Universidad de Alcalá
Escuela Politécnica Superior

Grado en Ingeniería en Tecnologías de la Telecomunicación

Trabajo Fin de Grado

Estimación de error de localización 3D de personas a partir de imágenes 2D

Autor: Pablo Sanz Miguel

Tutora: Marta Marrón Romera

Cotutor: Frank Sanabria Macías

2022

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería en Tecnologías de la Telecomunicación

Trabajo Fin de Grado

Estimación de error de localización 3D de personas a partir de imágenes 2D

Autor: Pablo Sanz Miguel

Tutora: Marta Marrón Romera

Cotutor: Frank Sanabria Macías

Tribunal:

Presidente: Miguel Ángel García Garrido

Vocal 1º: Alfredo Gardel Vicente

Vocal 2º: Marta Marrón Romera

Fecha de depósito: 20 de junio de 2022

A mi familia, amigos y compañeros...

"Empieza haciendo lo necesario, luego haz lo posible y de pronto empezarás a hacer lo imposible."
Francisco de Asís

Agradecimientos

A todos los que la presente vieren y entendieren.

Inicio de las Leyes Orgánicas. Juan Carlos I

A mis padres, Jesus y Tere, y a mis hermanos, Daniel y Teresa, por ser la familia que siempre he querido tener. Gracias a vosotros soy quien soy hoy en día y con vuestro apoyo y ayuda he podido alcanzar las metas que me he propuesto. Gracias por estar a mi lado en los buenos momentos pero, sobretodo, por levantarme en los malos.

A mis amigos de Guadalajara con los que paso los días. Porque sigamos apoyándonos unos a otros y sigamos celebrando los logros conseguidos de manera individual como si fuesen de todos.

A mi amigo Adrian, porque sólo tu sabes lo que hemos trabajado estos años hasta que hemos conseguido nuestro objetivo. Porque, aunque aguantarte ha sido difícil, no se me ocurre mejor compañero para haberlo logrado.

A todos los profesores con los que he coincidido en esta etapa, especialmente a Marta, Frank y Javi por el trato que me han dado y ayudarme a resolver todas las dudas que me han surgido durante la realización del presente trabajo.

Resumen

El siguiente Trabajo de Fin de Grado (TFG) se enmarca dentro de un proyecto más amplio y con mayor recorrido que trata de mejorar la fiabilidad en el seguimiento de múltiples locutores mediante información audiovisual [1] [2].

El objetivo del presente TFG se centra en el cálculo del error cometido durante el proceso de estimación de la posición de la boca de una o varias personas en un espacio 3D a partir de su imagen en 2D.

En este TFG se usa un trabajo del estado del arte [3] para la detección de marcas faciales en imágenes para, mediante alguno de los métodos matemáticos conocidos, llevar esa información redundante al espacio 3D, ubicando allí la posición de la boca de las personas, cuyas marcas faciales se han detectado en la imagen.

Este proceso matemático recursivo tiene, en general, un error que se puede estimar y minimizar, con objeto de mejorar la fiabilidad de la tarea de mapeo 2D-3D. En este trabajo se calcula y analiza este error tanto en coordenadas 2D, expresado en pixeles, como en coordenadas 3D, expresado en milímetros.

Con ayuda de la herramienta Matlab se va a crear un sistema capaz de realizar el análisis de una secuencia de entrada imagen a imagen, obteniendo la posición de la boca de los locutores que aparecen en dicha imagen en coordenadas 3D. Las secuencias con las que se trabajan pertenecen a dos bases de datos de uso extendido en el área de fusión audio-visual para el seguimiento de múltiples personas (AV6 [4] y CAVD [5]).

Para poder realizar el sistema, primero es necesario el cálculo de la posición 2D de la boca de las personas, estas coordenadas se obtienen gracias a la información de marcadores faciales en 2D obtenida mediante una propuesta “Deep Learning” de detección de caras en imágenes [6].

Una vez conocidos los marcadores faciales en 2D que componen el rostro de las personas que aparecen, se emplea el método Posit, Pose from Orthography and Scaling with ITerations (POSIT). Se obtiene así la posición de los puntos faciales en 3D correspondientes al rostro de las personas.

Por último, para calcular el error cometido durante el análisis, se compara la localización de la boca obtenida, en coordenadas 2D y 3D, con la posición exacta de la boca de la persona, Ground Truth (GT), proporcionada por la base de datos.

Palabras clave: Seguimiento de locutores, información audiovisual, marcadores faciales, 2D, 3D.

Abstract

The following final thesis is part of a larger and more extensive project that aims to improve the reliability of tracking multiple speakers using audiovisual information.

The objective of this work focuses on the calculation of the error made during the process of estimating the position of the mouth of one or several people in a 3D space from their 2D image.

In this work a state of the art work for the detection of facial marks in images is used to, by means of one of the known mathematical methods, take that redundant information to the 3D space, locating there the position of the mouth of the persons, whose facial marks have been detected in the image.

This recursive mathematical process has, in general, an error that can be estimated and minimized, in order to improve the reliability of the 2D-3D mapping task. In this work this error is calculated and analyzed both in 2D coordinates, expressed in pixels, and in 3D coordinates, expressed in millimeters.

With the help of the Matlab tool, a system capable of analyzing an input sequence frame by frame will be created, obtaining the position of the mouths of the speakers appearing in that frame in 3D coordinates. The sequences we are working with belong to two widely used databases in the area of audio-visual fusion for tracking multiple people (AV16 and CAV3D).

In order to realize the system, it is first necessary to calculate the 2D position of the mouths of people, these coordinates are obtained thanks to the information of 2D facial markers obtained through a "Deep Learning.[a]pproach to face detection in images.

Once the 2D facial markers that make up the face of the people shown are known, using the Posit method, the position of the 3D facial points corresponding to the face of the people is obtained.

Finally, to calculate the error made during the analysis, the location of the mouth obtained, in 2D and 3D coordinates, is compared with the exact position of the person's mouth provided by the database.

Keywords: Speaker tracking, audiovisual information, facial markers, 2D, 3D.

Índice general

Abstract **ix**

Índice general **xi**

Índice de figuras **xv**

Índice de tablas **xvii**

Índice de algoritmos **xix**

Lista de acrónimos **xix**

1 Introducción **1**

1.1 Contexto del trabajo 1

1.2 Objetivo 2

1.3 Organización de la memoria 2

2 Estudio teórico **5**

2.1 Introducción 5

2.2 Estimación de la "pose" 5

2.3 Método POSIT 7

2.4 Evaluación 9

2.4.1 Bases de datos 9

2.4.2 Métricas 12

3 Desarrollo **15**

3.1 Introducción 15

3.2 Procedimiento general 15

3.3 Extracción de puntos característicos en 2D 17

3.3.1 Secuencia de análisis 17

3.3.2 Obtención de puntos 2D 18

3.4 Reproyección 3D 19

3.4.1 Escalado de los 98 puntos 3D . . . 19
3.4.2 Selección del modelo . . . 20
3.4.3 Método POSIT . . . 22
3.4.4 Cambio del sistema de referencia . . . 23
3.5 Cálculo del error de "pose" . . . 24
3.5.1 Cálculo del error 2D . . . 24
3.5.2 Cálculo del error 3D . . . 24

4 Resultados **27**
4.1 Introducción . . . 27
4.2 AV16_3 . . . 28
4.2.1 Modelo 1 . . . 28
4.2.2 Modelo 2 . . . 29
4.2.3 Modelo 3 . . . 30
4.2.4 Modelo 4 . . . 31
4.3 CAV3D . . . 32
4.3.1 Modelo 1 . . . 32
4.3.2 Modelo 2 . . . 33
4.3.3 Modelo 3 . . . 34
4.3.4 Modelo 4 . . . 36

5 Conclusiones y líneas futuras **41**
5.1 Introducción . . . 41
5.2 Conclusiones . . . 41
5.3 Líneas futuras . . . 42

Bibliografía **45**

Apéndice A Recursos necesarios **47**
A.1 Introducción . . . 47
A.2 Hardware . . . 47
A.3 Software . . . 47

Apéndice B Presupuesto **49**
B.1 Coste recursos Software . . . 49
B.2 Coste recursos Hardware . . . 49
B.3 Coste recursos humanos . . . 49
B.4 Coste total de los recursos . . . 50
B.5 Costes de ejecución por contrato . . . 50

B.6 Tasas facultativas 50
B.7 Presupuesto total 50

Índice de figuras

1.1 Esquema general simplificado 2

2.1 Estimación de la "pose" de una persona detectando los puntos clave dentro el cuerpo humano 6
2.2 Rostro de una persona representado mediante 98 puntos faciales 7
2.3 Figura obtenida de [7]. Esquema de proyección en perspectiva y proyección ortográfica de un punto M_i 8
2.4 Habitación donde se graban y extrae la información en las secuencias de la base de datos CAV3D [5] 10
2.5 (a) Se muestra una imagen de una secuencia CAV3D_SOT (b) En esta imagen puede observarse un ejemplo de una secuencia CAV3D_SOT2 (c) Imagen sacada de una secuencia CAV3D_MOT 10
2.6 Habitación donde se graban y extrae la información en las secuencias de la base de datos AV16_3 [4] 11
2.7 (a) Imagen de una secuencia AV16_3_SOT (b) Imagen sacada de una secuencia AV16_3_MOT 11

3.1 Esquema del sistema completo 16
3.2 Secuencias de las dos bases de datos con los imágenes que se deben analizar 18
3.3 imagen de una secuencia en la que se pinta el **bbox** en azul y los **landmarks** en rojo . . 19
3.4 Visión de frente y perfil de los 98 puntos faciales en coordenadas 3D 20
3.5 En negro se muestra el punto correspondiente a la boca en coordenadas 3D del Ground Truth (GT). En rojo se muestra el punto de la boca en coordenadas 3D calculado por el sistema 25
3.6 Gráfica en la que se muestra el error cometido en cada uno de los imágenes 25

4.1 Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos AV16_3 28
4.2 Frames de la secuencia 25 donde el locutor uno tapa la visión sobre el locutor 2 28
4.3 Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo 29
4.4 Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos AV16_3 29
4.5 Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo 30

4.6 Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos AV16_3 30
4.7 Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo 31
4.8 Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos AV16_3 31
4.9 Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo 31
4.10 Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos CAV3D 32
4.11 a) Frames de la secuencia 6 donde se ve al locutor salir del campo de visión de la cámara b) Frames de la secuencia 6 donde se ve al locutor entrar en el campo de visión de la cámara 33
4.12 Frames de la secuencia 11 donde se ve al locutor de espaldas a la cámara 34
4.13 Frames pertenecientes a la secuencia 26 35
4.14 Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo 35
4.15 Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos CAV3D 36
4.16 Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo 36
4.17 Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos CAV3D 37
4.18 Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo 37
4.19 Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos CAV3D 38
4.20 Frames pertenecientes a la secuencia 12 39
4.21 Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo 39

5.1 Resumen de los resultados obtenidos mediante los cuatro modelos diferentes de análisis para las secuencias pertenecientes a las bases de datos AV16_3 41
5.2 Resumen de los resultados obtenidos mediante los cuatro modelos diferentes de análisis para las secuencias pertenecientes a las bases de datos CAV3D 42

Índice de tablas

B.1 Coste recursos software 49
B.2 Coste recursos hardware 49
B.3 Coste recursos humanos 50
B.4 Coste total de los recursos 50
B.5 Costes de ejecución por contrato 50
B.6 Tasas facultativas 50
B.7 Presupuesto total 51

Índice de algoritmos

3.1 Funcionalidad y módulos del sistema 17
3.2 Selección de la secuencia de entrada 17
3.3 Cálculo de los puntos 2D del rostro 19
3.4 Modelos de análisis 21
3.5 Corrección de la distorsión 21
3.6 Mandíbula 22
3.7 Cálculo de la matriz de proyección 22
3.8 Cálculo de la matriz posición 3D de la boca en las coordenadas de la cámara 23
3.9 Cálculo de posición 3D de la boca en el sistema de coordenadas del mundo 24

Capítulo 1

Introducción

Desocupado lector, sin juramento me podrás creer que quisiera que este libro [...] fuera el más hermoso, el más gallardo y más discreto que pudiera imaginarse.

Miguel de Cervantes, Don Quijote de la Mancha

1.1 Contexto del trabajo

Hoy en día, la visión artificial es una de las principales líneas de investigación sobre la que se está trabajando en la sociedad actual. Podríamos definirla como una parte de la Inteligencia Artificial (IA) que permite a una máquina extraer información de una imagen de la misma manera que lo haría un humano a través de la vista.

Para que el sistema pueda trabajar como si de la vista humana se tratarse, debe ser capaz de procesar, analizar y entender las imágenes que se vayan a utilizar. Esto se ha conseguido en los últimos años debido al gran avance logrado gracias al proceso de aprendizaje iterativo, que proporcionan las redes neuronales mediante las cuales somos capaces de emular en una máquina el modo de procesamiento de la información del cerebro humano.

Dentro de la visión artificial, existen varios campos diferentes como pueden ser, por ejemplo, la restauración de imágenes, el reconocimiento de objetos o la reconstrucción de una escena entre otros. Es importante que para llevar a cabo cualquiera de estas funcionalidades, los sistemas de visión artificial estén formados por una combinación de elementos Software (SW) y Hardware (HW), los cuales tienen la capacidad de capturar y procesar los datos de imágenes y/o vídeos.

Hoy por hoy, una de las funciones más importantes y que más se está trabajando en el mundo de la investigación es la de conocer la localización de una determinada persona en un momento dado. La mayoría de los sistemas de localización cuentan con sensores de audio y vídeo para poder realizar un análisis completo de todo lo que ocurre dentro del entorno que se quiere analizar.

El proyecto se centra en la estimación de la "pose" o "pose estimation". Esta estimación es una técnica de la visión artificial que determina y rastrea la localización de una persona o de un objeto a partir de una imagen o un vídeo determinado, teniendo en cuenta solo los elementos visuales, es decir, toda aquella información o señal que provenga de los sensores auditivos quedará excluida en el análisis.

1.2 Objetivo

El objetivo principal de este trabajo es calcular el error 3D producido en la estimación de la localización de la boca de los locutores que aparecen en secuencias conocidas por la comunidad científica para la detección multisensorial de locutores.

Se empleará para ello una propuesta estándar de la literatura de "face recognition" [3], que ubica la boca únicamente en la imagen (plano 2D) y el método POSIT estándar de visión por computador.

Como se observa en la figura 1.1, el objetivo incluirá el análisis también del error 2D cometido por el algoritmo de "face recognition".

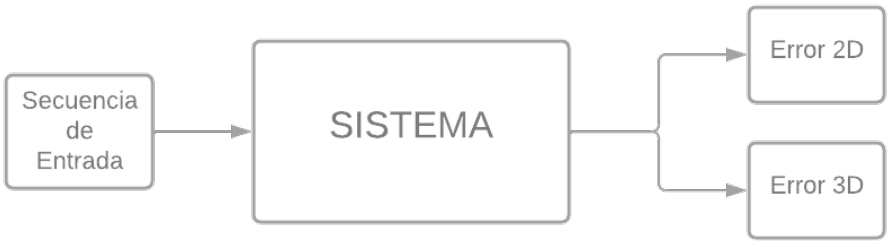

Figura 1.1: Esquema general simplificado

Los resultados se testarán, como se ha comentado, en dos de las bases de datos más conocidas en la literatura de interés, como son CAV3D [5] y AV16_3 [4].

Para poder conseguir el objetivo, se ha diseñado el sistema de la figura 1.1, que recibe como entrada la secuencia que se quiere analizar, y entrega a la salida el error producido en la estimación de la boca. Dentro del sistema pueden diferenciarse dos grandes bloques: el primero, que tiene como función calcular la posición de la boca de los locutores que aparezcan en el mundo 3D mediante POSIT; y el segundo, que calcula el error producido en la estimación 2D y en la 3D. En todo caso, se comparará la posición de la boca obtenida con el Ground Truth (GT) proporcionado por las bases de datos a las que pertenecen las secuencias analizadas, CAV3D y AV16_3.

1.3 Organización de la memoria

Esta memoria se organiza en cinco grandes capítulos, a continuación se explicará de una manera muy resumida el contenido de cada uno de estos.

- **Estudio Teórico**: En este capítulo se explica qué es la estimación de la "pose" y como se obtiene a partir del trabajo usado como referencia base, cómo trabaja teóricamente el método POSIT y cuáles son las principales características de las bases de datos a las que pertenecen las secuencias que se van a analizar.
- **Desarrollo**: Esta es la parte de la memoria donde se explica cómo se ha desarrollado el código fuente para realizar los análisis perseguidos de forma correcta. En primer lugar, se va a explicar como se ha realizado el código que genera los resultados POSIT y después se va a explicar el código que analiza estos resultados obtenidos.
- **Resultados**: En este capítulo se muestran y comentan los diferentes resultados obtenidos.

- **Conclusiones y Líneas Futuras**: En este capítulo se mostrarán las conclusiones obtenidas una vez evaluados los resultados y se comentaran las posibles líneas de trabajo realizables en un futuro.

Además, al final del documento se incluye el listado de referencias incluidas, así como el pliego de conidiciones y presupuesto necesarios para poder realizar el trabajo anteriormente expuesto.

Capítulo 2

Estudio teórico

Y así, del mucho leer y del poco dormir, se le secó el cerebro de manera que vino a perder el juicio.

Miguel de Cervantes Saavedra

2.1 Introducción

En este capítulo se va a explicar el fundamento teórico de la estimación de la "pose" utilizado en este Trabajo de Fin de Grado (TFG), y más concretamente del método empleado para el cálculo de la localización de la boca de la persona en el espacio 3D, método POSIT a partir de la ubicación de ésta en la imagen 2D.

Después, se van a explicar también las características más importantes de las bases de datos con las que se va a trabajar, y validar las propuestas planteadas para ello en el TFG.

Por último, se van a explicar las métricas con las que se va a trabajar durante el trabajo, para completar la validación descrita.

2.2 Estimación de la "pose"

Como se ha mencionado anteriormente, la estimación de la "pose" (o *pose estimation*) se puede definir como una técnica de visión artificial, que detecta la postura de personas u objetos en una imagen o en una secuencia de ellas, en un vídeo determinado. Otra manera de entenderlo, es considerar la estimación de la "pose" como el problema de determinar la posición y orientación de la cámara en función de un objeto determinado. Para que esta estimación tenga sentido, los puntos identificados tienen que ser representativos dentro de la persona o del objeto.

En la figura 2.1 se muestra un ejemplo de como se puede identificar una persona dentro de una imagen identificando los puntos clave de la misma. Es importante que estos puntos sean característicos y muestren una información relevante de la persona a la que pertenecen. De esta manera, si el objetivo es identificar un cuerpo completo, los puntos característicos deberán ser aquellos que identifiquen la cabeza, los hombros, los codos, las manos, las rodillas, los pies ... pero, si el objetivo es identificar el rostro de una persona, los puntos característicos serán los que muestren las cejas, los ojos, la nariz, la boca ...

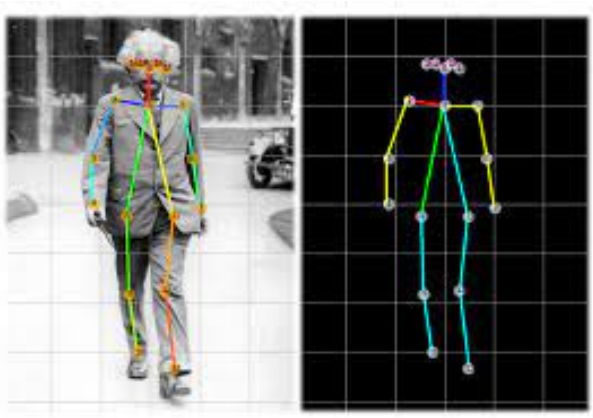

Figura 2.1: Estimación de la "pose" de una persona detectando los puntos clave dentro el cuerpo humano

Otra apreciación a tener en cuenta, es que dentro de la estimación de la "pose" existen dos modos diferentes de detección. La estimación de la "pose" en 2D, en la que se estima la ubicación de los puntos característicos mencionados en coordenadas 2D, y la estimación de la "pose" en 3D, donde se transforman éstos desde una imagen 2D al objeto en 3D añadiendo una proyección que se debe predecir. Gracias a esta nueva proyección se consigue determinar la posición espacial del objeto en 3D. Este segundo modo es el que se va a emplear en el trabajo para poder estimar la posición 3D de la boca de los locutores a partir de una secuencia de imágenes 2D de éstos.

De igual manera, en lo que a este trabajo se refiere, y como se puede ver en la figura 2.2, se va a trabajar con un modelo capaz de extraer 98 puntos característicos ($M_o \dots M_i \dots M_{97}$) dentro del rostro de una persona donde se pueden diferenciar los elementos que componen el rostro como son los ojos, la nariz, las cejas, la boca y la mandíbula.

Para que la obtención de estos 98 puntos sea lo más precisa posible, se ha empleado un modelo denominado Cascade of Recombinator Networks (CRN) donde la información de los rostros que aparecen en las imágenes se va extrayendo a diferentes escalas [3].

Gracias a este modelo, la información extraída de una imagen ha aumentado considerablemente con respecto a la extraída de los modelos con los que se trabajaba anteriormente.

Matemáticamente hablando, por tanto, el objetivo de esta primera parte en el TFG es extraer los puntos característicos 2D de la cara (y por tanto de la "pose" del locutor y su boca), $\mathbf{p}$, mediante el método propuesto en [3], para calcular después estos puntos característicos en sus coordenadas 3D, $\mathbf{P}$, en el espacio de trabajo. La relación que existe entre los dos espacios 2D-3D puede observarse en la ecuación 2.1:

$$\mathbf{p} = K[R|\mathbf{t}]\mathbf{P}, \tag{2.1}$$

donde K hace referencia a los parámetros intrínsecos de la cámara (como es la distancia focal f a los diferentes ejes de coordenadas) R es una matriz de *3 x 3* donde se indica la rotación de la cámara en el espacio 3D, y $\mathbf{t}$ es un vector columna (3,1) donde se muestra el movimiento de traslación de ésta en ese mismo espacio.

Para llevar a cabo esta transformación matemática y obtener los puntos $\mathbf{P}$, 3D, se va a emplear el método POSIT que se explica a continuación.

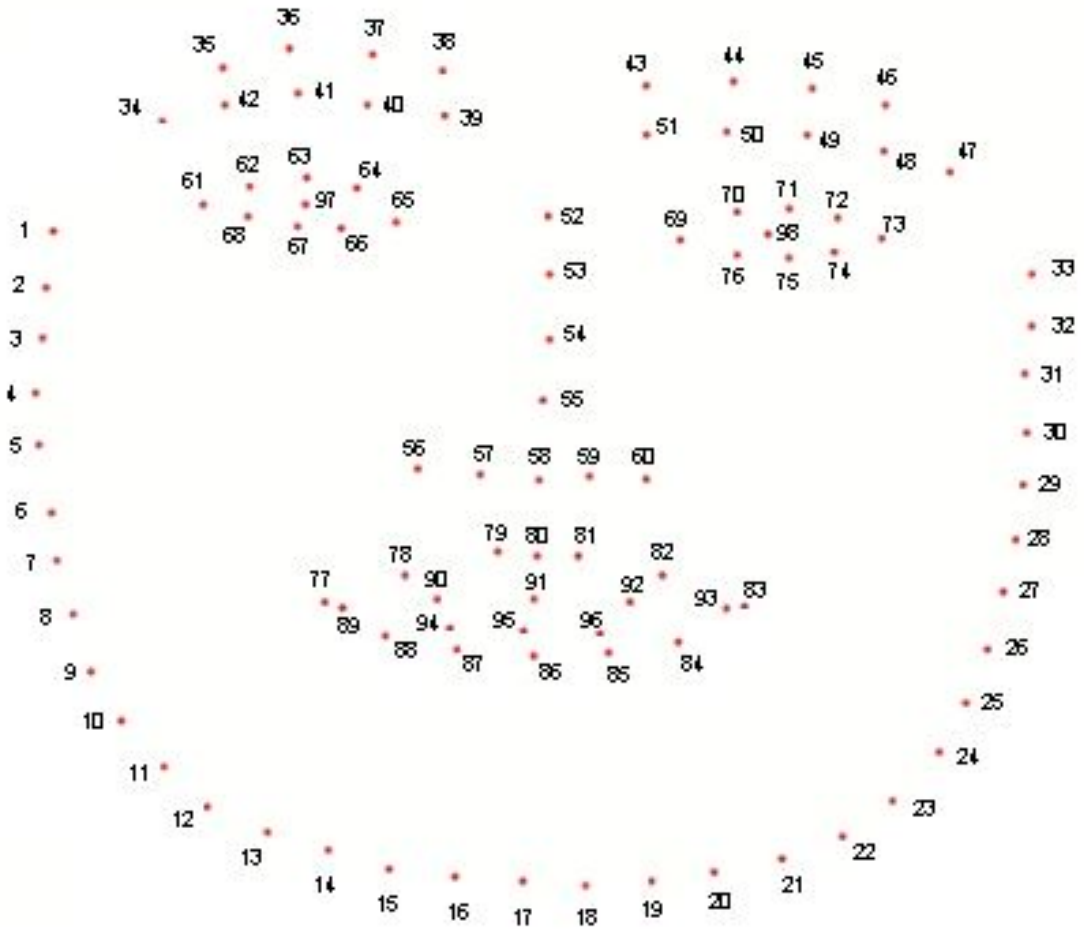

Figura 2.2: Rostro de una persona representado mediante 98 puntos faciales

2.3 Método POSIT

El método POSIT emplea desde hace décadas técnicas de álgebra lineal iterativamente, para resolver matrices de ecuaciones multivariable con múltiples soluciones. Se usa por ello intensamente en visión artificial para resolver problemas de reproyección a partir del modelo "pin-hole" de la cámara [7], como el expuesto en el apartado anterior.

En el primer paso de la iteración, el método encuentra una "pose" aproximada al multiplicar una matriz de posición de un objeto en el espacio 3D a partir de su imagen 2D, dependiendo de una distribución de los puntos característicos que componen el objeto, con dos vectores que dependen de las coordenadas de posición de las imágenes de los puntos característicos del objeto.

Los dos vectores resultantes normalizados forman las dos primeras filas de la matriz de rotación R, y la norma de estos vectores es igual al factor de escala de la proyección del espacio 2D al 3D, que proporciona, por tanto, el vector de traslación $\mathbf{t}$.

Las siguientes iteraciones del algoritmo realizan exactamente el mismo proceso pero escogiendo otros puntos característicos de la imagen, corrigiendo así, por tanto la primera iteración [7].

En la figura 2.3 se muestra el esquema básico de una cámara con el centro de proyección en **O**, distancia focal f, plano imagen **G** y puntos característicos $M_o \dots M_i \dots M_n$.

Hay que tener en cuenta que se conocen las coordenadas (U_i, V_i, W_i) del punto característico M_i con respecto al eje de coordenadas del objeto, y que también se conocen las coordenadas (x_i, y_i) de la imagen m_i del mismo punto.

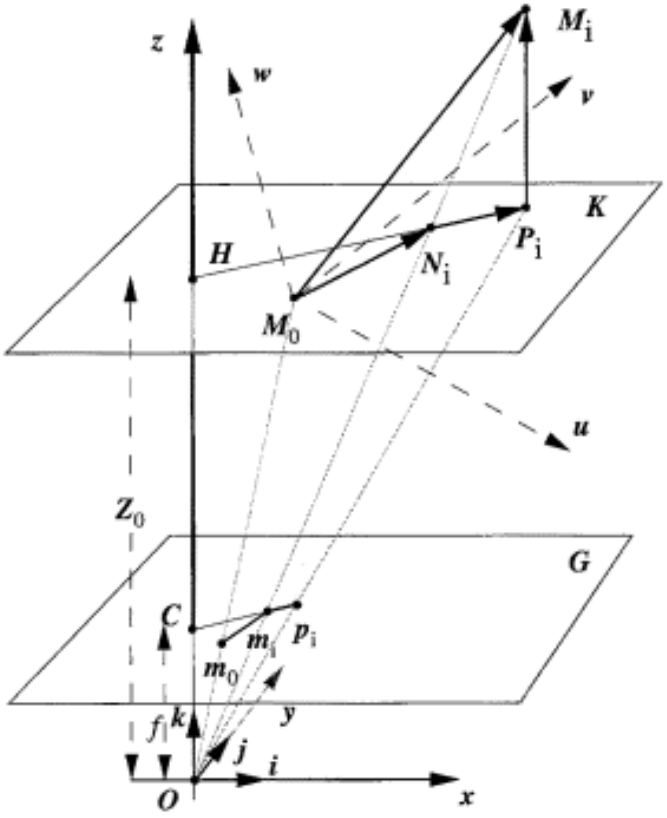

Figura 2.3: Figura obtenida de [7]. Esquema de proyección en perspectiva y proyección ortográfica de un punto M_i

Como se ha comentado anteriormente el objetivo es calcular la matriz de rotación y el vector de traslación. La matriz de rotación **R** es la matriz cuyas filas representan las coordenadas del vector unitario $(i, j, \boldsymbol{k})$ del sistema de coordenadas de la cámara en el sistema de coordenadas del objeto (U_i, V_i, W_i).

Una forma de representar la matriz de rotación es, $\mathbf{R}= \begin{bmatrix} i_u & i_v & i_w \\ j_u & j_v & j_w \\ k_u & k_v & k_w \end{bmatrix}$.

El vector de traslación, **t** es el vector entre el centro de proyección **O** y el punto de referencia M_o. Si este punto ha sido elegido como el punto visible del correspondiente en el plano imagen **G** m_o, el vector **t** es igual a $\left(\frac{Z_o}{f}\right) \boldsymbol{Om_o}$. De esta manera para poder calcular la "pose" es necesario obtener los valores (i, j, Z_o).

Teniendo en cuenta los valores conocidos y desconocidos comentados anteriormente, las ecuaciones 2.2 y 2.3 son las encargadas de relacionar estos valores y de las que, despejando las incógnitas deseadas, se obtiene la información necesaria para poder definir la "pose" de los puntos característicos del objeto.

$$\boldsymbol{M_oM_i}\left(\frac{f}{Z_o}\right) \cdot i = x_i \cdot (1 + \varepsilon_i) - x_o \tag{2.2}$$

$$\boldsymbol{M_oM_i}\left(\frac{f}{Z_o}\right) \cdot j = y_i \cdot (1 + \varepsilon_i) - y_o \tag{2.3}$$

Sabiendo que el valor ε sigue la expresión:

$$\varepsilon_i = \left(\frac{1}{Z_o}\right) \cdot \boldsymbol{M_o M_i} \cdot \boldsymbol{k}, \tag{2.4}$$

donde $\boldsymbol{k} = i \cdot j$.

Otra forma de reescribir las ecuaciones 2.2 y 2.3, para que sea más cómodo trabajar con ellas, es la que se muestra en las ecuaciones 2.5 y 2.6.

$$\boldsymbol{M_o M_i} \cdot \boldsymbol{I} = x_i \cdot (1 + \varepsilon_i) - x_o, \tag{2.5}$$

$$\boldsymbol{M_o M_i} \cdot \boldsymbol{J} = y_i \cdot (1 + \varepsilon_i) - y_o, \tag{2.6}$$

donde $\boldsymbol{I} = \frac{f}{Z_o} i$ y $\boldsymbol{J} = \frac{f}{Z_o} j$.

Una vez que se tienen claras las ecuaciones generales que se van a emplear para la resolución del método, hay que tener en cuenta cómo se deben aplicar para que el método se comporte como se ha explicado anteriormente.

Primero, y correspondiéndose con la primera iteración del método, dando valores a ε las ecuaciones 2.5 y 2.6 forman un sistema lineal del que se puede calcular el valor de las incógnitas $\boldsymbol{I}$ y $\boldsymbol{J}$. Normalizando estos valores se obtienen las incógnitas i y j.

Por otro lado, para calcular el valor de Z_o es necesario calcular la norma de los vectores $\boldsymbol{I}$ y $\boldsymbol{J}$.

Con estos valores calculados se llega a la parte iterativa del método en la que el objetivo es ajustar el valor de ε para que la "pose" del objeto sea más precisa.

2.4 Evaluación

2.4.1 Bases de datos

Las bases de datos sobre las que se van a realizar los análisis con el método explicado anteriormente son CAV3D [5] y AV16_3 [4], tal y como se explica en la introducción 1 del documento.

Estas bases de datos proporcionan una serie de secuencias audio-visuales en las que se puede observar y escuchar una o varias personas moviéndose por el interior de una sala. Además, proporcionan el GT en 3D de la posición de los locutores (personas que aparecen en la secuencia hablando) en cada momento. Esta información resulta muy útil para validar la fiabilidad del algoritmo realizado.

A continuación, se explicarán brevemente las particularidades que tienen cada una de las bases de datos anteriormente mencionadas.

La información proporcionada por CAV3D se recoge en una habitación de 4,77 x 5,95 x 4,5 metros mediante un conjunto de 8 micrófonos colocados en el centro de la sala, y 5 cámaras a color (1 cámara colocada en el centro de la sala junto a los micrófonos y las otras 4 colocados en las esquinas superiores de la sala). Todas ellas realizan la grabación de lo que ocurre en la sala a una velocidad de 15fps.

En la figura 2.4 se observa la sala sobre la que se recoge la información de esta base de datos.

Dependiendo del número de personas que participen en la secuencia, estas se clasifican en tres tipos diferentes tal y como se explica a continuación.

- CAV3D_SOT: una persona detectada en audio y vídeo.
- CAV3D_SOT2: una persona detectada en audio, múltiples personas detectadas en vídeo.

Figura 2.4: Habitación donde se graban y extrae la información en las secuencias de la base de datos CAV3D [5]

- CAV3D_MOT: múltiples personas detectadas tanto en audio como en vídeo.

Figura 2.5: (a) Se muestra una imagen de una secuencia CAV3D_SOT (b) En esta imagen puede observarse un ejemplo de una secuencia CAV3D_SOT2 (c) Imagen sacada de una secuencia CAV3D_MOT

Como puede desprenderse del listado anterior hay que distinguir entre las personas que son detectadas en audio y vídeo, a las que se denominan locutores, y las personas que solo se detectan en vídeo. Es importante realizar esta aclaración porque en el análisis solo se van a tratar a los locutores.

De esta manera, en la figura 2.5, se puede observar un ejemplo de los diferentes tipos de secuencias. A la izquierda, figura 2.5.(a), se muestra una imagen correspondiente a una secuencia tipo CAV3D_SOT en la que aparece un locutor. En la figura del centro, 2.5.(b), la imagen pertenece a una secuencia del tipo CAV3D_SOT2 en la que se puede observar una persona, que no habla en ningún momento, y un locutor. En cuanto a la figura de la derecha, 2.5.(c), se muestra una imagen en la que aparecen dos locutores que pertenece a una secuencia del tipo CAV3D_MOT, pues ambos se ven y hablan.

En cuanto a AV16_3 [4], la información se recoge en una sala de 8,2 x 3,6 x 2,4 metros mediante dos arrays de 8 micrófonos en el centro de la sala y 3 cámaras a color (colocadas en tres de las esquinas superiores de la sala) que realizan la grabación a una velocidad de 25fps.

En la figura 2.6 se muestra una imagen de la sala correspondiente en la que pueden observarse los arrays de micrófonos con los que se construye AV16_3.

Figura 2.6: Habitación donde se graban y extrae la información en las secuencias de la base de datos AV16_3 [4]

Aunque en esta base de datos no se incluye una clasificación como tal, pueden diferenciarse de igual manera que en CAV3D, distintos tipos de secuencia en cuanto al número de locutores que recogen las cámaras y los micrófonos, de este modo:

- AV16_3_SOT: una persona detectada en audio y vídeo.
- AV16_3_MOT: múltiples personas detectadas tanto en audio como en vídeo.

Figura 2.7: (a) Imagen de una secuencia AV16_3_SOT (b) Imagen sacada de una secuencia AV16_3_MOT

Una apreciación importante que cambia con respecto a la otra base de datos es que en esta todas las personas que aparecen en las secuencias son locutores, por eso, existen solo dos tipos de secuencias.

En la figura 2.7, se observa un ejemplo de cada uno de los tipos de secuencia. En la izquierda, figura 2.7.a, se muestra una imagen correspondiente a una secuencia del tipo AV16_3_SOT en la que aparece un único locutor. Por el contrario, la figura 2.7.b , muestra una imagen obtenida de una secuencia del tipo AV16_3_MOT en la que se observan a dos locutores.

Más adelante, en el apartado siguiente, se explicarán las diferencias que existen a la hora de realizar los análisis dependiendo del tipo de secuencia del que se trate.

2.4.2 Métricas

En este apartado se va a explicar que significa y en que unidades se mide cada uno de los resultados de medida de exactitud y fiabilidad (métricas) obtenidos para realizar el análisis del algoritmo propuesto en las secuencias de AV16_3 y CAV_3D, tanto en coordenadas 2D como en coordenadas 3D.

Se han elegido estas métricas porque son las de uso extendido en los trabajos relacionados con el seguimiento de objetos o personas [5].

- **inFov**: Indica el porcentaje de la secuencia donde el locutor se encuentra dentro del campo de visión de la cámara. Para ello, como puede verse en la ecuación 2.7, se divide el número de imágenes de los que se tiene información de "pose" en el GT, entre el número de imágenes totales que se analizan (componen) en esa secuencia.

$$\boldsymbol{inFov}=\frac{Número\_imagenes\_con\_info}{Número\_total\_imagenes}\cdot 100 \tag{2.7}$$

- **noDet**: Indica el porcentaje en el cual no se ha detectado el rostro de la persona. Para calcularlo, se divide el número de imágenes en los que no se detecta el rostro de ningún locutor, mediante el algoritmo usado, entre el número de imágenes de los que se tiene información (explicado anteriormente) como puede verse en la ecuación 2.8.

$$\boldsymbol{noDet}=\frac{Número\_imagenes\_sin\_detección}{Número\_imagenes\_con\_info}\cdot 100 \tag{2.8}$$

En cuanto a coordenadas 2D, se explica a continuación el listado de resultados obtenidos con el sistema propuesto, para el cálculo del error en la imagen, que por tanto se expresan en pixeles.

- $\mathcal{E}$ **2D**: Valor medio del error (distancia Euclidea de la estimación al GT) obtenido en la estimación de la posición calculada de la boca del locutor en las imágenes que componen la secuencia, indicado en pixeles.
- $\mathcal{E}'$ **2D**: Valor medio del error obtenido en la estimación de la posición calculada de la boca del locutor, en las imágenes que componen la secuencia, teniendo en cuenta solo los valores que no supera 1/30 del valor en pixeles de la diagonal(umbral). Con este error se quiere visualizar el error cometido, teniendo en cuenta solo los valores obtenidos correctamente, ya que si el valor obtenido supera el umbral, se debe a que ha ocurrido algún error durante el proceso y no se tiene en cuenta esa medida.
- **TLR**: Promedio de imágenes de la secuencia en los que el error cometido en la estimación de la posición de la boca del locutor es mayor que el umbral (1/30 del valor en pixeles de la diagonal). Esta tasa indica el porcentaje de imágenes en el que se ha producido algún error en el proceso de localización.

Por su parte, en los análisis en 3D, el error de posición cometido en la estimación se calcula en milímetros, a través de las métricas que se explican a continuación.

- $\mathcal{E}$ **3D**: Valor medio del error de posición (en distancia Euclidea con el valor de GT en cada caso, en el espacio 3D) obtenido en la estimación de la posición calculada de la boca del locutor expresado en mm.

- ε' **3D**: Valor medio del mismo error obtenido en la estimación de la posición calculada de la boca del locutor, teniendo en cuenta, como límite, solo los valores que no superan los 30cm de error.
- **TLR**: Promedio de imágenes, con respecto a las imágenes en las que se obtiene información, en las que el error cometido supera los 30cm. Las imágenes en las que se supera este valor indican que ha ocurrido un error durante el proceso.

Capítulo 3

Desarrollo

A fuerza de construir bien, se llega a buen arquitecto[1].

Aristóteles

3.1 Introducción

Este capítulo se encarga de explicar todo el trabajo realizado por el alumno, comentando como se han implementado mediante lenguaje de programación en Matlab los diferentes módulos que componen el sistema completo.

Es necesario mencionar que se ha partido de un código fuente inicial que se ha modificado para satisfacer completamente las necesidades que requiere este sistema y así, poder realizar su funcionalidad de una manera correcta.

Al comienzo del capítulo se explica el funcionamiento general del sistema mediante un diagrama de flujo en el que pueden observarse las diferentes fases por las que hay que pasar para obtener los resultados. De la misma manera, una vez explicado el funcionamiento general, se van a explicar los diferentes módulos que componen dicho sistema, indicando la función específica que tiene que realizar cada uno de ellos.

Para realizar mejor todas estas explicaciones se va a incluir pseudocódigo, esto facilitará la comprensión al lector a la hora de entender la funcionalidad de cada uno de los módulos.

3.2 Procedimiento general

El sistema creado recibe como entrada la secuencia que se quiere analizar y debe ser capaz de entregar a la salida el error cometido en la estimación de la localización de la boca de los locutores en coordenadas 2D y 3D. Como se explicará más adelante, a la hora de realizar el análisis el usuario deberá seleccionar la secuencia de entrada y escoger el modelo de análisis que quiere realizar.

En la figura 3.1 se muestra el diagrama de bloque general del sistema, en el se pueden observar los diferentes bloques que lo componen para que pueda cumplir con su función.

La primera parte del sistema tiene como objetivo obtener las coordenadas de la boca tanto en 2D como en 3D. Una vez calculadas esas coordenadas, la última parte del sistema es la encargada de calcular el error

[1] Tomado de ejemplos del proyecto TEXIS.

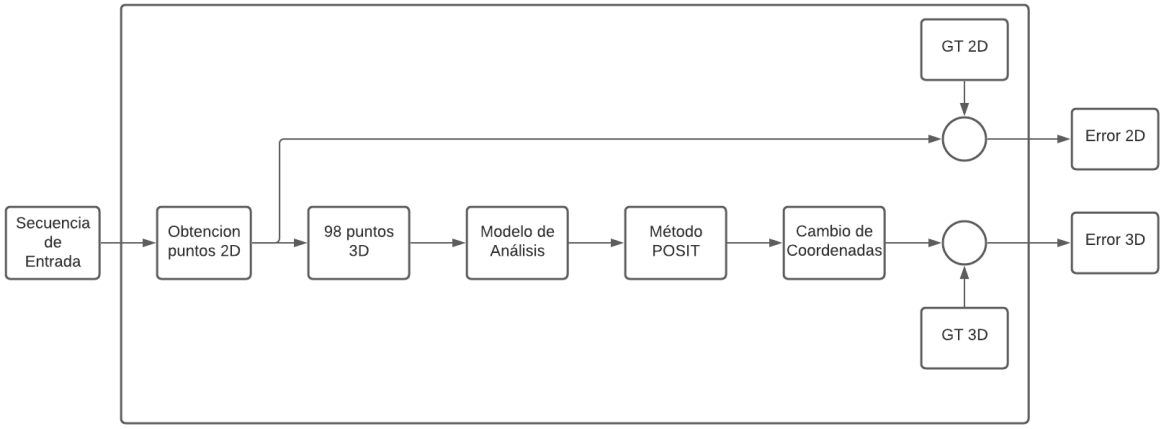

Figura 3.1: Esquema del sistema completo

cometido comparando las coordenadas obtenidas anteriormente con el Ground Truth (GT) proporcionado por las bases de datos.

Ahora que se ha entendido el funcionamiento general del sistema, es necesario comprender los diferentes pasos que se deben realizar para que el sistema funcione correctamente.

En primer lugar se deben obtener las coordenadas en 2D de los 98 puntos característicos del rostro de los locutores que aparecen en la secuencia. Estas coordenadas son necesarias para obtener el error calculado en 2D pero también para poder calcular las coordenadas en 3D de la boca del locutor.

En segundo lugar, para poder calcular el error cometido en 3D, es necesario introducir las coordenadas 3D de los 98 puntos faciales que componen un rostro. Estas coordenadas son necesarias para que el sistema haga la correspondencia entre los puntos 2D de la imagen y estos puntos 3D.

En tercer lugar, hay que seleccionar uno de los cuatro modelos diferentes de análisis con los que se va a trabajar. En estos modelos se elige si se quiere o no corregir la distorsión producida por la cámara y si se quieren escoger los 98 puntos del rostro o se eliminan los de la mandíbula.

En cuarto lugar, y aplicando el método POSIT, se obtienen el vector de rotación y el de traslación que son necesarios para poder calcular las coordenadas de la boca en 3D teniendo como referencia el sistema de coordenadas de la cámara.

Por último, realizando movimientos de rotación y traslación se realiza un cambio de coordenadas 3D de la boca del locutor para pasar del sistema de referencia de la cámara al sistema de coordenadas del mundo.

Llegados a este punto, el objetivo de la primera parte del sistema estaría cubierto, por lo que solo quedaría calcular el error cometido en el cálculo de las coordenadas 3D y 2D para conseguir el objetivo final. Para realizar el cálculo del error se calcula la distancia existente entre el el punto en 3D de la boca calculado y el que presenta el GT.

A continuación, se va a explicar en el pseudocódigo 3.1 lo comentado anteriormente para entenderlo mejor. Resumiendo las diferentes acciones por las que hay que pasar para poder cumplir con el objetivo.

Para poder explicar mejor cada uno de los bloques internos de funcionamiento del sistema propuesto en este TFG, listados en el algoritmo 3.1, se organizan en las tres secciones siguientes:

Algoritmo 3.1: Funcionalidad y módulos del sistema

Objetivo: Cálculo del error cometido en la estimación de la posición de la boca de los locutores 2D y en 3D;
Seleccionar la secuencia a analizar;
Cálculo de los puntos 2D del rostro en la imagen;
Coordenadas 3D de los 98 puntos que componen un rostro;
Elegir el modelo de análisis;
Obtener los puntos del rostro en 3D en coordenadas de la cámara;
Cambiar el sistema de coordenadas de referencia;
Calcular el error cometido en 2D y 3D comparando el resultado obtenido con el GT;

- **Extracción de puntos característicos en 2D**, en la sección 3.3.
- **Reproyección 3D**, en la sección 3.4.
- **Cálculo del error de "pose"**, en la sección 3.5.

Ahora que se tiene una idea general del funcionamiento interno del sistema y de los bloques que lo componen se va a explicar paso a paso la funcionalidad de cada uno de ellos.

3.3 Extracción de puntos característicos en 2D

3.3.1 Secuencia de análisis

Para que el sistema pueda realizar su función correctamente, es imprescindible que conozca la secuencia que debe analizar. Las secuencias con las que se va a trabajar pertenecen a las dos bases de datos comentadas anteriormente, CAV3D y AV16_3.

Para llegar hasta la secuencia, primero hay que indicar cual es el directorio en el que se encuentran las bases de datos. Una vez indicado el directorio, hay que seleccionar la base de datos a la que pertenece la secuencia de análisis para, por último, indicar el número de la secuencia a analizar.

Para que la explicación sea más detallada, en el pseudocódigo 3.2 se muestran los pasos comentados anteriormente para seleccionar la secuencia de entrada al sistema.

Algoritmo 3.2: Selección de la secuencia de entrada

```
Acceso al directorio de las bases de datos: addpath('../../../../../databases/');
Selección de la base de datos: database = 'CAV3D' o 'AV16_3'; ;
Selección de la secuencia;
if database == CAV3D then
    sequences='seq06','seq07','seq08','seq09','seq10', 'seq11','seq12','seq13','seq14','seq15','seq16',
     'seq17','seq18','seq19','seq20','seq21','seq22', 'seq23','seq24','seq25','seq26'
else if database == AV16_3 then
    sequences = 'seq08-1p-0100','seq11-1p-0100','seq12-1p-0100','seq18-2p-0101','seq19-2p-0101',
     'seq24-2p-0111','seq25-2p-0111','seq30-2p-1101';
```

Otra información importante a tener en cuenta por el sistema es conocer qué imágenes de la secuencia tiene que analizar. Esto es debido a que no se analizan todas las secuencias completas, si no que se analiza la parte de la secuencia que aporta información a la cámara con la que se esta trabajando.

En la figura 3.2 se muestra un listado de todas las secuencias que se van a analizar pertenecientes a las dos bases de datos, CAv3D y AV16_3. La columna initFrame indica el imagen en el que debe comenzar el análisis mientras que la columna endFrame, hace referencia al número de imagen donde debe terminar el análisis. Con esta información el sistema es capaz de analizar la parte de la secuencia de entrada de la que se quiere obtener la posición de la boca del locutor.

AV16_3		
	initFrame	endFrame
seq08-1p-0100	14	479
seq11-1p-0100	122	546
seq12-1p-0100	86	1157
seq18-2p-0101	17	1300
seq19-2p-0101	0	473
seq24-2p-0111	295	480
seq25-2p-0111	124	224

CAV3D		
	initFrame	endFrame
seq06	145	669
seq07	202	749
seq08	165	992
seq09	147	724
seq10	244	637
seq11	62	921
seq12	137	1227
seq13	178	1154
seq14	122	861
seq15	78	836
seq16	131	935
seq17	116	905
seq18	127	757
seq19	200	778
seq20	62	615
seq21	47	638
seq22	112	360
seq23	131	725
seq24	146	790
seq25	147	785
seq26	191	435

Figura 3.2: Secuencias de las dos bases de datos con los imágenes que se deben analizar

Una vez que el sistema sabe cual es la secuencia de entrada, se va a explicar de manera detallada todo el proceso que se realiza desde que la secuencia entra en el sistema hasta que se obtiene el error producido en el análisis.

Para ello se explicará como se han creado los diferentes módulos que componen el sistema y la funcionalidad de cada uno de ellos.

3.3.2 Obtención de puntos 2D

Una vez que la secuencia ha entrado en el sistema, el primer paso es obtener las coordenadas en 2D de los puntos que conforman el rostro de las personas que aparecen en la secuencia. Dichas coordenadas son esenciales para poder calcular el error cometido en la estimación de estos puntos 2D, pero a su vez, son necesarias para poder aplicar el método de análisis y obtener las coordenadas de esos puntos en 3D.

Para obtener toda la información relativa a los rostros de las personas que aparecen en las imágenes que componen la secuencia se ha empleado un sistema compuesto por un par de Red Convolucional Neuronal (CNN) colocados en forma de cascada. Esta información está recogida en el archivo **.json** perteneciente a esa secuencia [3].

En este archivo aparece, imagen a imagen, la siguiente información:

- **face_id**: Número identificador del rostro de cada persona.
- **bbox**: Coordenadas de las esquinas que forman el cuadrado de detección del rostro.
- **landmarks**: Coordenadas 2D de los puntos que conforman la cara.
- **landmarks_ids**: Número de identificación de cada punto.
- **headpose**: Orientación del rostro respecto a los ejes de movimiento de la cabeza *[yaw, pitch, roll]*.

Figura 3.3: imagen de una secuencia en la que se pinta el **bbox** en azul y los **landmarks** en rojo

En la figura 3.3 se muestra un ejemplo de un imagen de una secuencia en el que se ha pintado en azul el cuadrado que forman las esquinas proporcionadas por el **bbox** y en rojo los 98 puntos que componen el rostro de una persona cuyas coordenadas las proporcionan los **landmarks**.

La herramienta Matlab proporciona una una función para extraer de manera ordenada toda la información procedente de este tipo de archivos. En el pseudocódigo 3.3 se observa como gracias a esa función, **jsondecode()**, se almacena la información del archivo en la variable **jsonStruct**.

Algoritmo 3.3: Cálculo de los puntos 2D del rostro

```
jsonStruct = jsondecode(fileread(fname));
```

3.4 Reproyección 3D

3.4.1 Escalado de los 98 puntos 3D

Una vez que se conocen las coordenadas 2D de los puntos de la cara, para poder obtener la posición de la boca de las personas en coordenadas 3D del mundo, es necesario conocer las coordenadas en 3D de los

98 puntos que componen el rostro de una persona. Para ello, como se ha mencionado anteriormente, se va a trabajar con un estándar que diferencia 98 puntos característicos dentro de un rostro.

En la figura 3.4 se muestra una imagen del frente y perfil de los 98 puntos en coordenadas 3D comentados anteriormente.

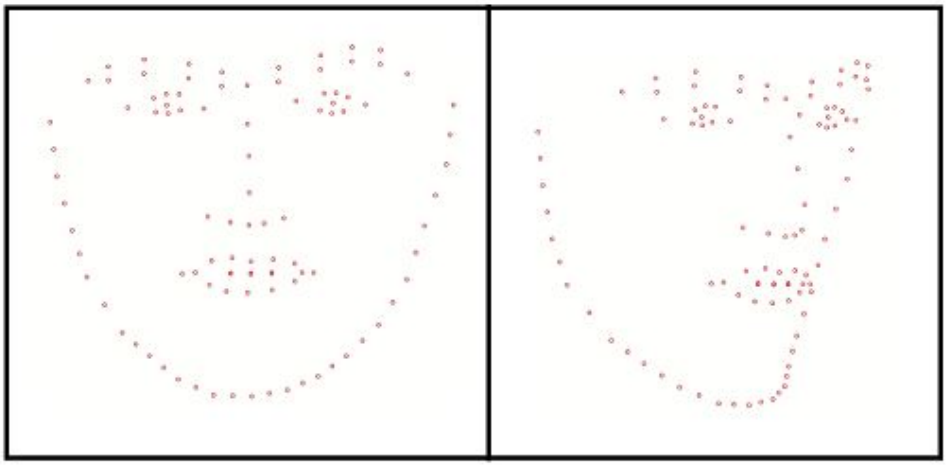

Figura 3.4: Visión de frente y perfil de los 98 puntos faciales en coordenadas 3D

En el estándar, el eje del sistema de coordenadas está situado en el punto característico que indica el centro de la nariz. Las coordenadas del resto de puntos, indican la distancia en milímetros que existe desde el eje de coordenadas hasta el punto en los tres ejes.

En este trabajo se ha colocado, mediante un movimiento de traslación, el eje de coordenadas en el punto característico que indica el centro de la boca, el punto 95 cuando se usan los 98 puntos y el punto 63 cuando no se tienen en cuenta los puntos de la mandíbula, y se han referenciado los demás puntos en función de esta nueva referencia. Otra modificación a tener en cuenta es que se ha realizado un escalado de todas las coordenadas para que la distancia entre los puntos se corresponda con la de una persona humana. La distancia media entre las pupilas de un adulto son 62.5 mm, por lo que para que la distancia entre pupilas del estándar se corresponda hay que multiplicar todas las coordenadas por 106.65.

Una vez realizados estos cambios dicha multiplicación todas las coordenadas están escaladas con las de un rostro adulto.

3.4.2 Selección del modelo

El siguiente paso es decidir con cual de los cuatro modelos diferentes se quiere realizar el análisis. Cada uno de ellos tiene una características diferentes que se explican a continuación.

- **Modelo 1**: Sin corrección de la distorsión y con los 98 puntos faciales. Este es el conjunto de datos crudos de partida.
- **Modelo 2**: Sin corrección de la distorsión y sin los puntos móviles(mandíbula). Al tratarse de secuencias de audio/vídeo, se produce un movimiento en los puntos de la mandíbula, lo que puede dificultar la extracción de la posición exacta de la boca. Con este modelo se quiere eliminar el error que se produce debido a la movilidad de estos puntos.
- **Modelo 3**: Con corrección de la distorsión y con los 98 puntos faciales. Con este nuevo modelo se pretende corregir el error de la distorsión "barrel", el cual puede observase en las figuras 2.6 y 2.4.

- **Modelo 4**: Con corrección de la distorsión y sin los puntos móviles(mandíbula). En este modelo se emplean las dos correcciones comentadas anteriormente en los modelos 2 y 3.

Como puede apreciarse, los cuatro modelos diferentes son las posibles combinaciones entre decidir si se quiere o no corregir la distorsión producida por la cámara, combinado con si se quieren o no coger los puntos móviles del rostro, mandíbula, a la hora de realizar el análisis.

En el pseudocódigo 3.4 puede apreciarse como los modelos cambian los valores de las variables **distorsion** y **noMandible** en función de las características que requiera el análisis.

Algoritmo 3.4: Modelos de análisis

```
Resultado: Selección del modelo de análisis.
switch metodo do
    case 1(Sin corrección y todos los puntos) do
        disorsion = Distorsion;
        noMandible = 0;
    case 2(Sin corrección y no todos los puntos) do
        disorsion = Distorsion;
        noMandible = 1;
    end
    case 3(Con corrección y todos los puntos) do
        disorsion = noDistorsion;
        noMandible = 0;
    end
    case 4(Con corrección y no todos los puntos) do
        disorsion = noDistorsion;
        noMandible = 1;
    end
end
```

Por un lado, la variable **disorsion** puede tomar el valor *Distorsion*, cuando no se quiere corregir la distorsión producida por la cámara, y el valor *noDistorsion* cuando se elimina dicha distorsión.

Para poder realizar la corrección de la distorsión producida por la cámara durante la grabación, es necesario conocer los parámetros intrínsecos de calibración y las coordenadas 2D de los puntos del rostro.

El cálulo de los parámetros intrínsecos de calibración, se realiza con una función que se encuentra en la librería de Matlab a la que hay que pasarle como parámetros la distancia focal, el punto de referencia (la boca), el tamaño de la imagen y el tipo de corrección que se quiere realizar. Para el correcto funcionamiento del sistema se deben corregir tanto la distorsión radial como la tangencial.

Una vez que se conocen los parámetros intrínsecos, al tener ya las coordenadas de los puntos 2D, es posible calcular las nuevas coordenadas de los puntos 2D del rostro sin distorsión.

En el pseudocódigo 3.5 se pueden observar las funciones empleadas para realizar el proceso.

Algoritmo 3.5: Corrección de la distorsión

```
Resultado: Obtención de los puntos característicos en 2D una vez corregida la distorsión.
if distorsion == noDistorsion then
    Cálculo de los parámetros intrínsecos de la cámara
    intrinsics = cameraIntrinsics(focal_length, face_center, tamano, 'RadialDistorsion', radial,
     'TangentialDistorsion', tangencial);
    Cálulo de los puntos sin distorsión.
    undistortedPoints2D = undistortPoints(facelandmarks2D, intrinsics);
end
```

Por otro lado, la variable **noMandible**, toma el valor 0 cuando se emplean todos los puntos del rostro y el valor 1 cuando no se tienen en cuenta los puntos de la mandíbula.

Las coordenadas de los 98 puntos que componen el rostro de una persona están guardadas en la variable **faceLandmarks3D**. De estos 98 puntos, los 33 primeros componen la mandíbula y el 95 indica la posición de la boca que es el punto de referencia. En los modelos que no se debe tener en cuenta los puntos de la mandíbula, modelos 2 y 4, es necesario eliminar dichos puntos de la variable **faceLandmarks3D** e indicar que la posición del punto de referencia, el de la boca, ya no ocupa la posición 95 sino que ocupa la posición 62 del array. En el pseudocódigo 3.6 se muestra como realizar lo anteriormente comentado.

Algoritmo 3.6: Mandíbula

```
Resultado: Selección puntos mandíbula.
if noMandible == 1 then
    faceLandmarks3D = FaceLandmarks3D(:,34:98);
    mouthIdx2D = mouthIdx2D-mandibleLandmarks;
end
```

3.4.3 Método POSIT

Este apartado tiene como objetivo el cálculo de la posición 3D de la boca de los locutores que aparecen en la secuencia que se esta analizando. La posición de la boca calculada en este apartado tiene como referencia el eje de coordenadas de la cámara. Para almacenar el valor de dichas coordenadas se emplea la variable **mouthLandmarks3Dcam**.

Para poder calcular la posición de la boca es necesario conocer, como se explicó en la parte de teoría, la matriz de proyección, **P**. Dicha matriz está formada por los vectores de rotación, **R_posit** y traslación, **tvec_posit**, calculados mediante el método POSIT.

La herramienta Matlab proporciona una función denominada *modernPosit()* a la que, pasando como parámetros las coordenadas de los puntos característicos en 2D,**facelandmarks2D**, las coordenadas de los puntos característicos 3D,**faceLandmarks3D**, la distancia focal y el punto que se quiere usar de referencia, nos devuelve como resultado las matrices de rotación y traslación comentadas anteriormente.

En el pseudocódigo 3.7 se puede observar el código empleado para el cálculo de los vectores y la matriz de proyección.

Algoritmo 3.7: Cálculo de la matriz de proyección

```
Resultado: Obtención de P
[R_posit tvec_posit] = modernPosit (faceLandmarks2D, faceLandmarks3D', focal_length,
  face_center);
P = [R_posit tvec_posit]
```

Con la matriz de proyección calculada, el siguiente paso es calcular la posición de la boca en el sistema de coordenadas3D de la cámara. Para ello, hay que multiplicar las coordenadas 3D correspondientes al punto que indica la boca, almacenadas en **mouthLandmark3D**, por la matriz de proyección calculada anteriormente. Una consideración a tener en cuenta para realizar correctamente la multiplicación es saber que se está trabajando con matrices. Por este motivo hay que tener en cuenta las dimensiones de cada matriz y hacer alguna modificación para que el producto pueda realizarse.

En el pseudocódigo 3.8 se pueden observar las líneas de código necesarias para obtener la posición de la boca en el sistema de coordenadas de la cámara.

Algoritmo 3.8: Cálculo de la matriz posición 3D de la boca en las coordenadas de la cámara

Resultado: Obtención de **mouthLandmarks3Dcam**

```
mouthLandmark3D = faceLandmarks3D(:,mouthIdx);
mouthLandmarks3DCam = P*[ mouthLandmark3D; ones(1,size(mouthLandmark3D,2))];
mouthLandmarks3DCam = [ mouthLandmarks3DCam
 ;ones(1,size(mouthLandmarks3DCam,2))];
```

3.4.4 Cambio del sistema de referencia

Para poder calcular el error cometido durante el análisis al compararlo con el GT es necesario que los dos puntos que se quieren comparar estén referenciados con respecto al mismo sistema de coordenadas.

El GT proporcionado por las bases de datos, entrega las coordenadas de los puntos con respecto a lo que se denomina, sistema de referencia del mundo. Por este motivo, como la posición de la boca obtenida mediante el método de análisis muestra sus coordenadas con respecto al sistema de coordenadas de la cámara, es necesario realizar un cambio en el sistema de coordenadas y referenciar la posición de la boca del locutor con respecto al sistema de coordenadas del mundo. Una vez realizado este cambio, la posición de la boca calculada y la que muestra el GT tendrán el mismo sistema de referencia y se podrán comparar.

Para poder llevar a cabo este cambio en el sistema de coordenadas, es necesario conocer los parámetros extrínsecos. Estos relacionan la posición que tiene la cámara en el sistema de coordenadas del mundo describiendo la rotación y traslación de la cámara en dicho sistema de coordenadas.

La ecuación 3.1 describe este cambio en el sistema de coordenadas donde **mouthLandmarks3Dcam** es la posición de la boca en las coordenadas de la cámara, **R_e** y **t_e** son los parámetros extrínsecos de la cámara y **mouthLandmarks3Dw** es la posición de la boca en las coordenadas del mundo.

$$\mathbf{mouthLandmarks3Dcam} = \mathbf{R\_e} \cdot \mathbf{mouthLandmarks3Dw} + \mathbf{t\_e} \tag{3.1}$$

Para poder calcular la posición de la boca en el sistema de coordenadas del mundo, es necesario despejar **mouthLandmarks3Dw** de la ecuación 3.1 teniendo en cuenta que se está trabajando con matrices. El desarrollo a seguir para despejar **mouthLandmarks3Dw** es el mostrado en la ecuación 3.2.

$$\mathbf{mouthLandmarks3Dcam} - \mathbf{t\_e} = \mathbf{R\_e} \cdot \mathbf{mouthLandmarks3Dw} \tag{3.2a}$$

$$\mathbf{Rinv} \cdot (\mathbf{mouthLandmarks3Dcam} - \mathbf{t\_e} = \mathbf{mouthLandmarks3Dw} \tag{3.2b}$$

$$\mathbf{Rinv} \cdot \mathbf{mouthLandmarks3Dcam} - \mathbf{Rinv} \cdot \mathbf{t\_e} = \mathbf{mouthLandmarks3Dw} \tag{3.2c}$$

$$\mathbf{Rinv} \cdot \mathbf{mouthLandmarks3Dcam} + \mathbf{Tinv} = \mathbf{mouthLandmarks3Dw} \tag{3.2d}$$

donde **Rinv** es la matriz inversa de **R_e** y **Tinv** es el resultado del producto entre -**Rinv** y **T_e**.

Si se concatenan las matrices **Rinv** y **Tinv** en una sola matriz a la que se denomina **Cinv**, se obtiene la ecuación 3.3.

$$\mathbf{mouthLandmarks3Dw} = \mathbf{Cinv} \cdot \mathbf{mouthLandmarks3Dcam} \tag{3.3}$$

En el pseudocódigo 3.9 se muestra la parte del código correspondiente al cambio de sistema de referencia comentado en este apartado.

Algoritmo 3.9: Cálculo de posición 3D de la boca en el sistema de coordenadas del mundo

Resultado: Obtención de **mouthLandmarks3Dw**

```
R_e = cameraCalib.R;
T_e = cameraCalib.T;
Rinv = inv(R_e);
Tinv = -Rinv*T_e;
Cinv = [ Rinv Tinv];
mouthLandmarks3Dw = Cinv * mouthLandmarks3DCam;
```

3.5 Cálculo del error de "pose"

3.5.1 Cálculo del error 2D

Para realizar el cálculo del error cometido en 2D, primero se calcula la distancia imagen a imagen existente entre el punto que representa la boca en el GT y el punto que referencia la boca obtenido del archivo **.json**. Dependiendo el modelo de análisis que se haya realizado, la información de la boca proporcionada por el **.json** se encontrará en una determinada posición dentro de la variable **facelandmaks2D**.

Para obtener la distancia entre ambos puntos de emplea la ecuación 3.4.

$$error2D = \sqrt{(u_2 - u_1)^2 + (v_2 - v_1)^2} \tag{3.4}$$

donde las coordenadas (u_1, v_1) hacen referencia al punto de la boca almacenado en **facelandmaks2D** mientras que los puntos del GT vienen definidos por las coordenadas (u_2, v_2)

Una vez se tiene calculada la distancia existente entre los puntos para todas las imágenes analizadas, el siguiente paso es realizar una media sumando todos los errores obtenidos y dividiéndolos entre el número de imágenes. La fórmula empleada se puede ver en la escuación 3.5.

$$error2Dmean = \frac{1}{T}\sum_{t=0}^{T} error2D(t) \tag{3.5}$$

Donde *error2D* es un array que almacena el valor del error cometido en cada cada uno de los imágenes analizados por el sistema.

3.5.2 Cálculo del error 3D

En este apartado las operaciones son similares al anterior pero trabajando con los puntos en coordenadas 3D. Para calcular el error cometido en la estimación de la posición 3D de la boca de los locutores, es necesario calcular la distancia existente entre las coordenadas 3D que el GT proporciona, y las coordenadas 3D de los puntos de la boca calculados por el sistema.

En la gráfica 3.5 se representa el punto correspondiente a la boca para cada una de las imágenes en coordenadas 3D. En negro se muestra el punto que corresponde al GT y en rojo está representado el punto calculado por el sistema.

Igual que antes, el primer paso es calcular el error cometido en cada uno de los imágenes como se muestra en la ecuación 3.6, guardando el resultado en la variable **error3D**.

$$error3D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{3.6}$$

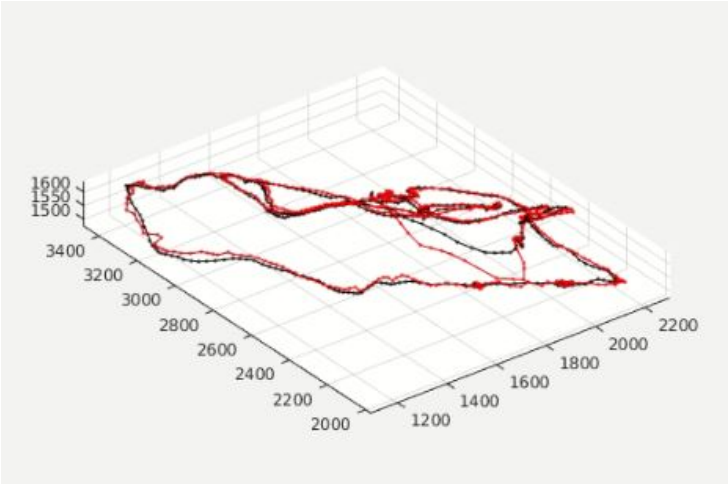

Figura 3.5: En negro se muestra el punto correspondiente a la boca en coordenadas 3D del GT. En rojo se muestra el punto de la boca en coordenadas 3D calculado por el sistema

Para ver de una manera visual este error, en la figura 3.6 se observa una gráfica donde el eje de abscisas indica el número de imagen del análisis mientras que en el eje de ordenadas se indica el error producido en milímetros.

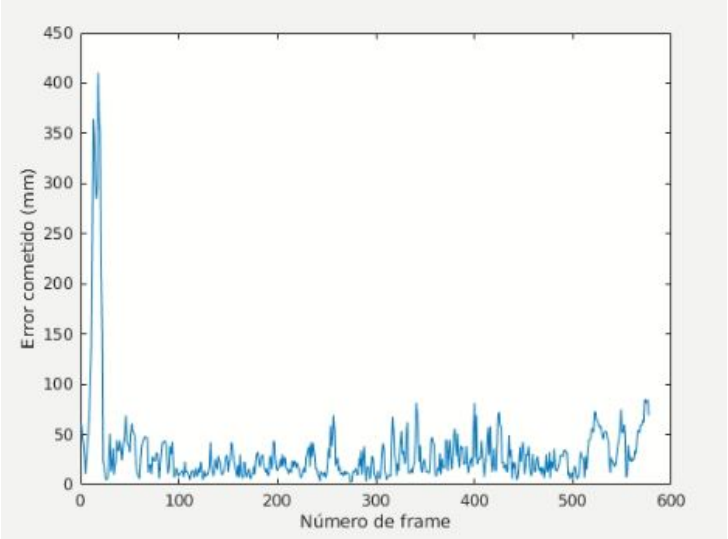

Figura 3.6: Gráfica en la que se muestra el error cometido en cada uno de los imágenes

Donde la posición de la boca calculada, **mouthLandmarks3Dw**, viene definida por las coordenadas (x_1, y_1, z_1) y las coordenadas del GT vienen definidas por (x_2, y_2, z_2).

Como el objetivo es calcular el error medio cometido en el análisis de la secuencia. En la ecuación 3.7 se muestra la fórmula empleada para alcanzar dicho objetivo.

$$error3Dmean = \frac{1}{T}\sum_{t=0}^{T} error3D(t) \tag{3.7}$$

Una vez explicado como el sistema calcula los errores cometidos tanto en coordenadas 2D como en 3D, es hora de pasar al siguiente capítulo, donde se realiza un análisis de los resultados obtenidos.

Capítulo 4

Resultados

Rem tene, verba sequentur (Si dominas el tema, las palabras vendrán solas)[1].

Catón el Viejo

4.1 Introducción

En este capítulo se van a exponer los resultados obtenidos al realizar el análisis de todas las secuencias pertenecientes a las dos bases de datos, CAV3D [5] y AV16_3 [4]. Aunque se muestren los resultados obtenidos, tanto en 2D como en 3D, solo se van a comentar los resultados obtenidos en 3D. Esto se debe a que la mejora de la exactitud del proceso de extracción de la posición de los puntos faciales 2D no es tarea de este TFG, pues, tal y como se explica en el capítulo 3, en este trabajo solo se aplica el algoritmo propuesto en [3], utilizándose como punto de partida del TFG los resultado obtenidos con él.

En primer lugar se van a mostrar los resultados obtenidos al analizar las secuencias de AV16_3 empleando los cuatro modelos de análisis diferentes. Después, se muestran, de igual manera, los resultados obtenidos al analizar las secuencias de CAV3D con los cuatro modelos.

En ambos casos primero se mostrará una tabla con los resultados individuales de cada secuencia y después, se mostrarán los resultados en función del número de locutores que aparecen en la secuencia (SOT, SOT2 o MOT, como se describe en la sección 2.4).

Con objeto de facilitar la lectura del trabajo, se resumen aquí las características de cada uno de los modelos con los que se testan las bases de datos indicadas, explicados completamente el capítulo 3:

- **Modelo 1**: A partir del conjunto de datos crudos obtenidos del extractor de características [3].
- **Modelo 2**: Eliminando los puntos móviles de la mandíbula, de los 98 usados en el modelo 1, se obtiene el conjunto de puntos fijos de la cara.
- **Modelo 3**: Datos crudos con corrección de la distorsión.
- **Modelo 4**: Puntos fijos de la cara con corrección de la distorsión.

[1] Tomado de ejemplos del proyecto TEXIS.

4.2 AV16_3

4.2.1 Modelo 1

En la siguiente figura, figura 4.1, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos AV16_3 empleando el modelo 1. Como se ha mencionado anteriormente, en este modelo se realiza el análisis de la secuencia sin corregir la distorsión y teniendo en cuenta los 98 puntos faciales.

sequence	camera	speaker	class	2D			3D			noDet (%)	InFoV (%)
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)		
8	1	1	SOT	0,0	1,6	1,6	29,0	233	154	0,43	100,0
11	1	1	SOT	0,0	2,5	2,5	63,7	530	154	6,13	100,0
12	1	1	SOT	0,0	1,8	1,8	14,3	179	143	1,12	100,0
18	1	1		0,5	1,4	1,3	33,1	263	168	0,47	100,0
	1	2	MOT	0,4	1,6	1,5	65,6	478	206	16,21	100,0
19	1	1		1,3	1,1	0,9	14,4	164	117	0,00	100,0
	1	2	MOT	1,3	2,2	1,8	6,1	114	96	0,42	100,0
24	1	1		5,7	1,4	1,2	8,6	190	178	5,95	100,0
	1	2	MOT	0,0	1,3	1,3	15,7	168	121	1,08	100,0
25	1	1		0,0	1,5	1,5	6,0	180	171	0,00	100,0
	1	2	MOT	0,0	2,1	2,1	41,0	368	180	28,0	100,0

Figura 4.1: Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos AV16_3

Con un cuadrado rojo, en la figura 4.1, se enmarcan los dos resultados que más sobresalen en la columna noDet(%). Este valor tan elevado se debe a los cruces producidos entre las personas que aparecen en las secuencias de tipo MOT, ya que la cámara no es capaz de detectar el rostro de la persona que se cruza por detrás.

En la figura 4.2 se puede observar el ejemplo comentado anteriormente donde el locutor del jersey amarillo, locutor uno, hace que la cámara no sea capaz de visualizar el rostro del locutor de la chaqueta negra, locutor dos, lo que conlleva al aumento de los frames donde no existe detección para el locutor dos.

Figura 4.2: Frames de la secuencia 25 donde el locutor uno tapa la visión sobre el locutor 2

En la figura 4.3 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT y MOT.

Si se observa la columna noDet(%) se ve como, debido a los cruces mencionados anteriormente, el valor es superior en las secuencias de tipo MOT.

Mirando la columna ε se puede ver como el valor es superior en las secuencias de tipo SOT. Esto sucede porque en este tipo de secuencias, los movimientos que realizan los locutores son más “extremos”, y

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,0	2,0	2,0	35,7	314	150	2,56
MOT	1,1	1,6	1,5	23,8	241	154	6,52

Figura 4.3: Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo

hay muchos frames en los que la localización es complicada. Este efecto también se observa en la columna TLR(%) donde el valor es superior para este tipo de secuencias.

Por último, en la columna ε' el valor de las secuencias SOT es ligeramente inferior al de las secuencias MOT. Lo que significa que, teniendo en cuenta solo los resultados en los que consideramos que no se ha producido un error, la localización es ligeramente mejor en las secuencias de tipo SOT.

4.2.2 Modelo 2

En la siguiente figura, figura 4.4, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos AV16_3 empleando el modelo 2. Como se ha mencionado anteriormente, en este modelo se realiza el análisis de la secuencia sin corregir la distorsión y sin tener en cuenta los puntos de la mandíbula.

sequence	camera	speaker	class	2D			3D			noDet (%)	inFoV (%)
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)		
8	1	1	SOT	0,0	1,6	1,6	41,1	333	121	0,43	100,0
11	1	1	SOT	0,0	2,5	2,5	75,9	735	138	6,13	100,0
12	1	1	SOT	0,0	1,8	1,8	42,9	301	134	1,12	100,0
18	1	1	MOT	1,5	1,4	1,2	15,8	185	134	0,47	100,0
	1	2		0,6	1,6	1,6	78,7	660	210	16,21	100,0
19	1	1	MOT	0,0	0,9	0,9	56,9	381	192	0,00	100,0
	1	2		0,0	2,0	2,0	32,8	266	168	0,42	100,0
24	1	1	MOT	4,9	2,2	1,3	12,4	173	139	5,95	100,0
	1	2		0,5	1,5	1,3	46,5	321	157	1,08	100,0
25	1	1	MOT	0,0	1,5	1,5	0,0	83	83	0,00	100,0
	1	2		0,0	2,1	2,1	60,0	616	147	28,00	100,0

Figura 4.4: Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos AV16_3

En esta ocasión, se han recuadrado en rojo, en la figura 4.4, los resultados 3D obtenidos en la secuencia 25 donde se observa como, una posible mejora, puede afectar de manera tan diferente a la estimación de la posición de la boca de los locutores que aparecen en la misma secuencia. En este caso, con el locutor uno, que durante la secuencia aparece centrado en la imagen y de frente a la cámara, hombre del jersey amarillo y negro de la figura 4.2, se mejoran los resultados obtenidos respecto al modelo anterior. Sin embargo, con el locutor dos, que aparece bordeando la sala y de perfil a la cámara, hombre del jersey negro de la figura 4.2, los resultados obtenidos son peores que en el modelo uno.

En la figura 4.5 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT y MOT.

Comparando estos resultados con los obtenidos en el modelo anterior, se observa como la posible mejora, empeora los resultados obtenidos.

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,0	2,1	2,1	58,5	534	130	2,56
MOT	0,9	1,7	1,5	37,9	336	154	6,52

Figura 4.5: Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo

El valor en las columnas ε y TLR(%) aumenta considerablemente con respecto a los valores obtenidos con el método 1. Este aumento se produce porque, por ejemplo, cuando el locutor esta de lado a la cámara el número de puntos característicos detectados se reduce mucho y el método POSIT no tiene tanta precisión.

Observando los valores de la columna ε' y con lo comentado en el párrafo anterior, se puede deducir que los frames en los que la localización de los puntos faciales es buena, el resultado no varía, pero en los que la localización es mala los resultados empeoran notablemente.

4.2.3 Modelo 3

En la siguiente figura, figura 4.6, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos AV16_3 empleando el modelo 3. Como se ha mencionado anteriormente, en este modelo se realiza el análisis de la secuencia corrigiendo la distorsión y teniendo en cuenta los 98 puntos faciales.

sequence	camera	speaker	class	2D			3D				
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)	InFoV (%)
8	1	1	SOT	0,0	1,6	1,6	37,6	256	177	0,43	100,0
11	1	1	SOT	0,0	2,5	2,5	28,5	272	123	6,13	100,0
12	1	1	SOT	0,0	1,8	1,8	13,2	189	160	1,12	100,0
18	1	1		0,0	1,4	1,4	50,7	327	197	0,47	100,0
	1	2	MOT	0,0	1,6	1,6	15,6	195	121	16,21	100,0
19	1	1		0,0	1,1	1,1	9,1	181	164	0,00	100,0
	1	2	MOT	0,0	2,2	2,2	4,0	120	109	0,42	100,0
24	1	1		0,0	1,2	1,2	25,4	235	185	5,95	100,0
	1	2	MOT	0,0	1,3	1,3	17,3	190	159	1,08	100,0
25	1	1		0,0	1,5	1,5	8,0	218	206	0,00	100,0
	1	2	MOT	0,0	2,1	2,1	30,0	272	143	28,0	100,0

Figura 4.6: Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos AV16_3

Siguiendo con el análisis de la secuencia 25, en la figura 4.6, puede verse en rojo que la corrección de la distorsión produce el efecto contrario a la eliminación de los puntos móviles. En el locutor uno, el error cometido es algo mayor que en el modelo 1. Pero, con el locutor dos, mejora notablemente la estimación respecto al mismo modelo. Esta mejora en el segundo locutor se debe a que aparece en el borde de las imágenes que es donde más se nota el efecto de la distorsión.

En la figura 4.7 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT y MOT.

Observando los resultados obtenidos en el modelo 3, y comparándolos con los del modelo 1, se puede decir que los resultados han mejorado notablemente con esta posible mejora.

En este caso, las columnas ε y TLR(%) reducen muchos los valores, pero la columna ε' los aumenta ligeramente. Este aumento se produce porque en muchos de los frames en los que la detección inicial no

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,0	2,0	2,0	26,4	239	153	2,56
MOT	0,0	1,5	1,5	20,0	217	161	6,52

Figura 4.7: Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo

es buena, y que con los otros modelos el error cometido superaba el umbral, con este modelo el error es menor que el umbral y hace aumentar un poco esta tasa de error.

4.2.4 Modelo 4

En la siguiente figura, figura 4.8, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos AV16_3 empleando el modelo 4. Como se ha mencionado anteriormente, en este modelo se realiza el análisis de la secuencia corrigiendo la distorsión y sin tener en cuenta los puntos de la mandíbula.

sequence	camera	speaker	class	2D			3D			noDet (%)	InFoV (%)
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)		
8	1	1	SOT	0	1,6	1,6	36,8	300	141	0,43	100,0
11	1	1	SOT	0,0	2,5	2,5	49,8	420	135	6,13	100,0
12	1	1	SOT	0,0	1,8	1,8	24,5	220	125	1,12	100,0
18	1	1	MOT	0,5	1,4	1,3	19,5	217	154	0,47	100,0
	1	2		0,4	1,6	1,5	41,8	339	196	16,21	100,0
19	1	1	MOT	0,2	0,9	0,9	14,2	165	112	0,00	100,0
	1	2		0,2	2,2	2,2	21,6	213	142	0,42	100,0
24	1	1	MOT	0,0	1,2	1,2	13,5	191	165	5,95	100,0
	1	2		0,0	1,3	1,3	5,4	156	145	1,08	100,0
25	1	1	MOT	0,0	1,5	1,5	0,0	97	97	0,00	100,0
	1	2		0,0	2,1	2,1	57,0	506	161	28,0	100,0

Figura 4.8: Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos AV16_3

De nuevo, en la figura 4.8, se muestra recuadrado en rojo los resultados obtenidos en la secuencia 25 donde se puede observar que realizando las dos correcciones, el error neto que se obtiene si solo se tienen en cuenta los resultados con error por debajo del umbral, ε', son menores en los dos locutores que los obtenidos con el primer modelo.

En la figura 4.9 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT y MOT.

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,0	2,0	2,0	37,0	313	134	2,56
MOT	0,2	1,5	1,5	21,6	236	146	6,52

Figura 4.9: Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos AV16_3 clasificando estas secuencias en función de su tipo

Los resultados obtenidos en este modelo mejoran de manera global los obtenidos mediante el modelo 1. Esta mejora es mayor en las secuencias de tipo MOT ya que mejora los resultados de las tres columnas, ε, ε' y TLR(%). En las secuencias de tipo SOT mejoran las columnas ε y ε', pero empeora ligeramente la columna TLR(%).

4.3 CAV3D

4.3.1 Modelo 1

En la siguiente figura, figura 4.10, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos CAV3D empleando el modelo 1.En este modelo se realiza el análisis de la secuencia sin corregir la distorsión y teniendo en cuenta los 98 puntos faciales.

sequence	camera	speaker	class	2D			3D			noDet (%)	inFoV (%)
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)		
6	5	1	SOT	0,0	2,7	2,7	7,8	185	120	6,15	58,9
7	5	1	SOT	0,0	2,8	2,8	13,3	260	124	9,09	46,2
8	5	1	SOT	0,0	3,6	3,6	37,7	349	136	6,57	69,8
9	5	1	SOT	0,0	2,7	2,7	23,1	213	162	0,00	100,0
10	5	1	SOT	0,0	3,7	3,7	29,0	330	118	2,21	80,5
11	5	1	SOT	0,6	5,4	4,9	43,7	635	116	18,54	89,1
12	5	1	SOT	0,1	4,0	4,0	21,1	220	100	5,09	82,8
13	5	1	SOT	0,0	2,3	2,3	12,5	224	121	2,21	60,3
14	5	1	SOT2	0,0	3,1	3,1	7,2	162	146	2,30	100,0
15	5	1	SOT2	0,0	2,8	2,8	21,8	213	154	1,39	95,0
16	5	1	SOT2	0,0	4,2	4,2	26,7	246	174	3,07	93,0
17	5	1	SOT2	0,0	2,6	2,6	12,5	155	102	5,45	99,9
18	5	1	SOT2	0,0	2,6	2,6	20,2	228	178	3,74	97,5
19	5	1	SOT2	0,0	2,7	2,7	17,8	223	156	0,93	74,3
20	5	1	SOT	0,0	2,4	2,4	17,2	184	134	1,90	94,9
21	5	1	SOT	0,0	2,4	2,4	8,5	123	95	1,69	100,0
22	5	1	MOT	0,0	3,6	3,6	36,7	323	143	1,26	96,1
	5	2		0,0	2,6	2,6	35,8	295	151	4,89	98,8
23	5	1	SOT	0,0	2,6	2,6	31,1	243	123	0,51	99,3
24	5	1	MOT	0,0	3,3	3,3	31,4	329	140	1,16	53,6
	5	2		0,0	2,5	2,5	22,7	353	109	7,67	46,5
25	5	1	MOT	0,0	2,5	2,5	16,5	256	166	3,95	55,4
	5	2		0,0	3,2	3,2	3,9	96	77	0,97	64,8
	5	3		0,0	3,0	3,0	20,5	443	137	6,22	35,2
26	5	1	MOT	0,0	2,3	2,3	79,1	433	218	4,90	100,0
	5	2		0,0	2,0	2,0	0,0	81	81	0,00	100,0
	5	3		0,0	1,4	1,4	0,0	55	55	0,00	100,0

Figura 4.10: Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos CAV3D

Una de las cosas que llama la atención al observar la figura 4.10 es que, a diferencia de lo que ocurría en la base de datos analizada previamente (AV16_3), en esta base de datos no transcurren todas las secuencias completas en el campo de visión de la cámara, por lo que la columna InFov(%) no es en todos los casos del 100 %.

En la figura 4.11.a, se pueden observar unos frames de la secuencia 6 donde se ve como el locutor se va saliendo del plano de visión de la cámara. Posteriormente, en la figura 4.11.b se observan los frames de la misma secuencia donde el locutor aparece de nuevo en el plano de visión de la cámara. Todos estos frames intermedios en los que el locutor no aparece en la imagen, son los frames de los que no se tiene información.

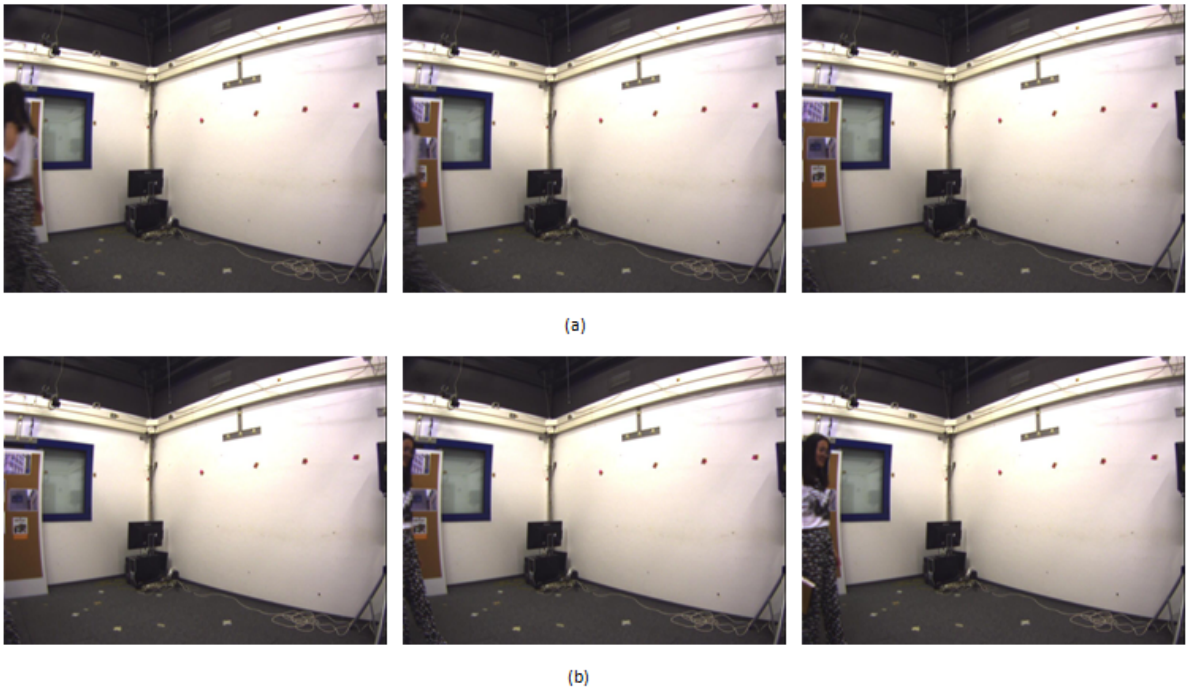

Figura 4.11: a) Frames de la secuencia 6 donde se ve al locutor salir del campo de visión de la cámara b) Frames de la secuencia 6 donde se ve al locutor entrar en el campo de visión de la cámara

Otro dato que llama la atención es el 18,54 % noDet de la secuencia 11. Este dato es tan elevado con respeto al resto porque, como puede verse en la figura 4.12, en esta secuencia el locutor se coloca de espaldas a la cámara en varias ocasiones haciendo imposible la detección de su rostro.

Si nos fijamos en los resultados 3D de la secuencia 26, enmarcados en rojo, se observa como en la misma secuencia los resultados varían mucho en función de los recorridos que hagan los locutores.

En la figura 4.13, se observa claramente como el locutor uno, hombre del jersey azul, se levanta y se sienta varias veces de la silla mientras que los locutores dos, mujer del jersey blanco y negro, y tres, mujer del jersey rojo, se encuentran estáticos mirando a la cámara por lo que su detección es mucho mejor.

En la figura 4.14 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT, SOT2 y MOT.

Observando los resultados, podemos apreciar como, a diferencia de lo que ocurría en la otra base de datos, en esta base de datos las secuencias SOT tiene una tasa noDet(%) superior a las secuencias MOT, siendo las secuencias SOT2 las de menor tasa noDet(%). Esta diferencia se debe a que en varias de las secuencias del tipo SOT el locutor se pasa una parte de la secuencia de espaldas a la cámara por lo que no es posible la detección de su rostro.

Si nos fijamos en las demás columnas, ε, ε' y TLR(%), vemos como todos los valores son inferiores a los obtenidos con el mismo modelo en la otra base de datos.

4.3.2 Modelo 2

En la siguiente figura, figura 4.15, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos CAV3D empleando el modelo 2. Como se ha

Figura 4.12: Frames de la secuencia 11 donde se ve al locutor de espaldas a la cámara

mencionado anteriormente, en este modelo se realiza el análisis de la secuencia sin corregir la distorsión y sin tener en cuenta los puntos de la mandíbula.

Observando los resultados en 3D obtenidos con el modelo 2,4.15, se puede ver como el error cometido y el TLR empeora en todas las secuencias con respecto al modelo anterior.

Si nos fijamos de nuevo en los resultados de la secuencia 26, enmarcada en rojo, se puede ver como ambos errores, ε y ε', son mayores para los tres locutores que aparecen en la secuencia. Uno de los motivos por los que esto ocurre es porque el método POSIT es más preciso cuando trabaja con mayor número de puntos característicos.

En la figura 4.16 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT, SOT2 y MOT.

Observando los valores, vemos como todos son superiores a los obtenidos con el modelo 1. Al igual que pasaba con la base de datos anterior, esta posible mejora, por norma general, empeora los resultados obtenidos como se ha explicado anteriormente para la secuencia 26.

4.3.3 Modelo 3

En la siguiente figura, figura 4.17, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos CAV3D empleando el modelo 3. Como se ha mencionado anteriormente, en este modelo se realiza el análisis de la secuencia corrigiendo la distorsión y teniendo en cuenta los 98 puntos faciales.

Observando los resultados en 3D de la figura 4.17 se aprecia como la corrección de la distorsión, conlleva una mejora importante en la estimación de la localización de la boca de los locutores ya que reduce mucho tanto los errores como el TLR.

Figura 4.13: Frames pertenecientes a la secuencia 26

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,1	3,2	3,1	22,3	270	123	4,91
SOT2	0,0	3,0	3,0	17,7	205	152	2,81
MOT	0,0	2,6	2,6	24,6	266	128	3,10

Figura 4.14: Resultados obtenidos al analizar con el modelo 1 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo

Siguiendo con la evaluación de la secuencia 26, en la figura 4.17, enmarcado en rojo, se pueden ver como los resultados de los tres locutores han mejorado. Centrando el análisis en los resultados del locutor uno, el hombre del jersey azul de la figura 4.13, se puede ver como el error ha disminuido muy considerablemente con respecto al resultado obtenido con el modelo 1. Esta mejora es tan alta porque el locutor uno está toda la secuencia cerca del límite del campo de visión de la cámara (borde de la imagen) y es justo ahí dónde se produce una mayor distorsión.

En la figura 4.18 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT, SOT2 y MOT.

A diferencia de lo que ocurría con el modelo anterior, modelo 2, la mejora introducida en este modelo mejora, con creces, los resultados obtenidos si los comparamos con los del modelo 1. La gran mejora producida se debe a que en esta base de datos, en muchas de las secuencias los locutores realizan el recorrido cerca del límite del campo de visión de la cámara y es en ese punto donde más distorsión se produce.

sequence	camera	speaker	class	2D			3D			noDet (%)	inFoV (%)
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)		
6	5	1	SOT	0,0	2,7	2,7	10,5	223	142	6,15	58,9
7	5	1	SOT	0,0	2,8	2,8	18,1	414	182	9,09	46,2
8	5	1	SOT	0,0	3,6	3,6	51,3	656	183	6,57	69,8
9	5	1	SOT	0,0	2,7	2,7	27,6	233	162	0,00	100,0
10	5	1	SOT	0,0	3,7	3,7	35,4	457	138	2,21	80,5
11	5	1	SOT	0,6	5,4	4,9	48,2	715	149	18,54	89,1
12	5	1	SOT	0,1	4,0	4,0	30,1	343	160	5,09	82,8
13	5	1	SOT	0,0	2,3	2,3	14,8	287	134	2,21	60,3
14	5	1	SOT2	0,0	3,1	3,1	34,1	265	183	2,30	100,0
15	5	1	SOT2	0,0	2,8	2,8	32,1	260	164	1,39	95,0
16	5	1	SOT2	0,0	4,2	4,2	47,5	336	193	3,07	93,0
17	5	1	SOT2	0,0	2,6	2,6	22,4	247	141	5,45	99,9
18	5	1	SOT2	0,0	2,6	2,6	31,9	301	216	3,74	97,5
19	5	1	SOT2	0,0	2,7	2,7	34,9	329	188	0,93	74,3
20	5	1	SOT	0,0	2,4	2,4	19,7	203	145	1,90	94,9
21	5	1	SOT	0,0	2,4	2,4	11,8	139	99	1,69	100,0
22	5	1	MOT	0,0	3,6	3,6	54,8	437	126	1,26	96,1
	5	2		0,0	2,6	2,6	54,5	426	200	4,89	98,8
23	5	1	SOT	0,0	2,6	2,6	38,2	284	149	0,51	99,3
24	5	1	MOT	0,0	3,3	3,3	34,6	383	148	1,16	53,6
	5	2		0,0	2,5	2,5	27,3	451	129	7,67	46,5
25	5	1	MOT	0,0	2,5	2,5	21,9	274	155	3,95	55,4
	5	2		0,0	3,2	3,2	7,8	174	129	0,97	64,8
	5	3		0,0	3,0	3,0	30,7	792	227	6,22	35,2
26	5	1	MOT	0,0	2,3	2,3	80,3	463	226	4,90	100,0
	5	2		0,0	2,0	2,0	0,0	117	117	0,00	100,0
	5	3		0,0	1,4	1,4	0,0	118	118	0,00	100,0

Figura 4.15: Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos CAV3D

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,1	3,2	3,1	27,8	360	149	4,91
SOT2	0,0	3,0	3,0	33,8	289	181	2,81
MOT	0,0	2,6	2,6	31,2	364	158	3,10

Figura 4.16: Resultados obtenidos al analizar con el modelo 2 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo

4.3.4 Modelo 4

En la siguiente figura, figura 4.19, se muestran los resultados obtenidos, tanto en 2D como en 3D, al analizar las secuencias correspondientes a la base de datos CAV3D empleando el modelo 4. Como se ha mencionado anteriormente, en este modelo se realiza el análisis de la secuencia corrigiendo la distorsión y sin tener en cuenta los puntos de la mandíbula.

Como ya se ha comentado anteriormente, el modelo 4 junta las dos mejoras que se realizan en los modelos 2 (eliminación de los puntos móviles del rostro) y 3 (corrección de la distorsión). De esta manera observando la figura 4.19,los resultados en 3D obtenidos mejoran en casi todas las secuencias los resultados calculados con el modelo 1. Esto se debe a que en la mayoría de las secuencias la mejora producida con la corrección de la distorsión es más importante que la no mejora producida por la eliminación de los puntos móviles.

En la secuencia 12, por ejemplo, se produce el efecto contrario al que se acaba de explicar. En esta secuencia el aumento producido al calcular la estimación por la eliminación de los puntos móviles es

sequence	camera	speaker	class	2D			3D			noDet (%)	InFoV (%)
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)		
6	5	1	SOT	0,0	2,7	2,7	0,2	46	45	6,15	58,9
7	5	1	SOT	0,0	2,8	2,8	2,4	84	60	9,09	46,2
8	5	1	SOT	0,0	3,6	3,6	10,4	129	79	6,57	69,8
9	5	1	SOT	0,0	2,7	2,7	1,0	32	28	0,00	100,0
10	5	1	SOT	0,0	3,7	3,7	9,2	120	55	2,21	80,5
11	5	1	SOT	0,6	5,4	4,9	37,5	446	70	18,54	89,1
12	5	1	SOT	0,0	4,0	4,0	9,5	125	72	5,09	82,8
13	5	1	SOT	0,0	2,3	2,3	0,9	67	58	2,21	60,3
14	5	1	SOT2	0,0	3,1	3,1	1,5	53	48	2,30	100,0
15	5	1	SOT2	0,0	2,8	2,8	4,4	73	54	1,39	95,0
16	5	1	SOT2	0,0	4,2	4,2	3,5	60	48	3,07	93,0
17	5	1	SOT2	0,0	2,6	2,6	5,3	73	44	5,45	99,9
18	5	1	SOT2	0,0	2,6	2,6	1,3	46	39	3,74	97,5
19	5	1	SOT2	0,0	2,7	2,7	0,0	52	52	0,93	74,3
20	5	1	SOT	0,0	2,4	2,4	0,0	29	29	1,90	94,9
21	5	1	SOT	0,0	2,4	2,4	0,0	47	47	1,69	100,0
22	5	1	MOT	0,0	3,6	3,6	15,5	155	94	1,26	96,1
	5	2		0,0	2,6	2,6	7,6	115	88	4,89	98,8
23	5	1	SOT	0,0	2,6	2,6	0,8	79	76	0,51	99,3
24	5	1	MOT	0,0	3,3	3,3	1,2	75	66	1,16	53,6
	5	2		0,0	2,5	2,5	3,7	132	103	7,67	46,5
25	5	1	MOT	0,0	2,5	2,5	0,0	46	46	3,95	55,4
	5	2		0,0	3,2	3,2	0,0	81	81	0,97	64,8
	5	3		0,0	3,0	3,0	5,8	152	75	6,22	35,2
26	5	1	MOT	0,0	2,3	2,3	0,4	55	54	4,90	100,0
	5	2		0,0	2,0	2,0	0,0	43	43	0,00	100,0
	5	3		0,0	1,4	1,4	0,0	36	36	0,00	100,0

Figura 4.17: Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos CAV3D

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,1	3,2	3,1	6,5	109	56	4,91
SOT2	0,0	3,0	3,0	2,7	60	48	2,81
MOT	0,0	2,6	2,6	3,4	89	68	3,10

Figura 4.18: Resultados obtenidos al analizar con el modelo 3 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo

superior a la mejora que produce la corrección de la distorsión. Esto se produce porque el locutor está la mayor parte del tiempo en el centro de la imagen, por lo que la corrección de la distorsión no es muy alta, y de lado a la cámara, con lo que hay pocos puntos 2D detectados del rostro y el método POSIT no puede realizar una transformación 3D tan precisa. En la figura 4.20 se muestran frames de esta secuencia donde se puede ver al locutor de lado a la cámara.

Igual que en los modelos anteriores, en los resultados en 3D de la secuencia 26 enmarcados en rojo, se observa como en este caso si se mejoran los resultados con respecto al modelo 1. En esta secuencia, como en la mayoría, la corrección de la distorsión produce un efecto mayor que la eliminación de los puntos móviles.

En la figura 4.21 se muestran los resultados que se obtienen al calcular la media de los errores, clasificando las secuencias en función de los locutores y personas que aparecen en ellas, SOT, SOT2 y MOT.

sequence	camera	speaker	class	2D			3D				
				TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)	inFoV (%)
6	5	1	SOT	0,0	2,7	2,7	0,6	61	58	6,15	58,9
7	5	1	SOT	0,0	2,8	2,8	8,2	176	89	9,09	46,2
8	5	1	SOT	0,0	3,6	3,6	28,0	290	116	6,57	69,8
9	5	1	SOT	0,0	2,7	2,7	1,6	47	42	0,00	100,0
10	5	1	SOT	0,0	3,7	3,7	17,6	214	80	2,21	80,5
11	5	1	SOT	0,6	5,4	4,9	41,2	565	88	18,54	89,1
12	5	1	SOT	0,0	4,0	4,0	20,1	222	110	5,09	82,8
13	5	1	SOT	0,0	2,3	2,3	3,9	102	65	2,21	60,3
14	5	1	SOT2	0,0	3,1	3,1	11,0	136	98	2,30	100,0
15	5	1	SOT2	0,0	2,8	2,8	8,2	103	70	1,39	95,0
16	5	1	SOT2	0,0	4,2	4,2	8,6	127	83	3,07	93,0
17	5	1	SOT2	0,0	2,6	2,6	10,3	138	69	5,45	99,9
18	5	1	SOT2	0,0	2,6	2,6	3,8	95	74	3,74	97,5
19	5	1	SOT2	0,0	2,7	2,7	6,7	124	93	0,93	74,3
20	5	1	SOT	0,0	2,4	2,4	0,4	29	28	1,90	94,9
21	5	1	SOT	0,0	2,4	2,4	0,0	43	43	1,69	100,0
22	5	1	MOT	0,0	3,6	3,6	29,7	258	131	1,26	96,1
	5	2		0,0	2,6	2,6	12,7	146	90	4,89	98,8
23	5	1	SOT	0,0	2,6	2,6	2,0	103	95	0,51	99,3
24	5	1	MOT	0,0	3,3	3,3	2,3	120	100	1,16	53,6
	5	2		0,0	2,5	2,5	7,3	162	101	7,67	46,5
25	5	1	MOT	0,0	2,5	2,5	0,0	59	59	3,95	55,4
	5	2		0,0	3,2	3,2	0,9	50	46	0,97	64,8
	5	3		0,0	3,0	3,0	17,1	472	177	6,22	35,2
26	5	1	MOT	0,0	2,3	2,3	0,8	81	79	4,90	100,0
	5	2		0,0	2,0	2,0	0,0	26	26	0,00	100,0
	5	3		0,0	1,4	1,4	0,0	87	87	0,00	100,0

Figura 4.19: Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos CAV3D

Observando los valores obtenidos juntando las dos posibles mejoras, se observa como los datos obtenidos son mejores que los del modelo 1 pero empeoran si los comparamos con los datos obtenidos mediante el modelo 3. El efecto es el mismo que se acaba de explicar para la secuencia 26, el efecto de la corrección de la distorsión es mayor que la eliminación de los puntos móviles.

Figura 4.20: Frames pertenecientes a la secuencia 12

	2D			3D			
	TLR (%)	ε (px)	ε' (px)	TLR (%)	ε (mm)	ε' (mm)	noDet (%)
SOT	0,1	3,2	3,1	11,2	168	74	4,91
SOT2	0,0	3,0	3,0	8,1	121	81	2,81
MOT	0,0	2,6	2,6	7,1	146	89	3,10

Figura 4.21: Resultados obtenidos al analizar con el modelo 4 las secuencias de la base de datos CAV3D clasificando estas secuencias en función de su tipo

Capítulo 5

Conclusiones y líneas futuras

5.1 Introducción

En este capítulo se van a explicar las conclusiones obtenidas en el TFG, analizando los resultados obtenidos en función del tipo de secuencia en el capítulo 4, y se van a exponer además las líneas futuras sobre las que debería seguir investigando.

El capítulo se estructura, por tanto, en dos grandes apartados. En el primero, se van a comentar las principales conclusiones que se extraen al observar los resultados 3D obtenidos durante los análisis. Al principio se van a extraer unas conclusiones independientes para cada una de las bases de datos, y después, se van a mostrar unas conclusiones generales. En el segundo, se expondrán las posibles líneas futuras.

5.2 Conclusiones

En este apartado se puede observar el resumen de los resultados obtenidos para cada uno de los diferentes tipos de secuencia existentes en las bases de datos CAV3D [5] y AV16_3 [4] (ver capítulo 2.4.1), empleando los cuatro modelos diferentes de análisis descritos en el capítulo 3.

En la figura 5.1 se muestran los resultados correspondientes a las secuencias de Av16_3 agrupadas en función del tipo de secuencia.

AV16.3		2D (px)				3D (mm)			
		Modelo 1	Modelo 2	Modelo 3	Modelo 4	Modelo 1	Modelo 2	Modelo 3	Modelo 4
SOT	TLR (%)	0,0	0,0	0,0	0,0	35,7	58,5	26,4	37,0
	ε	2,0	2,1	2,0	2,0	314	534	239	313
	ε'	2,0	2,1	2,0	2,0	150	130	153	134
MOT	TLR (%)	1,1	0,9	0,0	0,2	23,8	37,9	20,0	21,6
	ε	1,6	1,7	1,5	1,5	241	336	217	236
	ε'	1,5	1,5	1,5	1,5	154	154	161	146

Figura 5.1: Resumen de los resultados obtenidos mediante los cuatro modelos diferentes de análisis para las secuencias pertenecientes a las bases de datos AV16_3

Observando los resultados 3D, se pueden extraer varias conclusiones, que se listan a continuación por claridad:

- Con el modelo 3 se consigue la menor tasa de TLR además de conseguir el menor error absoluto, ε.
- Por el contrario, si nos fijamos en el error absoluto teniendo en cuenta solo los valores que se encuentran por debajo del umbral, ε', hay que diferenciar entre las secuencias SOT, donde el modelo con menor error es el 2 y las secuencias MOT, donde el modelo con el error más bajo es el 4.

En la figura 5.2 se muestran los resultados correspondientes a las secuencias de CAV3D agrupadas también en función del tipo de secuencia.

CAV3.D		2D (px)				3D (mm)			
		Modelo 1	Modelo 2	Modelo 3	Modelo 4	Modelo 1	Modelo 2	Modelo 3	Modelo 4
SOT	TLR (%)	0,1	0,1	0,1	0,1	22,3	27,8	6,5	11,2
	ε	3,2	3,2	3,2	3,2	270	360	109	168
	ε'	3,1	3,1	3,1	3,1	123	149	56	74
SOT 2	TLR (%)	0,0	0,0	0,0	0,0	17,7	33,8	2,7	8,1
	ε	3,0	3,0	3,0	3,0	205	289	60	121
	ε'	3,0	3,0	3,0	3,0	152	181	48	81
MOT	TLR (%)	0,0	0,0	0,0	0,0	24,6	31,2	3,4	7,1
	ε	2,6	2,6	2,6	2,6	266	364	89	146
	ε'	2,6	2,6	2,6	2,6	128	158	68	89

Figura 5.2: Resumen de los resultados obtenidos mediante los cuatro modelos diferentes de análisis para las secuencias pertenecientes a las bases de datos CAV3D

Observando los resultados obtenidos sobre CAV3D, se puede extraer como conclusión fundamental que con el análisis realizado con el modelo 3 se obtiene el menor error tanto en ε como en ε' y una menor tasa de TLR independientemente del tipo de secuencia que se esté analizando.

Viendo las conclusiones que se han extraído para ambas bases de datos, se pueden sacar dos conclusiones generales:

- La primera es que la corrección de la distorsión producida por la cámara al grabar las secuencias mejora notablemente los resultados obtenidos independientemente del tipo de secuencia y de la base de datos con la que se trabaje.
- La segunda es que eliminar los puntos móviles del rostro para realizar el análisis, no asegura que los resultados obtenidos vayan a ser mejores, esto depende de los movimientos que realicen los locutores en esa secuencia en concreto.

Con todo ello, y en cualquier caso, se puede concluir que los resultados obtenidos con la propuesta desarrollada y analizada en este TFG arrojan en general buenas tasas de fiabilidad en las bases de datos usadas por la comunidad científica en el área de interés, y por tanto son suficientemente robustos, como para ser usados en la aplicación de seguimiento de locutores con una sola cámara perseguida.

5.3 Líneas futuras

Como posibles líneas futuras derivadas de este trabajo se proponen las siguientes:

- Realizar el análisis con la información extraída con otras cámaras de estas bases de datos para tener un mayor número de resultados y poder desarrollar un análisis más completo de los resultados obtenidos.

- Realizar el análisis de las mismas secuencias con otro método, como el PnP [8], que transforma también los puntos 2D en coordenadas 3D. Posteriormente, sería posible realizar de este modo una comparación entre los resultados obtenidos con los diferentes métodos de análisis (POSIT y PnP).
- Buscar mejoras del método [3] para realizar la detección de los puntos faciales característicos de las imágenes, de forma más adecuada a la aplicación de interés. Por ejemplo, analizar si todos los puntos característicos contribuyen a mejorar la exactitud del sistema, pues en caso contrario se podría reentrenar el sistema de aprendizaje máquina para extraer menos puntos característicos pero con mayor fiabilidad, pues en estas bases de datos el tamaño de la cara es tan pequeño que tenemos la impresión de mejorar los resultados bajando las exigencias de resolución de la red.

Bibliografía

[1] F. Sanabria-Macías, M. M. Romera, J. Macías-Guarasa, D. Pizarro, J. N. Turnes, and E. J. M. Reyes, "Face tracking with a probabilistic viola and jones face detector," in *IECON 2019-45th annual conference of the ieee industrial electronics society*, vol. 1. IEEE, 2019, pp. 5616–5621.

[2] F. Sanabria-Macias, M. Marron-Romera, and J. Macias-Guarasa, "3d audiovisual speaker tracking with distributed sensors configuration," in *2020 28th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 256–260.

[3] R. Valle, J. M. Buenaposada, and L. Baumela, "Cascade of encoder-decoder cnns with learned coordinates regressor for robust facial landmarks detection," *Pattern Recognition Letters*, vol. 136, pp. 326–332, 2020.

[4] G. Lathoud, J.-M. Odobez, and D. Gatica-Perez, "Av16. 3: An audio-visual corpus for speaker localization and tracking," in *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2004, pp. 182–195.

[5] X. Qian, A. Brutti, O. Lanz, M. Omologo, and A. Cavallaro, "Multi-speaker tracking from an audio–visual sensing device," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2576–2588, 2019.

[6] R. Valle, J. M. Buenaposada, and L. Baumela, "Multi-task head pose estimation in-the-wild," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 8, pp. 2874–2881, 2020.

[7] D. F. DeMenthon and L. S. Davis, "Model-based object pose in 25 lines of code," *International journal of computer vision*, vol. 15, no. 1-2, pp. 123–141, 1995.

[8] S. Li, C. Xu, and M. Xie, "A robust o(n) solution to the perspective-n-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1444–1450, 2012.

Apéndice A

Recursos necesarios

A.1 Introducción

A continuación se detallan los recursos Hardware (HW) y Software (SW) empleados durante la realización del proyecto.

A.2 Hardware

Ordenador portátil (Samsung Ultrabook Serie 5), para la realización de la programación y la redacción de la documentación.

A.3 Software

- Matlab: entorno de cálculo donde se ha realizado la programación del sistema.
- TexStudio: editor de LaTex para la elaboración de la memoria.
- Lucidchart: aplicación para realizar los diagramas de bloques.

Apéndice B

Presupuesto

En este último apartado se va a estimar el coste que supone la realización total del proyecto. En primer lugar se realiza una estimación de los costes parciales en función de su procedencia para, al final, mostrar el presupuesto total.

B.1 Coste recursos Software

Para llevar a cabo este proyecto es necesario utilizar la herramienta de software Matlab, con la que se ha implementado el sistema.

En la tabla B.1 se detalla el coste de la licencia de esta aplicación, incluyéndose un coeficiente que cuantifica la vida de uso del software.

Concepto	Precio	Coeficiente	Subtotal
Matlab	200,00 €	0,0833	16,66 €

Tabla B.1: Coste recursos software

B.2 Coste recursos Hardware

El coste de los recursos hardware hace referencia al ordenador empleado para la realización del sistema.

Concepto	subtotal
Samsung Ultrabook serie 5	600,00 €

Tabla B.2: Coste recursos hardware

B.3 Coste recursos humanos

El coste de los recursos humanos procede de la mano de obra de un ingeniero responsable del desarrollo de todo el sistema. Este coste se incluye a continuación.

Concepto	Precio/Hora	Total horas	Subtotal
Ingeniero	50,00 €	500	25000,00 €

Tabla B.3: Coste recursos humanos

B.4 Coste total de los recursos

Los recursos totales son la suma de los recursos software y hardware, junto a los recursos humanos necesarios para la realización del proyecto.

Concepto	Subtotal
Coste recursos software	16,66 €
Coste recursos hardware	600 €
Coste recursos humanos	25000,00 €
TOTAL	25616,66 €

Tabla B.4: Coste total de los recursos

B.5 Costes de ejecución por contrato

Los costes de ejecución por contrato incluyen los gastos derivados de la utilización de las instalaciones donde se ha realizado el proyecto, las cargas fiscales, los gastos financieros, las tasas administrativas y las obligaciones de control del proyecto.

Estos gastos se asumen estableciendo un recargo sobre el coste del importe del presupuesto de ejecución material. Este recargo se establece equivalente al 22 % de dicho importe en este proyecto.

Concepto	Subtotal
21 % de los recursos totales	5503,67 €

Tabla B.5: Costes de ejecución por contrato

B.6 Tasas facultativas

Para este proyecto se fija un porcentaje del 7 % del coste total de ejecución por contrato.

Concept	Subtotal
7 % del coste de la ejecución por contrato	385,26 €

Tabla B.6: Tasas facultativas

B.7 Presupuesto total

En la table B.7 se muestra la suma de todos los costes parciales mencionados anteriormente.

Concepto	Subtotal
Recursos totales	25616,66 €
Coste de ejecución por contrato	5503,67 €
Tasas facultativas	385,26 €
TOTAL (sin IVA)	31605,59 €
IVA (22 %)	6931,23€
TOTAL	38536,82 €

Tabla B.7: Presupuesto total

Universidad de Alcalá
Escuela Politécnica Superior

ESCUELA POLITECNICA
SUPERIOR

Universidad
de Alcalá