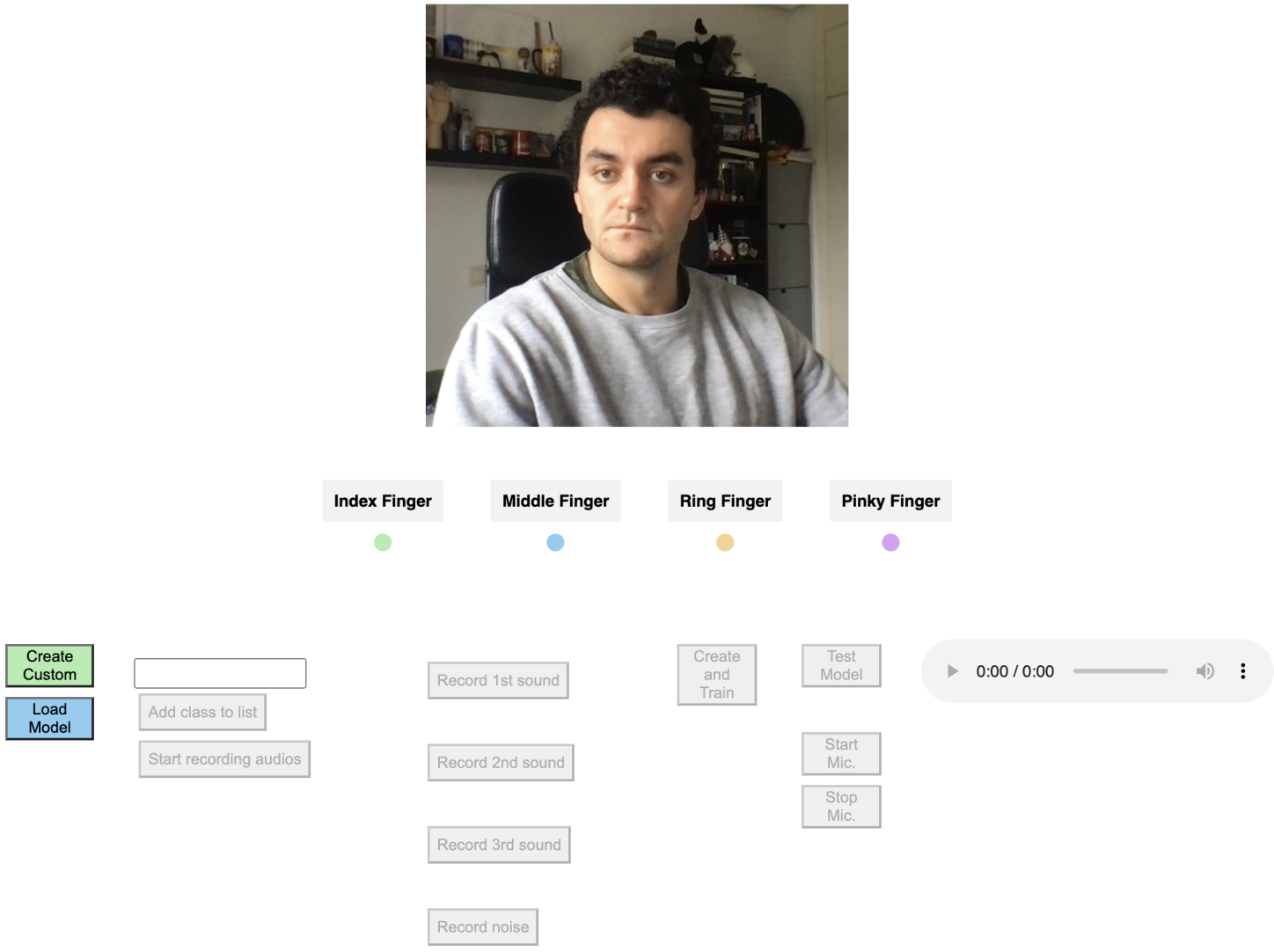

Figura 1: Interfaz completa

posible gesto podemos apreciar un código de colores para los botones. Estos colores se han establecido para poder utilizar el conjunto de botones de la interfaz de sonidos de manera sencilla, coloreando en cada momento el botón correspondiente al gesto que lo acciona.

Para el reconocimiento de signos se ha escogido la opción de desarrollo ad-hoc que recibe los puntos del modelo Hand Pose, comprueba en función de la mano recibida (Left o Right) que la palma se encuentra apuntando hacia la cámara, calculando las distancias entre los cuatro puntos correspondientes de los nudillos, y computa las distancias de los denominados Tips con respecto al Tip del Thumb. Si esta distancia es menor que un umbral durante 10 clasificaciones seguidas es considerado como un gesto adrede y por tanto interpretado como un click. Importante puntualizar que las distancias se comprueban de dedo más cercano al Thumb al más alejado sin comprobaciones sucesivas en caso de cumplir con la condición- Esto implica que si más de un dedo se encuentra cerca del Thumb va a ser interpretado como el más cercano.

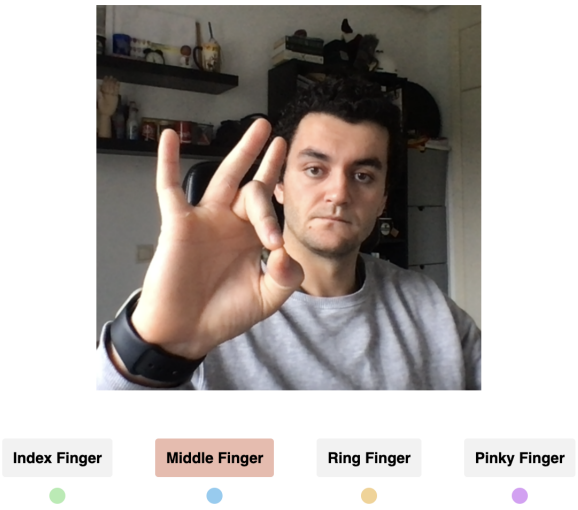

Figura 2: Gesto Middle Finger

Interfaz de sonidos

La interfaz de sonido permite al usuario utilizar un modelo pre-entrenado de reconocimiento de sonidos, explicado más adelante en esta memoria, o entrenar uno ad-hoc con clases a definir por el usuario. Para escoger la primera opción es necesario clickar (por interfaz grafica o gesto) el botón Load Model y para la segunda el botón Create Custom. Si volvemos a fijarnos en la Figura 1 el botón Create Model se presionaría con el gesto Index Finger mientras que el Load Model con el Middle Finger. Un caso donde se pueden usar todos los gestos al mismo tiempo es en el grabado de audios, ver Figura 3.

Para el caso de creación ad-hoc el sistema nos permitirá declarar el nombre de las tres clases a introducir (la cuarta se reservará para el ruido de fondo) y grabar una serie de audios para poder realizar el entrenamiento. Existen una serie de clases definidas que se

pueden usar y cuyo feedback en caso de ser clasificados tiene un feedback visual predefinido (baby, cat, dog, ridge, hood, microwave, noise y vacuum). En la Figura 3 podemos ver que en la creación de las clases en caso de pertenecer al grupo predefinido aparecerá un icono definiendo la clase.

Una vez el mínimo de audios por clase ha sido grabado (la interfaz es muy intuitiva) se habilitará el botón Create and Train para crear el clasificador Fully Connected y entrenar el modelo con el conjunto de audios creado.

Tanto el modelo creado como el cargado se pueden testar, como en el código proporcionado del Tutorial 4, con Test Model para usar audios grabados en su creación y reservados para testing o a través de el audio recogido en tiempo real por el micrófono.

Cuando se produce una clasificación existen dos feedbacks diferenciados que permiten saber al usuario de este proceso, uno visual y otro 'sensorial'. El primero, el sensorial, es un destello visual del fondo que sustituye lo que podría ser una vibración y estaría pensado para el caso de que la app se llevara activa en un móvil o wearable y mandará una señal al usuario de que se ha escuchado un sonido, similar a una notificación. El segundo feedback, el visual, es la aparición de un icono con animación que indica la clase clasificada (en caso de estar entre las predeterminadas) o una imagen por defecto en caso de no estarlo. Este feedback está pensado para cuando el usuario está mirando la pantalla y podría ser emitido cuando el usuario coge el móvil o levanta el brazo para mirar su reloj (despues de haber sentido la vibración), completando así un feedback conjunto.

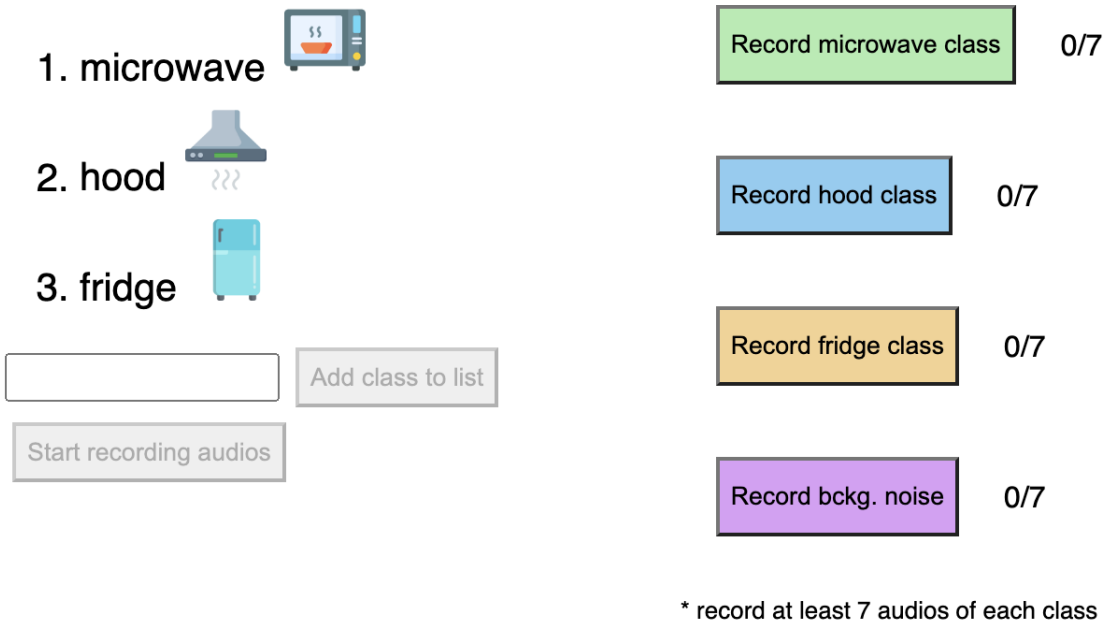

Figura 3: Interfaz de clases

Modelo preentrenado

El modelo pre-entrenado se ha creado con sonidos que normalmente se pueden escuchar en una cocina normal y que son parte del día a día de usuarios sin problemas auditivos pero que implican una dificultad para personas con este tipo de problemas. Este modelo incluye las clases:

- **Microwave:** Sonido final del microondas que permite saber al usuario que el temporizador ha llegado a su fin.
- **Hood:** Campana extractora de la cocina que avisa en caso de estar encendida. Este sonido por sí solo no es tan interesante ya que avisaría constantemente al usuario en caso de estar cocinando,en cambio combinado con un sistema de ADL (Activities of daily living) puede ser útil para, en caso de haber terminado de cocinar (clasificado por el ADL), avisar al usuario de no haber apagado la campana.
- **Fridge** Sonido que emite la nevera cuando se ha quedado abierta un periodo largo de tiempo.

Después de realizar distintas pruebas se ha podido setear el valor de los audios necesarios para una correcta clasificación en cinco, siendo cuatro de train y el último de test. Por este motivo además el valor por defecto mínimo pedido para entrenar el modelo es este. De todas maneras, el código está preparado para recibir cuantos el usuario crea conveniente, separando siempre un 10 % para testing y el resto para entrenar, incluso diferente número de audios de cada clase. En este último caso se computará el 10 % con respecto a la clase con menor número de audios con el fin de incluir el mismo número en el entrenamiento.

Trabajo futuro

Uno de las ideas de desarrollo que hubiera permitido que la interfaz pudiera ser completamente controlada por signos hubiera sido implementar el abecedario completo en LSE, de esta manera para hacer uso de la interfaz de sonidos completa no hubiera sido necesario el uso del teclado. La arquitectura necesaria no hubiera estado muy lejos del scope de esta práctica debido a que ya se están reconociendo signos con las manos, en cambio las horas de dedicación para la correcta clasificación de cada signo hacen de este un desarrollo muy costoso en términos de tiempo.

Para desarrollar esta funcionalidad, y reducir un poco los tiempos, se hubiera realizado una modificación a la arquitectura actual. Para el reconocimiento de los gestos se ha escogido la opción de desarrollo de código ad-hoc y elección de gestos sencillos de reconocer para facilitar la implementación. En el caso del abecedario se hubiera escogido un acercamiento diferente en el cual los puntos devueltos por la interfaz del modelo Hand Pose hubieran acabado en un modelo de clasificación del tipo KNN o con una capa Fully Connected final como el caso de la clasificación de sonidos.

El mayor problema para este desarrollo sería la creación de un dataset con todas las letras del abecedario, con cada una de las manos, en diferentes posiciones para poder hacer

un modelo eficaz. Además habría que repensar los signos escogidos para manejar la interfaz porque alguno de ellos coincide con una de las letras del abecedario (Index Finger y letra O) y otros son tan similares que el modelo tendría problemas para diferenciarlos (letras S,F y T) que además de ser muy similares entre ellos son muy similares a Index Finger.

Posibles Mejoras

Los gestos pueden ser en ocasiones algo complejos de detectar. La elección de código ad-hoc para su clasificación facilita su implementación pero al haber que qué tomar una serie de decisiones de error y tiempo para ser considerado como un click su implementación no es perfecta. La herramienta diseñada tiene un pequeño fallo ya que funciona muy bien en caso de determinadas posturas de la mano mientras que otras no las reconoce tan fácilmente. Para evitar la clasificación de un gesto erróneo se ha reducido la distancia entre dedos a una distancia muy pequeña en base a prueba y error. Por este motivo, y tal y como está diseñado el detector de puntos de la mano en ocasiones, aunque el usuario esté con los dedos juntos, no se registra como click. Un truco para su uso es, independientemente del gesto a realizar si se enseñan las uñas de ambos dedos unidos el sistema nunca falla, en cambio si no es así puede constarle su reconocimiento al no cumplir el umbral determinado. Podemos ver esto en la Figura 4.

Otro de las mejoras a implementar es la elección de gestos. Se han escogido gestos fáciles de detectar, pero en ocasiones complejos de realizar. El gesto Index Finger es un gesto muy natural y obligatorio en la práctica, en cambio a medida que avanzamos en la mano se complica cada vez más. Esto además de ser un inconveniente para todos los usuarios puede ser un problema de uso para alguno otro. Hay ciertas lesiones en la mano que impiden tener esa flexibilidad con el dedo gordo impidiendo llegar a los dedos más alejados de este. Esto no es un caso extremo ya que yo mismo durante más de un año y después de una operación y rehabilitación pude recuperar la movilidad tras una lesión muy común.

Finalmente añadir que el modelo presentado cuando recibe audios diurectamente del micrófono baja considerablemente el umbral de reconocimiento de los sonidos. Si para los audios de test se ha estipulado en 0.95 el limite inferior para considerar una clase como clasificada en el caso de micro abierto se ha bajado a 0.8 y aún asi falla en ocasiones al reconocer ya que lo predice por debajo de 0.6 en los peores casos. Faltaría estudiar un poco más ese umbral para delimitarlo sin que supongan errores de clasificación.

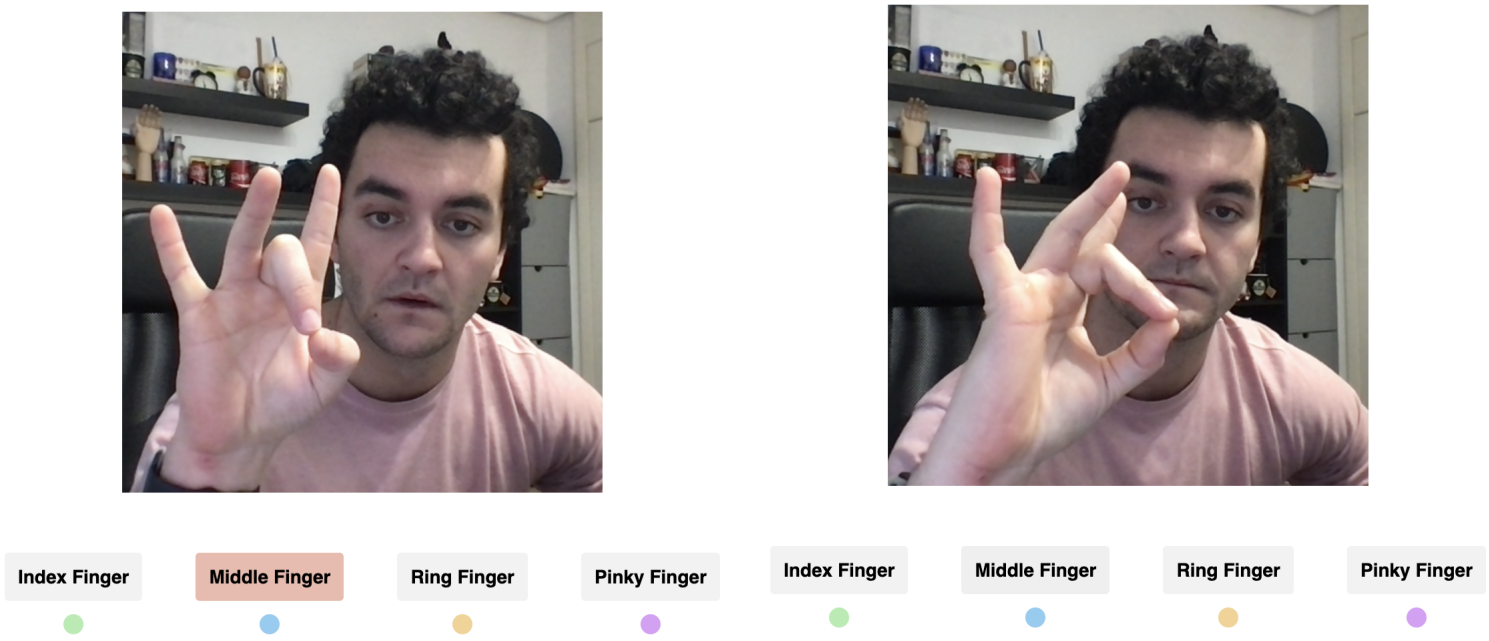

Figura 4: Problema reconociemiento de gestos