Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Lectura de labios en imágenes de vídeo

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: David Gimeno Gómez

Tutor: Carlos David Martínez Hinarejos

Curso 2018-2019

Resumen

Durante una conversación nuestro cerebro se encarga de combinar la información procedente de múltiples sentidos con el objetivo de mejorar nuestra capacidad a la hora de interpretar el mensaje percibido. Además, diferentes estudios han demostrado la relación existente entre las expresiones faciales y su sonido correspondiente. Este efecto nos ha impulsado hacia la construcción de un sistema capaz de leer los labios considerando únicamente la información procedente del canal visual, es decir, capaz de mimetizar la habilidad humana de interpretar el habla leyendo los labios del interlocutor. Para ello, en primer lugar, se ha construido un *dataset* compuesto por planos frontales de cuatro presentadores de telediario, así como las transcripciones asociadas a cada uno de los discursos. Para cada uno de estos discursos, se localiza la región bucal gracias a bibliotecas enfocadas al *machine learning,* como es el caso de *scikit-learn* Tras la compilación de este conjunto de datos, se han procesado los distintos planos de modo que puedan ser interpretados por el sistema. Además, después de este procesado, se han aplicado técnicas de selección de características para prescindir de aquellos datos que no aporten información relevante de cara al reconocimiento del habla. Por otra parte, nuestro sistema se compone de distintos módulos, entre los que destacamos los Modelos Ocultos de Markov Continuos por su gran aporte al ámbito del reconocimiento de voz, o texto manuscrito, entre otros. Estos modelos son entrenados con un subconjunto del *dataset* construido, mientras que sus prestaciones serán comprobadas con los datos restantes. Sin embargo, los resultados obtenidos tras el protocolo experimental no han sido mínimamente aceptables. Esto demuestra la dificultad que presenta la interpretación del habla continua y, más aún, si tenemos en cuenta los desafíos que supone la carencia de un sentido tan crucial como es el oído. Por tanto, nuestro sistema se proyecta sobre trabajos futuros en los cuales volcar el resto de nuestros esfuerzos.

Palabras clave: lectura de labios, *machine learning*, *dataset*, Modelo Oculto de Markov.

Abstract

During a conversation our brain is responsible for combining information from multiple senses in order to improve our ability to interpret the perceived message. In addition, different studies have shown the relationship between facial expressions and their corresponding sound. This effect has driven us towards the construction of a system capable of reading the lips considering only the information coming from the visual channel, that is, capable of mimicking the human ability to interpret speech by reading the interlocutor's lips. For this, first, a dataset composed of frontal views of four television news anchors has been constructed, as well as the transcriptions associated to each one of the speeches. For each of these speeches, the mouth region is located thanks to libraries focused on machine learning, as it is the case of *scikit-learn*. After the compilation of this dataset, the different views have been processed so that they can be interpreted by the system. In addition, after this processing, feature selection techniques have been applied to disregard data that does not provide relevant information for speech recognition. On the other hand, our system is composed of different modules, among which we highlight the Continuous Hidden Markov Models for their great contribution to the field of voice recognition, or handwritten text, among others. These models are trained with a subset of the constructed dataset, while their accuracy will be checked with the remaining data. However, the results obtained after the experimental protocol have not been minimally acceptable. This demonstrates the difficulty presented by the interpretation of continuous speech and, even more so, if we consider the challenges posed by the lack of such a crucial sense as hearing. Therefore, our system is projected on future works in which to focus the rest of our efforts.

Keywords: lipreading, machine learning, dataset, Hidden Markov Model.

Resum

Durant una conversació el nostre cervell s'encarrega de combinar la informació procedent de múltiples sentits amb l'objectiu de millorar la nostra capacitat a l'hora d'interpretar el missatge percebut. A més, diferents estudis han demostrat la relació existent entre les expressions facials i el seu so corresponent. Este efecte ens ha impulsat cap a la construcció d'un sistema capaç de llegir els llavis considerant únicament la informació procedent del canal visual, és a dir, capaç de mimetitzar l'habilitat humana d'interpretar la parla llegint els llavis de l'interlocutor. Per a això, en primer lloc, s'ha construït un *dataset* compost per plans frontals de quatre presentadors de telenotícies, així com les transcripcions associades a cada un dels discursos. Per a cada un d'estos discursos, es localitza la regió bucal gràcies a biblioteques enfocades al *machine learning*, com és el cas de *scikit-learn* Després de la compilació d'este conjunt de dades, s'han processat els distints plans de manera que puguen ser interpretats pel sistema. A més, després d'este processat, s'han aplicat tècniques de selecció de característiques per a prescindir d'aquelles dades que no aporten informació rellevant de cara al reconeixement de la parla. D'altra banda, el nostre sistema es compon de distints mòduls, entre els que destaquem els Models Ocults de Markov Continus per la seua gran aportació a l'àmbit del reconeixement de veu, o text manuscrit, entre altres. Estos models són entrenats amb un subconjunt del *dataset* construït, mentres que les seues prestacions seran comprovades amb les dades restants. No obstant això, els resultats obtinguts després del protocol experimental no han sigut mínimament acceptables. Açò demostra la dificultat que presenta la interpretació de la parla contínua i, més encara, si tenim en compte els desafiaments que suposa la carència d'un sentit tan crucial com és l'oïda. Per tant, el nostre sistema es projecta sobre treballs futurs en els quals bolcar la resta dels nostres esforços.

Paraules clau : lectura de llavis, *machine learning*, *dataset*, Model Ocult de Markov.

Tabla de contenidos

1. Introducción 13
1.1. Motivación 13
1.2. Objetivos 14
1.3. Estructura 14
2. Estado del arte 15
2.1. Introducción al aprendizaje automático 15
2.2. Datasets 16
2.3. Sistemas de Reconocimiento Automático del Habla 18
2.2.1. Aproximaciones tradicionales 19
2.2.2. Aproximaciones *Deep Learning* 20
3. Lectura de labios en imágenes de vídeo 23
3.1. El *dataset* de nuestro sistema 24
3.2. Detección de la región de la boca 25
3.3. Extracción de características visuales del movimiento labial 28
3.3.1. Introducción a las técnicas de extracción de características visuales del movimiento labial 29
3.3.2. LBP-TOP *feature extractor* 30
3.4. Selección de características 33
3.4.1. *Chi square statistic* (Ξ^2) 36
3.4.2. *Principal Component Analysis* 36
3.5. Aportación extra de una base gramatical 37
3.6. Entrenamiento y arquitectura del sistema 38
3.7. Resultados obtenidos 42
4. Conclusiones 43
5. Relación del trabajo desarrollado con los estudios cursados 45
6. Trabajos futuros 47
7. Referencias bibliográficas 49

Tabla de figuras

Figura 1. Esquema general de las fases propuestas para la construcción del sistema 23
Figura 2. Los cuatro presentadores que componen el *dataset* 24
Figura 3. Detección de la región de la boca 25
Figura 4. Secuencia de *frames* de la región de la boca en escala de grises 26
Figura 5. Errores a la hora de detectar la región de la boca 26
Figura 6. Los 68 puntos de referencia faciales para entrenar el predictor 27
Figura 7. Representación de cada par codificado para construir el *dataset* 28
Figura 8. Extracción de características con solapamiento 29
Figura 9. Ejemplo de extracción de características geométricas mediante AAM [9] 30
Figura 10. Representación del algoritmo LBP. (a) Algoritmo LBP básico. (b) Vecindario de ocho píxeles delimitado por un círculo de radio dos [18] 31
Figura 11. Síntesis del algoritmo LBP-TOP. (a) Secuencia de *frames*. (b) Región de la boca en el plano XY. (c) Región de la boca en el plano XT. (d) Región de la boca en el plano YT [4]. 32
Figura 12. Construcción del vector de características de un bloque de *frames* [4] 33
Figura 13. Construcción de la secuencia definitiva que integra los *frames* del discurso al completo 33
Figura 14. División de cada una de las secuencias del *dataset* en sus respectivos bloques de características 35
Figura 15. Restructuración del *dataset* con el objetivo de aplicar la selección de características 35
Figura 16. Arquitectura del sistema empleado 39
Figura 17. Topología del Modelo Oculto de Markov empleado en la lectura de labios 40

1. Introducción

Cuando estamos involucrados en una conversación, nuestro cerebro se encarga de percibir información a través de distintos medios sensoriales. Toda esta información se combina de manera subconsciente en un proceso inferencial en el que se asocian sonidos con ciertos movimientos o expresiones faciales, además de tener en cuenta otros factores como el contexto de la conversación o la gesticulación del emisor[1] . De esta forma, mejoramos nuestra capacidad a la hora de interpretar el discurso emitido. Por otra parte, este proceso interno de nuestro cerebro demuestra la relación existente entre la información visual percibida y su sonido correspondiente. Este principio fue consolidado por un estudio de McGurk y McDonald [1] que destacaba la influencia de la visión sobre la percepción del habla. Aquí encontramos una de las principales razones que nos ha impulsado a emplear técnicas de *machine learning* con el objetivo de construir un modelo capaz de leer los labios sin ningún tipo de información procedente del medio auditivo, es decir, considerando únicamente la información visual a partir de imágenes de vídeo. Al prescindir de un sentido tan crucial como es el oído, será necesario ser conscientes de los desafíos que presenta dicha carencia.

Como bien sabemos, la lectura de labios es una tarea compleja cuyo objetivo es interpretar el habla cuando el sonido no está disponible. Esta técnica abarca numerosos aspectos a tener en cuenta que permitirán mejorar la percepción y compresión de lo que el hablante está diciendo. En otras palabras, se trata de un proceso multimodal donde un oyente debe prestar atención tanto al movimiento de los labios como a la postura que adoptan la lengua y los dientes, entre otros aspectos ya comentados anteriormente. No obstante, antes de abordar esta técnica es necesario conocer dos conceptos fundamentales: el fonema y el visema. El primero de ellos se define como la unidad mínima de sonido con la que podemos distinguir una palabra de otra dentro de un lenguaje. Por otro lado, el visema[2] se asocia a la representación visual de un fonema, en concreto, mediante las expresiones faciales. Sin embargo, no existe una correspondencia directa entre ambos. Debido a esto, nos enfrentamos a distintos problemas que entorpecen la compresión del discurso cuando no se dan las condiciones óptimas para oír al emisor. Entre estos problemas destacamos aquellos fonemas que pertenezcan a un mismo visema, tal y como ocurre con los fonemas /p/ y /b/, así como los fonemas producidos desde la garganta y que, por lo tanto, no pueden ser vistos e interpretados por el receptor del mensaje.

De forma general, los principales desafíos son las ambigüedades visuales que pueden inducir a errores y las condiciones en las que se obtiene el vídeo. Esta última característica se encuentra altamente influenciada por la calidad del vídeo, las condiciones lumínicas o los movimientos de la cabeza del emisor. Esta dependencia genera un agravante respecto a los resultados esperados si los comparamos con aquellos sistemas que reconocen el habla en base al audio, tal y como se comentará a lo largo de toda la memoria.

1.1. Motivación

El principal motivo de este trabajo es la importancia que está tomando la inteligencia artificial tanto en el ámbito laboral como social. Este proyecto aporta una experiencia sobre el alumno

[1] https://en.wikipedia.org/wiki/Lip_reading
[2] https://en.wikipedia.org/wiki/Viseme

que puede servir de gran utilidad en su futuro desarrollo. Además, los aspectos técnicos relacionados con el *machine learning* resultan de gran interés, ya que sus fundamentos probabilísticos y sus aplicaciones intentan simular el comportamiento del cerebro humano en ciertas situaciones. Por otra parte, esta rama tecnológica conlleva tratar con proyectos que involucran a equipos de trabajo multidisciplinares. Es por ello que la inteligencia artificial abarca un amplio espectro de aplicaciones y todo apunta a que va a ser partícipe del desarrollo e innovación de nuestra sociedad.

Centrando la atención sobre la lectura de labios a partir de imágenes de vídeo, destacamos como motivo primordial la ayuda que supondría un sistema capaz de leer los labios a aquellas personas con problemas de audición o bien sordera. Aparte, este tipo de sistema sería útil en situaciones donde el audio se viese comprometido. Por otro lado, esta tecnología podría aplicarse para la protección de información sensible, de manera que podamos introducir contraseñas en nuestro dispositivo móvil simplemente vocalizando frente a la cámara en caso de que el teclado no funcionase.

1.2. Objetivos

Este trabajo tiene como principal propósito la construcción de un sistema capaz de leer los labios a partir de imágenes de vídeo, tal y como se ha comentado en secciones anteriores. Para facilitar tanto su realización como su seguimiento, se proponen los siguientes subobjetivos:

- Adquirir experiencia con todo el proceso que conlleva un sistema de *machine learning*, tomando parte en la elaboración de todas sus fases que describiremos posteriormente.
- Construir un *dataset* enfocado a la tarea de leer labios a partir del canal visual.
- Diseñar un sistema preliminar de lectura de labios en imágenes de vídeo.

1.3. Estructura

Respecto a la estructura sobre la que se apoya el trabajo, cabe destacar que en primera instancia se incluye el estado del arte en el **capítulo 2**, tras la introducción presente en el **capítulo 1**. Con este apartado se pretende dar a conocer los sistemas de reconocimiento del habla, así como su evolución a lo largo de la historia. Posteriormente, el **capítulo 3** de la memoria se centra en describir el proceso de construcción de un sistema de lectura de labios a partir de imágenes de vídeo. Este proceso está compuesto por numerosas fases que dividirán el capítulo en los apartados correspondientes. Cada uno de estos apartados explica los distintos aspectos vinculados a la fase en cuestión, así como la tecnología empleada. Además, en los casos en que sea necesario, se añadirán secciones a modo de introducción para facilitar la comprensión por parte del lector. Por otro lado, el **capítulo 4** contiene las conclusiones, donde se pone de manifiesto si se han alcanzado los distintos objetivos planteados. Tras las conclusiones se presenta el **capítulo 5** que describirá la relación entre los estudios cursados y el trabajo realizado. En cuanto al **capítulo 6,** se incluyen las posibles vías de desarrollo que se puedan llevar a cabo sobre el trabajo en un futuro, además de plantearse ideas, posibles alternativas o soluciones. Por último, se incluyen las referencias bibliográficas tomadas en cuenta para la realización del proyecto.

2. Estado del arte

Antes de abordar el trabajo es necesario tener una cierta noción del contexto histórico por el que se ha ido desarrollando la evolución de los sistemas automáticos de reconocimiento del habla, tomando como fuente principal el artículo [3]. Para ello, comenzaremos con una breve introducción al aprendizaje automático, también conocido como *machine learning*. Tras esta sección de carácter general, centraremos la explicación desde un punto de vista más cercano a aquellos sistemas cuyo principal objetivo es similar o coincide con el propuesto para este proyecto. De esta manera, podremos comprender mejor el entorno o estado en el que nos encontramos, así como justificar las alternativas que hemos tomado a lo largo del trabajo.

2.1. Introducción al aprendizaje automático

En la década de los años cuarenta se empezaron a desarrollar de forma paralela dos enfoques principales para la disciplina que hoy en día se conoce como Sistemas Inteligentes. El primero de ellos es la Inteligencia Artificial. Este enfoque se centra en los aspectos más cognitivos, siempre relacionado con la lógica, el conocimiento y su procesamiento. Por otro lado, está el Reconocimiento de Formas, que se ocupa de aspectos más perceptivos, como puedan ser la detección de objetos o el habla.

Dentro de las ciencias de la computación, alrededor de la década de los 80 o 90, surge el *machine learning* definido como la combinación de la Inteligencia Artificial con el Reconocimiento de Formas, entre otras disciplinas. Su objetivo consiste en desarrollar técnicas que permitan a los computadores aprender, tomar decisiones o identificar patrones a partir de un conjunto de datos y con la mínima intervención humana posible[3]. A menudo, el campo abarcado por el *machine learning* se ve solapado por la estadística inferencial, ya que ambas disciplinas se fundamentan en el análisis de datos. Por lo tanto, no solo interesa el aprendizaje de modelos propiamente dicho, sino todo el proceso a la hora de resolver problemas, el cual está basado en una aplicación rigurosa de la teoría de la decisión estadística.

Respecto al aprendizaje, se asume la existencia de un conjunto de datos de entrenamiento donde típicamente se dispone de los datos de entrada $x \epsilon X$ y, opcionalmente, su salida esperada $y \epsilon Y$. El objetivo es obtener un modelo que defina una función $f : X \rightarrow Y$ con la que generalizar los datos adecuadamente. Cuando decimos generalizar lo entendemos como predecir la salida a partir de nuevos datos de entrada diferentes a los que conformaron el entrenamiento. Se trataría entonces de un proceso de aprendizaje inductivo, porque el sistema dispone *a priori* de un escaso conocimiento respecto al problema a resolver. El sistema no es programado para realizar la tarea, sino que deberá construir sus modelos mediante la observación de ejemplos o muestras de aprendizaje como las comentadas anteriormente. Principalmente, se distinguen dos tipos de aprendizaje en función de la naturaleza del conjunto de entrenamiento:

- Aprendizaje supervisado: se relaciona con aquellos casos en el que el conjunto de entrenamiento dispone de la información completa, es decir, tanto de los datos de entrada como los de salida. Por ejemplo, podría entrenarse un sistema para que pudiera discernir si un correo electrónico es *spam* o no. Para ello, cada muestra de aprendizaje

[3] https://es.wikipedia.org/wiki/Aprendizaje_automático

consistiría en una tupla, donde la entrada sería el cuerpo del correo y la salida un indicador de si el correo en cuestión es o no *spam*.

- Aprendizaje no supervisado: cuando los datos de entrenamiento solo disponen de la información de entrada. A diferencia del caso anterior, el sistema no conoce qué información es satisfactoria o no para el objetivo del aprendizaje. Debido a esto, deberá extraer patrones e información para la agrupación de los datos en función de sus atributos. Sin embargo, este tipo de aprendizaje requiere de la interpretación de un ser humano para darle utilidad. Un ejemplo, sería proporcionar imágenes para que el sistema aprendiese a clasificarlas según si es un coche, un perro o un balón.

Por otro lado, existen dos vertientes dentro del *machine learning*. Dependiendo del dominio en el que se definan tanto los datos de entrada X como los de salida Y [2], distinguimos sistemas enfocados en:

- Regresión: aquel *dataset* donde tanto los datos de entrada como los de salida pertenecen a dominios arbitrarios. El sistema entrenado sería un modelo predictor, típicamente bajo el dominio de los números reales. A modo de ejemplo, podría entrenarse un sistema proporcionándole un gran conjunto de datos relacionados con el nivel de agua de un pantano, de modo que consigamos un sistema capaz de realizar predicciones sobre el nivel de agua que pueda alcanzarse.
- Clasificación: los datos de entrada pertenecerán a un dominio arbitrario pero los de salida serán de un conjunto finito, generalmente pequeño, de C elementos denominados clases. De esta manera, una vez el sistema se encuentre entrenado proveerá al usuario una clasificación de la nueva entrada que le proporcione.

Respecto a nuestro sistema, consideramos el aprendizaje como supervisado, ya que dispondremos tanto de la entrada visual codificada en vectores de características como de la transcripción del discurso asociada; es decir, el sistema dispone de la solución para su entrenamiento. Todo este conjunto de muestras de aprendizaje será detallado en apartados posteriores. Por otro lado, nuestro sistema no está ceñido a ninguna de las vertientes comentadas anteriormente sino que, más bien, se consideraría como un sistema de interpretación donde se procesan las distintas características visuales extraídas con el objetivo de conformar un objeto complejo como lo es una oración.

2.2. Datasets

Comenzaremos revisando la investigación en torno a los conjuntos de datos audiovisuales, que han acompañado al desarrollo de los Sistemas de Reconocimiento Automático del Habla (SRAH) enfocados a la lectura de labios. En la literatura encontramos diversos tipos de *datasets* que difieren en aspectos como el número de *speakers*, la talla del vocabulario involucrado, ajustes de grabación o la duración total [3].

En sus inicios, en la década de los noventa, los *datasets* se centraban en reconocimientos simples y específicos con un vocabulario restringido, como es el caso del reconocimiento del alfabeto o dígitos. En el ámbito del reconocimiento del alfabeto, destacamos AVLetters (1998) [29] como uno de los *corpus* más empleados. Por otro lado, el reconocimiento de dígitos está representado por el *dataset* XM2VTS [30], siendo una de las bases de datos más grandes con 295 participantes. Posteriormente, surgieron nuevos conjuntos de datos para suplir ciertas

debilidades de los anteriores *datasets*, o bien modificando ciertos aspectos de los comentados anteriormente. Este tipo de *corpus* han sido muy populares porque han permitido tratar con SRAH bajo un entorno controlado, siendo útil para analizar la efectividad de los algoritmos en las etapas tempranas de su desarrollo.

Sin embargo, los sistemas entrenados con estas muestras de aprendizaje distan mucho de ser útiles a la hora de construir modelos robustos y enfocados a aplicaciones realistas. Esto se debe al bajo número de temáticas tratadas, al vocabulario restringido y a la limitada cantidad de datos recogidos. Por ello, se ha impulsado en los últimos años la construcción de *datasets* de mayor magnitud para poder entrenar sistemas cuyo objetivo sea el de reconocer el habla continua. Además, con el desarrollo de técnicas de *Deep Learning* (DL) se ha hecho necesaria la adquisición de estos conjuntos de datos de gran escala que permitan el entrenamiento de estos nuevos sistemas. Puesto que a pesar de que las técnicas de DL han aportado grandes avances relacionados con la computación visual y las tareas de clasificación, esto sólo es posible si los *datasets* que conforman el entrenamiento son apropiados y disponen de una gran cantidad de información. Como ejemplos, destacamos el *corpus* GRID [31] (el cual contiene un gran número de elocuciones pero oraciones similares y limitadas, es decir, siguiendo un esquema acordado con anterioridad) y el RM-3000 [32] (que presenta un único *speaker* pero con un enorme vocabulario).

Por otro lado, recientes esfuerzos han sido encauzados hacia *datasets* de gran escala recopilados a partir de programas de televisión con el objetivo de proporcionar un amplio vocabulario bajo condiciones realistas. El *corpus* LRS [33] constituye un ejemplo de este último tipo de *datasets* explicado, con más de 100.000 elocuciones y sobre cien *speakers*. Además, para aportar más robustez a nuestro sistema, se pueden recoger no sólo planos frontales de la persona que habla, sino que podrían incluirse escenas donde el *speaker* presente una rotación de la cabeza o sea grabado desde otros ángulos, conformando así un *dataset* multivista, dado que así se reflejan escenarios más realistas a la hora de entrenar nuestro sistema. Además, algunos estudios con humanos han demostrado que desde ángulos ligeramente distintos al plano frontal se presentan mejores condiciones para la comprensión del mensaje a través de la lectura de labios. Esto ha llevado al desarrollo de *datasets* como, por ejemplo, el *dataset* AVICAR [34], grabado desde un coche usando cuatro cámaras distribuidas por el vehículo con las que se obtuvieron cuatro planos semi-frontales del *speaker*.

En conclusión, nuestro *dataset* será recopilado a partir de un programa de televisión, donde se tratarán planos frontales, aunque a menudo el *speaker* varíe la posición, rotación o ángulo mientras emite su discurso. Esto aportará, en principio, robustez a nuestro sistema, tal y como se ha comentado anteriormente. Además, estará enfocada al reconocimiento del habla continua, lo que supone un gran desafío. Por último, el idioma escogido ha sido el español, ya que este idioma no es muy común en la literatura que hemos investigado, donde sólo destacan los *corpus* AV@CAR [35] y VLRF [36]. Como ha podido observarse, la precisión o la calidad de los resultados generados por nuestro sistema dependerá en gran medida del conjunto de datos aportados para su entrenamiento. Por ello, más adelante, se detallarán las características y aspectos de nuestro conjunto de datos.

2.3. Sistemas de Reconocimiento Automático del Habla

El esquema que refleja el desarrollo de los SRAH se encuentra regido por una evolución constante, pasando por distintas etapas hasta llegar a la actualidad. Por ello, en esta sección se describirán las fases por las que han pasado este tipo de sistemas desde sus inicios, distinguiendo entre las aproximaciones tradicionales y aquellas que hagan uso de técnicas de *Deep Learning*. Respecto a éstas últimas, se consideran la causa del incremento sustancial de numerosos artículos científicos relacionados con la decodificación del habla, así como de la disponibilidad de *datasets* de gran escala. No obstante, todos estos sistemas se encuentran guiados por la misma estructura general de desarrollo: detección de la región de la boca, extracción de características y clasificación.

En sus orígenes, los SRAH se basaban únicamente en la información acústica, dado que las señales de audio contienen más información que las de vídeo respecto a nuestro objetivo. Hoy en día, este tipo de modelos son poderosos sistemas capaces de entender el lenguaje hablado con índices muy altos de reconocimiento cuando la señal acústica no está corrupta. Sin embargo, en las situaciones en las que esta señal se encuentra degradada decaen las prestaciones de este tipo de sistemas. Es entonces cuando aparece la necesidad de confiar en la información proveniente del canal visual, tomando de base el proceso subconsciente que realiza nuestro cerebro combinando la información de ambos canales para comprender mejor a su interlocutor, sobre todo en ambientes o entornos ruidosos. Este aspecto propulsó la investigación de los denominados Sistemas Audiovisuales de Reconocimiento Automático del Habla (SARAH). Los SARAH intentan equilibrar la contribución del audio y del vídeo para desarrollar sistemas que sean robustos ante adversidades acústicas, aportando una mejora significativa de las prestaciones en dichas condiciones.

Por otro lado, en las últimas décadas ha habido un incremento en el interés de decodificar el habla usando exclusivamente la información procedente del canal visual, mimetizando la capacidad humana de leer los labios. Como ya conocemos, esta rama de los SRAH se enfrenta a numerosos desafíos ya descritos en otras secciones. En este apartado se describirán las dos etapas presentes en la evolución de este último tipo de sistema, distinguiendo entre las dos aproximaciones mencionadas al inicio. Además, deberá destacarse tanto la evolución de estos sistemas como la de los *datasets* relacionados, ya que han compartido un proceso en que ambas partes han colaborado a mejorar y alcanzar el actual estado del arte.

En conclusión, tal y como se observará en los sucesivos apartados, nuestro sistema presenta nuevos aspectos como puede ser combinar un Modelo Oculto de Markov con la técnica de extracción de características LBP-TOP (Local Binary Patterns extracted from Three Orthogonal Planes) con el objetivo de cubrir un ámbito de la literatura. Además, presenta una tarea de interpretación, dejando de lado la clasificación en un número acotado de frases o palabras. Por otro lado, cabe destacar la influencia que presenta el *dataset* recopilado sobre los ratios de precisión del sistema presentado para este trabajo de final de grado.

2.2.1. Aproximaciones tradicionales

La mayoría de sistemas durante este período se conformaron a través de Modelos Ocultos de Márkov (HMM, sus siglas en inglés), los cuales se describen en el apartado 3.4.1, puesto que es el sistema escogido para nuestro proyecto. A continuación estarían las Máquinas de Vectores Soporte (SVM, sus siglas en inglés), ampliamente utilizadas.

Al igual que sucedía con los *datasets*, en sus inicios los sistemas estaban enfocados a tareas sencillas, como el reconocimiento del alfabeto y dígitos. En estos casos, observamos que gran parte de estos sistemas usa técnicas de extracción de características basadas en transformaciones de imágenes o en modelos de apariencia, cuyas diferencias se detallan en el apartado 3.3.1. Entre estas técnicas destacamos AAM (Active Appearance Model) o combinación de DCT (Discrete Cosine Transform) junto otras técnicas como LDA (Linear Discriminant Analysis) o PCA (Principal Component Analysis). A modo de ejemplo, presentaremos las características y prestaciones presentes en dos arquitecturas:

- La primera de ellas emplea un HMM, extrayendo las características del *dataset* CUAVE mediante la técnica LDA. En sus resultados informó de una precisión del 60.00% [12], expresado en WRR (Word Recognition Rates), mientras que si se utilizaba AAM's para la extracción de características se obtenía un 83.00% [13]. Por otro lado, en el caso de escoger el *dataset* XM2VTS junto a DCT para las características visuales se lograba el mejor WRR, con un 87.89% de precisión [14].
- La segunda arquitectura presenta una SVM para el *dataset* CUAVE, usando para la extracción de características una combinación de HOG (Histogram of Oriented Gradients) y MBH (Motion Boundary Histograms) obteniendo un 70.10% WRR [15].

En conclusión, los modelos mayoritariamente empleados han sido los HMM's, alcanzando su mayor precisión junto a las técnicas basadas en AAM's, aunque DCT haya sido el método más implementado en la literatura.

Sin embargo, como ya se ha comentado previamente en el apartado referente a los *datasets*, se produjo un cambio que encauzó la evolución de los SRAH hacia tareas más complejas dedicadas al reconocimiento de oraciones. No obstante, las técnicas para extraer las características visuales más usadas continúan siendo PCA, DCT y AAM. Cabe destacar que aunque estas técnicas no presentan las frecuencias más altas por ellas mismas, aparecen en múltiples ocasiones combinadas con otras. Si comparamos con el reconocimiento de dígitos y letras, observamos la presencia de nuevas técnicas para extraer características como, por ejemplo, LBP-TOP, que se encuentra comentado en el apartado 3.3.1. puesto que se trata de la técnica seleccionada para nuestro trabajo. Otros ejemplos son SDF (Shape Difference Feature) o STLF (Spatio-Temporal Lip Feature). Por otro lado, en cuanto a los clasificadores observamos que los HMM's siguen siendo el método de clasificación más usado, aunque existe un incremento considerable del uso de SVM's, ya mencionadas previamente. Respecto a las prestaciones de estos sistemas ante tales tareas, se proporcionan los ejemplos destacados a continuación:

- En la literatura, encontramos dos combinaciones en las que se emplean un clasificador SVM junto a la técnica LBP-TOP para extraer las características visuales a partir del *dataset* OuluVS, informando de la precisión de los resultados entre los valores de 62.40% [4] y de 81.30% [5] de WRR. La diferencia reside en el tratamiento al extraer

las características, siendo el modelado temporal un factor clave a la hora de mejorar las prestaciones.

- Por otro lado, enfocados hacia el *dataset* OuluVS2 fueron entrenados dos sistemas basados en HMM's. El primero de ellos combinó DCT y PCA, obteniendo un 63.00% de WRR, mientras que combinando DCT y HiLDA se alcanzó un 74.00% de WRR [16].

En conclusión, los sistemas dirigidos al reconocimiento de palabras u oraciones han mostrado un descenso de las prestaciones en gran parte de los casos, ya que la tarea adquiere un nuevo nivel de dificultad. Por lo tanto, es complicado determinar cuál podría ser el esquema que presente el mayor ratio de precisión, ya que existe una gran variedad y disparidad de experimentos. Sin embargo, es necesario destacar que en el caso del *dataset* en español AV@CAR se obtuvo uno de los índices más altos, respecto a precisión. Esto es debido al mapeado realizado en el ámbito de los fonemas y visemas, así como el tamaño del conjunto de datos que provoca la aportación de un amplio vocabulario.

2.2.2. Aproximaciones *Deep Learning*

Como se ha ido comentando, ha habido una mejora significativa en las prestaciones de los SRAH en los últimos años gracias a los avances de las Redes Neuronales Profundas (DNN, *Deep Neural Networks*) y la disponibilidad de *datasets* de gran escala. En sus inicios, fueron consideradas como extractores de características en combinación con HMM's. Sin embargo, la tendencia encauza el desarrollo hacia sistemas basados puramente en técnicas de *Deep Learning,* conocidos como *end-to-end* DNN's. Entre estas arquitecturas destacamos aquellas basadas en la combinación de *Convolutional Neural Networks*[4] (CNN) y *Long-Short Term Memory*[5] (LSTM). La primera de ellas es un tipo de red neuronal artificial donde las neuronas se organizan en campos perceptivos de una manera muy similar a las neuronas en la corteza visual primaria de un cerebro biológico. Básicamente, se trata de una modificación del perceptrón multicapa, pero como su aplicación se realiza mediante matrices bidimensionales, son muy efectivas en tareas enfocadas en la visión por computador o la clasificación a partir de imágenes. La segunda corresponde con un tipo de Red Neuronal Recurrente que se aprovechará de su distribución y arquitectura para almacenar información referente al contexto. Es por ello que este tipo de redes son adecuadas para procesar, clasificar y hacer predicciones sobre datos que dependan del eje temporal, presentando numerosas ventajas sobre modelos ya conocidos como pueden ser los HMM's. Por otro lado, encontramos arquitecturas basadas en la implementación de LSTM's junto a redes neuronales *feed-forward*, las cuales, a diferencia de las redes neuronales recurrentes, no presentan ciclos entre las conexiones de las neuronas. Dentro de este ámbito distinguimos clasificaciones tanto a nivel de oración como de palabra, mejorando distintos aspectos respecto a las aproximaciones tradicionales.

Si comparamos los sistemas tradicionales con las arquitecturas *Deep Learning* observamos que esta última aproximación proporciona una mejora sustancial en términos de precisión y prestaciones. Por ejemplo, para el *corpus* GRID, varias arquitecturas DL superaron considerablemente al mejor sistema tradicional con hasta un 40% de mejora, llegando a alcanzar algunos un 97% de precisión. Sin embargo, esto no puede ser extrapolado directamente al reconocimiento del habla continua (como es nuestro caso) debido a que estos sistemas

4 https://es.wikipedia.org/wiki/Redes_neuronales_convolucionales
5 https://en.wikipedia.org/wiki/ Long_short-term_memory

devuelven unos resultados restringidos a un número predefinido de posibles clases, en contraste con la lectura de labios del habla natural, donde debe decodificarse cualquier palabra del diccionario y formar frases cuyas palabras no están separadas por límites temporales fijos. Por esta razón han aparecido intentos de decodificar el habla continúa focalizando el entrenamiento a un nivel de carácter o fonema. Dentro de esta tendencia se ha conseguido un 50% WRR como máximo [17].

Como resultado, encontramos que aquellas aproximaciones basadas en arquitecturas *Deep Learning* obtienen resultados similares a las tradicionales en cuanto a tareas sencillas se refiere, mientras que aportan una mejora significativa cuando se aplican a tareas complejas, como, por ejemplo, el reconocimiento de oraciones. Debido a esto, las arquitecturas basadas en *Deep Learning* están dominando la evolución de este tipo de sistemas. Sin embargo, el modelado temporal sigue siendo un problema abierto, aunque se ha intentado solventar mediante el uso de Redes Neuronales Recurrentes como las ya comentadas LSTM's. Por ello, hemos observado el uso de redes bidireccionales, que aunque supongan un mayor coste computacional son ampliamente utilizadas en el reconocimiento de voz por su capacidad de modelar el contexto, siendo útiles para resolver las ambigüedades visuales. De hecho, la investigación gravita hacia técnicas que permitan una mayor comprensión, modelado e interpretabilidad del contexto retenido.

3. Lectura de labios en imágenes de vídeo

Este capítulo aborda las distintas fases de las que se compone todo el proceso de construcción de un sistema capaz de decodificar el habla a partir de imágenes de vídeo, es decir, sin la necesidad de información que provenga del medio acústico. Esto se debe al gran interés que ha adquirido este objetivo en la literatura, tal y como se ha comentado en el estado del arte. De esta manera, se logrará mimetizar la capacidad humana a la hora de leer los labios del emisor. Esto conlleva una serie de desafíos, tal y como se introdujo al principio de esta memoria. Entre estos desafíos destacamos las ambigüedades visuales presentes a nivel de palabra debido a los *homofemas*, es decir, aquellos fonemas que pueden confundirse con otros ya que producen las mismas o similares trazas bucales. Por otro lado, las condiciones en las que se obtiene el vídeo son otra fuente de dificultad. Esta última característica se encuentra altamente influenciada por la calidad con la que éste fue grabado, las condiciones lumínicas o los movimientos de la cabeza del emisor. Esta dependencia genera un agravante respecto a los resultados esperados si los comparamos con aquellos sistemas que reconocen el habla en base al audio o a una combinación de ambas informaciones, tal y como se ha comentado anteriormente.

Para facilitar la comprensión de todo el proceso, el capítulo se divide en distintos apartados en función de las fases en las que se divide la construcción del sistema. En cada uno de estos apartados se detallará el procedimiento llevado a cabo, así como la tecnología empleada en cada caso. Todo este proceso se puede observar de manera sintetizada en la Figura 1, donde podemos esbozar tres fases: obtención del *dataset*, así como el preproceso que conlleva, entrenamiento del sistema y el *test*. No obstante, en primera instancia se dedica una sección entera a conocer los detalles del *dataset* recopilado para este trabajo.

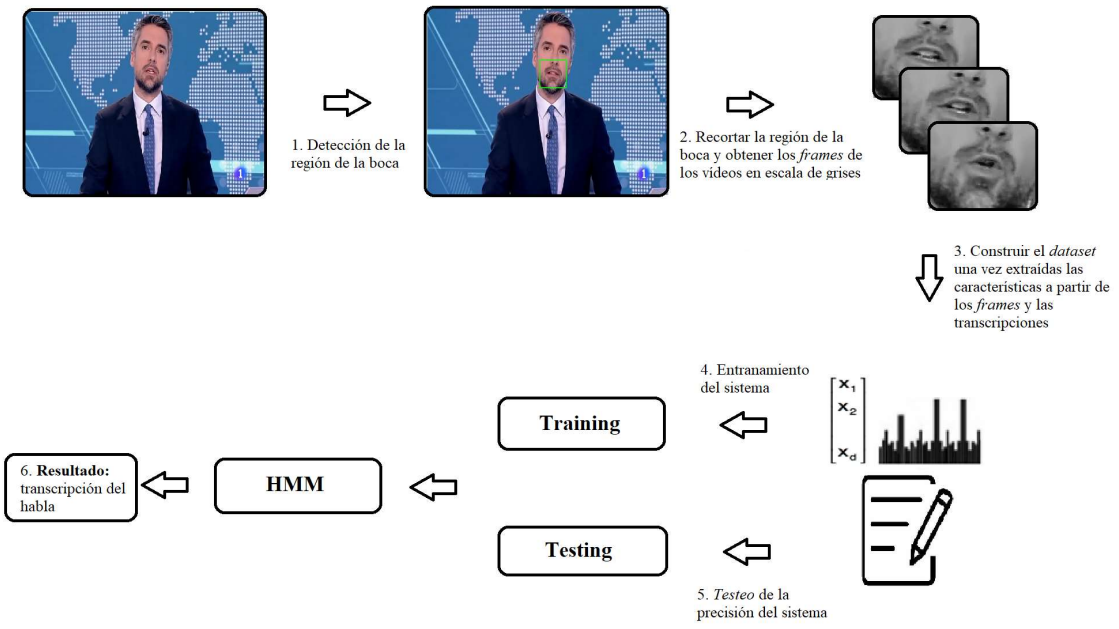

Figura 1. Esquema general de las fases propuestas para la construcción del sistema

3.1. El *dataset* de nuestro sistema

El *dataset* propuesto para la construcción del sistema y que ya ha sido comentado con anterioridad a lo largo de la memoria se creará a partir de un programa de televisión. Las razones que han impulsado esta decisión son el aumento de la robustez del sistema, el empleo de un amplio vocabulario y la aportación de escenarios realistas, dado que el interlocutor no aparece de una manera estática sino que puede realizar movimientos o rotaciones de la cabeza durante la emisión del discurso.

El *corpus* estará formado por planos frontales de 4 presentadores del telediario de Radio Televisión Española (RTVE), constituyendo una compilación de 3.800 segundos aproximadamente junto a las transcripciones correspondientes. Con el objetivo de contribuir a cierta variedad, se han recogido planos de dos varones, uno de ellos con barba, y dos mujeres a partir de los telediarios comprendidos entre mayo de 2018 y abril de 2019, tal y como se observa en la Figura 2. Para ello, se ha hecho uso de las herramientas de edición de vídeo disponibles en Windows y así poder recortar los planos frontales de los presentadores en cuestión. Es necesario evitar aquellos planos donde coincidan más de dos personas o bien haya un entorno abarrotado de detalles que puedan inducir a errores, como se explicará más adelante. Por otro lado, se observa una novedad respecto al idioma escogido para el *dataset*, ya que el español no es muy común en la literatura. Además, coexisten planos donde la distancia de la cámara al presentador difiere en cierta medida, aportando situaciones realistas que servirán a la hora de aplicarse la funcionalidad del sistema.

Este *corpus* se encontrará estructurado en función de qué presentador se trate. Cada presentador dispondrá de numerosas muestras de aprendizaje donde cada una de éstas estará formada por el plano frontal en concreto y la transcripción asociada de lo que está diciendo el presentador. Cabe destacar que esta transcripción ha sido objeto de un preproceso donde se han convertido a minúsculas todos los caracteres y se han eliminado todos los signos de puntuación. De esta manera, conseguimos facilitar la tarea al sistema, ya que si no fuera así las palabras “Hola”, “hola” y “hola,” serían consideradas por el sistema como palabras distintas. Por otro lado, tal y como se detallará en sucesivos apartados, la información visual será codificada con el objetivo de que pueda ser interpretada por el sistema.

Figura 2. Los cuatro presentadores que componen el *dataset*

3.2. Detección de la región de la boca

Una vez recopilados los planos frontales de los presentadores del telediario de RTVE requeridos para cubrir los segundos indicados en el apartado anterior, junto a sus respectivas transcripciones, es necesario extraer aquella área donde reside la información más relevante para alcanzar nuestro objetivo: la región de la boca.

En primer lugar, sobre los datos en formato de vídeo se aplicará un *software* encargado de detectar la zona bucal [6], al que se le han aplicado ciertas modificaciones en vistas a nuestro objetivo de obtener una secuencia de *frames* de la región bucal que represente el discurso en su totalidad. Para ello, se transforman los vídeos en una secuencia de *frames* mediante el uso de la biblioteca *skvideo*. En nuestro caso, la frecuencia con la que se tratan los vídeos es de 30 *frames* por segundo. Posteriormente, gracias a las bibliotecas OpenCV y *Dlib*, que comentaremos en el epígrafe siguiente, conseguimos detectar las caras en primera instancia. Tras comprobar que se ha detectado una cara en el *frame*, se emplea un predictor para determinar la región que comprende la boca del interlocutor. Por lo tanto, en caso de que en cada *frame* se encuentre una única cara y que sea posible extraer la región de la boca, se irá construyendo la secuencia de *frames* sobre la que nos apoyaremos para extraer las características del movimiento labial. Cabe destacar que los distintos fotogramas serán obtenidos en escala de grises, tal y como se muestran en la Figura 4. Sin embargo, para obtener el área de la región bocal son necesarios unos pasos previos. El primero de estos pasos consiste en definir tanto el centro de la boca así como sus extremos, gracias al predictor mencionado anteriormente. De esta forma, podemos determinar las dimensiones de un rectángulo que englobe la zona deseada. Esta área descrita puede visualizarse delimitada por un rectángulo verde sobre el vídeo original, tal y como se observa en la Figura 3.

Figura 3. Detección de la región de la boca

Después de este paso, sólo resta recortar el área calculada y almacenar los distintos *frames* de la región de la boca que componen el vídeo en su totalidad como resultado. Cada uno de estos *frames* se encontrará en escala de grises y con una relación de aspecto de 1.8, lo que nos evitará cómputo en fases posteriores. Aunque todos los *frames* tengan las mismas proporciones, algunos de éstos mostrarán la región de interés desde una distancia mayor y, por lo tanto, abarcando un área más extensa de las expresiones faciales. Esto se debe a que algunos vídeos

del telediario son tomados desde una distancia mayor, como se ha ido comentando a lo largo de la memoria, pero, como ya avanzamos en el estado del arte, estos detalles aportan al sistema un escenario realista sobre el que entrenar el modelo.

En otras palabras, nuestro *dataset* está compuesto en estos momentos por muestras de aprendizaje, donde cada una de ellas se corresponde con una secuencia de *frames* de la región de la boca, en escala de grises y con las mismas proporciones, junto a la transcripción asociada.

Figura 4. Secuencia de *frames* de la región de la boca en escala de grises

Sin embargo, nos hemos enfrentado a distintos problemas durante la recolección de los datos. Debido al *software* empleado, no se podían seleccionar escenas donde coincidiesen más de un *speaker*, ya que el programa no estaba diseñado para distinguir entre varias caras. Esto provocó errores en algunos vídeos donde el croma del plató presentaba una imagen abarrotada de gente, o bien con la cara de alguna persona. Entonces algunos *frames* recogían la cara de una persona que no era el presentador del telediario. Por otro lado, algunos rasgos se confundían con las comisuras o la fisonomía de la boca, como podría ser el dorso de la mano del presentador. Todos estos errores comentados se pueden observar en la Figura 5.

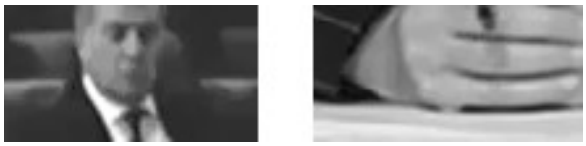

Figura 5. Errores a la hora de detectar la región de la boca

Tras descartar estos vídeos defectuosos, se completó el *dataset* con todos los datos preprocesados para la extracción de características.

En cuanto a la tecnología empleada, se han nombrado distintas bibliotecas que se han utilizado para la detección de la región de la boca. Estas técnicas presentes en el *software* deben ser descritas brevemente para comprender mejor cómo se ha realizado el proceso explicado con anterioridad a través de un *script* programado con el lenguaje *Python* [6]. Sobre éste se han aplicado ciertas modificaciones con el objetivo de amoldarlo a nuestras necesidades. Es por ello que el programa devuelve la secuencia de *frames* requerida para la fase de extracción de características.

La primera tecnología es OpenCV[6] (Open Source Computer Vision Library), la cual consiste en una biblioteca *software* de código abierto relacionada con la visión por computador y el *machine learning*. OpenCV fue creada con el objetivo de proporcionar un entorno de desarrollo

[6] https://opencv.org/about/

enfocado a aplicaciones de visión por computador y para acelerar el uso de la percepción automática en productos comerciales. La biblioteca dispone de más de 2500 algoritmos optimizados. Estos algoritmos abarcan la detección y reconocimiento de caras, identificar objetos, clasificar acciones humanas en vídeos, seguir el movimiento de los ojos, etc. Concretamente, en nuestro proyecto ha permitido el tratado de imágenes, permitiéndonos recortar la región de la boca además de convertirla en escala de grises, consiguiendo de esta manera el conjunto de *frames* deseados.

Sin embargo, en nuestro *dataset* se dispone de la información visual en formato de vídeo. Esto nos conduce a utilizar la biblioteca de *Python* Scikit-video o *skvideo,* dedicada al procesamiento de vídeos. En nuestro caso, se ha empleado *FFmpegReader* para realizar la lectura de los vídeos, recibidos como parámetro de entrada, para así obtener los *frames*. De esta manera, podemos conocer el número de *frames* que componen el vídeo, a la vez que algunos atributos como la altura y anchura de cada uno de éstos y el número de canales por píxel. Gracias a esta tecnología hemos podido desglosar el vídeo en sus *frames* y así alcanzar nuestro objetivo de obtener una secuencia de fotogramas para la extracción de características.

Una vez ya podemos trabajar individualmente con cada uno de los *frames* que forman la entrada del programa y poder tratarlos mediante el empleo de las bibliotecas ya descritas, toma lugar la principal tecnología, conocida como *Dlib*. Esta tecnología es una biblioteca multiplataforma de código libre escrita en el lenguaje de programación C++; es decir, se trata de una colección de distintas componentes independientes de *software*. Además, es muy útil tanto para investigación como para proyectos comerciales. *Dlib* incluye herramientas y entornos enfocados a las estructuras de datos, álgebra lineal, *machine learning* y procesado de imágenes, entre otros [7]. Más concretamente, en nuestro trabajo haremos uso de la biblioteca *Dlib* en primer lugar para detectar la cara del presentador en cuestión mediante el método *get_frontal_face_detector()*. Posteriormente, se dispondrá de un modelo entrenado para predecir la zona donde se ubica la boca del interlocutor. Este modelo predictor se entrena en base a los 68 puntos de referencia (*landmarks*) de la cara humana, los cuales pueden observarse en la Figura 6, siendo los puntos comprendidos entre 49 y 68 inclusive aquellos que nos permitirán extraer la zona de la boca.

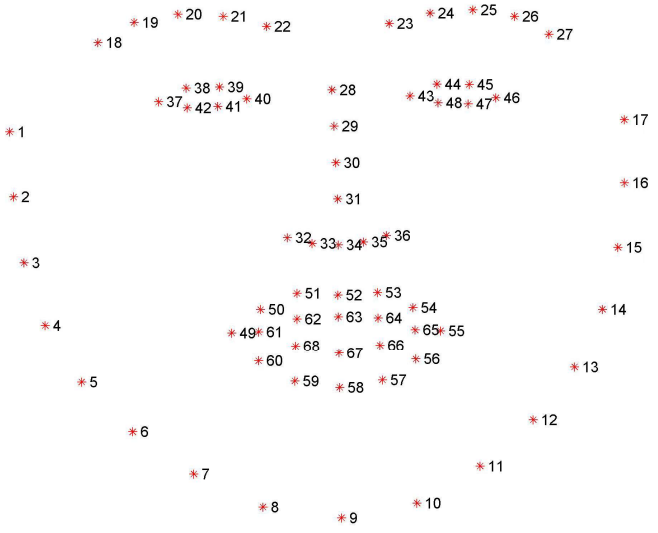

Figura 6. Los 68 puntos de referencia faciales para entrenar el predictor

3.3. Extracción de características visuales del movimiento labial

Una vez realizado todo el preproceso de los datos descrito hasta el momento, es necesario aplicar el último paso con el que completaremos nuestro *dataset*. Estamos hablando de aplicar un método que nos permita extraer las características más relevantes frente al reconocimiento automático del habla a partir de los *frames* obtenidos anteriormente. En primer lugar, es necesario comprender que esta fase es un elemento fundamental para la construcción de nuestro sistema. Tal y como se avanzó en el estado del arte, hemos escogido la técnica LBP-TOP [8], la cual es empleada en el artículo [4] y ofertada como *software* libre para el lenguaje de programación *Matlab*[7]. Sin embargo, como el código fuente de esta técnica estaba enfocado al reconocimiento de expresiones faciales, ha sido necesario realizar ciertas modificaciones que han estado guiadas por el artículo mencionado. De esta manera, tras la aplicación del *software*, obtendremos un *dataset* dividido en los cuatro presentadores que ya hemos comentado, donde cada uno de éstos dispondrá de un número de discursos. A su vez, cada discurso estará codificado en un par compuesto por la transcripción del mensaje (preprocesada tal y como se indicó en la sección 3.1) y un vector de características construido por la técnica escogida, tal y como se esboza en la Figura 7. Dicha técnica será descrita posteriormente con mayor detalle en el apartado 3.3.1.

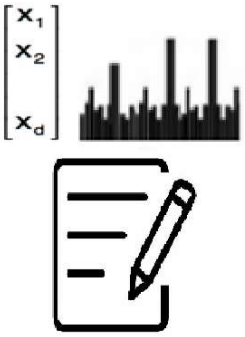

Figura 7. Representación de cada par codificado para construir el *dataset*

En una primera instancia, aplicábamos el *software* LBP-TOP sobre todos los *frames* que conformaban un discurso concreto del presentador. Sin embargo, siempre daba como resultado un vector de características de dimensión 3x59, independientemente del número de *frames* que fueran aportados como entrada del programa. Esto supone un problema, ya que nuestro objetivo es modelar e interpretar el habla continua, por lo que la dimensión del vector de características debe adaptarse a la longitud del discurso. En otras palabras, cuanto más tiempo dure el discurso, más elementos deberá contener el vector de características o histograma. Esto nos hace pensar que la técnica seleccionada estaría enfocada, en un principio, a un reconocimiento más específico y restringido como la clasificación de un número fijo de frases o palabras aisladas.

Si deseamos que cuanto más duración tenga el discurso más grande sea la dimensión del vector de características, debemos implementar un *script* tomando de base los originales. La idea consiste en extraer las características de un discurso dividiendo sus *frames* en bloques de un

7 http://www.cse.oulu.fi/wsgi/CMV/Downloads/LBPMatlab?action=AttachFile&do=view&target=STLBP_Matlab.zip

tamaño fijo para después suministrarlos al *software* por separado. De este modo, concatenando los distintos vectores obtenidos con cada bloque conformamos el vector de características definitivo. Por lo tanto, nuestro primer desafío es determinar el tamaño de estos bloques. Debido a que la frecuencia con la que se tratan los vídeos es de 30 *frames* por segundo (tal y como se comentó en el apartado 3.2.), se ha optado por englobar 6 *frames* en cada uno de los bloques, puesto que es un múltiplo de la frecuencia y así abarcamos (de una manera aproximada) la unidad mínima de articulación que pueda realizar el interlocutor. No obstante, debido a la topología del Modelo Oculto de Markov propuesto para este sistema, hemos optado por extraer las características de cada discurso mediante una aproximación con solapamiento. En otras palabras, se emplea un bloque del mismo tamaño que irá avanzando con un desplazamiento de una posición por cada vector de características obtenido. De esta manera, conseguimos que un bloque contenga cinco *frames* del bloque anterior solapados y, por lo tanto, tendrá otros cinco con el siguiente bloque, siempre y cuando no sean los bloques extremos de la secuencia de *frames*. Además, conseguimos modelar cierta información de contexto que puede ser de gran utilidad para el entrenamiento del sistema. Para facilitar la compresión de este método presentamos un esquema en la Figura 8.

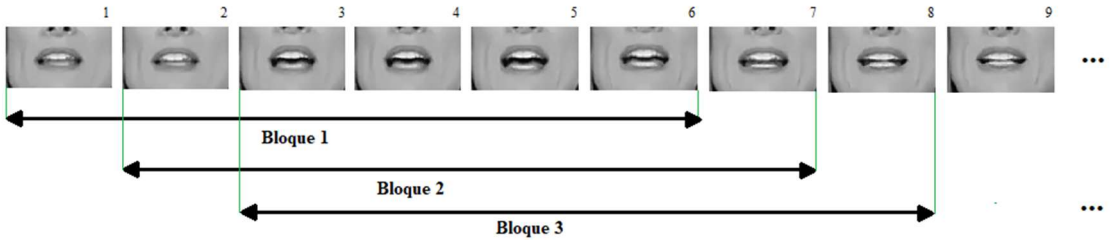

Figura 8. Extracción de características con solapamiento

De esta forma, el vector de características adquiere una longitud acorde con la topología del HMM, es decir, que pueda adaptarse al número de estados presentes en el sistema propuesto. Entonces, una vez completado el *dataset* y el respectivo aprendizaje, se procederá a analizar los resultados reportados en relación con la precisión del sistema.

3.3.1. Introducción a las técnicas de extracción de características visuales del movimiento labial

Antes de abordar la técnica escogida, es necesario introducir un contexto general respecto a las técnicas de extracción de características visuales del movimiento labial a partir de imágenes. Para ello, reiteramos que el cerebro humano utiliza de manera subconsciente la información visual percibida durante una conversación para comprender mejor el discurso del emisor. Esta influencia tanto del movimiento labial como de los gestos sobre la comprensión del mensaje queda demostrada por el llamado efecto McGurk [1], el cual también hace alusión a posibles confusiones a la hora de interpretar el habla. Aun así, proporciona una motivación para el reconocimiento del habla a partir únicamente del canal visual. Por otro lado, cabe destacar el desafío que presenta dicho propósito, ya que la región bucal es una de las partes más deformables y dinámicas de las expresiones faciales. Además, como ya se ha comentado a lo largo de toda la memoria, hay que tener en cuenta aspectos como los cambios de posición y rotación de la cabeza del *speaker*, la presencia o ausencia de elementos en el entorno, las condiciones lumínicas o la calidad con la que se obtuvieron las imágenes [9].

Si revisamos la literatura podemos clasificar las distintas técnicas de extracción de características en los tipos que se describen a continuación:

- Características geométricas: en esta aproximación se consideran la forma de la boca o puntos de referencia delimitando el contorno labial externo, interno o ambos. Además, a partir de estos puntos se pueden extraer otras métricas que representan características relevantes, como puede ser la altura, anchura y área bucal durante el discurso. En cambio, estos métodos requieren comúnmente de una detección precisa de la región de la boca, la cual resulta casi imposible si la resolución de los *frames* es muy baja. Por ejemplo, destacamos el modelo AAM, ya mencionado en el estado del arte.

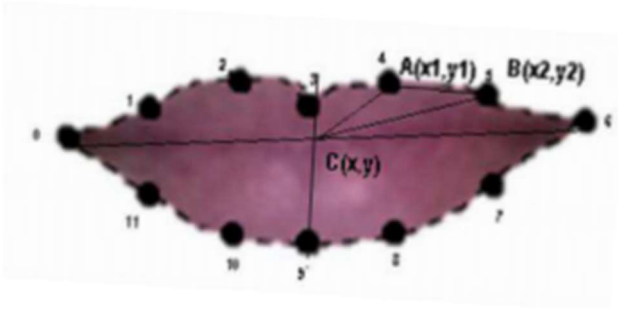

Figura 9. Ejemplo de extracción de características geométricas mediante AAM [9]

- Basados en la apariencia: en este caso se consideran, ya sea en escala de grises o en color, los valores de cada uno de los píxeles que componen la región de la boca; por ejemplo, la técnica PCA.
- Basados en la transformación de la imagen: donde se obtienen las características visuales mediante la transformación de la imagen de la boca a un espacio de características, usando para ello técnicas como DCT.
- Híbridas: son aquellas aproximaciones que combinan el potencial de más de una de las aproximaciones descritas hasta el momento.

En cuanto a la técnica que hemos empleado para alcanzar nuestro propósito, tenemos que clasificarla como una aproximación basada en la apariencia, aunque, además, presenta nuevos aspectos relevantes que pueden ser de gran utilidad para aumentar la robustez de nuestro sistema, tal y como se detallará en el siguiente apartado.

3.3.2. LBP-TOP *feature extractor*

La mayoría de los sistemas presentes en la literatura cuyo objetivo es el reconocimiento visual del habla han empleado las técnicas de extracción basadas en la apariencia, considerando únicamente las características globales de los *frames* sin tener en cuenta las locales. No obstante, estas últimas características pueden describir o aportar información respecto a los cambios locales que se producen, tanto en el tiempo como en el espacio, entre los sucesivos *frames* que componen el discurso del *speaker* en su totalidad. En otras palabras, el propósito es tener en cuenta el movimiento generado en la región de la boca, así como el orden temporal en el que sucede. Es por ello que hemos seleccionado la técnica LBP-TOP (Local Binary Pattern extracted from Three Orthogonal Planes) para extraer las características visuales con las que conformar definitivamente nuestro *dataset*.

En primer lugar, introducimos el operador LBP, ya que la técnica escogida es el producto de un proceso evolutivo. Dicho operador consiste en un descriptor visual en escala de grises enfocado, en sus inicios, a la clasificación de texturas a partir de imágenes [10]. La técnica recorre cada uno de los píxeles que forman la imagen, generando un código binario para cada uno de ellos. Este código es el resultado de aplicar una fórmula basada tanto en los valores del píxel seleccionado como en el de sus vecinos. Por tanto, el píxel escogido se denominará, de ahora en adelante, píxel central, debido a la posición que adopta (tal y como se puede comprobar en la Figura 10). Por otro lado, la fórmula mencionada se muestra en la Ecuación (1).

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) \cdot 2^p, \quad s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad (1)$$

Dicha ecuación tiene en cuenta los *P* píxeles vecinos que se deseen y la distancia a la que distan cada uno de estos píxeles con el píxel central, definiendo así un círculo de radio *R*, tal y como se puede apreciar en la Figura 10. Entonces, para calcular el código binario del píxel central se toma como umbral (*threshold*) el valor que adquiere éste en escala de grises (g_c) y se compara con el valor de cada uno de los píxeles vecinos (g_p) restantes. En cada iteración del sumatorio, tras hacer la diferencia de estas variables, se aplica sobre el resultado de esta operación la fórmula $s(x)$ para obtener un valor en el sistema binario. Después de esto, se multiplica el valor obtenido por el peso asociado al píxel vecino en cuestión (2^p). De esta manera, obtenemos su valor en el sistema decimal. Posteriormente, se realiza el sumatorio para obtener el resultado final. Por otro lado, cabe destacar que el sumatorio comienza desde el vecino que se encuentra en la esquina izquierda superior, tal y como se esboza en la Figura 10.

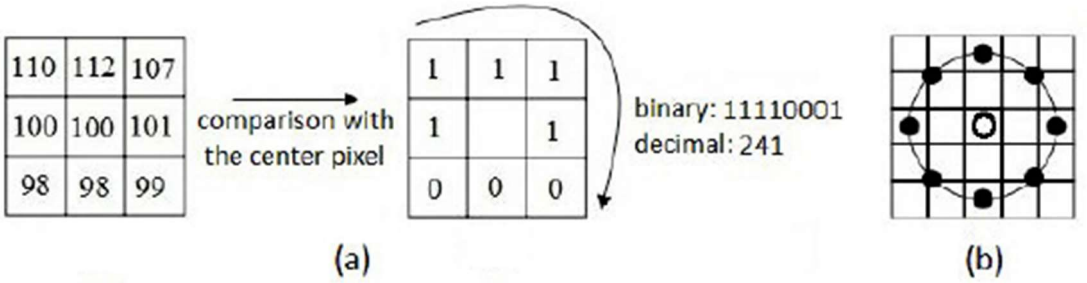

Figura 10. Representación del algoritmo LBP. (a) Algoritmo LBP básico. (b) Vecindario de ocho píxeles delimitado por un círculo de radio dos [18]

Con todo esto, se crea un histograma que recoge las apariciones de diferentes patrones binarios presentes en la imagen. Por otro lado, destacamos que se pueden compilar las imágenes en una matriz de manera que el *software* se aplique de una vez sobre un conjunto de múltiples imágenes, tal y como hemos realizado con nuestro *dataset*. De esta forma, se consigue extraer la información local de la secuencia de fotogramas.

Esta técnica ha conseguido acaparar la atención en cuanto al análisis de imágenes faciales, ya que presenta una alta robustez ante desafíos como pueden ser los cambios de pose por parte de la persona o los cambios en la iluminación de las imágenes tratadas. Esto ha supuesto uno de los factores clave por los que hemos elegido esta técnica.

Sin embargo, recientemente se propuso el método LBP-TOP [11] con el que la aproximación clásica del LBP, centrada en imágenes estáticas, derivó hacia el dominio espaciotemporal.

Además, sobre este nuevo *software*[8] (disponible como código libre) se han aplicado las modificaciones descritas en el artículo de referencia [4], para así adaptar el algoritmo a nuestro propósito final. Por otro lado, este nuevo enfoque obtiene las características visuales mediante la definición de tres planos sobre la secuencia de *frames,* tal y como se representa en la Figura 11. Por tanto, dependiendo del plano en el que nos encontremos podremos extraer diferente información referente a las características locales. Para ello, se han definido los ejes de abscisas y ordenadas, X e Y respectivamente, en cuanto a las características espaciales, mientras que respecto al tiempo se ha establecido el eje temporal T, el cual se fundamenta en la sucesión de *frames*. Por lo tanto, se presentan los siguientes planos a la hora de extraer las características que conformarán el vector o histograma:

- Plano XY: en este plano se capturan las características relacionadas con la apariencia de la región bucal.
- Plano XT: en este caso se sustraen las características asociadas al movimiento horizontal de los labios a lo largo de la secuencia de *frames* que componen el discurso.
- Plano YT: al igual que en el plano anterior se obtienen las características vinculadas al movimiento de los labios a lo largo de la secuencia, pero en esta ocasión respecto al movimiento vertical.

Como resultado, el *software* devuelve una matriz de dimensión 3x59, donde cada fila contiene las características de uno de los planos descritos. En otras palabras, las características procedentes de cada plano son concatenadas para conformar el histograma de un bloque de *frames*, tal y como puede observarse en la Figura 12.

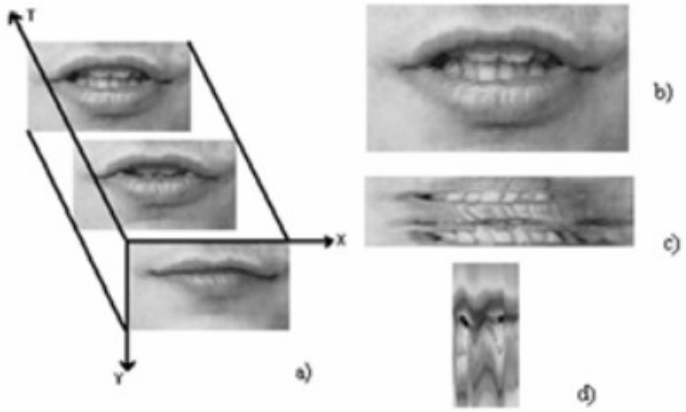

Figura 11. Síntesis del algoritmo LBP-TOP. (a) Secuencia de *frames*. (b) Región de la boca en el plano XY. (c) Región de la boca en el plano XT. (d) Región de la boca en el plano YT [4]

8 http://www.cse.oulu.fi/wsgi/CMV/Downloads/LBPMatlab?action=AttachFile&do=view&target=STLBP_Matlab.zip

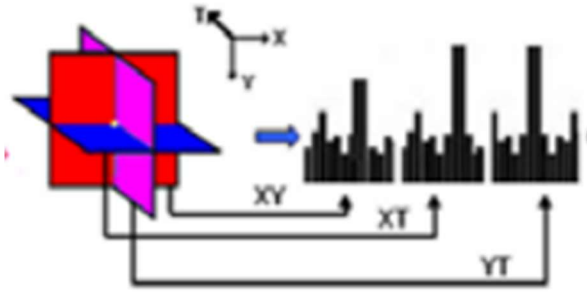

Figura 12. Construcción del vector de características de un bloque de *frames* [4]

Dada la habilidad para describir las señales espaciotemporales y la robustez ante los cambios lumínicos que causan variaciones en los valores de los píxeles, LBP-TOP es una de las técnicas más adecuadas para modelar el movimiento labial.

En nuestro caso, se emplearán bloques que contengan 6 *frames*, como ya dedujimos previamente. De esta manera, cada bloque genera un histograma o vector de características propio, que al final se concatenan en una secuencia obteniendo así las características definitivas que representan el discurso en su totalidad. Todo este proceso queda reflejado en la Figura 13. Por último, cabe destacar que dicho histograma es normalizado globalmente para obtener una descripción coherente.

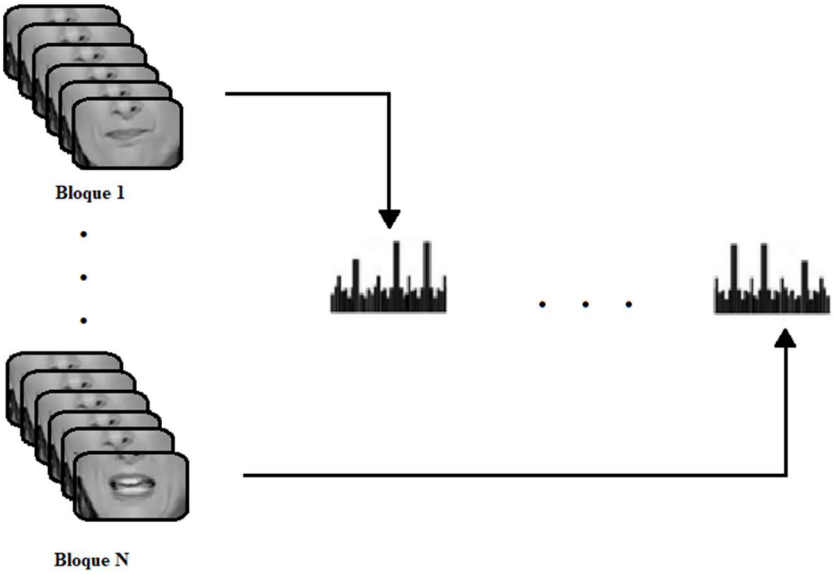

Figura 13. Construcción de la secuencia definitiva que integra los *frames* del discurso al completo

3.4. Selección de características

En *machine learning*, no todas las características recopiladas contribuyen de igual manera sobre la capacidad de nuestro sistema. A menudo los datos contienen características que son redundantes o que presentan una correlación entre ellas, por lo que su impacto sobre el entrenamiento del sistema es perjudicial. Además, estos aspectos contribuyen a que aumente la complejidad computacional a la hora de procesar la información, llegando incluso a ocasionar un sobreentrenamiento. En otras palabras, este suceso puede influir negativamente en cuanto a las prestaciones del sistema final. Por esta razón vamos a aplicar la técnica conocida como

selección de características sobre nuestro *dataset*, para así reducir la dimensionalidad de las secuencias obtenidas tras la extracción de características. De esta manera, al prescindir de las características no relevantes en cada una de las muestras de aprendizaje, reducimos la complejidad computacional y liberamos al sistema de cierta "confusión", consiguiendo mantener, o incluso mejorar, la capacidad discriminativa de nuestro modelo.

En base a la literatura, destacamos las técnicas Ξ^2 (*Chi square statistic*) y PCA (Principal Component Analysis), ambas detalladas en sucesivos apartados. Hemos escogido la primera de ellas porque fue empleada en el artículo de referencia [4], mientras que se ha tomado en cuenta PCA al ser una de las técnicas más conocidas en el ámbito del *machine learning*, adquiriendo un papel importante junto a modelos predictivos. Ambos métodos han sido utilizados gracias a la biblioteca *scikit-learn*[9], la cual integra una gran cantidad de algoritmos relacionados con el aprendizaje automático, como pueden ser aquellos vinculados a la clasificación, regresión lineal o selección de características. Esta biblioteca se divide en numerosos módulos entre los que destacamos *feature_selection* y *decomposition,* los cuales nos han provisto de las técnicas Ξ^2 y PCA, respectivamente.

Sin embargo, debido a nuestro propósito, nuestro *dataset* presenta la peculiaridad de que cada una de las secuencias, o muestra de aprendizaje, tiene una dimensión distinta en función de cuánto tiempo dura el discurso del emisor. Esto supone un problema a la hora de aplicar la selección de características, ya que a la hora de procesar las distintas técnicas sobre el conjunto de datos es necesario que cada una de las muestras presente la misma longitud para poder realizar los cálculos y comparaciones que determinen cuales son las características que deben ser descartadas. De esta manera, no se ve alterada la coherencia implícita de los datos. Entonces, nos hemos inclinado por una solución teniendo en cuenta los siguientes aspectos:

- Nuestro *dataset* está compuesto por secuencias. Una secuencia representa la codificación del discurso emitido por el presentador de telediario. Más concretamente, cada una de estas secuencias está construida por un conjunto de vectores de características extraídos mediante la técnica, ya descrita, LBP-TOP.
- A su vez, estos vectores se construyeron a partir de bloques de seis *frames*, tal y como se detalló en la sección 3.3. Y como ya sabemos, cada uno de estos vectores presenta una talla fija en forma de matriz. Las dimensiones de esta matriz son de 3x59, donde cada fila representaba las características en función del plano del que fueron extraídas. Sin embargo, estos vectores o histogramas son incorporados al sistema formando un único vector de 177 componentes.

Por lo tanto, hemos optado por desglosar cada una de las secuencias de nuestro *dataset* en sus respectivos bloques de características, tal y como se muestra en la Figura 14. De esta forma, nos aseguramos manipular la información en estructuras de datos que presenten el mismo número de componentes.

[9] https://scikit-learn.org/stable/index.html

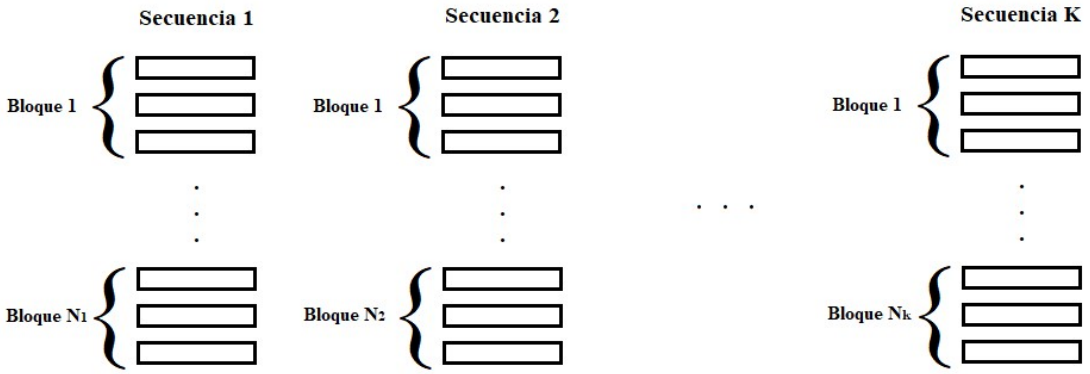

Figura 14. División de cada una de las secuencias del dataset en sus respectivos bloques de características

Posteriormente, conformaremos una matriz de datos, donde cada una de las filas compila cada bloque sustraído a partir de las secuencias. De este modo, cada fila constituye un vector de 177 elementos (3x59 elementos conforman el bloque). Con este formato ya somos capaces de aplicar la selección de características, puesto que cada una de las muestras de aprendizaje presenta el mismo número de características. No obstante, es necesario almacenar marcadores para la reconstrucción posterior del *dataset* reducido, es decir, agrupar los respectivos bloques para conformar de nuevo cada una de las secuencias. Al igual que antes, se ha adjuntado la Figura 15 para comprender mejor la metodología llevada a cabo.

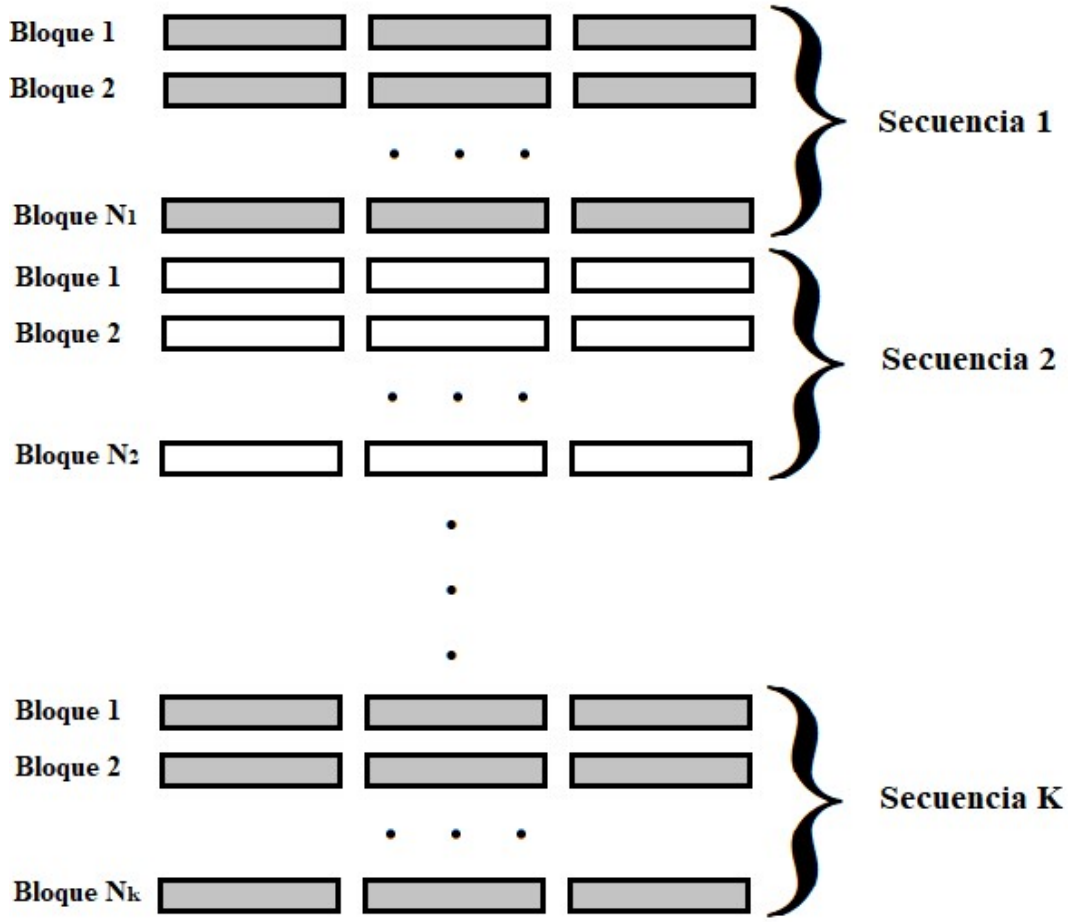

Figura 15. Restructuración del dataset con el objetivo de aplicar la selección de características

3.4.1. *Chi square statistic (Ξ^2)*

Esta técnica es empleada para seleccionar aquellas características con los valores más altos del *chi-square test*. En nuestro caso, se ha optado por utilizar el método *select_Percentile(%)* para mantener cierto porcentaje de las características que conforman cada bloque extraído, en lugar de definir, *a priori,* una dimensión específica a reducir. Es necesario indicar que el *chi-square test*[10] mide la dependencia existente entre variables estocásticas, por lo que aplicando este proceso de selección descartamos las características que no son relevantes para el entrenamiento. Por otro lado, esta técnica está enfocada hacia un aprendizaje supervisado, es decir, requiere de las etiquetas o transcripciones asociadas a cada bloque. Sin embargo, nuestro sistema se orienta a la interpretación del habla continua y, por lo tanto, a una tarea de predicción en la que no se han definido ningún tipo de clases entre las que distinguir. Por ello, se ha solventado este inconveniente aportando el vector de etiquetas con todos los valores puestos a 1, sin ofrecer ninguna distinción entre clases.

En primera instancia, al no conocer de antemano cuál sería la dimensión más adecuada para nuestro propósito, nos hemos inclinado por realizar múltiples selecciones que varíen en el porcentaje de características sin descartar. Por ello, teniendo en cuenta que originalmente cada bloque disponía de 177 componentes, distinguimos los siguientes *datasets* reducidos:

- *Dataset* al 50%: con esta aproximación mantenemos la mitad de las características contenidas en cada bloque. Entonces, los bloques con los que se ha reconstruido el *dataset* han reducido su dimensión a 88 componentes.
- *Dataset* al 60%: en este caso, los bloques del *dataset* han visto reducida su longitud a 104 características.
- *Dataset* al 70%: con este porcentaje, el *dataset* reduce la talla de sus bloques a 123 características.
- *Dataset al 80%:* por último, se descartan un número ínfimo de características, consiguiendo una dimensión de 141 por bloque.

Se han escogido dichos porcentajes para abarcar un espectro considerable, es decir, desde una reducción brusca (como puede ser aquella que descarta la mitad de las características), hasta una reducción escueta, en caso de que eliminar muchas componentes supusiera un decremento respecto a la capacidad discriminativa del sistema. Por otro lado, descartar un mayor número de componentes ocasionaba errores debido al posible ruido presente en los datos o, más bien, a una selección demasiado estricta.

3.4.2. *Principal Component Analysis*

Principal Component Analysis (PCA)[11] es una técnica que proyecta un conjunto de datos con el objetivo de reducir la dimensionalidad del conjunto descartando las características que presentan cierta correlación lineal entre ellas. El resultado obtenido son las denominadas componentes principales, las cuales se encuentran ordenadas en función de la varianza que contengan. En otras palabras, obtenemos las características que más contribuyen de cara al entrenamiento del modelo.

[10] https://en.wikipedia.org/wiki/Chi-squared_test
[11] https://en.wikipedia.org/wiki/Principal_component_analysis

Al emplear esta técnica es necesario definir, como parámetro, la dimensión a la que deseemos reducir cada una de las muestras de aprendizaje o secuencias. Sin embargo, desconocemos cuál es la dimensión más adecuada para nuestro propósito. Por ello, hemos realizado un proceso que nos permita determinar la dimensión adecuada dependiendo de la varianza mantenida en el *dataset* a pesar de la reducción aplicada. En otras palabras, buscamos mantener la máxima cantidad de varianza, para así no desaprovechar información relevante respecto a las prestaciones finales, pero con el mínimo número de características posibles. Este análisis ha sido llevado a cabo gracias al uso de las bibliotecas *sklearn*, *numpy* (en cuanto al análisis en sí) y *matplotlib.pyplot* (para poder representar la información de manera gráfica). Entonces, tras realizar dicho análisis, decidimos que cada uno de los bloques de características deberá contener 75 componentes, frente a los 177 originales; con los que preservaremos alrededor del 98% de la varianza total del *dataset*, tal y como se puede apreciar en la Figura 16.

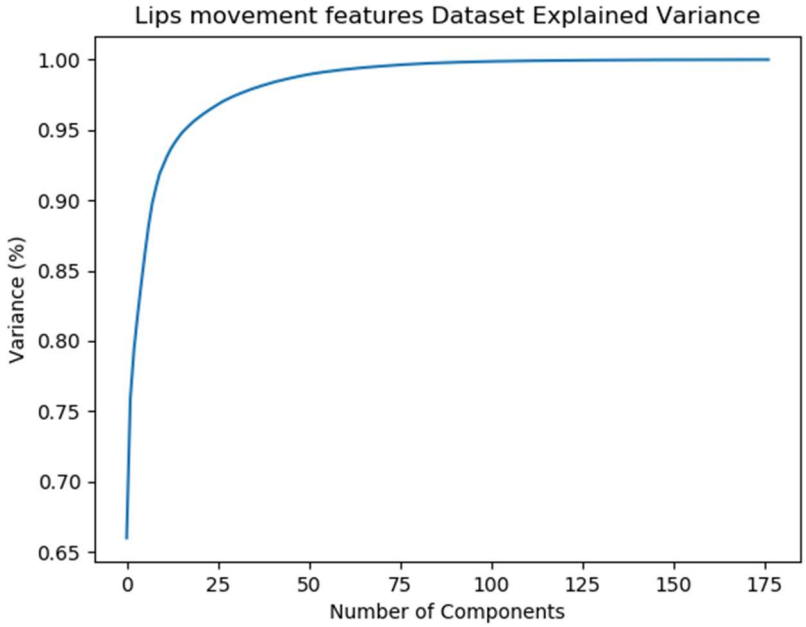

Figura 16. Varianza presente en función del número de componentes PCA contenidas en un bloque de características

3.5. Aportación extra de una base gramatical

Con el objetivo de mejorar la precisión del sistema se ha propuesto enriquecer el modelo de lenguaje presente en nuestro sistema. Este modelo, así como la arquitectura con la que se ha construido el sistema mencionado se describen en la sección 3.6 de la memoria.

Por tanto, se han recopilado un conjunto de noticias, en formato de texto, comprendidas en el mismo intervalo de tiempo que los vídeos usados en el *dataset* principal. Todas estas noticias fueron recogidas a través de la API ofrecida por RTVE. De esta manera, hemos obtenido un *corpus* compuesto por más de medio millón de palabras, con las cuales nutriremos al sistema de una base gramatical que consolide sus estimaciones.

3.6. Entrenamiento y arquitectura del sistema

Esta fase constituye uno de los pilares fundamentales del sistema, ya que determina la precisión de los resultados generados. Sin embargo, como ya se ha podido observar a lo largo de toda la memoria, la precisión con la que el sistema leerá los labios dependerá en gran medida del *dataset* recopilado y la información contenida en éste. No obstante, el modelo a entrenar influye significativamente, puesto que determina cómo el sistema aprende la tarea en cuestión.

Tal y como se mencionó en el apartado 2.1 del estado del arte, el aprendizaje automático se fundamenta en el análisis de datos y la estadística inferencial con el objetivo de permitir que un sistema aprenda, tome decisiones o identifique patrones a partir de un conjunto de datos y con la mínima intervención humana posible. Para comprender este proceso, en términos generales, suponga que tiene un examen dentro de poco y dispone de varios exámenes de años anteriores para estudiar. Usted, entonces, decide estudiar a base de los ejercicios propuestos en estos exámenes. Cada uno de estos ejercicios se compone del enunciado y de la solución asociada. En otras palabras, nos encontramos ante un aprendizaje supervisado, ya que tenemos a nuestro alcance los datos ofrecidos (enunciado) y la solución esperada. Sin embargo, dicho enunciado tendrá que sufrir un preproceso donde usted extraerá la información más relevante de cara a resolver el problema (lo que suele suponer a menudo un conocimiento previo), es decir, nos encontramos en la fase de extracción de características. Tras esta fase, comenzará el entrenamiento, o más bien comenzará a estudiar a partir de los ejemplos proporcionados por los exámenes. De esta manera, el cerebro infiere o identifica patrones que le permitirán resolver un ejercicio que comparta cierta relación con otro ya estudiado. Como todos sabemos, esta fase consiste en un proceso gradual en el que serán necesarios un amplio abanico de ejemplos si se desea alcanzar un sobresaliente. Aunque esto último no nos lo asegura nadie, ya que siempre pueden sorprendernos. Entonces, en último lugar nos resta la fase del *testeo*, es decir, el examen y la calificación obtenida.

Para nuestro propósito, se ha propuesto una arquitectura compuesta por múltiples modelos, donde destacamos que el modelo principal es un Modelo Oculto de Markov Continuo debido a que la mayoría de sus aplicaciones han estado enfocadas al tratamiento del lenguaje como el reconocimiento de la escritura manual o el reconocimiento del habla, tal y como se ha podido observar en la literatura.

Antes de abordar la arquitectura de nuestro modelo, es necesario indagar sobre los sistemas tradicionales construidos con anterioridad, de modo que observemos la gran relación entre ellos. Entonces, tal y como pudimos ver en el estado del arte, encontramos una extensa literatura en torno a los SRAH, es decir, aquellos sistemas cuyo objetivo es decodificar la transcripción de un mensaje a partir de su señal acústica. Se pueden orientar tanto a tareas de reconocimiento de palabras o dígitos aislados, como al reconocimiento de frases completas. Esta última orientación son los denominados Sistemas de Reconocimiento del Habla Continua (SRHC), como es nuestro caso, con la diferencia de que nuestro sistema se basará únicamente en la información procedente del canal visual.

En cualquier caso, la señal acústica es procesada con el objetivo de permitir al sistema interpretar la información contenida en ella. Básicamente, se pretende representar de una manera más clara los distintos sonidos articulados que componen el audio correspondiente. Este proceso convierte habitualmente la señal acústica en el dominio amplitud-tiempo a una señal

representada en el dominio frecuencia-tiempo. En la mayoría de los casos, el proceso más empleado es el que produce los coeficientes cepstrales en la escala de Mel (MFCC por sus siglas en inglés) [19]. De este modo, la señal acústica es transformada en una secuencia de vectores compuestos por números reales.

En la década de 1990, estos sistemas comenzaron a tener prestaciones aceptables. La tecnología estándar de aquellos sistemas se basaba en el empleo de tres tipos de modelo y un proceso de búsqueda (decodificación) sobre la combinación de éstos, tal y como se puede observar en el esquema presentado en la Figura 16. Estos modelos son:

- Modelo acústico: es un caso particular de los llamados modelos morfológicos, los cuales son aplicables a cualquier tarea donde la información sea compilada en señales en bruto compuestas por una secuencia de unidades definidas. En el caso del modelo acústico, se realiza la asociación de la señal acústica procesada con los sonidos básicos del lenguaje correspondiente (fonemas o unidades similares). Este modelo se implementa mediante Modelos Ocultos de Markov Continuos; son HMM's en los que la salida en cada uno de sus estados es modelada por una distribución de probabilidad formada por una mixtura de gaussianas de la dimensión correspondiente [20].
- Modelo léxico: este modelo indica cómo se combinan los sonidos con el objetivo de construir palabras. Su implementación pasa por modelos de estados finitos cuyas transiciones reflejan los sonidos (es decir, la secuencia de modelos acústicos) que forma cada palabra.
- Modelo de lenguaje: en este modelo se integran los procedimientos encargados de definir cómo combinan las palabras para formar frases con sentido, siempre dentro del dominio de trabajo del sistema. La implementación de este modelo se basa en los denominados *n*-gramas, es decir, modelos que indican la probabilidad de que se dé una cierta palabra dependiendo de las $n - 1$ anteriores.

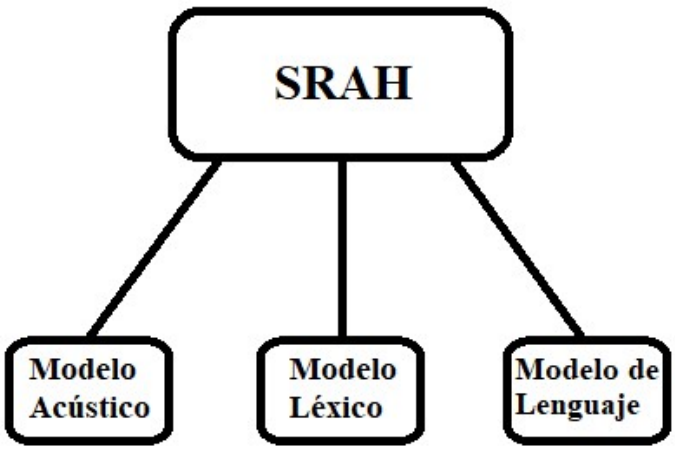

Figura 16. Arquitectura del sistema empleado

Este modelado presenta la ventaja de ser fácil de cara al entrenamiento mediante algoritmos ampliamente conocidos, siempre y cuando se aporte una cierta cantidad de datos apropiadamente anotados. Además, se integra de manera adecuada con el proceso de búsqueda posterior, implementado, en nuestro caso, mediante el algoritmo de Viterbi [21]. En esta época, muchas de las herramientas de creación de SRAH, como HTK [22], Sphinx [23] o iAtros [24], se basaron en este paradigma, extendiéndolo incluso a otras áreas como el reconocimiento de texto manuscrito [25].

Como hemos visto, las arquitecturas de los SRAH tradicionales trabajan sobre una señal de audio convertida a una secuencia de vectores. Generalizando esto, cualquier secuencia, sea cual sea su naturaleza, capaz de ser transformada a una secuencia de vectores sería susceptible de ser procesada por estos sistemas. Por tanto, la adaptación de un SRAH a la lectura de labios es tan sencillo como sustituir la fuente de información acústica por las características extraídas a partir de la secuencia de *frames*, tal y como se ha descrito en otras secciones del capítulo. Además, se necesitaría un etiquetado correcto (es decir, la transcripción) para poder entrenar apropiadamente los parámetros de los modelos morfológicos, lo cuales no sufrirán ninguna modificación debido a la asociación directa entre los movimientos labiales y la señal de audio. Por otra parte, tanto el modelo léxico como el de lenguaje se mantendrían sin cambios, ya que no se altera la naturaleza de las unidades básicas (sonidos) ni tampoco el dominio correspondiente.

Con todo esto, ya estamos preparados para describir el sistema empleado en nuestro proyecto. El esquema se basará en la arquitectura vinculada a los sistemas tradicionales, tal y como hemos podido inferir a lo largo de este apartado. Entonces, nuestra arquitectura se fundamenta en un conjunto de HMM's con el objetivo de modelar los fonemas estándar en español, así como el silencio. La topología escogida para los HMM's se ha basado en la empleada habitualmente en reconocimiento de habla. Esta topología define tres estados, cuyas transiciones avanzan de izquierda a derecha entre ellos. Además, se han añadido bucles en los distintos estados, así como transiciones al estado final desde cualquier estado, para evitar problemas con la longitud de las secuencias. En otras palabras, como cada una de las secuencias aportadas al sistema tiene una dimensión distinta, es necesario modelar esta característica del *dataset*. La topología descrita puede observarse de una manera esquemática en la Figura 17. Por otro lado, destacamos que estos modelos morfológicos han sido entrenados usando HTK [22] mediante el protocolo habitual.

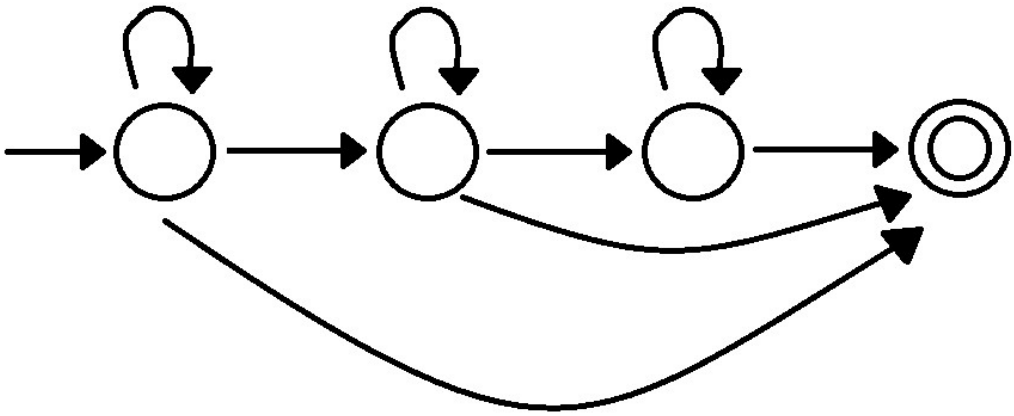

Figura 17. Topología del Modelo Oculto de Markov empleado en la lectura de labios

El modelo léxico se ha obtenido a través de las reglas de pronunciación para el español [26] definidas en el *script* identificado como eutranscribe. Por otra parte, el modelo de lenguaje se ha nutrido a partir de un conjunto de transcripciones de noticias pertenecientes al mismo intervalo

de tiempo que los vídeos empleados en el resto del entrenamiento, es decir, nos estamos refiriendo a la base gramatical comentada en la sección 3.5. A partir de esta base gramatical, se han inferido *n*-gramas de orden 2, 3 y 4, de modo que el sistema aprenda a conformar frases con cierto sentido. En último lugar, el proceso de búsqueda o decodificación se ha realizado empleando el sistema iAtros [24] desarrollado por el centro de investigación PRHLT[12].

En este sistema , además de la incorporación de los modelos indicados con anterioridad, se pueden ajustar diversos parámetros del proceso de reconocimiento. Más concretamente, estos parámetros son:

- Ancho de búsqueda (*beam*): como ya introdujimos, el proceso de búsqueda está gobernado por el algoritmo de Viterbi, el cual tiene como objetivo definir el camino o secuencia de estados más probable con la que se ha alcanzado la salida del HMM. Este proceso se realiza de forma gradual y, por lo tanto, se tendrán en cuenta diversos caminos hasta la decisión final. En cierta medida, es similar a un proceso de ramificación y poda, ya que disponemos de la capacidad para descartar la expansión de caminos cuya probabilidad parcial no es prometedora en comparación con el resto de caminos explorados. Entonces, este factor indica qué caminos deben descartarse, que serán aquellos cuya probabilidad sea *beam* veces menor que la probabilidad máxima hasta el momento actual. Por tanto, a mayor valor el sistema será más preciso y lento, mientras que a menor valor será más impreciso y rápido.
- Peso del modelo de lenguaje (*gramar-scale-factor*): dentro del proceso de decodificación (es decir, construir la transcripción del discurso) se combinan las probabilidades que dan tanto los modelos morfológicos (HMM) como el modelo de lenguaje (*n*-grama). Sin embargo, estas probabilidades suelen pertenecer a rangos muy distintos, acentuando la dependencia sobre uno de los modelos. Debido a esto, el parámetro en cuestión toma un papel importante a la hora de decodificar el mensaje. Con este factor se puede aumentar o disminuir la influencia del modelo de lenguaje, de forma que en sistemas donde el modelo morfológico presente una estimación pobre puedan mejorarse las prestaciones aumentando la dependencia respecto al modelo de lenguaje, siempre y cuando esté bien estimado.
- Penalización de inserción (*Word-insertion-penalty*): como ya sabemos, en el habla continua no se define ninguna pausa específica entre las palabras emitidas. Este suceso se ve reflejado sobre las secuencias extraídas, las cuales podrían ser decodificadas como una o múltiples palabras. Por ejemplo, la secuencia acústica " a b o g a d o" podría decodificarse como "abogado", "aboga do" o "a boga do". Este factor promueve la formulación de palabras largas frente a varias palabras cortas con el objetivo de mejorar la decodificación final.

Finalmente, el sistema ha sido parametrizado con los valores óptimos para cada uno de los factores comentados, de modo que se mejoren las prestaciones finales del sistema. Por otro lado, se ha aportado la base gramatical, para así estimar el modelo de lenguaje. Una vez configurado el modelo en su totalidad, se ha nutrido a éste con el *dataset* mencionado a lo largo de la memoria, siendo este el último paso antes de finalizar el entrenamiento del sistema.

[12] https://www.prhlt.upv.es/wp/es/

3.7. Resultados obtenidos

En esta sección se hace hincapié sobre el protocolo experimental llevado a cabo con el propósito de obtener los primeros resultados. En primer lugar, se ha construido un sistema con los datos descritos a lo largo del capítulo. Básicamente, la experimentación ha consistido en la construcción de los modelos morfológicos, léxico y de lenguaje a partir de la definición de un conjunto de entrenamiento y las posteriores pruebas de decodificación del habla sobre un conjunto de test.

Respecto al entrenamiento de los modelos morfológicos se han usado los datos extraídos de los locutores 0, 1 y 2. Los tipos de modelos morfológicos empleados son los conocidos como Modelos Ocultos de Markov con mixturas de gaussianas. Según la extracción de características utilizada se ha conseguido entrenar modelos hasta un determinado número de gaussianas, oscilando entre 2 (para selección de características Ξ^2 al 50%) y 64 (para selección de características por PCA). Por otra parte, los datos del locutor 3 han sido escogidos para realizar la fase del test con la que finalizaremos este capítulo.

En los procesos de decodificación se han probado distintas configuraciones de los parámetros mencionados en la sección anterior. Más concretamente, se ha experimentado con valores de *beam* comprendidos entre 100 y 500 a intervalos de 100, y valores de *gramar-scale-factor* entre 10 y 70 a intervalos de 10. Respecto al modelo de lenguaje, se emplearon los 4-gramas estimadas a partir del conjunto de datos de noticias que conformaron la base gramatical extra.

Tras la experimentación, lo primero que cabe destacar es la lentitud de los procesos de decodificación. Frases de apenas unos segundos tienen tiempos de decodificación de minutos u horas. No obstante, las decodificaciones obtenidas resultan casi aleatorias, aunque con cierta coherencia introducida por el modelo de lenguaje pero sin parecido razonable con las decodificaciones esperadas.

Por tanto, aunque hemos conseguido realizar los objetivos de este trabajo, los resultados del sistema no son mínimamente aceptables. Hay que tener en cuenta que en la práctica totalidad de la bibliografía consultada respecto al tema se indican tasas de acierto bastante reducidas, incluso para tareas mucho más simples (generalmente clasificación de palabras o frases concretas). Entonces, el haber obtenido en esta primera aproximación a un problema mucho más difícil como es la interpretación del habla continua este escaso éxito entra dentro de lo esperable.

4. Conclusiones

A pesar de que los resultados no han alcanzado una calidad esperable, tal y como se esperaba, el balance general del proyecto es positivo, ya que se han cumplido los objetivos propuestos en la introducción de la memoria.

En primer lugar, se ha construido un *dataset* enfocado a la lectura de labios a partir del telediario ofrecido en RTVE. Este *dataset* compila aproximadamente 3600 segundos, con lo que se puede considerar un conjunto de datos considerable aunque esté lejos de los *corpus* de gran escala orientados al entrenamiento de sistemas *Deep Learning*. Por otro lado, se ha creado un sistema preliminar de lectura de labios a partir, únicamente, de la información procedente del canal visual. Por último, pero no menos importante, el trabajo ha permitido al alumno adquirir experiencia con todo el proceso asociado a la construcción de un sistema *machine learning*, ya que se ha participado en todas las fases de las que se compone.

A modo de conclusión, es necesario destacar que a menudo lo que nos proponemos no siempre se alcanza en el primero de los intentos, sino que los dos aspectos más relevantes a tener en cuenta son la constancia y la paciencia. De hecho, se continuará con esta temática sobre futuros trabajos, tal y como demuestra la sección 5 de la memoria.

5. Relación del trabajo desarrollado con los estudios cursados

A lo largo de la carrera se han cursado materias que han permitido el desarrollo del proyecto propuesto en la memoria, teniendo en cuenta que se ha estudiado la Rama de Computación. Todas y cada una de ellas han aportado de una u otra manera, ya sea desde asentar los conocimientos elementales de la informática hasta proporcionar una base de la inteligencia artificial y el aprendizaje automático.

En primer lugar, destacamos asignaturas como PRG (Programación), EDA (Estructuras de Datos) y ALT (Algorítmica) puesto que han colaborado en la formación del alumno a la hora de interpretar o generar código, así como en la gestión de las diferentes estructuras de datos disponibles para el programador con las que poder manipular los datos de forma eficiente, siendo estas asignaturas una de las principales bases que han permitido generar o modificar los programas empleados en el trabajo.

Por otra parte, resaltamos las materias de SAR (Sistemas de Almacenamiento y Recuperación de la Información) y Percepción (PER) por la experiencia aportada con lenguajes de programación como *Python* y *Octave*. El primero de ellos supone el conocimiento de un lenguaje sencillo pero ampliamente usado y con un gran abanico de bibliotecas, tal y como se pudo apreciar en la apartado 3.2.1. En cuanto a *Octave*, se trata de un lenguaje enfocado al análisis numérico y el tratamiento de datos, siendo considerado como el equivalente libre de *Matlab*, que ha sido utilizado a la hora de extraer las características.

Por último, asignaturas como APR (Aprendizaje Automático) y PER han sentado las bases del aprendizaje automático o *machine learning*, dando a conocer el proceso que conlleva este tipo de proyectos, así como las alternativas presentes y futuras vías de desarrollo. No obstante, destacamos PER ya que ha consolidado los cimientos del tratamiento de la información para poder nutrir los sistemas de reconocimiento, entre otros aspectos. Respecto al modelo empleado (HMM) subrayamos el papel de asignaturas como Sistemas Inteligentes (SIN) que dieron a conocer estos modelos junto a las bases estadísticas relacionadas.

Respecto a las competencias transversales, durante la realización de este proyecto se han consolidado las competencias relacionadas con el ”aprendizaje constante”, “comunicación efectiva” y “planificación y gestión del tiempo”, entre otras.

6. Trabajos futuros

Dado que los resultados no han sido los esperados, se van a planificar una serie de acciones con el objetivo de mejorar las prestaciones del sistema propuesto. Estas acciones se plasmarán sobre futuros proyectos, tomando como base todo el trabajo realizado hasta el momento, aunque éste es susceptible de sufrir posibles cambios.

En primer lugar, se propone elaborar un cambio de paradigma en cuanto a la tecnología empleada para implementar el sistema. En otras palabras, se intentará focalizar el sistema hacia las técnicas asociadas con el aprendizaje profundo (*Deep Learning*), como otras aproximaciones presentes en el estado del arte. Entre las principales razones que han impulsado esta propuesta encontramos el auge de estas tecnologías en la pasada década. Por un lado, buena parte de los sistemas actuales ha mantenido la arquitectura comentada en anteriores secciones, pero cambiando la distribución de salida en los distintos estados de los HMM por redes neuronales profundas. Este es el caso de múltiples sistemas que han sido desarrollados en base a Kaldi [27]. Por otro lado, el empleo de modelos recurrentes (RNN por sus siglas en inglés) ha derivado en el abandono de los HMM's como modelo acústico para que las propias redes neuronales se encarguen de obtener la secuencia acústica correspondiente, sobre la que después se aplicarán modelos léxicos y de lenguaje como, por ejemplo, el sistema DeepSpeech [28]. Por tanto, se decidirá una arquitectura en concreto y se realizarán los experimentos apropiados en base a distintos aspectos.

Por otro lado, en vistas de la anterior acción, es necesario aplicar un esfuerzo en mejorar el *dataset* construido para este proyecto. En primera instancia, será conveniente incrementar el tamaño del *corpus*, es decir, aportar más segundos de telediario, ya que para el entrenamiento de redes neuronales es preciso disponer de un conjunto de datos a gran escala, tal y como se mencionó en el estado del arte. Por otra parte, podrían obtenerse los vídeos con una mejor calidad de imagen, o aplicar restricciones de manera que el *dataset* sea más homogéneo. En otras palabras, que este conjunto de datos no incluya tanta variabilidad como hasta el momento, y que, por ejemplo, todos los planos de los presentadores sean tomados desde distancias similares.

En último lugar, se tendrá en cuenta un estudio de las diferentes técnicas de extracción de características que puedan mejorar la precisión del sistema. Con esto, nos obligamos a experimentar y no estancarnos con el trabajo realizado hasta el momento.

7. Referencias bibliográficas

[1] H. McGurk, J. MacDonald, Hearing lips and seeing voices, Nature 264 (1976), pp 746-748.

[2] Enrique Vidal Ruiz, Francisco Casacuberta Nolla, Tema 1-Introducción, Aprendizaje Automático, Departamento de Sistemas Informáticos y Computación (DSIC), Universidad Politécnica de Valencia (UPV), 2018.

[3] Adriana Fernandez-Lopez, Federico M. Sukno, Survey on automatic lip-reading in the era of deep learning, *Image and Vision Computing,* 2018.

[4] G. Zhao, M. Barnard, M. Pietikainen, Lipreading with local spatiotemporal descriptors, IEEE Trans. Multimedia 11 (7) (2009), pp 1254-1265.

[5] Z. Zhou, G. Zhao, M. Pietikainen, Towards a practical lipreading system, Proc. Conference on Computer Vision and Pattern Recognition, 2011, pp 137-144. IEEE.

[6] Torfi, Amirsina and Iranmanesh, Seyed Mehdi and Nasrabadi, Nasser and Dawson, Jeremy. *visualizeLip.py*, 3D Convolutional Neural Networks for Cross Audio-Visual Matching Recognition. 2017, pp 22081-22091. IEEE *Access*.

[7] Davis E. King, Dlib-ml: A Machine Learning Toolkit. *Journal of Machine Learning Research,* 2009, pp 1755-1758.

[8] Guoying Zhao, Matti Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(6):915-928.

[9] Waqqas ur Rehman Butt, Luca Lombardi, A Survey of Automatic Lip Reading Approaches., *Eighth International Conference on Digital Information Management (ICDIM 2013)*. IEEE, 2013. p. 299-302.

[10] T. Ojala, M.Pietikäinen and T. Mäenpää, "Multiresolution gray scale and rotation invariant texture analysis with local binary patterns", *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(7), (2002) 971-987.

[11] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6) (2007) 915-928.

[12] M. Gurban, J.-P. Thiran, Information theoretic feature extraction for audio-visual speech recognition, Signal Process. 57 (12) (2009) 4765-4776.

[13] G. Papandreou, A. Katsamanis, V. Pitsikalis, P. Maragos, Adaptive multimodal fusion by uncertainty compensation with application to audio-visual speech recognition, Proc. International Conference on Multimodal Processing and Interaction, 2008. pp. 1-15.

[14] R. Seymour, D. Stewart, J. Ming, Comparison of image transform-based features for visual speech recognition in clean and corrupted videos, J. Signal Image Video Process, 2008, vol. 2008, p. 14-22.

[15] A. Rekik, A. Ben-Hamadou, W. Mahdi, An adaptive approach for lip-reading using image and depth data, Multimedia Tools Appl. 75 (14) (2016) 8609-8636.

[16] D. Lee, J. Lee, K.-E. Kim, Multi-view automatic lip-reading using neural network, Proc. Asian Conference on Computer Vision, 2016. pp. 290-302.

[17] T. Afouras, J.S. Chung, A. Zisserman, Deep lip reading: a comparison of models and an online application, Proceedings of Interspeech, 2018, pp 3514-3518.

[18] Tiago F. Pereira, Marcus A. Angeloni, Flávio O. Simoes, José Eduardo C. Silva, Video-Based Face Verification with Local Binary Patterns and SVM Using GMM Supervectors, Research and Development Center in Telecommunications, pp. 240-252, 2012.

[19] ETSI. Etsi es 201 108 (v1.1.3), speech processing, transmission and quality aspects (stq); distributed speech recognition; front end feature extraction algorithm, compression algorithms. Technical report, ETSI, 2003.

[20] Xuedong Huang, Yasuo Ariki, and Mervyn Jack. *Hidden Markov Models for Speech Recognition*, Columbia University Press, New York, NY, USA, 1990.

[21] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, Inc., Upper Saddle Rriver, NJ, USA, 1993.

[22] Steve J. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book Version 3.4.* Cambridge University Press, 2006.

[23] Willie Walker, Paul Lamere, Philip Kwok, Bhiksha Raj, Rita Singh, Evandro Gouvea, Peter Wolf, and Joe Woelfel. Sphinx-4: A flexible open source framework for speech recognition. Technical report, Mountain View, CA, USA, 2004.

[24] Míriam Luján-Mares, Vicent Tamarit, Vicent Alabau, Carlos D. Martínez Hinarejos, Moisés Pastor i Gadea, Alberto Sanchis, and Alejandro H. Toselli. iatros: A speech and handwriting recognition system. In *V Jornadas en Tecnologías del Habla (VJTH'2008)*, pages 75-78, 2008.

[25] Thomas Plötz and Gernot A. Fink. Markov models for offline handwriting recognition: a survey. *IDAR*, 12(4):269-298, 2009.

[26] Antonio Quilis. *Principios de fonología y fonética españolas*. Arco/Libros S.L., 2010.

[27] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yyanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely. The kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, December 2011. IEEE Catalog NO.: CFP11SRW-USB.

[28] Awni Y. Hannun, Carl Case, Jared Ccasper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. Deep speech: Scaling up end-to-end speech recognition. *CoRR*, abs/14122.5567, 2014.

[29] I. Matthews, T.F. Cootes, J.A. Bangham, S. Cox, R. Harvey, Extraction of visual features for lipreading, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2) (2002) 198-213.

[30] K. Messer, J. Matas, J. Kittler, J. Luettin, G. Maitre, XM2VTSDB: the extended M2VTS database, Proc. International Conference on Audio and Video-based Biometric Person Authentication, vol. 964, 1999. pp. 965-966.

[31] M. Cooke, J. Barker, S. Cunningham, X. Shao, An audio-visual corpus for speech perception and automatic speech recognition, J. Acoust. Soc. Am. 120 (5) (2006) 2421-2424.

[32] D. L. Howell, Confusion Modelling for Lip-reading, University of East Anglia. 2015.Ph.D. Thesis.

[33] J. S. Chung, A. Senior, O. Vinyals, A. Zisserman, Lip reading sentences in the wild, Proc. Conference on Computer Vision and Pattern Recognition, 2017. pp. 3444-3453.

[34] B. Lee, M. Hasegawa-Johnson, C. Goudeseune, S. Kamdar, S. Borys, M. Liu, T.S. Huang, AVICAR: audio-visual speech corpus in a car environment., Proceedings of Interspeech, 2004.

[35] A. Ortega, F. Sukno, E. Lleida, A.F. Frangi, A. Miguel, L. Buera, E. Zacur, AV@CAR: a Spanish multichannel multimodal corpus for in-vehicle automatic audio-visual speech recognition., Proc. International Conference on Language Resources and Evaluation, 2004. pp. 763-767.

[36] A. Fernandez-Lopez, O. Martínez, F.M. Sukno, Towards estimating the upper bound of visual-speech recognition: the visual lip-reading feasibility database, Proc. International Conference on Automatic Face and Gesture Recognition, 2017. pp. 208-2015.